Entwicklerhandbuch

AWS DeepRacer

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS DeepRacer: Entwicklerhandbuch

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Die Handelsmarken und Handelsaufmachung von Amazon dürfen nicht in einer Weise in Verbindung mit nicht von Amazon stammenden Produkten oder Services verwendet werden, durch die Kunden irregeführt werden könnten oder Amazon in schlechtem Licht dargestellt oder diskreditiert werden könnte. Alle anderen Handelsmarken, die nicht Eigentum von Amazon sind, gehören den jeweiligen Besitzern, die möglicherweise zu Amazon gehören oder nicht, mit Amazon verbunden sind oder von Amazon gesponsert werden.

Table of Contents

Was ist AWS DeepRacer? 1
Die DeepRacer AWS-Konsole 1
Das DeepRacer AWS-Fahrzeug 2
Die DeepRacer AWS-Liga 2
Erfahren Sie mehr über Reinforcement Learning 3
Konzepte und Terminologie 4
Terminologie für Rennveranstaltungen 10
Funktionsweise 12
Bestärkendes Lernen 12
Aktionsraum und Belohnungsfunktion 15
Trainingsalgorhythmen 18
DeepRacer AWS-Arbeitsablauf 19
S imulated-to-real Leistungslücken 21
Erste Schritte 22
Trainiere dein erstes Modell 22
Trainieren Sie ein Reinforcement-Learning-Modell mit der DeepRacer AWS-Konsole 22
Geben Sie den Modellnamen und die Umgebung an 23
Wähle einen Renntyp und einen Trainingsalgorhythmus 23
Definieren Sie den Aktionsraum 25
Wählen Sie ein virtuelles Auto 30
Passen Sie Ihre Belohnungsfunktion an 30
Evaluieren Sie Modelle in der Simulation 33
Modelle trainieren und auswerten 39
Renntypen verstehen und Sensoren aktivieren 41
Wählen Sie Sensoren 42
Konfigurieren Sie Ihren Agenten für Schulungen 44
Maßgeschneiderte Schulungen für Zeitfahren 46
Maßgeschneidertes Training für Rennen zur Objektvermeidung 47
Maßgeschneidertes Training für head-to-bot Rennen 49
Modelle mit der DeepRacer AWS-Konsole trainieren und auswerten 51
Erstellen Sie Ihre Prämienfunktion 52
Erkunden Sie den Aktionsraum 55
Passen Sie Hyperparameter an 57
Untersuchen Sie den Fortschritt der Schulungsarbeiten 62

Klonen Sie ein trainiertes Modell 64
Evaluieren Sie Modelle in Simulationen 65
Optimieren Sie das Training für reale Umgebungen 65
Referenz zur Prämienfunktion 69
Eingabeparameter der Belohnungsfunktion 69
Beispiele für Belohnungsfunktionen 85
..... 90
Kopieren Sie Ihr DeepRacer AWS-Modell nach Amazon S3 90
Importieren Sie Ihr DeepRacer AWS-Modell in die Konsole 93
Fehlerbehebung 95
Bedienen Sie Ihr Fahrzeug 99
Lernen Sie Ihr Fahrzeug kennen 99
Inspizieren Sie Ihr Fahrzeug 100
Laden Sie die Batterien auf und installieren Sie sie 102
Testen Sie das Rechenmodul 104
Schalten Sie Ihr Gerät aus 105
LED-Indikatoren 106
Ersatzteile für Geräte 108
Richten Sie Ihr Fahrzeug ein 118
Machen Sie sich bereit für die Einrichtung von Wi-Fi 119
Richten Sie WLAN ein und aktualisieren Sie die Software 120
Starten Sie die Gerätekonsole 121
Kalibrieren Sie Ihr Fahrzeug 123
Laden Sie Ihr Modell hoch 133
Fahren Sie Ihr Fahrzeug 134
Fahren Sie Ihr DeepRacer AWS-Fahrzeug manuell 135
Fahren Sie Ihr DeepRacer AWS-Fahrzeug autonom 136
Überprüfen und verwalten Sie die Fahrzeugeinstellungen 138
Fahrzeugprotokolle anzeigen 143
Aktualisieren und Wiederherstellen Ihres DeepRacer AWS-Geräts 145
Überprüfen Sie die Softwareversion Ihres Geräts 145
Erstellen Sie das Ubuntu 20.04-Installationsmedium 146
..... 146
Voraussetzungen 146
Vorbereitung 147
Bereiten Sie ein bootfähiges USB-Laufwerk vor 147

Aktualisieren Sie das Gerät auf Ubuntu 20.04 159
Erstellen Sie Ihre physische Strecke 161
Materialien und Werkzeuge 161
Materialien, die Sie möglicherweise benötigen 161
Werkzeuge, die Sie möglicherweise benötigen 162
Verschaffen Sie sich Ihren Weg 162
Abmessungsanforderungen 163
Überlegungen zur Modellleistung 164
Schritte zum Aufbau des Tracks 165
Entwurfsvorlagen verfolgen 169
Vorlage für eine Rennstrecke von A bis Z (Basic) 170
Streckenvorlage für AWS DeepRacer Smile Speedway (Fortgeschrittene) 171
RL Speedway (Advanced) Streckenvorlage 172
Vorlage für einen Single-Turn-Track 173
Vorlage für eine S-Kurven-Spur 174
Loop-Track-Vorlage 175
Nehmen Sie an einem Rennen teil 178
Arten von Rennveranstaltungen 178
Teilnahme an einem online gesponserten oder von der AWS Community gesponserten Rennen 179
Nehmen Sie an einem Virtual Circuit-Rennen teil 179
Nehmen Sie an einem Community-Rennen teil 181
Nehmen Sie als Rennteilnehmer an einem DeepRacer AWS-Community-Rennen teil 182
Nehmen Sie an einem LIVE-Rennen teil 191
Organisieren Sie ein Rennen 195
..... 195
Erstelle einen Schnellstart für ein Rennen 195
Passe ein Rennen an 200
Führen Sie ein LIVE-Rennen durch 207
Ein LIVE-Rennen übertragen 213
Rollen der Organisatoren 213
Szenen des Senders 214
DeepRacer AWS-Szenenvorlagen 215
Verwalte ein Rennen 220
Organisieren Sie eine Veranstaltung 225
Was ist eine DeepRacer AWS-Veranstaltung? 225

Wie funktionieren Veranstaltungen und was ist zu erwarten 225
Was ist zu beachten, bevor Sie beginnen 226
Arten von DeepRacer AWS-Rennen 228
Bewährte Methoden 229
Erste Schritte mit Ihrer Veranstaltung 230
Beispiele für DeepRacer AWS-Ereignisse 230
Weitere Ressourcen 235
Mehrbenutzermodus 236
Admin-Setup 236
Stakeholder, die mehrere Benutzer verwenden 237
Schritt 1. Voraussetzungen für den DeepRacer AWS-Mehrbenutzermodus 238
Schritt 2: Aktiviere den Mehrbenutzer-Kontomodus 239
Schritt 3: Laden Sie die Teilnehmer ein, gesponsert zu werden 241
Schritt 4: Legen Sie Nutzungskontingente fest 241
Schritt 5: Nutzung überwachen 242
Nächste Schritte 244
Einrichtung für Teilnehmer 245
Voraussetzungen 245
Schritt 1. Melden Sie sich mit den AWS Anmeldeinformationen des Sponsoring-Kontos bei der Konsole an 246
Schritt 2. Erstelle ein AWS Spielerkonto oder melde dich an 246
Schritt 3. Passen Sie Ihr Profil an 247
Schritt 4. Modelle trainieren 248
Schritt 5. Gesponserte Nutzung anzeigen 248
Schritt 6: (Optional) Fordere zusätzliche gesponserte Stunden an 248
Tools für Pädagogen 249
Integrieren Sie AWS DeepRacer Student in den Unterricht 249
Erstellen Sie Community-Rennen für Schüler 249
Erstellen Sie ein Studentenrennen 250
Passen Sie ein Studentenrennen an 252
Organisieren Sie ein Studentenrennen 255
Sicherheit 258
Datenschutz 258
DeepRacerAWS-abhängige Services 259
Erforderliche IAM-Rollen 261
AWS Identity and Access Management 262

Zielgruppe ... 262
Authentifizierung mit Identitäten ... 263
Zugriff mithilfe von Richtlinien verwalten ... 267
Wie AWS DeepRacer funktioniert mit IAM ... 270
Beispiele für identitätsbasierte Richtlinien ... 277
AWS verwaltete Richtlinien ... 280
Serviceübergreifende Confused-Deputy-Prävention ... 284
Fehlerbehebung ... 286
Tagging ... 290
Hinzufügen, Anzeigen und Bearbeiten von Tags für eine neue Ressource ... 291
Fügen Sie Tags für eine vorhandene Ressource hinzu, zeigen Sie sie an und bearbeiten Sie sie ... 293
Beheben Sie häufig auftretende Probleme ... 296
So lösen Sie häufig auftretende Probleme mit AWS DeepRacer LIVE ... 296
Ich kann das Rennvideo auf der LIVE-Rennseite nicht sehen ... 296
Der Name eines Rennfahrers in der Rennwarteschlange ist rot ... 297
Ich führe ein LIVE-Rennen durch und kann die Rennfahrer nicht starten ... 298
Ich verwende einen Chrome- oder Firefox-Browser, habe aber immer noch Probleme, das LIVE-Rennen zu sehen ... 300
Warum kann ich über eine USB-Verbindung zwischen meinem Computer und dem Fahrzeug keine Verbindung zur Gerätekonsole herstellen? ... 301
So schalten Sie die Stromquelle des DeepRacer AWS-Rechenmoduls von der Batterie auf eine Steckdose um ... 306
So verwenden Sie ein USB-Flash-Laufwerk, um AWS mit Ihrem Wi-Fi-Netzwerk DeepRacer zu verbinden ... 307
Wie lade ich die Batterie des Antriebsmoduls des Fahrzeugs auf ... 312
Wie lade ich die Batterie des Rechenmoduls des Fahrzeugs auf ... 316
Mein Akku ist geladen, aber mein Fahrzeug bewegt sich nicht ... 316
Beheben Sie die Sperrung der Fahrzeugbatterie ... 319
So verhindern Sie das Aussperren der -Fahrzeugbatterie ... 320
So entsperren Sie DeepRacer AWS-Fahrzeugbatterien ... 321
So wickeln Sie ein Dell-Akkuanschlusskabel bei der Installation eines LiDAR-Sensors ein ... 323
So halten Sie die Verbindung Ihres Fahrzeugs aufrecht ... 328
So beheben Sie Probleme mit der Wi-Fi-Verbindung, wenn die Wi-Fi-LED-Anzeige Ihres Fahrzeugs blau blinkt, dann zwei Sekunden lang rot leuchtet und schließlich erlischt ... 329

Was bedeutet es, wenn die WLAN- oder Betriebs-LED-Anzeige des Fahrzeugs blau blinkt? ... 330
Wie kann ich über den Hostnamen eine Verbindung zur Gerätekonsole des Fahrzeugs herstellen? ... 330
Wie stelle ich über die IP-Adresse eine Verbindung zur Gerätekonsole des Fahrzeugs her ... 330
Wie erhalte ich die Mac-Adresse Ihres Geräts ... 331
Wie kann ich das Standardkennwort für den Gerätecontroller wiederherstellen ... 332
Wie aktualisiere ich mein Gerät manuell ... 333
Wie diagnostiziert und löst man häufig auftretende Probleme mit dem Gerätebetrieb ... 334
Warum zeigt der Videoplayer auf der Gerätekonsole den Videostream von der Kamera meines Fahrzeugs nicht an? ... 335
Warum bewegt sich mein DeepRacer AWS-Fahrzeug nicht? ... 335
Warum sehe ich das neueste Geräte-Update nicht? Wie erhalte ich das neueste Update? ... 336
Warum ist mein DeepRacer AWS-Fahrzeug nicht mit meinem Wi-Fi-Netzwerk verbunden? ... 336
Warum dauert das Laden der DeepRacer AWS-Gerätekonsolenseite lange? ... 337
Warum funktioniert ein Modell nicht gut, wenn es in einem DeepRacer AWS-Fahrzeug eingesetzt wird? ... 337
Dokumentverlauf ... 339
AWS Glossar ... 343
... cccxliv

Was ist AWS DeepRacer?

AWS DeepRacer ist ein vollautonomer Rennwagen im Maßstab 1:18, der durch Reinforcement-Learning angetrieben wird. Die Lösung umfasst die folgenden Komponenten:

- DeepRacer AWS-Konsole: Ein AWS-Service für Machine Learning zum Training und zur Evaluierung von Reinforcement-Learning-Modellen in einer dreidimensionalen simulierten Umgebung für autonomes Fahren.
- DeepRacer AWS-Fahrzeug: Ein ferngesteuertes Fahrzeug im Maßstab 1:18, das Inferenzen anhand eines trainierten DeepRacer AWS-Modells für autonomes Fahren durchführen kann.
- AWS DeepRacer League: Die weltweit erste globale, autonome Rennliga. Rennen Sie um Preise, Ruhm und die Chance, den Weltmeisterschaftspokal zu erreichen. Weitere Informationen finden Sie in den Allgemeinen Geschäftsbedingungen.

Themen

- Die DeepRacer AWS-Konsole
- Das DeepRacer AWS-Fahrzeug
- Die DeepRacer AWS-Liga
- Nutzen Sie AWS DeepRacer , um Reinforcement Learning zu erkunden
- DeepRacer AWS-Konzepte und -Terminologie

Die DeepRacer AWS-Konsole

Die DeepRacer AWS-Konsole ist eine grafische Benutzeroberfläche für die Interaktion mit dem DeepRacer AWS-Service. Sie können die Konsole verwenden, um ein Reinforcement-Learning-Modell zu trainieren und die Modellleistung im DeepRacer AWS-Simulator zu bewerten. In der Konsole können Sie auch ein trainiertes Modell für den Einsatz in Ihrem DeepRacer AWS-Fahrzeug für autonomes Fahren in einer physischen Umgebung herunterladen.

Zusammenfassend unterstützt die DeepRacer AWS-Konsole die folgenden Funktionen:

- Erstellen eines Trainingsjobs, um ein Reinforcement-Learning-Modell mit einer definierten Belohnungsfunktion, einem Optimierungsalgorithmus, einer Umgebung und Hyperparametern zu trainieren.

- Wählen Sie eine simulierte Strecke, um ein Modell mithilfe von SageMaker KI zu trainieren und zu bewerten.
- Klonen eines trainierten Modells, um das Training über die Anpassung der Hyperparameter zu verbessern und die Leistung Ihres Modells zu optimieren.
- Laden Sie ein trainiertes Modell für den Einsatz auf Ihrem DeepRacer AWS-Fahrzeug herunter, damit es in einer physischen Umgebung fahren kann.
- Einreichen Ihres Modells für ein virtuelles Rennen und Einordnung seiner Leistung im Vergleich zu anderen Modellen in einer virtuellen Rangliste.

Wenn Sie die DeepRacer AWS-Servicekonsole verwenden, fallen Gebühren auf der Grundlage Ihrer Nutzung zum Trainieren, Evaluieren und Speichern von Modellen an.

Um Ihnen den Einstieg zu erleichtern, DeepRacer bietet AWS DeepRacer AWS-Erstbenutzern ein kostenloses Kontingent. Dies ist genug Zeit, um Ihr erstes Modell zu trainieren und zu optimieren und in die AWS DeepRacer League einzusteigen. Die Einreichung eines Modells für die Teilnahme an einer virtuellen Veranstaltung der AWS DeepRacer League ist kostenlos.

Einzelheiten zu den Preisen finden Sie auf der DeepRacer AWS-Service-Detailseite.

Das DeepRacer AWS-Fahrzeug

Das DeepRacer AWS-Fahrzeug ist ein Wi-Fi-fähiges, physisches Fahrzeug, das mithilfe eines Reinforcement-Learning-Modells selbstständig auf einer physischen Strecke fahren kann.

- Sie können das Fahrzeug manuell steuern oder ein Modell bereitstellen, damit das Fahrzeug autonom fährt.
- Der autonome Modus führt über das Computing-Modul des Fahrzeugs Inferenzen aus. Inferenzen verwenden Bilder, die von der Frontkamera des Fahrzeugs aufgenommen werden.
- Eine WLAN-Verbindung ermöglicht das Herunterladen von Software auf das Fahrzeug. Über die Verbindung kann der Benutzer außerdem auf die Gerätekonsole zugreifen, um das Fahrzeug über einen Computer oder ein Mobilgerät zu steuern.

Die DeepRacer AWS-Liga

Die AWS DeepRacer League ist ein wichtiger Bestandteil von AWS DeepRacer. Die AWS DeepRacer League soll Gemeinschaft und Wettbewerb fördern.

Mit der AWS DeepRacer League können Sie Ihre ML-Fähigkeiten mit denen anderer DeepRacer AWS-Entwickler bei einer physischen oder virtuellen Rennveranstaltung vergleichen. Sie haben nicht nur die Möglichkeit, Preise und Erfolge zu gewinnen, sondern haben auch die Möglichkeit, Ihre Reinforcement-Learning-Modelle zu testen. Sie können sich mit anderen Teilnehmern messen, voneinander lernen und sich gegenseitig inspirieren. Wenn Sie für Ihre Leistung in der AWS DeepRacer League Erfolge erzielen, können Sie diese mit Ihrer Community in den sozialen Medien teilen. Weitere Informationen finden Sie in den Allgemeinen Geschäftsbedingungen.

Nehmen Sie an einem Rennen teil oder lernen Sie, wie Sie ein Modell in der Liga schulen.

Nutzen Sie AWS DeepRacer , um Reinforcement Learning zu erkunden

Das Reinforcement Learning, insbesondere das Deep Reinforcement Learning, hat sich für eine Vielzahl von Problemen aus dem Bereich der autonomen Entscheidungsfindung bewährt. Es gibt Anwendungen im Finanzhandel, in der Kühlung von Rechenzentren, in der Flottenlogistik und im autonomen Rennsport, um nur einige zu nennen.

Reinforcement Learning hat das Potenzial, Probleme aus der Praxis zu lösen. Aufgrund seines umfangreichen technologischen Umfangs und seiner Tiefe weist es jedoch eine steile Lernkurve auf. Für Experimente in der realen Welt müssen Sie ein physikalisches Mittel konstruieren, z. B. einen autonomen Rennwagen. Es erfordert auch, dass Sie eine physische Umgebung sichern, z. B. eine Fahrstrecke oder eine öffentliche Straße. Die Einrichtung einer entsprechenden Umgebung kann kostspielig, gefährlich und zeitaufwendig sein. Es ist also mehr erforderlich als bloßes Wissen zum Thema Reinforcement Learning.

Um die Lernkurve zu verkürzen, DeepRacer vereinfacht AWS den Prozess auf drei Arten:

- Bietet step-by-step Unterstützung bei der Schulung und Bewertung von Reinforcement-Learning-Modellen. Die Anleitung umfasst vordefinierte Umgebungen, Zustände und Aktionen sowie anpassbare Belohnungsfunktionen.
- Bereitstellung eines Simulators zur Emulierung von Interaktionen zwischen einem virtuellen Agenten und einer virtuellen Umgebung.
- Verwendung eines DeepRacer AWS-Fahrzeugs als physischen Agenten. Verwenden Sie das Fahrzeug, um ein trainiertes Modell in einer physischen Umgebung zu evaluieren. Dies ähnelt stark einem realen Anwendungsfall.

Wenn Sie ein erfahrener Experte für maschinelles Lernen sind, ist AWS DeepRacer eine willkommene Gelegenheit, Reinforcement-Learning-Modelle für autonomes Fahren sowohl in virtuellen als auch in physischen Umgebungen zu entwickeln. Zusammenfassend lässt sich sagen, DeepRacer dass Sie mithilfe von AWS Reinforcement-Learning-Modelle für autonomes Fahren mit den folgenden Schritten erstellen können:

1. Trainieren eines individuellen Reinforcement-Learning-Modells für den autonomen Rennsport. Verwenden Sie dazu die in SageMaker KI integrierte DeepRacer AWS-Konsole.
2. Verwenden Sie den DeepRacer AWS-Simulator, um ein Modell zu evaluieren und autonome Rennen in einer virtuellen Umgebung zu testen.
3. Stellen Sie ein trainiertes Modell für DeepRacer AWS-Modellfahrzeuge bereit, um autonome Rennen in einer physischen Umgebung zu testen.

DeepRacer AWS-Konzepte und -Terminologie

AWS DeepRacer baut auf den folgenden Konzepten auf und verwendet die folgende Terminologie.

DeepRacer AWS-Dienst

AWS DeepRacer ist ein AWS-Service für Machine Learning zur Erforschung von Reinforcement Learning, der sich auf autonome Rennen konzentriert. Der DeepRacer AWS-Service unterstützt die folgenden Funktionen:

1. Training eines Verstärkungslernmodells in der Cloud.
2. Evaluieren Sie ein trainiertes Modell in der DeepRacer AWS-Konsole.
3. Einreichen eines trainierten Modells für ein virtuelles Rennen und Aufnahme von qualifizierten Modellen in die Rangliste der Veranstaltung.
4. Klonen Sie ein trainiertes Modell, um das Training fortzusetzen und die Leistung zu verbessern.
5. Laden Sie die trainierten Modellartefakte herunter, um sie auf ein DeepRacer AWS-Fahrzeug hochzuladen.
6. Platzieren des Fahrzeugs auf einer physischen Strecke für autonomes Fahren und bewerten der Leistung des Modells für die Praxis.
7. Eliminieren unnötiger Gebühren durch das Löschen nicht benötigter Modelle.

AWS DeepRacer

„AWS DeepRacer" kann sich auf drei verschiedene Fahrzeuge beziehen:

- Der virtuelle Rennwagen kann die Form des ursprünglichen DeepRacer AWS-Geräts, des Evo-Geräts oder verschiedener digitaler Prämien annehmen, die durch die Teilnahme an AWS DeepRacer League Virtual Circuit-Rennen verdient werden können. Sie können das virtuelle Auto auch personalisieren, indem Sie seine Farbe ändern.
- Das ursprüngliche DeepRacer AWS-Gerät ist ein physisches Modellauto im Maßstab 1:18. Es verfügt über eine montierte Kamera und ein Onboard-Rechenmodul. Das Rechenmodul sorgt für die Inferenz für das Fahren entlang der Strecke. Das Rechenmodul und das Fahrzeugchassis werden von speziellen Akkus versorgt, die als Rechenmodul-Akku bzw. Antriebsmodul-Akku bezeichnet werden.
- Das AWS DeepRacer Evo-Gerät ist das Originalgerät mit einem optionalen Sensorkit. Das Kit umfasst eine zusätzliche Kamera und LIDAR (Light Detection and Ranging), mit denen das Fahrzeug Objekte hinter und neben sich selbst erkennen kann. Das Kit enthält auch eine neue Schale.

Bestärkendes Lernen

Reinforcement Learning ist eine Methode des maschinellen Lernens, die sich auf die autonome Entscheidungsfindung eines Agenten konzentriert, um bestimmte Ziele durch Interaktionen mit einer Umgebung zu erreichen. Beim bestärkenden Lernen wird der Lerneffekt durch Versuch und Fehlschlag erreicht und das Training erfordert keine beschrifteten Eingaben. Das Training basiert auf der Belohnungshypothese, die davon ausgeht, dass alle Ziele erreicht werden können, indem eine future Belohnung nach Aktionssequenzen maximiert wird. Beim bestärkenden Lernen ist die Gestaltung der Belohnungsfunktion entscheidend. Besser ausgearbeitete Belohnungsfunktionen führen zu besseren Entscheidungen des Agenten.

Für das autonome Rennen ist der Agent ein Fahrzeug. Die Umgebung umfasst Reiserouten und Verkehrsbedingungen. Das Ziel besteht darin, dass das Fahrzeug schnell und ohne Unfälle sein Ziel erreicht. Belohnungen sind Punkte, die dazu dienen, eine sichere und schnelle Reise zum Ziel zu ermöglichen. Die Punktzahl bestraft gefährliches und verschwenderisches Fahren.

Zur Unterstützung des Lernens während des Trainings muss der Lernagent manchmal Aktionen durchführen können, die nicht zu Belohnungen führen. Dies ist der Kompromiss zwischen Erkundung und Nutzen. Er hilft dabei, die Wahrscheinlichkeit zu verringern oder zu beseitigen, dass der Agent zu falschen Zielen irregeführt wird.

Eine formalere Definition finden Sie unter Reinforcement Learning (dt: Bestärkendes Lernen) auf Wikipedia.

Reinforcement-Learning-Modell

Bei einem Modell des Reinforcement-Learnings handelt es sich um eine Umgebung, in der ein Agent handelt. Dabei werden drei Dinge festgelegt: die Zustände, die der Agent hat, die Aktionen, die der Agent ergreifen kann, und die Belohnungen, die er erhält, wenn er Maßnahmen ergreift. Die Strategie, mit der der Agent über seine Maßnahmen entscheidet, wird als Richtlinie bezeichnet. Die Richtlinie nimmt den Umgebungszustand als Eingabe und gibt die jeweiligen Aktionen aus. Beim Reinforcement-Learning wird die Politik häufig durch ein tiefes neuronales Netzwerk repräsentiert. Wir bezeichnen dies als das Modell des verstärkten Lernens. Jeder Trainingsauftrag erzeugt ein Modell. Auch bei vorzeitiger Beendigung des Trainingsauftrags kann ein Modell erstellt werden. Ein Modell ist unveränderlich, d. h. es kann nach der Erstellung nicht mehr geändert oder überschrieben werden.

DeepRacer AWS-Simulator

Der DeepRacer AWS-Simulator ist eine virtuelle Umgebung zur Visualisierung, Schulung und Evaluierung von DeepRacer AWS-Modellen.

DeepRacer AWS-Fahrzeug

Siehe AWS DeepRacer.

DeepRacer AWS-Auto

Bei diesem DeepRacer AWS-Fahrzeugtyp handelt es sich um ein Modellauto im Maßstab 1:18.

Leaderboard

Eine Bestenliste ist eine Rangliste der Leistungen von DeepRacer AWS-Fahrzeugen bei einer Rennveranstaltung der AWS DeepRacer League. Das Rennen kann ein virtuelles Event sein, das in der simulierten Umgebung durchgeführt wird, oder ein physisches Event, das in einer realen Umgebung durchgeführt wird. Die Leistungskennzahl hängt vom Renntyp ab. Dabei kann es sich um die schnellste Rundenzeit, Gesamtzeit oder durchschnittliche Rundenzeit handeln, die von DeepRacer AWS-Benutzern eingereicht wurde, die ihre trainierten Modelle auf einer Strecke bewertet haben, die mit der angegebenen Rennstrecke identisch oder ähnlich ist.

Wenn ein Fahrzeug drei aufeinander folgende Runden absolviert hat, kann es auf dem Leaderboard eingetragen werden. Die durchschnittliche Rundenzeit der ersten drei aufeinanderfolgenden Runden wird an das Leaderboard übermittelt.

Frameworks für maschinelles Lernen

Frameworks für maschinelles Lernen sind Softwarebibliotheken, die zur Erstellung von Algorithmen für maschinelles Lernen verwendet werden. Zu den unterstützten Frameworks für AWS DeepRacer gehört Tensorflow.

Richtliniennetzwerk

Ein Richtliniennetzwerk ist ein neuronales Netzwerk, das trainiert wird. Das Richtliniennetzwerk nimmt Videobilder als Eingabe und prognostiziert die nächsten Aktionen für den Agenten. Je nach Algorithmus kann es auch den Wert des aktuellen Zustands des Agenten auswerten.

Optimierungsalgorithmus

Ein Optimierungsalgorithmus ist der Algorithmus, mit dem ein Modell trainiert wird. Beim überwachten Training wird der Algorithmus optimiert, indem eine Verlustfunktion mit einer bestimmten Strategie zur Aktualisierung von Gewichtungen minimiert wird. Beim bestärkenden Lernen wird der Algorithmus optimiert, indem die erwarteten zukünftigen Belohnungen mit einer bestimmten Belohnungsfunktion maximiert werden.

Neuronales Netz

Ein neuronales Netzwerk (auch bekannt als künstliches neuronales Netzwerk) ist eine Sammlung verbundener Einheiten oder Knoten, die verwendet werden, um ein Informationsmodell auf der Grundlage biologischer Systeme aufzubauen. Jeder Knoten wird als künstliches Neuron bezeichnet und ahmt ein biologisches Neuron nach, indem er eine Eingabe (Stimulus) empfängt, aktiviert wird, wenn das Eingangssignal stark genug ist (Aktivierung), und eine Ausgabe erzeugt, die auf der Eingabe und Aktivierung basiert. Neuronale Netzwerke werden häufig beim maschinellen Lernen verwendet, da ein neuronales Netzwerk als universelle Näherung für jegliche Funktion dienen kann. Maschinen das Lernen beizubringen bedeutet, die optimale Funktionsapproximation für die gegebene Eingabe und Ausgabe zu finden. Beim tief gehenden bestärkenden Lernen stellt das neuronale Netzwerk die Strategie dar und wird oft als „Richtliniennetzwerk“ bezeichnet. Die Schulung des politischen Netzwerks läuft darauf hinaus, schrittweise Schritte durchzuführen, die das Sammeln von Erfahrungen auf der Grundlage der aktuellen Politik beinhalten, und anschließend das politische Netzwerk anhand der neu gewonnenen Erfahrungen zu optimieren. Der Prozess wird fortgesetzt, bis bestimmte Leistungsmetriken die geforderten Kriterien erfüllen.

Hyperparameter

Hyperparameter sind vom Algorithmus abhängige Variablen, die die Leistung des Trainings neuronaler Netze steuern. Ein Beispiel für einen Hyperparameter ist die Lernrate, die steuert,

wie viele neue Erfahrungen bei jedem Schritt beim Lernen berücksichtigt werden. Eine höhere Lernrate führt zu einem schnelleren Training, kann jedoch zu einer geringeren Qualität des trainierten Modells führen. Hyperparameter sind empirischer Natur und erfordern eine systematische Abstimmung für jedes Training.

DeepRacer AWS-Track

Eine Strecke ist ein Pfad oder eine Strecke, auf der ein DeepRacer AWS-Fahrzeug fährt. Die Strecke kann entweder in einer simulierten Umgebung oder in einer realen, physischen Umgebung existieren. Sie verwenden eine simulierte Umgebung, um ein DeepRacer AWS-Modell auf einer virtuellen Strecke zu trainieren. Die DeepRacer AWS-Konsole stellt virtuelle Tracks zur Verfügung. Sie verwenden eine reale Umgebung, um ein DeepRacer AWS-Fahrzeug auf einer physischen Strecke zu betreiben. Die AWS DeepRacer League bietet Veranstaltungsteilnehmern physische Wettkampfstrecken. Sie müssen Ihre eigene physische Strecke erstellen, wenn Sie Ihr DeepRacer AWS-Fahrzeug in einer anderen Situation einsetzen möchten. Weitere Informationen darüber, wie Sie Ihre eigene Strecke erstellen können, finden Sie unter Erstellen Sie Ihre physische Strecke.

Belohnungsfunktion

Eine Belohnungsfunktion ist ein Algorithmus innerhalb eines Lernmodells, der dem Agenten mitteilt, ob die ausgeführte Aktion zu folgenden Ergebnissen geführt hat:

- Ein gutes Ergebnis, das verstärkt werden sollte.
- Ein neutrales Ergebnis.
- Ein schlechtes Ergebnis, das vermieden werden sollte.

Die Belohnung ist ein wichtiger Bestandteil des bestärkenden Lernens. Sie bestimmt das Verhalten, das der Agent erlernt, indem sie bestimmte Aktionen anderen gegenüber bevorzugt. Der Benutzer stellt die Belohnungsfunktion mit Python bereit. Diese Belohnungsfunktion wird von einem Optimierungsalgorithmus zum Training des Bestärkungslernmodells verwendet.

Erfahrungsepisode

Eine Erlebnisepisode ist ein Zeitraum, in dem der Agent Erlebnisse als Trainingsdaten aus der Umgebung sammelt, indem er von einem bestimmten Startpunkt bis zum Abschluss des Tracks läuft oder von der Strecke abweicht. Episoden können unterschiedliche Längen haben. Dieser Vorgang wird auch als Episode oder erlebnisgenerierende Episode bezeichnet.

Erfahrungsiteration

Bei der Erfahrungswiederholung (auch bekannt als erfahrungsgenerierende Iteration) handelt es sich um eine Reihe aufeinanderfolgender Erfahrungen zwischen den einzelnen Policy-Iterationen, bei denen die Gewichtungen der Policy-Netzwerke aktualisiert werden. Am Ende jeder Erfahrungsiteration werden die erfassten Episoden einem Erfahrungs-Replay oder Puffer hinzugefügt. Die Größe kann in einem der Hyperparameter für das Training festgelegt werden. Das neuronale Netz wird anhand zufälliger Beispiele von Erfahrungen aktualisiert.

Richtlinieniteration

Die Policy-Iteration (auch als Policy-Update-Iteration bezeichnet) besteht aus einer beliebigen Anzahl von Durchläufen der nach dem Zufallsprinzip ausgewählten Trainingsdaten, um die Gewichtungen des neuronalen Netzwerks während des Gradientenaufstiegs zu aktualisieren. Ein einziger Durchlauf der Trainingsdaten zur Aktualisierung der Gewichte wird auch als Epoche bezeichnet.

Trainingsauftrag

Ein Trainingsjob ist ein Workload, der ein Reinforcement-Learning-Modell trainiert und trainierte Modellartefakte erzeugt, anhand derer Inferenzen ausgeführt werden können. Jeder Trainingsauftrag enthält zwei Unterprozesse:

1. Starten des Agenten zum Befolgen der aktuellen Richtlinie. Der Agent untersucht die Umgebung in einer Reihe von Episoden und erstellt Trainingsdaten. Diese Datengenerierung ist selbst ein iterativer Prozess.
2. Anwenden der neuen Trainingsdaten zur Berechnung neuer Richtliniengradienten. Aktualisieren der Netzwerk-Gewichtungen und Fortsetzen des Trainings. Wiederholen von Schritt 1, bis eine Stopp-Bedingung erfüllt ist.

Jeder Trainingsauftrag erstellt ein trainiertes Modell und gibt die Modellartefakte zu einem bestimmten Datenspeicher aus.

Auswertungsauftrag

Ein Evaluierungsjob ist ein Workload, der die Leistung eines Modells testet. Die Leistung wird anhand von bestimmten Metriken gemessen, nachdem der Trainingsauftrag abgeschlossen ist. Die standardmäßige DeepRacer AWS-Leistungskennzahl ist die Fahrzeit, die ein Agent benötigt, um eine Runde auf einer Strecke zurückzulegen. Eine weitere Metrik ist der Prozentsatz der abgeschlossenen Runde.

Terminologie für Rennveranstaltungen

DeepRacer AWS-Rennveranstaltungen verwenden die folgenden Konzepte und Begriffe.

Liga/Wettbewerb

Im Zusammenhang mit Veranstaltungen der AWS DeepRacer League beziehen sich die Begriffe Liga und Wettbewerb auf die Wettbewerbsstruktur. AWS sponsert die AWS DeepRacer League, was bedeutet, dass wir sie besitzen, entwerfen und betreiben. Ein Wettbewerb hat ein Start- und Enddatum.

Saison

Ein Wettbewerb kann sich in den folgenden Jahren wiederholen. Wir rufen die verschiedenen Saison aus (zum Beispiel die Saison 2019 oder 2020). Regeln können sich von Saison zu Saison ändern, sind aber üblicherweise innerhalb einer Saison konsistent. Die Allgemeinen Geschäftsbedingungen für die AWS DeepRacer League können von Saison zu Saison variieren.

Der virtuelle Circuit

Der Virtual Circuit bezieht sich auf die Rennen, die während der DeepRacer AWS-League-Saison in der DeepRacer AWS-Konsole AWS stattfinden, gesponsert werden.

Ereignis

Gemäß den Regeln ist eine Veranstaltung eine Veranstaltung in der AWS DeepRacer League, an der Sie an einem Rennen teilnehmen können. Eine Veranstaltung hat ein Start- und Enddatum. Virtual Circuit-Veranstaltungen dauern in der Regel einen Monat. In einer Saison kann es viele Events geben, und einige Regeln — wie wir die Teilnehmer einer Veranstaltung einstufen, wer gewinnt und was danach passiert — können sich ändern.

Renntyp

Alle Rennfahrer können an Zeitfahren (TT), Rennen zur Objektvermeidung (OA) oder (H2B) teilnehmen. head-to-bot Jeder Renntyp gibt die Anzahl der Runden und die Rangfolge der Rennfahrer an.

Nationale Saisonwertung

Eine nationale Saisonwertung bezieht sich auf die Bestenliste eines Rennfahrers unter anderen Rennfahrern in seinem Land. Alle Rennfahrer können in monatlichen virtuellen Rennen gegen andere Rennfahrer in ihrem Land antreten.

Regionale Saisonwertung

Eine regionale Saisonwertung bezieht sich auf die Bestenliste eines Rennfahrers im Vergleich zu anderen Rennfahrern in seiner Region.

Weltmeisterschaft

Die monatliche Virtual Circuit-Bestenliste der AWS DeepRacer League ist nach Ländern und Regionen unterteilt. Die besten Rennfahrer jeder Region haben die Möglichkeit, sich für die Weltmeisterschaften bei AWS re:Invent zu qualifizieren. Weitere Informationen finden Sie in den Allgemeinen Geschäftsbedingungen.

So DeepRacer funktioniert AWS

Ein DeepRacer AWS-Fahrzeug ist ein Fahrzeug im Maßstab 1:18, das autonom auf einer Strecke fahren oder gegen ein anderes Fahrzeug antreten kann. Das Fahrzeug kann mit verschiedenen Sensoren ausgestattet werden, darunter eine Frontkamera, Stereokameras, Radargeräte oder eine LiDAR. Die Sensoren erfassen Daten über die Umgebung, in der das Fahrzeug operiert. Unterschiedliche Sensoren bieten Anzeigen in verschiedenen Maßstäben.

AWS DeepRacer verwendet Reinforcement-Learning, um autonomes Fahren für das DeepRacer AWS-Fahrzeug zu ermöglichen. Um dies zu erreichen, trainieren und evaluieren Sie ein Reinforcement-Learning-Modell in einer virtuellen Umgebung mit einer simulierten Strecke. Nach dem Training laden Sie die trainierten Modellartefakte auf Ihr DeepRacer AWS-Fahrzeug hoch. Sie können das Fahrzeug dann in einer physischen Umgebung mit einer echten Strecke autonom fahren lassen.

Das Trainieren eines Reinforcement-Learning-Modells kann eine Herausforderung sein, insbesondere wenn Sie neu auf diesem Gebiet sind. AWS DeepRacer vereinfacht den Prozess, indem es die erforderlichen Komponenten zusammenführt und easy-to-follow assistentenähnliche Aufgabenvorlagen bereitstellt. Es ist jedoch hilfreich, ein gutes Verständnis der Grundlagen der in AWS implementierten Reinforcement-Learning-Schulungen zu haben DeepRacer.

Themen

- Verstärktes Lernen in AWS DeepRacer
- DeepRacer AWS-Aktionsraum und Belohnungsfunktion
- DeepRacerAWS-Schulungsalgorithmen
- Arbeitsablauf DeepRacer für AWS-Lösungen
- S imulated-to-real Leistungslücken

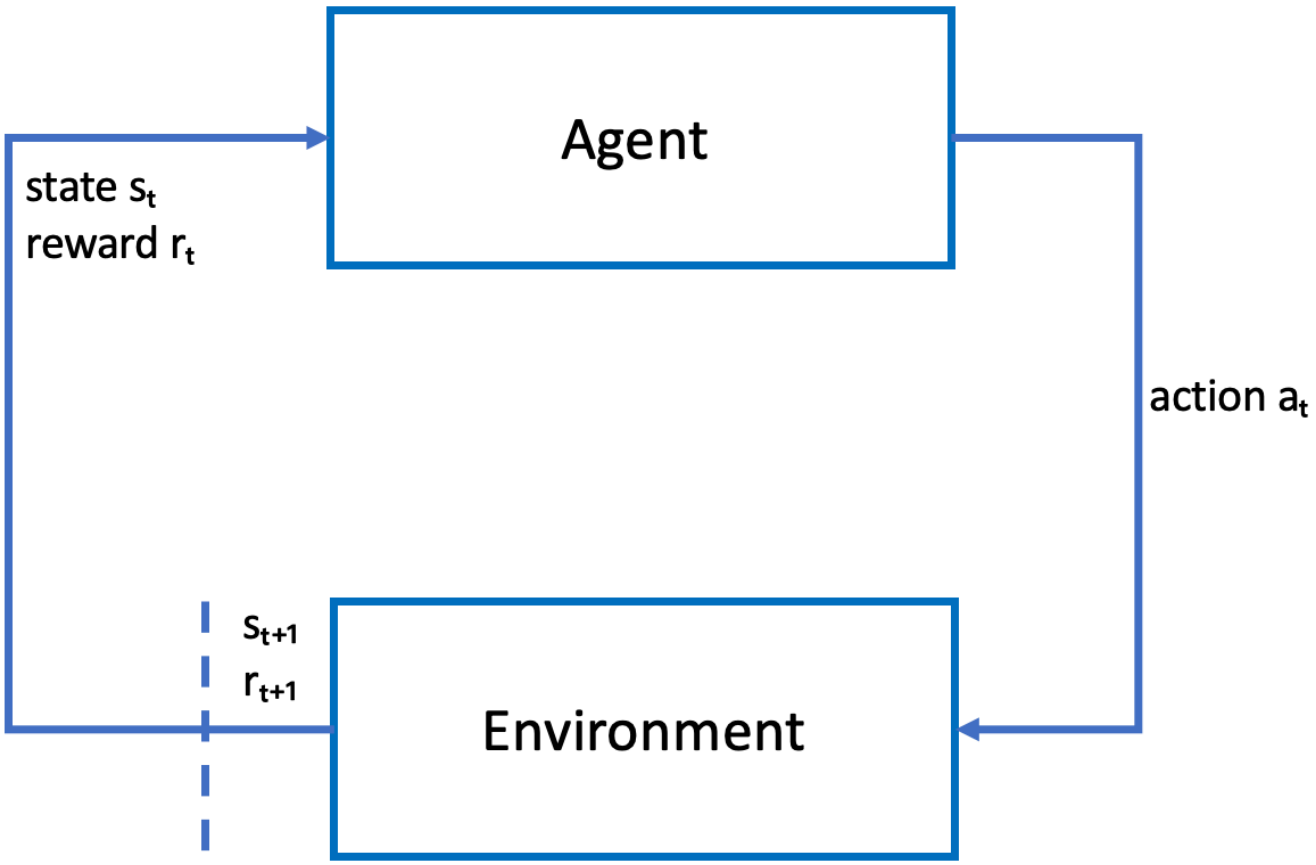
Verstärktes Lernen in AWS DeepRacer

Beim Reinforcement-Learning interagiert ein Agent, z. B. ein physisches oder virtuelles DeepRacer AWS-Fahrzeug, mit dem Ziel, ein angestrebtes Ziel zu erreichen, mit einer Umgebung, um die Gesamtvergütung des Agenten zu maximieren. Der Agent führt eine Aktion aus. Dabei wird er von einer Strategie (Policy) angeleitet. Die Aktion findet aus einen gegebenen Umgebungszustand statt und führt zu einem neuen Zustand. Mit jeder Aktion ist eine direkte Belohnung verbunden.

Die Belohnung ist ein Maß für die Zweckmäßigkeit der Aktion. Diese direkte Belohnung wird als Feedback der Umwelt betrachtet.

Das Ziel des Reinforcement-Learnings in AWS DeepRacer besteht darin, die optimale Richtlinie in einer bestimmten Umgebung zu erlernen. Das Lernen ist ein iterativer Prozess aus Versuchen und Fehlern. Der Agent führt eine zufällige Anfangsaktion aus, um in einen neuen Zustand zu gelangen. Dann wiederholt der Agent den Schritt vom neuen Zustand zum nächsten. Im Laufe der Zeit entdeckt der Agent Aktionen, die zu den größten langfristigen Belohnungen führen. Die Interaktion des Agenten von einem Ausgangszustand zu einem Endzustand wird als Episode bezeichnet.

Die folgende Skizze veranschaulicht diesen Lernprozess:



Der Agent ist ein neuronales Netzwerk, das eine Funktion zur Annäherung an die Strategie des Agenten darstellt. Das Bild der Frontkamera des Fahrzeugs stellt den Umgebungszustand dar. Die Aktion des Agenten wird durch die Geschwindigkeit und den Lenkwinkel des Agenten definiert.

Der Agent erhält positive Belohnungen, wenn er auf der Strecke bleibt und das Rennen beenden kann. Er erhält eine negative Belohnung für das Verlassen der Strecke. Zum Beginn einer Episode

befindet sich der Agent irgendwo auf der Rennstrecke. Sie endet, wenn der Agent entweder von der Strecke abkommt oder eine Runde absolviert.

> **Note**
>
> Streng genommen bezieht sich der Umgebungszustand auf alles, was für das Problem relevant ist. Zum Beispiel die Position des Fahrzeugs auf der Strecke sowie die Form der Strecke. Das Bild, das durch die an der Fahrzeugfront montierte Kamera übertragen wird, erfasst nicht den gesamten Zustand der Umgebung. Aus diesem Grund wird davon ausgegangen, dass die Umgebung nur teilweise beobachtet wurde, und die Informationen, die dem Agenten übermittelt werden, werden als Beobachtung und nicht als Zustand bezeichnet. Der Einfachheit halber verwenden wir in dieser Dokumentation die Begriffe Zustand und Beobachtung austauschbar.

Das Training des Agenten in einer simulierten Umgebung hat folgende Vorteile:

- Die Simulation kann abschätzen, wie viel Fortschritt der Agent gemacht hat und wann er von der Strecke abkommt. So kann sie eine Belohnung berechnen.
- Die Simulation entlastet den Entwickler/Trainer. Er muss das Fahrzeug nicht wie bei einer physischen Strecje bei jedem Verlassen der Strecke neu platzieren.
- Die Simulation kann das Training beschleunigen.
- Die Simulation ermöglicht eine bessere Kontrolle der Umgebungsbedingungen, z. B. die Auswahl verschiedener Strecken, Hintergründe und Fahrzeugbedingungen.

Die Alternative zum Reinforcement Learning ist das überwachte Lernen (auch Imitationslernen oder "Modelllernen"). Hier wird ein in einer bestimmten Umgebung gesammelter bekannter Datenbestand (aus [Bild, Aktion]-Tupeln) zum Training des Agenten verwendet. Modelle, die per Imitationslernen trainiert werden, können für das autonome Fahren eingesetzt werden. Sie funktionieren nur dann gut, wenn die Bilder der Kamera den Bildern im Trainingsdatenbestand ähnlich sehen. Für eine zuverlässige Fahrweise muss der Trainingsdatenbestand umfassend sein. Im Gegensatz dazu erfordert das Reinforcement Learning keinen so großen Erfassungsaufwand. Bei diesem Ansatz kann vollständig in der Simulation trainiert werden. Da das Reinforcement Learning mit zufälligen Aktionen startet, lernt der Agent eine Vielzahl von Umgebungs- und Streckenbedingungen kennen. Dies macht das trainierte Modell robust.

DeepRacer AWS-Aktionsraum und Belohnungsfunktion

Aktionsraum

Beim Reinforcement-Learning wird die Gesamtheit aller gültigen Aktionen oder Wahlmöglichkeiten, die einem Agenten bei der Interaktion mit einer Umgebung zur Verfügung stehen, als Aktionsraum bezeichnet. In der DeepRacer AWS-Konsole können Sie Agenten entweder in einem diskreten oder kontinuierlichen Aktionsbereich schulen.

Diskreter Aktionsraum

Ein diskreter Aktionsraum stellt alle möglichen Aktionen eines Agenten für jeden Zustand in einer endlichen Menge dar. Für AWS bedeutet dies DeepRacer, dass das neuronale Netzwerk des Agenten für jede inkrementell unterschiedliche Umgebungssituation eine Geschwindigkeit und Richtung für das Fahrzeug auswählt, basierend auf den Eingaben seiner Kamera (s) und des (optionalen) LiDAR-Sensors. Die Auswahl ist auf eine Gruppe von vordefinierten Kombinationen von Lenkwinkel- und Drosselklappenwerten beschränkt.

Ein DeepRacer AWS-Fahrzeug in einem diskreten Aktionsraum, das sich einer Kurve nähert, kann wählen, ob es beschleunigen oder bremsen und nach links, rechts oder geradeaus abbiegen möchte. Diese Aktionen sind als eine Kombination aus Lenkwinkel und Geschwindigkeit definiert, wodurch ein Optionsmenü (0-9) für den Agenten erstellt wird. Beispielsweise könnte 0 für -30 Grad und 0,4 m/s stehen, 1 für -30 Grad und 0,8 m/s, 2 für -15 Grad und 0,4 m/s, 3 für -15 Grad und 0,8 m/s usw. bis 9. Negative Grade drehen das Auto nach rechts, positive Grade drehen das Auto nach links und 0 hält die Räder gerade.

Der DeepRacer standardmäßige AWS-Bereich für diskrete Aktionen enthält die folgenden Aktionen:

DeepRacer AWS-Standardbereich für diskrete Aktionen

Aktionsnummer	Lenken	Geschwindigkeit
0	-30 Grad	0,4 m/s
1	0,4 m/s	0,8 m/s
2	-15 Grad	0,4 m/s
3	-15 Grad	0,8 m/s

Aktionsnummer	Lenken	Geschwindigkeit
4	0 Grad	0,4 m/s
5	0 Grad	0,8 m/s
6	15 Grad	0,4 m/s
7	15 Grad	0,8 m/s
8	30 Grad	0,4 m/s
9	30 Grad	0,8 m/s

Kontinuierlicher Aktionsraum

Ein kontinuierlicher Aktionsbereich ermöglicht es dem Agenten, für jeden Status eine Aktion aus einem Wertebereich auszuwählen. Genau wie bei einem diskreten Aktionsraum bedeutet dies, dass das neuronale Netzwerk des Agenten für jede inkrementell unterschiedliche Umgebungssituation eine Geschwindigkeit und Richtung für das Auto auf der Grundlage der Eingaben seiner Kamera (s) und des (optionalen) LiDAR-Sensors auswählt. In einem kontinuierlichen Aktionsraum können Sie jedoch den Bereich der Optionen definieren, aus denen der Agent seine Aktion auswählt.

In diesem Beispiel kann das DeepRacer AWS-Auto in einem Raum mit kontinuierlicher Aktion, das sich einer Kurve nähert, eine Geschwindigkeit von 0,75 m/s bis 4 m/s wählen und nach links, rechts oder geradeaus abbiegen, indem es einen Lenkwinkel von -20 bis 20 Grad wählt.

Diskret oder kontinuierlich

Der Vorteil eines kontinuierlichen Aktionsraums besteht darin, dass Sie Belohnungsfunktionen schreiben können, mit denen Modelle trainiert werden, um an bestimmten Punkten auf einer Strecke Anreize für Geschwindigkeits- und Lenkaktionen zu schaffen und so die Leistung zu optimieren. Die Auswahl aus einer Reihe von Aktionen ermöglicht auch sanfte Änderungen der Geschwindigkeit und der Lenkwerte, die bei einem gut trainierten Modell unter realen Bedingungen zu besseren Ergebnissen führen können.

Wenn Sie die Auswahl eines Agenten auf eine begrenzte Anzahl vordefinierter Aktionen beschränken, liegt es in Ihrer Verantwortung, die Auswirkungen dieser Aktionen zu verstehen und

sie auf der Grundlage der Umgebung (Strecke, Rennformat) und Ihrer Belohnungsfunktionen zu definieren. In einer Einstellung mit kontinuierlichem Aktionsbereich lernt der Agent jedoch, anhand der Min-/Max-Grenzen, die Sie im Training festlegen, die optimalen Werte für Geschwindigkeit und Lenkung auszuwählen.

Obwohl es die bessere Option zu sein scheint, dem Modell eine Reihe von Werten zur Auswahl zu geben, muss der Agent länger trainieren, um zu lernen, die optimalen Aktionen auszuwählen. Der Erfolg hängt auch von der Definition der Belohnungsfunktion ab.

Belohnungsfunktion

Während der Agent die Umgebung erkundet, lernt der Agent eine Wertfunktion kennen. Die Wertfunktion hilft Ihrem Agenten, anhand der Beobachtung der Umgebung zu beurteilen, wie gut eine ergriffene Maßnahme ist. Die Wertfunktion verwendet die Belohnungsfunktion, die Sie in die DeepRacer AWS-Konsole schreiben, um die Aktion zu bewerten. Beispiel: Bei der Funktion „Follow the center line“ in der DeepRacer AWS-Konsole würde eine gute Aktion den Agenten in der Mitte des Tracks halten und höher bewertet werden als eine schlechte Aktion, wodurch der Agent von der Mitte des Tracks wegbewegt würde.

Im Laufe der Zeit hilft die Wertfunktion dem Agenten dabei, sich mit den Richtlinien vertraut zu machen, die die Gesamtprämie erhöhen. Die optimale oder beste Richtlinie würde die Zeit, die der Agent mit der Erkundung der Umgebung verbringt, mit der Zeit, die er damit verbringt, das, was die Richtlinie durch Erfahrung gelernt hat, auszunutzen oder optimal zu nutzen, abwägen.

Im Beispiel der DeepRacer AWS-Beispielbelohnungsfunktion „Follow the center line“ führt der Agent zunächst zufällige Aktionen durch, um die Umgebung zu erkunden, was bedeutet, dass er nicht besonders gut darin ist, in der Mitte des Tracks zu bleiben. Mit der Zeit beginnt der Agent zu lernen, welche Aktionen ihn in der Nähe der Mittellinie halten. Wenn er dies jedoch tut, indem er weiterhin zufällige Aktionen ausführt, wird es lange dauern, bis er gelernt hat, die gesamte Runde in der Mitte der Strecke zu bleiben. Wenn die Richtlinie also anfängt, gute Aktionen zu erlernen, beginnt der Agent, diese Aktionen zu verwenden, anstatt zufällige Aktionen zu ergreifen. Wenn der Agent jedoch immer die guten Aktionen verwendet oder ausnutzt, macht er keine neuen Entdeckungen, weil er die Umgebung nicht mehr erkundet. Dieser Kompromiss wird in RL oft als das Problem der Erforschung und Ausbeutung bezeichnet.

Experimentieren Sie mit den Standard-Aktionsräumen und Beispielbelohnungsfunktionen. Sobald Sie sie alle erforscht haben, können Sie Ihr Wissen nutzen, indem Sie Ihre eigenen benutzerdefinierten Aktionsräume und benutzerdefinierten Belohnungsfunktionen entwerfen.

DeepRacerAWS-Schulungsalgorithmen

Proximal Policy Optimization (PPO) versus Soft Actor Critic (SAC)

Die Algorithmen SAC und PPO lernen beide gleichzeitig eine Richtlinie und eine Wertfunktion, aber ihre Strategien unterscheiden sich in dreierlei Hinsicht:

PPO	SAC
Funktioniert sowohl in diskreten als auch in kontinuierlichen Aktionsräumen	Funktioniert in einem kontinuierlichen Aktionsraum
Entspricht den Richtlinien	Nicht richtlinienkonform
Verwendet die Regularisierung der Entropie	Fügt dem Maximierungsziel Entropie hinzu

Stabil oder datenhungrig

Die Informationen, die durch die Richtlinien der PPO- und SAC-Algorithmen bei der Erkundung einer Umgebung gewonnen werden, werden unterschiedlich genutzt. PPO nutzt politisches Lernen, was bedeutet, dass es seine Wertfunktion aus Beobachtungen lernt, die im Rahmen der aktuellen Umweltpolitik gemacht wurden. SAC nutzt außerpolitisches Lernen, was bedeutet, dass es auf Beobachtungen zurückgreifen kann, die im Rahmen der Umwelterkundung früherer politischer Maßnahmen gemacht wurden. Der Kompromiss zwischen außerpolitischem und politikinternem Lernen besteht häufig in Stabilität und Dateneffizienz. Politische Algorithmen sind tendenziell stabiler, aber datenhungriger, wohingegen Algorithmen außerhalb der Politik eher das Gegenteil bewirken.

Erkundung versus Ausbeutung

Exploration versus Ausbeutung ist eine zentrale Herausforderung in RL. Ein Algorithmus sollte bekannte Informationen aus früheren Erfahrungen nutzen, um höhere kumulative Belohnungen zu erzielen, aber er muss auch forschen, um neue Erfahrungen zu sammeln, die genutzt werden können, um die optimale Strategie für die future zu finden. Wenn eine Politik in mehreren Iterationen trainiert wird und mehr über ein Umfeld lernt, wird sie sicherer, ob sie eine Aktion für eine bestimmte Beobachtung auswählt. Wenn die Politik jedoch nicht genug erforscht, wird sie sich wahrscheinlich an bereits gewonnene Informationen halten, auch wenn diese nicht optimal sind. Der PPO-Algorithmus fördert die Exploration mithilfe der Entropie-Regularisierung, wodurch verhindert wird, dass Agenten

zu lokalen Optima konvergieren. Der SAC-Algorithmus schafft ein außergewöhnliches Gleichgewicht zwischen Exploration und Ausbeutung, indem er seinem Maximierungsziel die Entropie hinzufügt.

Entropie

In diesem Zusammenhang ist „Entropie“ ein Maß für die Unsicherheit in der Politik. Sie kann also als Maß dafür interpretiert werden, wie sicher eine Politik bei der Wahl einer Maßnahme für einen bestimmten Staat ist. Eine Politik mit niedriger Entropie ist bei der Wahl einer Maßnahme sehr zuversichtlich, wohingegen eine Politik mit hoher Entropie sich nicht sicher ist, welche Aktion sie wählen soll.

Die Strategie des SAC-Algorithmus zur Maximierung der Entropie hat ähnliche Vorteile wie die Verwendung der Entropie durch den PPO-Algorithmus als Regularisierer. Wie PPO fördert er eine umfassendere Erforschung und vermeidet die Konvergenz zu einem schlechten lokalen Optimum, indem er den Agenten dazu anregt, eine Aktion mit höherer Entropie zu wählen. Im Gegensatz zur Entropieregulation hat die Entropiemaximierung einen einzigartigen Vorteil. Sie neigt dazu, Strategien aufzugeben, die kein vielversprechendes Verhalten wählen. Dies ist ein weiterer Grund dafür, dass der SAC-Algorithmus tendenziell dateneffizienter ist als PPO.

Passen Sie das Ausmaß der Entropie in SAC mithilfe des SAC-Alpha-Hyperparameters an. Der maximale SAC-Alpha-Entropiewert (1,0) begünstigt die Exploration. Der Mindestwert (0,0) stellt das Standard-RL-Ziel wieder her und neutralisiert den Entropiebonus, der Anreize zur Erkundung bietet. Ein guter SAC-Alphawert, um mit dem Experimentieren zu beginnen, ist 0,5. Passen Sie Ihre Modelle entsprechend an, während Sie an Ihren Modellen iterieren.

Probieren Sie sowohl PPO- als auch SAC-Algorithmen aus, experimentieren Sie mit ihren Hyperparametern und erkunden Sie sie in verschiedenen Aktionsräumen.

Arbeitsablauf DeepRacer für AWS-Lösungen

Die Schulung eines DeepRacer AWS-Modells umfasst die folgenden allgemeinen Aufgaben:

1. Der DeepRacer AWS-Service initialisiert die Simulation mit einer virtuellen Strecke, einem Agenten, der das Fahrzeug repräsentiert, und dem Hintergrund. Der Agent ist ein neuronales Strategienetz, das mit Hyperparametern angepasst werden kann (wie im PPO-Algorithmus definiert).
2. Der Agent agiert (wie per Lenkwinkel und einer Geschwindigkeit festgelegt) basierend auf einem bestimmten Zustand (das Bild der Frontkamera).

3. Die simulierte Umgebung aktualisiert die Position des Agenten basierend auf der Aktion des Agenten und gibt eine Belohnung und ein aktualisiertes Kamerabild zurück. Die in Form von Zustand, Aktion, Belohnung und neuem Zustand gesammelten Erfahrungen werden genutzt, um das neuronale Netz regelmäßig zu aktualisieren. Die aktualisierten Netzwerkmodelle werden verwendet, um weitere Erfahrungen zu erzeugen.
4. Sie können das laufende Training auf der simulierten Strecke mit einer First-Person-Ansicht aus der Sicht des Agenten verfolgen. Sie können Metriken wie Belohnungen pro Episode, den Wert der Verlustfunktion und die Entropie der Strategie anzeigen. Auch die CPU- oder Speicherauslastung im Trainingsverlauf kann angezeigt werden. Darüber hinaus werden detaillierte Protokolle zur Analyse und zum Debuggen aufgezeichnet.
5. Der DeepRacer AWS-Service speichert das neuronale Netzwerkmodell regelmäßig im persistenten Speicher.
6. Das Training endet entsprechend einer zeitlichen Begrenzung.
7. Sie können das trainierte Modell in einem Simulator evaluieren. Reichen Sie dazu das trainierte Modell für Zeitversuche mit einer ausgewählten Anzahl von Läufen auf der ausgewählten Strecke ein.

Nachdem das Modell erfolgreich trainiert und bewertet wurde, kann es auf einen physischen Agenten (ein DeepRacer AWS-Fahrzeug) hochgeladen werden. Der Prozess umfasst die folgenden Schritte:

1. Laden Sie das trainierte Modell aus seinem persistenten Speicher (einem Amazon S3 S3-Bucket) herunter.
2. Verwenden Sie die Gerätesteuerkonsole des Fahrzeugs, um das trainierte Modell auf das Gerät hochzuladen. Verwenden Sie die Konsole, um das Fahrzeug zu kalibrieren und so den simulierten Aktionsraum auf den physischen Aktionsraum abzubilden. Sie können die Konsole außerdem verwenden, um die Drosselparität zu überprüfen, den Feed der Frontkamera zu betrachten, ein Modell in die Inferenz-Engine zu laden und das Fahrzeug auf einer echten Strecke zu beobachten.

   Die Gerätesteuerkonsole des Fahrzeugs ist ein Webserver, der auf dem Computing-Modul des Fahrzeugs gehostet wird. Sie können per WLAN über die Fahrzeug-IP-Adresse mit einem Webbrowser auf einem Computer oder einem Mobilgerät auf die Konsole zugreifen.
3. Experimentieren Sie mit dem Fahrzeug. Lassen Sie es mit unterschiedlichen Lichtverhältnissen, Batterieladezuständen, Oberflächentexturen und -farben fahren.

   Die Leistung des Geräts in einer physischen Umgebung entspricht aufgrund von Modellbeschränkungen oder unzureichendem Training möglicherweise nicht der Leistung in einer

simulierten Umgebung. Dieses Phänomen wird als sim2real-Leistungslücke bezeichnet. Mehr zur Verringerung dieser Abweichung erfahren Sie unter the section called “S imulated-to-real Leistungslücken”.

S imulated-to-real Leistungslücken

Da die Simulation nicht alle Aspekte der realen Welt genau abbilden kann, funktionieren die in der Simulation trainierten Modelle in der realen Welt möglicherweise nicht zufriedenstellend. Solche Diskrepanzen werden oft als simulated-to-real (sim2real) -Leistungslücken bezeichnet.

In AWS wurden Anstrengungen unternommen, DeepRacer um die Leistungslücke von sim2real zu minimieren. So ist beispielsweise der simulierte Agent so programmiert, dass er etwa 10 Aktionen pro Sekunde ausführt. Dies entspricht der Häufigkeit, mit der das DeepRacer AWS-Gerät Inferenzen ausführt, etwa 10 Inferenzen pro Sekunde. Außerdem wird beispielsweise zu Beginn jeder Episode im Training die Position des Agenten zufällig bestimmt. Dies maximiert die Wahrscheinlichkeit, dass der Agent alle Teile der Spur gleichmäßig lernt.

Um die real2sim-Leistungsunterschiede zu verringern, sollten Sie sowohl für die simulierten Strecken als auch für die echten Strecken Spuren derselben Farben, Formen und Abmessungen verwenden. Um visuelle Ablenkungen zu reduzieren, platzieren Sie Barrikaden um die reale Strecke herum. Kalibrieren Sie außerdem die Geschwindigkeits- und Lenkwinkelbereiche des Geräts sorgfältig, sodass der beim Training genutzte Aktionsraum der realen Welt entspricht. Die Evaluierung der Modellleistung auf einer anderen Simulationsstrecke als der im Training verwendeten kann den Umfang der real2real-Leistungslücke aufzeigen.

Weitere Informationen darüber, wie Sie die sim2real-Lücke beim Training eines DeepRacer AWS-Modells verringern können, finden Sie unter. the section called “Optimieren Sie das Training für reale Umgebungen”

Erste Schritte mit AWS DeepRacer

Um mit AWS zu beginnen DeepRacer, gehen wir zunächst die Schritte durch, um mithilfe der DeepRacer AWS-Konsole einen Agenten mit geeigneten Sensoren für Ihre Anforderungen an autonomes Fahren zu konfigurieren, ein Reinforcement-Learning-Modell für den Agenten mit den angegebenen Sensoren zu trainieren und das trainierte Modell zu bewerten, um die Qualität des Modells zu bestimmen. Sobald Sie Ihr Modell trainiert haben, können Sie es iterieren und es an einem Rennen teilnehmen lassen.

Themen

- Trainieren Sie Ihr erstes DeepRacer AWS-Modell
- Evaluieren Sie Ihre DeepRacer AWS-Modelle in der Simulation

Trainieren Sie Ihr erstes DeepRacer AWS-Modell

Diese exemplarische Vorgehensweise zeigt, wie Sie Ihr erstes Modell mit der DeepRacer AWS-Konsole trainieren.

Trainieren Sie ein Reinforcement-Learning-Modell mit der DeepRacer AWS-Konsole

Erfahren Sie, wo Sie in der DeepRacer AWS-Konsole die Schaltfläche Modell erstellen finden, um mit dem Modelltraining zu beginnen.

Um ein Reinforcement-Learning-Modell zu trainieren

1. Wenn Sie AWS zum ersten Mal verwenden DeepRacer, wählen Sie auf der Service-Landingpage die Option Modell erstellen oder im Hauptnavigationsbereich unter der Überschrift Reinforcement Learning die Option Erste Schritte aus.
2. Wählen Sie auf der Seite Erste Schritte mit Reinforcement-Learning unter Schritt 2: Modell erstellen die Option Modell erstellen aus.

   Sie können auch im Hauptnavigationsbereich unter der Überschrift Reinforcement-Learning die Option Ihre Modelle auswählen. Wählen Sie auf der Seite Your Models (Ihre Modelle) die Option Create model (Modell erstellen) aus.

Geben Sie den Modellnamen und die Umgebung an

Benennen Sie Ihr Modell und erfahren Sie, wie Sie die für Sie passende Simulationsstrecke auswählen.

Um den Modellnamen und die Umgebung anzugeben

1. Geben Sie auf der Seite Modell erstellen unter Trainingsdetails einen Namen für Ihr Modell ein.
2. Fügen Sie optional eine Beschreibung der Ausbildungsstelle hinzu.
3. Weitere Informationen zum Hinzufügen optionaler Tags finden Sie unterTagging.
4. Wählen Sie unter Umgebungssimulation einen Kurs aus, der als Schulungsumgebung für Ihren DeepRacer AWS-Agenten dienen soll. Wählen Sie unter Track-Richtung die Option Im Uhrzeigersinn oder Gegen den Uhrzeigersinn aus. Wählen Sie anschließend Weiter.

   Wählen Sie für die erste Ausführung eine Strecke mit einer einfachen Form und sanften Kurven aus. In späteren Iterationen können Sie komplexere Strecken auswählen, um die Modelle schrittweise zu verbessern. Um ein Modell für eine bestimmte Rennveranstaltung zu trainieren, wählen Sie die Strecke aus, die der Strecke der Veranstaltung am ähnlichsten ist.
5. Wählen Sie unten auf der Seite „Weiter“ aus.

Wähle einen Renntyp und einen Trainingsalgorhythmus

Die DeepRacer AWS-Konsole bietet drei Renntypen und zwei Trainingsalgorhythmen, aus denen Sie wählen können. Erfahren Sie, welche für Ihr Qualifikationsniveau und Ihre Trainingsziele geeignet sind.

Um einen Renntyp und einen Trainingsalgorhythmus auszuwählen

1. Wählen Sie auf der Seite Modell erstellen unter Renntyp die Option Zeitfahren, Objektvermeidung oder H ead-to-bot aus.

   Für deinen ersten Lauf empfehlen wir, Time Trial zu wählen. Hinweise zur Optimierung der Sensorkonfiguration Ihres Agenten für diesen Renntyp finden Sie unterthe section called “Maßgeschneiderte Schulungen für Zeitfahren”.
2. Wählen Sie bei späteren Läufen optional die Option „Objektvermeidung“, um stationäre Hindernisse zu umgehen, die an festen oder zufälligen Stellen entlang der ausgewählten Strecke

platziert sind. Weitere Informationen finden Sie unter the section called “Maßgeschneidertes Training für Rennen zur Objektvermeidung”.

a. Wählen Sie „Feste Position“, um Boxen an festen, vom Benutzer festgelegten Positionen auf den beiden Spuren der Strecke zu generieren, oder wählen Sie „Zufälliger Ort“, um Objekte zu generieren, die zu Beginn jeder Episode Ihrer Trainingssimulation zufällig auf die beiden Spuren verteilt werden.

b. Wählen Sie als Nächstes einen Wert für die Anzahl der Objekte auf einer Strecke.

c. Wenn Sie „Feste Position“ wählen, können Sie die Position der einzelnen Objekte auf der Spur anpassen. Wählen Sie für die Platzierung der Fahrspur zwischen der Innenspur und der Außenspur. Standardmäßig sind Objekte gleichmäßig über die Spur verteilt. Um zu ändern, wie weit sich ein Objekt zwischen Start- und Ziellinie befindet, geben Sie in das Feld Position (%) zwischen Start und Ziel einen Prozentsatz dieser Entfernung zwischen sieben und 90 ein.

3. Optional kannst du für anspruchsvollere Läufe Head-to-bot Rennen wählen, bei denen du gegen bis zu vier Bot-Fahrzeuge antrittst, die sich mit konstanter Geschwindigkeit bewegen. Weitere Informationen hierzu finden Sie unter the section called “Maßgeschneidertes Training für head-to-bot Rennen”.

   a. Wähle unter Anzahl der Bot-Fahrzeuge auswählen aus, mit wie vielen Bot-Fahrzeugen dein Agent trainieren soll.

   b. Wählen Sie als Nächstes die Geschwindigkeit in Millimetern pro Sekunde aus, mit der die Bot-Fahrzeuge auf der Strecke fahren sollen.

   c. Aktivieren Sie optional das Kontrollkästchen Spurwechsel aktivieren, um den Bot-Fahrzeugen die Möglichkeit zu geben, alle 1-5 Sekunden nach dem Zufallsprinzip die Spur zu wechseln.

4. Wählen Sie unter Trainingsalgorithmus und Hyperparameter den Algorithmus Soft Actor Critic (SAC) oder Proximal Policy Optimization (PPO) aus. In der DeepRacer AWS-Konsole müssen SAC-Modelle in kontinuierlichen Aktionsbereichen trainiert werden. PPO-Modelle können entweder in kontinuierlichen oder diskreten Aktionsräumen trainiert werden.

5. Verwenden Sie unter Trainingsalgorithmus und Hyperparameter die standardmäßigen Hyperparameterwerte unverändert.

   Um die Schulungsleistung zu verbessern, erweitern Sie später Hyperparameters (Hyperparameter) und ändern Sie die Standardwerte für Hyperparameter wie folgt:

a. Wählen Sie für Gradient Descent Batch Size (Gradientenabstieg-Batchgröße) die Option Available options (Verfügbare Optionen) aus.

b. Legen Sie für Number of epochs (Anzahl der Epochen) einen gültigen Wert fest.

c. Legen Sie für Learning rate (Lernrate) einen gültigen Wert fest.

d. Geben Sie für den SAC-Alphawert (nur SAC-Algorithmus) einen gültigen Wert ein.

e. Legen Sie für Entropy (Entropie) einen gültigen Wert fest.

f. Legen Sie für Discount factor (Abschlagfaktor) einen gültigen Wert fest.

g. Wählen Sie für Loss type (Loss-Typ) verfügbare Optionen aus.

h. Legen Sie für Number of experience episodes between each policy-updating iteration (Anzahl der Erfahrungsepisoden zwischen den einzelnen Strategieaktualisierungs-Iterationen) einen gültigen Wert fest.

Weitere Informationen zu Hyperparametern finden Sie unter Passen Sie Hyperparameter systematisch an.

6. Wählen Sie Weiter.

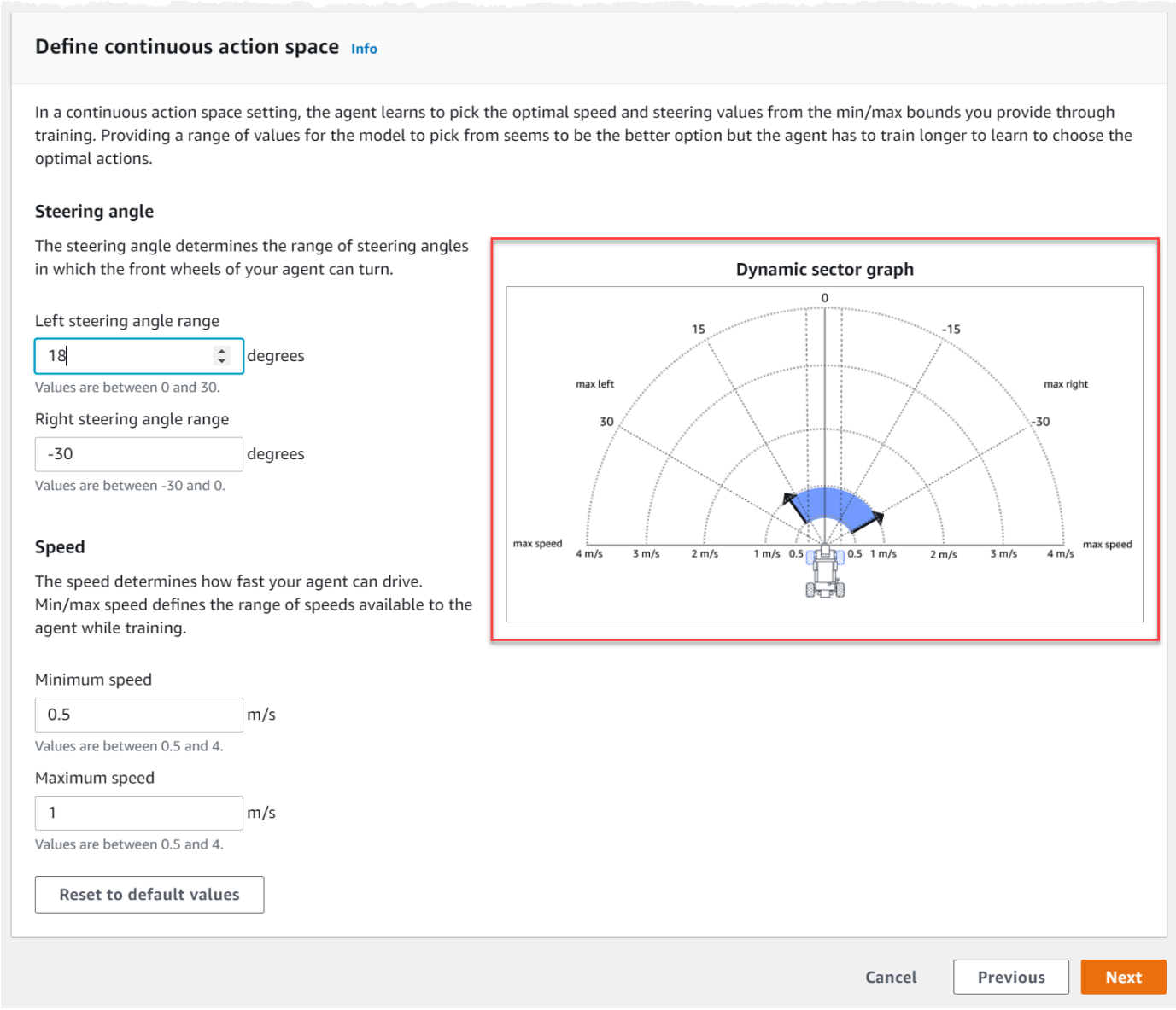
Definieren Sie den Aktionsraum

Wenn Sie sich auf der Seite Aktionsraum definieren dafür entschieden haben, mit dem Soft Actor Critic (SAC) -Algorithmus zu trainieren, ist Ihr Standard-Aktionsraum der kontinuierliche Aktionsraum. Wenn Sie sich für das Training mit dem PPO-Algorithmus (Proximal Policy Optimization) entschieden haben, wählen Sie zwischen Continuous Action Space und Discrete Action Space. Weitere Informationen darüber, wie die einzelnen Aktionsbereiche und Algorithmen das Trainingserlebnis des Agenten beeinflussen, finden Sie unterthe section called “Aktionsraum und Belohnungsfunktion”.

So definieren Sie einen kontinuierlichen Aktionsraum (SAC- oder PPO-Algorithmen)

1. Wählen Sie unter Kontinuierlichen Aktionsraum definieren die Grade Ihres linken Lenkwinkelbereichs und des rechten Lenkwinkelbereichs aus.

   Versuchen Sie, für jeden Lenkwinkelbereich unterschiedliche Grade einzugeben, und beobachten Sie, wie sich die Darstellung Ihres Bereichs ändert, um Ihre Auswahl im dynamischen Sektordiagramm darzustellen.



2. Geben Sie unter Geschwindigkeit eine Mindest- und Höchstgeschwindigkeit für Ihren Agenten in Millimetern pro Sekunde ein.

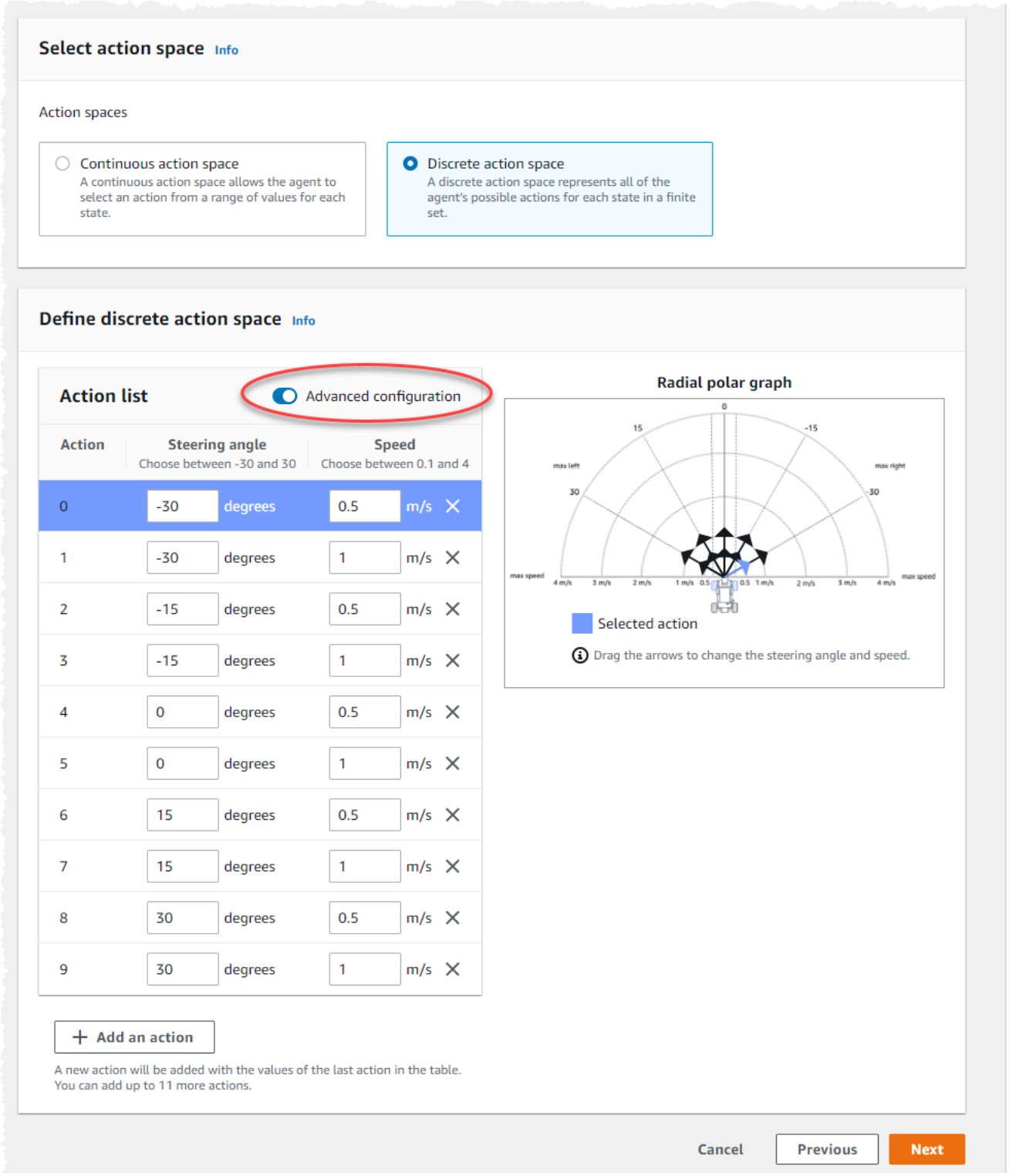
   Beachten Sie, wie sich Ihre Änderungen im dynamischen Sektordiagramm widerspiegeln.

3. Wählen Sie optional Auf Standardwerte zurücksetzen, um unerwünschte Werte zu löschen. Wir empfehlen, verschiedene Werte in der Grafik auszuprobieren, um zu experimentieren und zu lernen.

4. Wählen Sie Weiter.

Um einen diskreten Aktionsraum zu definieren (nur PPO-Algorithmus)

1. Wählen Sie einen Wert für die Granularität des Lenkwinkels aus der Dropdownliste aus.
2. Wählen Sie einen Wert in Grad zwischen 1 und 30 für den maximalen Lenkwinkel Ihres Agenten.
3. Wählen Sie aus der Dropdownliste einen Wert für die Granularität der Geschwindigkeit aus.
4. Wählen Sie für die Höchstgeschwindigkeit Ihres Agenten einen Wert in Millimetern pro Sekunde zwischen 0,1 und 4 aus.
5. Verwenden Sie die standardmäßigen Aktionseinstellungen in der Aktionsliste oder schalten Sie optional die Option Erweiterte Konfiguration ein, um Ihre Einstellungen zu verfeinern. Wenn Sie „Zurück“ wählen oder „Erweiterte Konfiguration“ deaktivieren, nachdem Sie die Werte angepasst haben, gehen Ihre Änderungen verloren.



a. Geben Sie in der Spalte Lenkwinkel einen Wert in Grad zwischen -30 und 30 ein.

b. Geben Sie in der Spalte Geschwindigkeit einen Wert zwischen 0,1 und 4 in Millimetern pro Sekunde für bis zu neun Aktionen ein.

c. Wählen Sie optional Aktion hinzufügen aus, um die Anzahl der Zeilen in der Aktionsliste zu erhöhen.

d. Wählen Sie optional X in einer Zeile aus, um sie zu entfernen.

6. Wählen Sie Weiter.

Wählen Sie ein virtuelles Auto

Erfahren Sie, wie Sie mit virtuellen Autos beginnen können. Verdiene dir jeden Monat neue Custom-Autos, Lackierungen und Modifikationen, indem du in der Open Division antrittst.

Um ein virtuelles Auto auszuwählen

1. Wähle auf der Seite Fahrzeugschale und Sensorkonfiguration auswählen eine Schale aus, die mit deinem Renntyp und deinem Aktionsraum kompatibel ist. Wenn Sie in Ihrer Garage kein passendes Auto haben, gehen Sie im Hauptnavigationsbereich unter der Überschrift Reinforcement-Learning zu Ihrer Garage, um eines zu erstellen.

   Für das Zeitfahrtraining benötigen Sie lediglich die Standard-Sensorkonfiguration und die Einlinsen-Kamera von The Original DeepRacer, aber alle anderen Gehäuse und Sensorkonfigurationen funktionieren, solange das Aktionsfeld passt. Weitere Informationen finden Sie unter the section called “Maßgeschneiderte Schulungen für Zeitfahren”.

   Für das Training zur Objektvermeidung sind Stereokameras hilfreich, aber eine einzelne Kamera kann auch verwendet werden, um stationären Hindernissen an festen Orten auszuweichen. Ein LiDAR-Sensor ist optional. Siehe the section called “Aktionsraum und Belohnungsfunktion”.

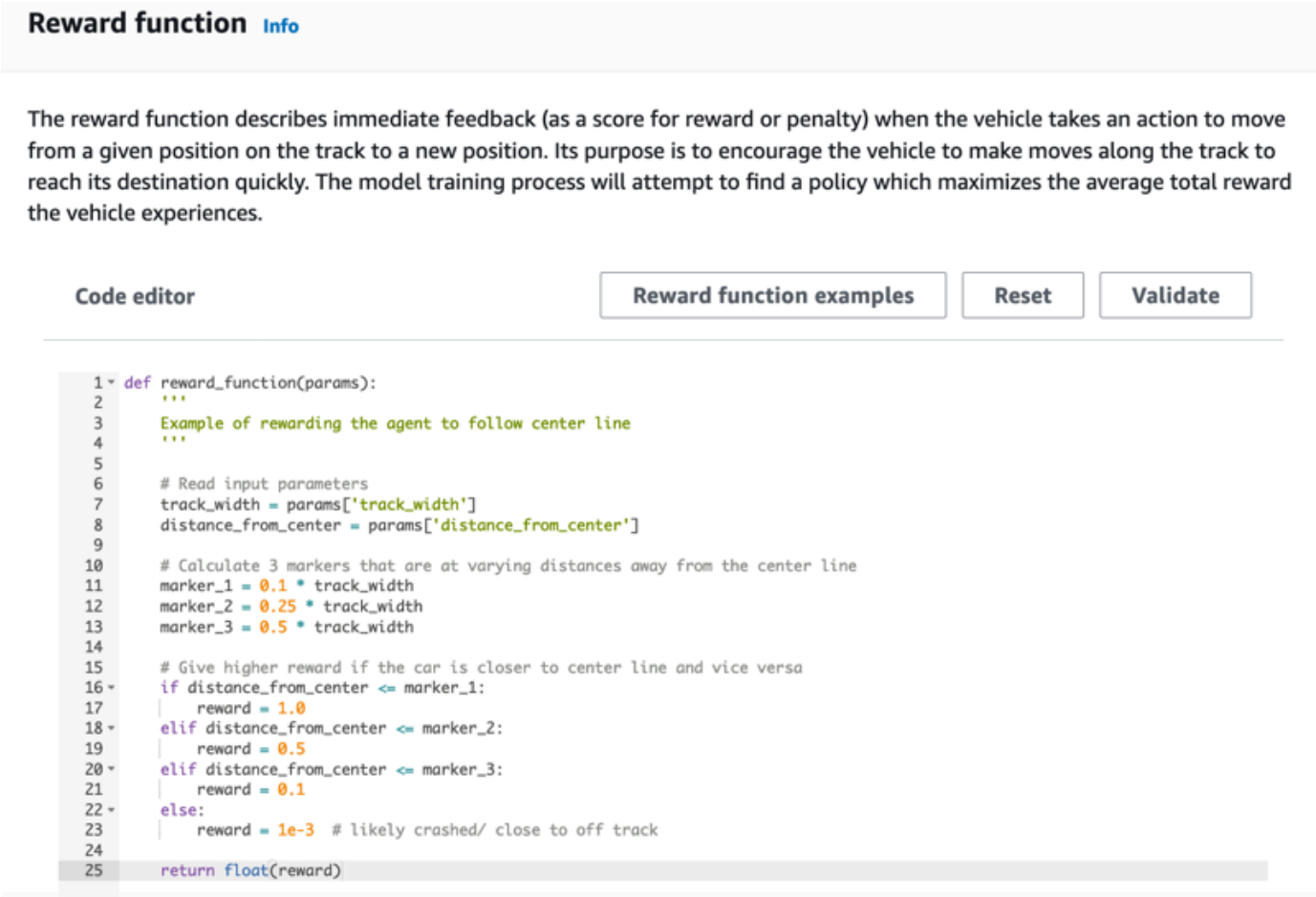
   Für das ead-to-botH-Training eignet sich neben einer Einzelkamera oder einer Stereokamera auch eine LiDAR-Einheit optimal, um tote Winkel zu erkennen und zu vermeiden, wenn Sie an anderen fahrenden Fahrzeugen vorbeifahren. Weitere Informationen hierzu finden Sie unter the section called “Maßgeschneidertes Training für head-to-bot Rennen”.
2. Wählen Sie Weiter.

Passen Sie Ihre Belohnungsfunktion an

Die Belohnungsfunktion ist das Herzstück von Reinforcement Learning. Lernen Sie, sie zu nutzen, um Ihrem Auto (Agenten) Anreize zu geben, bestimmte Maßnahmen zu ergreifen, während es die Strecke (Umgebung) erkundet. Sie können dieses Tool verwenden, um Ihr Auto zu ermutigen, eine Runde so schnell wie möglich zu beenden und es davon abzuhalten, von der Strecke abzukommen oder mit Objekten zu kollidieren.

So passen Sie Ihre Belohnungsfunktion an

1. Verwenden Sie auf der Seite Create model (Modell erstellen) unter Reward function (Belohnungsfunktion) das Standard-Belohnungsfunktionsbeispiel für Ihr erstes Modell unverändert.



Später können Sie Reward function examples (Belohnungsfunktionsbeispiele) wählen, um eine andere Beispielfunktion auszuwählen und dann Use code (Code verwenden), um die ausgewählte Belohnungsfunktion zu akzeptieren.

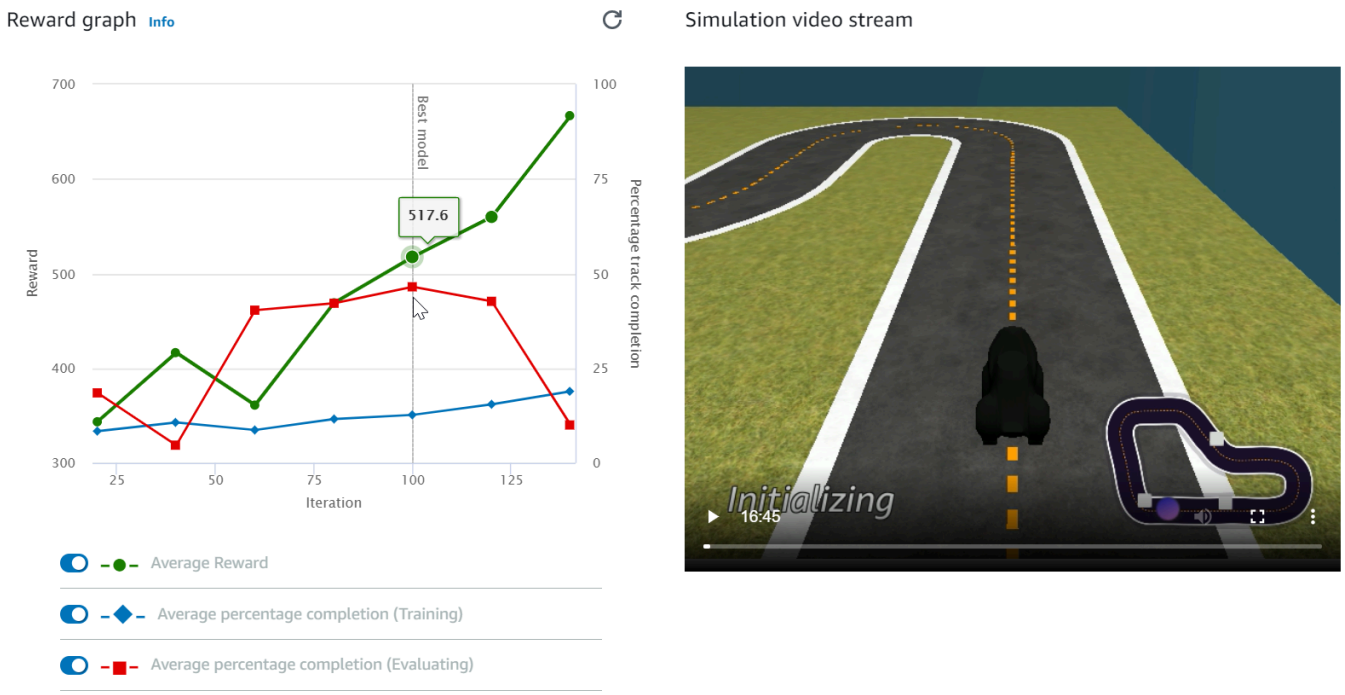
Es gibt vier Beispielfunktionen, mit denen Sie beginnen können. Sie veranschaulichen, wie man der Gleismitte folgt (Standard), wie man den Agenten innerhalb der Gleisgrenzen hält, wie man Zick-Zack-Fahrten verhindert und wie man verhindert, dass man gegen stehende Hindernisse oder andere fahrende Fahrzeuge stößt.

Weitere Informationen zur Belohnungsfunktion finden Sie unter the section called “Referenz zur Prämienfunktion”.

2. Lassen Sie unter Stoppbedingungen den Standardwert für maximale Zeit unverändert, oder legen Sie einen neuen Wert fest, um den Trainingsjob zu beenden, um zu verhindern, dass lange andauernde (und möglicherweise unkontrollierte) Trainingsjobs laufen.

   Beim Experimentieren in der frühen Phase des Trainings sollten Sie mit einem kleinen Wert für diesen Parameter beginnen und dann das Training schrittweise verlängern.
3. Unter Automatisch an AWS einreichen DeepRacer, Dieses Modell nach Abschluss der Schulung DeepRacer automatisch an AWS senden und die Möglichkeit haben, Preise zu gewinnen, ist standardmäßig aktiviert. Optional können Sie sich von der Teilnahme an Ihrem Modell abmelden, indem Sie das Häkchen anklicken.
4. Wählen Sie unter Liga-Anforderungen Ihr Wohnsitzland aus und akzeptieren Sie die Allgemeinen Geschäftsbedingungen, indem Sie das Kästchen ankreuzen.
5. Wählen Sie Modell erstellen, um mit der Erstellung des Modells und der Bereitstellung der Trainingsjob-Instanz zu beginnen.
6. Achten Sie nach der Übermittlung darauf, dass der Schulungsauftrag initialisiert und anschließend ausgeführt wird.

   Es dauert einige Minuten, bis der Initialisierungsvorgang von Initialisierung zu In Bearbeitung wechselt.
7. In Reward graph (Belohnungsdiagramm) und Simulation video stream (Simulationsvideo-Stream) können Sie den Fortschritt der Trainingsaufgabe beobachten. Sie können regelmäßig die Aktualisierungsschaltfläche neben Reward graph (Belohnungsdiagramm) auswählen, um Reward graph (Belohnungsdiagramm) zu aktualisieren, bis die Trainingsaufgabe beendet ist.



Der Trainingsjob wird in der AWS Cloud ausgeführt, sodass Sie die DeepRacer AWS-Konsole nicht geöffnet lassen müssen. Sie können jederzeit zur Konsole zurückkehren, um Ihr Modell zu überprüfen, während der Job ausgeführt wird.

Wenn das Simulationsvideostream-Fenster oder die Anzeige des Prämiendiagramms nicht mehr reagieren, aktualisieren Sie die Browserseite, um den Trainingsfortschritt zu aktualisieren.

Evaluieren Sie Ihre DeepRacer AWS-Modelle in der Simulation

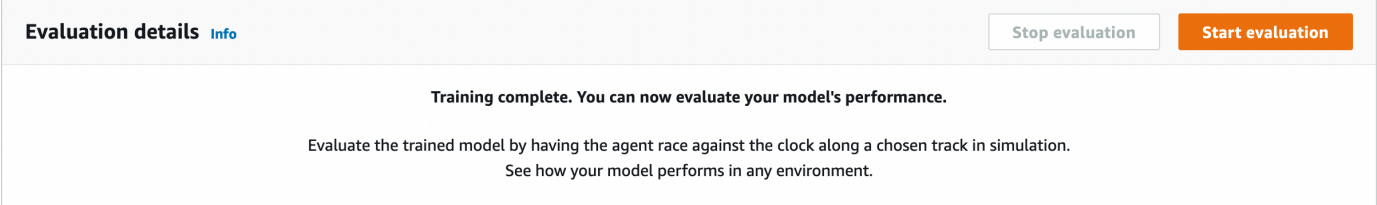
Nachdem Ihr Schulungsauftrag abgeschlossen ist, sollten Sie das geschulte Modell evaluieren, um dessen Konvergenzverhalten zu bewerten. Die Evaluierung erfolgt, indem eine Reihe von Läufen auf einer ausgewählten Strecke abgeschlossen und der Agent auf der Strecke nach wahrscheinlichen Aktionen, die vom geschulten Modell abgeleitet werden, bewegt wird. Die Leistungsmetriken umfassen den Prozentsatz der vollständigen Bewältigung der Strecke und die Zeit, die auf den einzelnen Strecken vom Start bis zum Ende oder außerhalb der Strecke gefahren wird.

Um Ihr trainiertes Modell zu evaluieren, können Sie die DeepRacer AWS-Konsole verwenden. Führen Sie hierzu die Schritte in diesem Thema aus.

Um ein trainiertes Modell in der DeepRacer AWS-Konsole zu evaluieren

1. Öffnen Sie die DeepRacer AWS-Konsole unter https://console.aws.amazon.com /deepracer.

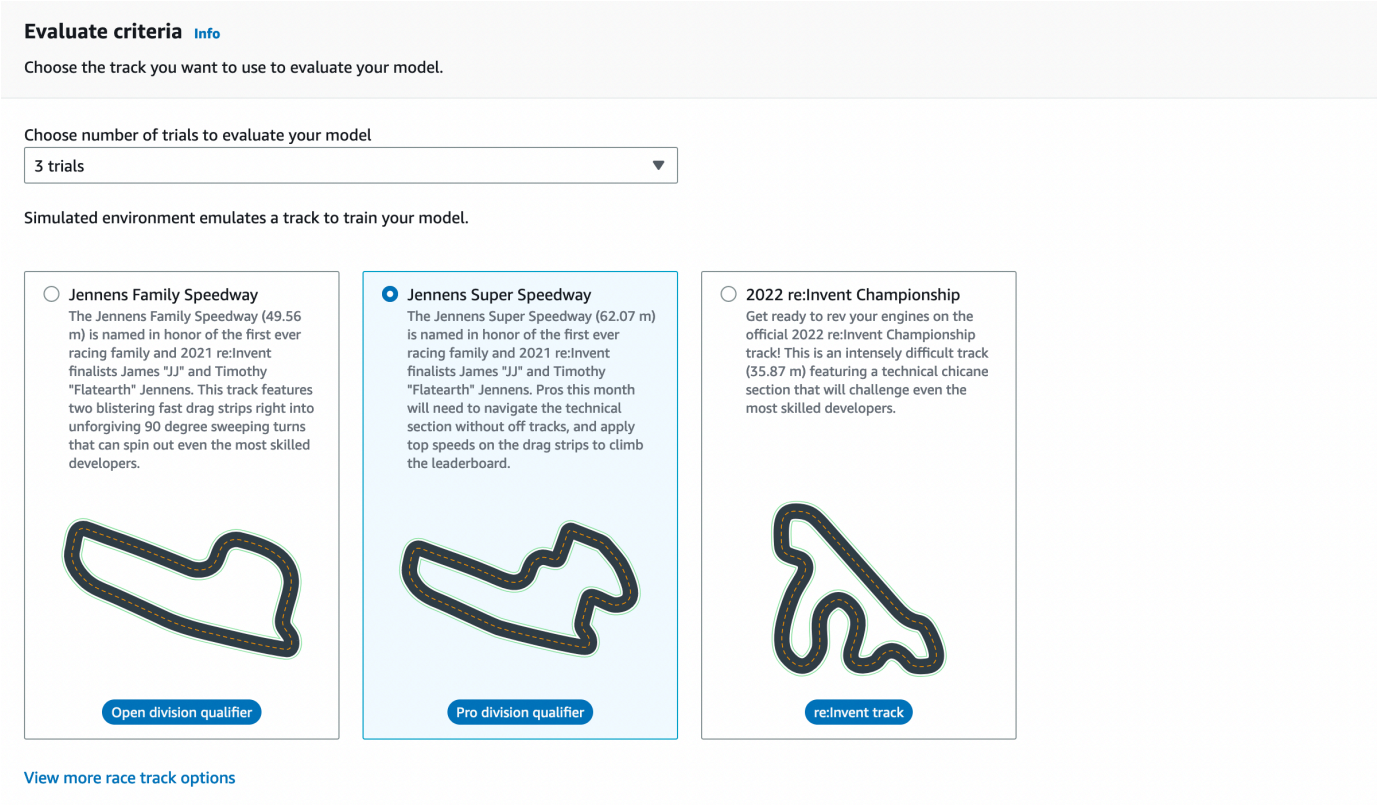
2. Wählen Sie im Hauptnavigationsbereich Models (Modelle) und dann das gerade geschulte Modell aus der Liste Models (Modelle) aus, um die Modelldetailseite zu öffnen.
3. Wählen Sie die Registerkarte Evaluation aus.
4. Wählen Sie unter Bewertungsdetails die Option Bewertung starten aus.



   Sie können eine Evaluierung starten, nachdem sich der Schulungsauftragsstatus zu Completed (Abgeschlossen) oder der Status des Modells zu Ready (Bereit) geändert hat, wenn der Schulungsauftrag nicht abgeschlossen wurde.

   Ein Modell ist bereit, wenn der Schulungsauftrag abgeschlossen ist. Wenn die Schulung nicht abgeschlossen wurde, kann sich das Modell auch im Zustand Ready (Bereit) befinden, wenn es bis zum Fehlerpunkt geschult wurde.
5. Geben Sie auf der Seite Modell bewerten unter Renntyp einen Namen für Ihre Bewertung ein und wählen Sie dann den Renntyp aus, den Sie für das Training des Modells ausgewählt haben.

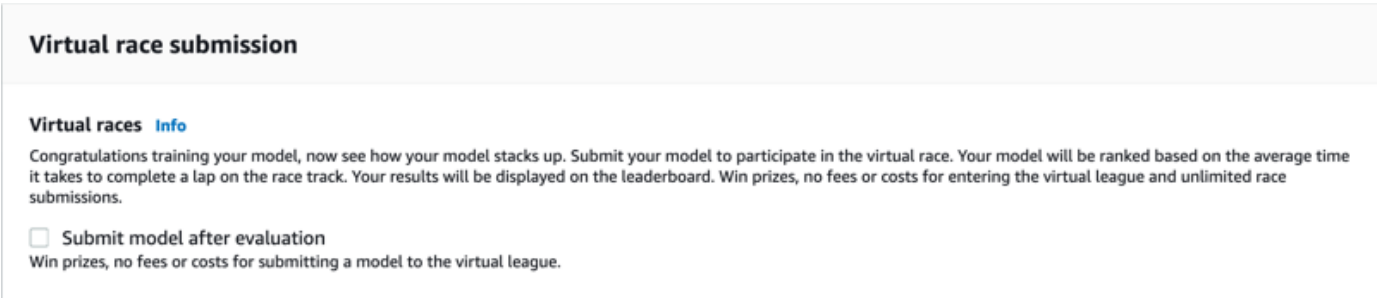
   Für die Evaluierung können Sie einen anderen Renntyp als den in der Schulung verwendeten Renntyp auswählen. Sie können beispielsweise ein Modell für head-to-bot Rennen trainieren und es dann für Zeitfahren auswerten. Im Allgemeinen muss das Modell gut generalisieren, wenn sich der Schulungstyp vom Renntyp für die Evaluierung unterscheidet. Für den ersten Lauf sollten Sie den gleichen Renntyp sowohl für die Evaluierung als auch für die Schulung verwenden.
6. Wählen Sie auf der Seite Modell evaluieren unter Bewertungskriterien die Anzahl der Versuche aus, die Sie durchführen möchten, und wählen Sie dann eine Strecke aus, auf der das Modell bewertet werden soll.



Normalerweise sollten Sie eine Strecke auswählen, die der entspricht, die Sie beim Training des Modells verwendet haben (bzw. dieser ähnlich ist). Sie können zur Evaluierung Ihres Modells eine beliebige Strecke auswählen. Die beste Leistung ist jedoch auf einer Strecke zu erwarten, die derjenigen im Training am nächsten kommt.

Um zu sehen, ob Ihr Modell gut generalisiert, wählen Sie eine andere Evaluierungsstrecke als die in der Schulung verwendete Strecke aus.

7. Deaktivieren Sie auf der Seite Evaluate model (Modell evaluieren) unter Virtual Race Submission (Virtuelle Renneinreichung) für Ihr erstes Modell die Option Submit model after evaluation (Modell nach Evaluierung senden). Wenn Sie später an einer Rennveranstaltung teilnehmen möchten, lassen Sie diese Option aktiviert.



8. Wählen Sie auf der Seite Evaluate model (Modell evaluieren) die Option Start evaluation (Evaluierung starten), um mit der Erstellung und Initialisierung des Evaluierungsauftrags zu beginnen.

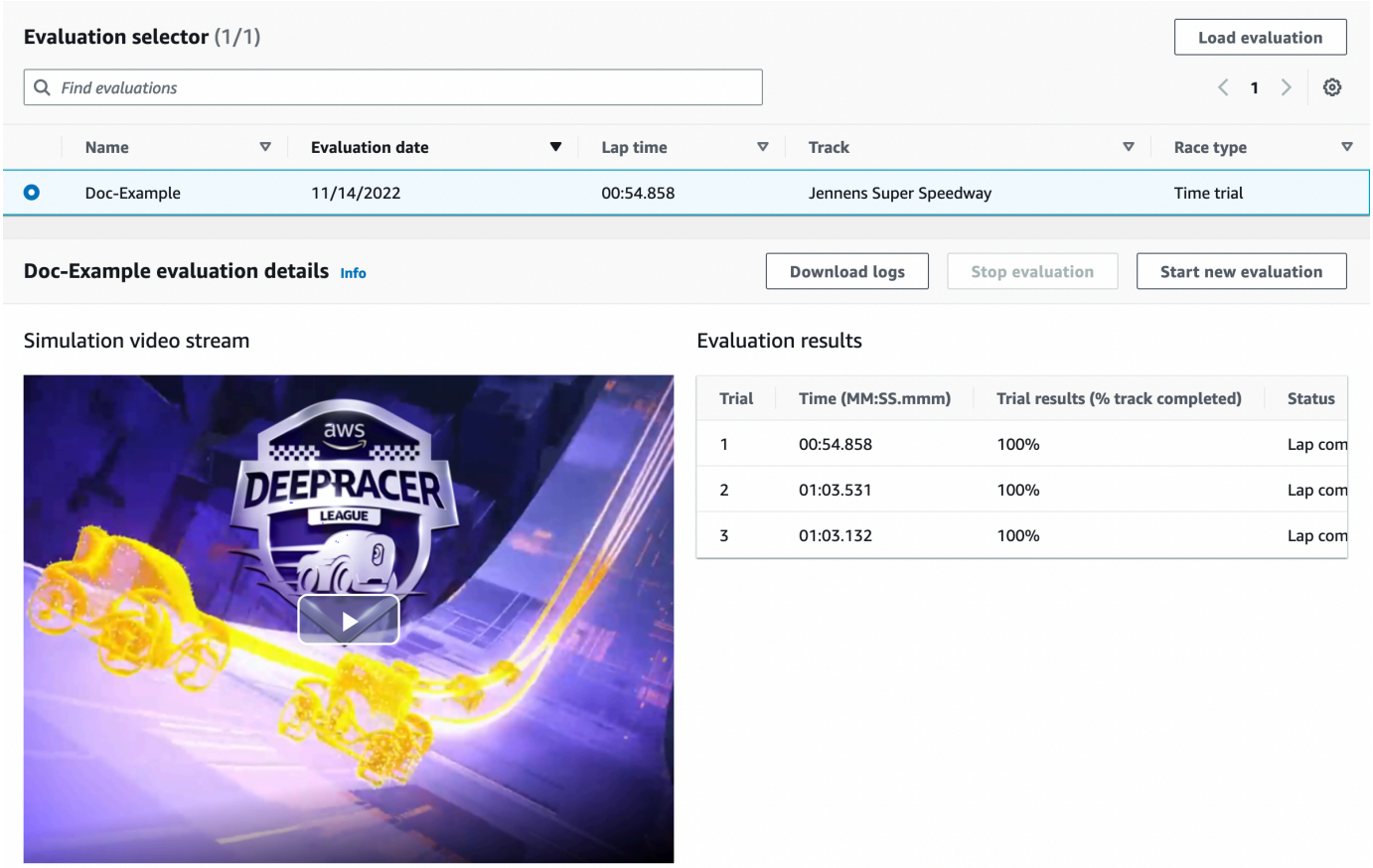
   Dieser Initialisierungsprozess dauert etwa 3 Minuten.

9. Im Verlauf der Evaluierung werden die Bewertungsergebnisse, einschließlich der Testzeit und der Abschlussquote der Strecke, nach jedem Test unter den Bewertungsdetails angezeigt. Im Fenster Simulation Video Stream (Simulationsvideostream) können Sie sehen, wie der Agent auf der gewählten Strecke funktioniert.

   Sie können einen Evaluierungsauftrag beenden, bevor er abgeschlossen wird. Um einen Evaluierungsauftrag zu beenden, wählen Sie Stop evaluation (Evaluierung stoppen) in der oberen rechten Ecke der Karte Evaluation (Evaluierung) und bestätigen Sie dies dann, um die Evaluierung zu stoppen.

10. Nachdem der Evaluierungsauftrag abgeschlossen ist, untersuchen Sie die Leistungsmetriken aller Läufe unter Evaluation results (Evaluierungsergebnisse). Der dazugehörige Simulationsvideostream ist nicht mehr verfügbar.

    Eine Historie der Evaluierungen Ihres Modells ist in der Evaluationsauswahl verfügbar. Um die Details einer bestimmten Bewertung anzuzeigen, wählen Sie die Bewertung aus der Auswahlliste für die Bewertung aus und klicken Sie dann in der oberen rechten Ecke der Bewertungsauswahlkarte auf Bewertung laden.



Bei dieser speziellen Bewertungsaufgabe schließt das trainierte Modell die Tests ab, was zu erheblichen Zeitverzögerungen führt. Bei einem ersten Durchlauf ist dies nicht ungewöhnlich. Mögliche Gründe dafür sind, dass die Schulung nicht konvergiert hat und mehr Zeit benötigt, der Aktionsbereich vergrößert werden muss, um dem Agenten mehr Raum für Reaktionen zu geben, oder dass die Belohnungsfunktion aktualisiert werden muss, um unterschiedliche Umgebungen zu bewältigen.

In diesem Fall können Sie das Modell weiter verbessern, indem Sie ein zuvor geschultes Modell klonen, die Belohnungsfunktion ändern, Hyperparameter optimieren und dann den Prozess wiederholen, bis die Gesamtbelohnung konvergiert und sich die Leistungsmetriken verbessern. Weitere Informationen zur Verbesserung des Trainings finden Sie unter Modelle trainieren und auswerten.

Um Ihr vollständig trainiertes Modell für den Betrieb in einer physischen Umgebung auf Ihr DeepRacer AWS-Gerät zu übertragen, müssen Sie die Modellartefakte herunterladen. Wählen Sie dazu auf der Detailseite des Modells Download model (Modell herunterladen) aus. Wenn Ihr DeepRacer physisches AWS-Gerät keine neuen Sensoren unterstützt und Ihr Modell mit den neuen

Sensortypen trainiert wurde, erhalten Sie eine Fehlermeldung, wenn Sie das Modell auf Ihrem DeepRacer AWS-Gerät in einer realen Umgebung verwenden. Weitere Informationen zum Testen eines DeepRacer AWS-Modells mit einem physischen Gerät finden Sie unterBedienen Sie Ihr Fahrzeug.

Sobald Sie Ihr Modell auf einer Strecke trainiert haben, die mit der Strecke identisch oder ähnlich ist, die bei einer AWS DeepRacer League-Rennveranstaltung oder einem DeepRacer AWS-Community-Rennen angegeben wurde, können Sie das Modell für die virtuellen Rennen in der DeepRacer AWS-Konsole einreichen. Folgen Sie dazu im Hauptnavigationsbereich den Rennen der AWS virtuellen Rennstrecke oder der Community. Weitere Informationen finden Sie unter Nehmen Sie an einem Rennen teil.

Um ein Modell für die Vermeidung von Hindernissen oder head-to-bot Rennen zu trainieren, müssen Sie dem Modell und dem physischen Gerät möglicherweise neue Sensoren hinzufügen. Weitere Informationen finden Sie unter the section called “Renntypen verstehen und Sensoren aktivieren”.

DeepRacer AWS-Modelle trainieren und auswerten

Wenn Ihr DeepRacer AWS-Fahrzeug selbstständig auf einer Strecke fährt, erfasst es die Umgebungsbedingungen mit der an der Vorderseite montierten Kamera und reagiert auf die Beobachtungen. Ihr DeepRacer AWS-Modell ist eine Funktion, die die Beobachtungen und Aktionen der erwarteten Belohnung zuordnet. Ihr Modell zu trainieren bedeutet, die Funktion zu suchen oder zu erlernen, die die erwartete Belohnung maximiert, sodass das optimierte Modell vorgibt, welche Aktionen (Kombinationen aus Geschwindigkeit und Lenkwinkel) Ihr Fahrzeug ausführen kann, um sich vom Anfang bis zum Ende entlang der Strecke zu bewegen.

In der Praxis ist diese Funktion ein neuronales Netz. Das Training des Netzes umfasst das Finden der optimalen Netzwerkgewichtungen anhand von Sequenzen beobachteter Umgebungszustände und der Aktionen des reagierenden Fahrzeugs. Die zugrunde liegenden Optimalitätskriterien werden durch die Belohnungsfunktion des Modells beschrieben. Diese fördert das korrekte und produktive Fahren des Fahrzeugs, ohne dass es zu Unfällen oder Regelverletzungen kommt. Eine einfache Belohnungsfunktion könnte eine Belohnung von 0 zurückgeben, wenn sich das Fahrzeug auf der Strecke befindet. Sie könnte -1 zurückgeben, wenn es sich außerhalb der Strecke befindet. Sie könnte außerdem +1 zurückgeben, wenn es die Ziellinie erreicht. Mit dieser Belohnungsfunktion wird das Fahrzeug für das Verlassen der Strecke bestraft und für das Erreichen des Ziels belohnt. Dies kann eine gute Belohnungsfunktion sein, wenn Zeit oder Geschwindigkeit kein Thema ist.

Angenommen, Sie sind daran interessiert, das Fahrzeug so schnell wie möglich zu fahren, ohne von einer geraden Spur zu kommen. Während das Fahrzeug beschleunigt und bremst, kann es nach links oder rechts lenken, um Hindernisse zu umgehen oder auf der Strecke zu bleiben. Eine Kurve bei hoher Geschwindigkeit zu weit zu nehmen, könnte das Fahrzeug leicht von der Strecke abbringen. Eine zu enge Kurve führt jedoch möglicherweise zur Kollision mit einem Hindernis oder einem anderen Fahrzeug. Im Allgemeinen wäre eine optimale Aktion, eine größere Kurve bei geringerer Geschwindigkeit zu machen oder in einer schärferen Kurve weniger zu lenken. Um dieses Verhalten zu fördern, muss Ihre Belohnungsfunktion kleineren Kurven mit einer höheren Geschwindigkeit eine positive Bewertung zuweisen und/oder größere Kurven mit einer höheren Geschwindigkeit mit einer negativen Bewertung bestrafen. Ebenso kann die Belohnungsfunktion eine positive Belohnung für die Beschleunigung auf einer geradlinigeren Strecke oder das Abbremsen in der Nähe eines Hindernisses zurückgeben.

Die Belohnungsfunktion ist ein wichtiger Bestandteil Ihres DeepRacer AWS-Modells. Sie müssen es beim Training Ihres DeepRacer AWS-Modells angeben. Das Training umfasst wiederholte Episoden entlang der Strecke, von deren Anfang bis zu deren Ende. In einer Episode interagiert der

Agent mit der Strecke, um die optimalen Aktionen zu erlernen, indem er die erwartete kumulative Belohnung maximiert. Am Ende des Trainings wird ein Verstärkungs-Lernmodell erstellt. Nach dem Training kann der Agent autonom fahren, indem er Inferenzen über das Modell ausführt, um in jedem beliebigen Zustand eine optimale Aktion durchzuführen. Dies kann entweder in der simulierten Umgebung mit einem virtuellen Agenten oder in einer realen Umgebung mit einem physischen Agenten, z. B. einem Fahrzeug im DeepRacer AWS-Maßstab, erfolgen.

Um ein Reinforcement-Learning-Modell in der Praxis zu trainieren, müssen Sie einen Lernalgorithmus auswählen. Derzeit unterstützt die DeepRacer AWS-Konsole nur die Algorithmen Proximal Policy Optimization (PPO) und Soft Actor Critic (SAC). Sie können dann ein Deep-Learning-Framework wählen, das den ausgewählten Algorithmus unterstützt, es sei denn, Sie möchten eines von Grund auf neu schreiben. AWS ist in SageMaker KI DeepRacer integriert, um einige beliebte Deep-Learning-Frameworks, wie TensorFlowz. B., sofort in der DeepRacer AWS-Konsole verfügbar zu machen. Die Verwendung eines Frameworks vereinfacht die Konfiguration und Ausführung von Trainingsjobs und ermöglicht es Ihnen, sich auf die Erstellung und Erweiterung von Belohnungsfunktionen zu konzentrieren, die speziell auf Ihre Probleme zugeschnitten sind.

Das Training eines Reinforcement-Learning-Modells ist ein iterativer Prozess. Erstens ist es eine Herausforderung, eine Belohnungsfunktion zu definieren, die alle wichtigen Verhaltensweisen eines Agenten in einer Umgebung auf einmal abdeckt. Zweitens werden Hyperparameter oft so eingestellt, dass eine zufriedenstellende Trainingsleistung gewährleistet ist. Beides erfordert Experimente. Ein umsichtiger Ansatz besteht darin, mit einer einfachen Belohnungsfunktion zu beginnen und diese dann schrittweise zu verbessern. AWS DeepRacer erleichtert diesen iterativen Prozess, indem es Ihnen ermöglicht, ein trainiertes Modell zu klonen und es dann als Starthilfe für die nächste Schulungsrunde zu verwenden. Bei jeder Iteration können Sie eine oder mehrere komplexere Behandlungen in die Belohnungsfunktion einführen, um mit zuvor ignorierten Variablen umzugehen, oder Sie können Hyperparameter systematisch anpassen, bis das Ergebnis konvergiert.

Genau wie bei allen Verfahren maschinellen Lernens müssen Sie ein trainiertes Reinforcement-Learning-Modell evaluieren, um seine Wirksamkeit zu überprüfen, bevor Sie es an einen physischen Agenten weitergeben, der Inferenzen in einer realen Situation nutzt. Bei autonomen Fahrten kann die Evaluierung darauf basieren, wie oft ein Fahrzeug von Anfang bis Ende auf einer bestimmten Strecke bleibt oder wie schnell es die Strecke beenden kann, ohne von der Strecke abzukommen. Mit der DeepRacer AWS-Simulation können Sie die Bewertung durchführen und die Leistungskennzahlen zum Vergleich mit Modellen veröffentlichen, die von anderen DeepRacer AWS-Benutzern trainiert wurden, auf einer Bestenliste.

Themen

- Renntypen verstehen und von AWS unterstützte Sensoren aktivieren DeepRacer
- Trainieren und evaluieren Sie DeepRacer AWS-Modelle mit der DeepRacer AWS-Konsole
- Referenz zur DeepRacer AWS-Prämienfunktion

Renntypen verstehen und von AWS unterstützte Sensoren aktivieren DeepRacer

In der AWS DeepRacer League können Sie an den folgenden Arten von Rennveranstaltungen teilnehmen:

- Time Trial: Rennen gegen die Uhr auf einer Strecke ohne Hindernisse mit dem Ziel, eine Runde möglichst schnell zu absolvieren.
- Objektvermeidung: Rennen gegen die Uhr auf einer Strecke mit stationären Hindernissen und dem Ziel, eine Runde möglichst schnell zu absolvieren.
- Head-to-bot Rennen: Tritt gegen ein oder mehrere andere Fahrzeuge auf derselben Strecke an und versuche, die Ziellinie vor anderen Fahrzeugen zu überqueren.

AWS DeepRacer Community Races unterstützt derzeit nur Zeitfahren.

Sie sollten mit verschiedenen Sensoren an Ihrem DeepRacer AWS-Fahrzeug experimentieren, damit es seine Umgebung für einen bestimmten Renntyp ausreichend beobachten kann. Im nächsten Abschnitt werden die von AWS DeepRacer unterstützten Sensoren beschrieben, die die unterstützten Typen von autonomen Rennveranstaltungen ermöglichen können.

Themen

- Wählen Sie Sensoren für DeepRacer AWS-Renntypen
- Agenten für das Training von DeepRacer AWS-Modellen konfigurieren
- Maßgeschneiderte DeepRacer AWS-Schulungen für Zeitfahren
- Maßgeschneiderte DeepRacer AWS-Schulungen für Rennen zur Objektvermeidung
- Maßgeschneiderte DeepRacer AWS-Schulungen für head-to-bot Rennen

Wählen Sie Sensoren für DeepRacer AWS-Renntypen

Ihr DeepRacer AWS-Fahrzeug ist mit einer nach vorne gerichteten monokularen Kamera als Standardsensor ausgestattet. Sie können eine weitere nach vorne gerichtete Monokularkamera hinzufügen, um nach vorne gerichtete Stereokameras zu erhalten oder um die Monokular- oder Stereokameras mit einer LiDAR-Einheit zu ergänzen.

Die folgende Liste fasst die Funktionsfähigkeit der von AWS DeepRacer unterstützten Sensoren zusammen mit kurzen cost-and-benefit Analysen zusammen:

Nach vorne gerichtete Kamera

Eine nach vorne gerichtete Kamera mit einem Objektiv kann Bilder der Umgebung vor dem Hostfahrzeug aufnehmen, einschließlich Streckenbegrenzungen und -formen. Dies ist der kostengünstigste Sensor, der sich für einfachere autonome Fahraufgaben wie Time Trials ohne Hindernisse auf gut markierten Strecken eignet. Bei richtiger Schulung kann es stationäre Hindernisse an festen Stellen auf der Strecke vermeiden. Die Informationen zur Positionen von Hindernissen sind jedoch in das geschulte Modell integriert, weshalb das Modell wahrscheinlich übermäßig ausgestattet ist und möglicherweise keine Generalisierungen für andere Hindernisorte durchführen kann. Bei stationären Objekten an zufälligen Stellen oder anderen sich bewegenden Fahrzeugen auf der Strecke, ist es unwahrscheinlich, dass das Modell konvergiert.

In der realen Welt ist das DeepRacer AWS-Fahrzeug mit einer Frontkamera mit einem Objektiv als Standardsensor ausgestattet. Die Kamera verfügt über ein 120-Grad-Weitwinkelobjektiv und erfasst RGB-Bilder, die dann in Graustufenbilder von 160 x 120 Pixeln bei 15 Bildern pro Sekunde (fps) konvertiert werden. Diese Sensoreigenschaften bleiben im Simulator erhalten, um die Möglichkeit zu maximieren, dass das geschulte Modell gut von der Simulation in die reale Welt übertragen werden kann.

Nach vorne gerichtete Stereokamera

Eine Stereokamera hat zwei oder mehr Objektive, die Bilder mit der gleichen Auflösung und Frequenz aufnehmen. Bilder von beiden Objektiven werden verwendet, um die Tiefe der beobachteten Objekte festzustellen. Die Tiefeninformationen von einer Stereokamera sind wertvoll für das Hostfahrzeug, um Kollisionen mit Hindernissen oder anderen Fahrzeugen nach vorn zu vermeiden, besonders in dynamischeren Umgebungen. Zusätzliche Tiefeninformationen führen jedoch zu einer langsameren Konvergenz bei der Schulung.

Auf dem DeepRacer physischen AWS-Fahrzeug wird die Stereokamera mit doppelter Linse konstruiert, indem eine weitere Einzelobjektivkamera hinzugefügt und jede Kamera an der linken

und rechten Seite des Fahrzeugs montiert wird. Die DeepRacer AWS-Software synchronisiert Bildaufnahmen von beiden Kameras. Die erfassten Bilder werden in Graustufen umgewandelt, gestapelt und zu Inferenzzwecken in das neuronale Netzwerk eingespeist. Der gleiche Mechanismus wird im Simulator dupliziert, um das Modell zu schulen, damit es brauchbare Generalisierungen für eine reale Umgebung leisten kann.

LiDAR-Sensor

Ein LiDAR-Sensor verwendet rotierende Laser, um Lichtimpulse außerhalb des sichtbaren Spektrums auszusenden, wobei die Zeit bis zur Rückkehr des Impulses gemessen wird. Die Richtung und Entfernung zu den Objekten, auf die ein bestimmter Impuls trifft, werden als Punkt in einer großen 3D-Karte aufgezeichnet, die um die LiDAR-Einheit zentriert ist.

LiDAR hilft beispielsweise bei der Erkennung von toten Winkeln des Hostfahrzeugs, um Kollisionen zu vermeiden, während das Fahrzeug die Fahrspur wechselt. Durch die Kombination von LiDAR mit Mono- oder Stereokameras können Sie dem Hostfahrzeug ermöglichen, genügend Informationen erfassen, um geeignete Maßnahmen zu ergreifen. Ein LiDAR-Sensor kostet jedoch mehr im Vergleich zu Kameras. Das neuronale Netzwerk muss lernen, die LiDAR-Daten zu interpretieren. Dadurch dauert die Schulung länger bis zur Konvergierung.

Am DeepRacer physischen AWS-Fahrzeug ist ein LiDAR-Sensor an der Rückseite montiert und um 6 Grad nach unten geneigt. Er dreht sich mit einer Winkelgeschwindigkeit von 10 Umdrehungen pro Sekunde und hat einen Bereich von 15 cm bis 2 m. Er kann Objekte hinter und neben dem Hostfahrzeug sowie hohe Objekte erkennen, die von den Fahrzeugteilen an der Vorderseite nicht behindert werden. Der Winkel und die Reichweite sind so gewählt, dass die LiDAR-Einheit weniger anfällig für Umgebungsgeräusche ist.

Sie können Ihr DeepRacer AWS-Fahrzeug mit der folgenden Kombination der unterstützten Sensoren konfigurieren:

- Nur nach vorne gerichtete Kamera mit einem Objektiv

  Diese Konfiguration eignet sich sowohl für Time Trials, als auch für die Hindernisvermeidung mit Objekten an festen Standorten.
- Nur nach vorne gerichtete Stereokamera

  Diese Konfiguration eignet sich gut zur Vermeidung von Hindernissen bei Objekten an festen oder zufälligen Orten.
- Nach vorne gerichtete Kamera mit einem Objektiv und LiDAR.

Diese Konfiguration eignet sich gut zur Hindernisvermeidung oder head-to-bot für Rennen.

- Nach vorne gerichtete Stereokamera mit LiDAR.

  Diese Konfiguration eignet sich gut für die Vermeidung von Hindernissen oder für head-to-bot Rennen, aber wahrscheinlich nicht die wirtschaftlichste Lösung für Zeitfahren.

Wenn Sie weitere Sensoren hinzufügen, damit Ihr DeepRacer AWS-Fahrzeug von Zeitfahren über Objektvermeidung bis hin zu head-to-bot Rennen durchläuft, sammelt das Fahrzeug mehr Daten über die Umgebung, die beim Training in das zugrunde liegende neuronale Netzwerk eingespeist werden. Dies macht die Schulung anspruchsvoller, da das Modell mit mehr Komplexität umgehen muss. Für Sie wird es dadurch anspruchsvoller, Modelle zu schulen.

Um schrittweise zu lernen, sollten Sie zunächst mit dem Training für Zeitfahren beginnen, bevor Sie zur Objektvermeidung und dann zum Rennen übergehen. head-to-bot Ausführlichere Empfehlungen finden Sie im nächsten Abschnitt.

Agenten für das Training von DeepRacer AWS-Modellen konfigurieren

Um ein Reinforcement-Learning-Modell für das DeepRacer AWS-Fahrzeug zu trainieren, damit es bei Hindernisvermeidung oder head-to-bot Rennen Rennen fahren kann, müssen Sie den Agenten mit den entsprechenden Sensoren konfigurieren. Für einfache Time Trials können Sie den Standardagenten verwenden, der mit einer Kamera mit einem Objektiv konfiguriert ist. Bei der Konfiguration des Agenten können Sie den Aktionsbereich anpassen und eine neuronale Netzwerktopologie auswählen, damit diese besser mit den ausgewählten Sensoren arbeiten, um die jeweiligen Fahranforderungen zu erfüllen. Darüber hinaus können Sie das Erscheinungsbild des Agenten zur visuellen Identifizierung während der Schulung ändern.

Nach der Konfiguration wird die Agentenkonfiguration als Teil der Metadaten des Modells für Schulung und Evaluierung aufgezeichnet. Zur Auswertung ruft der Agent automatisch die aufgezeichnete Konfiguration ab, um die angegebenen Sensoren, den Aktionsbereich und die neuronale Netzwerktechnologie zu verwenden.

In diesem Abschnitt werden Sie durch die Schritte zur Konfiguration eines Agenten in der DeepRacer AWS-Konsole geführt.

So konfigurieren Sie einen DeepRacer AWS-Agenten in der DeepRacer AWS-Konsole

1. Melden Sie sich bei der DeepRacerAWS-Konsole an.

2. Wählen Sie im primären Navigationsbereich die Option Garage aus.
3. Wenn Sie Garage zum ersten Mal verwenden, wird Ihnen das Dialogfeld WELCOME TO THE GARAGE (WILLKOMMEN IN DER GARAGE) angezeigt. Wählen Sie **>** oder **<**, durchsuchen Sie die Einführung zu verschiedenen Sensoren, die für das DeepRacer AWS-Fahrzeug unterstützt werden, oder wählen Sie X, um das Dialogfeld zu schließen. Diese einleitenden Informationen finden Sie im Hilfebereich in Garage.
4. Wählen Sie auf der Seite Garage die Option Build new vehicle (Neues Fahrzeug bauen).
5. Wählen Sie auf der Seite Mod your own vehicle (Eigenes Fahrzeug modifizieren) unter Mod specifications (Mod-Spezifikationen) einen oder mehrere Sensoren aus, um die beste Kombination zu erlernen, die Ihren beabsichtigten Renntypen entspricht.

   Um für Ihre DeepRacer AWS-Fahrzeugzeittests zu trainieren, wählen Sie Kamera. Für die Vermeidung von Hindernissen oder head-to-bot Rennen möchten Sie andere Sensortypen verwenden. Um sich für eine Stereokamera zu entscheiden, stellen Sie sicher, dass Sie eine zusätzliche Einzelobjektivkamera erworben haben. AWS DeepRacer stellt die Stereokamera aus zwei Einlinsenkameras her. Sie können entweder eine Kamera mit einem Objektiv oder eine Stereokamera mit zwei Objektiven an einem Fahrzeug haben. In beiden Fällen können Sie dem Agenten einen LiDAR-Sensor hinzufügen, wenn Sie nur möchten, dass das trainierte Modell blinde Winkel bei der Hindernisvermeidung oder head-to-bot bei Rennen erkennen und vermeiden kann.
6. Wählen Sie auf der Seite Garage und unter Neural network topologies (Neuronale Netzwerktopologien) eine unterstützte Netzwerktopologie aus.

   Im Allgemeinen ist ein tieferes neuronales Netzwerk (mit mehr Ebenen) besser geeignet, um auf komplizierteren Strecken mit zahlreichen und scharfen Kurven zu fahren, um stationäre Hindernisse zu vermeiden oder um gegen andere fahrende Fahrzeuge zu konkurrieren. Die Schulung eines tieferen neuronalen Netzwerks ist jedoch teurer und es dauert länger, bis das Modell konvergiert. Andererseits kostet ein flacheres Netzwerk (mit weniger Ebenen) weniger und benötigt eine kürzere Zeit für die Schulung. Das geschulte Modell ist in der Lage, einfachere Streckenbedingungen oder Fahranforderungen zu bewältigen, wie Time Trials auf einer hindernisfreien Strecke ohne Konkurrenten.

   Insbesondere DeepRacer unterstützt AWS 3-Layer-CNN oder 5-Layer-CNN.
7. Wählen Sie auf der Seite Garage die Option Next (Weiter), um mit der Einrichtung des Aktionsbereichs des Agenten fortzufahren.

8. Lassen Sie auf der Seite Action space (Aktionsbereich) die Standardeinstellungen für Ihre erste Schulung unverändert. Experimentieren Sie für nachfolgende Schulungen mit verschiedenen Einstellungen für Lenkwinkel, Höchstgeschwindigkeit und ihre Granularitäten. Wählen Sie anschließend Weiter.
9. Geben Sie auf der Seite Farbe Ihr Fahrzeug ein, um sich von der Masse abzuheben, einen Namen in das Feld Name your ein DeepRacer und wählen Sie dann aus der Liste der Fahrzeugfarben eine Farbe für den Agenten aus. Wählen Sie dann Submit (Senden).
10. Überprüfen Sie auf der Seite Garage die Einstellungen des neuen Agenten. Um weitere Änderungen vorzunehmen, wählen Sie Mod vehicle (Fahrzeug modifizieren) und wiederholen Sie die vorherigen Schritte ab Schritt 4.

Jetzt ist Ihr Agent bereit zur Schulung.

Maßgeschneiderte DeepRacer AWS-Schulungen für Zeitfahren

Wenn Sie AWS zum ersten Mal verwenden DeepRacer, sollten Sie mit einem einfachen Zeitfahren beginnen, um sich damit vertraut zu machen, wie Sie DeepRacer AWS-Modelle für das Fahren Ihres Fahrzeugs trainieren können. Auf diese Weise erhalten Sie eine behutsame Einführung in grundlegende Konzepte der Belohnungsfunktion sowie zu den Begriffen Agent, Umgebung usw. Ihr Ziel ist es, ein Modell zu schulen, um das Fahrzeug auf der Strecke zu halten und eine Runde so schnell wie möglich zu absolvieren. Anschließend können Sie das trainierte Modell auf Ihrem DeepRacer AWS-Fahrzeug einsetzen, um Testfahrten auf einer physischen Strecke ohne zusätzliche Sensoren zu testen.

Um ein Modell für dieses Szenario zu trainieren, können Sie den Standardagenten aus Garage auf der DeepRacer AWS-Konsole auswählen. Der Standard-Agent wurde mit einer einzigen nach vorne gerichteten Kamera, einem Standard-Aktionsbereich und einer standardmäßigen neuronalen Netzwerktopologie konfiguriert. Es ist hilfreich, mit dem Training eines DeepRacer AWS-Modells mit dem Standard-Agenten zu beginnen, bevor Sie zu anspruchsvolleren Agents übergehen.

Befolgen Sie die folgenden Empfehlungen, um Ihr Modell mit dem Standard-Agenten zu schulen.

1. Beginnen Sie die Schulung Ihres Modells mit einer einfachen Strecke mit regelmäßigeren Formen und weniger scharfen Kurven. Verwenden Sie die Standard-Belohnungsfunktion. Schulen Sie das Modell für 30 Minuten. Evaluieren Sie Ihr Modell nach dem Schulungsauftrag auf derselben Strecke, um zu sehen, ob der Agent eine Runde beenden kann.

2. Lesen Sie mehr über die Belohnungsfunktionsparameter. Setzen Sie die Schulung mit verschiedenen Anreizen fort, um den Agenten dafür zu belohnen, schneller zu fahren. Verlängern Sie die Schulungszeit für das nächste Modell auf 1 bis 2 Stunden. Vergleichen Sie die Belohnungsgrafik zwischen der ersten Schulung und dieser zweiten Schulung. Experimentieren Sie weiter, bis sich das Belohnungsdiagramm nicht mehr verbessert.
3. Lesen Sie mehr über den Aktionsbereich. Trainieren Sie das Modell zum dritten Mal, indem Sie die Höchstgeschwindigkeit erhöhen (z. B. 1 m/s). Um den Aktionsbereich zu ändern, müssen Sie unter Garage einen neuen Agenten erstellen, wenn Sie die Möglichkeit haben, die Änderung vorzunehmen. Beachten Sie bei der Aktualisierung der Höchstgeschwindigkeit Ihres Agenten Folgendes: Je höher die Höchstgeschwindigkeit, desto schneller kann der Agent die zu bewertende Strecke abschließen und desto schneller kann Ihr DeepRacer AWS-Fahrzeug eine Runde auf einer physischen Strecke beenden. Eine höhere Höchstgeschwindigkeit bedeutet jedoch oft eine längere Zeit bis zur Konvergierung der Schulung, da der Agent dabei eher in einer Kurve überschießt und von der Strecke abkommt. Möglicherweise möchten Sie die Granularitäten verringern, um dem Agenten mehr Raum zum Beschleunigen und Abbremsen zu geben oder die Belohnungsfunktion in anderer Weise zu ändern, damit die Schulung schneller konvergiert. Nachdem die Schulung konvergiert ist, evaluieren Sie das 3. Modell, um zu sehen, ob sich die Rundenzeit verbessert. Fahren Sie so fort, bis es keine Verbesserung mehr gibt.
4. Wählen Sie eine kompliziertere Strecke und wiederholen Sie Schritt 1 bis Schritt 3. Evaluieren Sie Ihr Modell auf einer Strecke, die sich von der Schulungsstrecke unterscheidet, um zu sehen, wie das Modell auf verschiedene virtuelle Strecken verallgemeinert werden kann, um Generalisierungen für reale Umgebungen durchzuführen.
5. (Optional) Experimentieren Sie mit unterschiedlichen Werten der Hyperparameter, um den Schulungsprozess zu verbessern, und wiederholen Sie Schritt 1 bis Schritt 3.
6. (Optional) Untersuchen und analysieren Sie die DeepRacer AWS-Protokolle. Beispielcode, den Sie zur Analyse der Protokolle verwenden können, finden Sie unter https://github.com/aws-samples/aws-deepracer-workshops/tree/master/log-analysis.

Maßgeschneiderte DeepRacer AWS-Schulungen für Rennen zur Objektvermeidung

Nachdem Sie sich mit Time Trials vertraut gemacht und einige konvergierte Modelle geschult haben, gehen Sie zur nächsten anspruchsvolleren Herausforderung über: zur Hindernisvermeidung. Hier ist Ihr Ziel, ein Modell zu schulen, das eine Runde so schnell wie möglich absolvieren kann, ohne von der Strecke zu geraten, und dabei Kollisionen mit Objekten auf der Strecke zu vermeiden. Dies

ist offensichtlich ein schwierigeres Problem für den Agenten und die Schulung dauert länger bis zur Konvergenz.

Die DeepRacer AWS-Konsole unterstützt zwei Arten von Hindernisvermeidungstrainings: Hindernisse können an festen oder zufälligen Stellen entlang der Strecke platziert werden. Bei festen Standorten bleiben die Hindernisse während des Schulungsauftrags an der gleichen Stelle fixiert. Bei zufälligen Standorten ändern die Hindernisse ihre jeweiligen Orte zufällig von Episode zu Episode.

Es ist einfacher für die Schulung, ortsfeste Hindernisse zu vermeiden, da das System dabei einen niedrigeren Freiheitsgrad hat. Modelle können jedoch übermäßig angepasst werden, wenn die Standortinformationen in die geschulten Modelle integriert sind. Infolgedessen können die Modelle übermäßig angepasst sein und möglicherweise schlechter generalisieren. Bei der Vermeidung zufällig positionierter Hindernisse ist die Konvergenz schwieriger zu erreichen, da der Agent weiterhin lernen muss, an Orten, die er noch nicht gesehen hat, Kollisionen mit Hindernissen zu vermeiden. Modelle, die mit dieser Option geschult wurden, neigen jedoch dazu, besser zu generalisieren, und können gut auf reale Rennen übertragen werden. Platzieren Sie zu Beginn Hindernisse an festen Standorten, machen Sie sich mit den Verhaltensweisen vertraut und gehen Sie dann zu zufälligen Standorten über.

Im DeepRacer AWS-Simulator sind die Hindernisse quaderförmige Boxen mit den gleichen Abmessungen (9,5" (L) x 15,25" (B) x 10/5" (H)) wie die Paketbox des AWS-Fahrzeugs. DeepRacer Dies macht es einfacher, das geschulte Modell vom Simulator in die reale Welt zu übertragen, wenn Sie die Verpackungsbox als Hindernis auf der physischen Spur platzieren.

Um mit der Hindernisvermeidung zu experimentieren, befolgen Sie die empfohlene Vorgehensweise, die in den folgenden Schritten beschrieben wird:

1. Verwenden Sie den Standard-Agenten oder experimentieren Sie mit neuen Sensoren und Aktionsbereichen, indem Sie einen vorhandenen Agenten anpassen oder einen neuen erstellen. Sie sollten die Höchstgeschwindigkeit auf unter 0,8 m/s und die Geschwindigkeitsgranularität auf 1 oder 2 Stufen begrenzen.

   Beginnen Sie mit der Schulung eines Modells für ca. 3 Stunden mit zwei Objekten an festen Standorten. Verwenden Sie die Beispiel-Belohnungsfunktion und schulen Sie das Modell auf der Strecke, auf der Sie fahren werden, oder auf einer Strecke, die dieser ähnelt. Der AWS DeepRacer Smile Speedway (Intermediate) ist eine einfache Strecke, weshalb er sich hervorragend für die Vorbereitung von Gipfelrennen eignet. Evaluieren Sie das Modell auf der gleichen Strecke mit der gleichen Anzahl von Hindernissen. Beobachten Sie, wie die gesamte erwartete Belohnung konvergiert, wenn überhaupt.

2. Lesen Sie mehr über die Belohnungsfunktionsparameter. Experimentieren Sie mit Variationen Ihrer Belohnungsfunktion. Erhöhen Sie die Zahl der Hindernisse auf vier. Schulen Sie den Agenten, um zu sehen, ob die Schulung in der gleichen Schulungszeit konvergiert. Wenn dies nicht der Fall ist, optimieren Sie Ihre Belohnungsfunktion erneut, senken Sie die Höchstgeschwindigkeit oder reduzieren Sie die Anzahl der Hindernisse und schulen Sie den Agenten erneut. Wiederholen Sie die Experimente, bis es keine signifikante Verbesserung mehr gibt.
3. Gehen Sie nun zur Schulung zur Vermeidung von Hindernissen an zufälligen Orten über. Sie müssen den Agenten mit zusätzlichen Sensoren konfigurieren, die in Garage in der DeepRacer AWS-Konsole verfügbar sind. Sie können eine Stereokamera verwenden. Oder Sie können eine LiDAR-Einheit entweder mit einer Kamera mit einem Objektiv oder einer Stereokamera kombinieren. Sie sollten dabei jedoch eine längere Schulungszeit einkalkulieren. Stellen Sie den Aktionsbereich auf eine relativ niedrige Höchstgeschwindigkeit (z. B. 2 m/s) ein, damit das Training schneller zusammenläuft. Verwenden Sie für die Netzwerkarchitektur ein flaches neuronales Netzwerk, das als für die Vermeidung von Hindernissen ausreichend befunden wurde.
4. Beginnen Sie die Schulung mit vier Stunden für den neuen Agenten zur Hindernisvermeidung mit vier zufällig platzierten Objekten auf einer einfachen Strecke. Evaluieren Sie dann Ihr Modell auf der gleichen Strecke, um zu sehen, ob es Runden mit zufällig positionierten Hindernissen absolvieren kann. Wenn dies nicht der Fall ist, können Sie Ihre Belohnungsfunktion optimieren, andere Sensoren ausprobieren und längere Schulungszeiten verwenden. Ein weiterer Tipp: Sie können versuchen, ein bestehendes Modell zu klonen, um die Schulung fortzusetzen und dafür zuvor erlernte Erfahrungen zu nutzen.
5. (Optional) Wählen Sie eine höhere Höchstgeschwindigkeit für den Aktionsbereich oder lassen Sie mehr Hindernisse zufällig auf der Strecke platzieren. Experimentieren Sie mit verschiedenen Sensoren und optimieren Sie die Belohnungsfunktionen und Hyperparameterwerte. Experimentieren Sie mit der 5-layer CNN-Netzwerktopologie. Schulen Sie das Modell anschließend neu, um zu bestimmen, wie sich dies auf die Konvergenz der Schulung auswirkt.

Maßgeschneiderte DeepRacer AWS-Schulungen für head-to-bot Rennen

Nachdem du Trainingshindernisse gemeistert hast, bist du nun bereit, dich der nächsten Herausforderung zu stellen: Trainingsmodelle für head-to-bot Rennen. Im Gegensatz zu Veranstaltungen zur Hindernisvermeidung herrscht im head-to-bot Rennsport ein dynamisches Umfeld mit sich bewegenden Fahrzeugen. Ihr Ziel ist es, Modelle für Ihr DeepRacer AWS-Fahrzeug so zu trainieren, dass sie gegen andere fahrende Fahrzeuge antreten, um als Erster die Ziellinie

zu erreichen, ohne von der Strecke abzukommen oder mit anderen Fahrzeugen zu kollidieren. In der DeepRacer AWS-Konsole können Sie ein head-to-bot Rennmodell trainieren, indem Sie Ihren Agenten gegen 1—4 Bot-Fahrzeuge antreten lassen. Allgemein sollten Sie dabei mehr Hindernisse auf einer längeren Strecke platzieren.

Jedes Bot-Fahrzeug folgt einem vordefinierten Weg mit konstanter Geschwindigkeit. Sie können Ihrem Fahrzeug möglichen, Fahrspuren zu wechseln oder es auf seiner Startspur bleiben lassen. Ähnlich wie bei der Schulung zur Vermeidung von Hindernissen können Sie die Bot-Fahrzeuge gleichmäßig über die Strecke auf beiden Spuren verteilen. Die Konsole beschränkt Sie auf bis zu 4 Bot-Fahrzeuge auf der Strecke. Mehr konkurrierende Fahrzeuge auf der Strecke geben dem Lernagenten mehr Möglichkeiten, unterschiedlichen Situationen mit anderen Fahrzeugen zu begegnen. Auf diese Weise lernt es mehr in einem Schulungsauftrag und der Agent wird schneller geschult. Allerdings wird jede Schulung wahrscheinlich länger dauern, bis Konvergenz erreicht ist.

Um einen Agenten mit Bot-Fahrzeugen zu schulen, sollten Sie die Höchstgeschwindigkeit des Aktionsbereichs des Agenten höher einstellen als die (konstante) Geschwindigkeit der Bot-Fahrzeuge, so dass der Agent während der Schulung mehr Überholmöglichkeiten hat. Als guten Ausgangspunkt sollten Sie die Höchstgeschwindigkeit des Agenten auf 0,8 m/s und die Bewegungsgeschwindigkeit des Bot-Fahrzeugs auf 0,4 m/s einstellen. Wenn Sie es den Bots ermöglichen, die Fahrspuren zu wechseln, wird die Schulung schwieriger, weil der Agent nicht nur lernen muss, wie man eine Kollision mit einem fahrenden Fahrzeug vorne auf der gleichen Spur vermeiden kann, sondern auch, wie man eine Kollision mit einem anderen fahrenden Fahrzeug vorne auf der anderen Spur vermeiden kann. Sie können die Bots so einstellen, dass sie die Spuren in zufälligen Intervallen ändern. Die Länge eines Intervalls wird zufällig aus einem Zeitbereich (z. B. 1 bis 5 Sekunden) ausgewählt, den Sie vor Beginn des Trainingsjobs angeben. Dieses Verhalten beim Spurwechsel ähnelt eher dem realen head-to-bot Rennverhalten, und der geschulte Agent sollte bessere Ergebnisse erzielen. Es dauert jedoch länger, bis das Modell zur Konvergenz kommt.

Befolge diese empfohlenen Schritte, um dein Training für den Rennsport zu wiederholen: head-to-bot

1. Erstellen Sie in der Garage der DeepRacer AWS-Konsole einen neuen Trainingsagenten, der sowohl mit Stereokameras als auch mit einer LiDAR-Einheit konfiguriert ist. Es ist möglich, ein relativ gutes Modell nur mit einer Stereokamera gegen Bot-Fahrzeuge zu schulen. LiDAR hilft dabei, blinde Flecken zu reduzieren, wenn der Agent die Fahrspuren wechselt. Stellen Sie die Höchstgeschwindigkeit nicht zu hoch ein. Ein guter Ausgangspunkt ist 1 m/s.
2. Beginnen Sie mit zwei Bot-Fahrzeugen, um für den head-to-bot Rennsport zu trainieren. Stellen Sie die Bewegungsgeschwindigkeit des Bots niedriger als die Höchstgeschwindigkeit Ihres Agenten ein (z. B. 0,5 m/s, wenn die Höchstgeschwindigkeit des Agenten 1 m/s beträgt).

Deaktivieren Sie die Option „Spurwechsel“ und wählen Sie dann den gerade erstellten Schulungsagent aus. Verwenden Sie eines der Beispiele für Belohnungsfunktionen oder nehmen Sie die minimal notwendigen Änderungen vor und schulen Sie dann für drei Stunden. Verwenden Sie die Strecke, auf der Sie fahren werden, oder eine Strecke, die genau dieser Strecke entspricht. Der AWS DeepRacer Smile Speedway (Intermediate) ist eine einfache Strecke, weshalb er sich hervorragend für die Vorbereitung von Gipfelrennen eignet. Nachdem die Schulung abgeschlossen ist, evaluieren Sie das geschulte Modell auf derselben Strecke.

3. Für anspruchsvollere Aufgaben können Sie Ihr trainiertes Modell für ein zweites head-to-bot Rennmodell klonen. Experimentieren Sie entweder mit mehr Bot-Fahrzeugen oder aktivieren Sie die Spurwechseloptionen. Beginnen Sie mit langsamen Spurwechselvorgängen in zufälligen Intervallen, die länger als zwei Sekunden sind. Sie können auch mit benutzerdefinierten Belohnungsfunktionen experimentieren. Im Allgemeinen kann Ihre benutzerdefinierte Belohnungsfunktionslogik mit der zur Vermeidung von Hindernissen vergleichbar sein, wenn Sie nicht den Ausgleich zwischen dem Überholen anderer Fahrzeuge und dem Verbleiben auf der Strecke berücksichtigen. Je nachdem, wie gut das vorherige Modell ist, müssen Sie möglicherweise weitere drei bis sechs Stunden schulen. Evaluieren Sie Ihre Modelle und prüfen Sie, wie gut das Modell funktioniert.

Trainieren und evaluieren Sie DeepRacer AWS-Modelle mit der DeepRacer AWS-Konsole

Um ein Reinforcement-Learning-Modell zu trainieren, können Sie die DeepRacer AWS-Konsole verwenden. Erstellen Sie in der Konsole einen Trainingsjob, wählen Sie ein unterstütztes Framework und einen verfügbaren Algorithmus aus, fügen Sie eine Belohnungsfunktion hinzu und konfigurieren Sie die Trainingseinstellungen. Sie können das Training in einem Simulator beobachten. Die step-by-step Anweisungen finden Sie unterthe section called “Trainiere dein erstes Modell”.

In diesem Abschnitt wird erklärt, wie ein DeepRacer AWS-Modell trainiert und bewertet wird. Er zeigt außerdem, wie man eine Belohnungsfunktion erstellt und verbessert, wie ein Aktionsraum die Modellleistung beeinflusst und wie Hyperparameter die Trainingsleistung beeinflussen. Sie erfahren zudem, wie Sie ein Trainingsmodell klonen können, um eine Trainingseinheit zu erweitern, wie Sie den Simulator zur Evaluierung der Trainingsleistung verwenden und wie Sie einige der Simulationen an die Herausforderungen der realen Welt anpassen können.

Themen

- Erstellen Sie Ihre Prämienfunktion

- Erkunden Sie den Aktionsraum, um ein robustes Modell zu trainieren
- Passen Sie Hyperparameter systematisch an
- Untersuchen Sie den Fortschritt der DeepRacer AWS-Schulungen
- Klonen Sie ein trainiertes Modell, um einen neuen Trainingspass zu starten
- Evaluieren Sie DeepRacer AWS-Modelle in Simulationen
- Optimieren Sie DeepRacer AWS-Schulungsmodelle für reale Umgebungen

Erstellen Sie Ihre Prämienfunktion

Eine Belohnungsfunktion beschreibt sofortiges Feedback (als Prämien- oder Strafpunktzahl), wenn sich Ihr DeepRacer AWS-Fahrzeug von einer Position auf der Strecke zu einer neuen Position bewegt. Der Zweck der Funktion ist es, das Fahrzeug dazu zu bringen, ein Ziel durch Bewegungen entlang der Strecke schnell und ohne Unfall oder Regelverstöße zu erreichen. Eine wünschenswerte Bewegung bringt eine bessere Bewertung für die entsprechende Aktion oder den Zielstatus. Eine illegale oder unnötige Bewegung führt zu einer niedrigeren Bewertung. Beim Training eines DeepRacer AWS-Modells ist die Prämienfunktion der einzige anwendungsspezifische Teil.

Im Allgemeinen gestalten Sie Ihre Belohnungsfunktion so, dass sie wie ein Incentive-Plan funktioniert. Verschiedene Incentive-Strategien können zu einem unterschiedlichen Fahrzeugverhalten führen. Um die Fahrt des Fahrzeugs zu beschleunigen, sollte die Funktion das Folgen der Strecke belohnen. Die Funktion sollte Strafen verteilen, wenn das Fahrzeug zum Beenden einer Runde zu lange braucht oder die Strecke verlässt. Um Zickzack-Fahrmuster zu vermeiden, könnte das Fahrzeug dafür belohnt werden, auf geraden Abschnitten der Strecke weniger zu lenken. Die Belohnungsfunktion kann positive Werte liefern, wenn das Fahrzeug bestimmte Meilensteine passiert (über `waypoints`). Dies könnte verhindern, dass das Fahrzeug abwartet oder in die falsche Richtung fährt. Vermutlich werden Sie die Belohnungsfunktion verändern, um die Streckenbedingungen zu berücksichtigen. Je stärker Ihre Belohnungsfunktion umgebungsspezifische Informationen berücksichtigt, desto wahrscheinlicher ist es jedoch, dass Ihr trainiertes Modell zu stark angepasst und weniger universell einsatzbar ist. Um Ihr Modell universeller einsetzbar zu gestalten, können Sie Aktionsräume ausprobieren.

Wenn ein Incentive-Plan nicht sorgfältig geprüft wird, kann er zu unbeabsichtigten Folgen mit gegenteiliger Wirkung führen. Dies liegt daran, dass das direkte Feedback eine notwendige, aber nicht ausreichende Voraussetzung für das Reinforcement Learning ist. Eine individuelle direkte Belohnung allein zeigt nicht, ob eine Aktion wünschenswert ist. An einer bestimmten Position kann eine Aktion beispielsweise eine hohe Belohnung erzielen. Eine nachfolgende Aktion könnte das

Fahrzeug von der Strecke führen und eine niedrige Bewertung erzielen. In diesem Fall sollte das Fahrzeug die Bewegung mit der höchsten Bewertung an dieser Position vermeiden. Erst wenn alle zukünftigen Bewegungen von einer bestimmten Position zu einer hohen Durchschnittsbewertung führen, sollte der Wechsel zur nächsten Position als wünschenswert erachtet werden. Das zukünftige Feedback wird niedriger bewertet. So werden weniger zukünftige Bewegungen oder Positionen in die Berechnung der durchschnittlichen Belohnung einbezogen.

Zum Erstellen einer Belohnungsfunktion empfiehlt es sich, mit einer einfachen Funktion zu beginnen, die grundlegende Szenarien abdeckt. Sie können die Funktion erweitern, um mehr Aktionen zu verarbeiten. Schauen wir uns nun einige einfache Belohnungsfunktionen an.

Themen

- Einfache Beispiele für Belohnungsfunktionen
- Verbessern Sie Ihre Belohnungsfunktion

Einfache Beispiele für Belohnungsfunktionen

Wir können mit der Erstellung der Belohnungsfunktion beginnen, indem wir zuerst die grundlegendste Situation berücksichtigen. Das ist eine Situation, in der von Anfang bis Ende auf einer geraden Strecke gefahren wird, ohne die Strecke zu verlassen. In diesem Szenario hängt die Logik der Belohnungsfunktion nur von `on_track` und `progress` ab. Als Versuch könnten Sie mit der folgenden Logik beginnen:

```
def reward_function(params):
    if not params["all_wheels_on_track"]:
        reward = -1
    else if params["progress"] == 1 :
        reward = 10
    return reward
```

Diese Logik bestraft den Agenten, wenn er die Strecke verlässt. Sie belohnt den Agenten, wenn er zur Ziellinie fährt. Es ist sinnvoll, das gesteckte Ziel zu erreichen. Der Agent bewegt sich jedoch frei zwischen dem Startpunkt und der Ziellinie (einschließlich der Rückwärtsfahrt auf der Strecke). Das Training könnte nicht nur lange dauern, sondern das trainierte Modell würde beim Einsatz in einem realen Fahrzeug auch zu einer weniger effizienten Fahrweise führen.

In der Praxis lernt ein Mitarbeiter effektiver, wenn er dies bit-by-bit im Laufe der Schulung tun kann. Dies bedeutet, dass eine Belohnungsfunktion Schritt für Schritt auf der Strecke kleinere Belohnungen

ausgeben sollte. Damit der Agent auf der geraden Strecke fährt, können wir die Belohnungsfunktion wie folgt verbessern:

```
def reward_function(params):
    if not params["all_wheels_on_track"]:
        reward = -1
    else:
        reward = params["progress"]
    return reward
```

Mit dieser Funktion erhält der Agent umso mehr Belohnung, je näher er die Ziellinie erreicht. Dies sollte unproduktive Versuche der Rückwärtsfahrt reduzieren oder beseitigen. Im Allgemeinen wollen wir, dass die Belohnungsfunktion die Belohnung gleichmäßiger über den Aktionsraum verteilt. Das Entwickeln einer effektiven Belohnungsfunktion kann eine Herausforderung sein. Sie sollten mit einer einfachen Funktion beginnen und diese schrittweise erweitern oder verbessern. Durch systematisches Experimentieren kann die Funktion stabiler und effizienter werden.

Verbessern Sie Ihre Belohnungsfunktion

Nachdem Sie Ihr DeepRacer AWS-Modell erfolgreich für die einfache gerade Strecke trainiert haben, kann das DeepRacer AWS-Fahrzeug (virtuell oder physisch) selbstständig fahren, ohne von der Strecke abzukommen. Wenn Sie das Fahrzeug auf einer Ringstrecke fahren lassen, bleibt es nicht auf der Strecke. Die Belohnungsfunktion hat die Aktionen ignoriert, um Kurven zu lenken und so der Strecke zu folgen.

Damit Ihr Fahrzeug diese Aktionen ausführen kann, müssen Sie die Belohnungsfunktion erweitern. Die Funktion muss eine Belohnung gewähren, wenn der Agent eine zulässige Drehung macht, und eine Strafe vorsehen, wenn der Agent eine unzulässige Drehung macht. Dann sind Sie bereit, eine weitere Trainingsrunde zu beginnen. Um die Vorteile des vorherigen Trainings zu nutzen, können Sie das neue Training beginnen, indem Sie das vorher trainierte Modell klonen und so das zuvor erlernte Wissen weitergeben. Sie können diesem Muster folgen, um die Prämienfunktion schrittweise um weitere Funktionen zu erweitern und Ihr DeepRacer AWS-Fahrzeug darauf vorzubereiten, in immer komplexeren Umgebungen zu fahren.

Weitere erweiterte Belohnungsfunktionen finden Sie in den folgenden Beispielen:

- the section called “Beispiel 1: Folgen Sie beim Zeitfahren der Mittellinie”
- the section called “Beispiel 2: Bleiben Sie beim Zeitfahren innerhalb der beiden Grenzen”
- the section called “Beispiel 3: Vermeiden Sie Zick-Zack-Bewegungen bei Zeitfahren”

- the section called “Beispiel 4: Bleiben Sie auf einer Spur, ohne gegen stehende Hindernisse oder fahrende Fahrzeuge zu stoßen”

Erkunden Sie den Aktionsraum, um ein robustes Modell zu trainieren

In der Regel sollten Sie Ihr Modell so zuverlässig wie möglich trainieren. Nur so können Sie es in möglichst vielen Umgebungen einsetzen. Ein zuverlässiges Modell ist ein Modell, das in einer Vielzahl von Streckenformen und -bedingungen eingesetzt werden kann. Ein zuverlässiges Modell ist grundsätzlich nicht "intelligent", denn seine Belohnungsfunktion deckt kein explizites, umgebungsspezifisches Wissen ab. Wenn dies der Fall wäre, würde Ihr Modell wahrscheinlich nur in einer Umgebung anwendbar sein, die der trainierten Umgebung ähnlich ist.

Die explizite Einbeziehung umgebungsspezifischer Informationen in die Belohnungsfunktion wird als Feature-Engineering (Funktionalitätsentwicklung) bezeichnet. Mit dem Feature-Engineering können Sie die Trainingszeit verkürzen. Sie können es für Lösungen einsetzen, die auf eine bestimmte Umgebung zugeschnitten sind. Um ein allgemein einsetzbares Modell zu trainieren, sollten Sie jedoch auf den großzügigen Einsatz von Feature-Engineering verzichten.

Wenn Sie beispielsweise ein Modell auf einer Rundstrecke trainieren, können Sie kein trainiertes Modell erwarten, das auf einer nicht kreisförmige Strecke einsetzbar ist (sofern Sie entsprechende geometrische Eigenschaften in die Belohnungsfunktion integriert haben).

Wie würden Sie ein Modell so zuverlässig wie möglich trainieren und gleichzeitig die Belohnungsfunktion so einfach wie möglich halten? Eine Methode besteht darin, den Aktionsraum mit den möglichen, vom Agenten durchführbaren Aktionen zu analysieren. Eine weitere ist das Experimentieren mit Hyperparametern für den zugrunde liegenden Trainingsalgorithmus. Meistens kommen beide Methoden zur Anwendung. Hier konzentrieren wir uns darauf, wie Sie den Aktionsraum erkunden können, um ein robustes Modell für Ihr DeepRacer AWS-Fahrzeug zu trainieren.

Beim Training eines DeepRacer AWS-Modells ist eine Aktion (a) eine Kombination aus Geschwindigkeit (tMeter pro Sekunde) und Lenkwinkel (sin Grad). Der Aktionsraum des Agenten definiert die möglichen Geschwindigkeitsbereiche und Lenkwinkel des Agenten. Für einen diskreten Aktionsraum von m Geschwindigkeiten ($(v_1, .., v_n)$) und n Lenkwinkeln ($(s_1, .., s_m)$) umfasst der Aktionsraum m*n mögliche Aktionen:

a_1: (v_1, s_1)
...
a_n: (v_1, s_n)

```
...
a(i-1)*n+j:   (vi, sj)
...

a(m-1)*n+1:   (vm, s1)
...
am*n:          (vm, sn)
```

Die tatsächlichen Werte von (v_i, s_j) hängen von den Bereichen von v_{max} und $|s_{max}|$ ab und sind nicht einheitlich verteilt.

Jedes Mal, wenn Sie mit dem Training oder der Iteration Ihres DeepRacer AWS-Modells beginnen, müssen Sie zunächst die $n$$m$, v_{max} und $|s_{max}|$ oder angeben, dass deren Standardwerte verwendet werden. Basierend auf Ihrer Wahl generiert der DeepRacer AWS-Service die verfügbaren Aktionen, die Ihr Agent in der Schulung auswählen kann. Die erzeugten Aktionen sind nicht gleichmäßig über den Aktionsraum verteilt.

Im Allgemeinen geben eine größere Anzahl von Aktionen und größere Aktionsbereiche Ihrem Agenten mehr Raum oder Möglichkeiten, um auf unterschiedlichste Streckenbedingungen zu reagieren (z. B. eine Kurvenstrecke mit unregelmäßigen Kurvenwinkeln oder -richtungen). Je mehr Optionen dem Agenten zur Verfügung stehen, desto besser kann er mit Streckenvariationen umgehen. Infolgedessen können Sie erwarten, dass das trainierte Modell auch mit einer einfachen Belohnungsfunktion umfassender einsetzbar ist.

So kann Ihr Agent mithilfe eines grob eingeteilten Aktionsraums mit wenigen Geschwindigkeiten und Lenkwinkeln beispielsweise den Umgang mit einer geradlinigen Strecke schnell erlernen. Bei einer Strecke mit Kurven wird dieser grob eingeteilte Aktionsraum jedoch vermutlich dazu führen, dass der Agent bei Kurvenfahrten über die Streckenbegrenzung hinausfährt. Dies liegt daran, dass ihm nicht genügend Möglichkeiten zur Verfügung stehen, um die Geschwindigkeit oder die Lenkung zu ändern. Wenn Sie die Anzahl der Geschwindigkeiten oder der Lenkwinkel bzw. beides erhöhen, sollte der Agent in der Lage sein, durch Kurven zu manövrieren und innerhalb der Begrenzungen zu bleiben. Wenn sich Ihr Agent im Zickzack bewegt, können Sie außerdem versuchen, die Anzahl der Lenkbereiche zu erhöhen. So können Sie das Auftreten extremer Drehungen bei einem einzelnen Schritt reduzieren.

Wenn der Aktionsraum zu groß ist, kann die Trainingsleistung leiden. Denn in diesem Fall kann die Erkundung des Aktionsraums länger dauern. Stellen Sie sicher, dass Sie die Vorteile der allgemeinen Einsetzbarkeit eines Modells mit den Anforderungen an die Trainingsleistung in Einklang bringen. Diese Optimierung umfasst systematisches Experimentieren.

Passen Sie Hyperparameter systematisch an

Eine Möglichkeit zur Verbesserung der Leistung Ihres Modells besteht darin, einen besseren oder effektiveren Trainingsprozess zu implementieren. Um ein zuverlässiges Modell zu erhalten, muss das Training beispielsweise mehr oder weniger gleichmäßig über den Aktionsraum des Agenten verteilte Stichproben generieren. Dies erfordert eine ausreichende Mischung aus Erkundung und Förderung. Zu den diesbezüglichen Variablen gehören die Menge der verwendeten Trainingsdaten (`number of episodes between each training` und `batch size`), die Lerngeschwindigkeit des Agenten (`learning rate`) und der Erkundungsanteil (`entropy`). Um das Training praxisnah zu gestalten, können Sie den Lernprozess beschleunigen. Zu den entsprechenden Variablen gehören `learning rate`, `batch size`, `number of epochs` und `discount factor`.

Die Variablen mit Einfluss auf den Trainingsprozess werden als Hyperparameter des Trainings bezeichnet. Diese Algorithmusattribute stellen keine Eigenschaften des zugrunde liegenden Modells dar. Leider sind Hyperparameter empirischer Natur. Ihre optimalen Werte sind nicht für alle Anwendungszwecke bekannt und erfordern systematisches Experimentieren.

Bevor wir uns mit den Hyperparametern befassen, die angepasst werden können, um die Leistung beim Training Ihres DeepRacer AWS-Modells zu optimieren, wollen wir die folgende Terminologie definieren.

Datenpunkt

Ein Datenpunkt (auch Erfahrung) ist ein Tupel von (s,a,r,s'), wobei s für eine Beobachtung (oder einen Zustand) steht, die von der Kamera erfasst wird. a steht für eine vom Fahrzeug durchgeführte Aktion. r steht für die erwartete Belohnung, die durch die Aktion ausgelöst wird. s' steht für die neue Beobachtung, nachdem die Aktion durchgeführt wurde.

Episode

Eine Episode ist ein Zeitraum, in dem das Fahrzeug von einem bestimmten Ausgangspunkt aus startet und die Strecke an deren Ende abschließt bzw. die Strecke verlässt. Sie stellt eine Abfolge von Erfahrungen dar. Episoden können unterschiedliche Längen haben.

Erfahrungspuffer

Ein Erfahrungspuffer besteht aus einer Reihe von geordneten Datenpunkten, die über eine feste Anzahl von Episoden unterschiedlicher Länge während des Trainings gesammelt werden. Für AWS entspricht es den Bildern DeepRacer, die von der an Ihrem DeepRacer AWS-Fahrzeug montierten Kamera aufgenommen wurden, und den vom Fahrzeug ausgeführten Aktionen. Es

dient als Quelle, aus der Eingaben für die Aktualisierung der zugrunde liegenden (Richtlinien- und Wert-) neuronalen Netzwerke gezogen werden.

Stapel

Ein Batch ist eine geordnete Liste von Erfahrungen, die einen Teil der Simulation über einen bestimmten Zeitraum darstellen und zur Aktualisierung der Gewichtung der Strategienetzwerke verwendet werden. Es handelt sich um eine Teilmenge des Erfahrungspuffers.

Trainingsdaten

Trainingsdaten sind eine Reihe von Batches, die nach dem Zufallsprinzip aus einem Erfahrungspuffer entnommen und für das Training der Gewichtung der Strategienetzwerke verwendet werden.

Algorithmische Hyperparameter und ihre Auswirkungen

Hyperparameter	Beschreibung
Gradientenabstieg-Batchgröße	Die Anzahl der letzten Fahrzeugerfahrungen, die zufällig aus einem Erfahrungspuffer entnommen und zur Aktualisierung der zugrunde liegenden Gewichtung der neuronalen Deep-Learning-Netze verwendet wurden. Zufallsstichproben helfen, die in den Eingangsdaten enthaltenen Korrelationen zu reduzieren. Verwenden Sie eine größere Batchgröße, um stabilere und reibungslosere Aktualisierungen der Gewichtungen der neuronalen Netze zu fördern. Bedenken Sie jedoch, dass das Training länger dauern bzw. langsamer sein kann. Erforderlich Ja Zulässige Werte Positive ganze Zahl von `(32, 64, 128, 256, 512)` Standardwert `64`
Number of epochs	Die Anzahl der Durchläufe durch die Trainingsdaten zur Aktualisierung der Gewichtungen des neuronalen Netzes während des Gradientenabstiegs. Die

Hyperparameter	Beschreibung
(Anzahl der Epochen)	Trainingsdaten entsprechen Stichproben aus dem Erfahrungspuffer. Verwenden Sie eine größere Anzahl von Epochen, um zuverlässigere Aktualisierungen zu fördern. Bedenken Sie jedoch, dass das Training länger dauern bzw. langsamer sein kann. Wenn die Batchgröße klein ist, können Sie eine kleinere Epochenanzahl verwenden. Erforderlich Nein Zulässige Werte Positive ganze Zahl zwischen `[3 - 10]` Standardwert `3`
Learning rate (Lernrate)	Bei jeder Aktualisierung kann ein Teil der neuen Gewichtung aus dem Gradientenabstieg (oder Aufstieg) stammen. Der Rest kann dem vorhandenen Gewichtungswert entnommen werden. Die Lernrate steuert, wie stark ein Gradientenabstieg (oder Aufstieg) zur Gewichtung der Netze beiträgt. Verwenden Sie eine höhere Lernrate, um mehr Daten aus dem Gradientenabstieg einzubeziehen. So wird das Training schneller. Bedenken Sie jedoch, dass sich die erwartete Belohnung möglicherweise nicht annähert, wenn die Lernrate zu hoch ist. Erforderlich Nein Zulässige Werte Zahl zwischen `0.00000001` (oder 10^{-8}) und `0.001` (oder 10^{-3}) Standardwert `0.0003`

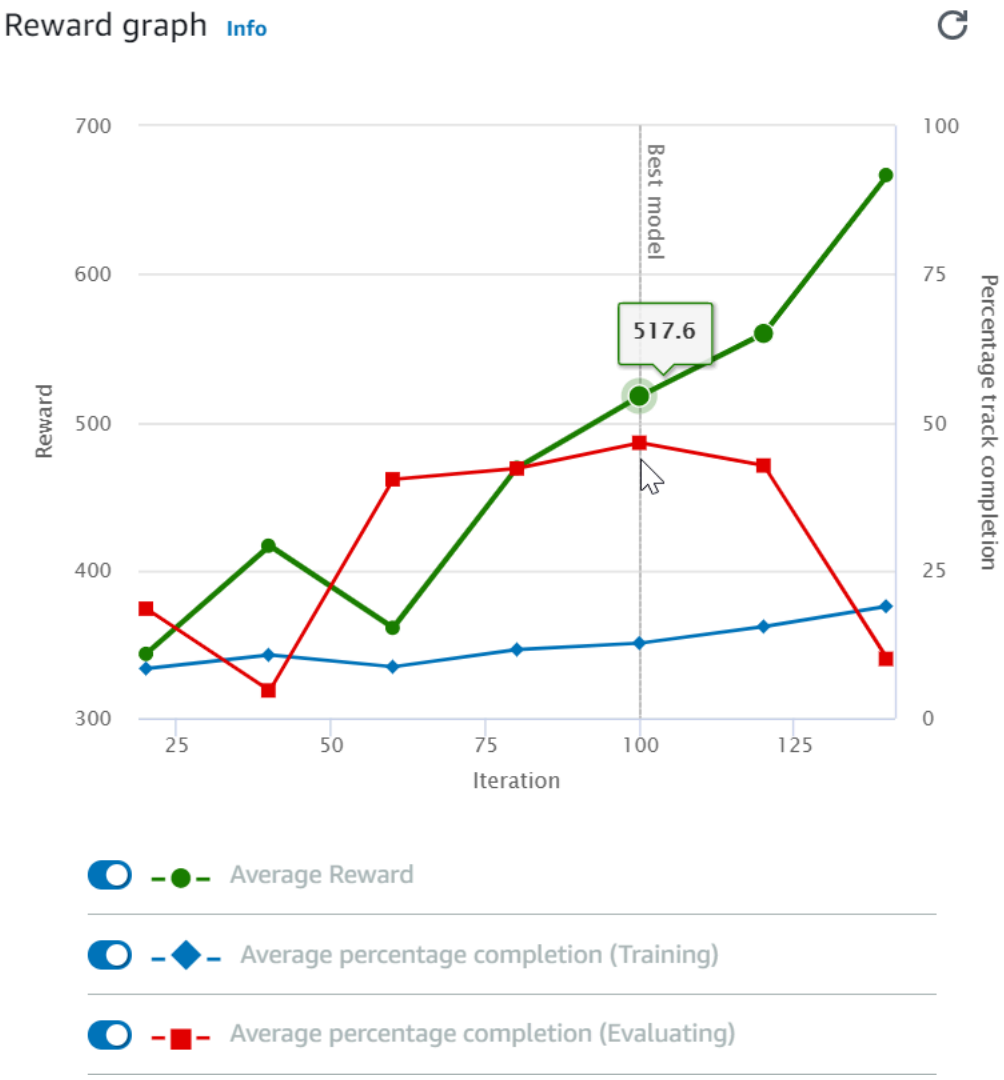
Hyperparameter	Beschreibung
`Entropy`	Ein gewisses Maß an Unsicherheit, das für einen Zufallsfaktor in der Strategieverteilung verwendet wird. Die zusätzliche Unsicherheit hilft dem DeepRacer AWS-Fahrzeug, den Aktionsraum umfassender zu erkunden. Ein größerer Entropiewert veranlasst das Fahrzeug, den Aktionsraum gründlicher zu erforschen. Erforderlich Nein Zulässige Werte Zahl zwischen `0` und `1`. Standardwert `0.01`
Discount factor (Rabattfaktor)	Ein Faktor gibt an, wie viele der zukünftigen Belohnungen zu der erwarteten Belohnung beitragen. Je größer der Discount factor (Abschlagfaktor)-Wert ist, desto weiter entfernt liegen die Beiträge, die das Fahrzeug für eine Bewegung berücksichtigt, und desto langsamer ist das Training. Mit dem Abschlagfaktor 0,9 berücksichtigt das Fahrzeug Belohnungen aus einer Schrittlänge von 10 für zukünftige Bewegungen. Mit dem Abschlagfaktor 0,999 berücksichtigt das Fahrzeug Belohnungen aus einer Schrittlänge von 1000 für zukünftige Bewegungen. Die empfohlenen Werte für den Abschlagfaktor sind 0,99, 0,999 und 0,9999. Erforderlich Nein Zulässige Werte Zahl zwischen `0` und `1`. Standardwert `0.999`

Hyperparameter	Beschreibung
Loss type (Schadensart)	Typ der Zielfunktion, mit der die Netzwerkgewichte aktualisiert werden. Ein guter Trainingsalgorithmus sollte schrittweise Änderungen an der Strategie des Agenten vornehmen, sodass er allmählich von zufälligen Aktionen zu strategischen Aktionen übergeht, um die Belohnung zu erhöhen. Aber wenn es eine zu große Veränderung bewirkt, dann wird das Training instabil und der Agent lernt am Ende nicht. Die Typen Huber loss (Huber-Verlust) und Mean squared error loss (Mittlerer quadratischer Fehlerverlust) verhalten sich für kleine Updates ähnlich. Aber wenn die Updates größer werden, nimmt Huber loss (Huber-Verlust) kleinere Inkremente im Vergleich zu Mean squared error loss (Mittlerer quadratischer Fehlerverlust) vor. Wenn Sie Konvergenzprobleme haben, verwenden Sie den Typ Huber loss (Huber-Verlust). Wenn die Konvergenz gut ist und Sie schneller trainieren wollen, verwenden Sie den Typ Mean squared error loss (Mittlerer quadratischer Fehlerverlust). Erforderlich Nein Zulässige Werte (Huber loss (Huber-Verlust), Mean squared error loss (Mittlerer quadratischer Fehlerverlust)) Standardwert Huber loss (Huber-Verlust)

Hyperparameter	Beschreibung
Anzahl der Erfahrungsepisoden zwischen den einzelnen Iterationen zur Strategieaktualisierung	Die Größe des Erfahrungspuffers, aus dem Trainingsdaten zum Lernen der Gewichtung der Strategienetzwerke entnommen werden. Eine Episode ist ein Zeitraum, in dem der Agent von einem bestimmten Ausgangspunkt aus startet und die Strecke an deren Ende abschließt bzw. die Strecke verlässt. Sie besteht aus einer Reihe von Erfahrungen. Episoden können unterschiedliche Längen haben. Bei einfachen Reinforcement-Learning-Problemen kann ein kleiner Erfahrungspuffer ausreichen. Der Lernprozess ist schnell. Bei komplexeren Problemen mit mehr lokalen Maximallösungen ist ein größerer Erfahrungspuffer erforderlich, um mehr unkorrelierte Datenpunkte bereitzustellen. In diesem Fall ist das Training langsamer, aber stabiler. Die empfohlenen Werte sind 10, 20 und 40. Erforderlich Nein Zulässige Werte Ganzzahl zwischen `5` und `100` Standardwert `20`

Untersuchen Sie den Fortschritt der DeepRacer AWS-Schulungen

Nachdem Sie Ihren Trainingsauftrag gestartet haben, können Sie die Trainingsmetriken der Belohnungen und des Streckenabschlusses pro Episode prüfen, um die Leistung des Trainingsauftrags Ihres Modells zu ermitteln. Auf der DeepRacer AWS-Konsole werden die Metriken im Prämiendiagramm angezeigt, wie in der folgenden Abbildung dargestellt.



Sie können die pro Episode erworbene Belohnung, die gemittelte Belohnung pro Iteration, den Fortschritt pro Episode, den gemittelten Fortschritt pro Iteration oder eine beliebige Kombination hiervon anzeigen. Schalten Sie dazu die Schalter Reward (Episode, Average) (Belohnung (Episode, Durchschnitt)) oder Progress (Episode, Average) (Fortschritt (Episode, Durchschnitt)) unten im Reward graph (Belohnungsdiagramm)um. Die Belohnung und der Fortschritt pro Episode werden als Streudiagramme in verschiedenen Farben angezeigt. Die gemittelte Belohnung und der Streckenabschluss werden in Liniendiagrammen angezeigt und beginnen nach der ersten Iteration.

Der Belohnungsbereich wird auf der linken Seite des Diagramms angezeigt und der Fortschrittsbereich (0 - 100) auf der rechten Seite. Um den genauen Wert einer Trainingsmetrik zu lesen, bewegen Sie den Mauszeiger in die Nähe des Datenpunkts auf dem Diagramm.

Die Diagramme werden während des Trainings automatisch alle 10 Sekunden aktualisiert. Sie können die Schaltfläche „Refresh (Aktualisieren)” auswählen, um die Metrikanzeige manuell zu aktualisieren.

Ein Trainingsauftrag ist gut, wenn die gemittelte Belohnung und der Streckenabschluss die Tendenz zeigen, zu konvergieren. Insbesondere ist das Modell wahrscheinlich konvergiert, wenn der Fortschritt pro Episode kontinuierlich 100 % erreicht und die Belohnung ausgeglichen ist. Wenn dies nicht der Fall ist, klonen Sie das Modell und trainieren Sie es neu.

Klonen Sie ein trainiertes Modell, um einen neuen Trainingspass zu starten

Wenn Sie ein zuvor trainiertes Modell als Ausgangspunkt für eine neue Trainingsrunde klonen, können Sie die Trainingseffizienz verbessern. Modifizieren Sie dazu die Hyperparameter, um das bereits erlernte Wissen zu nutzen.

In diesem Abschnitt erfahren Sie, wie Sie ein trainiertes Modell mithilfe der DeepRacer AWS-Konsole klonen.

Um das Reinforcement-Learning-Modell mithilfe der DeepRacer AWS-Konsole zu trainieren

1. Melden Sie sich bei der DeepRacer AWS-Konsole an, falls Sie noch nicht angemeldet sind.
2. Wählen Sie auf der Seite Modelle ein trainiertes Modell aus und wählen Sie dann in der Drop-down-Menüliste Aktion die Option Clone aus.
3. Gehen Sie für Model details (Modelldetails) wie folgt vor:

    a. Geben Sie `RL_model_1` als Model name (Modellname) ein, wenn Sie nicht möchten, dass ein Name für das geklonte Modell generiert wird.

    b. Geben Sie optional im Feld to-be-cloned Modellbeschreibung — optional eine Beschreibung für das Modell ein.
4. Wählen Sie für die Umgebungssimulation eine andere Spuroption.
5. Wählen Sie für die Reward function (Belohnungsfunktion) eines der verfügbaren Beispiele für Belohnungsfunktionen aus. Modifizieren Sie die Belohnungsfunktion. Berücksichtigen Sie zum Beispiel das Lenken.
6. Erweitern Sie Algorithm settings (Algorithmeneinstellungen) und probieren Sie verschiedene Optionen aus. Ändern Sie beispielsweise den Wert Gradient descent batch size (Gradientenabstieg-Batchgröße) von 32 auf 64 oder erhöhen Sie die Learning Rate (Lernrate), um das Training zu beschleunigen.
7. Experimentieren Sie mit verschiedenen Einstellungen der Stop conditions (Stoppbedingungen).

8. Wählen Sie Start training (Training starten) aus, um eine neue Runde des Trainings zu beginnen.

Wie beim Training eines robusten Machine-Learning-Modells im Allgemeinen ist es wichtig, dass Sie systematische Experimente durchführen, um die beste Lösung zu finden.

Evaluieren Sie DeepRacer AWS-Modelle in Simulationen

Ein Modell auszuwerten bedeutet, die Leistung eines trainierten Modells zu testen. In AWS DeepRacer ist die Standard-Leistungskennzahl die durchschnittliche Zeit, in der drei aufeinanderfolgende Runden beendet wurden. Bei Verwendung dieser Metrik ist für zwei beliebige Modelle ein Modell besser als das andere, wenn es den Agenten auf derselben Strecke schneller voranbringen kann.

Im Allgemeinen umfasst die Bewertung eines Modells die folgenden Aufgaben:

1. Konfigurieren und starten Sie einen Auswertungsauftrag.
2. Beobachten Sie die laufende Auswertung, während der Auftrag läuft. Dies kann im DeepRacer AWS-Simulator erfolgen.
3. Überprüfen Sie die Auswertungszusammenfassung, nachdem der Auswertungsauftrag abgeschlossen ist. Sie können einen laufenden Auswertungsauftrag jederzeit abbrechen.

   > **Note**
   >
   > Die Testzeit hängt von den von Ihnen ausgewählten Kriterien ab. Wenn Ihr Modell die Bewertungskriterien nicht erfüllt, wird die Bewertung fortgesetzt, bis die Obergrenze von 20 Minuten erreicht ist.

4. Reichen Sie das Bewertungsergebnis optional bei einer qualifizierten DeepRacer AWS-Bestenliste ein. Das Ranglistenergebnis auf der Rangliste zeigt Ihnen, wie gut sich Ihr Modell im Vergleich zu anderen Teilnehmern schlägt.

Testen Sie ein DeepRacer AWS-Modell mit einem DeepRacer AWS-Fahrzeug, das auf einer physischen Strecke fährt, sieheBedienen Sie Ihr Fahrzeug.

Optimieren Sie DeepRacer AWS-Schulungsmodelle für reale Umgebungen

Viele Faktoren beeinflussen die praktische Leistung eines trainierten Modells. Dazu gehören beispielsweise der Aktionsraum, die Belohnungsfunktion und die Hyperparameter im Training, die

Fahrzeugkalibrierung und die Bedingungen der Strecke in der echten Welt. Darüber hinaus ist die Simulation nur eine (oft grobe) Annäherung an die Praxis. Diese Faktoren sorgen dafür, dass das Training eines Modells in der Simulation und dessen späterer Einsatz in der echten Welt mit einer zufriedenstellenden Leistung schwierig ist.

Das Trainieren eines Modells für eine solide Leistung in der Praxis erfordert oft zahlreiche Iterationen zur Erforschung der Belohnungsfunktion, Aktionsräume, Hyperparameter und Bewertungen in der Simulation sowie dem Testen in einer realen Umgebung. Der letzte Schritt beinhaltet den sogenannten simulation-to-real Welttransfer (sim2real) und kann sich unhandlich anfühlen.

Berücksichtigen Sie die folgenden Punkte, um den sim2real-Vorgang einfacher zu gestalten:

- Stellen Sie sicher, dass Ihr Fahrzeug sauber kalibriert ist.

  Dies ist wichtig, da die simulierte Umgebung höchstwahrscheinlich eine teilweise Darstellung der realen Umgebung ist. Außerdem führt der Agent bei jedem Schritt eine Aktion durch, die auf der aktuellen von der Kamera aufgenommenen Streckenbedingung basiert. Es kann nicht weit genug sehen, um seine Route mit hoher Geschwindigkeit zu planen. Um dies zu berücksichtigen, setzt die Simulation Grenzen für die Geschwindigkeit und Lenkung. Um sicherzustellen, dass das trainierte Modell in der realen Welt funktioniert, muss das Fahrzeug entsprechend diesen und anderen Simulationseinstellungen kalibriert sein. Weitere Informationen zur Kalibrierung Ihres Fahrzeugs finden Sie unter the section called “Kalibrieren Sie Ihr Fahrzeug”.
- Testen Sie Ihr Fahrzeug zunächst mit dem Standardmodell.

  Ihr DeepRacer AWS-Fahrzeug wird mit einem vortrainierten Modell geliefert, das in die Inferenz-Engine geladen ist. Bevor Sie Ihr eigenes Modell in der realen Welt testen, vergewissern Sie sich, dass das Fahrzeug mit dem Standardmodell recht gut funktioniert. Wenn nicht, überprüfen Sie das physische Strecken-Setup. Das Testen eines Modells in einer falsch gebauten physischen Strecke führt wahrscheinlich zu einer schlechten Leistung. In solchen Fällen sollten Sie Ihre Strecke neu konfigurieren oder reparieren, bevor Sie mit dem Testen beginnen oder den Test fortsetzen.

  > **Note**
  >
  > Beim Betrieb Ihres DeepRacer AWS-Fahrzeugs werden Aktionen gemäß dem trainierten Policy-Netzwerk abgeleitet, ohne dass die Belohnungsfunktion aufgerufen wird.

- Stellen Sie sicher, dass das Modell in der Simulation funktioniert.

Wenn Ihr Modell in der realen Welt nicht gut funktioniert, könnten das Modell oder die Strecke fehlerhaft sein. Zur Klärung der Ursachen sollten Sie zunächst das Modell in Simulationen evaluieren, um zu prüfen, ob der simulierte Agent mindestens eine Runde beenden kann, ohne von der Strecke abzukommen. Sie können dies tun, indem Sie die Konvergenz der Belohnungen überprüfen, während Sie die Bewegungsbahn des Agenten im Simulator beobachten. Wenn die Belohnung das Maximum erreicht und wenn die simulierten Agenten ohne Unterbrechung eine Runde beenden, ist das Modell wahrscheinlich in Ordnung.

- Trainieren Sie das Modell nicht zu stark.

  Das Fortfahren mit dem Training, nachdem das Modell die Strecke in der Simulation konsequent absolviert hat, führt zu einer Überanpassung des Modells. Ein übertrainiertes Modell funktioniert in der realen Welt nicht besonders gut. Es kann selbst geringe Abweichungen zwischen der simulierten Strecke und der realen Umgebung nicht bewältigen.

- Verwenden Sie mehrere Modelle aus verschiedenen Iterationen.

  Eine typische Trainingseinheit produziert eine Reihe von Modellen, die irgendwo zwischen Unter- und Überanpassung liegen. Da es keine standardmäßigen Kriterien zur Bestimmung eines genau passenden Modells gibt, sollten Sie einige Modellkandidaten zwischen dem Zeitpunkt auswählen, an dem der Agent eine einzelne Runde im Simulator beendet, und dem Zeitpunkt, an dem er die Runde konsistent absolviert.

- Beginnen Sie langsam und erhöhen Sie die Fahrgeschwindigkeit im Test schrittweise.

  Wenn Sie das in Ihrem Fahrzeug eingesetzte Modell testen, beginnen Sie mit einem niedrigen Höchstgeschwindigkeitswert. Beispielsweise können Sie die Testgeschwindigkeit auf weniger als 10 % der Trainingsgeschwindigkeit festlegen. Erhöhen Sie dann schrittweise die Testgeschwindigkeitsbegrenzung, bis sich das Fahrzeug in Bewegung setzt. Die Testgeschwindigkeitsbegrenzung legen Sie bei der Kalibrierung des Fahrzeugs über die Gerätesteuerkonsole fest. Wenn das Fahrzeug zu schnell fährt, beispielsweise wenn die Geschwindigkeit die Geschwindigkeit übersteigt, die während des Trainings im Simulator gemessen wurde, ist es unwahrscheinlich, dass das Modell auf der realen Strecke gut abschneidet.

- Testen Sie ein Modell mit Ihrem Fahrzeug in verschiedenen Ausgangspositionen.

  Das Modell lernt, einen bestimmten Weg in der Simulation zu nehmen, und kann auf seine Position innerhalb der Strecke reagieren. Sie sollten die Fahrzeugtests mit verschiedenen Positionen innerhalb der Streckenbegrenzungen (links, Mitte, rechts) starten, um zu sehen, ob das Modell von bestimmten Positionen aus gut funktioniert. Die meisten Modelle neigen dazu, das Fahrzeug

nahe an der Seite einer der weißen Linien zu halten. Um den Weg des Fahrzeugs zu analysieren, zeichnen Sie Schritt für Schritt die Positionen des Fahrzeugs (x, y) aus der Simulation auf. So können Sie mögliche Wege identifizieren, die Ihr Fahrzeug in einer realen Umgebung zurücklegen muss.

- Beginnen Sie die Tests mit einer geraden Strecke.

  Eine gerade Strecke ist viel einfacher zu bewältigen als eine kurvige Strecke. Mit einer geraden Strecke zu beginnen bietet die Möglichkeit, schlechte Modelle schnell auszusortieren. Wenn ein Fahrzeug meistens daran scheitert, einer geraden Strecke zu folgen, wird das Modell auch auf kurvenreichen Strecken nicht gut funktionieren.

- Achten Sie auf Verhaltensweisen, bei denen das Fahrzeug nur eine Art von Aktionen ausführt.

  Wenn Ihr Fahrzeug nur eine Art von Maßnahmen ergreifen kann, z. B. das Fahrzeug nur nach links zu lenken, ist das Modell wahrscheinlich über- oder unterausgestattet. Zu viele Trainings-Iterationen bei gegebenen Modellparametern können dazu führen, dass das Modell übermäßig angepasst wird. Zu wenig Iterationen können dazu führen, dass es nicht richtig angepasst ist.

- Achten Sie auf die Fähigkeit des Fahrzeugs, seinen Weg entlang einer Streckenbegrenzung zu korrigieren.

  Ein gutes Modell bewirkt, dass sich das Fahrzeug bei Annäherung an die Streckenbegrenzungen selbst korrigiert. Die meisten gut trainierten Modelle verfügen über diese Fähigkeit. Wenn sich das Fahrzeug an beiden Streckenbegrenzungen selbst korrigieren kann, gilt das Modell als stabiler und hochwertiger.

- Achten Sie auf ein inkonsistentes Verhalten des Fahrzeugs.

  Ein Strategiemodell stellt eine Wahrscheinlichkeitsverteilung für das Ergreifen einer Aktion in einem bestimmten Zustand dar. Wenn das trainierte Modell in die Inferenz-Engine geladen wird, wählt ein Fahrzeug schrittweise entsprechend den Vorgaben des Modells die wahrscheinlichste Aktion aus. Wenn die Aktionswahrscheinlichkeiten gleichmäßig verteilt sind, kann das Fahrzeug alle Aktionen mit identischen oder sehr ähnlichen Wahrscheinlichkeiten durchführen. Dies führt zu einem unregelmäßigen Fahrverhalten. Wenn das Fahrzeug beispielsweise manchmal (z. B. die Hälfte der Zeit) einem geraden Weg folgt und zu anderen Zeiten unnötig abbiegt, ist das Modell entweder zu wenig oder zu stark ausgestattet.

- Achten Sie darauf, dass das Fahrzeug nur einmal abbiegt (links oder rechts).

  Wenn das Fahrzeug sehr gut nach links abbiegt, aber nicht nach rechts lenkt (oder andersrum), müssen Sie die Lenkung Ihres Fahrzeugs sorgfältig kalibrieren oder neu kalibrieren. Alternativ

können Sie versuchen, ein Modell zu verwenden, das mit Einstellungen trainiert wurde, die den zu testenden physischen Einstellungen entsprechen.

- Achten Sie darauf, dass das Fahrzeug plötzlich abbiegt und von der Strecke abkommt.

  Wenn das Fahrzeug den größten Teil des Weges korrekt zurücklegt, aber plötzlich von der Strecke abkommt, liegt dies wahrscheinlich an Ablenkungen in der Umgebung. Zu den häufigsten Ablenkungen gehören unerwartete oder unbeabsichtigte Lichtreflexionen. Verwenden Sie in solchen Fällen Barrieren um die Strecke herum oder andere Mittel, um grelles Licht zu reduzieren.

Referenz zur DeepRacer AWS-Prämienfunktion

Im Folgenden finden Sie die technische Referenz der DeepRacer AWS-Prämienfunktion.

Themen

- Eingabeparameter der DeepRacer AWS-Belohnungsfunktion
- Beispiele AWS DeepRacer AWS-Prämienfunktionen

Eingabeparameter der DeepRacer AWS-Belohnungsfunktion

Die DeepRacer AWS-Belohnungsfunktion verwendet ein Wörterbuchobjekt als Eingabe.

```
def reward_function(params) :

    reward = ...

    return float(reward)
```

Das `params`-Dictionary-Objekt enthält die folgenden Schlüssel/Werte-Paare:

```
{
    "all_wheels_on_track": Boolean,        # flag to indicate if the agent is on the
 track
    "x": float,                            # agent's x-coordinate in meters
    "y": float,                            # agent's y-coordinate in meters
    "closest_objects": [int, int],         # zero-based indices of the two closest
 objects to the agent's current position of (x, y).
    "closest_waypoints": [int, int],       # indices of the two nearest waypoints.
    "distance_from_center": float,         # distance in meters from the track center
```

```
    "is_crashed": Boolean,                # Boolean flag to indicate whether the agent
 has crashed.
    "is_left_of_center": Boolean,         # Flag to indicate if the agent is on the
 left side to the track center or not.
    "is_offtrack": Boolean,               # Boolean flag to indicate whether the agent
 has gone off track.
    "is_reversed": Boolean,               # flag to indicate if the agent is driving
 clockwise (True) or counter clockwise (False).
    "heading": float,                     # agent's yaw in degrees
    "objects_distance": [float, ],        # list of the objects' distances in meters
 between 0 and track_length in relation to the starting line.
    "objects_heading": [float, ],         # list of the objects' headings in degrees
 between -180 and 180.
    "objects_left_of_center": [Boolean, ], # list of Boolean flags indicating whether
 elements' objects are left of the center (True) or not (False).
    "objects_location": [(float, float),], # list of object locations [(x,y), ...].
    "objects_speed": [float, ],           # list of the objects' speeds in meters per
 second.
    "progress": float,                    # percentage of track completed
    "speed": float,                       # agent's speed in meters per second (m/s)
    "steering_angle": float,              # agent's steering angle in degrees
    "steps": int,                         # number steps completed
    "track_length": float,                # track length in meters.
    "track_width": float,                 # width of the track
    "waypoints": [(float, float), ]       # list of (x,y) as milestones along the
 track center

}
```

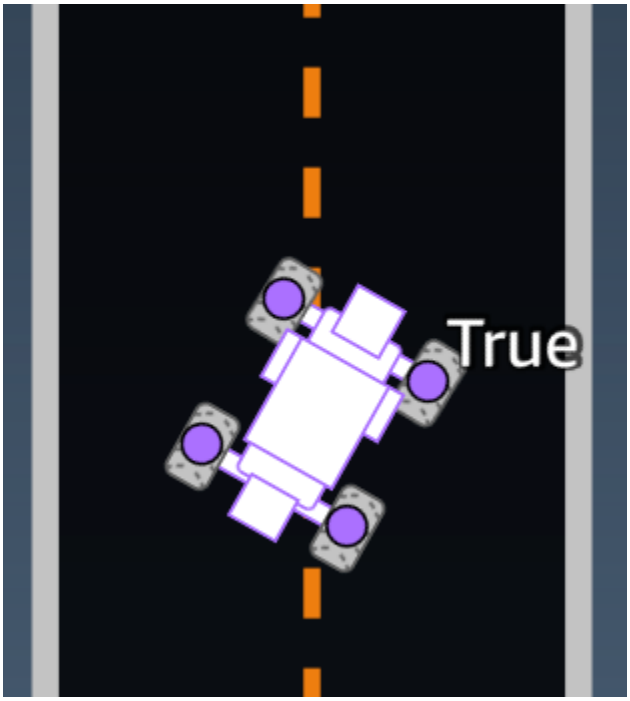
Es folgt eine detaillierte technische Referenz der Eingangsparameter.

all_wheels_on_track

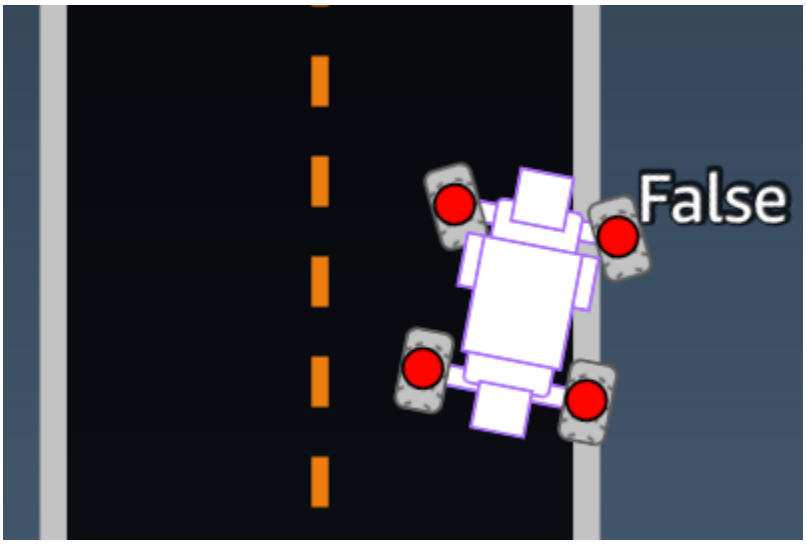
Typ: `Boolean`

Bereich `(True:False)`

Ein `Boolean`-Flag, das anzeigt, ob sich der Agent auf der Strecke oder außerhalb der Strecke befindet. Es ist außerhalb der Strecke (`False`), wenn sich eines seiner Räder außerhalb der Streckenbegrenzungen befindet. Es ist auf der Strecke (`True`), wenn sich alle Räder innerhalb der beiden Streckenbegrenzungen befinden. Die folgende Abbildung zeigt, dass sich der Agent auf der Strecke befindet.



Die folgende Abbildung zeigt, dass sich der Agent außerhalb der Strecke befindet.



Beispiel: Eine Belohnungsfunktion mit dem Parameter *`all_wheels_on_track`*.

```
def reward_function(params):
    ################################################################################
    '''
    Example of using all_wheels_on_track and speed
    '''

    # Read input variables
```

```
    all_wheels_on_track = params['all_wheels_on_track']
    speed = params['speed']

    # Set the speed threshold based your action space
    SPEED_THRESHOLD = 1.0

    if not all_wheels_on_track:
        # Penalize if the car goes off track
        reward = 1e-3
    elif speed < SPEED_THRESHOLD:
        # Penalize if the car goes too slow
        reward = 0.5
    else:
        # High reward if the car stays on track and goes fast
        reward = 1.0

    return float(reward)
```

closest_waypoints

Typ: `[int, int]`

Bereich: `[(0:Max-1),(1:Max-1)]`

Die nullbasierten Indizes der beiden benachbarten `waypoints`, die der aktuellen Position `(x, y)` des Agenten am nächsten liegen. Die Entfernung wird über den euklidischen Abstand von der Mitte des Fahrzeugs aus gemessen. Das erste Element bezieht sich auf den nächsten Wegpunkt hinter dem Agenten und das zweite Element bezieht sich auf den nächsten Wegpunkt vor dem Agenten. `Max` ist die Länge der Wegpunktliste. In der in waypoints gezeigten Abbildung wären die `closest_waypoints [16, 17]`.

Beispiel: Eine Belohnungsfunktion mit dem Parameter `closest_waypoints`.

Die folgende exemplarische Belohnungsfunktion zeigt, wie man mit `waypoints` und `closest_waypoints` sowie `heading` direkte Belohnungen berechnet.

AWS DeepRacer unterstützt die folgenden Bibliotheken: math, random, NumPy SciPy, und Shapely. Um eine zu verwenden, fügen Sie über Ihrer Funktionsdefinition eine Importanweisung hinzu`def function_name(parameters):` `import` *`supported library`*

```
# Place import statement outside of function (supported libraries: math, random, numpy,
 scipy, and shapely)
# Example imports of available libraries
#
# import math
# import random
# import numpy
# import scipy
# import shapely

import math

def reward_function(params):
    ################################################################################
    '''
    Example of using waypoints and heading to make the car point in the right direction
    '''

    # Read input variables
    waypoints = params['waypoints']
    closest_waypoints = params['closest_waypoints']
    heading = params['heading']

    # Initialize the reward with typical value
    reward = 1.0

    # Calculate the direction of the center line based on the closest waypoints
    next_point = waypoints[closest_waypoints[1]]
    prev_point = waypoints[closest_waypoints[0]]

    # Calculate the direction in radius, arctan2(dy, dx), the result is (-pi, pi) in
 radians
    track_direction = math.atan2(next_point[1] - prev_point[1], next_point[0] -
 prev_point[0])
    # Convert to degree
    track_direction = math.degrees(track_direction)

    # Calculate the difference between the track direction and the heading direction of
 the car
    direction_diff = abs(track_direction - heading)
    if direction_diff > 180:
        direction_diff = 360 - direction_diff
```

```
    # Penalize the reward if the difference is too large
    DIRECTION_THRESHOLD = 10.0
    if direction_diff > DIRECTION_THRESHOLD:
        reward *= 0.5

    return float(reward)
```

closest_objects

Typ: `[int, int]`

Bereich: `[(0:len(objects_location)-1), (0:len(objects_location)-1)]`

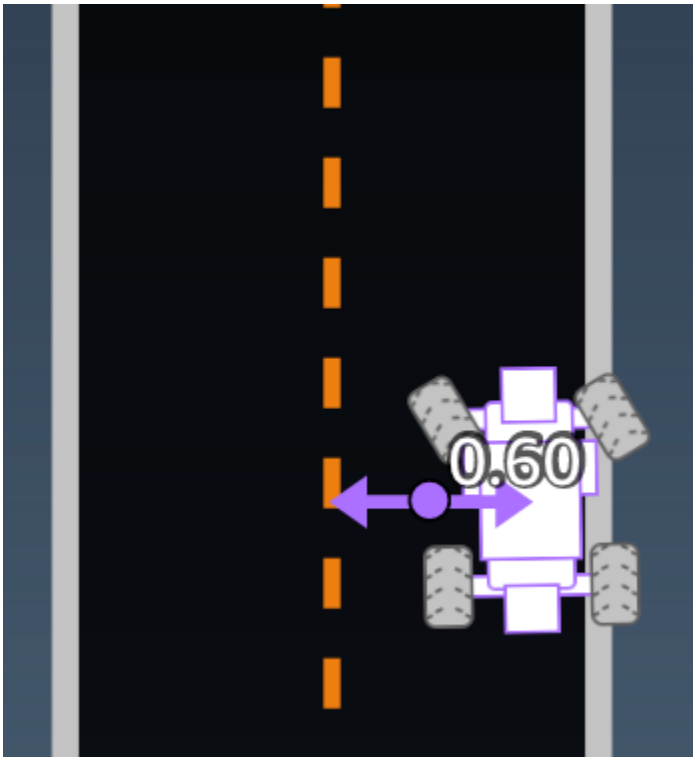
Die nullbasierten Indizes der beiden Objekte, die der aktuellen Position des Agenten (x, y) am nächsten liegen. Der erste Index bezieht sich auf das nächste Objekt hinter dem Agenten und der zweite Index bezieht sich auf das nächste Objekt vor dem Agenten. Wenn nur ein Objekt vorhanden ist, sind beide Indizes 0.

distanz_from_center

Typ: `float`

Bereich: `0:~track_width/2`

Verschiebung zwischen der Mitte des Agenten und der Streckenmitte in Metern. Die feststellbare maximale Verschiebung tritt auf, wenn sich eines der Räder des Agenten außerhalb einer Streckenbegrenzung befindet. Je nach Breite der Streckenbegrenzung kann sie etwas kleiner oder größer als die Hälfte von `track_width` sein.



Beispiel: Eine Belohnungsfunktion mit dem *`distance_from_center`*-Parameter.

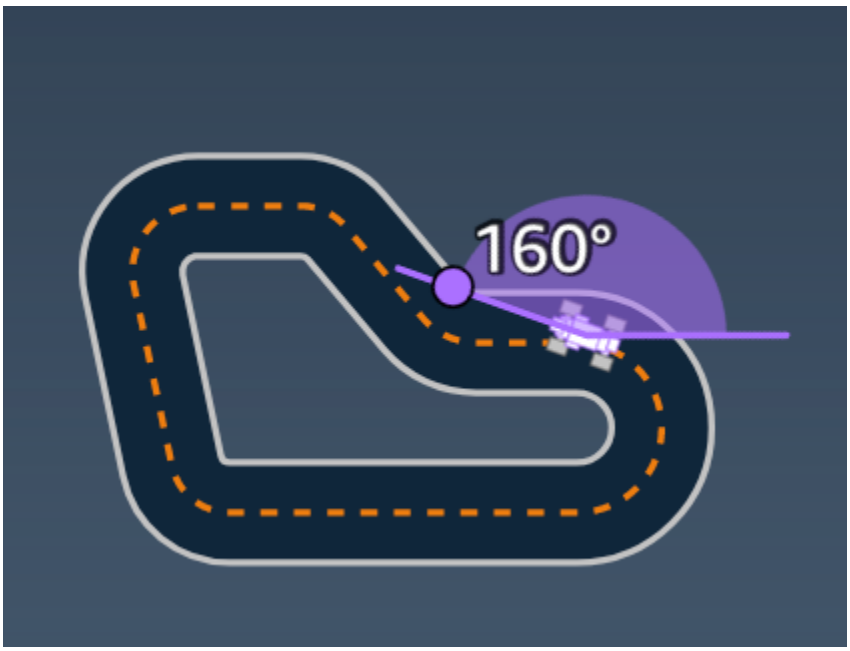
```
def reward_function(params):
    ################################################################################
    '''
    Example of using distance from the center
    '''

    # Read input variable
    track_width = params['track_width']
    distance_from_center = params['distance_from_center']

    # Penalize if the car is too far away from the center
    marker_1 = 0.1 * track_width
    marker_2 = 0.5 * track_width

    if distance_from_center <= marker_1:
        reward = 1.0
    elif distance_from_center <= marker_2:
        reward = 0.5
    else:
        reward = 1e-3  # likely crashed/ close to off track

    return float(reward)
```

heading

Typ: `float`

Bereich: `-180:+180`

Fahrtrichtung des Agenten in Grad in Bezug auf die x-Achse des Koordinatensystems.



Beispiel: Eine Belohnungsfunktion mit dem *`heading`*-Parameter.

Weitere Informationen finden Sie unter `closest_waypoints`.

is_crashed

Typ: `Boolean`

Bereich: `(True:False)`

Ein boolesches Flag, das angibt, ob der Agent als Abschlussstatus mit einem anderen Objekt kollidiert ist (`True`) oder nicht ist (`False`).

is_left_of_center

Typ: `Boolean`

Bereich: `[True : False]`

Ein `Boolean`-Flag, das anzeigt, ob sich der Agent auf der linken Seite (`True`) oder rechten Seite (`False`) der Streckenmitte befindet.

is_offtrack

Typ: `Boolean`

Bereich: `(True:False)`

Ein boolesches Flag, das angibt, ob sich der Agent als Abschlussstatus außerhalb der Spur (True) oder nicht (False) befindet.

is_reversed

Typ: `Boolean`

Bereich: `[True:False]`

Ein boolesches Flag, das anzeigt, ob der Agent im Uhrzeigersinn (True) oder gegen den Uhrzeigersinn (False) fährt.

Es wird verwendet, wenn Sie die Richtungsänderung für jede Episode aktivieren.

objects_distance

Typ: `[float, … ]`

Bereich: `[(0:track_length), … ]`

Eine Liste der Abstände zwischen Objekten in der Umgebung in Bezug auf die Startlinie. Das i-te Element misst den Abstand in Metern zwischen dem i-ten Objekt und der Startlinie entlang der Streckenmittellinie.

> **Note**
>
> abs | (var1) - (var2) | = how close the car is to an object , WENN var1 = ["objects_distance"] [index] und var2 = params ["progress"] *params ["track_length"]
> Mit dem Parameter „closest_objects“ erhalten Sie einen Index des nächstgelegenen Objekts vor dem Fahrzeug und des nächstgelegenen Objekts hinter dem Fahrzeug.

objects_heading

Typ: `[float, … ]`

Bereich: `[(-180:180), … ]`

Liste der Fahrkurse von Objekten in Grad. Das i-te Element misst den Fahrkurs des i-ten Objekts. Bei stationären Objekten sind ihre Fahrkurse 0. Bei einem Bot-Fahrzeug ist der Wert des entsprechenden Elements der Kurswinkel des Fahrzeugs.

objects_left_of_center

Typ: `[Boolean, … ]`

Bereich: `[True|False, … ]`

Liste der booleschen Flags. Der i-te Elementwert gibt an, ob sich das i-te Objekt links (True) oder rechts (False) von der Streckenmitte befindet.

objects_location

Typ: `[(x,y), … ]`

Bereich: `[(0:N,0:N), … ]`

Liste aller Objektpositionen, jede Position ist ein n-Tupel von (x, y).

Die Größe der Liste entspricht der Anzahl der Objekte auf der Strecke. Beachten Sie, dass das Objekt ein stationäres Hindernis oder ein sich bewegendes Bot-Fahrzeug sein könnte.

objects_speed

Typ: `[float, … ]`

Bereich: `[(0:12.0), … ]`

Liste der Geschwindigkeiten (Meter pro Sekunde) für die Objekte auf der Strecke. Bei stationären Objekten sind ihre Geschwindigkeiten 0. Bei einem Bot-Fahrzeug entspricht der Wert der Geschwindigkeit, die Sie im Training festgelegt haben.

progress

Typ: `float`

Bereich: `0:100`

Prozentsatz der abgeschlossenen Strecke.

Beispiel: Eine Belohnungsfunktion mit dem *`progress`*-Parameter.

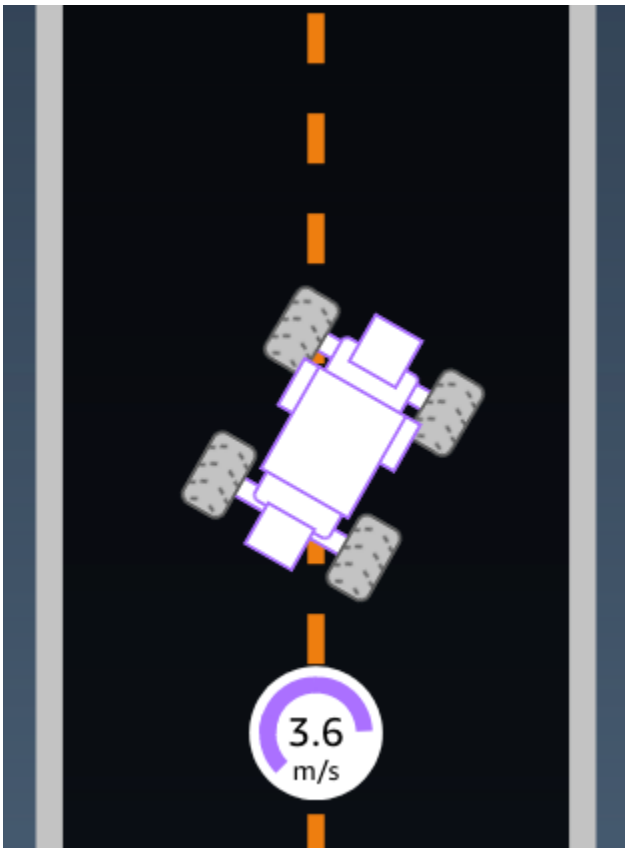
Weitere Informationen finden Sie unter Schritte.

speed

Typ: `float`

Bereich: `0.0:5.0`

Die beobachtete Geschwindigkeit des Agenten in Metern pro Sekunde (m/s).



Beispiel: Eine Belohnungsfunktion mit dem *`speed`*-Parameter.

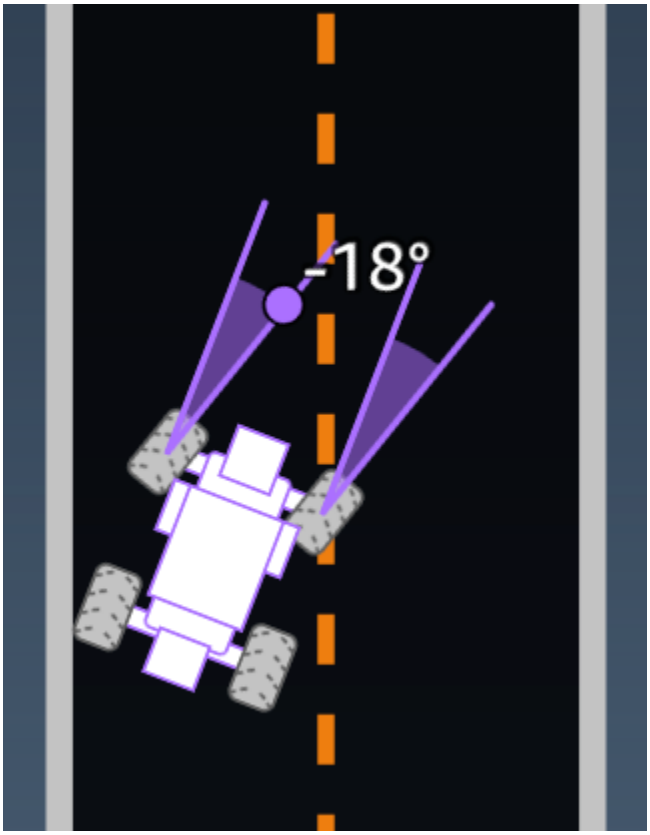
Weitere Informationen finden Sie unter all_wheels_on_track.

steering_angle

Typ: `float`

Bereich: `-30:30`

Lenkwinkel der Vorderräder in Grad von der Mittellinie des Agenten aus. Das negative Vorzeichen (-) steht für eine Lenkung nach rechts. Das positive (+) Vorzeichen steht für eine Lenkung nach links. Die Mittellinie des Agenten ist nicht unbedingt parallel zur Streckenmittellinie, wie in der folgenden Abbildung dargestellt.



Beispiel: Eine Belohnungsfunktion mit dem *`steering_angle`*-Parameter.

```
def reward_function(params):
    '''
    Example of using steering angle
    '''

    # Read input variable
    abs_steering = abs(params['steering_angle']) # We don't care whether it is left or
 right steering

    # Initialize the reward with typical value
    reward = 1.0

    # Penalize if car steer too much to prevent zigzag
```

```
    ABS_STEERING_THRESHOLD = 20.0
    if abs_steering > ABS_STEERING_THRESHOLD:
        reward *= 0.8

    return float(reward)
```

steps

Typ: `int`

Bereich: `0:`N_{step}

Anzahl der abgeschlossenen Schritte. Ein Schritt entspricht einer Aktion, die der Agent gemäß der aktuellen Richtlinie durchführt.

Beispiel: Eine Belohnungsfunktion mit dem *`steps`*-Parameter.

```
def reward_function(params):
    #############################################################################
    '''
    Example of using steps and progress
    '''

    # Read input variable
    steps = params['steps']
    progress = params['progress']

    # Total num of steps we want the car to finish the lap, it will vary depends on the
 track length
    TOTAL_NUM_STEPS = 300

    # Initialize the reward with typical value
    reward = 1.0

    # Give additional reward if the car pass every 100 steps faster than expected
    if (steps % 100) == 0 and progress > (steps / TOTAL_NUM_STEPS) * 100 :
        reward += 10.0

    return float(reward)
```

track_length

Typ: `float`

Bereich: `[0:`L_{max}`]`

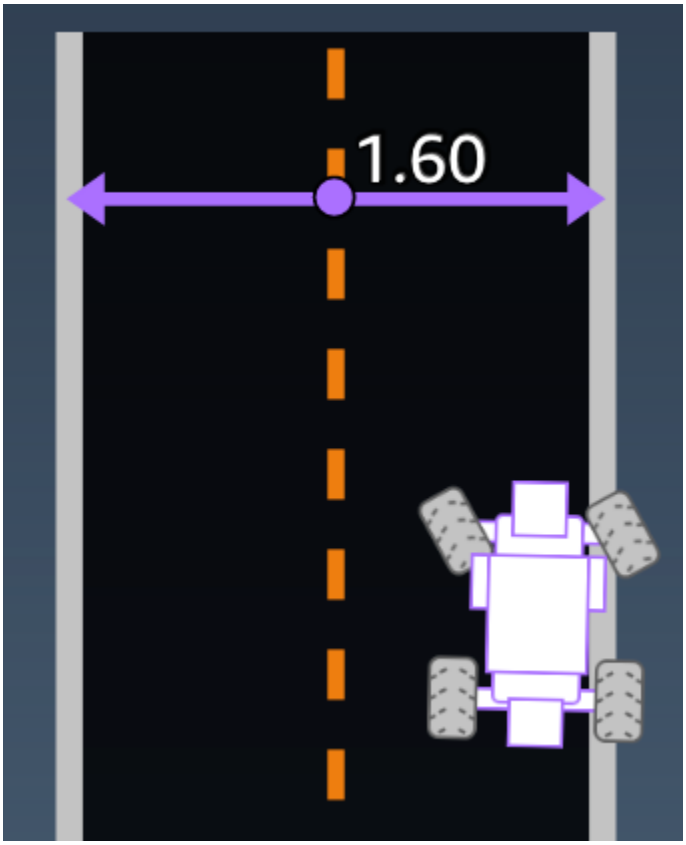
Die Streckenlänge in Metern. L_{max} `is track-dependent.`

track_width

Typ: `float`

Bereich: `0:`D_{track}

Streckenbreite in Metern.



Beispiel: Eine Belohnungsfunktion mit dem *`track_width`*-Parameter.

```
def reward_function(params):
    ################################################################################
    '''
```

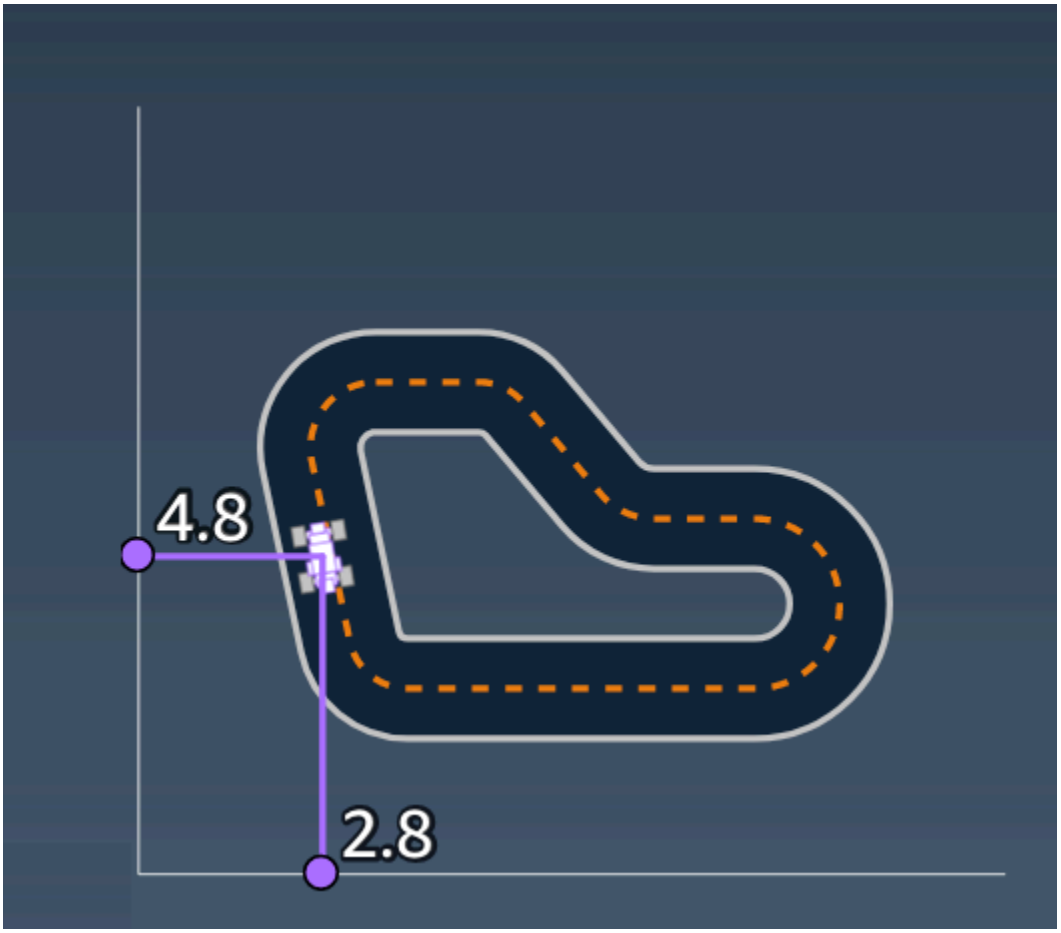
```
    Example of using track width
    '''

    # Read input variable
    track_width = params['track_width']
    distance_from_center = params['distance_from_center']

    # Calculate the distance from each border
    distance_from_border = 0.5 * track_width - distance_from_center

    # Reward higher if the car stays inside the track borders
    if distance_from_border >= 0.05:
        reward = 1.0
    else:
        reward = 1e-3 # Low reward if too close to the border or goes off the track

    return float(reward)
```

x, y

Typ: `float`

Bereich: `0:N`

Position der Mitte des Agenten zur x- und y-Achse der simulierten Umgebung mit der Strecke in Metern. Der Nullpunkt liegt in der unteren linken Ecke der simulierten Umgebung.

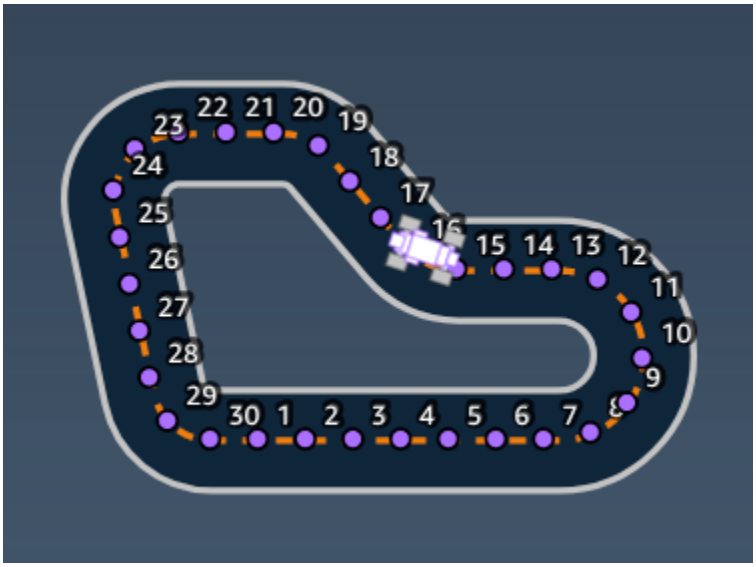


waypoints

Typ: `list` mit `[float, float]`

Bereich: `[[`$x_{w,0}$`,`$y_{w,0}$`]` … `[`$x_{w,Max-1}$`,` $y_{w,Max-1}$`]]`

Eine sortierte Liste von streckenabhängigen `Max`-Meilensteinen entlang der Streckenmitte. Jeder Meilenstein wird durch eine Koordinate von ($x_{w,i}$, $y_{w,i}$) beschrieben. Bei einer Schleife sind die ersten und die letzte Zwischenposition identisch. Bei einer geraden Strecke oder einer Strecke, die keine Schleife bildet, unterscheiden sich die erste und die letzte Zwischenposition.



Beispiel Eine Belohnungsfunktion mit dem Parameter *`waypoints`*.

Weitere Informationen finden Sie unter `closest_waypoints`.

Beispiele AWS DeepRacer AWS-Prämienfunktionen

Im Folgenden sind einige Beispiele für die DeepRacer AWS-Prämienfunktion aufgeführt.

Themen

- Beispiel 1: Folgen Sie beim Zeitfahren der Mittellinie
- Beispiel 2: Bleiben Sie beim Zeitfahren innerhalb der beiden Grenzen
- Beispiel 3: Vermeiden Sie Zick-Zack-Bewegungen bei Zeitfahren
- Beispiel 4: Bleiben Sie auf einer Spur, ohne gegen stehende Hindernisse oder fahrende Fahrzeuge zu stoßen

Beispiel 1: Folgen Sie beim Zeitfahren der Mittellinie

In diesem Beispiel wird festgestellt, wie weit der Agent von der Mittellinie entfernt ist, und er erhält eine höhere Belohnung, wenn er näher an der Mitte der Spur liegt; dadurch wird er veranlasst, der Mittellinie möglichst genau zu folgen.

```
def reward_function(params):
    '''
    Example of rewarding the agent to follow center line
    '''
```

```
    # Read input parameters
    track_width = params['track_width']
    distance_from_center = params['distance_from_center']

    # Calculate 3 markers that are increasingly further away from the center line
    marker_1 = 0.1 * track_width
    marker_2 = 0.25 * track_width
    marker_3 = 0.5 * track_width

    # Give higher reward if the car is closer to center line and vice versa
    if distance_from_center <= marker_1:
        reward = 1
    elif distance_from_center <= marker_2:
        reward = 0.5
    elif distance_from_center <= marker_3:
        reward = 0.1
    else:
        reward = 1e-3  # likely crashed/ close to off track

    return reward
```

Beispiel 2: Bleiben Sie beim Zeitfahren innerhalb der beiden Grenzen

Dieses Beispiel bietet einfach hohe Belohnungen, wenn der Agent innerhalb der Grenzen bleibt, und lässt den Agenten herausfinden, wie er eine Runde am besten beenden kann. Es ist einfach zu programmieren und zu verstehen, aber die Konvergenz dauert wahrscheinlich länger.

```
def reward_function(params):
    '''
    Example of rewarding the agent to stay inside the two borders of the track
    '''

    # Read input parameters
    all_wheels_on_track = params['all_wheels_on_track']
    distance_from_center = params['distance_from_center']
    track_width = params['track_width']

    # Give a very low reward by default
    reward = 1e-3

    # Give a high reward if no wheels go off the track and
    # the car is somewhere in between the track borders
```

```
    if all_wheels_on_track and (0.5*track_width - distance_from_center) >= 0.05:
        reward = 1.0

    # Always return a float value
    return reward
```

Beispiel 3: Vermeiden Sie Zick-Zack-Bewegungen bei Zeitfahren

In diesem Beispiel wird der Agent belohnt, wenn er der Mittellinie folgt, bei zu vielen Lenkbewegungen wird er jedoch bestraft, wodurch Zick-Zack-Bewegungen vermieden werden sollen. Der Agent lernt, im Simulator reibungslos zu fahren, und behält wahrscheinlich dasselbe Verhalten bei, wenn er auf dem physischen Fahrzeug eingesetzt wird.

```
def reward_function(params):
    '''
    Example of penalize steering, which helps mitigate zig-zag behaviors
    '''

    # Read input parameters
    distance_from_center = params['distance_from_center']
    track_width = params['track_width']
    abs_steering = abs(params['steering_angle']) # Only need the absolute steering
 angle

    # Calculate 3 marks that are farther and father away from the center line
    marker_1 = 0.1 * track_width
    marker_2 = 0.25 * track_width
    marker_3 = 0.5 * track_width

    # Give higher reward if the car is closer to center line and vice versa
    if distance_from_center <= marker_1:
        reward = 1.0
    elif distance_from_center <= marker_2:
        reward = 0.5
    elif distance_from_center <= marker_3:
        reward = 0.1
    else:
        reward = 1e-3  # likely crashed/ close to off track

    # Steering penality threshold, change the number based on your action space setting
    ABS_STEERING_THRESHOLD = 15

    # Penalize reward if the car is steering too much
```

```
    if abs_steering > ABS_STEERING_THRESHOLD:
        reward *= 0.8

    return float(reward)
```

Beispiel 4: Bleiben Sie auf einer Spur, ohne gegen stehende Hindernisse oder fahrende Fahrzeuge zu stoßen

Diese Belohnungsfunktion belohnt den Agenten dafür, dass er sich innerhalb der Streckengrenzen aufhält, und bestraft den Agenten, wenn er Objekten vor der Strecke zu nahe kommt. Der Agent kann sich von Spur zu Spur bewegen, um Kollisionen zu vermeiden. Die Gesamtbelohnung ist eine gewichtete Summe aus den Belohnungen und Bestrafungen. In diesem Beispiel wird der Strafe mehr Gewicht beigemessen, um Abstürze zu vermeiden. Experimentieren Sie mit unterschiedlichen Gewichten zur Mittelwertbildung, um für unterschiedliche Verhaltensergebnisse zu trainieren.

```
import math
def reward_function(params):
    '''
    Example of rewarding the agent to stay inside two borders
    and penalizing getting too close to the objects in front
    '''
    all_wheels_on_track = params['all_wheels_on_track']
    distance_from_center = params['distance_from_center']
    track_width = params['track_width']
    objects_location = params['objects_location']
    agent_x = params['x']
    agent_y = params['y']
    _, next_object_index = params['closest_objects']
    objects_left_of_center = params['objects_left_of_center']
    is_left_of_center = params['is_left_of_center']
    # Initialize reward with a small number but not zero
    # because zero means off-track or crashed
    reward = 1e-3
    # Reward if the agent stays inside the two borders of the track
    if all_wheels_on_track and (0.5 * track_width - distance_from_center) >= 0.05:
        reward_lane = 1.0
    else:
        reward_lane = 1e-3
    # Penalize if the agent is too close to the next object
    reward_avoid = 1.0
```

```
    # Distance to the next object
    next_object_loc = objects_location[next_object_index]
    distance_closest_object = math.sqrt((agent_x - next_object_loc[0])**2 + (agent_y -
next_object_loc[1])**2)
    # Decide if the agent and the next object is on the same lane
    is_same_lane = objects_left_of_center[next_object_index] == is_left_of_center
    if is_same_lane:
        if 0.5 <= distance_closest_object < 0.8:
            reward_avoid *= 0.5
        elif 0.3 <= distance_closest_object < 0.5:
            reward_avoid *= 0.2
        elif distance_closest_object < 0.3:
            reward_avoid = 1e-3  # Likely crashed
    # Calculate reward by putting different weights on
    # the two aspects above
    reward += 1.0 * reward_lane + 4.0 * reward_avoid
    return reward
```

Modelle in der DeepRacer AWS-Konsole importieren und exportieren

Es gibt Szenarien, in denen Sie möglicherweise ein DeepRacer AWS-Modell importieren oder exportieren müssen. Rennfahrer, die an einer vom Arbeitgeber gesponserten Veranstaltung teilgenommen haben, können ihre Modelle exportieren, um zu verhindern, dass sie den Zugriff auf sie verlieren. Rennadministratoren können den Teilnehmern vorab geschulte Modelle zur Verfügung stellen, die sie importieren und während der Veranstaltung verwenden können. Verwenden Sie die Seite „Ihre Modelle“, um DeepRacer AWS-Modelle in der Konsole zu importieren und zu exportieren.

Themen

- Kopieren Sie Ihr DeepRacer AWS-Modell nach Amazon S3
- Importieren Sie Ihr DeepRacer AWS-Modell in die Konsole
- Fehlerbehebung

Kopieren Sie Ihr DeepRacer AWS-Modell nach Amazon S3

Um ein DeepRacer AWS-Modell nach Amazon S3 zu kopieren

1. Melden Sie sich bei der DeepRacer AWS-Konsole an.
2. Wählen Sie im Navigationsbereich unter Reinforcement Learning die Option Ihre Modelle aus.
3. Wählen Sie das Modell aus, das Sie importieren möchten, indem Sie das Kontrollkästchen neben dem Modellnamen aktivieren. Sie können jeweils nur ein Modell von der Konsole auf Amazon S3 kopieren.
4. Wählen Sie das Drop-down-Menü mit der Schaltfläche „Aktionen“ und dann „Nach S3 kopieren“.

   Eine neue Seite „Nach Amazon S3 kopieren“ wird geöffnet.
5. Verwenden Sie auf der Seite Nach Amazon S3 kopieren die Dropdownauswahl für Amazon S3 S3-Bucket, um einen Amazon S3 S3-Bucket auszuwählen, in den das Modell exportiert werden soll. AWS DeepRacer S3-Buckets müssen Deepacer im Namen enthalten.

   - Wenn Sie keinen gültigen Amazon S3 S3-Bucket haben, erstellen Sie einen, indem Sie Neuen Bucket erstellen wählen. Die Dropdownauswahl füllt einen Bucket-Namen mit dem folgenden Format aus. `aws-deepracer-assets-XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX`

6. Fügen Sie dem Amazon S3-Objekt im Feld S3-Objektpräfix ein optionales Ordnerpräfix hinzu.
7. Nachdem Sie den S3-Bucket eingerichtet haben, wählen Sie aus, welche Assets Sie einbeziehen möchten. Sie müssen mindestens einen Asset-Typ auswählen, um fortzufahren.
   - Modell: Der Modellordner enthält alle erforderlichen Dateien für einen Modellimport.
   - Protokolle: Kopiert die Trainings- und Evaluierungsprotokolle für das Modell. Diese Option umfasst die `sim-trace/` Ordner `logs/metrics/`, und.
   - Video: Diese Option kopiert den Videoordner in Ihren Amazon S3 S3-Bucket. Der Videoordner enthält die `training/` Ordner `evaluation/` und. Diese Ordner enthalten Videos aus der Draufsicht, einer 45-Grad-Ansicht und einer 45-Grad-Ansicht, wobei das Konsolen-Overlay die Position des Fahrzeugs auf der Strecke anzeigt.
8. Nachdem Sie auf Kopieren geklickt haben, wird ein Pop-up angezeigt, das Sie darüber informiert, dass Sie für die Amazon S3 S3-Datenspeicherkosten verantwortlich sind. Wenn Sie mit den Bedingungen einverstanden sind, klicken Sie im Pop-up auf die Schaltfläche Kopieren.
9. Sobald der Kopiervorgang gestartet ist, kehren Sie in der Konsole zur Seite „Ihre Modelle“ zurück. Ein Banner oben auf der Seite zeigt den aktuellen Status an. Wenn der Exportvorgang abgeschlossen ist, bestätigt das Banner den erfolgreichen Export.

Erforderliche Dateien für den Modellimport

Um einen Modellordner für ein Modell hochzuladen, das außerhalb der Konsole trainiert wurde, folgen Sie den Schritten auf der Seite Objekte hochladen der Amazon S3 S3-Dokumentation. Die folgende Tabelle enthält eine Liste der Dateien, die für den Modellimport erforderlich sind. Wenn eine der erforderlichen Dateien fehlt, schlägt der Modellimport fehl.

In der DeepRacer AWS-Konsole trainierte Modelle haben das OrdnernamenformatDAY/MONTH/YEAR/TIME GMT. Unser Beispielmodell wurde am 30. November 2023 exportiert und der Ordnername lautetThu, 30 Nov 2023 19:01:24 GMT. In diesem Beispiel bezeichnen wir diesen Ordner als Root.

Erforderliche Dateien für Modellimporte

Dateiname	Ordnerpfad	Beschreibung
.coach_checkpoint	Wurzel/Modell/	Die Coach-Checkpoint-Datei enthält den Schlüssel für den

Dateiname	Ordnerpfad	Beschreibung
		Modell-Checkpoint, der beim Import verwendet wurde.
CKPT-Dateien	Stamm/Modell/	Checkpoint-Dateien sind Schnappschüsse der Modellgewichte, die in verschiedenen Phasen des Trainings aufgenommen wurden. Sie enthalten die Dateien `ckpt.index ckpt.data` , und`ckpt.meta` .
model_metadata.json	wurzel/	Die Modell-Metadatenda tei enthält Einstellungen, zu denen Definitionen des Aktionsraums, die Sensorkon figuration und die Auswahl des Trainingsalgorhythmus gehören.
reward_function.py	wurzel/	Eine Python-Datei, die die Belohnungsfunktion enthält, mit der das Modell trainiert wurde.

Metrikdateien sind für den Import Ihres Modells nicht erforderlich. Wenn diese Dateien nicht enthalten sind, sind die Trainingsmetriken und das Belohnungsdiagramm für das Modell nicht auf der Konsole verfügbar.

Optionale Dateien für Modellimporte

Dateiname	Pfad des Ordners	Beschreibung
training_params.yaml	wurzel/	Die Datei training_params enthält Trainingsauftragsd

Dateiname	Pfad des Ordners	Beschreibung
		aten, darunter Strecken- und Fahrzeuginformationen, Namen von Rennfahrern und Modellen sowie Ordnerpfade für Trainingsartefakte.
hyperparameters.json	root/ip/	Enthält die Hyperpara meterinformationen des Modells wie Chargengröße, Verlusttyp, Lernrate und Anzahl der Epochen.
training-*.json	root/metrics/training/	Wird verwendet, um die Trainingsmetriken des Modells in der DeepRacer AWS-Konso le zu visualisieren.

Importieren Sie Ihr DeepRacer AWS-Modell in die Konsole

In diesem Abschnitt werden Sie Schritt für Schritt durch den Import eines DeepRacer AWS-Modells in die Konsole geführt. Bevor Sie ein Modell importieren können, müssen Sie die Amazon S3 S3-URL für den Modellordner kopieren.

Kopieren Sie die URL des AWS DeepRacer Amazon S3 S3-Buckets

1. Melden Sie sich bei der Amazon S3 S3-Konsole an und rufen Sie die Buckets-Seite auf.
2. Wählen Sie den Amazon S3 S3-Bucket aus, den Sie für Ihr DeepRacer AWS-Modell erstellt haben, indem Sie auf den Link im Namen des Buckets klicken. Das Format der in der DeepRacer AWS-Konsole erstellten S3-Buckets ist `aws-deepracer-assets-XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX`.
3. Gehen Sie auf der Registerkarte Objekte des DeepRacer AWS-Buckets wie folgt vor:

    a. Wählen Sie das Modell aus, das Sie importieren möchten, indem Sie im Feld Name auf den Link mit dem Namen des Modellobjekts klicken.

    Eine Liste mit Unterordnern wird angezeigt.

b. Wählen Sie das Stammverzeichnis des Modellordners aus, indem Sie das Kontrollkästchen neben dem Ordnernamen aktivieren. In der DeepRacer AWS-Konsole trainierte Modelle haben das OrdnernamenformatDAY/MONTH/YEAR/TIME GMT.

4. Sobald Sie zum Stammverzeichnis des Modellordners navigiert haben, klicken Sie auf die Schaltfläche S3-URL kopieren. Die Schaltflächen S3-URL kopieren, URL kopieren, Öffnen und Löschen sind ausgegraut, bis Sie das Kontrollkästchen neben dem Modellnamen aktivieren.

> **Note**
>
> Wenn Sie während des Vorgangs „Nach S3 kopieren“ ein Präfix hinzugefügt haben*my_model/version_2*, lautet `deep_racer_bucket/model_name/`*my_model/version_2*`/root/` z. B. der Pfad Ihres Modellordners.

Importieren Sie Ihr Modell in die DeepRacer AWS-Konsole

1. Rufen Sie in der DeepRacer AWS-Konsole die Seite Ihre Modelle auf.
2. Wählen Sie im Modellcontainer die Schaltfläche Modell importieren aus.

   Die Seite „Modell importieren“ wird angezeigt.
3. Im Bereich Import:

   - Geben Sie die Amazon S3 S3-URL für den Modellordner ein, den Sie importieren möchten. Die Amazon S3 S3-URL hat das Format`s3://`*deep_racer_bucket/model_name/prefix/root*.
4. Im Abschnitt „Details“:

   a. Geben Sie den Modellnamen ein.

   b. Fügen Sie eine optionale Beschreibung für das Modell hinzu.

   c. Wenn Sie ein Administratorkonto im Mehrbenutzermodus verwenden, wählen Sie den Benutzer, für den Sie das Modell importieren, aus der Dropdownauswahl aus.
5. Wählen Sie unten auf dem Bildschirm die Schaltfläche Importieren aus.
6. Sobald der Importvorgang gestartet wird, kehren Sie zur Seite „Ihre Modelle“ in der Konsole zurück. Ein Banner oben auf der Seite zeigt den aktuellen Status an und das Modell wird in Ihrer Modellliste mit Import angezeigt. als Status. Wenn der Importvorgang abgeschlossen ist,

bestätigt das Banner den erfolgreichen Import und der Status Ihres Modells ändert sich von Import.. auf Bereit.

Fehlerbehebung

Fehler beim Kopieren des Modells

Wir konnten Ihr Modell trotz mehrerer Versuche nicht kopieren. Wenn sich das Modell immer noch in Ihrem S3-Bucket befindet, wiederholen Sie den Modellimport, indem Sie das Modell aus der Tabelle mit den Modellfehlern auswählen, Aktualisieren und dann Importieren auswählen. Oder, wenn Sie eine lokale Kopie des Modells haben, können Sie es manuell importieren, indem Sie den Schritten auf der Seite Objekte hochladen der Amazon S3 S3-Dokumentation folgen.

Der Amazon S3 S3-Bucket ist nicht vorhanden

Wir konnten das Modell nicht kopieren, da der S3-Bucket, in dem dieses Modell gespeichert war, gelöscht wurde. Wenn Sie eine Kopie des Modells haben, platzieren Sie es in einem S3-Bucket mit dem `deepracer` Namen und versuchen Sie erneut, es zu importieren, indem Sie die Schritte im Importieren Sie Ihr DeepRacer AWS-Modell in die Konsole Abschnitt befolgen.

Ich kann nicht auf den Amazon S3 S3-Bucket zugreifen

Die Berechtigungen für den Amazon S3 S3-Bucket, in dem dieses Modell gespeichert ist, haben sich geändert, sodass wir das Modell nicht kopieren konnten. Dies kann aus zwei Gründen geschehen: Sie haben die Berechtigungen für die AWS DeepRacer S3- oder die DeepRacer AWS-Servicerollenrichtlinie direkt bearbeitet. Wenn Sie die Berechtigungen für Ihren AWS DeepRacer S3-Bucket direkt bearbeitet haben, stellen Sie die Bucket-Berechtigungen wieder her, indem Sie die Schritte unter Hinzufügen einer Bucket-Richtlinie mithilfe der Amazon S3 S3-Konsolenseite mithilfe der folgenden Richtlinie ausführen:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Stmt1586917903457",
            "Effect": "Allow",
            "Principal": {
                "Service": "deepracer.amazonaws.com"
                },
            "Action": [
```

```
                "s3:GetObjectAcl",
                "s3:GetObject",
                "s3:PutObject",
                "s3:PutObjectAcl"
            ],
            "Resource": [
               "arn:aws:s3:::your-bucket-name",
               "arn:aws:s3:::your-bucket-name/*"
            ]
        }
    ]
}
```

Um das Modell nach dem Wiederherstellen der Bucket-Berechtigungen zu importieren, wählen Sie das Modell aus der Tabelle mit den Modellfehlern aus und klicken Sie auf Aktualisieren. Wenn die Seite zum Modellimport angezeigt wird, wählen Sie Import aus.

Die Modelldatei ist nicht vorhanden

Wir können das Modell nicht kopieren, da es aus dem Amazon S3 S3-Bucket gelöscht wurde. Wenn Sie die Datei immer noch haben, versuchen Sie, sie in Ihrem DeepRacer AWS-Bucket wiederherzustellen, wählen Sie dann das Modell aus der Tabelle mit den Modellfehlern aus und klicken Sie auf Aktualisieren. Wenn die Seite zum Modellimport angezeigt wird, wählen Sie Import aus. Wenn Sie über eine lokale Kopie des Modells verfügen, können Sie die Dateien manuell importieren, indem Sie den Schritten auf der Seite Objekte hochladen der Amazon S3 S3-Dokumentation folgen.

Die Coach-Datei existiert nicht

Wir können das Modell nicht kopieren, da die Metadaten des Coach-Checkpoints aus dem Amazon S3 S3-Bucket gelöscht wurden. Wenn Sie die Datei immer noch haben, versuchen Sie, sie in Ihrem DeepRacer AWS-Bucket wiederherzustellen, wählen Sie dann das Modell aus der Tabelle mit den Modellfehlern aus und klicken Sie auf Aktualisieren. Wenn die Seite zum Modellimport angezeigt wird, wählen Sie Import aus. Wenn Sie über eine lokale Kopie des Modells verfügen, können Sie die Dateien manuell importieren, indem Sie den Schritten auf der Seite Objekte hochladen der Amazon S3 S3-Dokumentation folgen.

Die Checkpoint-Datei ist nicht vorhanden

Wir können das Modell nicht kopieren, da die Checkpoint-Dateien aus dem Amazon S3 S3-Bucket gelöscht wurden. Wenn Sie die Dateien immer noch haben, versuchen Sie, sie in Ihrem

DeepRacer AWS-Bucket wiederherzustellen, wählen Sie dann das Modell aus der Tabelle mit den Modellfehlern aus und wählen Sie Aktualisieren aus. Wenn die Seite zum Modellimport angezeigt wird, wählen Sie Import aus. Wenn Sie über eine lokale Kopie der Dateien verfügen, können Sie sie manuell importieren, indem Sie den Schritten auf der Seite Objekte hochladen der Amazon S3 S3-Dokumentation folgen.

Die Modelldatei ist zu groß

Ihre Modelldatei überschreitet die Dateigrößenbeschränkung von 1 GB, die der Service erstellen kann, sodass Ihre Datei bearbeitet wurde. Dieses Modell wird nicht importiert. Um diese Meldung zu entfernen, wählen Sie das Modell aus der Tabelle mit den Modellfehlern aus und klicken Sie auf Löschen.

Die Checkpoint-Datei ist zu groß

Ihre Checkpoint-Datei überschreitet die Dateigrößenbeschränkung von 1 GB, die der Service erstellen kann, sodass Ihre Datei bearbeitet wurde. Dieses Modell wird nicht importiert. Um diese Meldung zu entfernen, wählen Sie das Modell aus der Tabelle mit den Modellfehlern aus und klicken Sie auf Löschen.

Die Metadatendatei ist zu groß

Ihre YAML-Datei überschreitet die Dateigrößenbeschränkung von 10 MB, die der Dienst erstellen kann, sodass Ihre Datei bearbeitet wurde. Dieses Modell wird nicht importiert. Um diese Meldung zu entfernen, wählen Sie das Modell aus der Tabelle mit den Modellfehlern aus und klicken Sie auf Löschen.

Das Modell ist nicht gültig

Wir können Ihr Modell nicht validieren, da es bearbeitet wurde. Wenn Sie über eine Kopie des Modells verfügen, versuchen Sie, diese in Ihrem AWS DeepRacer S3-Bucket zu ersetzen. Wählen Sie dann das Modell aus der Tabelle mit den Modellfehlern aus und wählen Sie Aktualisieren aus. Wenn die Seite zum Modellimport angezeigt wird, wählen Sie Import aus.

Fehlende oder falsche Berechtigungen

Wir konnten das Modell nicht kopieren, da die Berechtigungen, die bei AWS verfügbar waren, DeepRacer als Sie es trainiert haben, entfernt wurden. Um AWS zu autorisieren, die erforderlichen Berechtigungen neu DeepRacer zu erstellen, wählen Sie das Modell aus der Tabelle mit den

Modellfehlern aus und klicken Sie dann auf Aktualisieren. Wenn die Seite zum Modellimport angezeigt wird, wählen Sie Import. AWS DeepRacer erstellt die Berechtigungen neu und kopiert dann das Modell.

Betreiben Sie Ihr DeepRacer AWS-Fahrzeug

Nachdem Sie die Schulung und Evaluierung eines DeepRacer AWS-Modells im DeepRacer AWS-Simulator abgeschlossen haben, können Sie das Modell auf Ihrem DeepRacer AWS-Fahrzeug bereitstellen. Sie können das Fahrzeug so einrichten, dass es auf einer Strecke fährt, und die Leistung des Modells in einer physischen Umgebung bewerten. Dies entspricht einem autonomen Rennen in einer echten Umgebung.

Bevor Sie Ihr Fahrzeug zum ersten Mal fahren lassen, müssen Sie das Fahrzeug einrichten, Softwareupdates installieren und das Antriebskettensubsystem kalibrieren.

Um Ihr Fahrzeug auf einer physischen Strecke fahren zu lassen, benötigen Sie eine Strecke. Weitere Informationen finden Sie unter Erstellen Sie Ihre physische Strecke

Themen

- Lernen Sie Ihr DeepRacer AWS-Fahrzeug kennen
- Wählen Sie ein Wi-Fi-Netzwerk für Ihr DeepRacer AWS-Fahrzeug
- Starten Sie die Gerätekonsole des DeepRacer AWS-Fahrzeugs
- Kalibrieren Sie Ihr AWS-Fahrzeug DeepRacer
- Laden Sie ein Modell auf Ihr DeepRacer AWS-Fahrzeug hoch
- Fahren Sie Ihr DeepRacer AWS-Fahrzeug
- Überprüfen und verwalten Sie Ihre DeepRacer AWS-Fahrzeugeinstellungen
- Ihre DeepRacer AWS-Fahrzeugprotokolle anzeigen

Lernen Sie Ihr DeepRacer AWS-Fahrzeug kennen

Ihr DeepRacer AWS-Fahrzeug ist ein für maschinelles Lernen geeignetes, batteriebetriebenes Allradfahrzeug im Wi-Fi-connected Maßstab 1:18 mit einer 4-Megapixel-Kamera an der Vorderseite und einem Rechenmodul auf Ubuntu-Basis.

Das Fahrzeug kann autonom fahren, indem es eine Inferenz ausführt, die auf einem Reinforcement-Learning-Modell in seinem Computing-Modul basiert. Sie können das Fahrzeug auch manuell ohne ein Reinforcement-Learning-Modell fahren. Wenn Sie noch kein DeepRacer AWS-Fahrzeug erworben haben, können Sie hier eines bestellen.

Das DeepRacer AWS-Fahrzeug wird von einem Bürstenmotor angetrieben. Die Fahrgeschwindigkeit wird von einem Spannungsregler geregelt. Dieser steuert, wie schnell sich der Motor dreht. Der Servomechanismus (Servo), der das Lenksystem bedient, ist durch die schwarze Abdeckung im DeepRacer AWS-Fahrzeugchassis geschützt.

Themen

- Inspizieren Sie Ihr DeepRacer AWS-Fahrzeug
- Laden und installieren Sie Ihre DeepRacer AWS-Batterien
- Testen Sie Ihr DeepRacer AWS-Rechenmodul
- Schalten Sie Ihr DeepRacer AWS-Fahrzeug aus
- DeepRacer AWS-Fahrzeug-LED-Anzeigen
- Ersatzteile für DeepRacer AWS-Geräte

Inspizieren Sie Ihr DeepRacer AWS-Fahrzeug

Wenn Sie Ihre DeepRacer AWS-Fahrzeugbox öffnen, sollten Sie die folgenden Komponenten und Zubehörteile finden:

Komponenten	Kommentare
Fahrzeugchassis [1]	Enthält eine Frontkamera zur Erfassung von Fahrerfahrungen im Fahrzeug und das Computing-Modul für autonomes Fahren. Sie können die von der Kamera aufgenommenen Bilder über die Gerätekonsole des Fahrzeugs als Streaming-Video ansehen. Das Chassis enthält einen Elektro-Bürstenmotor, einen elektronischen Drehzahlregler (Electronic Speed Controller, ESC) und eine Servomechanik (Servo).
Fahrzeugkarosserie [2]	Entfernen Sie diese bei der Einrichtung des Fahrzeugs.
Adapterkabel Micro-USB auf USB-A [3]	Verwenden Sie dieses Kabel für die USB-OTG-Funktionalität.
Verarbeitungsbatterie [4]	Verwenden Sie dies, um das Rechenmodul mit Strom zu versorgen, das Inferenz für ein heruntergeladenes DeepRacer AWS-Reinf orcement-Learning-Modell ausführt.
Anschlusskabel Verarbeitungsbatterie [5]	Verwenden Sie dieses USB C/USB C-Kabel, um das Datenverarbeitungsmodul an das Netzteil anzuschließen. Wenn Sie eine Dell-Vera rbeitungsbatterie haben, ist dieses Kabel länger.
Netzstromkabel [6a]	Mit dieser Komponente schließen Sie das Netzteil an eine Steckdose an.
Netzteil [6b]	Mit dieser Komponente laden Sie Verarbeit ungsbatterie und das Datenverarbeitungsmodul auf.

Komponenten	Kommentare
Stifte (Ersatzteile) [7]	Mit dieser Komponente befestigen Sie das Datenverarbeitungsmodul am Fahrzeugchassis. Dies ist Sonderzubehör.
Fahrzeugbatterie [8]	Ein LiPo 7,4-V-Batteriepaket zur Stromversorgung des Motors.
Ladeadapter für Fahrzeugbatterie [9a]	Verwenden Sie diese Komponente, um die Fahrzeugbatterie für den Fahrzeugantrieb aufzuladen.
Ladekabel für die Fahrzeugbatterie [9b]	Verwenden Sie das Netzteil, um das Fahrzeugbatterieladegerät mit einer Steckdose zu verbinden.
Batterieentsperrkabel [10]	Verwenden Sie diese Option, wenn Ihre Batterie in den Sperrzustand wechselt.

Um Ihr DeepRacer AWS-Fahrzeug einzurichten, müssen Sie außerdem die folgenden Artikel bereit haben:

- Einen Computer mit USB-Anschluss und Internetzugang.
- Ein WLAN, das mit dem Internet verbunden ist.
- Ein AWS-Konto.

Befolgen Sie nun die Anweisungen im nächsten Abschnitt, um sicherzustellen, dass die Fahrzeugbatterie und die Power-Bank geladen sind.

Laden und installieren Sie Ihre DeepRacer AWS-Batterien

Ihr DeepRacer AWS-Fahrzeug verfügt über zwei Stromquellen: die Fahrzeugbatterie und die Powerbank des Rechenmoduls.

Die Power-Bank versorgt das Computing-Modul. Das Rechenmodul hält die Wi-Fi-Verbindung aufrecht, führt Inferenzen gegen ein DeepRacer bereitgestelltes AWS-Modell durch und gibt dem Fahrzeug einen Befehl, eine Aktion auszuführen.

Die Fahrzeugbatterie versorgt den Motor, um das Fahrzeug zu bewegen. Es gibt zwei Sätze von Kabeln. Der zweiadrige Satz der roten und schwarzen Kabel wird verwendet, um eine Verbindung zum ESC des Fahrzeugs herzustellen, und die dreiadrigen blauen (oder schwarzen), weißen und roten Kabel dienen zum Verbinden mit dem Ladegerät. Für die Fahrt sollte nur der zweiadrige Kabelsatz mit dem Fahrzeug verbunden werden.

Nach dem vollständigen Aufladen sinkt die Batteriespannung, während sich die Batterien entladen. Durch den Spannungsabfall sinkt auch das verfügbare Drehmoment. Daher hat dieselbe Geschwindigkeitseinstellung eine langsamere Geschwindigkeit auf der Strecke zur Folge. Wenn die Batterie vollständig leer ist, stoppt das Fahrzeug. Für autonomes Fahren unter normalen Bedingungen hält die Batterie in der Regel 15 bis 25 Minuten. Um ein konsistentes Verhalten sicherzustellen, sollte die Batterie jeweils nach 15 Minuten Nutzung aufgeladen werden.

Um die Fahrzeugbatterie und die Power-Bank zu montieren und aufzuladen, führen Sie die folgenden Schritte aus.

1. Entfernen Sie Ihre DeepRacer AWS-Fahrzeugschale.
2. Entfernen Sie die vier Chassisstifte. Heben Sie das Fahrzeugchassis mit den noch verbundenen Kabeln vorsichtig an.
3. Gehen Sie wie folgt vor, um die Fahrzeugbatterie aufzuladen und einzusetzen:

    a. Um die Batterie aufzuladen, schließen Sie den dreiadrigen Kabelsatz von der Batterie an das Ladegerät an, um die Batterie mit dem Netzteil zu verbinden, und schließen das Netzteil dann an eine Steckdose oder an einen USB-Anschluss an, wenn zum Laden der Batterie ein USB-Kabel verwendet wird.

        Eine grafische Darstellung, wie Sie die Fahrzeugbatterie mit dem beigefügten Ladegerät aufladen, finden Sie unter the section called “Wie lade ich die Batterie des Antriebsmoduls des Fahrzeugs auf”.

    b. Schließen Sie nach dem Laden der Batterie den zweiadrigen Kabelsatz des Fahrzeugbatteriekabels an den schwarzen und roten Kabelanschluss Ihres Fahrzeugs an.

    c. Um die Fahrzeugbatterie zu sichern, befestigen Sie diese mit den beiliegenden Befestigungsbändern unter dem Fahrzeugchassis.

        Achten Sie darauf, dass sich alle Kabel im Fahrzeuginneren befinden.

    d. Um zu überprüfen, ob die Fahrzeugbatterie geladen ist, gehen Sie wie folgt vor:

        i. Bewegen Sie den Netzschalter des Fahrzeugs, um das Fahrzeug einzuschalten.

ii. Warten Sie auf zwei kurze Pieptöne.

Wenn Sie die Pieptöne nicht hören, ist das Fahrzeug nicht aufgeladen. Entfernen Sie die Batterie aus dem Fahrzeug und wiederholen Sie Schritt 1 oben, um die Batterie wieder aufzuladen.

iii. Wenn Sie das Fahrzeug nicht verwenden, schieben Sie den Netzschalter des Fahrzeugs zurück, um die Fahrzeugbatterie auszuschalten.

4. Um den Ladezustand des Netzteils zu überprüfen, gehen Sie wie folgt vor:

   a. Drücken Sie den Netzschalter am Netzteil.

   b. Prüfen Sie die vier LED-Leuchten neben dem Netzschalter, um den Ladezustand zu bestimmen.

      Wenn alle vier LED-Leuchten leuchten, ist das Netzteil vollständig geladen. Wenn keine der LED-Leuchten leuchtet, muss das Netzteil aufgeladen werden.

   c. Um das Netzteil aufzuladen, stecken Sie den USB-C-Stecker des Netzteils in den USB-C-Anschluss des Netzteils. Es dauert einige Zeit, bis das Netzteil vollständig aufgeladen ist. Wenn es aufgeladen ist, wiederholen Sie Schritt 4, um zu prüfen, ob das Netzteil vollständig aufgeladen ist.

5. Um die Power-Bank zu montieren, gehen Sie wie folgt vor:

   a. Stecken Sie das Netzteil in die Halterung. Netzschalter und USB-C-Anschluss müssen zur Rückseite des Fahrzeugs ausgerichtet sein.

   b. Verwenden Sie das Befestigungsband, um das Netzteil sicher am Fahrzeugchassis zu befestigen.

> **Note**
>
> Schließen Sie in diesem Schritt noch nicht das Netzteil an das Computing-Modul an.

Testen Sie Ihr DeepRacer AWS-Rechenmodul

Testen Sie das Computing-Modul, um sicherzustellen, dass es erfolgreich gestartet werden kann. Um das Modul mit einer externen Stromquelle zu testen, führen Sie die folgenden Schritte aus:

So testen Sie das Computing-Modul Ihres Fahrzeugs:

1. Schließen Sie das Computing-Modul an eine Stromquelle an. Schließen Sie das Netzkabel an das Netzteil an, stecken Sie das Netzkabel in eine Steckdose und stecken Sie den USB-C-Stecker des Netzteils in den USB-C-Anschluss des Computing-Moduls.
2. Schalten Sie das Computing-Modul des Fahrzeugs ein, indem Sie den Netzschalter am Modul drücken.
3. Um den Status des Computing-Moduls zu überprüfen, überprüfen Sie die LED-Leuchten:

   - Durchgehend blau

     Das Computing-Modul ist gestartet, mit dem angegebenen WLAN verbunden und betriebsbereit.

     In diesem Status können Sie sich am Computing-Modul anmelden. Sie müssen es nur mit einem HDMI-Kabel an einen Monitor anschließen und mit einer USB-Maus und einer USB-Tastatur verbinden. Verwenden Sie für die erstmalige Anmeldung `deepracer` sowohl für den Benutzernamen als auch für das Passwort. Sie werden dann aufgefordert, das Passwort für zukünftige Anmeldungen zurückzusetzen. Wählen Sie aus Sicherheitsgründen ein starkes neues Passwort.
   - Blinkend rot

     Das Computing-Modul befindet sich im Einrichtungsmodus.
   - Durchgehend gelb

     Das Computing-Modul wird initialisiert.
   - Durchgehend rot

     Das Computing-Modul konnte sich nicht mit dem WLAN verbinden.
4. Wenn Sie mit dem Test fertig sind, drücken Sie den Netzschalter am Computing-Modul, um es auszuschalten. Trennen Sie es dann von der externen Stromquelle.

Schalten Sie Ihr DeepRacer AWS-Fahrzeug aus

Um Ihr DeepRacer AWS-Fahrzeug auszuschalten, trennen Sie das Fahrzeug von der externen Stromquelle. Sie können auch den Netzschalter am Gerät drücken, bis die Stromanzeige erlischt.

DeepRacer AWS-Fahrzeug-LED-Anzeigen

Ihr DeepRacer AWS-Fahrzeug verfügt über zwei LED-Anzeigen für den Fahrzeugstatus bzw. für die anpassbare visuelle Identifizierung Ihres Fahrzeugs.

Die Details werden wie folgt erläutert.

Themen

- LED-Anzeigen DeepRacer für AWS-Fahrzeugsysteme
- DeepRacer AWS-Fahrzeugidentifikation LEDs

LED-Anzeigen DeepRacer für AWS-Fahrzeugsysteme

Die LED-Anzeigen des DeepRacer AWS-Fahrzeugsystems befinden sich auf der linken Seite des Fahrzeugchassis, wenn sich das Fahrzeug in der vorderen Position vor Ihnen befindet.

Die drei Systeme LEDs befinden sich hinter der RESET-Taste. Die erste LED (links im Sichtfeld) zeigt den Status der Systemleistung an. Die zweite (mittlere) LED ist für die zukünftige Verwendung reserviert. Die letzte (rechte) LED zeigt den Status der WLAN-Verbindung an.

LED-Typ	Farbe	Status
Stromversorgung	Aus	Es ist kein Netzteil vorhanden.
	Gelb blinkend	BIOS und Betriebssystem werden geladen.
	Dauerhaft gelb	Betriebssystem ist geladen.
	Dauerhaft blau	Eine Anwendung wird ausgeführt.
	Blau blinken	Eine Softwareaktualisierung wird ausgeführt.
	Dauerhaft rot	Beim Hochfahren des Systems oder beim Starten einer Anwendung tritt ein Fehler auf.
WLAN	Aus	Es besteht keine WLAN-Verb indung.
	Blau blinken	Das Fahrzeug stellt eine Verbindung mit dem WLAN-Netzwerk her.
	Dauerhaft rot für 2 Sekunden und dann aus	Die WLAN-Verbindung ist fehlgeschlagen.
	Dauerhaft blau	Die WLAN-Verbindung wurde hergestellt.

DeepRacer AWS-Fahrzeugidentifikation LEDs

Die kundenspezifischen DeepRacer AWS-Fahrzeuge LEDs befinden sich am Heck des Fahrzeugs. Sie dienen zur Identifizierung Ihres Fahrzeugs in Rennen, wenn mehrere Fahrzeuge vorhanden sind. Sie können die DeepRacer AWS-Gerätekonsole verwenden, um ihnen eine unterstützte Farbe Ihrer Wahl zuzuweisen.

Ersatzteile für DeepRacer AWS-Geräte

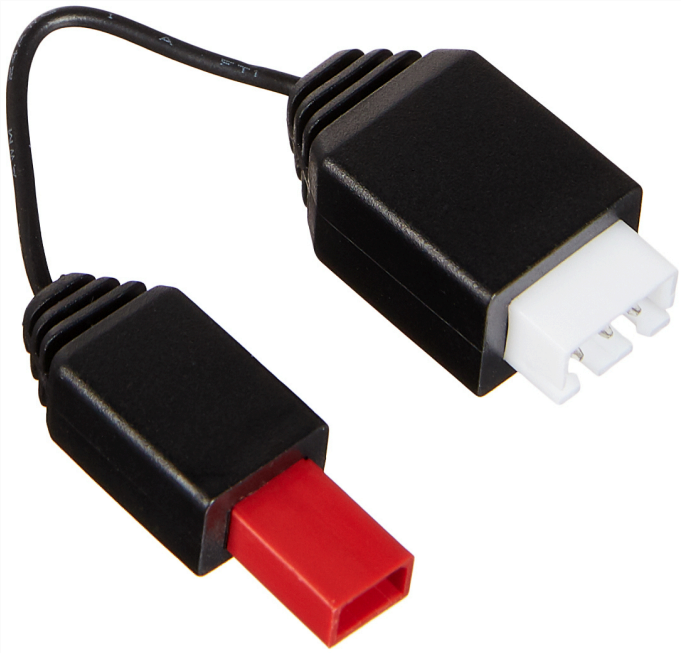
> **Note**
>
> Das DeepRacer AWS-Gerät verwendet die WLToys ferngesteuerten Fahrzeuggehäuse A949 und A979. Besuchen Sie die DeepRacer AWS-Storefront, um eine aktuelle Liste der verfügbaren Teile für Ihr DeepRacer AWS-Gerät zu durchsuchen.

Ersatzteile für DeepRacer AWS-Geräte

Teil	Name
	Reserve-Rechenbatterie

Teil	Name
	Ersatzrechenmodul

Teil	Name
	Reserve-Rechenbatterie

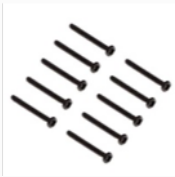
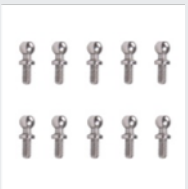
Teil	Name
	Lithium-Batterie mit 7,4 V, 1100 mAh
	AWS-Entsperrkabel für DeepRacer Autobatterien
	Reifen
	Frontstoßstange

Teil	Name
	Querlenker
	Zugstange
	C-Stil-Halterung
	Antriebswelle
	Rundkopfschraube, M2 x 17,5 mm
	Fahrzeugchassis-Boden
	Lenkgestänge

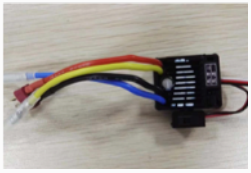
Teil	Name
	Hinterradaufhängungsrahmen
	Metall-Sechskantset
	Getriebegehäuse
	Differenzialgehäuse
	Differenzialantriebsschale
	Heckschürze vorne
	Servohalterung

Teil	Name
	Zentralantriebswelle
	Stoßdämpferrahmen
	Servoarm
	Differentialmechanismus
	Untersetzungsgetriebe
	Motorsockel
	17g Lenkmotor

Teil	Name
	Schraubendichtung für Motor
	390 Motor
	Sechskantverbinder 4 x 8 x 3 mm
	Sechskantverbinder 8 x 12 x 3,5 mm
	Kugellager 7 x 11 x 3 mm
	Kugellager 8 x 12 x 3,5 mm
	Mittelachsenscheibenplatte

Teil	Name
	Schraube 2,6 x 6 mm
	Schraube 2 x 7 mm
	Schraube 2,5 x 8 mm
	Schraube 2 x 16 mm
	Schraube 2,5 x 6 mm
	Schraube M3 x 5 mm
	Kugelschraube 10,8 x 4 mm

Teil	Name
	Schraube 2 x 6 mm
	Schraube 2 x 9,5 mm
	M3-Kontermutter
	Achsengelenkstift
	Antriebswelle
	Schwenkarmstift
	Schraube 2 x 29 KM

Teil	Name
	Haarstift
	Frontstoßdämpfer
	Ladegerät
	Motorritzel aus Metall
	Heckstoßdämpfer
	ESC

Wählen Sie ein Wi-Fi-Netzwerk für Ihr DeepRacer AWS-Fahrzeug

Wenn Sie Ihr DeepRacer AWS-Fahrzeug zum ersten Mal öffnen, müssen Sie es so einrichten, dass es eine Verbindung zu einem Wi-Fi-Netzwerk herstellt. Schließen Sie dieses Setup ab, um die

Fahrzeugsoftware zu aktualisieren und die IP-Adresse für den Zugriff auf die Gerätekonsole des Fahrzeugs zu erhalten.

In diesem Abschnitt werden Sie durch die Schritte für die folgenden Aufgaben geführt:

- Verbinden Sie Ihren Notebook- oder Desktop-Computer mit dem Fahrzeug.
- Richten Sie die WLAN-Verbindung des Fahrzeugs ein.
- Aktualisieren Sie die Software des Fahrzeugs.
- Ermitteln Sie die IP-Adresse des Fahrzeugs.
- Führen Sie eine Probefahrt mit dem Fahrzeug durch.

Verwenden Sie einen Notebook- oder Desktop-Computer, um die Setup-Aufgaben auszuführen. Wir bezeichnen diesen Setup-Computer als "Ihren Computer", um mögliche Verwechslungen mit dem Computing-Modul des Fahrzeugs zu vermeiden (setzt Ubuntu ein).

Nach der ersten Einrichtung der WLAN-Verbindung können Sie denselben Anweisungen folgen, um ein anderes WLAN auszuwählen.

> **Note**
>
> AWS unterstützt DeepRacer kein Wi-Fi-Netzwerk, für dessen Anmeldung eine aktive Captcha-Überprüfung erforderlich ist.

Themen

- Machen Sie sich bereit, eine Wi-Fi-Verbindung für Ihr DeepRacer AWS-Fahrzeug einzurichten
- Richten Sie eine Wi-Fi-Verbindung ein und aktualisieren Sie die Software Ihres DeepRacer AWS-Fahrzeugs

Machen Sie sich bereit, eine Wi-Fi-Verbindung für Ihr DeepRacer AWS-Fahrzeug einzurichten

Um die Wi-Fi-Verbindung Ihres Fahrzeugs einzurichten, verbinden Sie Ihren Laptop oder Desktop-Computer über das mitgelieferte USB-to-USB C-Kabel mit dem Rechenmodul Ihres Fahrzeugs.

Um Ihren Computer mit dem Computing-Modul Ihres Fahrzeugs zu verbinden, führen Sie die folgenden Schritte aus.

1. Stellen Sie sicher, dass Ihr Computer vom WLAN getrennt ist, bevor Sie Ihr Gerät verbinden.
2. Stecken Sie das USB-Ende des USB-to-USB C-Kabels in den USB-Anschluss Ihres Computers.
3. Stecken Sie das USB-C-Ende des Kabels in den USB-C-Anschluss Ihres Fahrzeugs.

Jetzt können Sie die WLAN-Verbindung Ihres Fahrzeugs einrichten.

Richten Sie eine Wi-Fi-Verbindung ein und aktualisieren Sie die Software Ihres DeepRacer AWS-Fahrzeugs

Bevor Sie die Schritte zum Einrichten der WLAN-Verbindung ausführen, vergewissern Sie sich, dass Sie die Schritte unter the section called “ Machen Sie sich bereit für die Einrichtung von Wi-Fi ” abgeschlossen haben.

1. Schauen Sie sich die Unterseite Ihres Fahrzeugs an und notieren Sie sich das unter Host name aufgeführte Passwort. Sie benötigen es, um sich an der Gerätekonsole anzumelden und die Einrichtung durchzuführen.
2. Wechseln Sie auf Ihrem Computer zu `https://deepracer.aws`, um die Gerätesteuerkonsole Ihres Fahrzeugs zu starten.
3. Wenn angezeigt wird, dass die Verbindung nicht privat oder sicher ist, führen Sie einen der folgenden Schritte aus.

    a. Wählen Sie in Chrome „Erweitert“ und anschließend „Weiter zu ***<device_console_ip_address>*** (unsicher)“.

    b. Wählen Sie in Safari Details aus, folgen Sie dem Link Diese Website besuchen und wählen Sie Websites besuchen aus. Wenn Sie zur Eingabe Ihres Passworts aufgefordert werden, um Einstellungen für die Zertifikatsvertrauensstellungen zu aktualisieren, geben Sie das Passwort ein. Wählen Sie dann Einstellungen aktualisieren aus.

    c. Wählen Sie in Opera Trotzdem fortfahren aus, wenn Sie vor einem ungültigen Zertifikat gewarnt werden.

    d. Wählen Sie in Edge Details und dann Weiter zur Webseite (nicht empfohlen) aus.

    e. Wählen Sie in Firefox Erweitert aus. Wählen Sie dann Ausnahme hinzufügen aus und wählen Sie Sicherheitsausnahme bestätigen aus.
4. Geben Sie unter Ihr DeepRacer AWS-Fahrzeug entsperren das in Schritt 1 angegebene Passwort ein und wählen Sie dann Access vehicle aus.

5. Wählen Sie im Bereich Connect your vehicle to your Wi-Fi network (Verbinden Sie Ihr Fahrzeug mit Ihrem WLAN) Ihren WLAN-Namen aus dem Dropdown-Menü Wi-Fi network name (SSID) (WLAN-Name (SSID)) aus, geben Sie das Passwort Ihres WLANs unter Wi-Fi-password (WLAN-Passwort) ein und wählen Sie Connect (Verbinden) aus.

6. Warten Sie, bis sich der Status der WLAN-Verbindung von Connecting to Wi-Fi network... (Verbinden mit dem WLAN...) zu Connected (Verbunden) ändert. Wählen Sie anschließend Weiter.
7. Wenn im Bereich Software Update ein Softwareupdate erforderlich ist, schalten Sie das Computing-Modul des Fahrzeugs mit dem mitgelieferten Netzkabel und dem Netzteil ein und wählen Sie dann Install software update (Softwareupdate installieren) aus.

   Die Stromversorgung des Fahrzeugs mit einer externen Stromquelle sorgt dafür, dass eine Unterbrechung des Softwareupdates aufgrund einer leeren Batterie des Computing-Moduls vermieden wird.
8. Warten Sie, bis sich der Status des Softwareupdates von Installing software update (Softwareupdate wird installiert) zu Software update installed successfully (Softwareupdate erfolgreich installiert) ändert.
9. Notieren Sie die unter Wi-Fi network details (WLAN-Details) angezeigte IP-Adresse. Sie benötigen diese, um die Gerätesteuerkonsole des Fahrzeugs nach der Erstinstallation und jeder späteren Änderung der WLAN-Einstellungen zu öffnen.

Starten Sie die Gerätekonsole des DeepRacer AWS-Fahrzeugs

Nachdem Sie die WLAN-Verbindung des Fahrzeugs eingerichtet und die erforderlichen Softwareupdates installiert haben, sollten Sie die Gerätekonsole öffnen, um zu überprüfen, ob die Netzwerkverbindung des Fahrzeugs funktioniert. Später können Sie die Gerätekonsole starten, um die anderen Einstellungen des Fahrzeugs zu überprüfen, zu kalibrieren und zu verwalten. Der Prozess umfasst die Anmeldung bei der Gerätekonsole Ihres Fahrzeugs unter Verwendung der IP-Adresse Ihres Fahrzeugs.

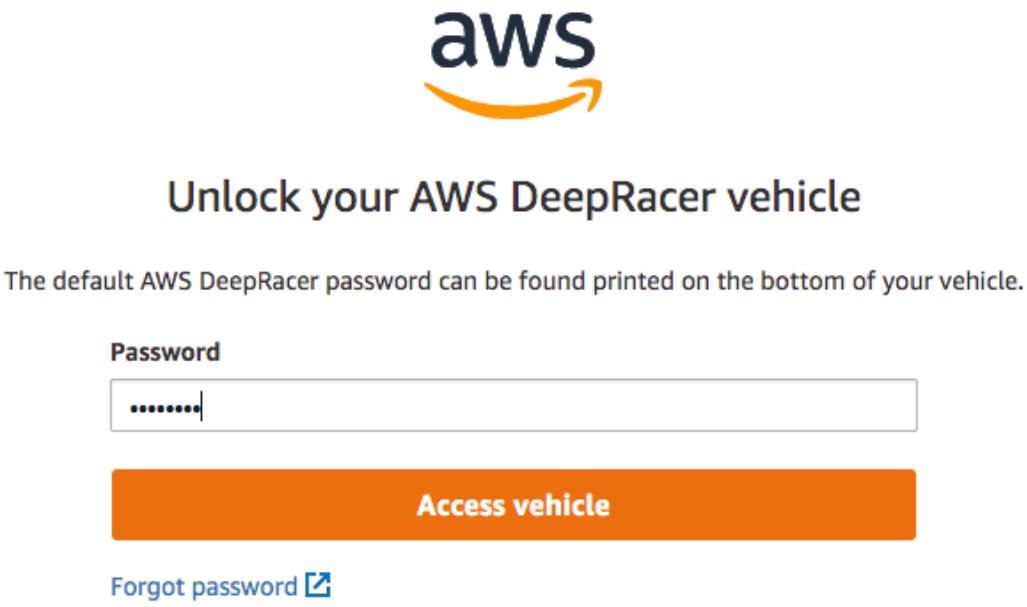
Die Gerätesteuerkonsole wird im Fahrzeug gehostet und über die IP-Adresse aufgerufen, die Sie am Ende des Abschnitts WLAN-Einrichtung erhalten haben.

Um über die Wi-Fi-Verbindung auf die Gerätekonsole Ihres DeepRacer AWS-Fahrzeugs zuzugreifen

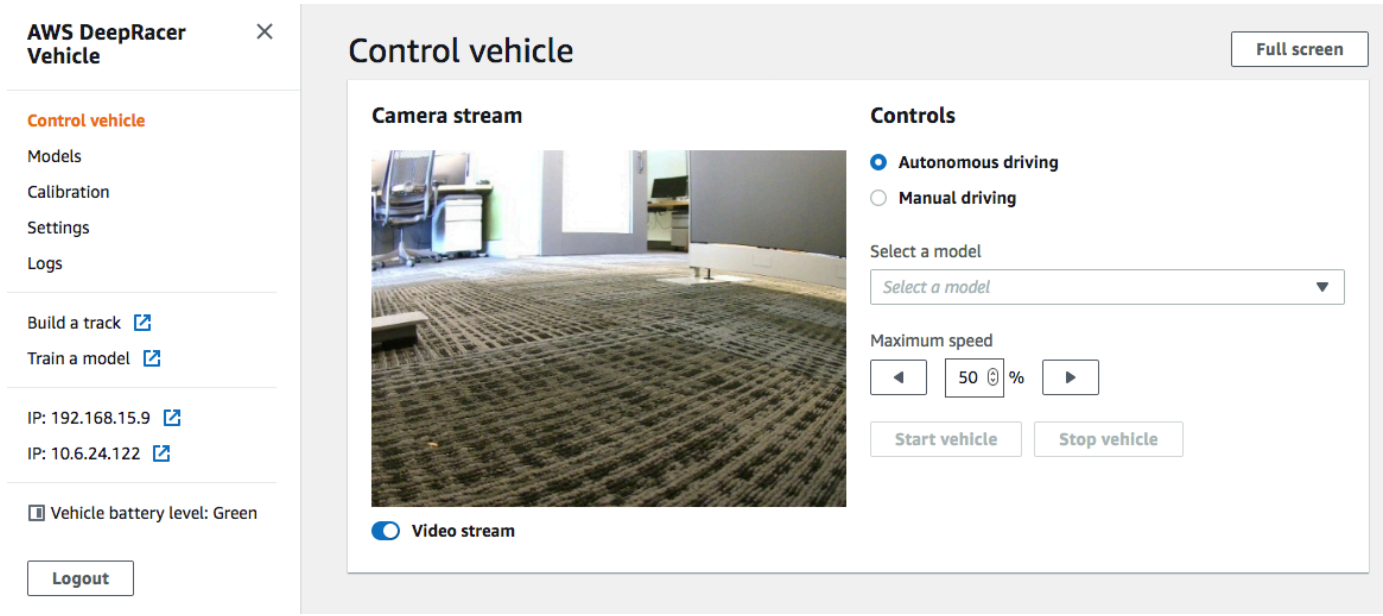
1. Um auf die Gerätekonsole Ihres Fahrzeugs zuzugreifen, öffnen Sie einen Webbrowser auf Ihrem Computer, Tablet oder Smartphone und geben Sie die IP-Adresse Ihres Fahrzeugs in die Adressleiste ein.

   Sie können diese IP-Adresse beim Einrichten der WLAN-Verbindung des Fahrzeugs abrufen. Zur Veranschaulichung verwenden wir `10.92.206.61` als Beispiel.

   Wenn eine Warnmeldung angezeigt wird, dass die Verbindung ist nicht privat oder sichere ist, ignorieren Sie die Meldung fahren mit der Verbindung zur Gerätekonsole fort.
2. Geben Sie unter Ihr DeepRacer AWS-Fahrzeug entsperren das Passwort der Gerätekonsole in das Feld Passwort ein und wählen Sie dann Access vehicle aus.



   Sie finden das Standardpasswort auf der Unterseite Ihres Fahrzeugs (unter Host Name).
3. Wenn Sie erfolgreich angemeldet sind, sehen Sie die folgende Startseite der Gerätekonsole.



Nun können Sie Ihr Fahrzeug kalibrieren und betreiben. Wenn dies Ihre erste Inbetriebnahme des Fahrzeugs ist, fahren Sie jetzt mit Kalibrieren des Fahrzeugs fort.

Kalibrieren Sie Ihr AWS-Fahrzeug DeepRacer

Um die beste Leistung zu erzielen, ist es wichtig, dass Sie einige physische Teile Ihres DeepRacer AWS-Fahrzeugs kalibrieren. Wenn Sie ein nicht kalibriertes Fahrzeug verwenden, kann dies zu zusätzlicher Unsicherheit beim Testen Ihres Modells führen. Wenn die Leistung des Fahrzeugs nicht optimal ist, sind Sie möglicherweise versucht, nur den Deep-Learning-Modellcode anzupassen. Bei einer mechanischen Ursache können Sie jedoch die Fahrzeugleistung auf diese Art nicht verbessern. Justieren Sie die Mechanik mittels Kalibrierung.

Um Ihr DeepRacer AWS-Fahrzeug zu kalibrieren, legen Sie den Betriebszyklusbereich für das elektronische Steuersystem (ECS) des Fahrzeugs bzw. seinen Servomechanismus (Servo) fest. Sowohl der Servo als auch das ECS akzeptieren PWM-Signale (Pulsweitenmodulation)) als Steuereingang vom Computing-Modul des Fahrzeugs. Das Computing-Modul passt sowohl die Geschwindigkeit als auch den Lenkwinkel des Fahrzeugs an, indem es das Tastverhältnis der PWM-Signale ändert.

Die maximale Geschwindigkeit und der Lenkwinkel definieren den Umfang des Aktionsraums. Sie können die maximale Geschwindigkeit und den maximalen Lenkwinkel während des Trainings in der Simulation festlegen. Wenn Sie das trainierte Modell auf Ihrem DeepRacer AWS-Fahrzeug einsetzen,

um auf einer realen Strecke zu fahren, müssen die Höchstgeschwindigkeit und der Lenkwinkel des Fahrzeugs so kalibriert werden, dass sie denen entsprechen, die im Simulationstraining verwendet wurden.

Um sicherzustellen, dass die realen Erfahrungen mit den simulierten Erfahrungen übereinstimmen, sollten Sie Ihr Fahrzeug so kalibrieren, dass die maximale Geschwindigkeit und der maximale Lenkwinkel zwischen Simulation und der realen Welt übereinstimmen. Im Allgemeinen gibt es zwei Möglichkeiten, diese Kalibrierung durchzuführen:

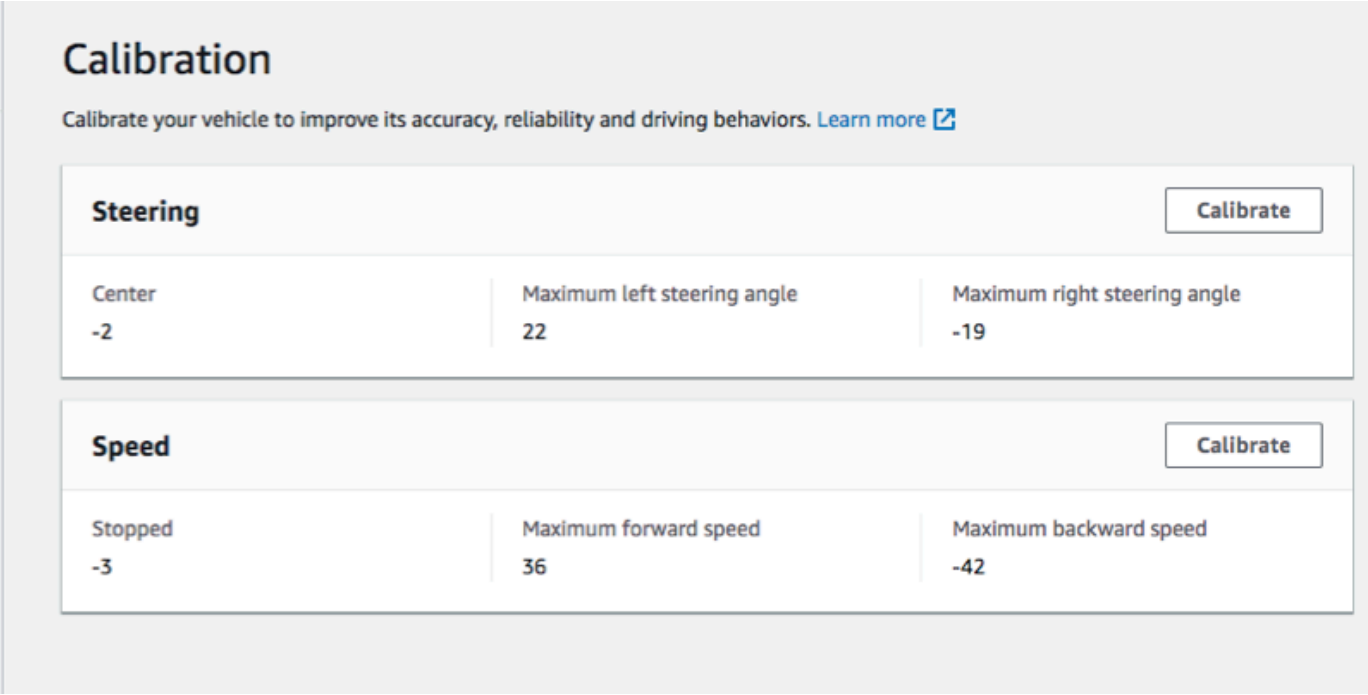
- Definieren Sie den Aktionsraum im Training und kalibrieren Sie das physische Fahrzeug entsprechend den Einstellungen.
- Messen Sie die tatsächliche Leistung Ihres Fahrzeugs und ändern Sie die Einstellungen des Aktionsraums in der Simulation.

Ein robustes Modell kann bestimmte Unterschiede zwischen der Simulation und der realen Welt ausgleichen. Sie sollten jedoch mit beiden Ansätzen und Iterationen experimentieren, um die besten Ergebnisse zu erzielen.

Schalten Sie das Computing-Modul ein, bevor Sie mit der Kalibrierung beginnen. Nachdem es gestartet ist und die Betriebs-LED dauerhaft blau leuchtet, schalten Sie die Fahrzeugbatterie ein. Nachdem Sie zwei kurze und einen langen Piepton gehört haben, können Sie mit der Kalibrierung fortfahren.

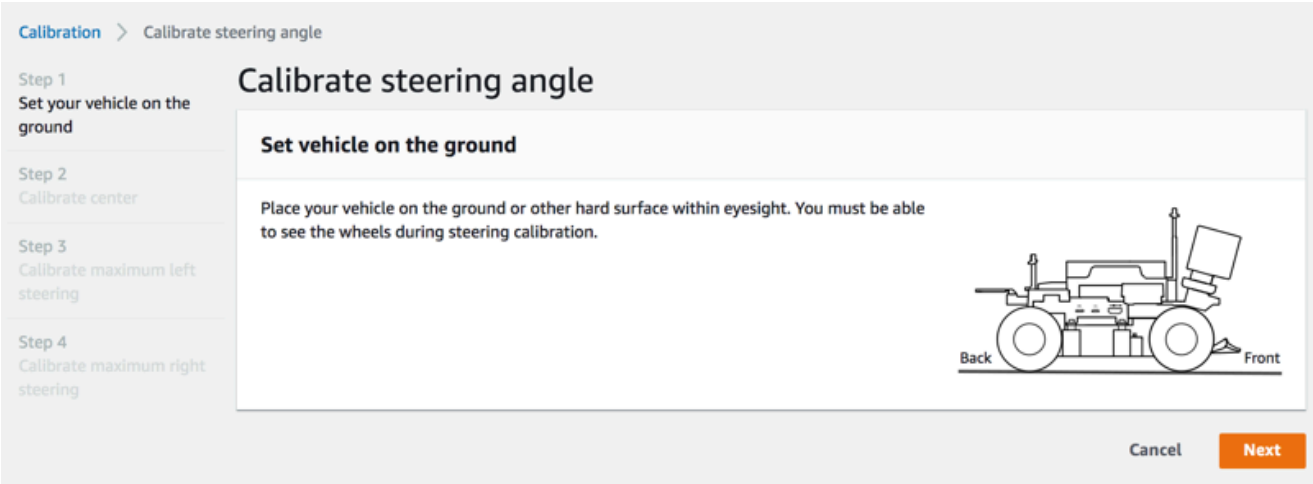
So kalibrieren Sie Ihr DeepRacer AWS-Fahrzeug so, dass es den Trainingseinstellungen entspricht:

1. Befolgen Sie diese Anweisungen, um auf Ihr Fahrzeug zuzugreifen und die Gerätesteuerkonsole zu öffnen.
2. Wählen Sie im Hauptmenü Calibration (Kalibrierung) aus.



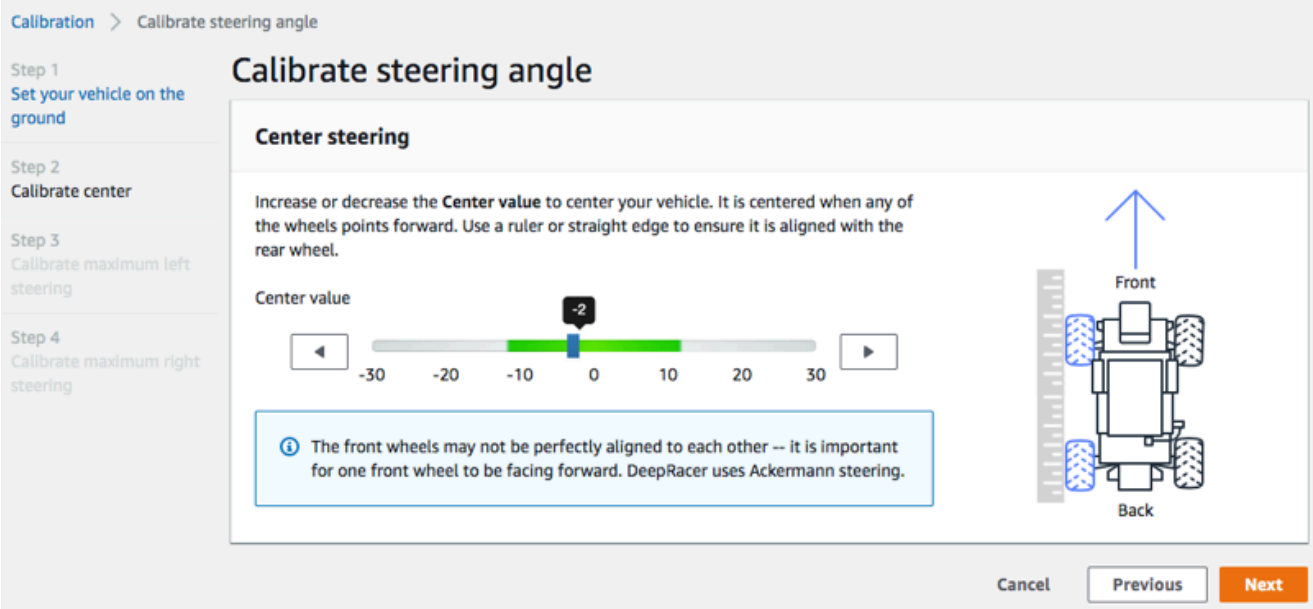
3. Wählen Sie auf der Seite Calibration (Kalibrierung) Calibrate (Kalibrieren) in Steering (Steuerung) aus und führen Sie dann die folgenden Schritte aus, um die maximalen Lenkwinkel des Fahrzeugs zu kalibrieren.

   a. Stellen Sie das Fahrzeug auf den Boden oder eine andere harte Oberfläche, auf der Sie die Räder während der Lenkkalibrierung sehen können. Wählen Sie Weiter.



   Das Lenken eines Fahrzeugs auf einer Strecke erfordert viel kleinere Lenkwinkel als bei Rädern in der Luft. Um die tatsächlichen Lenkwinkel der Räder zu messen, ist es daher wichtig, dass Sie das Fahrzeug auf der Fahrbahnoberfläche abstellen.

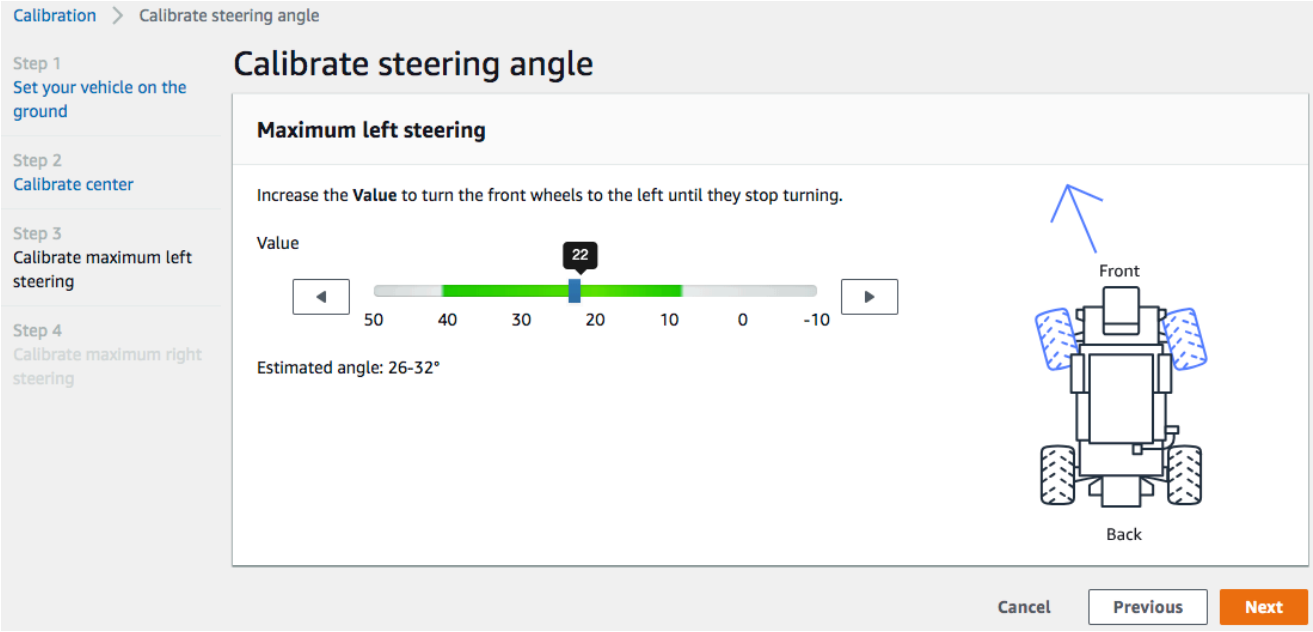
b. Bewegen Sie unter Center steering (Lenkmitte) den Regler schrittweise oder drücken Sie den linken oder rechten Pfeil, bis mindestens eines der Vorderräder am Hinterrad auf derselben Seite ausgerichtet ist. Wählen Sie Weiter.



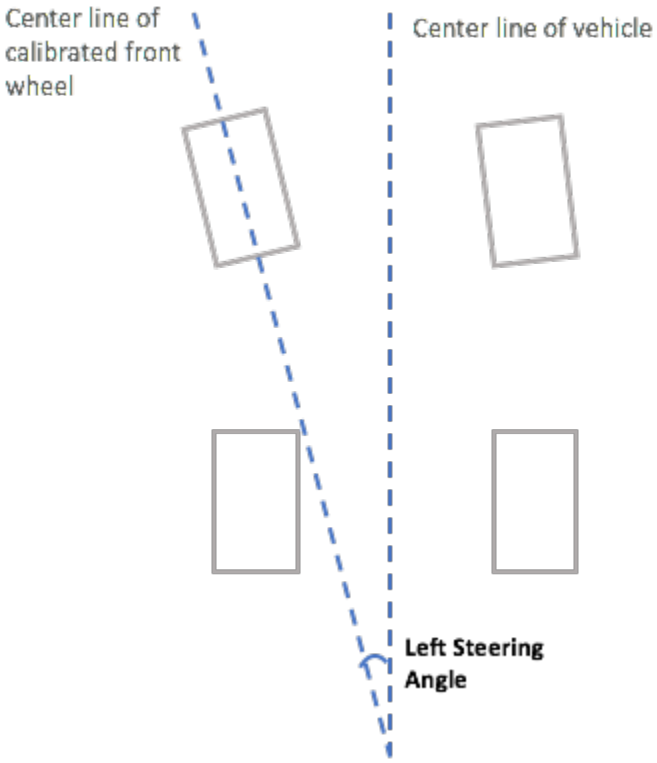
AWS DeepRacer verwendet die Ackermann-Vorderradlenkung, um Räder innerhalb und außerhalb einer Kurve zu drehen. Das bedeutet, dass sich das linke und rechte Vorderrad im Allgemeinen in unterschiedlichen Winkeln drehen. In AWS DeepRacer erfolgt die Kalibrierung anhand des Mittelwerts. Daher müssen Sie die Räder auf der ausgewählten Seite so einstellen, dass sie in einer geraden Linie ausgerichtet sind.

> **Note**
>
> Achten Sie darauf, Ihr DeepRacer AWS-Fahrzeug gut zu kalibrieren, damit es die Mittellenkung so gerade wie möglich halten kann. Sie können dies testen, indem Sie das Fahrzeug von Hand schieben, um sicherzustellen, dass es gerade fährt.

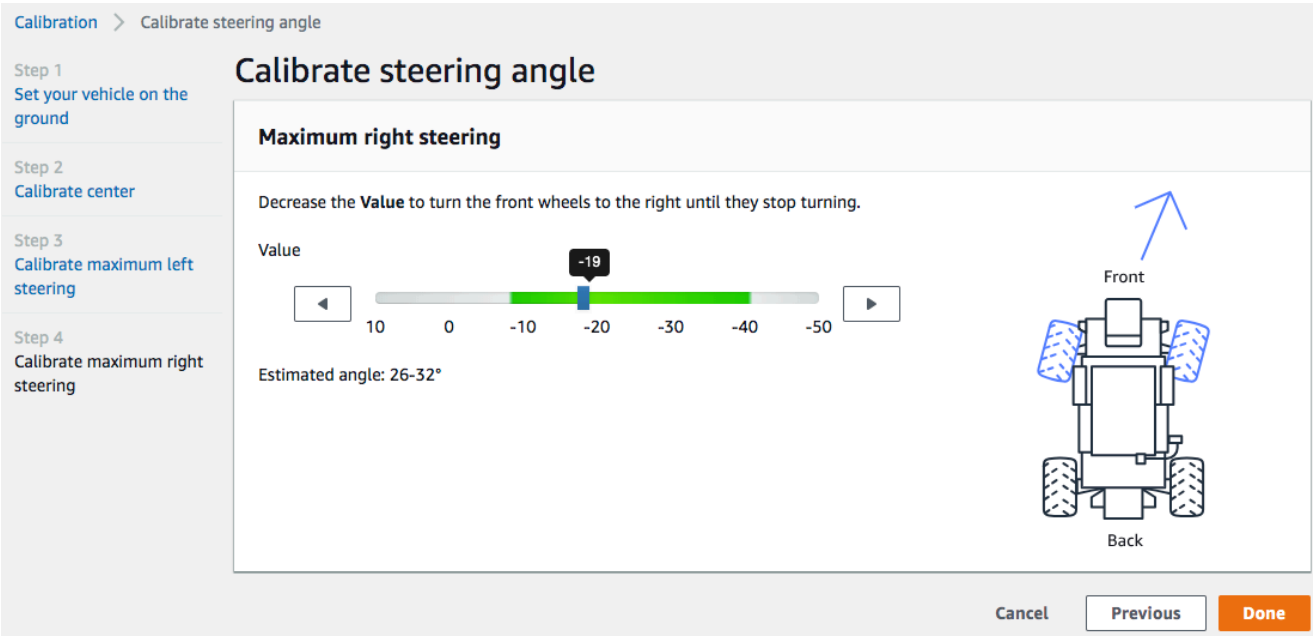
c. Bewegen Sie unter Maximum left steering (Maximale linker Lenkwinkel) den Schieberegler allmählich nach links oder drücken Sie den linken Pfeil, bis sich die Vorderräder des Fahrzeugs nicht mehr weiter nach links drehen. Es gibt ein leises Geräusch. Wenn Sie ein lautes Geräusch hören, sind Sie zu weit. Die Position entspricht dem maximalen linken Lenkwinkel. Wenn Sie Ihren Lenkwinkel im simulierten Aktionsraum begrenzt haben, passen Sie hier den entsprechenden Wert an. Wählen Sie Weiter.



Um den tatsächlichen maximalen linken Lenkwinkel zu ermitteln, zeichnen Sie eine Mittellinie für das Fahrzeug. Markieren Sie die beiden Randpunkte des zur Kalibrierung ausgewählten Vorderrads und ziehen Sie die Mittellinie dieses Vorderrads, bis diese die Mittellinie des Fahrzeugs schneidet. Verwenden Sie einen Winkelmesser, um den Winkel zu messen. Mehr dazu in der folgenden Abbildung. Wenn Sie in Ihrem Training den tatsächlichen Winkel verwenden möchten, können Sie in Ihrem nächsten Trainingsjob denselben Wert für den Aktionsradius festlegen.



d. Abbildung: Bestimmen Sie bei der Kalibrierung für das Fahrzeug einen maximalen rechten Lenkwinkel. Es gibt ein leises Geräusch. Wenn Sie ein lautes Geräusch hören, sind Sie zu weit. Die Position entspricht dem maximalen rechten Lenkwinkel. Wenn Sie Ihren Lenkwinkel im simulierten Aktionsraum begrenzt haben, passen Sie hier den entsprechenden Wert an. Wählen Sie Erledigt aus.

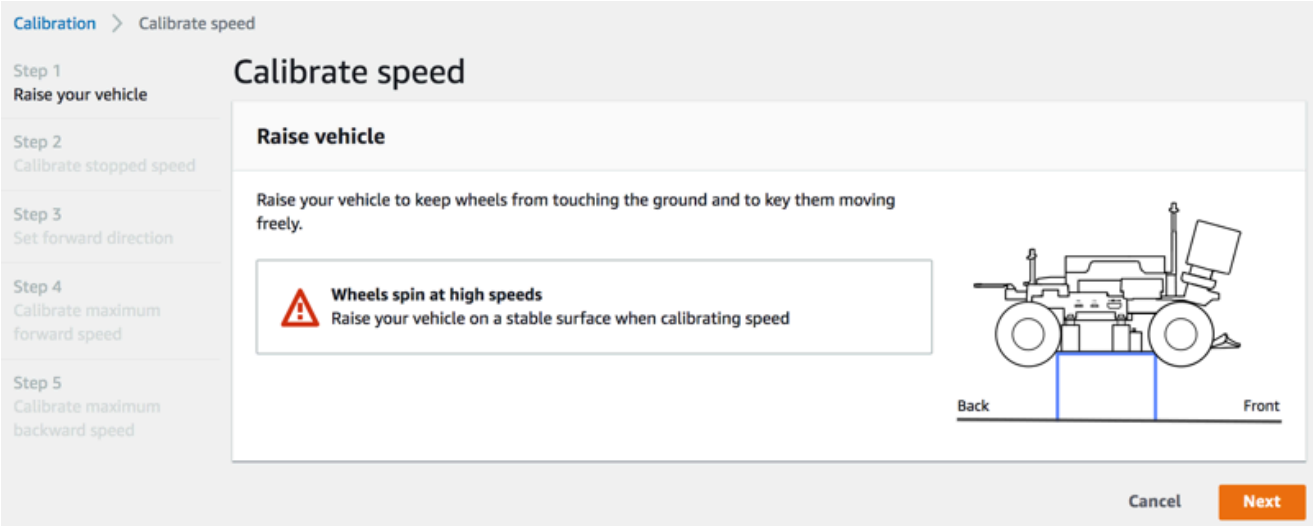


Um den tatsächlichen maximalen rechten Lenkwinkel zu bestimmen, folgen Sie den gleichen Schritten wie bei der Messung des maximalen linken Lenkwinkels.

Damit ist die Lenkkalibrierung für Ihr DeepRacer AWS-Fahrzeug abgeschlossen.

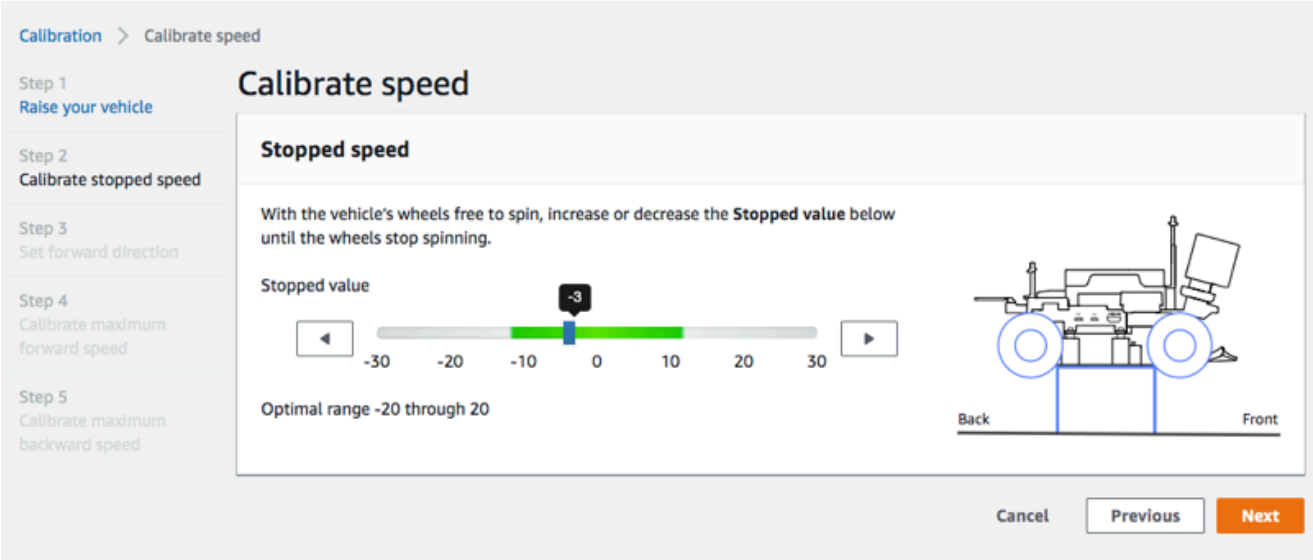
4. Um die Höchstgeschwindigkeit des Fahrzeugs zu kalibrieren, wählen Sie Calibrate (Kalibrieren) in Speed (Geschwindigkeit) auf der Seite Calibration (Kalibrierung) und führen Sie dann die folgenden Schritte aus.

   a. Heben Sie das Fahrzeug so an, dass sich die Räder frei drehen können. Wählen Sie in der Gerätesteuerkonsole Next (Weiter) aus.



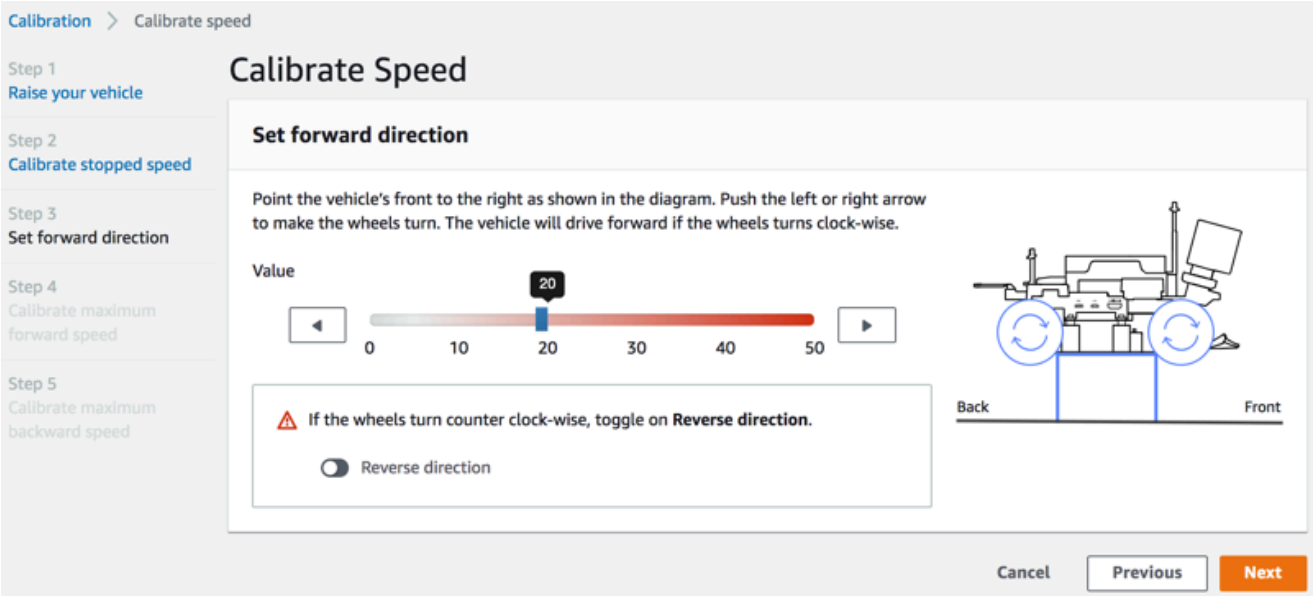
> **Note**
>
> Wenn die Geschwindigkeit des Fahrzeugs zu hoch eingestellt wurde, kann es während der Kalibrierung zu schnell fahren und Schäden an der Umgebung, dem Fahrzeug oder Personen in der Nähe verursachen. Sie sollten das Fahrzeug, wie hier beschrieben, anheben. Sie sollten es jedoch nicht in den Händen halten.

   b. Um die Geschwindigkeit im Stillstand zu kalibrieren, drücken Sie den Pfeil nach links oder rechts, um in der Gerätesteuerungskonsole Stopped value (Gestoppter Wert) unter Stopped speed (Gestoppte Geschwindigkeit) zu ändern, bis sich die Räder nicht mehr drehen. Wählen Sie Weiter.



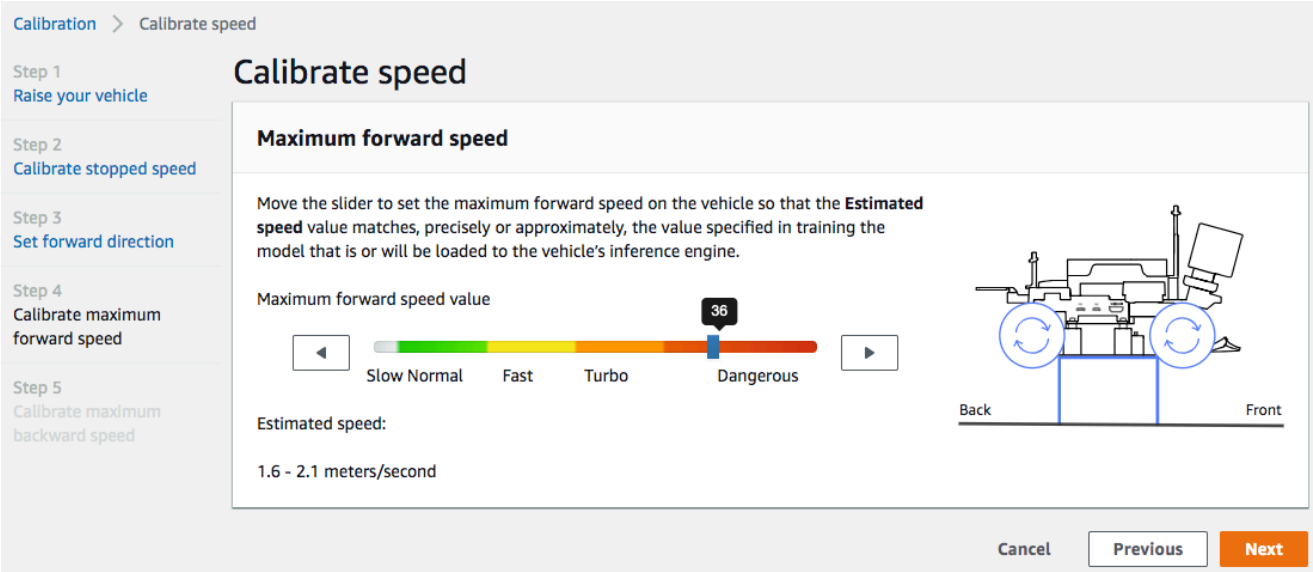
> **Note**
>
> Wenn Sie Sie Geräusche hören, während Sie den Stopped value (Gestoppten Wert) weiter nach links oder rechts verschieben, bewegen sich die Räder. Der ideale Nullpunkt liegt in der Mitte der beiden Werte. Wenn Sie beispielsweise bei 16 auf der linken Seite und bei -4 auf der rechten Seite ein Geräusch hören, ist der optimale Stoppwert 10.

c. Um die Vorwärtsrichtung des Fahrzeugs festzulegen, platzieren Sie das Fahrzeug wie auf der Seite und auf der Abbildung hier. Drücken Sie dann den Pfeil nach links oder rechts, damit sich die Räder drehen. Wenn sich die Räder im Uhrzeigersinn drehen, ist die Vorwärtsrichtung festgelegt. Wenn nicht, schalten Sie die Reverse direction (Richtungsumkehr) um. Wählen Sie Weiter.



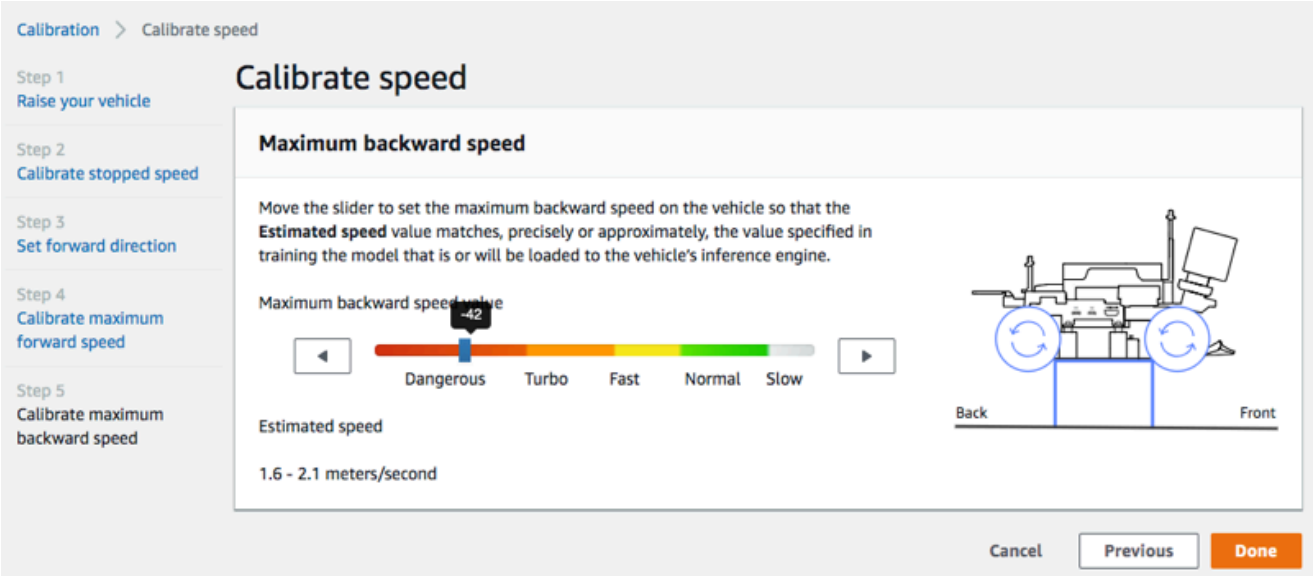
> **Note**
>
> Bei Fahrzeugen, die auf der AWS re:Invent 2018 verteilt wurden, ist möglicherweise die Vorwärtsrichtung rückwärts eingestellt. Achten Sie in diesem Fall darauf, dass Sie die Reverse direction (Richtungsumkehr) umschalten.

d. Um die maximale Vorwärtsgeschwindigkeit zu kalibrieren, bewegen Sie den Schieberegler unter Maximum forward speed (Maximale Vorwärtsgeschwindigkeit) vorsichtig nach links oder rechts, um den Maximum forward speed value (Wert für maximale Vorwärtsgeschwindigkeit) schrittweise auf einen so positiven Wert zu justieren, dass der Estimated speed (Geschätzte Geschwindigkeit)-Wert gleich oder ähnlich der in der Simulation angegebenen maximalen Geschwindigkeit ist. Wählen Sie Weiter.



> **Note**
>
> Die tatsächliche Höchstgeschwindigkeit Ihres Fahrzeugs hängt von der Reibung der Fahrbahnoberfläche sowie dem Batteriezustand des Fahrzeugs ab. Für mehr Flexibilität können Sie die Drosselung des Fahrzeugs 20-30 Prozent über der für das Training in der Simulation festgelegten Höchstgeschwindigkeit festlegen. Generell sollten Sie den Wert für die Höchstgeschwindigkeit innerhalb des grünen Bereichs festlegen. Bei höheren Werten kann es vorkommen, dass Ihr Fahrzeug zu schnell fährt und ein erhöhtes Risiko für Beschädigungen besteht. Außerdem unterstützt der Aktionsraum für das Training keine Höchstgeschwindigkeit von mehr als 2 m/s.

e. Um die maximale Rückwärtsgeschwindigkeit zu kalibrieren, bewegen Sie den Schieberegler unter Maximum backward speed (Maximale Rückwärtsgeschwindigkeit) vorsichtig nach links oder rechts, um den Maximum backward speed value (Wert für maximale Rückwärtsgeschwindigkeit) schrittweise so auf einen negativen Wert zu justieren, dass der Wert Estimated speed (Geschätzte Geschwindigkeit) gleich oder ähnlich der in der Simulation angegebenen maximalen Geschwindigkeit ist. Wählen Sie Erledigt aus.



> **Note**
>
> Das DeepRacer AWS-Fahrzeug verwendet im autonomen Fahrmodus keine Rückwärtsgeschwindigkeit. Sie können die Rückwärtsgeschwindigkeit auf einen beliebigen Wert festlegen, mit dem Sie das Fahrzeug im manuellen Fahrmodus steuern können.

Damit ist die Kalibrierung der Höchstgeschwindigkeit Ihres DeepRacer AWS-Fahrzeugs abgeschlossen.

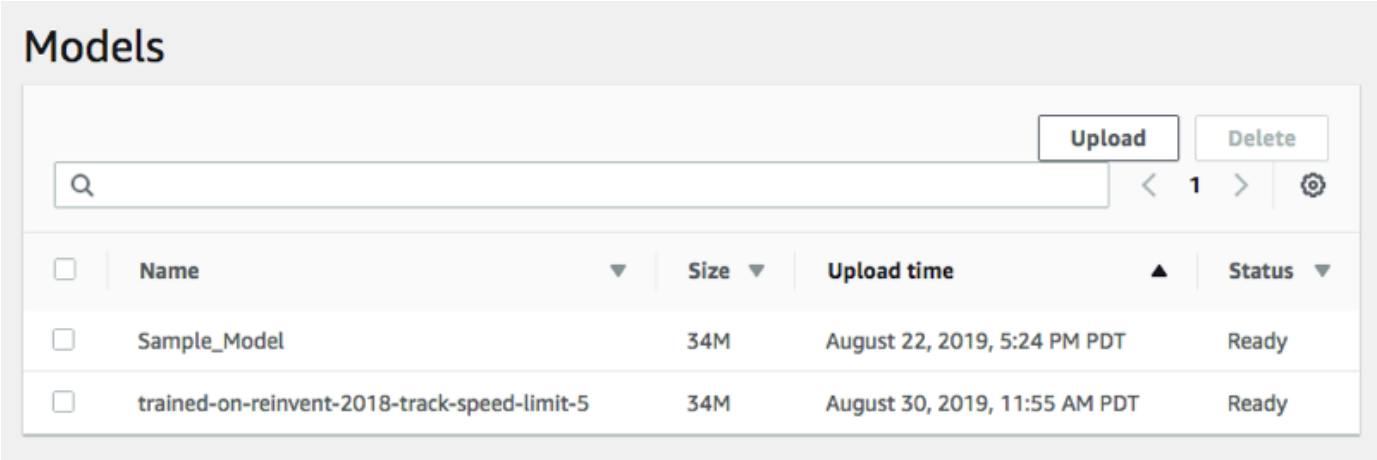
Laden Sie ein Modell auf Ihr DeepRacer AWS-Fahrzeug hoch

Um Ihr DeepRacer AWS-Fahrzeug beim autonomen Fahren zu starten, müssen Sie mindestens ein DeepRacer AWS-Modell auf Ihr DeepRacer AWS-Fahrzeug hochgeladen haben.

Um ein Modell hochzuladen, müssen Sie das Modell trainiert und bewertet haben. Sie können das Modell mit der DeepRacer AWS-Konsole trainieren. Danach müssen Sie die Modellartefakte aus dem Amazon S3 S3-Speicher auf ein (lokales oder Netzwerk-) Laufwerk herunterladen, auf das Ihr Computer zugreifen kann.

So laden Sie ein trainiertes Modell auf Ihr Fahrzeug hoch:

1. Wählen Sie im Hauptnavigationsbereich der Gerätekonsole Models (Modelle) aus.



2. Wählen Sie auf der Seite Modelle oberhalb der Modellliste die Option Hochladen aus.
3. Navigieren Sie in der Dateiauswahl zu dem Laufwerk oder der Freigabe, auf das/die Sie Ihre Modellartefakte heruntergeladen haben, und wählen Sie die komprimierte Modelldatei (mit der Erweiterung `*.tar.gz`) aus, die hochgeladen werden soll.

   Nur ein erfolgreich hochgeladenes Modell wird der Liste Models (Modelle) hinzugefügt und kann von Ihnen im autonomen Fahrmodus in die Inferenz-Engine des Fahrzeugs geladen werden. Anweisungen zum Laden eines Modells in die Inferenz-Engine Ihres Fahrzeugs finden Sie unter Fahren Sie Ihr DeepRacer AWS-Fahrzeug autonom .

Fahren Sie Ihr DeepRacer AWS-Fahrzeug

Nachdem Sie Ihr DeepRacer AWS-Fahrzeug eingerichtet haben, können Sie beginnen, Ihr Fahrzeug manuell zu fahren oder es mithilfe der Gerätekonsole des Fahrzeugs autonom fahren zu lassen.

Für autonomes Fahren müssen Sie ein DeepRacer AWS-Modell trainiert haben und die trainierten Modellartefakte im Fahrzeug einsetzen lassen. Im autonomen Rennmodus steuert das im Inferenzmodul laufende Modell die Fahrtrichtungen und die Geschwindigkeit des Fahrzeugs. Ohne dass ein trainiertes Modell in das Fahrzeug heruntergeladen wurde, können Sie die Gerätekonsole des Fahrzeugs verwenden, um das Fahrzeug manuell zu steuern.

Die Leistung des Fahrzeugs beim autonomen Fahren wird durch viele Faktoren beeinflusst. Dazu gehören das trainierte Modell, die Fahrzeugkalibrierung, die Streckenbedingungen wie z. B. die Oberflächenreibung, Farbkontraste und Lichtreflexionen u.v.m. Damit Ihr Fahrzeug eine optimale Leistung erreicht, müssen Sie sicherstellen, dass die Abbildung des Modells aus der Simulation auf die echte Welt präzise, relevant und aussagekräftig ist. Weitere Informationen finden Sie unter the section called “Optimieren Sie das Training für reale Umgebungen”.

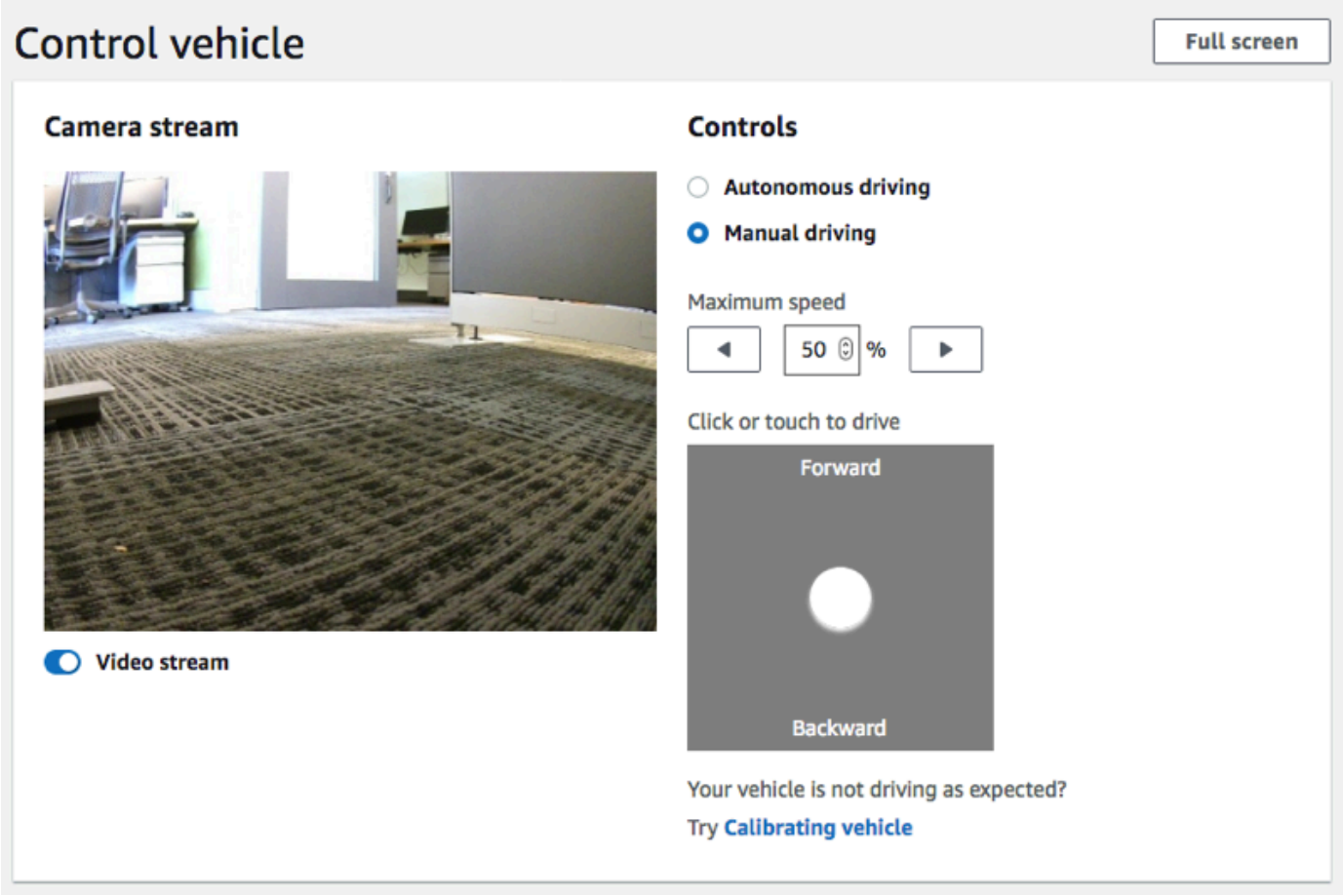
Fahren Sie Ihr DeepRacer AWS-Fahrzeug manuell

Wenn Sie kein Modell trainiert oder kein trainiertes Modell für Ihr DeepRacer AWS-Fahrzeug bereitgestellt haben, können Sie es nicht selbst fahren lassen. Sie können es aber dennoch manuell fahren.

Gehen Sie wie folgt vor, um ein DeepRacer AWS-Fahrzeug manuell zu fahren.

Um Ihr DeepRacer AWS-Fahrzeug manuell zu fahren

1. Wenn Ihr DeepRacer AWS-Fahrzeug mit dem Wi-Fi-Netzwerk verbunden ist, folgen Sie den Anweisungen, um sich an der Gerätesteuerkonsole des Fahrzeugs anzumelden.
2. Wählen Sie auf der Seite „Fahrzeug steuern“ unter Steuerung die Option Manuelles Fahren aus.



3. Klicken oder tippen Sie auf Click or touch to drive (Zum Fahren klicken oder tippen) und klicken oder tippen Sie auf eine Position innerhalb des Fahrpads, um das Fahrzeug zu fahren. Bilder, die von der Frontkamera des Fahrzeugs aufgenommen wurden, werden im Videoplayer unter Camera stream (Kamera-Stream) angezeigt.

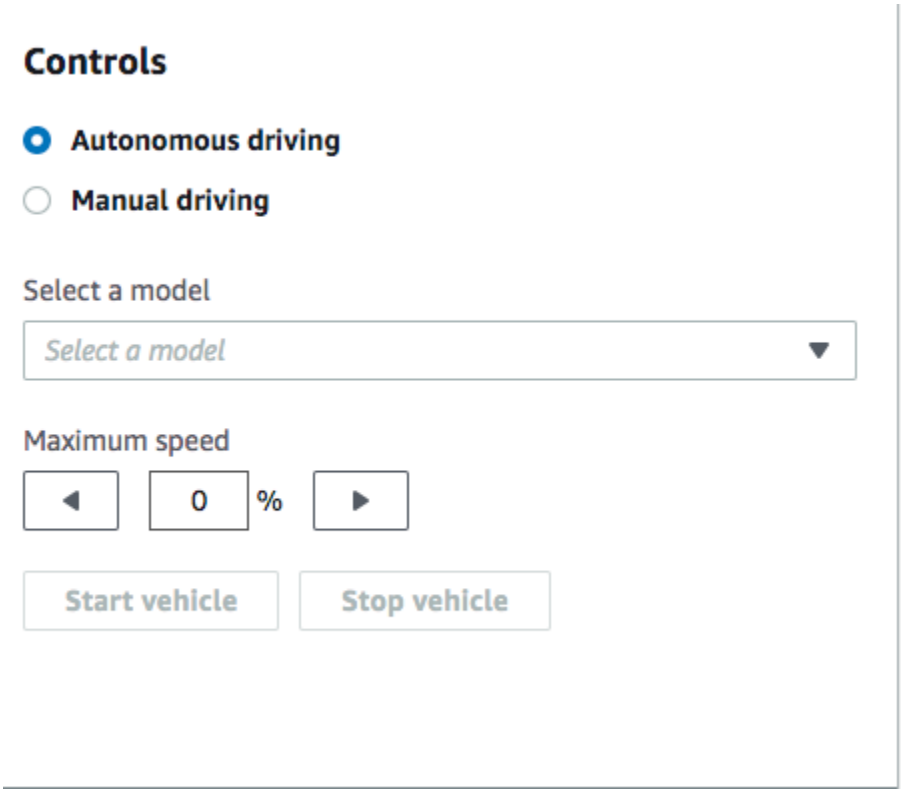
4. Um den Videostream auf der Gerätekonsole zu aktivieren oder zu deaktivieren, während Sie das Fahrzeug fahren, schalten Sie die Option Video Stream unter der Anzeige Camera stream (Kamera-Stream)um.
5. Wiederholen Sie den Vorgang ab Schritt 3, um das Fahrzeug an verschiedene Positionen zu fahren.

Fahren Sie Ihr DeepRacer AWS-Fahrzeug autonom

Um mit dem autonomen Fahren zu beginnen, stellen Sie das Fahrzeug auf eine physische Strecke und gehen Sie wie folgt vor:

Um Ihr DeepRacer AWS-Fahrzeug autonom zu fahren

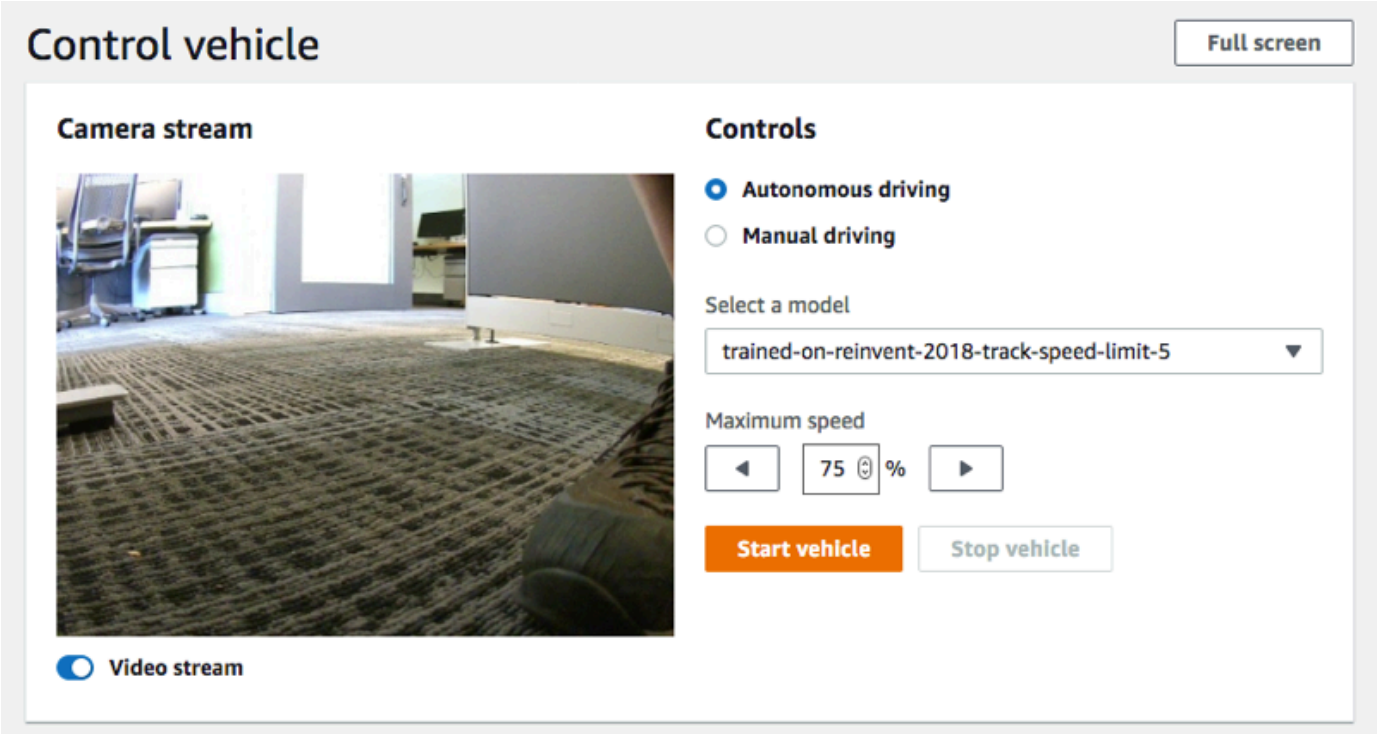
1. Befolgen Sie die Anweisungen, um sich an der Gerätekonsole des Fahrzeugs anzumelden. Führen Sie dann die folgenden Schritte für autonomes Fahren durch:
2. Wählen Sie auf der Seite Fahrzeug steuern unter Steuerung die Option Autonomes Fahren aus.



3. Wählen Sie aus der Dropdown-Liste Select a model (Modell auswählen) ein hochgeladenes Modell aus. Wählen Sie dann Load model (Modell laden) aus. Dadurch wird das Modell in die Inferenz-Engine geladen. Der Vorgang dauert etwa 10 Sekunden.

4. Legen Sie die Einstellung Maximum speed (Maximalgeschwindigkeit) des Fahrzeugs auf einen Prozentsatz der beim Training des Modells verwendeten Maximalgeschwindigkeit fest.

   Bestimmte Faktoren, wie z. B. die Oberflächenreibung der realen Strecke, können die Höchstgeschwindigkeit des Fahrzeugs gegenüber der im Training verwendeten Höchstgeschwindigkeit reduzieren. Sie müssen experimentieren, um die optimale Einstellung zu finden.



5. Wählen Sie Start vehicle (Fahrzeug starten) aus, um das Fahrzeug für das autonome Fahren einzustellen.
6. Um den Videostream auf der Gerätekonsole zu aktivieren oder zu deaktivieren, während Sie das Fahrzeug fahren, schalten Sie die Option Video Stream unter der Anzeige Camera stream (Kamera-Stream)um.
7. Beobachten Sie die Fahrt des Fahrzeugs auf der physischen Strecke oder im Streaming-Videoplayer in der Gerätekonsole.
8. Um das Fahrzeug anzuhalten, wählen Sie Stop vehicle (Fahrzeug anhalten) aus.

Wiederholen Sie die Schritte ab Schritt 3 für einen weiteren Lauf mit demselben oder einem anderen Modell.

Überprüfen und verwalten Sie Ihre DeepRacer AWS-Fahrzeugeinstellungen

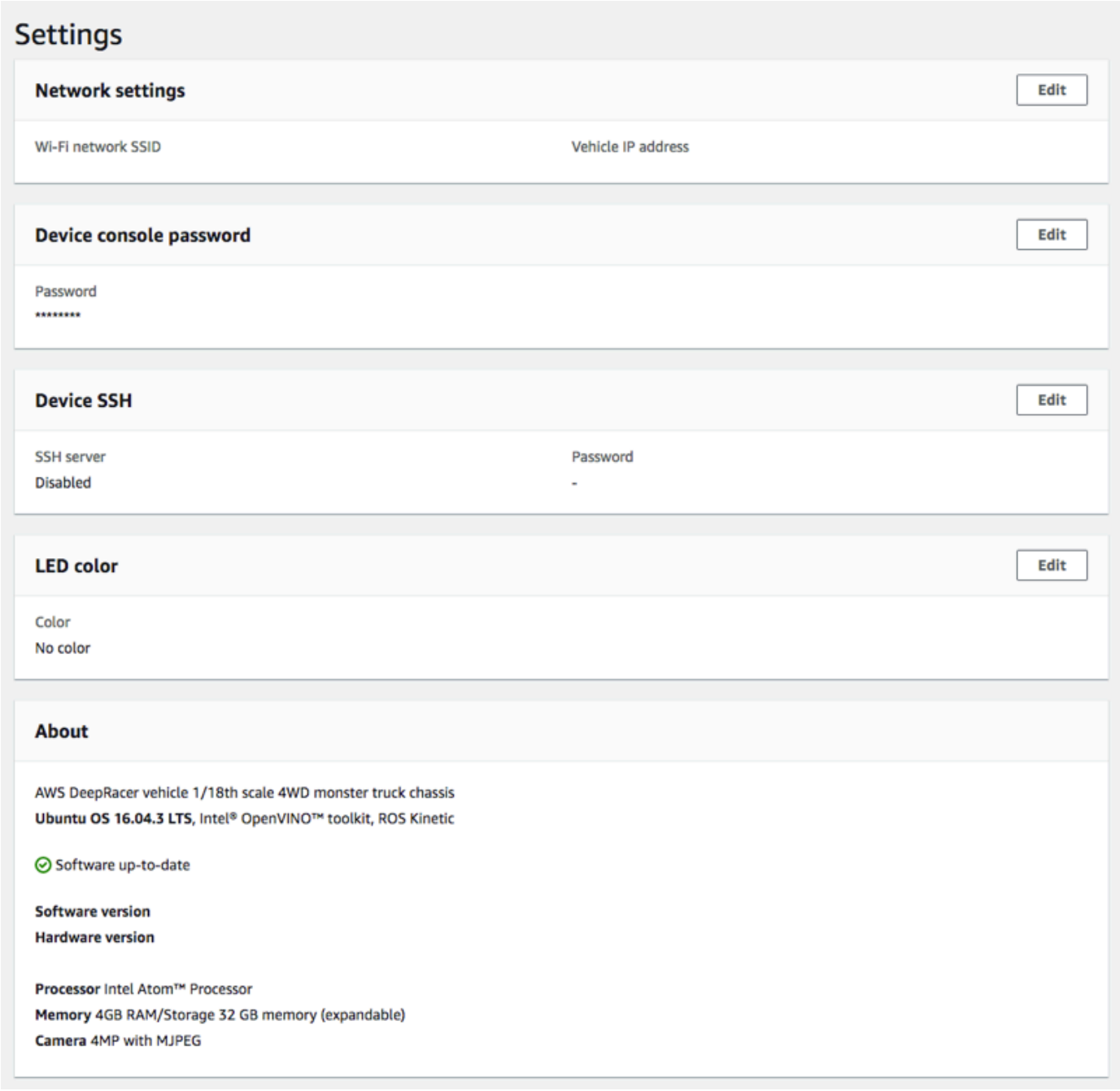
Nach der Ersteinrichtung können Sie die DeepRacer AWS-Gerätesteuerungskonsole verwenden, um die Einstellungen Ihres Fahrzeugs zu verwalten. Die Aufgaben umfassen folgende Schritte:

- Auswahl eines anderen WLANs
- Zurücksetzen des Passworts für die Gerätekonsole
- Aktivieren oder Deaktivieren der Geräte-SSH-Einstellungen
- Konfigurieren der LED-Farbe des Rücklichts des Fahrzeugs
- Überprüfen der Gerätesoftware- und -hardwareversionen
- Überprüfen des Batteriestands des Fahrzeugs.

Im Folgenden werden Sie durch diese Aufgaben geführt.

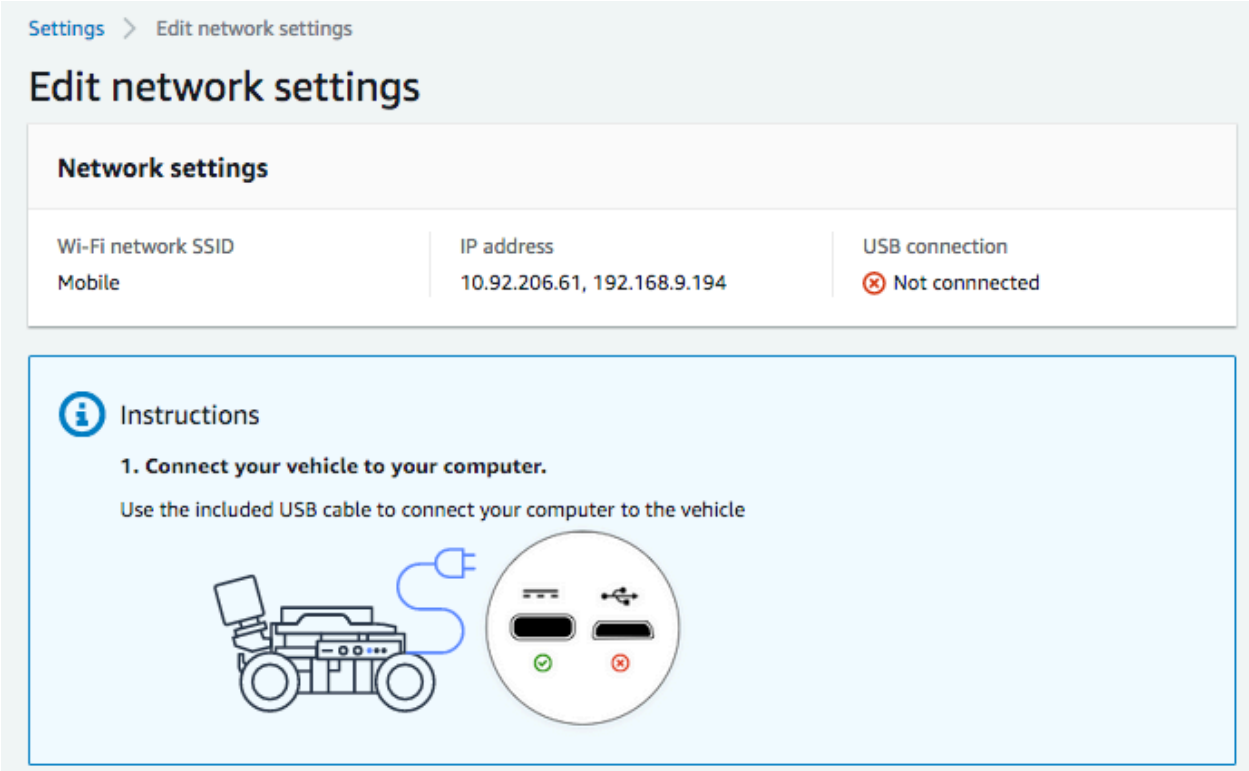
So überprüfen und verwalten Sie die Einstellungen Ihres Fahrzeugs

1. Wenn Ihr DeepRacer AWS-Fahrzeug mit dem Wi-Fi-Netzwerk verbunden ist, folgen Sie den Anweisungen, um sich an der Gerätesteuerkonsole des Fahrzeugs anzumelden.
2. Wählen Sie Settings (Einstellungen) im Hauptnavigationsbereich aus.
3. Führen Sie auf der Seite Settings (Einstellungen) eine oder mehrere der folgenden Aufgaben Ihrer Wahl aus.



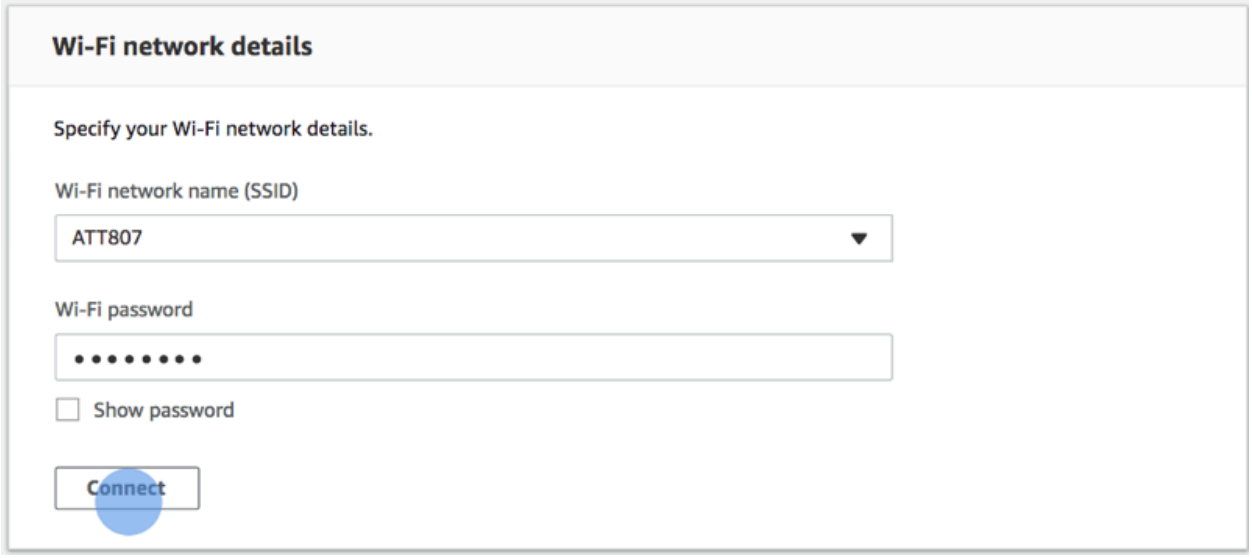
a. Um ein anderes Wi-Fi-Netzwerk auszuwählen, wählen Sie Bearbeiten für Netzwerkeinstellungen und gehen Sie dann wie folgt vor.

   i. Folgen Sie den Anweisungen unter Netzwerkeinstellungen bearbeiten, um Ihr Fahrzeug über das USB-to-USB-C Kabel mit Ihrem Computer zu verbinden. Nachdem sich der USB connection (USB-Verbindung)-Status auf Connected (Verbunden) geändert hat, wählen Sie die Schaltfläche Go to deepracer.aws (Zu deepracer.aws wechseln) aus, um die Anmeldeseite der Gerätekonsole zu öffnen.



ii. Geben Sie auf der Anmeldeseite der Gerätekonsole das auf der Unterseite Ihres Fahrzeugs aufgedruckte Passwort ein und wählen Sie dann Access vehicle (Auf Fahrzeug zugreifen) aus.

iii. Wählen Sie unter Wi-Fi network details (WLAN-Details) ein WLAN aus der Dropdown-Liste aus, geben Sie das Passwort des ausgewählten Netzwerks ein und wählen Sie dann Connect (Verbinden) aus.



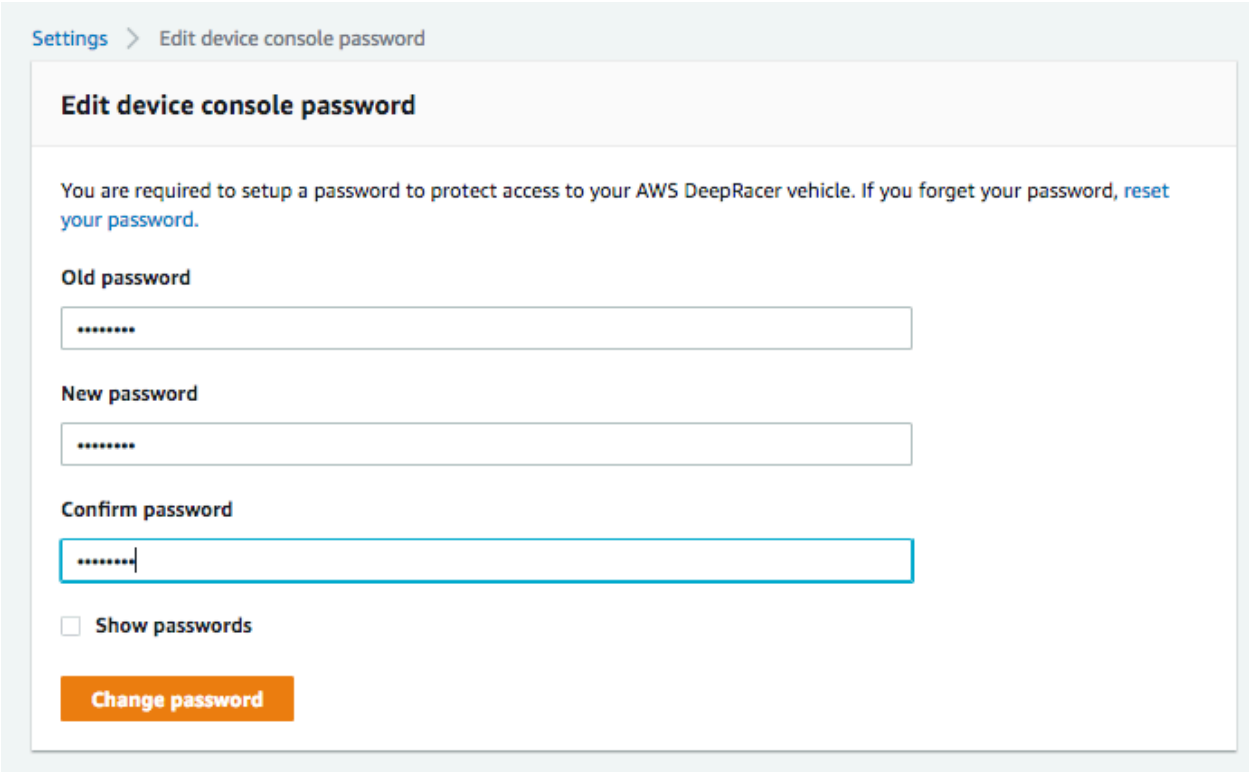
iv. Nachdem der Vehicle status (Fahrzeugstatus) der WLAN-Verbindung zu Connecte (Verbunden) wechselt, wählen Sie Next (Weiter) aus, um zur Seite Settings (Einstellungen) der Gerätekonsole zurückzukehren. Dort sehen Sie die neue IP-Adresse des Fahrzeugs.

b. Um das Passwort für die Anmeldung an der Gerätekonsole zurückzusetzen, wählen Sie Bearbeiten als Passwort für die Gerätekonsole und gehen Sie dann wie folgt vor.

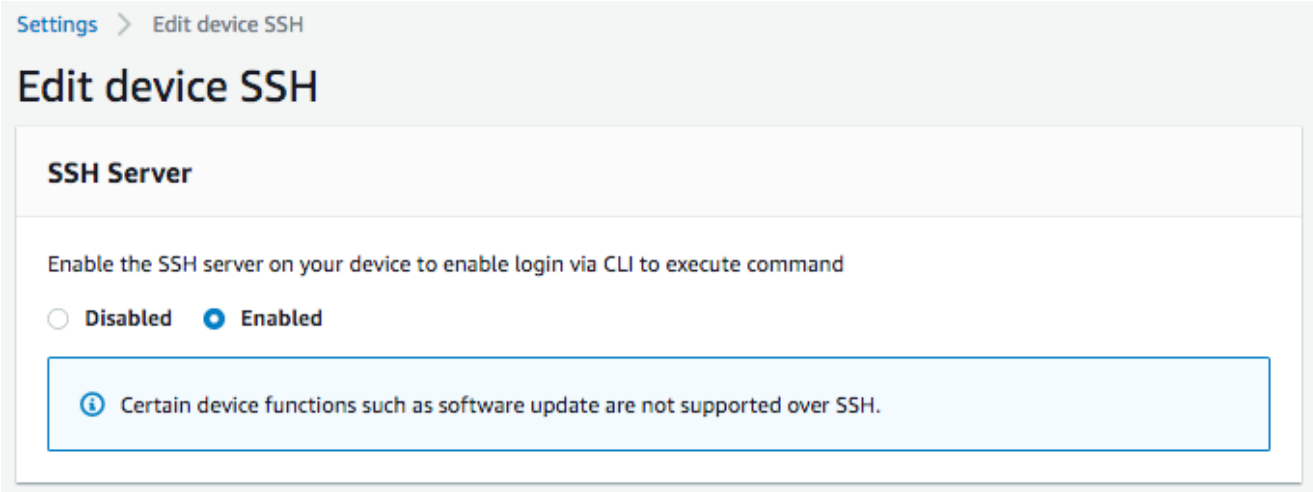
i. Geben Sie auf der Seite Edit device console password (Gerätekonsolenpasswort bearbeiten) unter New password (Neues Passwort) ein neues Passwort ein.

ii. Geben Sie das neue Passwort in Confirm password (Passwort bestätigen) erneut ein, um die Änderung zu bestätigen. Sie können erst fortfahren, wenn die Passwörter identisch sind.

iii. Wählen Sie Change password (Passwort ändern= aus, um die Aufgabe abzuschließen. Diese Option ist nur aktiviert, wenn Sie in den obigen Schritten ein gültiges Passwort eingegeben und bestätigt haben.

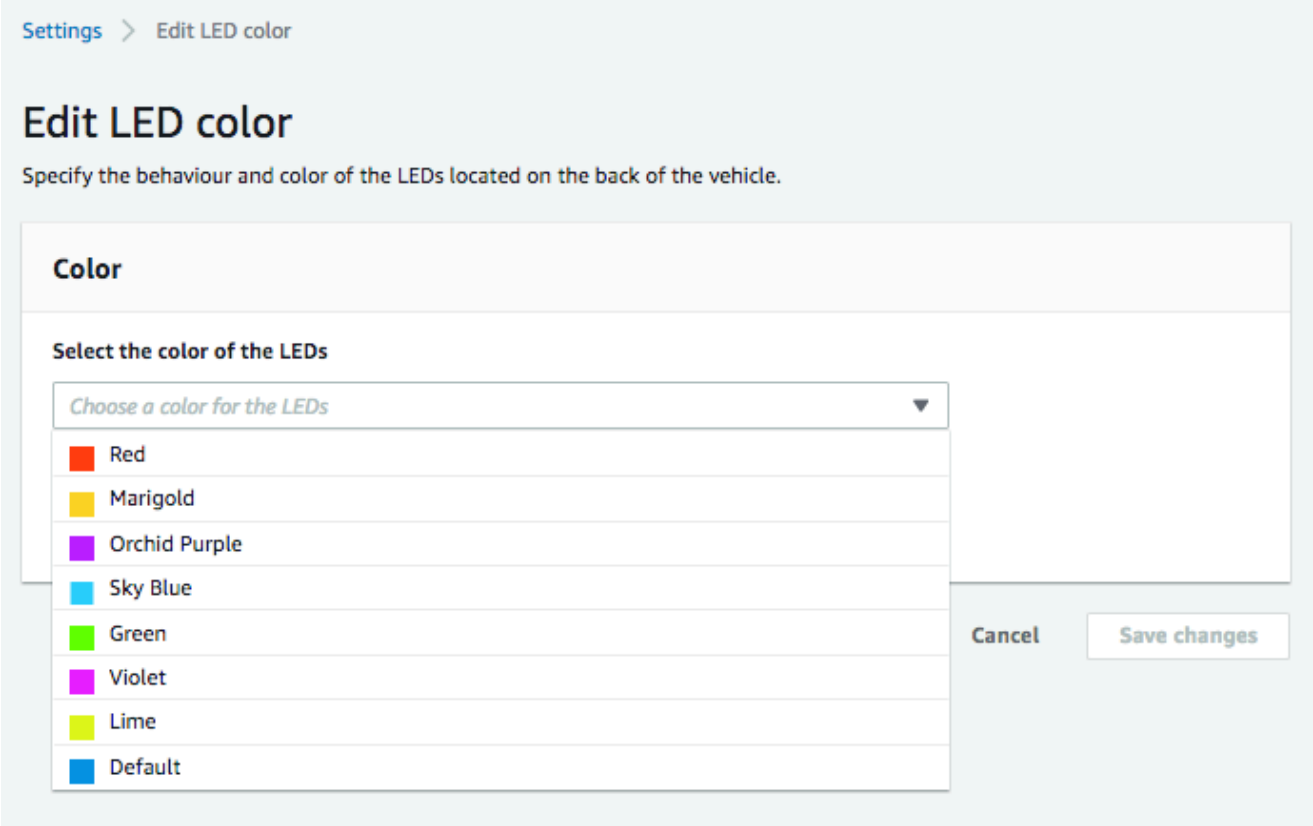


c. Um die SSH-Verbindung zum Fahrzeug zu aktivieren oder zu deaktivieren, wählen Sie Bearbeiten für Geräte-SSH und dann Aktivieren oder Deaktivieren.



4. Um die LED-Farbe des Rücklichts des Fahrzeugs zu ändern, um Ihr Fahrzeug auf einer Strecke zu unterscheiden, wählen Sie Edit (Bearbeiten) für LED color (LED-Farbe) auf der Seite Settings (Einstellungen) aus und gehen Sie wie folgt vor.

   a. Wählen Sie auf der Seite LED-Farbe bearbeiten aus der LEDs Dropdownliste „Farbe der Farbe auswählen“ eine verfügbare Farbe aus.



   Sie sollten eine Farbe auswählen, mit der Sie Ihr Fahrzeug von anderen Fahrzeugen auf der Strecke unterscheiden können.

b. Wählen Sie Save changes (Änderungen speichern) aus, um die Aufgabe abzuschließen.

   Die Funktionalität Save changes (Änderungen speichern) ist erst aktiviert, wenn Sie eine Farbe ausgewählt haben.

5. Um die Gerätesoftware- und -hardwareversionen zu überprüfen und die System- und Kamerakonfigurationen zu ermitteln, lesen Sie den Abschnitt About (Informationen) unter Settings (Einstellungen).
6. Überprüfen Sie den unteren Teil des primären Navigationsbereichs, um den Ladezustand der Fahrzeugbatterie zu überprüfen.

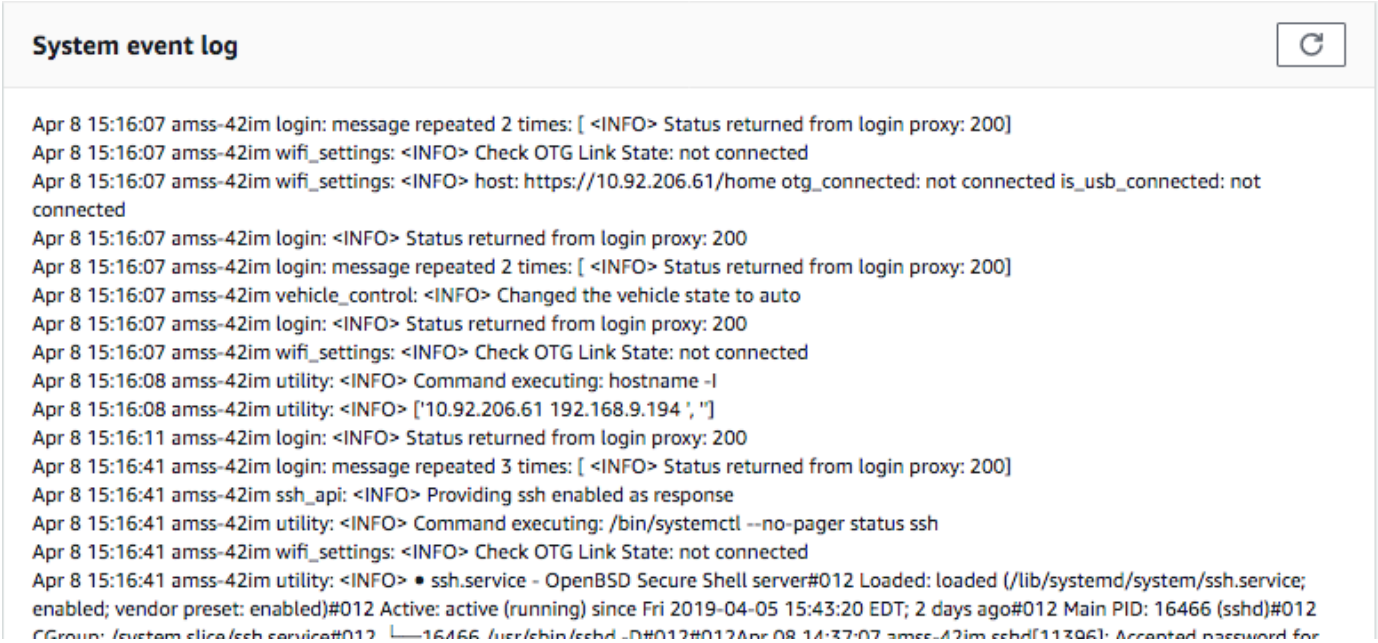
Ihre DeepRacer AWS-Fahrzeugprotokolle anzeigen

Ihr DeepRacer AWS-Fahrzeug protokolliert Betriebsereignisse, die bei der Behebung von Problemen beim Betrieb Ihres Fahrzeugs hilfreich sein können. Es gibt zwei Arten von DeepRacer AWS-Fahrzeugprotokollen:

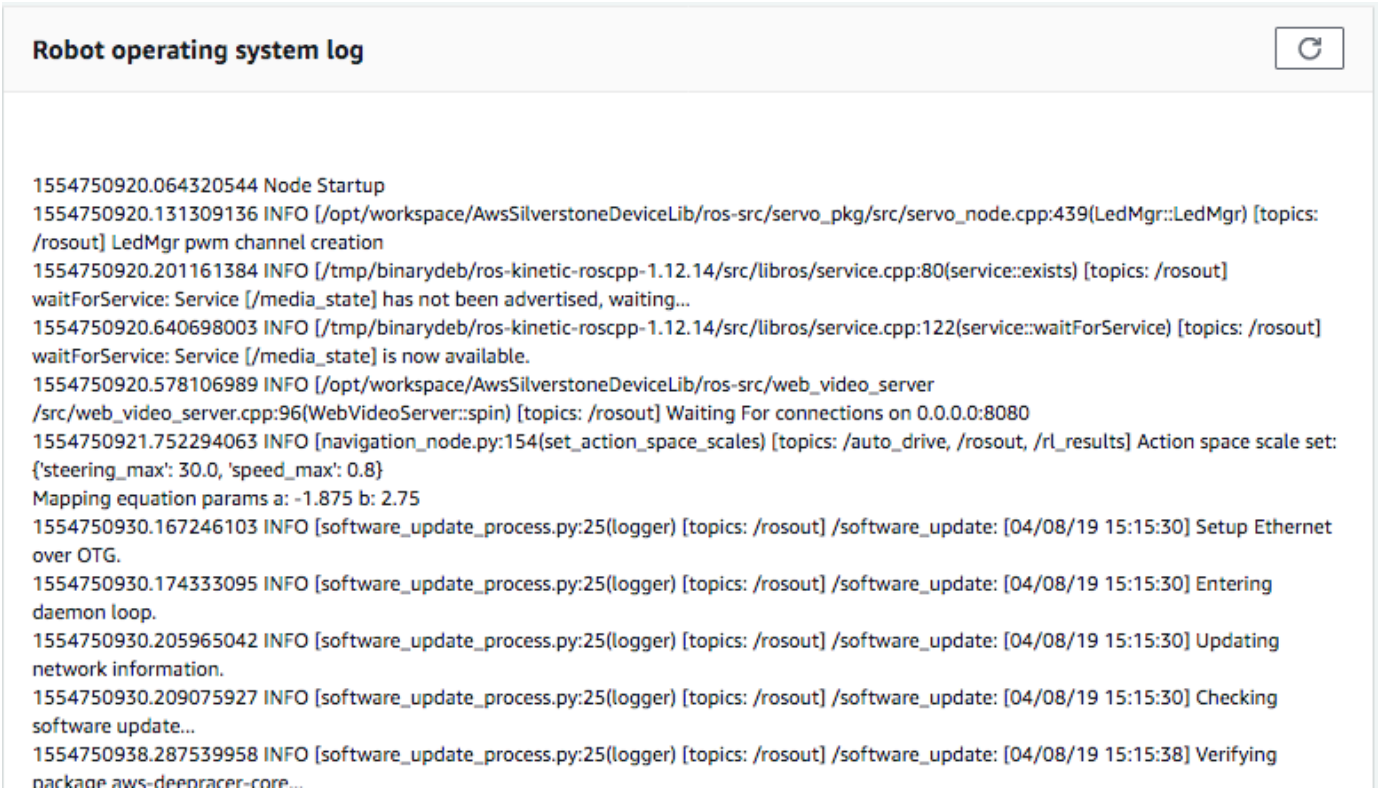
- Das Systemereignisprotokoll verfolgt die Vorgänge, die im Computerbetriebssystem des Fahrzeugs stattfinden, wie z. B. Prozessmanagement, WLAN-Verbindung oder Passwortrücksetzung.
- Das Robot-Betriebssystemprotokoll protokolliert die Zustände von Vorgängen, die im Betriebssystemknoten für Robot-Operationen des Fahrzeugs stattfinden (einschließlich Fahrzeugführung, Video-Streaming und Strategieinferenzierung).

Führen Sie die folgenden Schritte aus, um die Geräteprotokolle anzuzeigen.

1. Wenn Ihr DeepRacer AWS-Fahrzeug mit dem Wi-Fi-Netzwerk verbunden ist, folgen Sie den Anweisungen, um sich an der Gerätesteuerkonsole des Fahrzeugs anzumelden.
2. Wählen Sie im Hauptnavigationsbereich der Gerätekonsole die Option Protokolle (Logs) aus.
3. Um die Systemereignisse anzuzeigen, scrollen Sie in der Ereignisliste unter System event log (Systemereignisprotokoll) nach unten.



4. Um die Robot-Betriebssystemereignisse anzuzeigen, scrollen Sie in der Ereignisliste unter Robot operating system log (Robot-Betriebssystemprotokoll) nach unten.



Aktualisieren und Wiederherstellen Ihres DeepRacer AWS-Geräts

Aktualisieren Sie Ihr DeepRacer AWS-Gerät auf den neuesten Software-Stack, einschließlich Ubuntu 20.04 Focal Fossa, Intel® OpenVINO™ Toolkit 2021.1.110, Foxy Fitzroy und Python 3.8. ROS2 Dieses Update ist für die Ausführung von DeepRacer AWS-Open-Source-Projekten erforderlich, ansonsten jedoch optional. AWS unterstützt DeepRacer nur Ubuntu 20.04 Focal Fossa und ROS2 Foxy Fitzroy.

> **Important**
>
> Durch die Aktualisierung auf den neuen DeepRacer AWS-Softwarestack werden alle Daten auf Ihrem DeepRacer AWS-Gerät gelöscht.

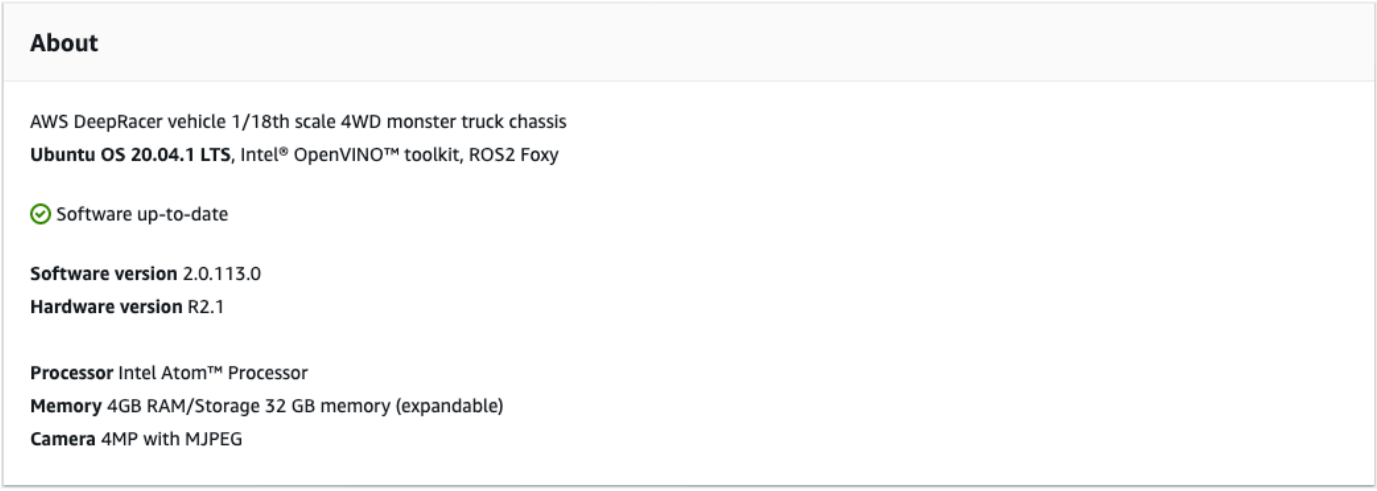
Themen

- Prüfen Sie, welche Softwareversion auf Ihrem DeepRacer AWS-Gerät derzeit ausgeführt wird
- Bereiten Sie sich darauf vor, Ihr DeepRacer AWS-Gerät auf den Ubuntu 20.04-Softwarestack zu aktualisieren
- Aktualisieren Sie Ihr DeepRacer AWS-Gerät auf den Ubuntu 20.04-Softwarestack

Prüfen Sie, welche Softwareversion auf Ihrem DeepRacer AWS-Gerät derzeit ausgeführt wird

Um zu überprüfen, welche Softwareversion auf Ihrem DeepRacer AWS-Gerät derzeit ausgeführt wird

1. Melden Sie sich bei der DeepRacer AWS-Gerätekonsole an. Um zu erfahren, wie das geht, folgen Sie den Schritten unterthe section called “Starten Sie die Gerätekonsole”.
2. Wählen Sie im Navigationsbereich Einstellungen aus.
3. Überprüfen Sie im Abschnitt Über uns, welche Softwareversion auf Ihrem DeepRacer AWS-Fahrzeug derzeit ausgeführt wird.



Bereiten Sie sich darauf vor, Ihr DeepRacer AWS-Gerät auf den Ubuntu 20.04-Softwarestack zu aktualisieren

Dieses Thema führt Sie durch den Prozess zur Erstellung der AWS DeepRacer Ubuntu-Installationsmedien. Für die Vorbereitung des bootfähigen USB-Laufwerks ist zusätzliche Hardware erforderlich.

Voraussetzungen

Bevor Sie beginnen, stellen Sie sicher, dass Sie die folgenden Dinge parat haben:

- Ein DeepRacer AWS-Gerät
- Ein USB-Flash-Laufwerk (32 GB oder größer)
- Ein benutzerdefiniertes AWS DeepRacer Ubuntu-ISO-Image.
- Das neueste DeepRacer AWS-Softwareupdatepaket.
- Eine Kopie von, die mit Ihrem Betriebssystem UNetbootinkompatibel ist.
- Ein Computer, auf dem Ubuntu, Windows oder macOS ausgeführt wird, um das USB-Installationsmedium vorzubereiten. Sie können das Rechenmodul auf Ihrem DeepRacer AWS-Gerät auch als Linux-Computer verwenden, indem Sie Maus, Tastatur und Monitor mit einem HDMI-Typ-A-Kabel verbinden.

Vorbereitung

Um die DeepRacer AWS-Aktualisierungsmedien vorzubereiten, führen Sie die folgenden Aufgaben aus:

- Formatieren Sie das USB-Laufwerk in die folgenden zwei Partitionen:
  - Eine FAT32 4-GB-Boot-Partition
  - Eine NTFS-Datenpartition von mindestens 18 GB
- Machen Sie das USB-Laufwerk bootfähig, um das Update beim Neustart zu starten:
  - Brennen Sie das erforderliche benutzerdefinierte Ubuntu-ISO-Image auf die Startpartition
  - Kopieren Sie die erforderlichen Aktualisierungsdateien auf die Datenpartition des USB-Laufwerks

Bereiten Sie ein bootfähiges USB-Laufwerk vor

Folgen Sie diesen Anweisungen, um Ihre DeepRacer AWS-Aktualisierungsmedien auf Ubuntu (Linux), Windows oder macOS vorzubereiten. Je nachdem, welchen Computer Sie verwenden, können sich die einzelnen Aufgaben von Betriebssystem zu Betriebssystem unterscheiden. Wählen Sie die Registerkarte, die Ihrem Betriebssystem entspricht.

Ubuntu

Folgen Sie den Anweisungen hier, um einen Ubuntu-Computer, einschließlich des Rechenmoduls Ihres DeepRacer AWS-Geräts, zur Vorbereitung der Aktualisierungsmedien für Ihr DeepRacer AWS-Gerät zu verwenden. Wenn Sie eine andere Linux-Distribution verwenden, ersetzen Sie die `apt-get *` Befehle durch Befehle, die mit dem Paketmanager Ihres Betriebssystems kompatibel sind.

Um das USB-Laufwerk zu löschen und zu partitionieren

1. Führen Sie zur Installation und zum Starten die folgenden Befehle aus GParted.

```
sudo apt-get update; sudo apt-get install gparted
sudo gparted
```

2. Um Ihr USB-Laufwerk zu löschen, benötigen Sie seinen Gerätepfad. Gehen Sie wie folgt vor, um es auf der GParted Konsole zu finden und das USB-Laufwerk zu löschen:

a. Wählen Sie in der Menüleiste „Ansicht“ und anschließend „Geräteinformationen“. Eine Seitenleiste mit Modell, Größe und Pfad der ausgewählten Festplatte wird angezeigt.

b. Wählen Sie Ihr USB-Laufwerk aus, indem Sie in der Menüleiste GPartedauf Geräte klicken und schließlich Ihr USB-Laufwerk aus der Liste auswählen. Passen Sie die in der Gerätebeschreibung angegebene Größe und das Modell an Ihr USB-Laufwerk an.

c. Wenn Sie sicher sind, dass Sie die richtige Festplatte ausgewählt haben, löschen Sie alle vorhandenen Partitionen.

   Wenn die Partitionen gesperrt sind, öffnen Sie das Kontextmenü (rechte Maustaste) und wählen Sie unmount (Mounting aufheben).

3. Um die FAT32 Startpartition mit einer Kapazität von 4 GB zu erstellen, wählen Sie das Dateisymbol oben links aus, legen Sie die folgenden Parameter fest und wählen Sie Hinzufügen.

   Free space preceding (Freier Platz vorher): **`1`**

   New size (Neue Größe):**`4096`**

   Free space following (Freier Platz danach): **`<remaining size>`**

   Align to (Ausrichten auf): **`MiB`**

   Create as (Erstellen als): **`Primary Partition`**

   Partition name (Partitionsname):

   File system (Dateisystem): **`fat32`**

   Bezeichnung: **`BOOT`**

4. Um die NTFS-Datenpartition mit einer Kapazität von mindestens 18 GB zu erstellen, wählen Sie das Dateisymbol aus, legen Sie die folgenden Parameter fest und wählen Sie Hinzufügen.

   Free space preceding (Freier Platz vorher): **`0`**

   New size (Neue Größe):**`<remaining size>`**

   Free space following (Freier Platz danach): **`0`**

Align to (Ausrichten auf): **MiB**

Create as (Erstellen als): **Primary Partition**

Partition name (Partitionsname):

File system (Dateisystem): **nfts**

Bezeichnung: **Data**

5. Wählen Sie in der Menüleiste „Bearbeiten“ und dann „Alle Operationen anwenden“. In einer Warnmeldung werden Sie gefragt, ob Sie die Änderungen übernehmen möchten. Wählen Sie Anwenden aus.
6. Nachdem die Partitionen FAT32 und NTFS erstellt wurden, werden die Partitionsinformationen des USB-Laufwerks in der GParted Konsole angezeigt. Notieren Sie sich den Laufwerkspfad der B00T Partition. Sie benötigen ihn, um den nächsten Schritt abzuschließen.

Um das USB-Laufwerk von der Partition aus bootfähig zu machen FAT32

1. Stellen Sie sicher, dass Sie das benutzerdefinierte Ubuntu-ISO-Image aus dem Abschnitt mit den Voraussetzungen heruntergeladen haben.
2. Wenn Sie Ubuntu 20.04 verwenden, müssen Sie es UNetbootin mit der zugehörigen Binärdatei ausführen. So gehen Sie vor:

   a. Laden Sie die neueste UNetbootin Binärdatei in Ihren Download-Ordner herunter. In unserem Beispiel verwenden wir`unetbootin-linux64-702.bin`.

   b. Drücken Sie Strg+Alt+T, um ein neues Terminalfenster zu öffnen. Sie können auch in der Menüleiste Aktivitäten auswählen, **terminal** in der Suchleiste etwas eingeben und dann das Terminalsymbol auswählen.

   c. Verwenden Sie die folgenden Befehle, um zum Speicherort der Binärdatei zu navigieren, der Datei die Ausführungsberechtigung zu erteilen und sie auszuführen UNetbootin. Stellen Sie sicher, dass Sie den Dateinamen in den Befehlen anpassen, falls die Version nicht mit der Version in Ihrer heruntergeladenen Binärdatei übereinstimmt.

```
cd Downloads
sudo chmod +x ./unetbootin-linux64-702.bin
```

```
sudo ./unetbootin-linux64-702.bin
```

Wenn Sie eine ältere Version von Ubuntu verwenden, installieren Sie sie UNetbootin aus dem zugehörigen Repository, indem Sie die folgenden Befehle ausführen:

```
sudo add-apt-repository ppa:gezakovacs/ppa
sudo apt-get update; sudo apt-get install unetbootin
sudo unetbootin
```

3. Gehen Sie auf der UNetbootinKonsole wie folgt vor:

   a. Wählen Sie das Optionsfeld Disk Image aus.

   b. Wählen Sie für den Disk-Image-Typ ISO aus der Dropdownliste aus.

   c. Öffnen Sie die Dateiauswahl und wählen Sie die Ubuntu-ISO aus, die im Abschnitt mit den Voraussetzungen angegeben ist.

   d. Wählen Sie als Typ (Type) USB Drive (USB-Laufwerk) aus.

   e. Wählen Sie für Drive den Laufwerkspfad für Ihre B00T Partition, in unserem Fall. **`/dev/sda1`**

   f. Wählen Sie OK aus.

> **Tip**
>
> Wenn Sie die Warnmeldung /dev/sda1 not mounted erhalten, wählen Sie OK, um die Meldung zu schließen, trennen Sie das USB-Laufwerk, schließen Sie das Laufwerk wieder an und folgen Sie dann den vorherigen Schritten, um das Ubuntu-ISO-Image zu erstellen.

Um die DeepRacer AWS-Aktualisierungsdateien auf die NTFS-Partition zu extrahieren

1. Entpacken Sie das Software-Update-Paket, das Sie aus dem Abschnitt mit den Voraussetzungen heruntergeladen haben.
2. Extrahieren Sie den Inhalt des Aktualisierungspakets in das Stammverzeichnis der Datenpartition (NTFS) Ihres USB-Laufwerks.

Windows

Folgen Sie den Anweisungen hier, um die Aktualisierungsmedien für Ihr DeepRacer AWS-Gerät auf einem Windows-Computer vorzubereiten.

Um das USB-Laufwerk zu löschen

1. Öffnen Sie die Windows-Befehlszeile`diskpart`, geben Sie die Eingabetaste ein und wählen Sie OK, um Windows zu starten DiskPart.
2. Sobald das Terminal für Microsoft DiskPart geöffnet ist, listen Sie die verfügbaren Festplatten auf, um das USB-Laufwerk zu finden, das Sie reinigen möchten, indem Sie `list disk` nach der Eingabeaufforderung DISKPART> eingeben.
3. Wählen Sie die Festplatte aus, die Ihrem USB-Laufwerk entspricht. Zum Beispiel haben wir `select Disk 2` nach der Eingabeaufforderung DISKPART> eingegeben. Lesen Sie die Ausgabe sorgfältig durch, um sicherzustellen, dass Sie die Festplatte ausgewählt haben, die Sie reinigen möchten, da der nächste Schritt irreversibel ist.
4. Wenn Sie sicher sind, dass Sie die richtige Festplatte ausgewählt haben, geben Sie `Clean` nach der Eingabeaufforderung DISKPART> die Eingabe ein.
5. Geben Sie `list disk` nach der Eingabeaufforderung DISKPART> erneut ein. Suchen Sie in der Tabelle nach der Festplatte, die Sie gereinigt haben, und vergleichen Sie die

Festplattengröße mit dem freien Speicherplatz. Wenn die beiden Werte übereinstimmen, war die Reinigung erfolgreich.

6. Beenden Sie die DiskPartWindows-Konsole, indem Sie `Exit` nach der Eingabeaufforderung DISKPART> Folgendes eingeben.

Um das USB-Laufwerk zu partitionieren

1. Öffnen Sie die Windows-Eingabeaufforderung`diskmgmt.msc`, geben Sie die Eingabetaste ein und wählen Sie OK, um die Disk Management Console zu starten.
2. Wählen Sie in der Disk Management Console Ihr USB-Laufwerk aus.
3. Um die FAT32 Partition mit einer Kapazität von 4 GB zu erstellen, öffnen Sie das Kontextmenü (Rechtsklick) im Bereich Nicht zugewiesener Speicherplatz Ihres USB-Laufwerks und wählen Sie New Simple Volume. Der New Simple Volume Wizard wird angezeigt.
4. Sobald der New Simple Volume Wizard angezeigt wird, gehen Sie wie folgt vor:

   a. Stellen Sie auf der Seite „Volume-Größe angeben“ den folgenden Parameter ein und wählen Sie dann Weiter.

      Einfache Volumengröße in MB: **`4096`**

   b. Aktivieren Sie auf der Seite „Laufwerkbuchstaben oder Pfad zuweisen“ das Optionsfeld Den folgenden Laufwerkbuchstaben zuweisen:, wählen Sie einen Laufwerkbuchstaben aus der Dropdownliste aus und klicken Sie dann auf Weiter. Notieren Sie sich den zugewiesenen Laufwerksbuchstaben. Sie benötigen ihn später, um die FAT32 Partition bootfähig zu machen.

   c. Aktivieren Sie auf der Seite Partition formatieren das Optionsfeld Dieses Volume mit den folgenden Einstellungen formatieren, legen Sie die folgenden Parameter fest und wählen Sie dann Weiter.

      File system (Dateisystem): **`FAT32`**

      Größe der Zuordnungseinheit: **`Default`**

      Etikett des Datenträgers: **`BOOT`**

      Lassen Sie die Option Schnellformatierung ausführen aktiviert.

5. Um die NTFS-Partition mit der verbleibenden Festplattenkapazität zu erstellen, öffnen Sie das Kontextmenü (Rechtsklick) auf dem verbleibenden nicht zugewiesenen Speicherplatz Ihres USB-Laufwerks und wählen Sie New Simple Volume. Der New Simple Volume Wizard wird angezeigt.
6. Sobald der New Simple Volume Wizard angezeigt wird, gehen Sie wie folgt vor:

   a. Stellen Sie auf der Seite „Volume-Größe angeben“ die Größe des einfachen Datenträgers in MB so ein, dass sie mit dem maximalen Festplattenspeicher in MB übereinstimmt, und wählen Sie dann Weiter.

   b. Aktivieren Sie auf der Seite „Laufwerkbuchstaben oder Pfad zuweisen“ das Optionsfeld Den folgenden Laufwerkbuchstaben zuweisen:, wählen Sie einen Laufwerksbuchstaben aus der Dropdownliste aus und klicken Sie dann auf Weiter.

   c. Aktivieren Sie auf der Seite Partition formatieren das Optionsfeld Dieses Volume mit den folgenden Einstellungen formatieren, legen Sie die folgenden Parameter fest und wählen Sie dann Weiter.

      File system (Dateisystem): **`NTFS`**

      Größe der Zuordnungseinheit: **`Default`**

      Etikett des Datenträgers: **`Data`**

      Lassen Sie die Option Schnellformatierung ausführen aktiviert.

Um das USB-Laufwerk von der Partition aus bootfähig zu machen FAT32

1. Stellen Sie sicher, dass Sie das benutzerdefinierte Ubuntu-ISO-Image aus dem Abschnitt mit den Voraussetzungen heruntergeladen haben.
2. Starten Sie nach dem Herunterladen UNetbootindie UNetbootinKonsole.
3. Gehen Sie auf der UNetbootin Konsole wie folgt vor:

   a. Aktivieren Sie das Optionsfeld Disk Image.

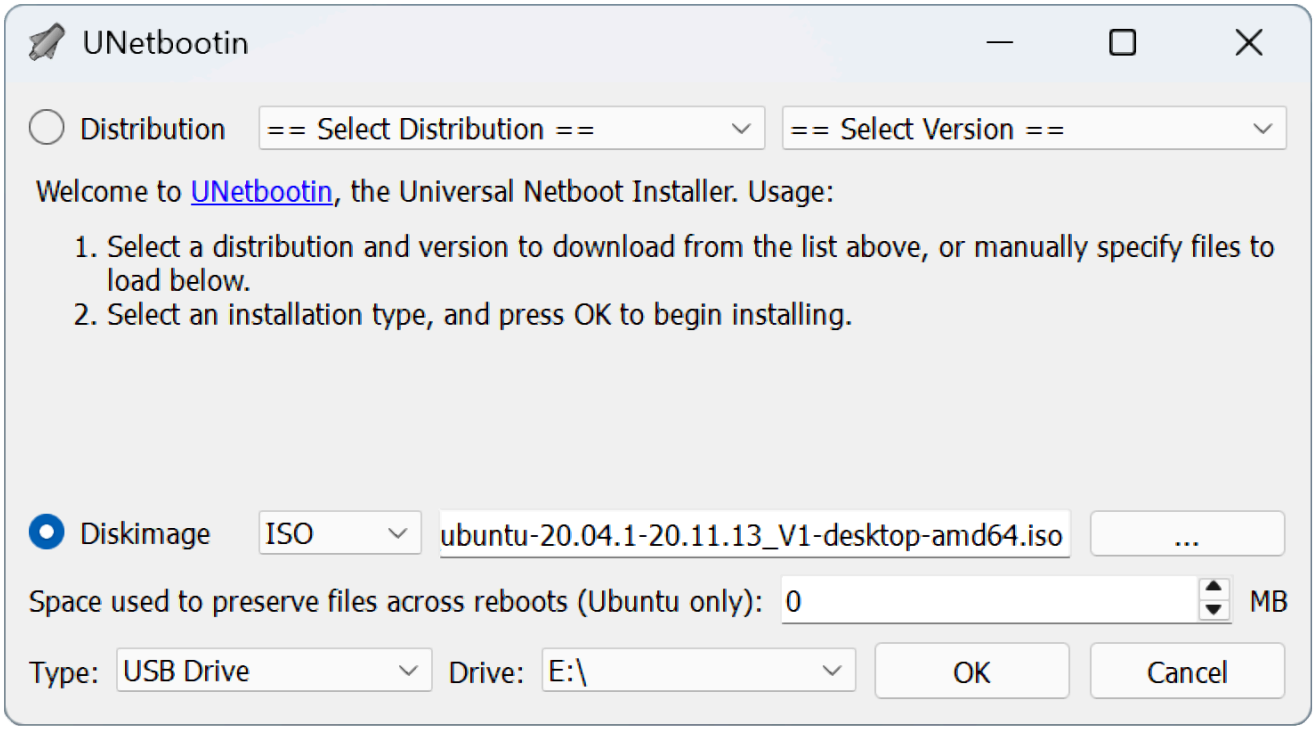
   b. Wählen Sie für Disk-Image die Option ISO aus der Drop-down-Liste aus.

   c. Öffnen Sie die Dateiauswahl und wählen Sie die benutzerdefinierte Ubuntu-ISO-Datei aus.

   d. Wählen Sie als Typ (Type) USB Drive (USB-Laufwerk) aus.

e. Wählen Sie für Drive den Laufwerksbuchstaben, der der von Ihnen erstellten FAT32 Partition entspricht. In unserem Fall ist es`E:\`.

f. Wählen Sie OK aus.



Um die DeepRacer AWS-Aktualisierungsdateien auf die NTFS-Partition zu extrahieren

1. Entpacken Sie das Software-Update-Paket, das Sie aus dem Abschnitt mit den Voraussetzungen heruntergeladen haben.

   > **Tip**
   >
   > Wenn Ihr bevorzugtes Tool die Datei nicht erfolgreich entpacken kann, versuchen Sie es mit dem Befehl PowerShell Expand-Archive.

2. Extrahieren Sie den Inhalt des Aktualisierungspakets in das Stammverzeichnis der Datenpartition (NTFS) Ihres USB-Laufwerks.

macOS

Folgen Sie den Anweisungen hier, um die Aktualisierungsmedien für Ihr DeepRacer AWS-Gerät mit einem Mac vorzubereiten.

Um das USB-Laufwerk zu löschen und zu partitionieren

1. Schließen Sie das USB-Laufwerk an Ihren Mac an.
2. Drücken Sie Befehlstaste+Leertaste, um das Spotlight-Suchfeld zu öffnen, und geben Sie dann die Eingabetaste ein. `Disk Utility`

   Sie können auch Finder > Anwendungen > Dienstprogramme > Festplatten-Hilfsprogramm wählen, um das Festplatten-Hilfsprogramm zu öffnen.
3. Wählen Sie in der Menüleiste „Ansicht“ und anschließend „Alle Geräte anzeigen“.
4. Wählen Sie in der Seitenleiste unter Extern das USB-Laufwerk aus, das Sie formatieren möchten, und wählen Sie dann Löschen.
5. In einem neuen Fenster werden Sie aufgefordert, zu bestätigen, dass Sie Ihr USB-Laufwerk löschen möchten. Außerdem können Sie dessen Namen, Format und Partitionsschema ändern. Sie müssen den Namen noch nicht ändern. Wählen Sie für Format und Schema die folgenden Optionen und wählen Sie Löschen.

   - Format: Mac OS Extended (Journaled)
   - Schema: GUID-Partitionstabelle

   Sobald der Löschvorgang abgeschlossen ist, wählen Sie im Dialogfenster die Option Fertig.
6. Wählen Sie im Hauptfenster des Festplatten-Dienstprogramms in der Seitenleiste Ihr USB-Laufwerk aus und wählen Sie in der oberen Werkzeugleiste die Option Partition aus. Ein Fenster mit dem Titel Partitionsgerät "***YOUR-USB-DRIVE***„? wird angezeigt. Wählen Sie die Schaltfläche Hinzufügen (+), um eine neue Partition zu erstellen.
7. Nachdem Sie die neue Partition erstellt haben, wählen Sie unter Partitionsinformationen Folgendes aus und geben Sie es ein:

   - Name: **BOOT**
   - Format: MS-DOS (FAT)
   - Größe: GB **4**

> **Tip**
>
> Wenn das Eingabefeld Größe ausgegraut ist, nachdem Sie MS-DOS (FAT) als Format ausgewählt haben, können Sie das Steuerelement zur Größenänderung im Partitionsdiagramm ziehen, bis die **BOOT** Partition 4 GB groß ist.

Klicken Sie noch nicht auf Anwenden.

8. Wählen Sie die andere Partition ohne Titel aus, wählen Sie unter Partitionsinformationen die folgenden Optionen aus und geben Sie sie ein:

   - Name: **Data**
   - Format: exFAT
   - Größe: der verbleibende Speicherplatz des USB-Laufwerks (in GB)

   Wählen Sie Anwenden aus.

9. Ein neues Fenster öffnet sich und zeigt Ihnen die Änderungen, die am USB-Laufwerk vorgenommen werden. Stellen Sie sicher, dass diese Änderungen korrekt sind. Um zu bestätigen und mit der Erstellung der neuen Partitionen zu beginnen, wählen Sie Partition.
10. Wählen Sie auf der Disk Utility-Konsole in der Seitenleiste die BOOT-Partition aus und wählen Sie dann in der Werkzeugleiste „Info“ aus. Notieren Sie sich den Wert des BSD-Geräteknotens. Er kann sich von dem Wert unterscheiden, der in diesem Tutorial verwendet wurde. In unserem Fall ist der zugewiesene Wert`disk4s2`. Sie müssen diesen Pfad angeben, wenn Sie das USB-Laufwerk von der FAT32 Partition aus bootfähig machen möchten.

Um das USB-Laufwerk von der Partition aus bootfähig zu machen FAT32

1. Stellen Sie sicher, dass Sie das benutzerdefinierte Ubuntu-ISO-Image aus dem Abschnitt mit den Voraussetzungen heruntergeladen haben.
2. Wählen Sie nach dem Herunterladen UNetbootinim Kontextmenü (Rechtsklick) die Option Öffnen aus. In einer Sicherheitsabfrage werden Sie gefragt, ob Sie die Anwendung öffnen möchten. Wählen Sie Öffnen, um die UNetbootin Konsole zu starten.

Wenn Sie einen Mac mit Apple Silicon verwenden und die UNetbootin Konsole nach Auswahl von Öffnen nicht angezeigt wird, stellen Sie sicher, dass Rosetta 2 installiert ist, indem Sie die folgenden Schritte ausführen:

a. Öffnen Sie ein Terminalfenster, indem Sie Finder > Programme > Dienstprogramme > Terminal wählen.

b. Geben Sie den folgenden Befehl ein, um Rosetta 2 zu installieren:

```
softwareupdate --install-rosetta
```

c. Versuchen Sie erneut zu öffnen. UNetbootin

3. Gehen Sie auf der UNetbootin Konsole wie folgt vor:

a. Aktivieren Sie das Optionsfeld Disk Image.

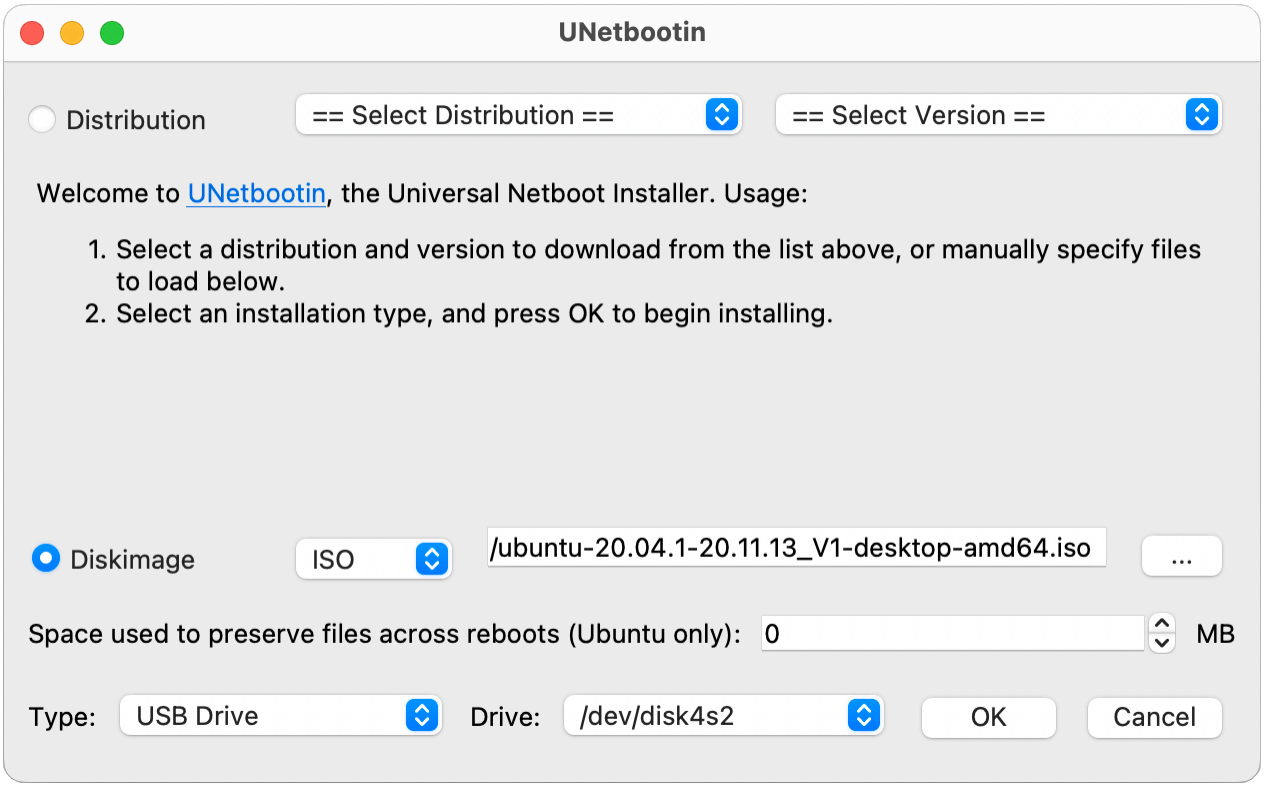
b. Wählen Sie für Disk-Image die Option ISO aus der Drop-down-Liste aus.

c. Öffnen Sie die Dateiauswahl und wählen Sie die benutzerdefinierte Ubuntu-ISO-Datei aus.

d. Wählen Sie als Typ (Type) USB Drive (USB-Laufwerk) aus.

e. Wählen Sie für Drive den BSD-Geräteknoten für Ihre BOOT-Partition, in unserem Fall. `/dev/disk4s2`

f. Wählen Sie OK aus.



> **Tip**
>
> Wenn Sie die Warnmeldung /dev/disk4s2 not mounted erhalten, wählen Sie OK, um die Meldung zu schließen, trennen Sie das USB-Laufwerk, stecken Sie das Laufwerk wieder ein und folgen Sie dann den obigen Schritten, um das Ubuntu-ISO-Image zu erstellen.

Um die DeepRacer AWS-Aktualisierungsdateien auf die exFAT-Partition zu extrahieren

1. Entpacken Sie das Software-Update-Paket, das Sie aus dem Abschnitt mit den Voraussetzungen heruntergeladen haben.
2. Extrahieren Sie den Inhalt des Aktualisierungspakets in das Stammverzeichnis der Datenpartition (exFAT) Ihres USB-Laufwerks.

Aktualisieren Sie Ihr DeepRacer AWS-Gerät auf den Ubuntu 20.04-Softwarestack

Sobald Sie das USB-Aktualisierungsmedium wie in den vorherigen Schritten beschrieben erstellt haben, können Sie Ihr DeepRacer AWS-Gerät auf den neuesten Software-Stack aktualisieren, einschließlich Ubuntu 20.04 Focal Fossa, Intel® OpenVINO™ Toolkit 2021.1.110, Foxy Fitzroy und Python 3.8. ROS2

> **Important**
>
> Durch die Aktualisierung auf den neuen DeepRacer AWS-Softwarestack werden alle Daten auf Ihrem DeepRacer AWS-Gerät gelöscht.

So aktualisieren Sie Ihre DeepRacer AWS-Gerätesoftware auf den Ubuntu 20.04-Stack

1. Connect Sie Ihr DeepRacer AWS-Gerät mit einem Monitor. Sie benötigen ein HDMI-to-HDMI, HDMI-to-DVI, oder ein ähnliches Kabel. Stecken Sie das HDMI-Ende des Kabels in den HDMI-Anschluss des Rechenmoduls und das andere Ende in einen kompatiblen Anschluss am Monitor.
2. Schließen Sie eine USB-Tastatur und -Maus an. Das Rechenmodul des DeepRacer AWS-Geräts verfügt über drei USB-Anschlüsse an der Vorderseite des Fahrzeugs, auf beiden Seiten und einschließlich des Anschlusses, an den die Kamera angeschlossen ist. Ein vierter USB-Anschluss befindet sich an der Rückseite des Fahrzeugs, im Raum zwischen der Rechenbatterie und dem LED-Rücklicht.
3. Stecken Sie das USB-Aktualisierungsmedium in einen verfügbaren USB-Anschluss Ihres Rechenmoduls. Schalten Sie Ihr DeepRacer AWS-Gerät ein oder setzen Sie es zurück und drücken Sie wiederholt die ESC-Taste, um das BIOS aufzurufen.
4. Wählen Sie im BIOS-Fenster Boot From File und dann die Option mit dem Namen Ihrer Startpartition aus, in unserem Fall heißt sie BOOT<EFI>, wählen Sie dann <BOOT>und schließlich BOOTx64.EFI.
5. Nach dem Booten des Rechenmoduls erscheint auf dem Desktop ein Terminalfenster, in dem der Fortschritt angezeigt wird. Das DeepRacer AWS-Gerät beginnt nach zehn Sekunden automatisch mit dem Aktualisierungsvorgang. Zu diesem Zeitpunkt müssen Sie keine Angaben machen.

Wenn ein Fehler auftritt und das Update fehlschlägt, starten Sie den Vorgang ab Schritt 1 erneut. Ausführliche Fehlermeldungen finden Sie in der `result.log` Datei, die auf der Datenpartition des USB-Laufwerks generiert wurde.

6. Warten Sie, bis das Update abgeschlossen ist. Wenn das Zurücksetzen auf die Werkseinstellungen abgeschlossen ist, wird das Terminalfenster automatisch geschlossen.
7. Nachdem die Gerätesoftware aktualisiert wurde, trennen Sie das USB-Laufwerk vom Rechenmodul. Sie können Ihr DeepRacer AWS-Gerät jetzt neu starten oder herunterfahren.
8. Das DeepRacer AWS-Gerät verwendet nach dem Update standardmäßig die folgenden Benutzeranmeldeinformationen. Bei Ihrer ersten Anmeldung werden Sie aufgefordert, Ihr Passwort zu ändern.

   Benutzer: Deepracer

   Passwort: `deepracer`

Erstellen Sie Ihren physischen Track für AWS DeepRacer

In diesem Abschnitt wird beschrieben, wie Sie einen physischen Track für ein DeepRacer AWS-Modell erstellen können. Um Ihr AWS DeepRacer autonom zu betreiben und Ihr Reinforcement-Learning-Modell in einer physischen Umgebung zu testen, benötigen Sie einen physischen Track. Ihr Track ähnelt dem simulierten Track, der beim Training verwendet wurde, und repliziert die Umgebung, in der das bereitgestellte DeepRacer AWS-Modell trainiert wurde.

Für ein optimales Erlebnis empfehlen wir, vorgedruckte Gleise und Gleisbarrieren zu verwenden. Die Verwendung von vorgedruckten Gleisen und Barrieren erleichtert die reibungslose Einrichtung und Installation der DeepRacer AWS-Trackumgebung. Anstatt ein Gleis von Grund auf neu zu bauen, bauen Sie vorgedruckte Gleisabschnitte und Gleisbarrieren zusammen. Wenn Ihre Veranstaltung vorbei ist, können Sie die vorgedruckten Spuren und Barrieren zerlegen und für future Veranstaltungen wiederverwenden. Vorgedruckte Gleise und Barrieren sowie Informationen zur Schätzung des Platzbedarfs und anderer Anforderungen für Veranstaltungen sind in AWS DeepRacer Storefront verfügbar.

Themen

- Verfolge Materialien und baue Tools
- Machen Sie sich auf den Weg zu AWS DeepRacer
- DeepRacer AWS-Track-Designvorlagen

Verfolge Materialien und baue Tools

Bevor Sie mit dem Bau Ihrer Strecke beginnen, bereiten Sie die folgenden Materialien und Werkzeuge vor.

Themen

- Materialien, die Sie möglicherweise benötigen
- Werkzeuge, die Sie möglicherweise benötigen

Materialien, die Sie möglicherweise benötigen

Um eine Strecke zu bauen, benötigen Sie die folgenden Materialien:

- Für Streckenbegrenzungen:

Sie können eine Strecke mit Klebeband erstellen, das etwa 5 cm (2 Zoll) breit und weiß oder cremefarben ist. Damit begrenzen Sie die dunkel gefärbte Strecke. Für eine dunkle Oberfläche verwenden Sie ein weißes oder cremefarbenes Klebeband. Zum Beispiel 1,88 Zoll Klebeband in creme-weiß oder 1,88 Zoll Maskierungsklebeband (weniger haftfest).

- Für die Streckenoberfläche:

  Sie können eine Strecke auf einem dunkel gefärbten Hartboden wie Hartholz, Teppich, Beton oder Asphaltvlies erstellen. Letzteres imitiert die reale Fahrbahnoberfläche mit minimaler Reflexion. Auch verzahnte Schaumstoff- oder Gummiplatten sind eine gute Wahl.

Werkzeuge, die Sie möglicherweise benötigen

Die folgenden Werkzeuge sind entweder erforderlich oder hilfreich, um Ihre Strecke zu entwerfen und zu bauen:

- Maßband und Schere

  Ein gutes Maßband und eine Schere sind unerlässlich, um Ihre Strecke zu bauen. Wenn Sie noch keine haben, können Sie hier ein Maßband und hier eine Schere bestellen.

- Optionale Designwerkzeuge

  Um Ihre eigene Strecke zu entwerfen, benötigen Sie möglicherweise einen Winkelmesser, ein Lineal, einen Bleistift, ein Messer und einen Kompass.

Machen Sie sich auf den Weg zu AWS DeepRacer

Wenn Sie Ihre Strecke bauen, ist es sinnvoll, mit einem einfachen Design zu beginnen – beispielsweise einer geraden Strecke oder einer Strecke mit nur einer Kurve. Danach können Sie zu den Rundstrecken übergehen. Hier führen wir Sie am Beispiel einer Strecke mit nur einer Kurve durch die Schritte zum Bau Ihrer eigenen Strecke. Betrachten wir zunächst die Abmessungsanforderungen an eine Strecke.

Themen

- Abmessungsanforderungen
- Überlegungen zur Modellleistung

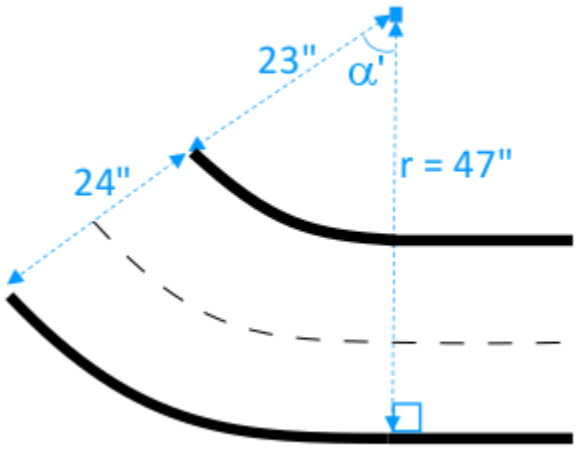
- Schritte zum Aufbau des Tracks

Abmessungsanforderungen

Sie können eine Strecke in beliebiger Form bauen. Sie muss jedoch die folgenden Anforderungen erfüllen.

- Minimaler Kurvenradius:

  Bei einer Strecke mit Kurven wird der Kurvenradius (`r`) von der Kreismitte bis zur Außenkante gemessen (wie unten dargestellt).



  Der Mindestwenderadius (r_{min}) hängt vom Wendewinkel des Gleises (α) an einer Ecke ab und sollte den folgenden Grenzwerten entsprechen:

  - Wenn der Kurvenradius der Strecke $\alpha \leq 90$ degrees beträgt,

    $r_{min} \geq 25$ inches

    Wir empfehlen 76 cm (30 Zoll).

  - Wenn der Wendewinkel des Gleises α ist $\alpha > 90$ degrees

    $r_{min} \geq 30$ inches.

    Wir empfehlen 89 cm (35 Zoll).

- Streckenbreite.

  Die Streckenbreite (w_{track}) sollte den folgenden Grenzwert einhalten:

  $w_{track} \geq 24 \pm 3$ inches.
- Streckenoberfläche:

  Die Streckenoberfläche sollte glatt und von einheitlicher dunkler Farbe sein. Der minimale umschlossene Bereich sollte eine Größe von `30 inches x 60 inches` haben.

  Teppichböden und Holzböden eignen sich gut. Verzahnte Schaumstoff- oder Gummiplatten passen sich der simulierten Umgebung besser an als Holz. Dies ist jedoch nicht zwingend erforderlich. Betonböden können aufgrund von Lichtreflexionen auf der Oberfläche problematisch sein.
- Streckenabsperrung

  Obwohl nicht erforderlich, empfehlen wir Ihnen, die Strecke mit gleichfarbigen Barrieren zu umfassen, die mindestens 76 cm (2,5 Fuß) hoch und überall 60 cm (2 Fuß) von der Strecke entfernt sind.

Überlegungen zur Modellleistung

Der Bau einer Strecke kann sich auf die Zuverlässigkeit und Leistung eines trainierten Modells auswirken. Die folgenden Faktoren sollten Sie bei der Erstellung Ihrer eigenen Strecken berücksichtigen.

1. Platzieren Sie keine weißen Objekte auf oder in der Nähe Ihrer Strecke. Entfernen Sie bei Bedarf weiße Gegenstände von der Strecke oder aus ihrer Umgebung. Das Training in der simulierten Umgebung geht davon aus, dass nur die Streckenbegrenzungen weiß sind.
2. Verwenden Sie sauberes und durchgehendes Klebeband, um die Streckenbegrenzungen zu markieren. Durchtrennte oder geknickte Streckenbegrenzungen können die Leistung des trainierten Modells beeinträchtigen.
3. Vermeiden Sie die Verwendung einer reflektierenden Oberfläche als Streckenuntergrund. Reduzieren Sie Blendeffekte durch helle Lichter. Spiegelungen an geraden Kanten können als Objekte oder Begrenzungen fehlinterpretiert werden.
4. Verwenden Sie keinen Streckenuntergrund mit von den Streckenlinien abweichenden Linienmarkierungen. Das Modell könnte die zusätzlichen Linien als Teil der Strecke interpretieren.

5. Platzieren Sie Barrieren um die Strecke herum, um Ablenkungen durch Hintergrundobjekte zu reduzieren.

Schritte zum Aufbau des Tracks

Zur Veranschaulichung verwenden wir die einfachste Strecke mit einer Kurve. Sie können die Anweisungen so modifizieren, dass eine komplexere Strecke wie eine S-Kurve, eine Schleife oder die AWS re:invent 2018 Strecke entsteht.

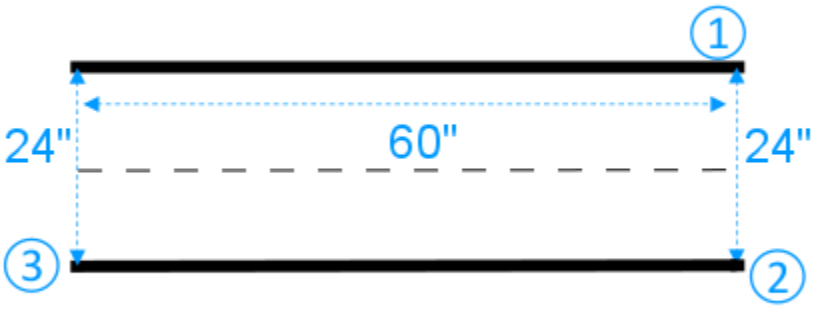
So bauen Sie eine DeepRacer AWS-Single-Turn-Strecke

1. Um den geraden Abschnitt der Schiene zu konstruieren, führen Sie die folgenden Schritte aus und orientieren Sie sich an der Abbildung.

   a. Legen Sie ein 152 cm (60 Zoll) langes Stück Klebeband auf den Boden, um die erste Begrenzung in einer geraden Linie (1) zu platzieren.

   b. Verwenden Sie ein Maßband, um die beiden Endpunkte (2) und (3) der zweiten Begrenzung zu bestimmen. Platzieren Sie diese 60 cm (24 Zoll) von den beiden Enden der ersten Begrenzung entfernt.

   c. Platzieren Sie ein weiteres 60 cm (24 Zoll) langes Stück Klebeband auf dem Boden, um die zweite Begrenzung zur Verbindung der beiden Endpunkte (2) und (3) zu verlegen.

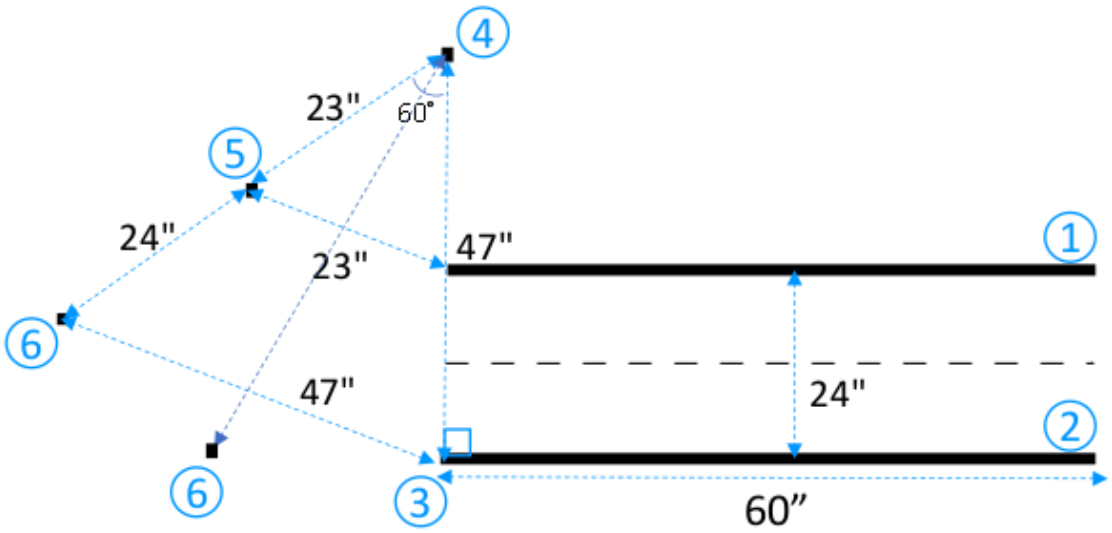


Wir gehen davon aus, dass das gerade Streckensegment 152 cm (60 Zoll) lang und 60 cm (24 Zoll) breit ist. Sie können die Länge und Breite an Ihren Raum anpassen, sofern die Abmessungsanforderungen erfüllt sind.

2. Um die Strecke mit einer Kurve im Winkel von 60 Grad zu verlegen, gehen Sie wie folgt vor (siehe Diagramm):

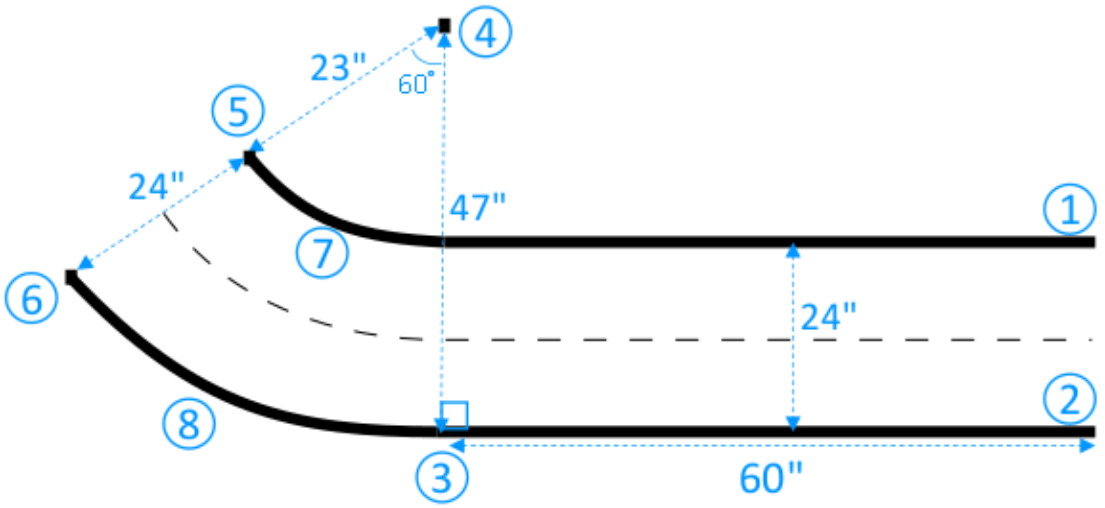
   a. Verwenden Sie das Maßband, um die Mitte (4) des Kurvenradius (4-3 oder 4-6) zu bestimmen. Markieren Sie die Mitte mit einem Stück Klebeband.

   b. Zeichnen Sie ein gleichseitiges Dreieck. Die drei Seiten sind (3-4), (4-6) und (6-3).



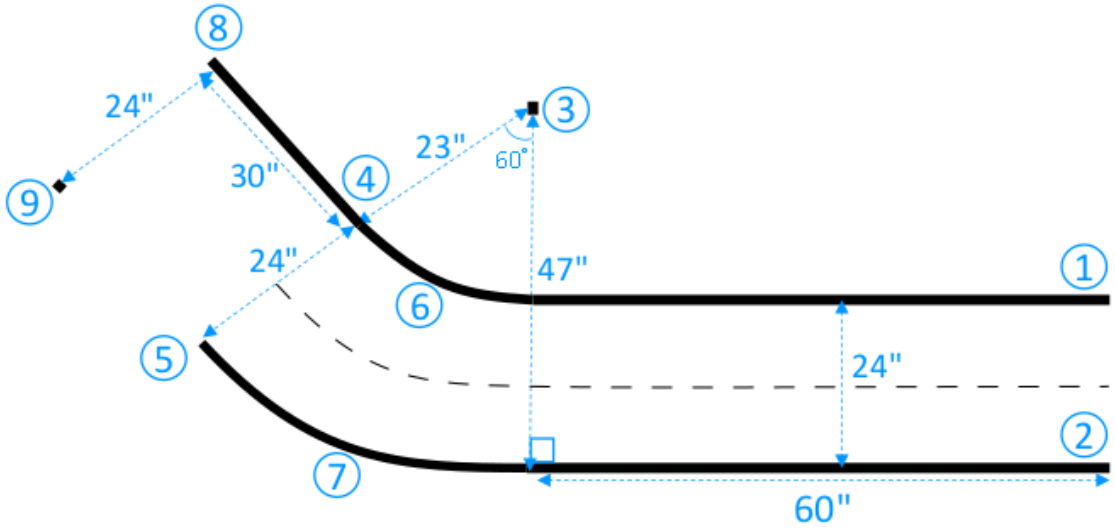
   Um eine 60-Grad-Kurve entlang der Strecke zu erstellen, verwenden Sie das gleichseitige Dreieck (3-4-6) zur Bestimmung der Positionen der beiden letzten Endpunkte (5) und (6) für das gekrümmte Streckensegment. Für Kurven mit einem anderen Winkel können Sie mit einem Winkelmesser (oder einer Winkelmesser-App) die beiden Endpunkte (5) und (6) des Kurven-Streckensegments bestimmen. Kurven mit abweichendem Radius sind akzeptabel, solange die in Schritt 2 vorgeschriebenen Mindestkurvenradien eingehalten werden.

   c. Platzieren Sie kleine Klebebandsegmente, z. B. je 10 cm (4 Zoll), auf dem Boden, um die Begrenzungen der Kurvensegmente (7) und (8) zu verlegen und mit den geraden Begrenzungen zu verbinden. Die beiden Kurvenbegrenzungen müssen nicht parallel sein.



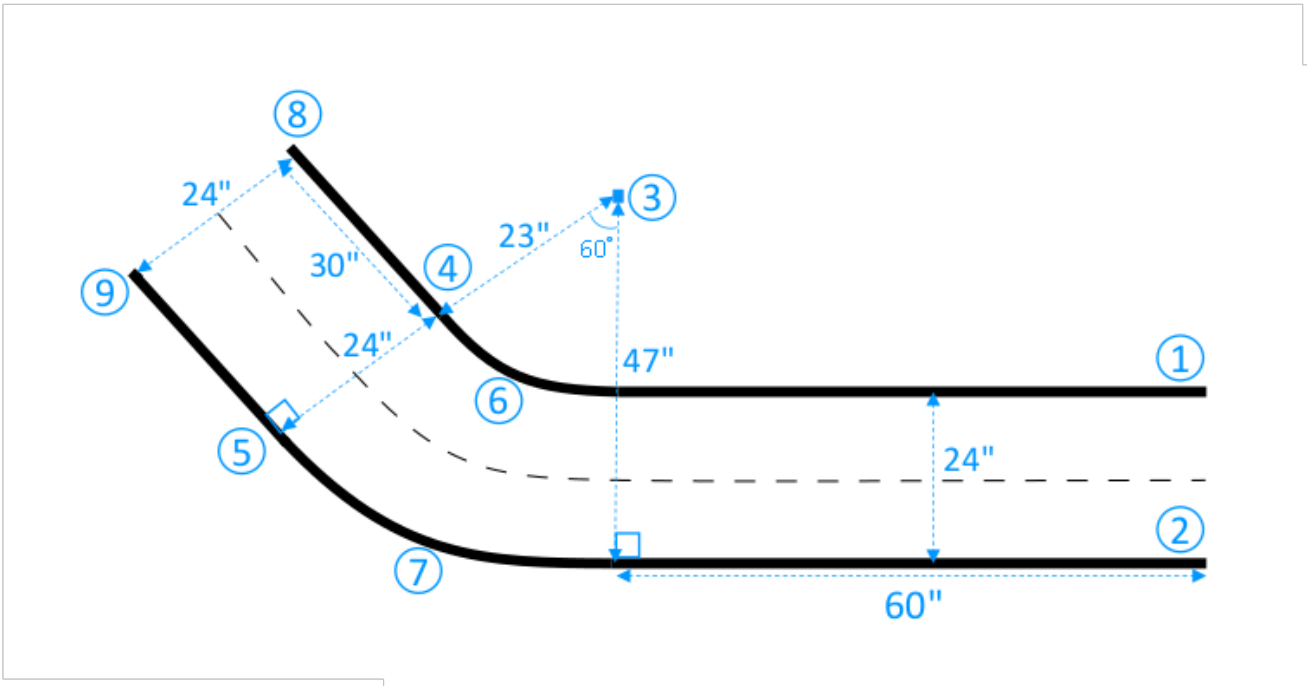
3. Um die Strecke mit dem nächsten geraden Segment von 76 cm (30 Zoll) Länge und 60 cm (24 Zoll) Breite zu verlängern, gehen Sie wie folgt vor:

   a. Platzieren Sie ein 76 cm (30 Zoll) langes Stück Klebeband auf dem Boden, um die erste Begrenzung (4-8) senkrecht zur Kante (3-5) zu verlegen.



b. Verwenden Sie das Maßband, um den Endpunkt der zweiten Begrenzung (9) zu bestimmen. Sie können die Länge der geraden Linien an den verfügbaren Platz anpassen.

c. Platzieren Sie ein weiteres 76 cm (30 Zoll) langes Stück Klebeband auf den Boden, um die zweite Begrenzung (5-9) senkrecht zur Kante (3-5) zu verlegen.



Wir gehen davon aus, dass das zweite gerade Streckensegment 76 cm (30 Zoll) lang und 60 cm (24 Zoll) breit ist. Sie können die Länge und Breite an Ihren Platz anpassen. Es müssen jedoch die Abmessungsanforderungen erfüllt sein und die Abmessungen müssen mit den anderen Streckensegmenten übereinstimmen.

4. Optional können Sie Klebebandsegmente von 10 cm (4 Zoll) Länge zuschneiden und dann die Klebebandsegmente 5 cm (2 Zoll) auseinander entlang der Streckenmitte platzieren, um die gestrichelte Mittellinie zu bilden.

Sie haben nun den Bau der Strecke mit einer Kurve abgeschlossen. Damit Ihr Fahrzeug die befahrbaren Oberflächen besser von nicht befahrbaren Oberflächen unterscheiden kann, sollten Sie die nicht befahrbare Oberfläche mit einer Farbe versehen, die einen ausreichenden Kontrast zur Oberflächenfarbe auf der Strecke aufweist. Um die Sicherheit zu gewährleisten, können Sie die Strecke mit gleichfarbigen Barrieren umrunden, die an jeder Stelle mindestens 76 cm (2,5 Fuß) hoch und 60 cm (2 Fuß) von der Strecke entfernt sind.

Sie können die Anweisungen nutzen, um die Strecke auf komplexere Formen zu erweitern.

DeepRacer AWS-Track-Designvorlagen

Die folgenden Vorlagen für das Streckendesign zeigen DeepRacer AWS-Tracks, die Sie erstellen können, indem Sie den Anweisungen in diesem Abschnitt folgen.

> **Note**
>
> In diesem Abschnitt werden auch Vorlagen für Tracks vorgestellt, die vorgedruckt erhältlich sind. Die Montage von vorgedruckten Gleisen benötigt weniger Zeit und ist einfacher als die Konstruktion von Gleisen mit Ihren eigenen Materialien. Wir empfehlen, vorgedruckte Gleise und Barrieren zu verwenden. Um vorgedruckte Titel zu kaufen, besuchen Sie die DeepRacerAWS-Storefront.

Verwenden Sie für alle Titel die folgenden Spezifikationen, um dieselbe Farbproduktion zu reproduzieren:

- Grün: PMS 3395C
- Orange: PMS 137C
- Schwarz: PMS 432C
- Weiß: CMYK 0-0-2-2-9

Diese Spuren wurden mit den folgenden Materialien auf ihre Oberflächen getestet:

- PVC

  Die Tracks wurden auf 13-Unzen-Scrim-Vinyl mit einer matten Oberfläche gedruckt, um die Blendung zu reduzieren. PVC ist in der Regel billiger als Teppiche und bietet eine gute Leistung. PVC ist nicht so langlebig wie Teppiche.
- Teppichboden

  Die Tracks wurden auf einem 8-Unzen-Teppich mit farbstoffsublimierter Polyesterbeschichtung und gummierter Latexunterlage gedruckt. Der Teppich ist strapazierfähig und bietet eine hervorragende Leistung, ist aber teuer.

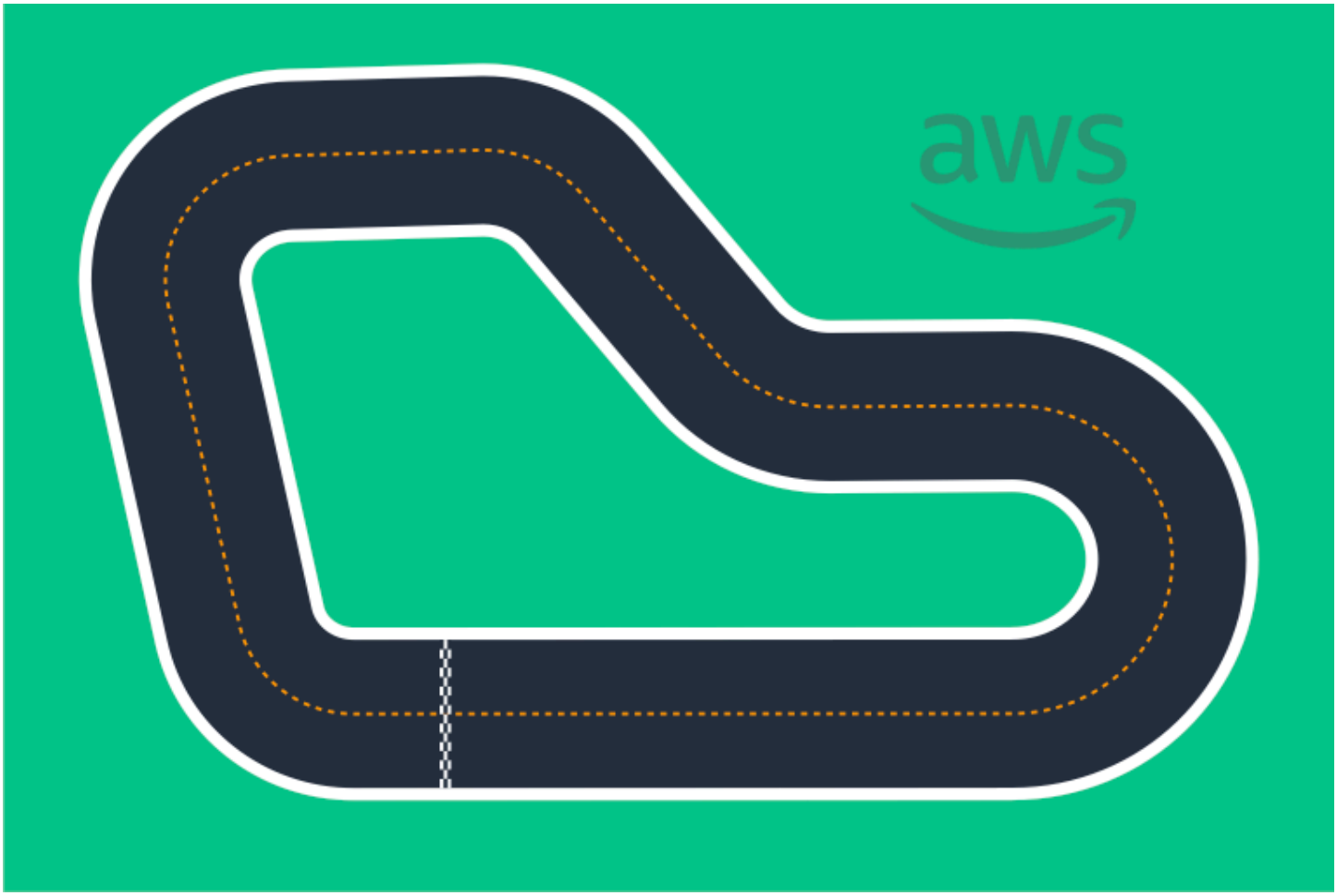
Aufgrund ihrer Größe können die Tracks nicht einfach auf ein einzelnes Stück Material gedruckt werden. Richten Sie die Streckenlinien beim Verbinden der Teile gut aus.

Themen

- Vorlage für AWS-Rennstrecken von DeepRacer A bis Z Speedway (Basic)
- Streckenvorlage für AWS DeepRacer Smile Speedway (Fortgeschrittene)
- Streckenvorlage für AWS DeepRacer RL Speedway (Advanced)
- AWS-Vorlage für einen DeepRacer Single-Turn-Track
- Vorlage AWS DeepRacer AWS-S-Kurvenbahnen
- DeepRacer AWS-Loop-Track-Vorlage

Vorlage für AWS-Rennstrecken von DeepRacer A bis Z Speedway (Basic)

Die AWS DeepRacer A to Z Speedway (Basic) -Strecke ist die beliebteste physische Wettkampfstrecke in der DeepRacer Geschichte von AWS. Es wurde ursprünglich auf der AWS re:invent 2018 veröffentlicht und hat den geringsten Platzbedarf aller DeepRacer physischen AWS-Wettkampfstrecken. Es ist vorgedruckt bei AWS DeepRacer Storefront erhältlich.

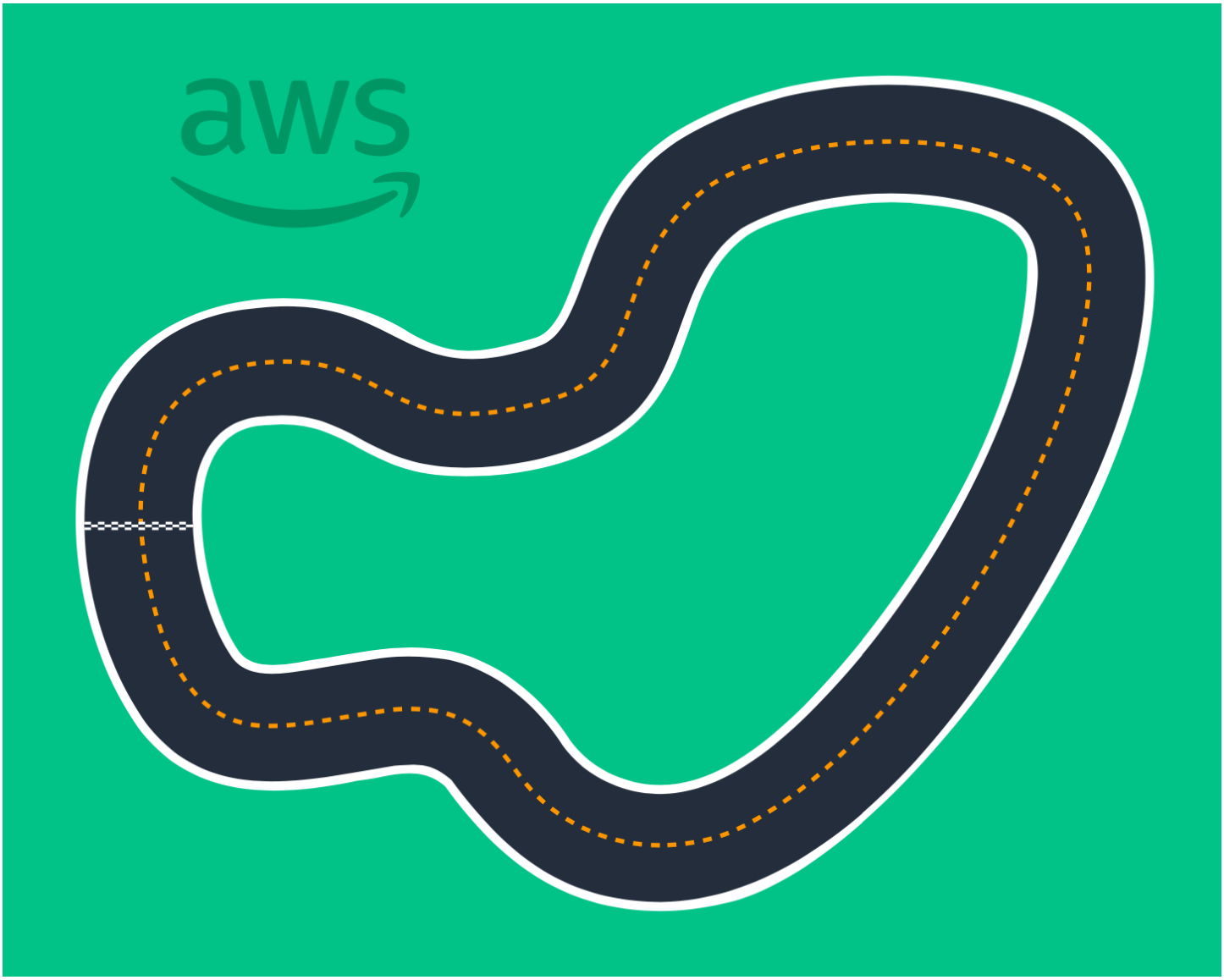


Wir empfehlen diese Strecke für Einsteiger und Rennanfänger. Mit einer Vielzahl von Pisten und Geraden bietet sie sowohl für Anfänger als auch für erfahrene Rennfahrer eine überzeugende Herausforderung. Die AWS-Strecke DeepRacer A bis Z Speedway (Basic) ist eine physische 1:1 -Reproduktion der virtuellen Strecke, die in der Konsole verfügbar ist. Es bietet Rennfahrern die Möglichkeit, ein Modell in einer virtuellen Umgebung zu trainieren und das Modell dann auf einem physischen DeepRacer AWS-Gerät für autonome Rennen auf einer physischen Strecke bereitzustellen.

Laden Sie diese AWS A to Z Speedway (Basic) -Datei herunter, um Ihre eigene Strecke von DeepRacer A bis Z Speedway (Basic) zu drucken oder zu erstellen.

Streckenvorlage für AWS DeepRacer Smile Speedway (Fortgeschrittene)

Die Strecke AWS DeepRacer Smile Speedway wurde ursprünglich als Strecke der AWS DeepRacer Championship 2019 veröffentlicht. Es ist vorgedruckt bei AWS DeepRacer Storefront erhältlich.



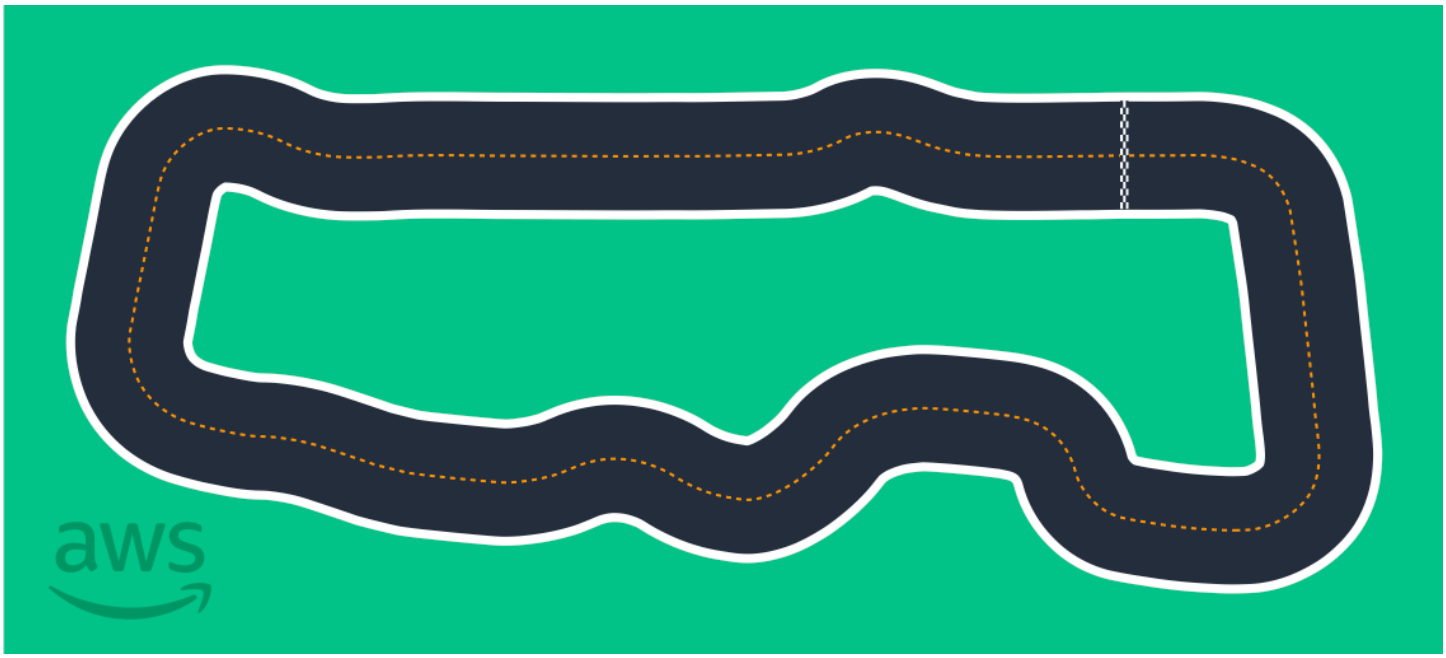
Wir empfehlen diese Zwischenstrecke für Veranstaltungen mit erfahrenen Rennfahrern und größeren physischen Räumen. Es handelt sich um eine physische 1:1 -Reproduktion der virtuellen Strecke, die in der Konsole verfügbar ist. Es bietet Rennfahrern die Möglichkeit, ein Modell in einer virtuellen Umgebung zu trainieren und das Modell dann auf einem physischen DeepRacer AWS-Gerät für autonome Rennen auf einer physischen Strecke bereitzustellen.

Um Ihren eigenen AWS DeepRacer Smile Speedway (Intermediate) -Track zu drucken oder zu erstellen, laden Sie diese AWS DeepRacer Smile Speedway (Intermediate) -Track-Datei herunter.

Streckenvorlage für AWS DeepRacer RL Speedway (Advanced)

Die Strecke AWS DeepRacer RL Speedway (Advanced) (auch bekannt als AWS DeepRacer Summit Speedway) wurde ursprünglich für DeepRacer AWS-Gipfeltreffen im Jahr 2022 veröffentlicht und ist

die längste physische Strecke in der Geschichte von AWS. DeepRacer Es ist vorgedruckt bei AWS DeepRacer Storefront erhältlich.

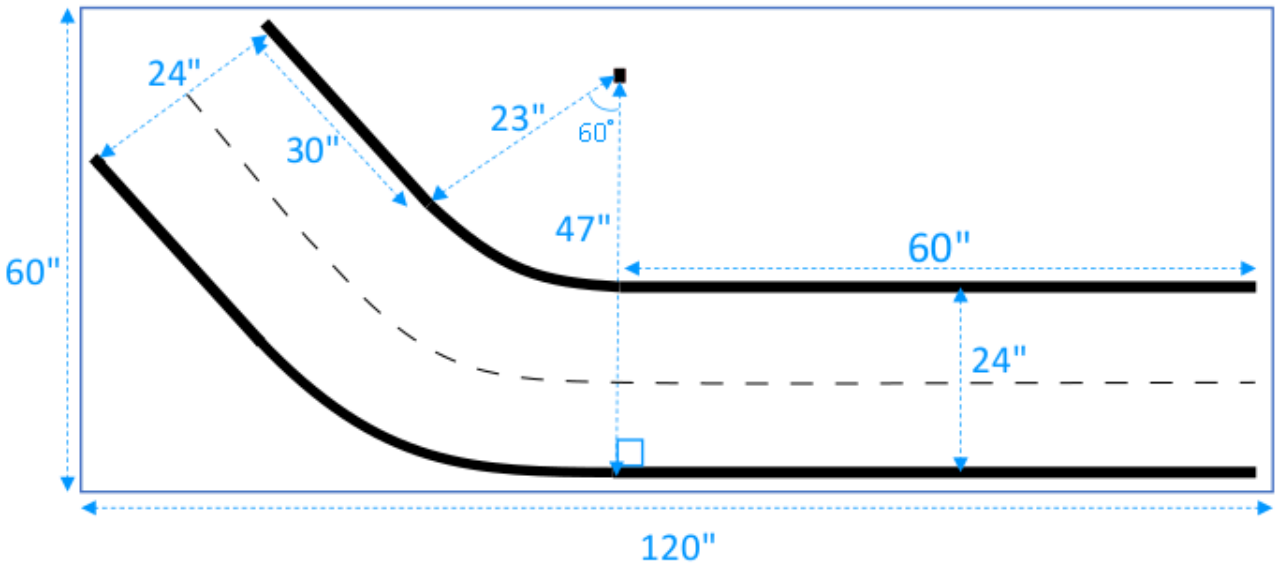


Wir empfehlen die Strecke AWS DeepRacer RL Speedway (Advanced) für Veranstaltungen mit erfahrenen Rennfahrern. Sie bietet eine überzeugende Herausforderung für Rennfahrer, die es genießen, auf Geraden schnell zu fahren. Die AWS DeepRacer RL Speedway (Advanced) -Strecke ist eine physische 1:1 -Reproduktion der virtuellen Strecke, die in der Konsole verfügbar ist. Es bietet Rennfahrern die Möglichkeit, ein Modell in einer virtuellen Umgebung zu trainieren und das Modell dann auf einem physischen DeepRacer AWS-Gerät für autonome Rennen auf einer physischen Strecke bereitzustellen.

Um Ihre eigene AWS RL Speedway (Advanced) -Strecke zu drucken oder zu erstellen, laden Sie diese AWS DeepRacer RL Speedway (Advanced) -Track-Datei herunter.

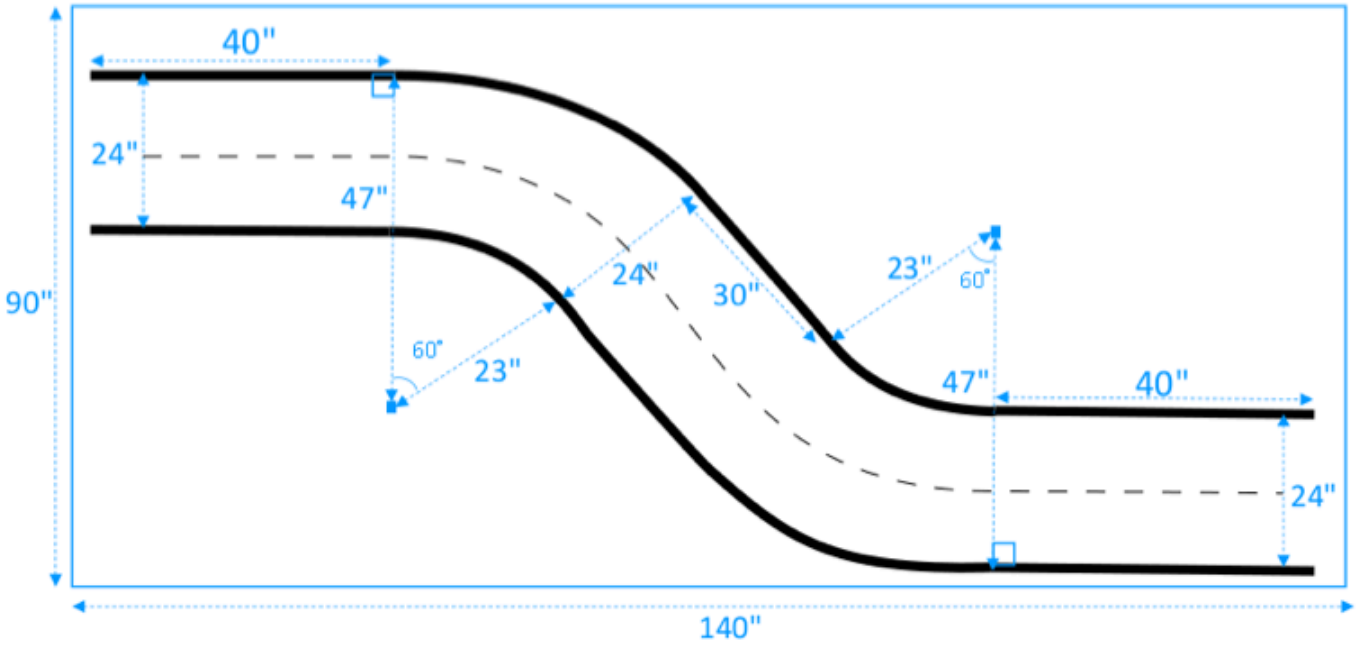
AWS-Vorlage für einen DeepRacer Single-Turn-Track

Diese Basis-Streckenvorlage besteht aus zwei geraden Streckensegmenten, die durch ein Kurven-Streckensegment verbunden sind. Modelle, die mit dieser Strecke trainiert wurden, sollten Ihr DeepRacer AWS-Fahrzeug geradeaus fahren oder in eine Richtung abbiegen lassen.



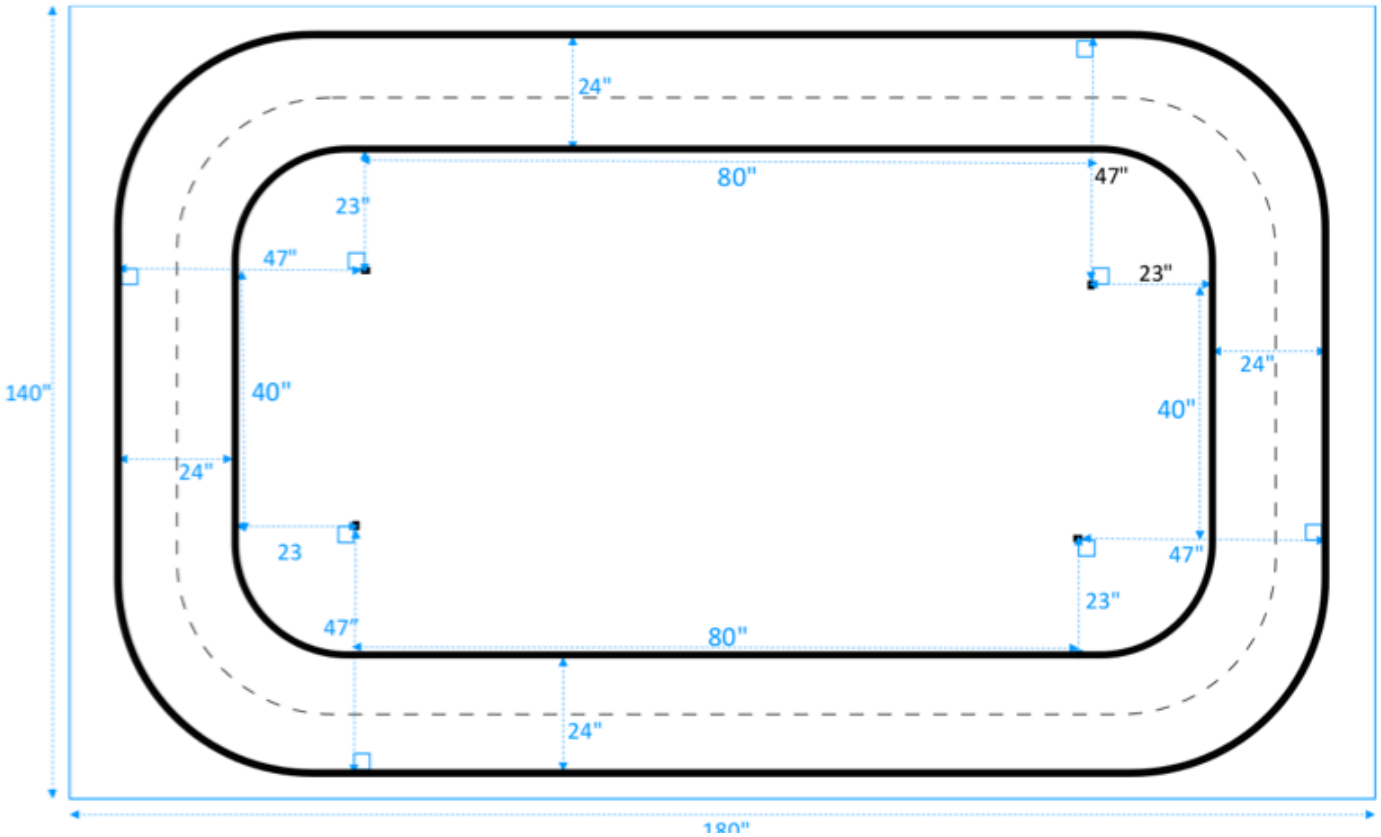
Vorlage AWS DeepRacer AWS-S-Kurvenbahnen

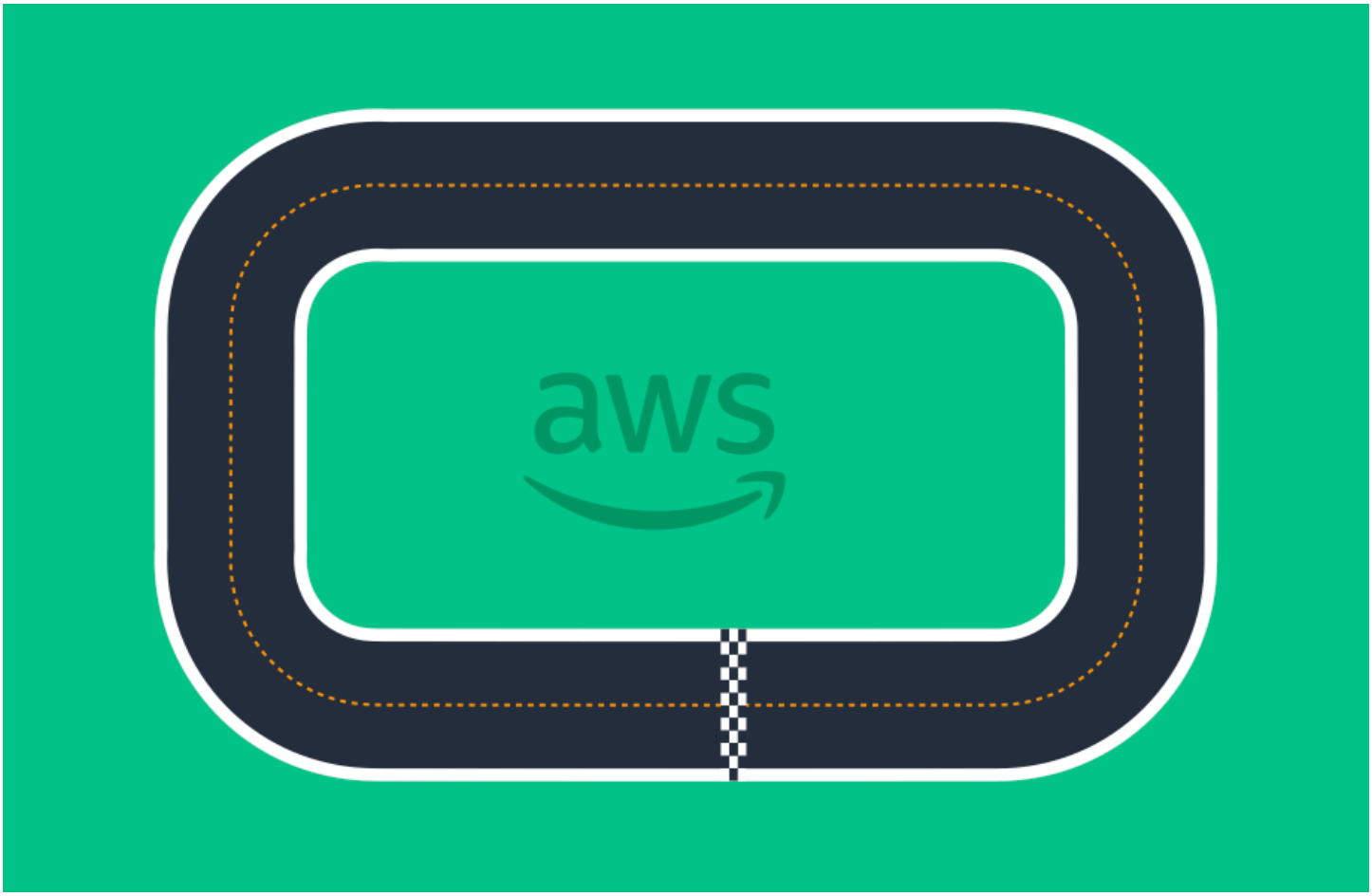
Die Strecke ist komplexer als die mit einer Kurve, da das Modell lernen muss, Kurven in zwei Richtungen zu fahren. Sie können die Bauanleitung für eine Strecke mit einer Kurve leicht auf diese Strecke erweitern, indem Sie nach der ersten Kurve eine Kurve in die entgegengesetzte Richtung einbauen.



DeepRacer AWS-Loop-Track-Vorlage

Dieser normale Rundkurs besteht aus wiederkehrenden Kurven im Winkel von 90 Grad. Für die Verlegung des gesamten Gleises ist eine größere Umfassungsfläche erforderlich.


24"
80"
47"
23"
47"
23"
140"
40"
24"
40"
24"
23
47"
23"
47"
80"
24"
180"


aws

Nehmen Sie an einem DeepRacer AWS-Rennen teil

Nachdem Sie Ihr Modell erfolgreich in der Simulation trainiert und bewertet haben, vergleichen Sie die Leistung Ihres Modells mit den Modellen anderer Rennfahrer, indem Sie an einem Rennen teilnehmen. Rennen sind eine unterhaltsame Möglichkeit, Feedback zu Ihrem Modell zu erhalten, Auszeichnungen und Preise zu gewinnen, andere Mitglieder der DeepRacer AWS-Community kennenzulernen, sich über Möglichkeiten zum Lernen und Verbessern Ihrer Fähigkeiten zu informieren und Spaß zu haben.

Rennen können persönlich oder online (virtuell) stattfinden und virtuelle Rennen können synchron als LIVE-Rennen oder asynchron als klassische Rennen formatiert werden. Virtuelle LIVE- und klassische Rennen können privat oder öffentlich übertragen werden.

In diesem Abschnitt wird beschrieben, wie Sie an einem AWS DeepRacer League Virtual Circuit-Rennen oder einem Community-basierten virtuellen Rennen teilnehmen und welche Formatierungsoptionen Sie haben.

Arten AWS DeepRacer AWS-Rennveranstaltungen

Eine Veranstaltung kann von ihrem Sponsor oder Veranstalter kategorisiert werden. Sowohl AWS DeepRacer League- als auch Community-Rennveranstaltungen können persönlich auf einer physischen Strecke oder online auf einer virtuellen Strecke stattfinden.

- AWS-gesponserte Rennveranstaltungen — Rennveranstaltungen, die von gesponsert AWS werden, werden als DeepRacer AWS-League-Veranstaltungen bezeichnet und stehen allen DeepRacer AWS-Benutzern offen. Erstmalige Rennfahrer können ihre Ligareise beginnen, indem sie an einem monatlichen virtuellen Rennen teilnehmen. Sobald ein Rennfahrer ein Modell für das Rennen eingereicht hat, sammelt er Punkte und erhält seine nationale und regionale Saisonwertung.

- Von der Community gesponserte Rennveranstaltungen — Rennveranstaltungen, die von DeepRacer AWS-Benutzern erstellt wurden, werden als Community-Rennveranstaltungen bezeichnet.

Teilnahme an einem online gesponserten oder von der AWS Community gesponserten Rennen

Sie können die DeepRacer AWS-Konsole verwenden, um an einem AWS DeepRacer League Virtual Circuit-Event oder einem Community-basierten Online-Rennen teilzunehmen.

- Jeder DeepRacer AWS-Benutzer kann an einem AWS DeepRacer League Virtual Circuit-Online-Rennen teilnehmen.
- Nur eingeladene Benutzer können auf virtuelle Veranstaltungen im Community-Rennen zugreifen oder an diesen teilnehmen. Benutzer werden eingeladen, wenn sie einen Einladungslink erhalten, der vom Veranstalter gesendet oder von einem anderen Rennteilnehmer weitergeleitet wird.

Themen

- the section called “ Nehmen Sie an einem Virtual Circuit-Rennen teil ”
- the section called “ Nehmen Sie an einem Community-Rennen teil ”
- the section called “ Nehmen Sie an einem LIVE-Rennen teil ”
- the section called “Terminologie für Rennveranstaltungen”

Nehmen Sie an einem AWS DeepRacer League Virtual Circuit-Rennen teil

In diesem Abschnitt erfahren Sie, wie Sie mit der DeepRacer AWS-Konsole Ihr trainiertes Modell für ein Virtual Circuit-Rennen einreichen können.

Um am AWS DeepRacer League Virtual Circuit teilzunehmen

1. Melden Sie sich bei der DeepRacer AWS-Konsole an.
2. Wählen Sie im Hauptnavigationsbereich AWS Virtual Circuit aus.
3. Wählen Sie auf der Seite AWS Virtual Circuit im Abschnitt Offene Rennen die Option Rennen eingeben aus.
4. Wenn Sie zum ersten Mal an einem AWS DeepRacer League-Rennevent teilnehmen, geben Sie Ihren Alias in Racer-Name unter AWS DeepRacer League-Rennfahrername ein.

5. Wählen Sie unter Modell auswählen das Modell, das Sie verwenden möchten, aus der Modellliste aus. Vergewissern Sie sich, dass Ihr Modell für den Umgang mit der Schienenform trainiert wurde.
6. Wenn Sie zum ersten Mal an einem AWS DeepRacer League-Event teilnehmen, wählen Sie unter Liga-Anforderungen Ihr Wohnsitzland aus. Sobald Sie Ihr Wohnsitzland ausgewählt und Ihr erstes Modell eingereicht haben, ist es für die Rennsaison gesperrt und wird bei der Preisverleihung überprüft. Akzeptieren Sie anschließend die Allgemeinen Geschäftsbedingungen, indem Sie das Kästchen anklicken.
7. Wählen Sie An Rennen teilnehmen, um die Einreichung abzuschließen. Die Einreichungsquote für jedes Rennen beträgt 50.

   Nachdem Ihr Modell eingereicht wurde, beginnt die DeepRacer AWS-Konsole mit der Evaluierung. Der Vorgang kann bis zu 10 Minuten dauern.
8. Überprüfen Sie auf der Rennseite die Renndetails.
9. Notieren Sie sich auf der Rennseite Ihren Einreichungsstatus unter dem Namen Ihres Rennfahrers.
10. Sehen Sie sich auf der Rennseite die Rangliste in der Bestenliste an, um zu sehen, wie Ihr Modell im Vergleich zu anderen abschneidet.

    Wenn Ihr Modell die Strecke in drei aufeinanderfolgenden Prüfungen nicht beendet, wird es nicht in der Rangliste der Bestenliste aufgeführt. Ihr Ranking in der Bestenliste spiegelt Ihren Beitrag mit der besten Leistung wider. Außerdem erhältst du eine nationale und regionale Saisonwertung, anhand derer du einschätzen kannst, wo du im Vergleich zu anderen Rennfahrern in deinem Land und deiner Region abschneidest.

    Nachdem Sie ein Modell eingereicht haben, versuchen Sie, dessen Leistung zu verbessern, indem Sie Ihre Belohnungsfunktion verfeinern und an Ihrem Modell weiterarbeiten. Sie können ein neues Modell auch mit einem anderen Algorithmus oder Aktionsraum trainieren. Lerne, passe dich an und fahre erneut Rennen, um deine Chancen auf Belohnungen zu erhöhen.

Um an einem DeepRacer AWS-Community-Rennen teilzunehmen

> **Note**
>
> Um an einem DeepRacer AWS-Community-Rennen teilzunehmen, benötigen Sie zunächst vom Rennorganisator einen Link zum Rennen.

Wenn Sie eine Einladung zu einem DeepRacer AWS-Rennen erhalten, finden Sie heraus, ob es sich um ein LIVE - oder ein klassisches Rennen handelt.

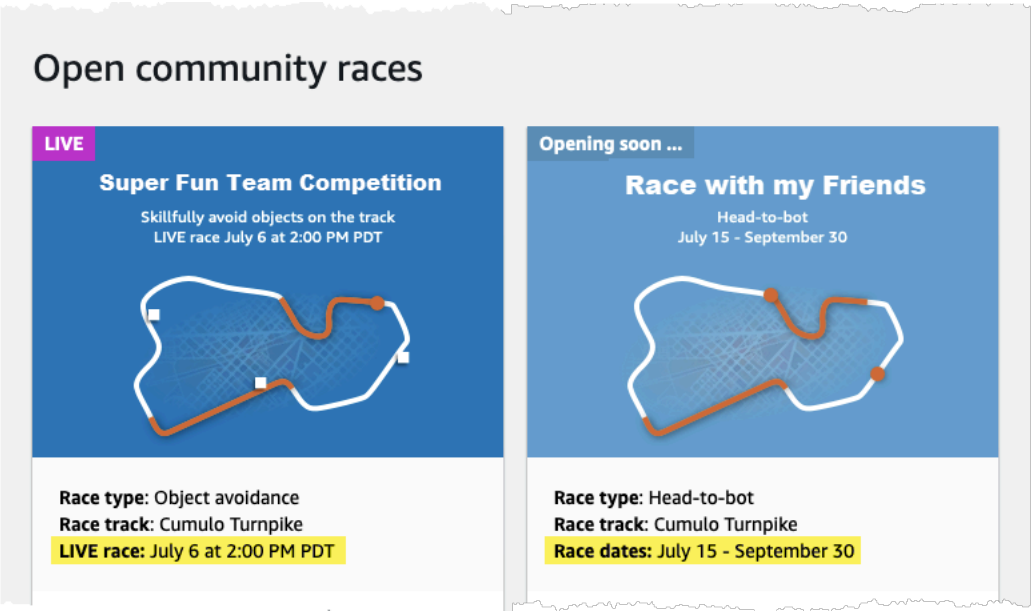
Klassisches Rennen

Klassische Rennen sind asynchrone Ereignisse, für die keine Interaktion in Echtzeit erforderlich ist. Über Ihren Einladungslink können Sie ein Modell für das Rennen einreichen und sich die Bestenliste ansehen. Du kannst zu jeder Zeit innerhalb der Eröffnungs- und Schlusstermine des Rennens unbegrenzt viele Modelle einreichen, um deine beste Platzierung in der Bestenliste zu erreichen. Ergebnisse und Videos für klassische Rennen können für eingereichte Models auf der Bestenlisten-Seite eingesehen werden, sobald das Rennen gestartet wird. Alle klassischen Rennen sind private Veranstaltungen.

LIVE-Rennen

LIVE-Rennen sind Rennveranstaltungen in Echtzeit, bei denen Sie sich virtuell mit anderen Rennfahrern treffen, die sich abwechseln, um um die schnellste Zeit in der Bestenliste zu kämpfen. Sie können mehrere Modelle einreichen, es wird jedoch nur das Modell verwendet, das Sie zuletzt eingereicht haben, bevor das Einreichungsfenster geschlossen wird. Während Ihres Rennens haben Sie die Möglichkeit, interaktive Geschwindigkeitssteuerungen auszuprobieren, die vorübergehend die Geschwindigkeitsparameter Ihres Modells außer Kraft setzen und Ihnen die Möglichkeit geben, strategische Anpassungen in Echtzeit vorzunehmen. LIVE-Rennen können privat zwischen eingeladenen Rennfahrern oder öffentlich übertragen werden, damit jeder sie sehen kann.

Wenn das Wettbewerbsformat in Ihrer Einladung nicht angegeben ist, überprüfen Sie Ihre Rennkarte. Bei LIVE-Rennen steht „LIVE“ und Sie erhalten Datum und Uhrzeit der synchronen Veranstaltung. Klassische Rennen geben dir den Zeitraum für den asynchronen Wettkampf.



Nehmen Sie als Rennteilnehmer an einem DeepRacer AWS-Community-Rennen teil

Wenn Sie neu bei einem AWS-Community-Rennen sind AWS und eine Einladung erhalten, an einem DeepRacer AWS-Community-Rennen teilzunehmen, folgen Sie den Schritten unter So nehmen Sie als neuer Benutzer teil. Wenn Sie zu einem aktiven Community-Rennen eingeladen wurden und schon einmal an einem DeepRacer AWS-Rennen teilgenommen haben, folgen Sie den Schritten unten unter So nehmen Sie an einem Classic-Rennen teil oder Nehmen Sie an einem LIVE-Rennen teil, je nach Ihrem Wettbewerbsformat.

Um als neuer Benutzer teilzunehmen

Wenn Sie neu bei AWS sind AWS und einen Einladungslink zur Teilnahme an einem DeepRacer AWS-Community-Rennen erhalten, wählen Sie den Link, um zur DeepRacer AWS-Konsole zu gelangen, und registrieren Sie sich dann für ein AWS Konto, bevor Sie am Rennen teilnehmen.

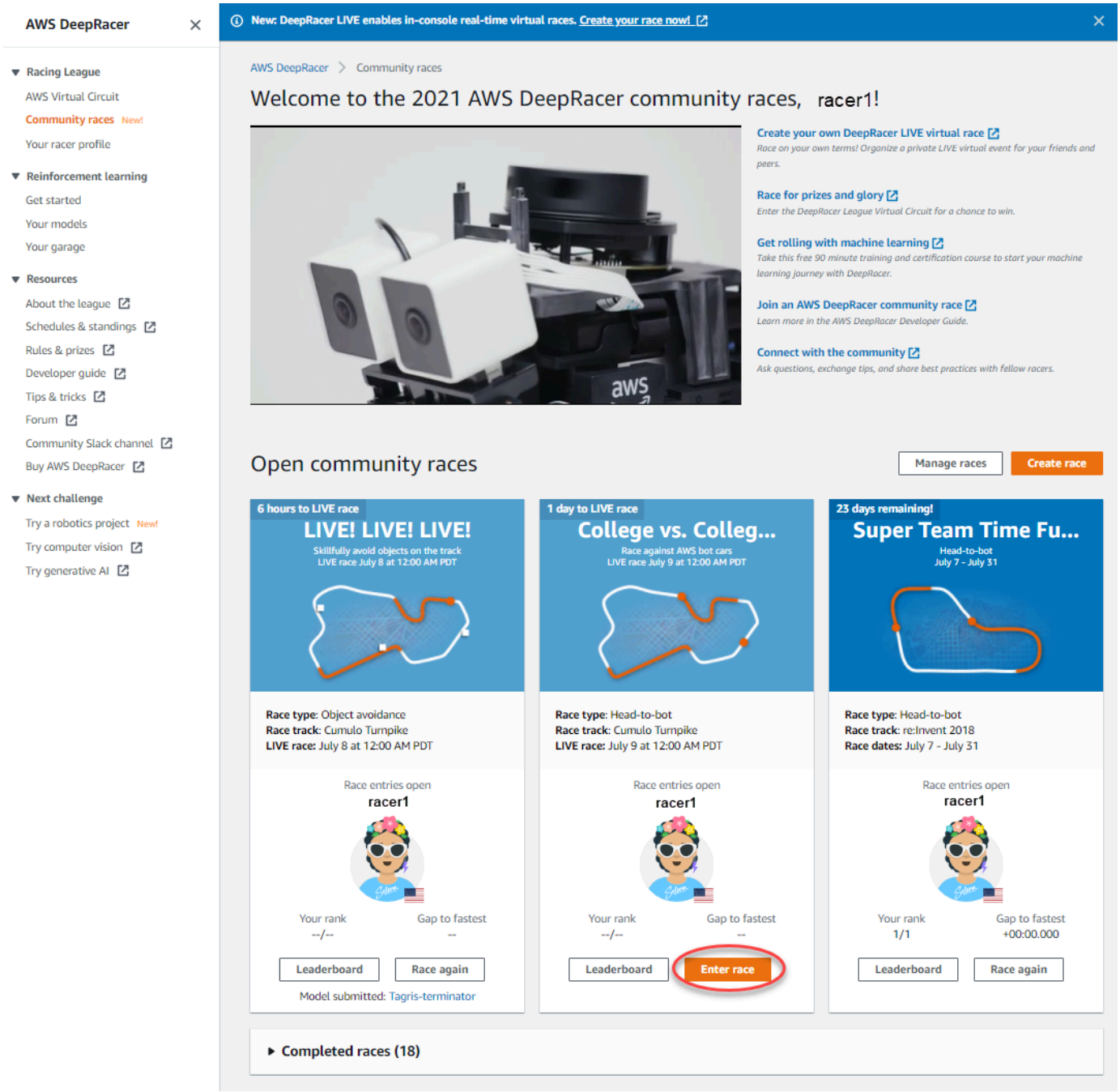
Als neuer DeepRacer AWS-Benutzer oder als Erstteilnehmer an einem DeepRacer AWS-Rennen folgen Sie den Schritten, um an einem Community-Rennen in der DeepRacer AWS-Konsole teilzunehmen.

Um als neuer Benutzer an einem Rennen teilzunehmen

1. Erstellen Sie ein AWS Konto in der DeepRacer AWS-Konsole.

2. Sobald Sie eingerichtet und angemeldet sind, wählen Sie den Link, den Ihnen der Rennorganisator mitgeteilt hat, um die Rennseite zu öffnen.
3. Wenn Sie aufgefordert werden, einen DeepRacer AWS-Rennfahrernamen zu erstellen, geben Sie einen Namen ein, den Sie als Identifikation in allen DeepRacer AWS-Bestenlisten verwenden werden. Sobald Sie einen Rennfahrernamen ausgewählt haben, können Sie ihn nicht mehr ändern.
4. Erweitern Sie auf der Seite mit den Renndetails die Option Los geht's mit dem Rennen.
5. Wählen Sie Erste Schritte mit RL, um eine kurze Einführung in die Schulung eines DeepRacer AWS-Modells für autonomes Fahren zu erhalten.
6. Trainieren und evaluieren Sie Ihr Modell für das Rennen in der DeepRacer AWS-Konsole.

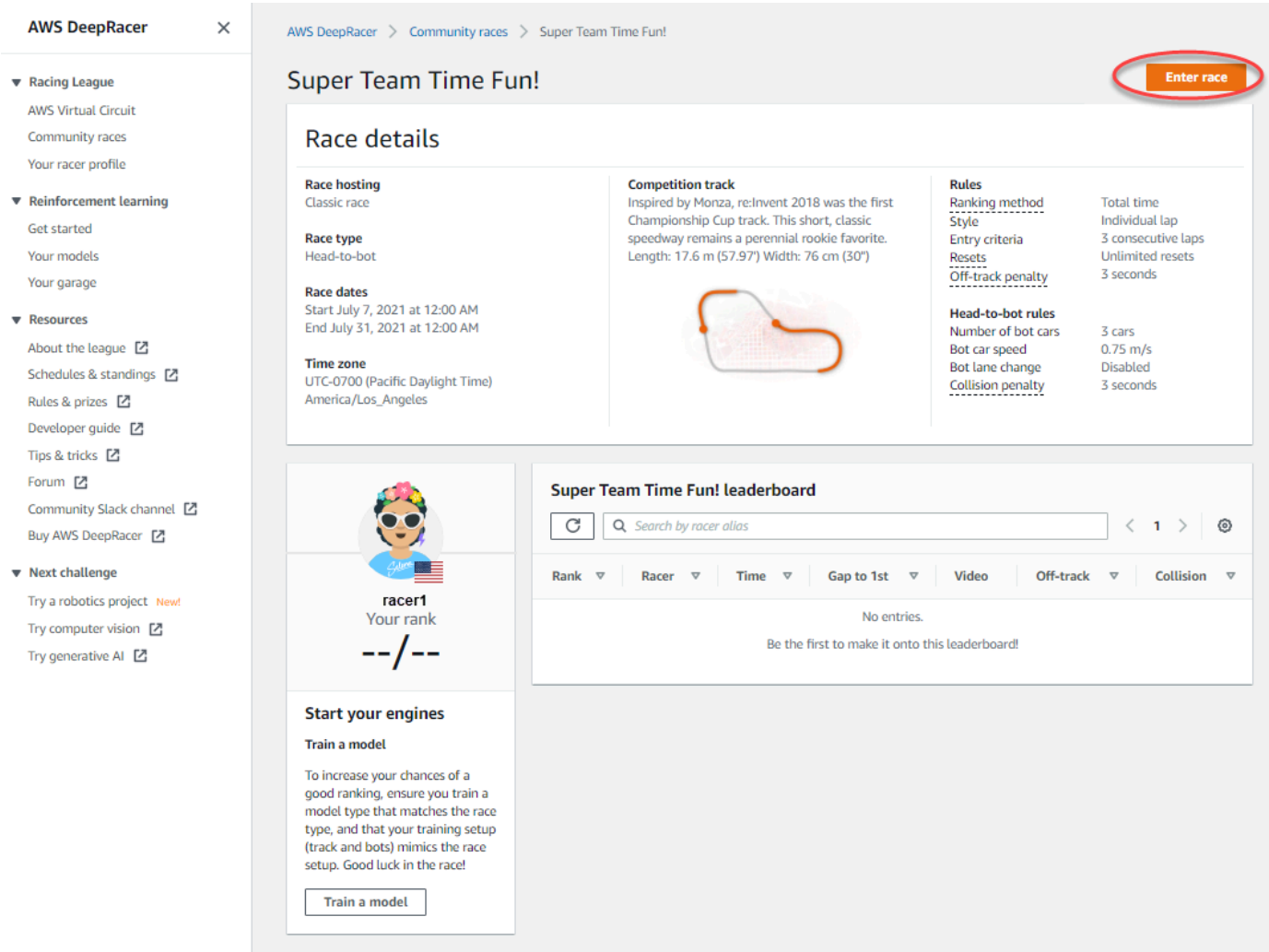
   Weitere Informationen zur Schulung Ihres Modells finden Sie unter Trainieren Sie Ihr erstes DeepRacer AWS-Modell .
7. Navigieren Sie zu Community-Rennen.
8. Finde das Rennen, zu dem du eingeladen bist. Wähle auf der Rennkarte die Option An Rennen teilnehmen aus.



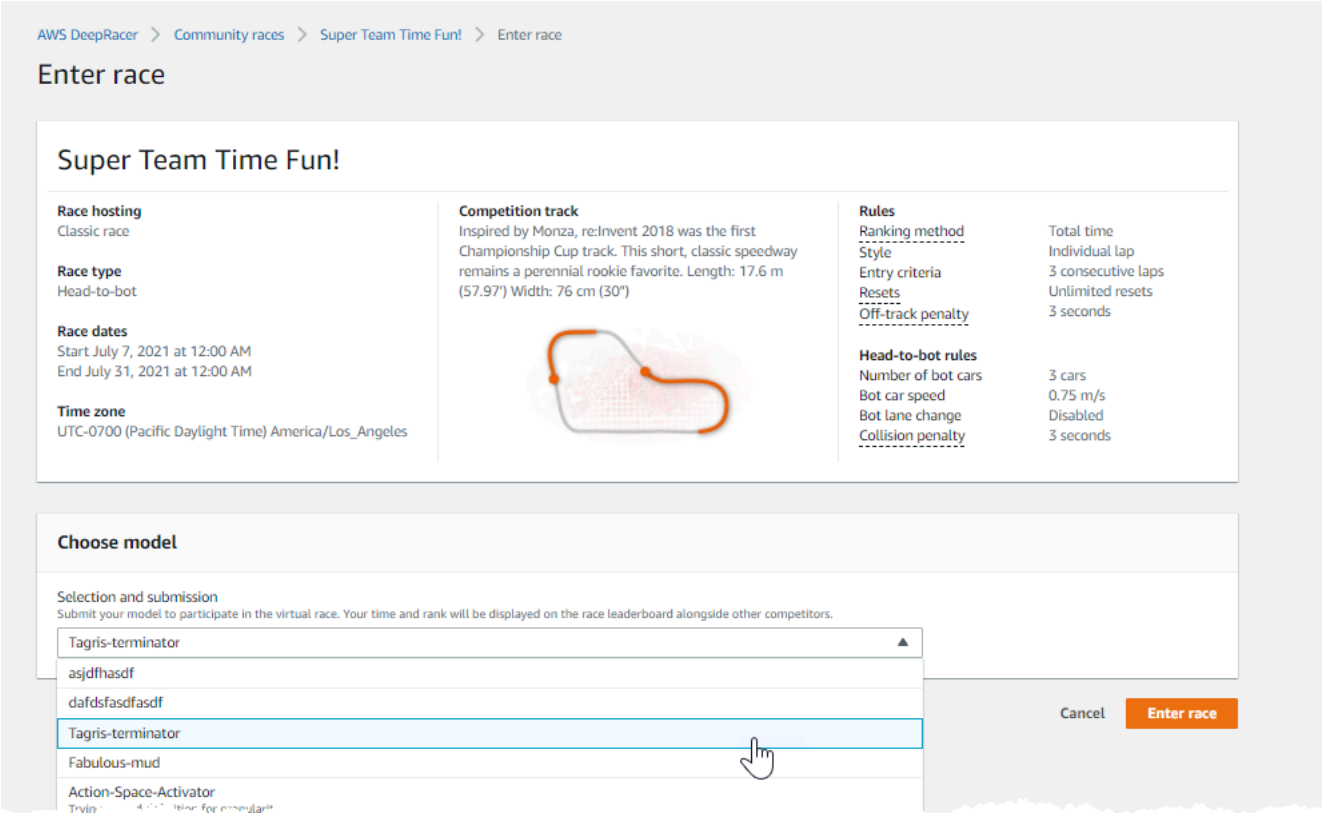
9. Folgen Sie den Anweisungen unter So nehmen Sie an einem Classic-Rennen teil oder Nehmen Sie an einem LIVE-Rennen teil, je nachdem, wie es für das Wettbewerbsformat Ihres Rennens angemessen ist.

Um an einem klassischen Rennen teilzunehmen

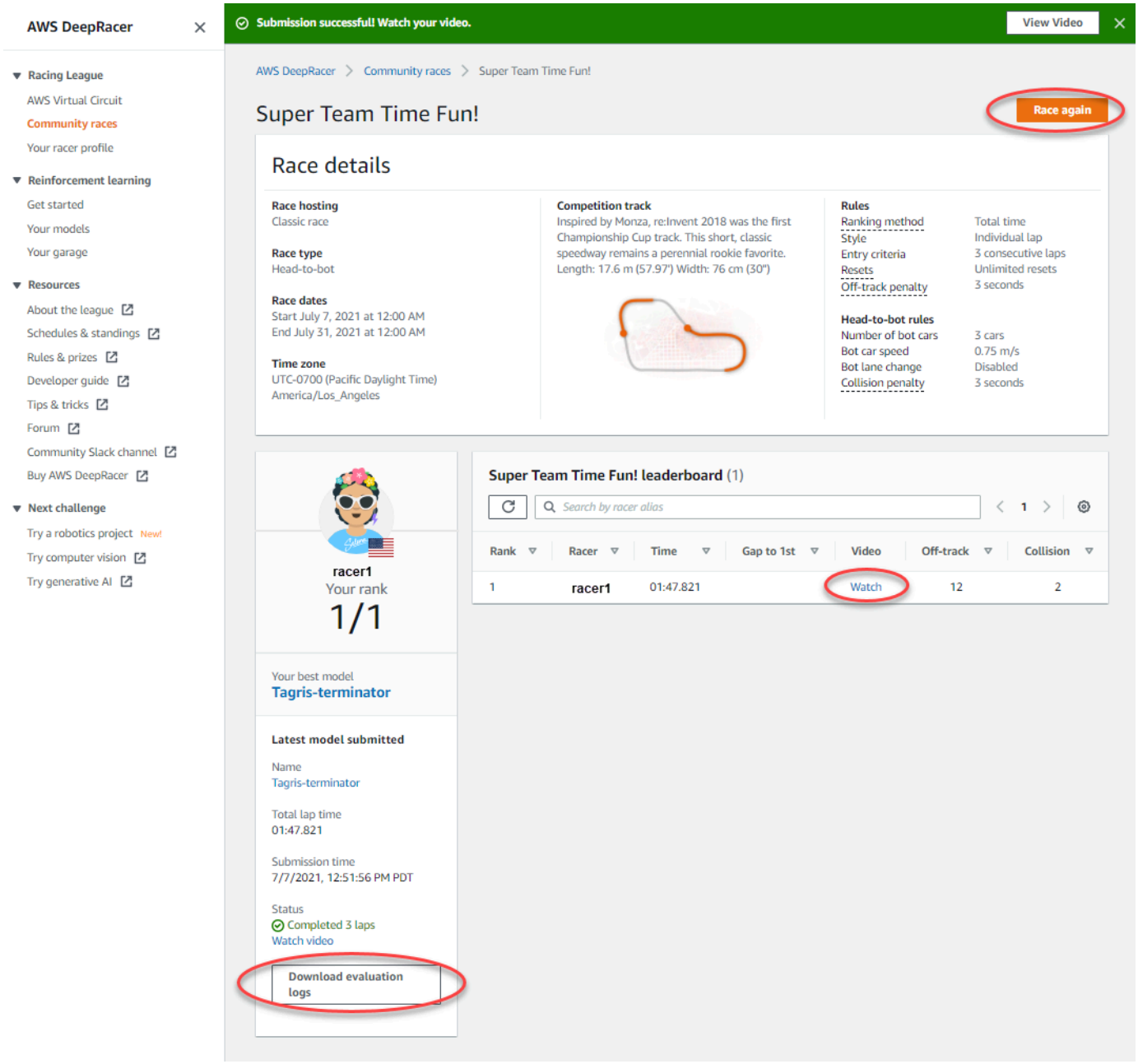
1. Wählen Sie den Link aus, den Sie vom Rennorganisator erhalten haben. Wenn Sie in der DeepRacer AWS-Konsole noch nicht mit Ihrem Konto angemeldet sind, werden Sie aufgefordert, sich anzumelden.
2. Sobald Sie sich bei der DeepRacer AWS-Konsole angemeldet haben, gelangen Sie über den Link zur Rennseite. Auf der Rennseite werden die Renndetails, die Bestenliste und Ihre Rennfahrerinformationen angezeigt. Wähle „An Rennen teilnehmen“.



3. Wählen Sie auf der Seite „Rennen starten“ unter Modell auswählen ein trainiertes Modell aus und klicken Sie dann auf Rennen teilnehmen.



4. Wenn Ihr Modell anhand der Regenerieren erfolgreich evaluiert wurde, sehen Sie sich die Bestenliste der Veranstaltung an, um zu sehen, wo Ihr Modell gegenüber anderen Teilnehmern steht.
5. Wählen Sie optional „Ansehen“, um sich ein Video über die Leistung Ihres Fahrzeugs anzusehen, oder „Bewertungsprotokolle herunterladen“, um einen detaillierten Überblick über die erzielten Ergebnisse zu erhalten.

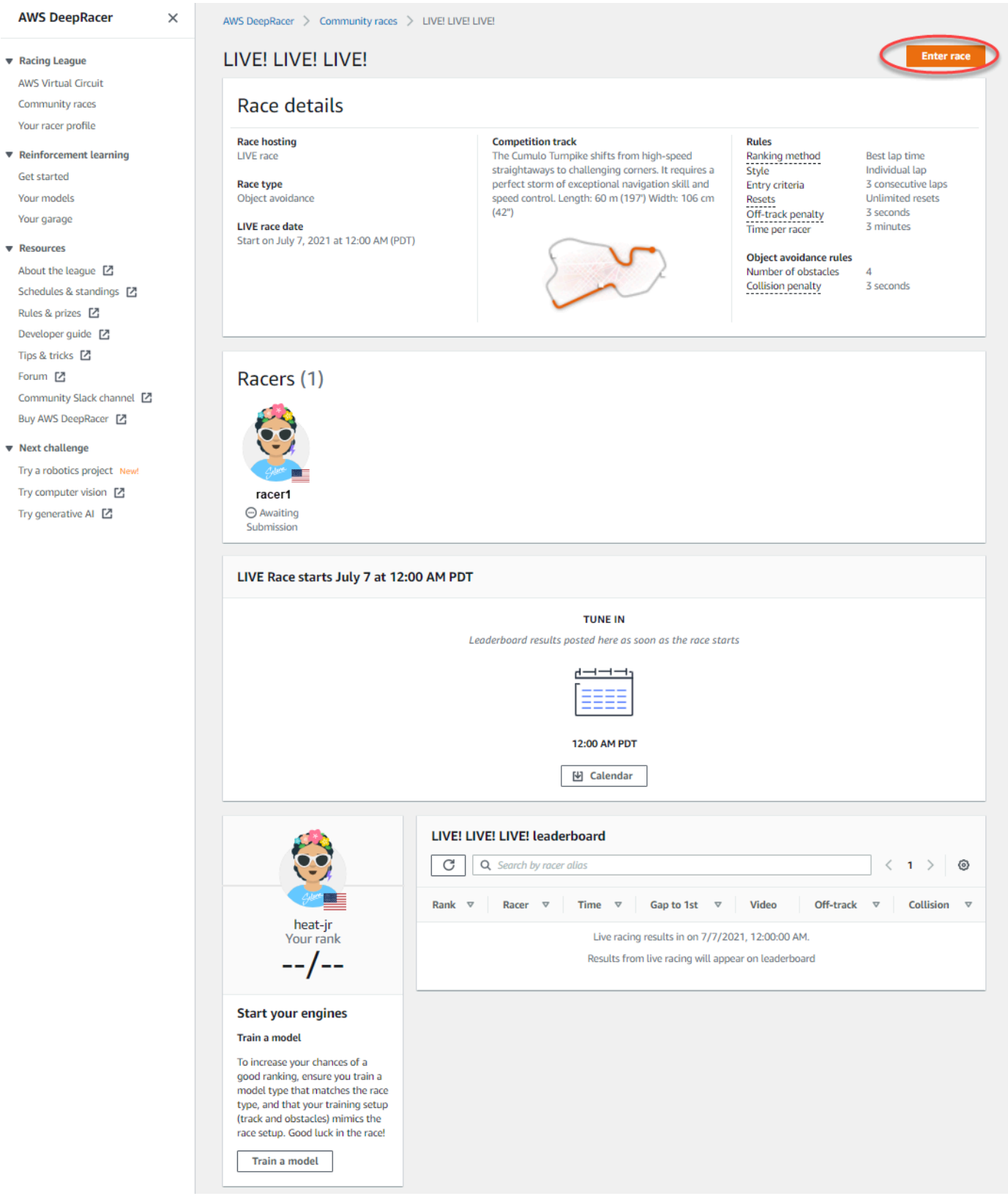


6. Wählen Sie erneut Rennen, um ein anderes Modell einzugeben. Du kannst innerhalb der Eröffnungs- und Schlusstermine des Rennens jederzeit eine unbegrenzte Anzahl von Modellen einreichen, um deine beste Platzierung in der Bestenliste zu erreichen.

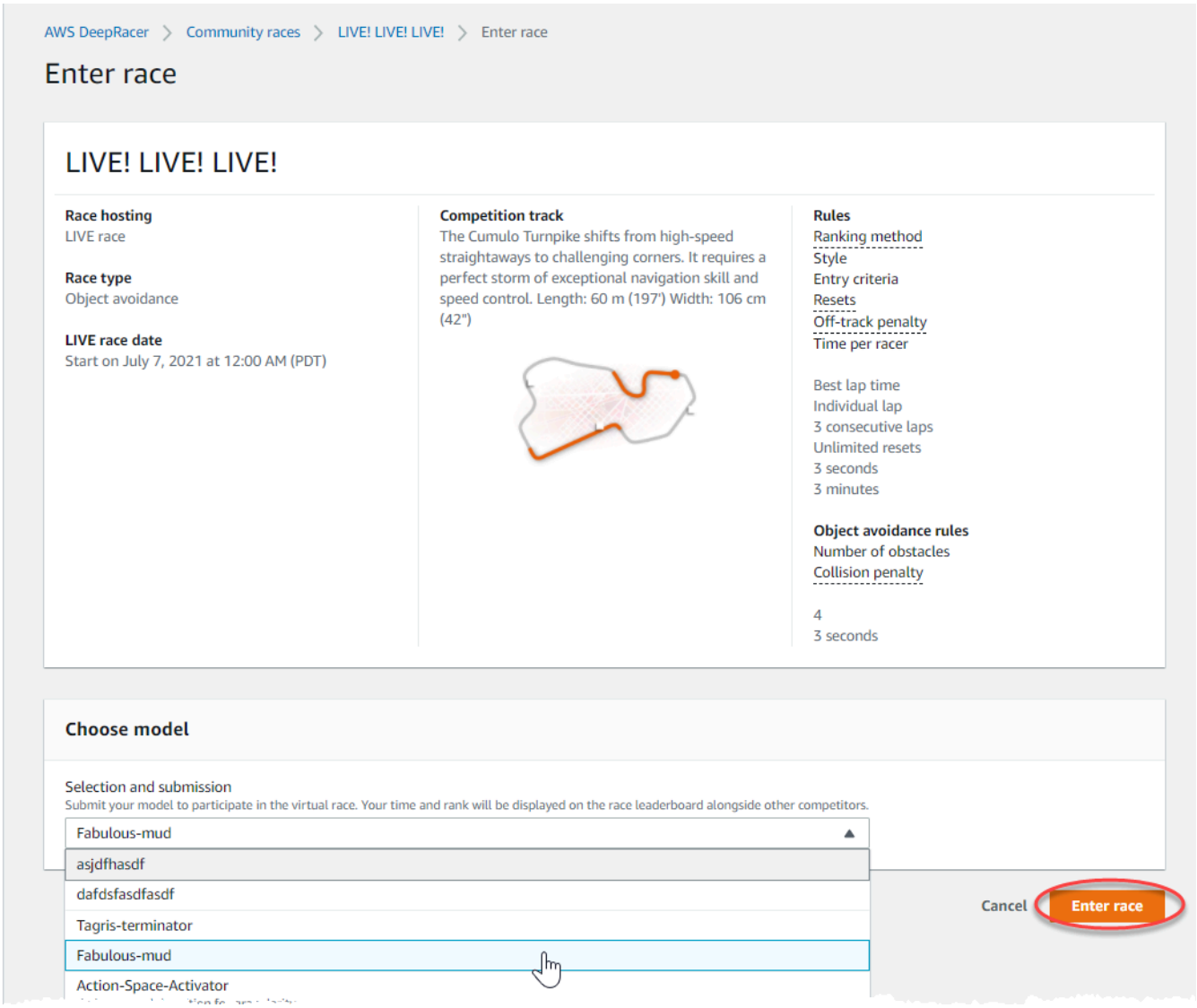
Um an einem LIVE-Rennen teilzunehmen

1. Wählen Sie den Link aus, den Sie vom Rennorganisator erhalten haben. Wenn Sie in der DeepRacer AWS-Konsole noch nicht mit Ihrem Konto angemeldet sind, werden Sie aufgefordert, sich anzumelden.

2. Sobald Sie sich bei der DeepRacer AWS-Konsole angemeldet haben, gelangen Sie über den Link zur Rennseite. Auf der Rennseite werden die Renndetails und die Bestenliste angezeigt. Wähle „An Rennen teilnehmen“.



3. Wählen Sie auf der Seite „Rennen starten“ unter Modell auswählen ein trainiertes Modell aus und klicken Sie dann auf Rennen teilnehmen.



4. Wenn Ihr Modell anhand der Regenerieren erfolgreich evaluiert wurde, sehen Sie sich die Bestenliste der Veranstaltung an, um zu sehen, wo Ihr Modell gegenüber anderen Teilnehmern steht.
5. Wählen Sie für LIVE-Rennen optional Kalender aus, um das LIVE-Rennereignis zu Ihrem Kalender hinzuzufügen.
6. Wählen Sie erneut Rennen, um ein anderes Modell einzugeben. Sie können mehrere Modelle einreichen, es wird jedoch nur das Modell verwendet, das Sie zuletzt eingereicht haben, bevor das Einreichungsfenster geschlossen wird.

Nehmen Sie an einem AWS DeepRacer LIVE-Rennen teil

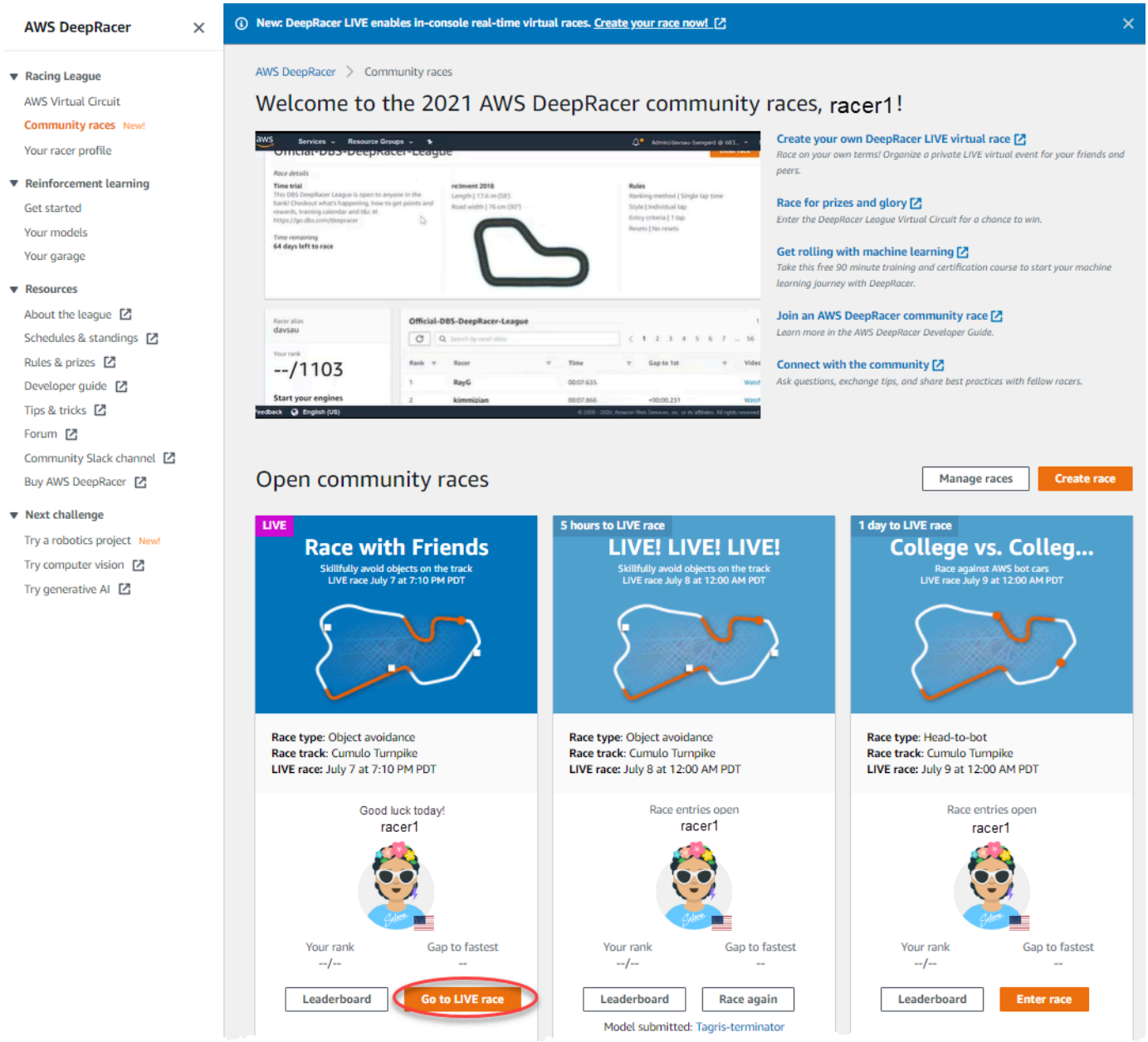
> **Note**
>
> Reichen Sie Ihr Modell mindestens eine Stunde vor Beginn des LIVE-Rennens ein. Sie können mehrere Modelle einreichen, es wird jedoch nur das letzte Modell verwendet, das Sie vor dem Ende des Einreichungsfensters eingereicht haben.

Bevor Sie beginnen

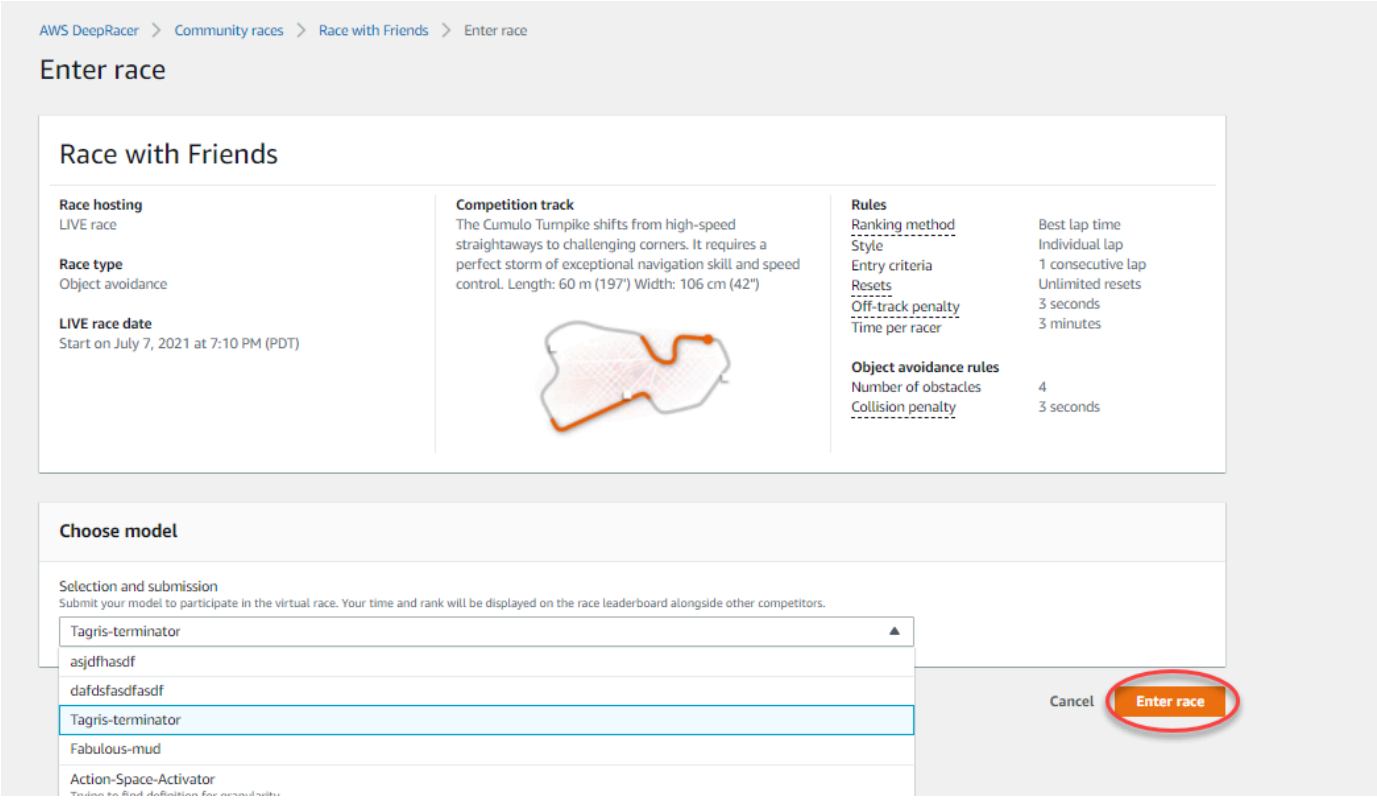
- Verwenden Sie einen Chrome- oder Firefox-Browser (stellen Sie sicher, dass Ihr Browser auf dem neuesten Stand ist).
- Trennen Sie das virtuelle private Netzwerk (VPN), falls Sie eines verwenden.
- Schließen Sie alle zusätzlichen Tabs.

Um an einem LIVE-Rennen teilzunehmen

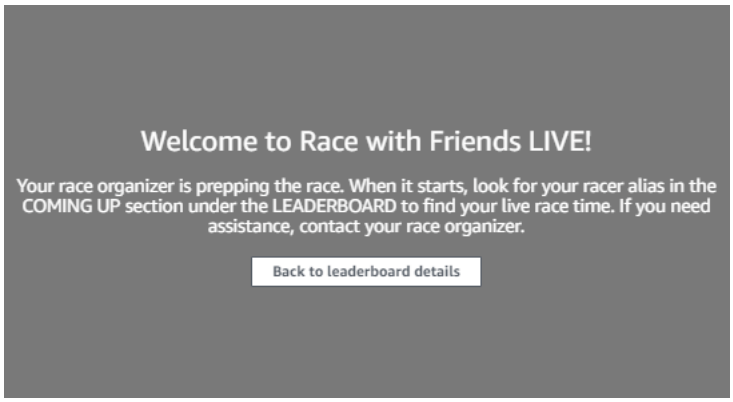
1. Melden Sie sich bei der DeepRacer AWS-Konsole an.
2. Wenn Sie kein Modell eingereicht haben, suchen Sie die Rennkarte für das Rennen, an dem Sie teilnehmen möchten, und wählen Sie Gehe zum LIVE-Rennen aus.



3. Wähle auf der Seite Rennen die Option An Rennen teilnehmen aus.
4. Wählen Sie auf der Seite „An Rennen teilnehmen“ unter Modell auswählen aus dem Drop-down-Menü das Modell aus, das Sie einreichen möchten, und wählen Sie „Rennen starten“ aus.



5. Wähle auf der Seite Rennen die Option Gehe zum LIVE-Rennen.
6. Auf der Seite mit dem LIVE-Rennen siehst du eine Wartemeldung. Navigiere zu der Konferenzbrücke, die dir dein Rennorganisator zur Verfügung gestellt hat.



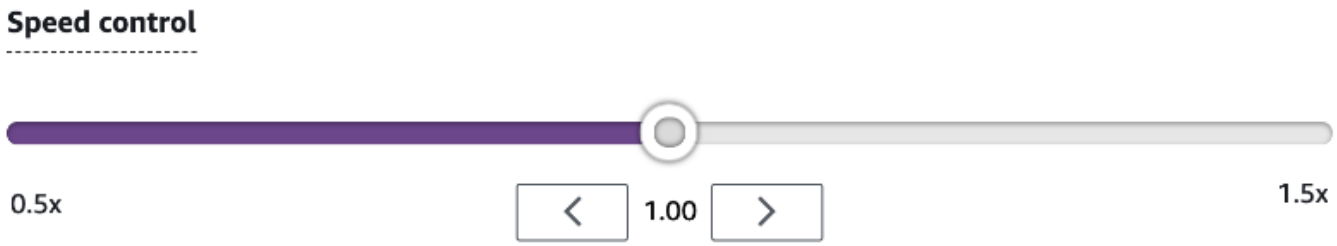
7. Erkundigen Sie sich bei Ihrem Rennorganisator, der die Rennregeln überprüft und Fragen der Rennfahrer beantwortet.
8. Unter BESTENLISTE findest du unter BESTENLISTE deine Live-Rennzeit und sei bereit, wenn der Rennorganisator bekannt gibt, dass du als Nächstes dabei bist.
9. Wenn du an der Reihe bist, gibt es 10, 9, 8, 7, 6... Der Countdown wird in der Konsole animiert, wenn der Rennorganisator dein Rennen startet. Auf geht's! Sie werden Zugriff auf die optionale Geschwindigkeitssteuerung haben. Um wichtige Momente auszuwählen, um die

Geschwindigkeit Ihres Modells zu erhöhen oder zu verlangsamen. Es gibt drei Möglichkeiten, die Geschwindigkeitsregelungsfunktion zu bedienen:

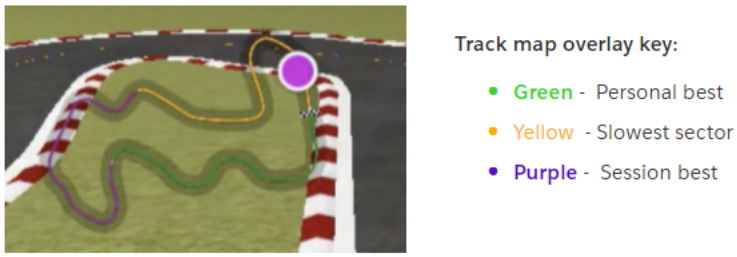
a. Ziehen Sie den Schieberegler mit der Maus Ihres Computers.

b. Wählen Sie alternativ die Pfeiltasten **<**/**>** in der Konsole.

c. Sie können auch den Schieberegler auswählen, um den Schieberegler zu aktivieren, und dann Ihre Pfeiltasten **#** und die **#** Tastatur verwenden.



10. Setzen Sie den Multiplikator auf 1 zurück, um wieder die Geschwindigkeitsparameter Ihres Modells zu verwenden.
11. Schauen Sie sich während des Rennens das Video-Overlay Ihres LIVE-Rennens an, um Ihre Leistung zu optimieren. Das Streckenkarten-Overlay ist in drei Sektoren unterteilt, deren Farbe sich je nach Tempo ändert. Grün steht für den Abschnitt der Strecke, in dem Sie Ihre persönliche Bestleistung erzielt haben, Gelb steht für den am langsamsten gefahrenen Sektor und Violett für die beste Trainingseinheit. Außerdem findest du Statistiken mit Angaben zu deiner besten Rundenzeit, deiner verbleibenden Geschwindigkeit in m/s, zu Resets und zur aktuellen Rundenzeit.



12. Das Rennen endet, wenn Sie das Zielflaggensymbol in der Konsole sehen. Die Geschwindigkeitssteuerung ist deaktiviert und eine Wiederholung Ihres Rennens wird auf dem Videobildschirm angezeigt. Sie werden in der Bestenliste anhand Ihrer besten Rundenzeit eingestuft.

Organisieren Sie ein DeepRacer AWS-Community-Rennen

Community-Rennen sind Rennen, die von DeepRacer AWS-Benutzern organisiert werden, die nicht offiziell von gesponsert werden AWS.

Sie können Ihr eigenes Community-Rennen erstellen und Ihre Kollegen, Klassenkameraden oder Freunde einladen, indem Sie einen Link zur Einladung zu einem Rennen teilen.

Wenn du ein Rennen für Schüler organisieren möchtest, sieh dir das anTools für Lehrkräfte für AWS-Studierende DeepRacer .

Themen

- the section called “Erstelle einen Schnellstart für ein Rennen”
- the section called “Passe ein Rennen an”
- the section called “Führen Sie ein LIVE-Rennen durch”
- the section called “Verwalte ein Rennen”
- the section called “Terminologie für Rennveranstaltungen”

Erstellen Sie ein virtuelles Community-Rennen: eine Kurzanleitung

Mit den Standardeinstellungen für Community-Rennen kannst du schnell ein virtuelles Rennen einrichten. Wenn du bereit bist, mehr über all deine Optionen zu erfahren, gehe zuthe section called “Passe ein Rennen an”.

Bevor du ein virtuelles Rennen erstellst, solltest du dir überlegen, ob ein Classic - oder ein LIVE-Rennen am besten zu deiner Gruppe passt. Falls du dich für ein LIVE-Rennen entscheidest, ob du es privat oder öffentlich teilst?

Klassisches Rennen

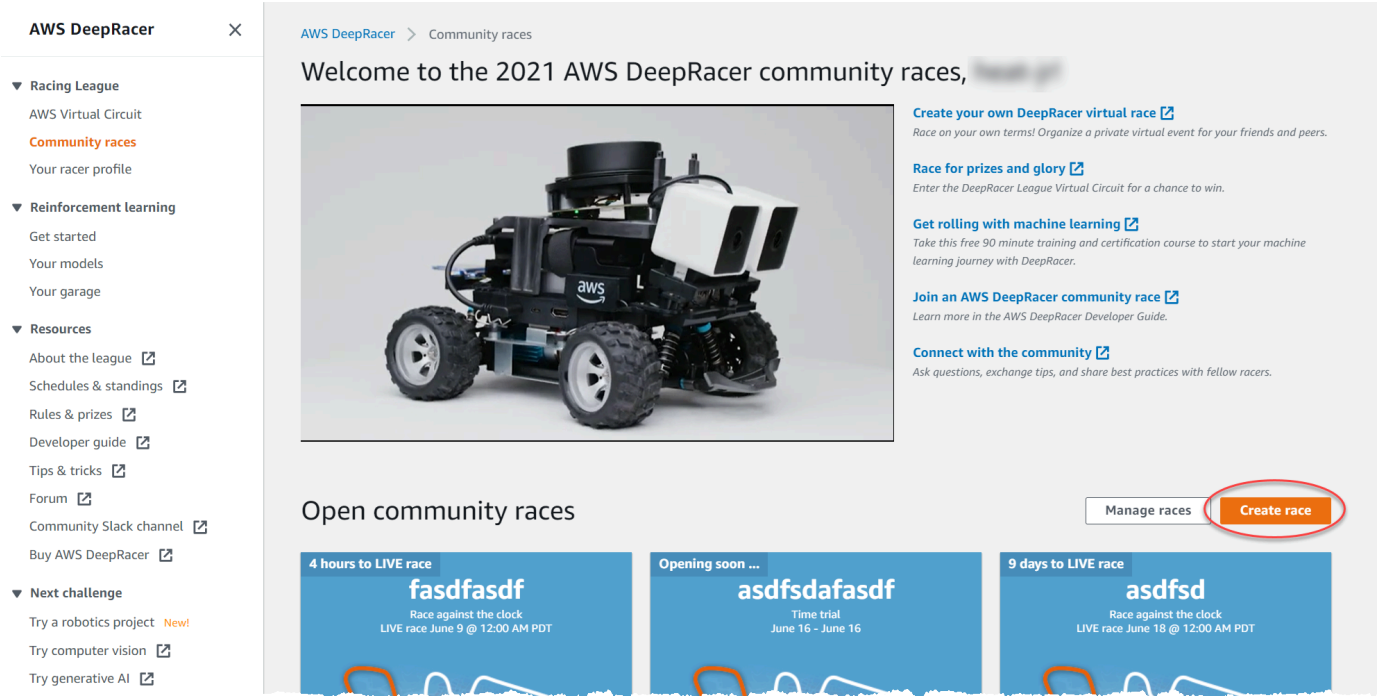
> Klassische Rennen sind asynchrone Ereignisse, für die keine Interaktion in Echtzeit erforderlich ist. Die Teilnehmer müssen einen Einladungslink erhalten, um ein Modell für das Rennen einzureichen und sich die Bestenliste anzusehen. Rennfahrer können innerhalb eines bestimmten Zeitraums jederzeit eine unbegrenzte Anzahl von Modellen einreichen, um in der Bestenliste nach oben zu klettern. Geschwindigkeitskontrollen sind nicht verfügbar. Ergebnisse und Videos für klassische Rennen können für eingereichte Models auf der Bestenlisten-Seite eingesehen werden, sobald das Rennen gestartet wird. Alle klassischen Rennen sind private Veranstaltungen.

LIVE-Rennen

LIVE-Rennen sind synchrone Ereignisse, die zu einer bestimmten Zeit stattfinden und in ihrem Umfang von kleinen Veranstaltungen, bei denen ein Rennorganisator eine private Videokonferenz moderiert, bis hin zu großen Veranstaltungen reichen, die von einem kleinen Team von Organisatoren, Kommentatoren und Sendern öffentlich übertragen werden. Sie können die Tür für die Einreichung von Modellen jederzeit öffnen und schließen. Teilen Sie den Rennfahrern daher die Frist mit. Die Teilnehmer können mehrere Modelle einreichen, aber nur das letzte Modell, das sie einreichen, bevor Sie die Tür schließen, kann während der Veranstaltung Rennen fahren. Bei LIVE-Rennen haben Teilnehmer in der Warteschlange die Möglichkeit, interaktive Geschwindigkeitskontrollen zu verwenden, um ihrem Modell einen Wettbewerbsvorteil zu verschaffen, wenn sie an der Reihe sind. Teilnehmer an LIVE-Rennen müssen außerdem einen Einladungslink erhalten, um ein Modell für das Rennen einzureichen. Du kannst jedoch wählen, ob du die Veranstaltung nur für eingeladene Teilnehmer privat oder öffentlich über einen LIVE-Streaming-Dienst wie Twitch übertragen möchtest. Weitere Informationen hierzu finden Sie unter the section called “Ein LIVE-Rennen übertragen”.

Um mit der Erstellung eines Community-Rennens zu beginnen

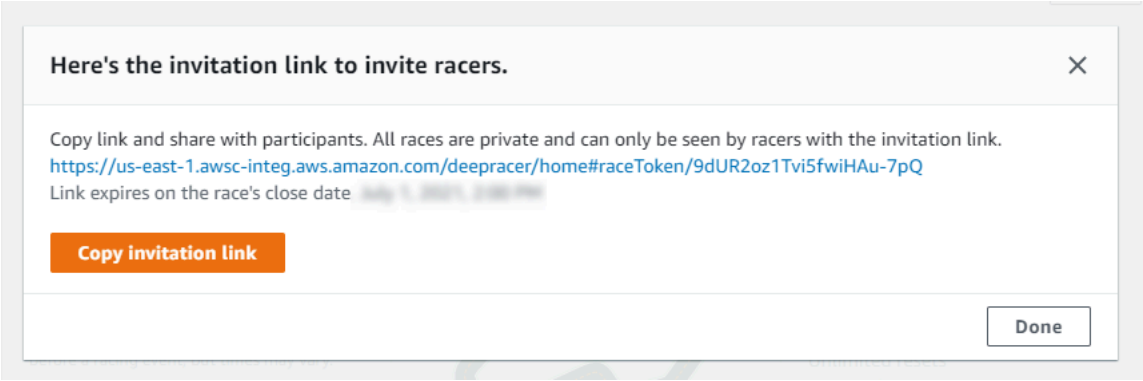
1. Öffnen Sie die DeepRacer AWS-Konsole.
2. Wählen Sie Community-Rennen.
3. Wähle auf der Seite Community-Rennen die Option Rennen erstellen aus.



4. Wählen Sie auf der Seite mit den Renndetails ein Wettbewerbsformat aus: ein klassisches Rennen, an dem Ihre Gäste innerhalb des von Ihnen festgelegten Zeitrahmens nach ihrem eigenen Zeitplan teilnehmen können, oder ein LIVE-Rennen, das privat oder öffentlich als Echtzeit-Event übertragen werden kann.

Um weiterhin ein klassisches Rennen zu erstellen

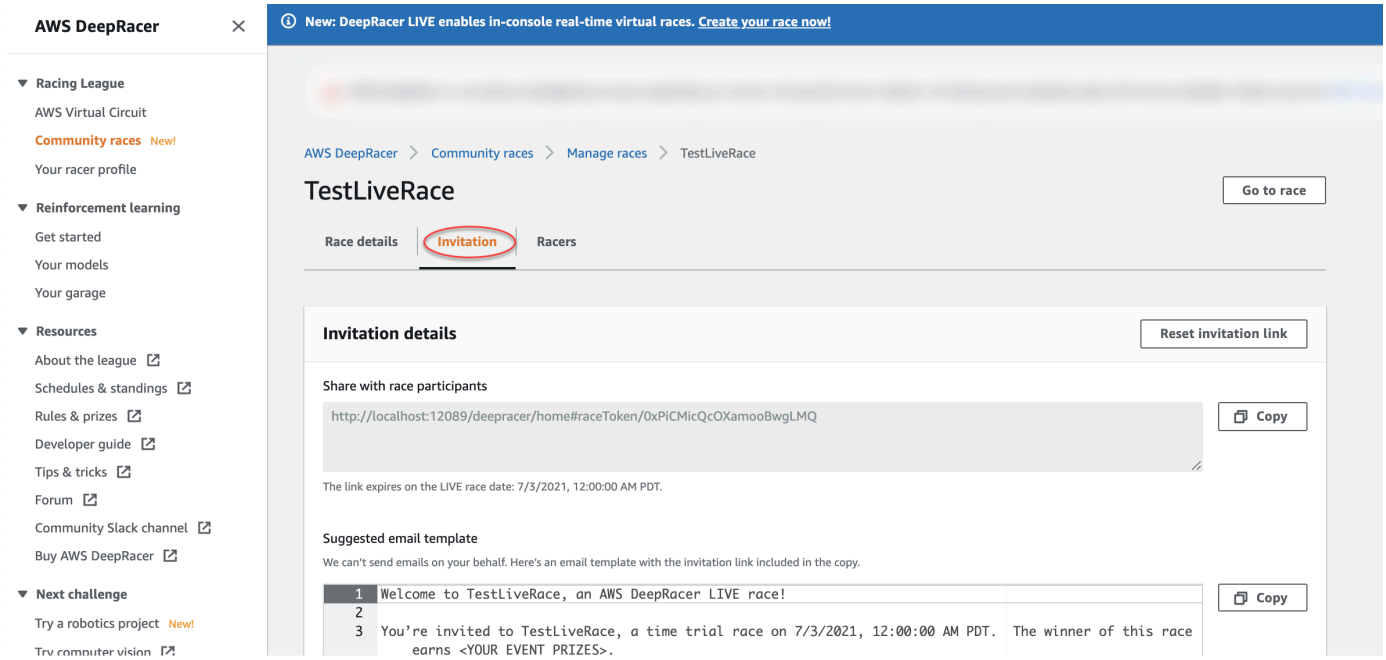
1. Wähle einen Renntyp. Die Komplexität der Renntypen nimmt von Zeitfahren über Objektvermeidung bis hin zu H zu. ead-to-bot Für Erstfahrer empfehlen wir Zeitfahren. Für Zeitfahrrennen ist nur eine Kamera erforderlich, sodass die Sensorkonfiguration einfacher ist und die Modelle mit Reinforcement-Learning (RL), die für diese Art von Rennen trainiert wurden, schneller konvergieren. Weitere Informationen zu den Renntypen finden Sie unter Tailor AWS DeepRacer Training for Time Trials, Object Avoidance and Head-to-Bot Races.
2. Geben Sie einen originellen, aussagekräftigen Namen für das Rennen ein.
3. Geben Sie das Startdatum und die Uhrzeit der Veranstaltung im 24-Stunden-Format an. Die DeepRacer AWS-Konsole erkennt Ihre Zeitzone automatisch. Geben Sie für klassische Rennen auch ein Enddatum und eine Endzeit ein. LIVE-Rennen haben eine Standarddauer von vier Stunden. Wenden Sie sich an den Kundensupport, um ein längeres Rennen zu vereinbaren. Falls dein LIVE-Rennen vorzeitig endet, kannst du nichts unternehmen.
4. Um die Standard-Renneinstellungen zu verwenden, wähle Weiter.
5. Überprüfe auf der Seite mit den Renndetails überprüfen die Rennspezifikationen. Um Änderungen vorzunehmen, wähle Bearbeiten oder Zurück, um zur Seite mit den Renndetails zurückzukehren. Wenn du bereit bist, den Einladungslink zu erhalten, wähle Absenden.
6. Um dein Rennen zu teilen, wähle im Modal den Einladungslink kopieren und füge ihn in E-Mails, Textnachrichten und deine bevorzugten Social-Media-Anwendungen ein. Alle klassischen Rennen sind privat und können nur von Rennfahrern mit dem Einladungslink gesehen werden. Der Link läuft am Schlusstag des Rennens ab.



7. Wählen Sie Erledigt aus. Die Seite „Rennen verwalten“ wird angezeigt.
8. Wenn sich dein klassischer Rennzeitraum dem Ende zuneigt, notiere dir unter Rennfahrer auf der Detailseite zur Bestenliste, wer ein Modell eingegeben hat und wer dies noch tun muss.

Um mit der Erstellung eines LIVE-Rennens fortzufahren

1. Wähle einen Renntyp. Die Komplexität der Renntypen nimmt von Zeitfahren über Objektvermeidung bis hin zu H zu. ead-to-bot Für Erstfahrer empfehlen wir Zeitfahren. Für Zeitfahrrennen ist nur eine Kamera erforderlich, sodass die Sensorkonfiguration einfacher ist und die Modelle mit Reinforcement-Learning (RL), die für diese Art von Rennen trainiert wurden, schneller konvergieren. Weitere Informationen zu den Renntypen finden Sie unter Tailor AWS DeepRacer Training for Time Trials, Object Avoidance and Head-to-Bot Races.
2. Geben Sie einen originellen, aussagekräftigen Namen für das Rennen ein.
3. Geben Sie das Startdatum und die Uhrzeit der Veranstaltung im 24-Stunden-Format an. Die DeepRacer AWS-Konsole erkennt Ihre Zeitzone automatisch. Geben Sie für klassische Rennen auch ein Enddatum und eine Endzeit ein. LIVE-Rennen haben eine Standarddauer von vier Stunden. Wenden Sie sich an den Kundensupport, um ein längeres Rennen zu vereinbaren. Falls dein LIVE-Rennen vorzeitig endet, kannst du nichts unternehmen.
4. Um die Standard-Renneinstellungen zu verwenden, wähle Weiter.
5. Überprüfe auf der Seite mit den Renndetails überprüfen die Rennspezifikationen. Um Änderungen vorzunehmen, wähle Bearbeiten oder Zurück, um zur Seite mit den Renndetails zurückzukehren. Wenn du bereit bist, den Einladungslink zu erhalten, wähle Absenden.
6. <Your Race Name>Wähle auf der Seite den Tab Einladung, um dein Rennen mit anderen zu teilen.



7. Wähle unter Einladungsdetails die Option Kopieren aus, um den Einladungslink in E-Mails, Textnachrichten und deine bevorzugten Social-Media-Anwendungen einzufügen.
8. Wählen Sie optional neben der vorgeschlagenen E-Mail-Vorlage „Kopieren“ aus und geben Sie Ihre Preise, den Zeitrahmen für die Einreichung des Modells und den Link zur Konferenzbrücke ein, über den sich Ihre Fahrer treffen, um sich in die Warteschlange zu stellen und sich auf das Rennen vorzubereiten.

   LIVE-Rennen sind privat und können nur von Rennfahrern mit dem Einladungslink gesehen werden, es sei denn, du entscheidest dich dafür, sie öffentlich zu übertragen. Weitere Informationen hierzu finden Sie unter the section called “Ein LIVE-Rennen übertragen”. Der Link läuft am Schlusstag des Rennens um 12:00 Uhr MESZ ab.
9. Wähle den Tab Renndetails.
10. Notiere dir unter Renndetails die Optionen für die Übertragung deines LIVE-Rennens. Sobald Sie sich entschieden haben, ob Sie Ihr Rennen öffentlich oder privat übertragen möchten, verwenden Sie zunächst Playbooks, die vom AWS DeepRacer League-Team erstellt wurden. Mit der Schaltfläche „Übertragungsmodus anzeigen“ können Sie die Seite des LIVE-Rennens so formatiert sehen, dass sie mit markenspezifischen grafischen Overlays verwendet werden kann, die Ausschnitte für Kommentatoren-Streams enthalten.
11. <Your Race Name>Wenn Ihr LIVE-Renntermin näher rückt, notieren Sie sich unter dem Tab Einladung auf der Seite, wer ein Model angemeldet hat und wer dies noch tun muss.

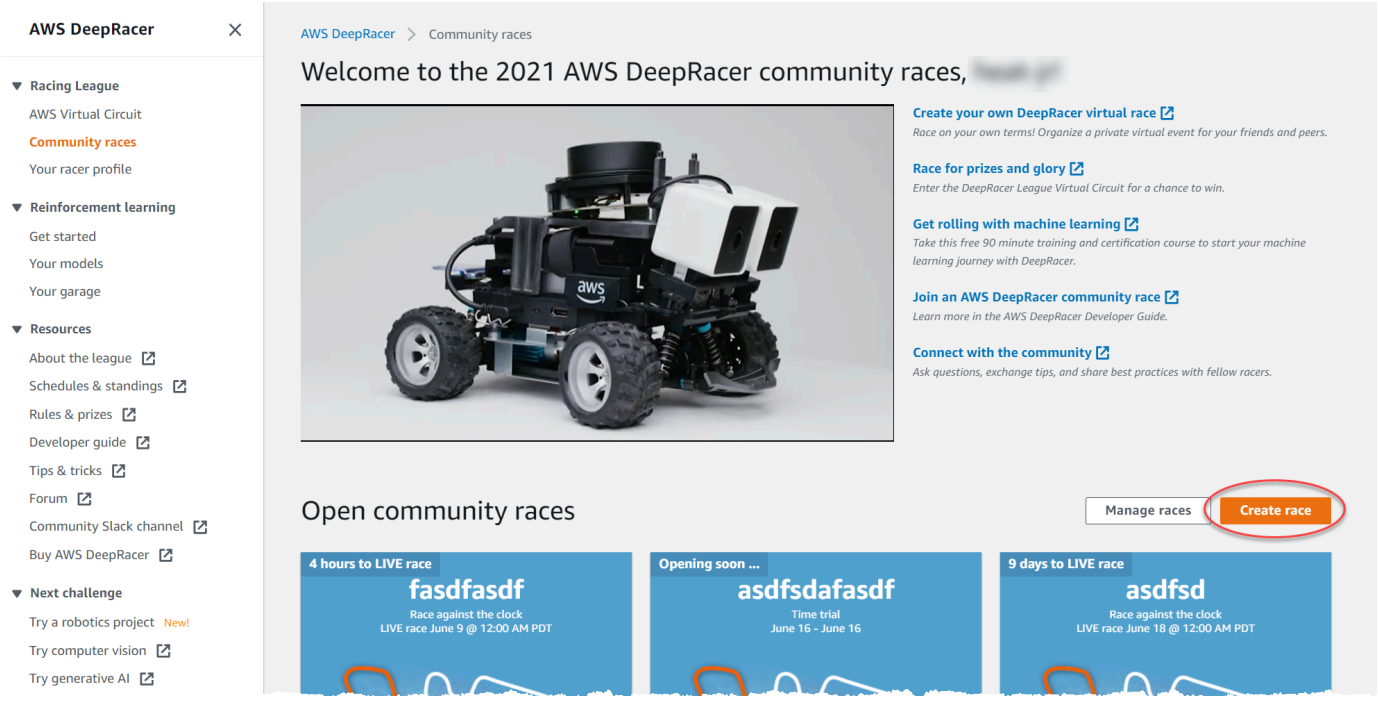
Um die gewählte Rennstrecke zu ändern, eine Rennbeschreibung hinzuzufügen, eine Rangierungsmethode auszuwählen, zu entscheiden, wie viele Resets Rennfahrer erlaubt sind, die Mindestanzahl an Runden festzulegen, die ein RL-Modell absolvieren muss, um sich für Ihr Rennen zu qualifizieren, die Off-Track-Strafe festzulegen und andere Renndetails anzupassen, wählen Sie unter Community-Rennen verwalten die Option Renndetails bearbeiten.

Passe ein Rennen an

Um ein Rennen zu erstellen, das auf deine Gruppe zugeschnitten ist, erweitere Rennanpassungen auf der Seite mit den Renndetails. Die Einstellungen für ein Zeitfahrrennen gelten auch für die Vermeidung von Objekten und head-to-bot Rennen, aber für Objektvermeidung und head-to-bot Renntypen gibt es zusätzliche Einstellungen, mit denen Sie Rennumgebungen einrichten können, die speziell auf Ihre Veranstaltungsziele abgestimmt sind.

Um ein Rennen individuell zu gestalten

1. Öffnen Sie die DeepRacer AWS-Konsole.
2. Wählen Sie Community-Rennen.
3. Wähle auf der Seite Community-Rennen die Option Rennen erstellen aus.

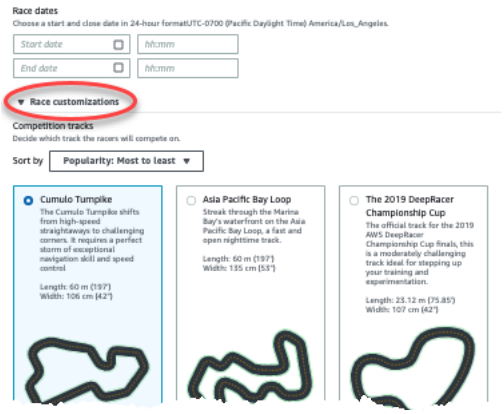


4. Wählen Sie auf der Seite mit den Renndetails ein Wettbewerbsformat aus: ein klassisches Rennen, an dem Ihre Gäste innerhalb des von Ihnen festgelegten Zeitrahmens nach ihrem

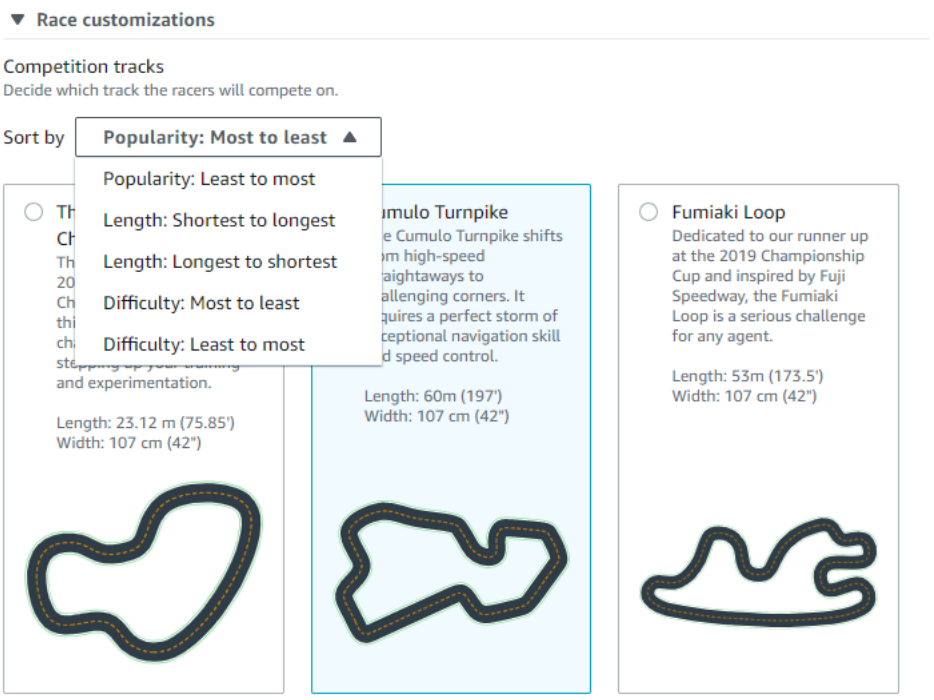
eigenen Zeitplan teilnehmen können, oder ein LIVE-Rennen, das privat oder öffentlich als Echtzeit-Event übertragen werden kann.

5. Folgen Sie je nach ausgewähltem Wettbewerbsformat den Schritten 1—3 von So erstellen Sie ein klassisches Rennen weiter oder So erstellen Sie weiterhin ein LIVE-Rennen inthe section called “Erstelle einen Schnellstart für ein Rennen”.

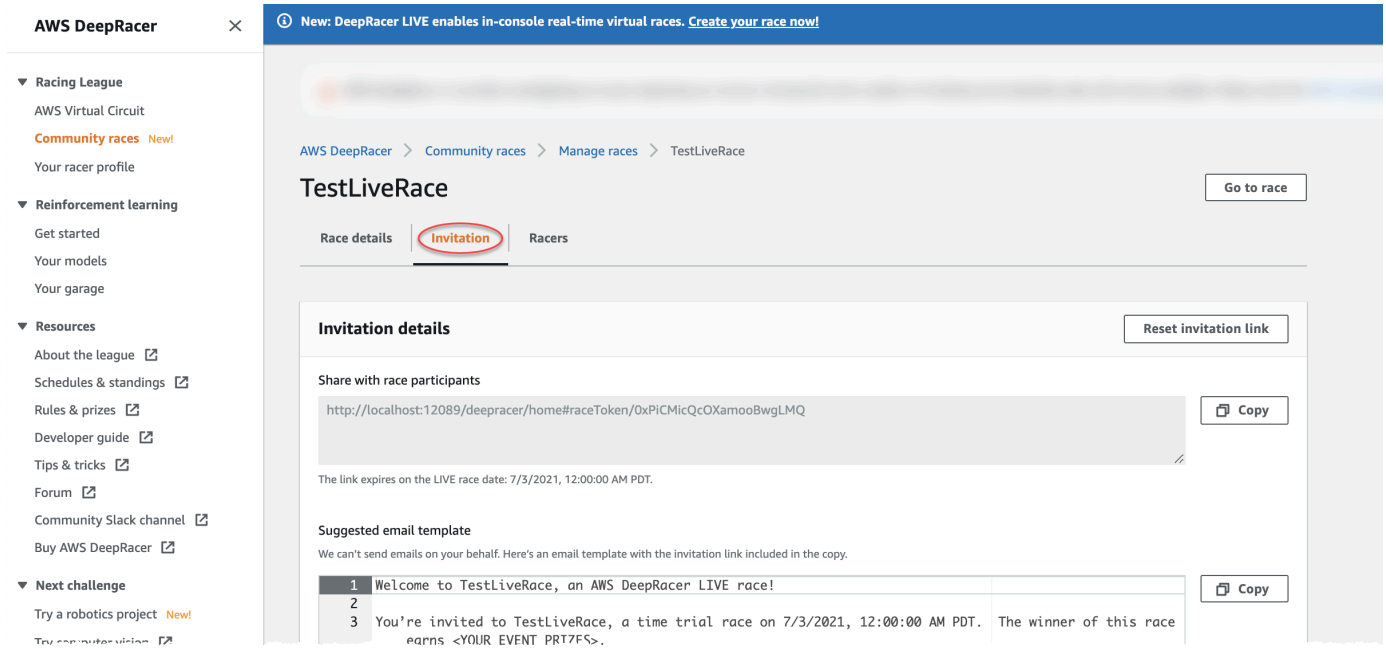
6. Nachdem du deine Renndaten ausgewählt hast, erweitere die Renn-Anpassungen.



7. Wählen Sie eine Wettkampfstrecke aus. Du kannst Strecken nach Beliebtheit sortieren: Am meisten bis am wenigsten, am wenigsten bis am meisten, Schwierigkeitsgrad: am meisten bis am wenigsten, am wenigsten bis am meisten und Länge: am längsten bis am kürzesten. Um alle Strecken in jeder Kategorie zu sehen, wähle „Weitere Rennstreckenoptionen anzeigen“. Um das erweiterte Menü zu schließen, wähle „Weniger Rennstreckenoptionen anzeigen“.



8. Schreiben Sie optional eine Beschreibung für Ihr Rennen, in der die Ziele und Regeln der Veranstaltung für die Teilnehmer zusammengefasst sind. Fügen Sie bei LIVE-Rennen den Link für die Videokonferenz oder den LIVE-Stream Ihrer Veranstaltung hinzu. Die Beschreibung erscheint in den Details deiner Bestenliste.
9. Wählen Sie für die Ranglistenmethode für ein klassisches Rennen zwischen der besten Rundenzeit, bei der der Gewinner der Fahrer ist, der die schnellste Runde gefahren hat, der Durchschnittszeit, bei der nach mehreren Versuchen innerhalb des Zeitrahmens der Rennfahrer mit der besten Durchschnittszeit der Gewinner ist, oder Gesamtzeit, bei der der Fahrer mit dem schnellsten Gesamtdurchschnitt der Gewinner ist. Die Bestenlisten aller Live-Rennen werden nach der besten Rundenzeit geordnet, sodass dieses Feld nicht angezeigt wird.
10. Wählen Sie für klassische Rennen einen Wert für „Mindestrunden“. Dabei handelt es sich um die Anzahl der aufeinanderfolgenden Runden, die ein Rennfahrer zurücklegen muss, um sich für die Einreichung des Ergebnisses in der Bestenliste des Rennens zu qualifizieren. Wählen Sie für ein Anfängerrennen eine kleinere Zahl. Für fortgeschrittene Benutzer wählen Sie eine größere Anzahl. Diese Anpassung ist für LIVE-Rennen nicht verfügbar, da die Standardeinstellung eine Runde ist.
11. Wählen Sie für die Strafe außerhalb der Strecke die Anzahl der Sekunden aus, die zur Zeit eines Rennfahrers hinzugefügt werden soll, wenn sein RL-Modell von der Strecke abkommt.
12. Du hast jetzt alle Anpassungsoptionen für ein Zeitfahrrennen abgeschlossen. Wenn du dich für ein Zeitfahren-Format entschieden hast, wähle Weiter, um die Renndetails zu überprüfen. Wenn Sie sich für das Format Objektvermeidung oder ead-to-botH-Rennen entschieden haben, fahren Sie mit dem entsprechenden Verfahren fort, um die Anpassung Ihres Rennens abzuschließen.
13. Überprüfe auf der Seite „Renndetails überprüfen“ die Rennspezifikationen. Um Änderungen vorzunehmen, wähle Bearbeiten oder Zurück, um zur Seite mit den Renndetails zurückzukehren. Wenn du bereit bist, den Einladungslink zu erhalten, wähle Absenden.
14. Um dein Rennen zu teilen, wähle im Modal in deine Zwischenablage die Option Einladungslink kopieren und füge ihn in E-Mails, Textnachrichten und deine bevorzugten Social-Media-Anwendungen ein. Du kannst auch den Tab Einladung wählen, um dein Rennen auf der <Your Race Name>Seite zu teilen. Der Link läuft am Schlusstag des Rennens ab.

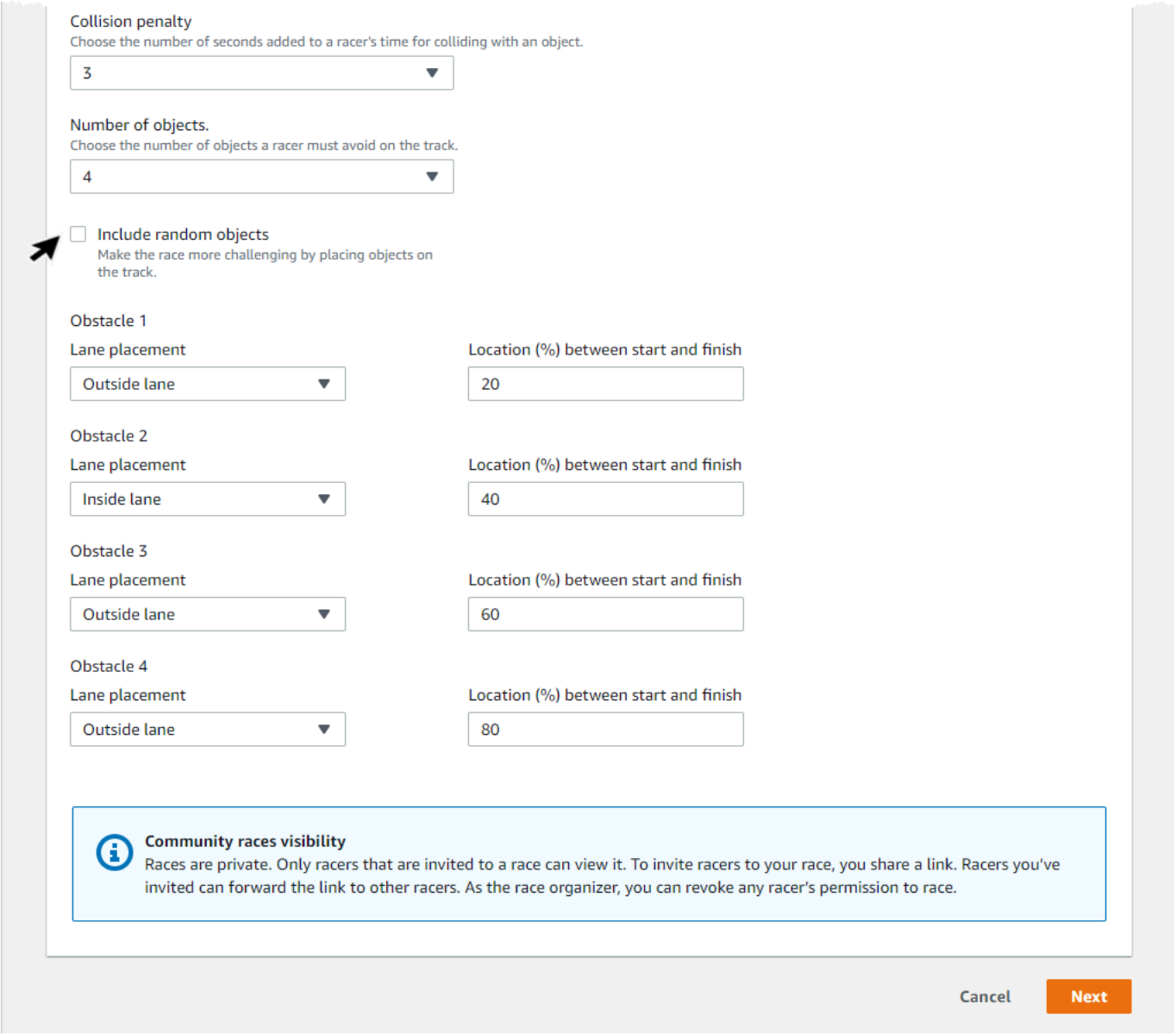


15. Wählen Sie Erledigt aus. Die Seite „Rennen verwalten“ wird angezeigt.

Unter Community-Rennen verwalten erfährst du, wie du unsere E-Mail-Vorlage verwenden kannst, um neue Rennfahrer einzuladen, Rennfahrer aus deinem Rennen zu entfernen, den Status der eingereichten Modelle von Rennfahrern zu überprüfen und mehr.

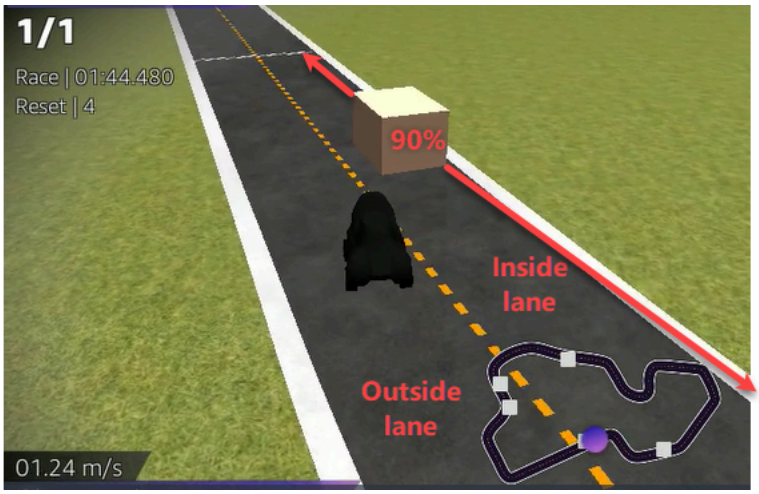
Um das Anpassen eines Wettlaufs zur Vermeidung von Objekten abzuschließen

1. Wählen Sie für die Kollisionsstrafe die Anzahl der Sekunden aus, die der Rennfahrer für die Kollision mit einem Objekt oder einem Bot zur Zeit hinzugerechnet hat. Je mehr Sekunden hinzugefügt werden, desto größer ist die Herausforderung.



2. Wählen Sie unter Anzahl der Objekte aus, wie vielen Hindernissen ein Rennfahrer auf der Strecke ausweichen muss. Je mehr Objekte, desto schwieriger das Rennen.
3. Um der Rennstrecke zufällige Objekte hinzuzufügen, die für jeden Rennfahrer an unterschiedlichen Orten erscheinen, wähle „Zufällige Objekte einbeziehen“. Das ist für die Teilnehmer schwieriger, da man länger trainiert und die Funktion Versuch und Irrtum belohnt, um RL-Modelle zu erstellen, die sich gut auf zufällige Ereignisse wie unerwartete Objekte auf einer Rennstrecke verallgemeinern lassen.
4. Wählen Sie aus, wo jedes Objekt platziert werden soll, indem Sie eine Fahrspurnummer oder eine Objektposition für die Spurplatzierung auswählen. Die Spur ist an der Mittellinie in zwei

Hälften geteilt, wodurch innere und äußere Fahrspuren entstehen. Sie können ein Objekt entweder auf der inneren oder äußeren Fahrbahn platzieren.

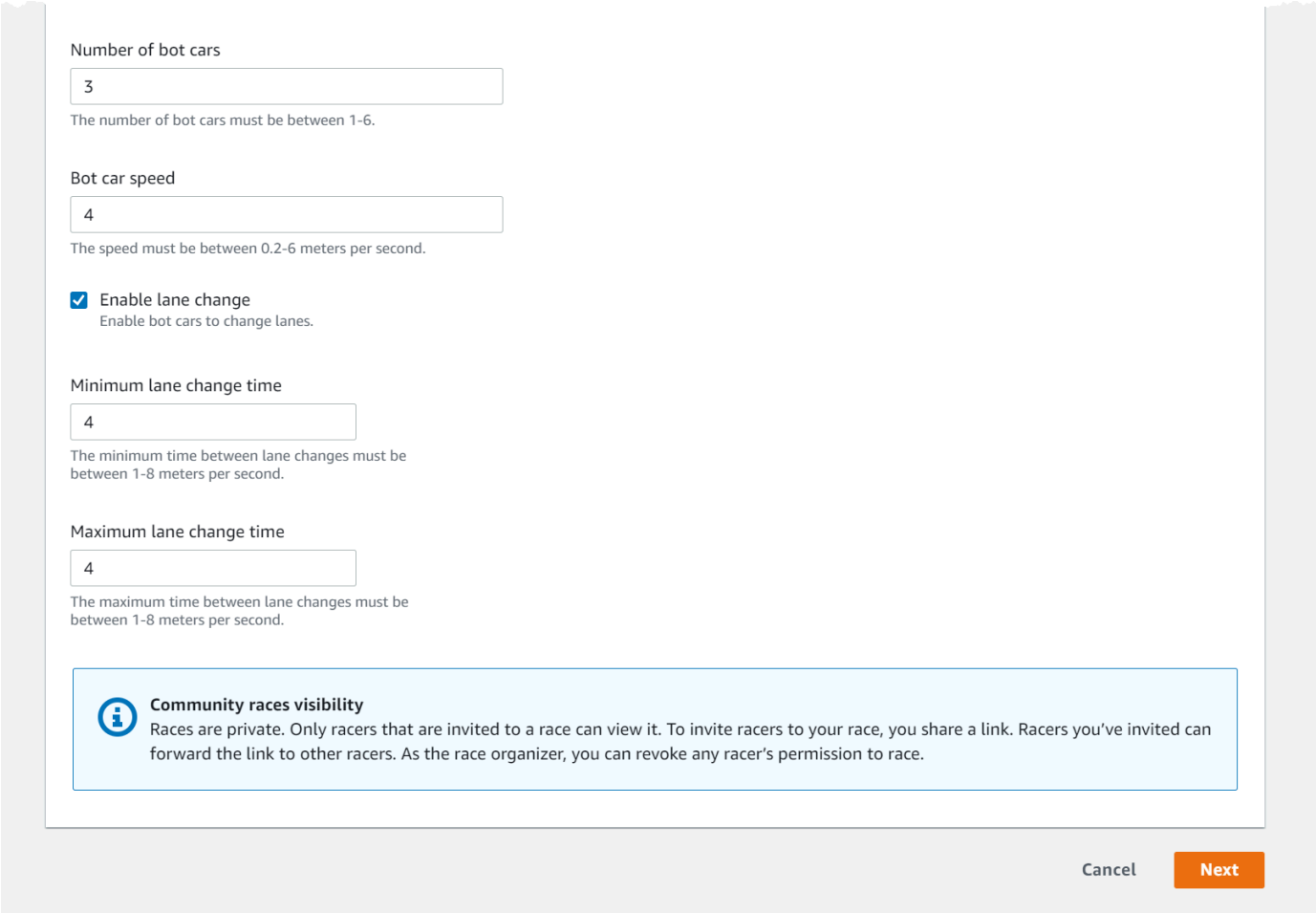


5. Wählen Sie für jedes Objekt einen Wert für Standort (%) zwischen Start und Ziel. Die Zahl steht für die Position, ausgedrückt als Prozentsatz, zwischen der Start- und Ziellinie Ihrer Strecke, an der Sie das Objekt platzieren möchten.
6. Sie haben jetzt alle einzigartigen Anpassungsoptionen für ein Rennen zur Vermeidung von Objekten abgeschlossen. Wählen Sie Weiter.
7. Überprüfe auf der Seite mit den Renndetails die Rennspezifikationen. Um Änderungen vorzunehmen, wähle Bearbeiten oder Zurück, um zur Seite mit den Renndetails zurückzukehren. Wenn du bereit bist, den Einladungslink zu erhalten, wähle Absenden.
8. Um dein Rennen zu teilen, wähle Einladungslink kopieren und füge ihn in E-Mails, Textnachrichten und deine bevorzugten Social-Media-Anwendungen ein. Alle Rennen sind privat und können nur von Rennfahrern mit dem Einladungslink gesehen werden. Der Link läuft am Schlusstag des Rennens ab.
9. Wählen Sie Erledigt aus. Die Seite Rennen verwalten wird angezeigt.

Unter Community-Rennen verwalten erfährst du, was du mit deinem Rennen machen kannst.

Um die Anpassung eines head-to-bot Rennens abzuschließen

1. Wählen Sie unter Anzahl der Bot-Autos die Anzahl der Autos aus, die Sie gegen die AWS DeepRacer RL-Modelle Ihrer Teilnehmer antreten möchten. Bot-Autos ähneln KI-Fahrzeugen für Videospiele. Da es sich um zufällige Objekte handelt, die sich bewegen, weisen sie gegenüber stationären Objekten eine höhere Komplexität auf. Je mehr Bots auf der Strecke sind, desto herausfordernder ist das Rennen. Wähle bis zu sechs.



2. Wählen Sie für die Geschwindigkeit der Bot-Autos aus, wie schnell sich die Bot-Autos auf der Strecke bewegen sollen. Die Geschwindigkeit wird in Metern pro Sekunde gemessen. Die Geschwindigkeit muss zwischen 0,2 und 6 Metern pro Sekunde liegen.
3. Wenn Sie Bots den Spurwechsel ermöglichen möchten, was die Herausforderung für die DeepRacer AWS-RL-Modelle Ihrer Rennfahrer noch komplexer macht, wählen Sie „Spurwechsel aktivieren“.
4. Wählen Sie für Minimale Spurwechselzeit die Mindestanzahl von Sekunden aus, die zwischen den Instanzen vergehen, in denen die Bot-Autos die Spur wechseln.
5. Wählen Sie für Maximale Spurwechselzeit die maximale Anzahl von Sekunden aus, die zwischen den Instanzen vergehen, in denen die Bot-Autos die Spur wechseln.
6. Du hast jetzt alle einzigartigen Anpassungsoptionen für ein head-to-bot Rennen abgeschlossen. Wählen Sie Weiter.
7. Überprüfe auf der Seite „Renndetails überprüfen“ die Rennspezifikationen. Um Änderungen vorzunehmen, wähle Bearbeiten oder Zurück, um zur Seite mit den Renndetails zurückzukehren. Wenn du bereit bist, den Einladungslink zu erhalten, wähle Absenden.

8. Um dein Rennen zu teilen, wähle Einladungslink kopieren und füge ihn in E-Mails, Textnachrichten und deine bevorzugten Social-Media-Anwendungen ein. Alle Rennen sind privat und können nur von Rennfahrern mit dem Einladungslink gesehen werden. Der Link läuft am Schlusstag des Rennens ab.
9. Wählen Sie Erledigt aus. Die Seite Rennen verwalten wird angezeigt.

Unter Community-Rennen verwalten erfährst du, wie du dein Rennen bearbeiten und löschen kannst.

Führen Sie ein DeepRacer LIVE-AWS-Community-Rennen durch

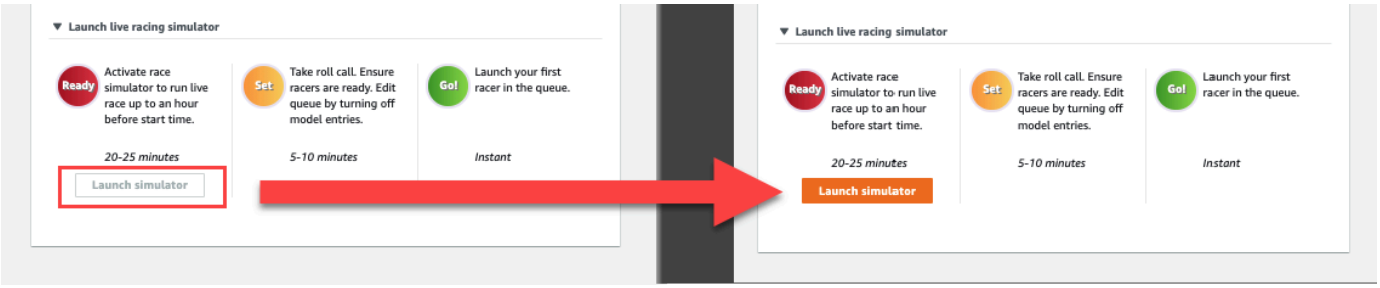
Du hast ein LIVE-Rennen erstellt und Rennfahrer eingeladen. Sie haben entschieden, ob Sie Ihre Veranstaltung privat oder öffentlich mit Unterstützung von the section called “Ein LIVE-Rennen übertragen” übertragen möchten. Erfahre jetzt, wie du die Warteschlange verwaltest, den Rennsimulator einrichtest und deine Rennfahrer startest.

Bevor Sie beginnen

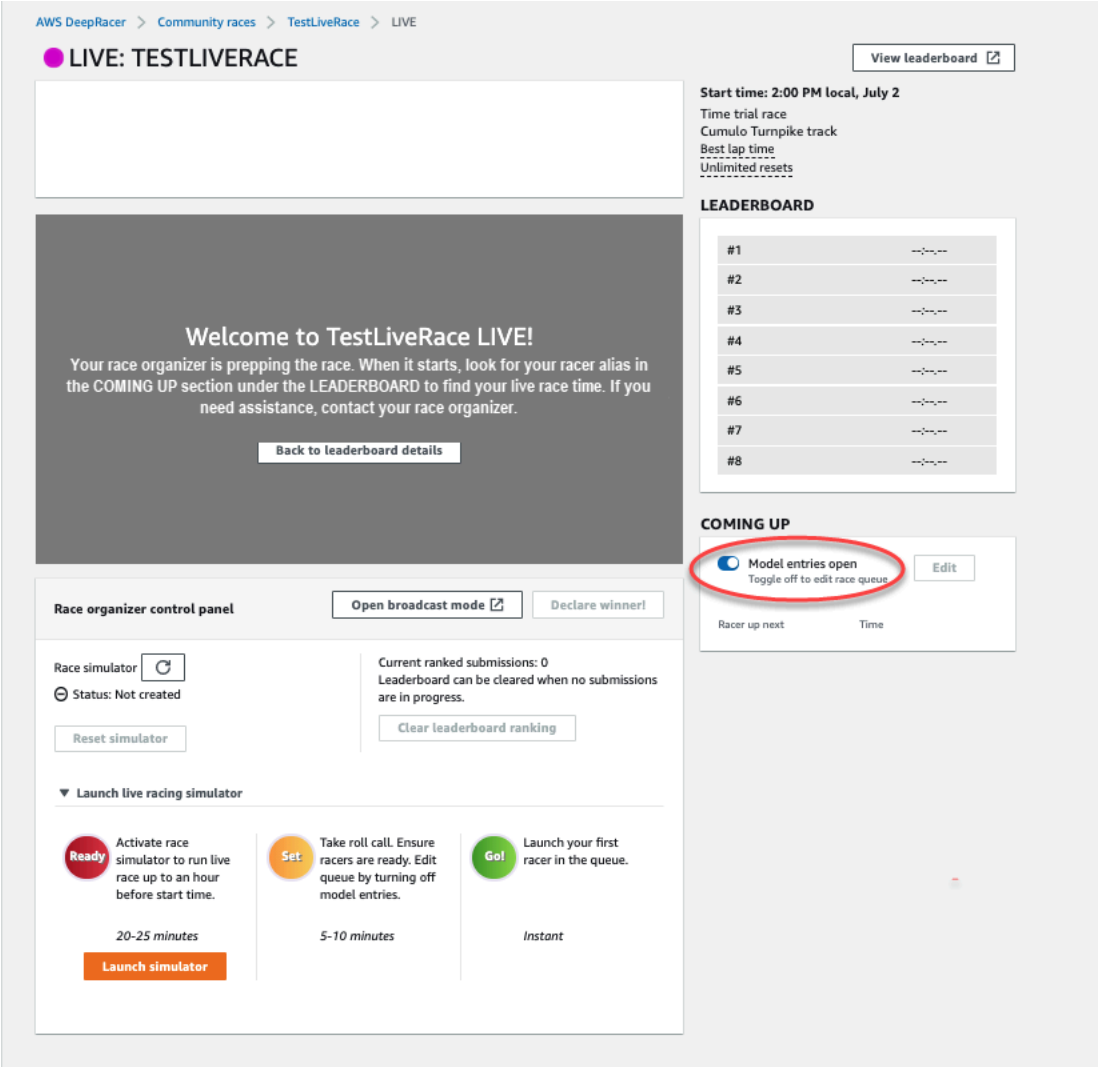
- Verwenden Sie einen Chrome- oder Firefox-Browser (stellen Sie sicher, dass Ihr Browser auf dem neuesten Stand ist).
- Trennen Sie das virtuelle private Netzwerk (VPN), falls Sie eines verwenden.
- Schließen Sie alle zusätzlichen Tabs.

Um ein virtuelles LIVE-Rennen zu veranstalten

1. Suchen Sie auf der Seite Community-Rennen die Rennkarte für das Rennen, das Sie moderieren möchten, und wählen Sie Jetzt beitreten, um das Rennen anzusehen.
2. Wähle auf der <Your Race Name>Seite LIVE: unter dem Kontrollpanel Rennorganisator die Option Simulator starten aus. Diese Schaltfläche ist eine Stunde vor dem Start des Rennens nutzbar. Du kannst diesen Bereich im Kontrollpanel des Rennorganisators ausblenden, indem du den Header LIVE-Rennsimulator starten auswählst.



3. Schalten Sie unter DEMNÄCHST die Option Modelleinträge öffnen aus, um Einreichungen zu schließen. Dadurch werden die eingereichten Modelle geschlossen und unterhalb des Schalters eine Warteschlange für Rennfahrer erstellt, die bearbeitet werden kann. Du kannst Rennfahrer erst starten, wenn der Schalter ausgeschaltet ist.



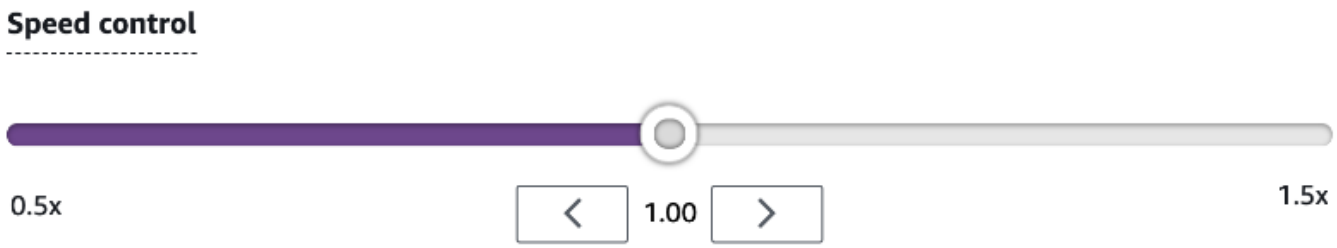
4. Öffnen Sie die Videokonferenz, die Sie erstellt haben, um Ihre Rennfahrer zu versammeln.
5. Initiieren Sie einen Roll-Call für die Rennfahrer:
   a. Erkundigen Sie sich bei den Rennfahrern, ob sie Sie deutlich hören können.
   b. Verwenden Sie zunächst ein Video, um sich vorzustellen. Möglicherweise möchten Sie es später ausschalten, um die Bandbreite zu optimieren.
   c. Vergewissern Sie sich, dass die Liste der Teilnehmer des Anrufs mit der Liste der Rennfahrer in Ihrer Gruppe übereinstimmt.
6. Initiieren Sie einen Model-Roll-Call:

a. Vergewissere dich, dass die Liste der Aliase in der Rennfahrer-Warteschlange mit denen deiner Rennfahrer übereinstimmt und dass keiner von ihnen rot markiert ist, was bedeutet, dass ihr Modell nicht erfolgreich eingereicht wurde.

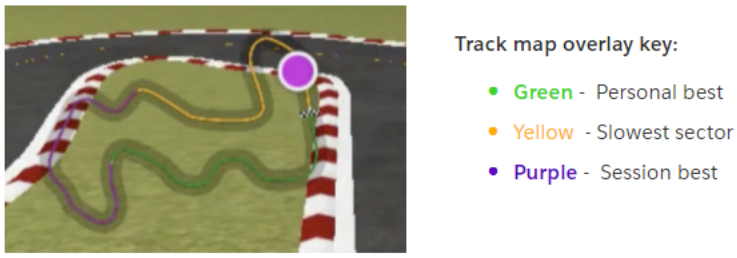
b. Erkundigen Sie sich bei Ihren Rennfahrern, ob sie Probleme beim Einreichen ihrer Modelle haben.

7. Überprüfe den Rennplan und die Regeln. Teilen Sie den Rennfahrern mit, wie viel Zeit sie für das Rennen haben, wenn sie an der Reihe sind, und erinnern Sie sie daran, dass die Bestenliste von ihrer schnellsten Runde in diesem Zeitraum abhängt.

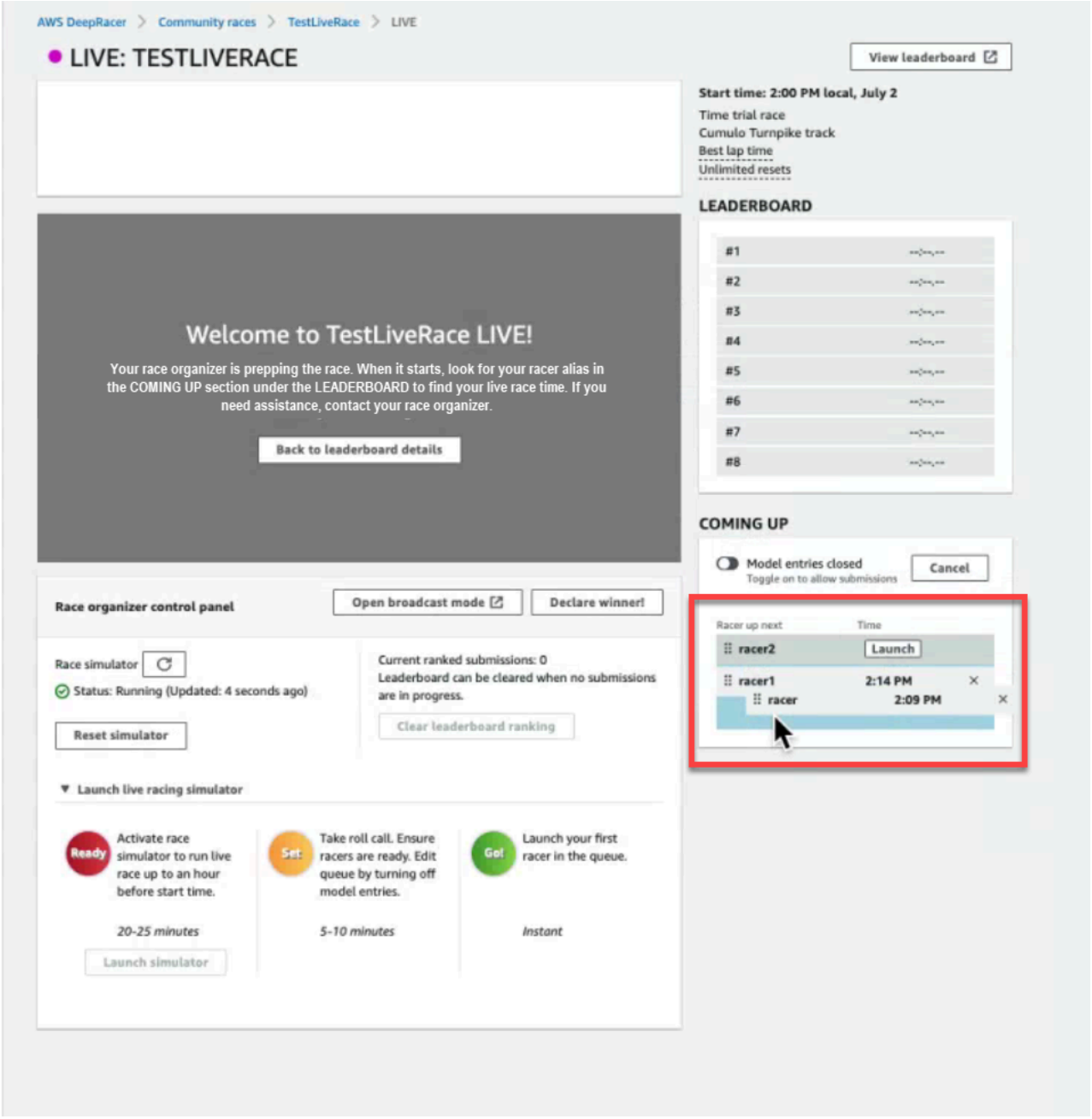
8. Erklären Sie, dass sie mithilfe der Geschwindigkeitsregelungsfunktion, die nur für den Rennfahrer während des Rennens sichtbar ist, die Höchstgeschwindigkeit manuell über den Geschwindigkeitsregler einstellen können. Dadurch werden vorübergehend die Geschwindigkeitsparameter des Modells außer Kraft gesetzt, nicht jedoch der Lenkwinkel. Das Modell lenkt immer noch, aber die Rennfahrer können jetzt wichtige Momente wählen, um die Geschwindigkeit ihres Autos zu erhöhen oder zu verringern, indem sie die Geschwindigkeit multiplizieren. Um wieder die Geschwindigkeitsparameter des Modells zu verwenden, können Rennfahrer den Multiplikator auf 1 zurücksetzen. Erinnern Sie die Rennfahrer daran, dass der Schieberegler für die Geschwindigkeitssteuerung nicht das Gaspedal ist, sondern eine Gelegenheit für eine strategische Anpassung in Echtzeit bietet.



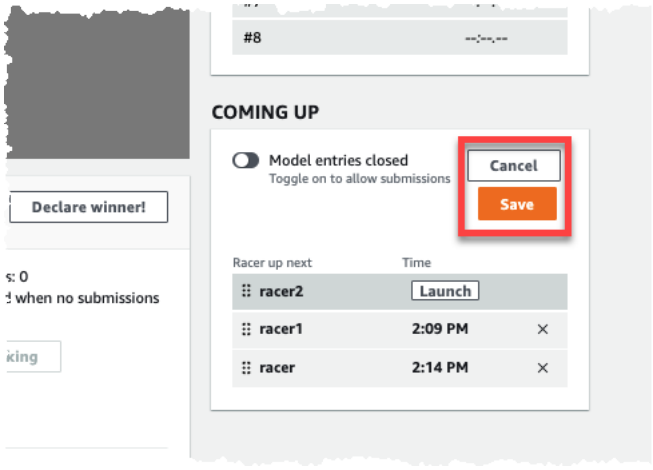
9. Erklären Sie als Nächstes, dass das Video-Overlay des Rennfensters Informationen enthält, mit deren Hilfe die Leistung eines Rennfahrers optimiert werden kann. Das Streckenkarten-Overlay ist in drei Sektoren unterteilt, deren Farbe sich je nach Tempo des Rennfahrers ändert. Grün steht für den Abschnitt der Strecke, in dem ein Rennfahrer seine persönliche Bestzeit erzielt hat, Gelb steht für den am langsamsten gefahrenen Sektor und Violett für die beste Trainingseinheit. Rennfahrer finden hier auch Statistiken mit Angaben zu ihrer besten Rundenzeit, ihrer verbleibenden Geschwindigkeit in m/s, zu Resets und zur aktuellen Rundenzeit.



10. Beantworten Sie Fragen von Rennfahrern.
11. Wählen Sie optional unter DEMNÄCHST die Option Bearbeiten aus, um Ihre Rennwarteschlange neu zu ordnen, indem Sie die Namen der Rennfahrer eingeben und dort ablegen.



12. Wenn du Änderungen an deiner Rennfahrer-Warteschlange vornimmst, wähle Speichern, um deine Änderungen beizubehalten, oder Abbrechen, um sie zu verwerfen.



13. Starte den ersten Rennfahrer in deiner Warteschlange:

    a. Starte jeden Rennfahrer manuell, indem du auf die Schaltfläche Starten neben dem Namen der obersten Rennfahrer-Warteschlange klickst. In der Runde jedes Rennfahrers gibt es 10, 9, 8, 7, 6... Der Countdown wird nach dem Start in der Konsole animiert.

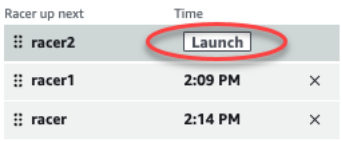
    b. Auf dem Sprung! , das Modell läuft für die von Ihnen gewählte Zeit und wird gleichzeitig in Echtzeit ausgewertet.

    c. Falls ein Modell mitten im Rennen ausfällt, musst du den Rennfahrer neu starten, indem du in der Racer-Warteschlange auf die Schaltfläche Starten neben seinem Alias klickst.

    d. Etwa 2 Minuten bevor der aktuelle Rennfahrer fertig ist, kontaktiere die nächsten 2 Rennfahrer in der Warteschlange über deine Konferenzbrücke und bestätige, dass sie bereit für das Rennen sind.

    e. 30 Sekunden vor dem Ziel des aktuellen Rennfahrers sollten Sie den nächsten Fahrer 30 Sekunden lang warnen.

    f. Starte den nächsten Rennfahrer, sobald du siehst, dass der aktuelle Rennfahrer fertig ist. Das Ende des Rennens wird durch ein kariertes Flaggensymbol in der Konsole angezeigt. Die Geschwindigkeitskontrolle des Rennfahrers ist deaktiviert und eine Wiederholung des Rennens wird auf dem Videobildschirm gestartet.



14. Wählen Sie optional „Simulator zurücksetzen“, wenn Sie Probleme mit dem Simulator haben.

15. Du kannst auch „Rangliste löschen“ wählen, wenn du aus irgendeinem Grund die Bestenliste zurücksetzen möchtest, wodurch alle Einträge gelöscht werden.
16. Wähle am Ende deines Rennens den Gewinner ausfindig machen! klicken Sie auf die Schaltfläche, machen Sie den Rennfahrern letzte Bemerkungen, erklären Sie, wie die Preise verteilt werden, beantworten Sie Fragen und schließen Sie die Videokonferenz.

Übertragen Sie ein LIVE-Community-Rennen mithilfe von AWS DeepRacer League-Produktions-Playbooks

LIVE-Rennen sind Ereignisse in Echtzeit, die an einem bestimmten Datum und zu einer bestimmten Uhrzeit stattfinden. Ihr Umfang reicht von kleinen Veranstaltungen, bei denen ein Rennorganisator eine private Videokonferenz veranstaltet, bis hin zu großen Veranstaltungen, die von einem kleinen Team von Organisatoren, Kommentatoren und Sendern über einen LIVE-Streaming-Dienst wie Twitch öffentlich übertragen werden.

Rollen der Organisatoren

Im Folgenden finden Sie Vorschläge für Rollen, die Organisatoren während einer AWS DeepRacer LIVE-Veranstaltung spielen können. Je komplexer die Veranstaltung, die Sie planen, desto mehr Hilfe benötigen Sie möglicherweise, um sich anzumelden.

Organisatoren

Die Organisatoren des Rennens organisierten das Rennen und die dazugehörige Videokonferenz, um die Rennfahrer zu organisieren und zu begleiten. Während eines LIVE-Rennens nutzen die Organisatoren die Steuerung des Organisators, um sich in die Warteschlange zu stellen, die Rennfahrer zu starten und einen Gewinner auszurufen. Die Organisatoren erscheinen nicht auf dem LIVE-Kanal.

Kommentatoren

Kommentatoren besprechen das Rennen, während es stattfindet, und bieten eine Reihe play-by-play von Ereignissen, zusätzliche Informationen und Insiderwissen über die Veranstaltung und ihre Teilnehmer. Kommentatoren sind die Hauptredner der öffentlichen Veranstaltung.

Rundfunkveranstalter

Rundfunkveranstalter verwenden Streaming-Software, um Szenen im Voraus zu erstellen und sie während des LIVE-Rennens zu überspielen. Ein Sender verwaltet auch die Video-Feeds. Die Sender

erscheinen nicht auf dem LIVE-Kanal. Sie agieren während der Veranstaltung als Produzenten von Inhalten.

Szenen des Senders

Der LIVE-Stream einer DeepRacer AWS-Veranstaltung erzählt die Geschichte Ihres Rennens. Verwenden Sie Szenen, um das Engagement zu Beginn, in der Mitte und am Ende Ihrer Veranstaltung zu fördern. Dabei handelt es sich um Animationen und Layouts, die aus grafischen Überlagerungen und Videostreams bestehen und die verschiedenen Segmente Ihrer Veranstaltung unterstreichen.

Ein Overlay ist eine Grafik (normalerweise eine transparente PNG-Datei), die sich über dem Broadcaster-Modus-Fenster Ihres Rennens und den (optionalen) Webcam-Streams oder Ihren Kommentatoren befindet. Es ist wie eine Maske für deinen Stream. Platzieren Sie Ihre Inhalte darunter, sodass alles nahtlos angeordnet ist und ein einheitliches Layout entsteht.

Verwende Streaming-Software wie OBS, um deine Szenen vor der Übertragung einzurichten. Wechseln Sie während der Veranstaltung reibungslos durch sie, um für ein dynamisches Tempo zu sorgen und das Publikum zu begeistern. Verwenden Sie zum Beispiel eine Intro-Animationsszene, um die Veranstaltung einzuleiten. Gehen Sie dann zu Ihrer primären Inhaltsszene (PCS) über. Dabei handelt es sich um das Hauptlayout, das die Rennansicht und ein oder zwei Fenster für Kommentatoren enthält. Wechseln Sie zu einer Szene mit dualem Kommentator oder Kommentator und Interviewpartner im Vollbildmodus, um das Ganze lebendig zu halten, und beenden Sie mit einer Bestenlisten-Szene. Erstelle optional Werbeszenen, um zwischen den Rennen zu schneiden.

DeepRacer AWS-Szenenvorlagen

Das AWS DeepRacer League Virtual Circuit-Team hat eine Sammlung von Vorlagendateien erstellt, die Sie für Ihre LIVE-Community-Rennen verwenden können. Laden Sie die DeepRacer AWS-Szenenvorlagen herunter und verwenden Sie sie, um eine professionell aussehende Veranstaltung zu übertragen.

Szenentypen und wie man sie verwendet

1. Einführung in die DeepRacer AWS-Schildanimation
2. Ansicht „Nur zum Teilen auf der Konsole“:
   - Basisebene — Bildschirmübertragung der URL deines Rennens im Broadcaster-Modus. Passen Sie die Größe an die Einzelbilder der Szene an.
3. Einzelkommentator-Ansicht (1up):
   - Basisebene — Bildschirmübertragung der URL deines Rennens im Broadcaster-Modus. Passen Sie die Größe an die Einzelbilder der Szene an.

- Nächste Ebene — OBS Ninja oder lokale Webcam, falls sich der Kommentator, den Sie filmen, im selben Raum befindet. Ziehen Sie die Maus hinein und ändern Sie die Größe unter dem Szenenrahmen im oberen rechten Bild-im-Bild-Fenster (PIP).

4. Kommentator plus Interviewpartner oder Doppelkommentator (2up):
   - Basisebene — Bildschirmübertragung der URL deines Rennens im Broadcaster-Modus. Passen Sie die Größe an die Einzelbilder der Szene an.
   - Nächste Ebene — OBS Ninja oder lokale Webcam, falls sich der Kommentator, den Sie filmen, im selben Raum befindet. Ziehen Sie die Maus hinein und ändern Sie die Größe unter dem Szenenrahmen im oberen rechten Bild-im-Bild-Fenster (PIP).
   - Ziehen Sie Dual-Webcam-Feeds oder Ninja-Feeds in die oberen rechten Fenster, deren Größe an die Größe angepasst werden kann (im Setup eine Woche vor Ihrer Veranstaltung — AV überprüft alle Ihre Feeds und weist Kameras in OBS zu)
5. Doppelter Kommentator im Vollbildmodus (keine Rennansicht, nur Interview):
   - Keine Baselayer-Konsole, nur zwei Kamera-Feeds.
6. Ende der Bestenlisten:
   - Geben Sie die Ergebnisse der Bestenliste in Echtzeit manuell über den Szenen-Layer ein.

Tipps AWS DeepRacer AWS-Szenenvorlagendateien

- 34 — Konfigurieren Sie Ihre Titel für Kommentatoren (vorgefertigte Szenen mit Namen) PIPs
- 234 — Rennansichten
  - Erwägen Sie, das AWS DeepRacer League-Logo oben links durch Ihr Firmenlogo zu ersetzen.
  - Ersetzen Sie den Text unten links durch Ihren Rennnamen und Ihre Informationen im vertikalen Text.

Um eine private LIVE-Sendung zu produzieren — ein 5-minütiges Setup

Eine private Übertragung eines AWS DeepRacer LIVE Community Race eignet sich gut für ein kleines, informelles Rennen.

Rollen der Organisatoren

- Für ein Standardrennen benötigen Sie nur einen Organisator.

Hardware (Hardware)

- Empfohlene Hardware: mindestens 16 GB RAM
- (Optional) Hochwertige Mikrofone, Headsets oder AirPods
- (Optional) LED-Ringlicht — Um zu vermeiden, dass das Ringlicht auf der Brille reflektiert wird, positionieren Sie es in einem Winkel zum Gesicht des Brillenträgers.
- (Optional) Webcams und — um das Filmmaterial abwechslungsreicher zu gestalten GoPros

Tipps

- Verwenden Sie einen Chrome- oder Firefox-Browser (stellen Sie sicher, dass Ihr Browser auf dem neuesten Stand ist)
- Trennen Sie die Verbindung zum VPN, wenn Sie
- Schließe alle zusätzlichen Tabs

Um ein privates LIVE DeepRacer AWS-Event durchzuführen

1. Öffnen Sie die DeepRacer AWS-Konsole.
2. Wählen Sie Community-Rennen.
3. Wähle auf der Seite Community-Rennen die Option Rennen erstellen aus.
4. Entscheide, an welchem Datum und zu welcher Uhrzeit du ein normales LIVE-Community-Rennen veranstalten möchtest.
5. Bevor du den Schritten zur Erstellung eines LIVE-Community-Rennens folgst, überprüfe unter Renndatum, ob dieser Zeitrahmen verfügbar ist. LIVE-Community-Rennen können bis zu vier Stunden dauern. Wenden Sie sich an den Kundensupport, um ein längeres Rennen zu vereinbaren.
6. Wenn Sie sich für ein verfügbares Datum und eine Uhrzeit entschieden haben, erstellen Sie eine entsprechende Videokonferenz für die Organisatoren und Teilnehmer des Rennens. Wenn Sie ein kleines Rennen mit wenig bis gar keinem Publikum veranstalten, ist eine Videokonferenz alles, was Sie benötigen. Wenn Sie ein größeres privates Rennen veranstalten möchten, richten Sie eine weitere Videokonferenz ein, um Ihr Rennen vor einem Publikum zu übertragen.
7. Folgen Sie den Anweisungen unter the section called “Erstelle einen Schnellstart für ein Rennen” und wählen Sie „Um mit der Erstellung eines LIVE-Rennens fortzufahren“ aus.

- Wählen Sie optional in Schritt 8 neben der E-Mail-Vorlage „Vorgeschlagene E-Mail“ die Option Kopieren aus und erstellen Sie eine E-Mail für Rennfahrer und Rennorganisatoren. Geben Sie Ihre Preise, den Zeitrahmen für die Einreichung des Modells und den Link zur Konferenzbrücke ein, über den sich Ihre Rennfahrer treffen, um sich in die Warteschlange zu stellen und sich auf das Rennen vorzubereiten.

8. Folgen Sie am Renntag den Anweisungen zuthe section called “Führen Sie ein LIVE-Rennen durch”.
9. Verteilen Sie gegebenenfalls Preise an die Rennteilnehmer.

Um eine öffentliche LIVE-Übertragung zu produzieren — Dauer: 2 Stunden

Eine Premium-Übertragung eines AWS DeepRacer LIVE-Community-Rennens verwendet mehrere Übertragungsszenen, eine Crew von drei oder mehr Personen, um ein Rennen auf einer globalen Streaming-Plattform zu übertragen. In der folgenden Anleitung wird Twitch als Beispiel verwendet.

Rollen von Organisatoren

- Organisatoren
- Kommentatoren/MC
- Rundfunkveranstalter
- Twitch-Moderator — fakultativ

Hardware (Hardware)

- Empfohlene Hardware: Sie sollten mindestens 16 GB RAM haben
- (Optional) Hochwertige Mikrofone, Headsets oder AirPods
- (Optional) LED-Ringlicht: Um zu vermeiden, dass das Ringlicht auf der Brille reflektiert wird, positionieren Sie es in einem Winkel zum Gesicht des Brillenträgers.
- (Optional) Webcams und GoPros: Verwenden Sie diese, um das Filmmaterial zu diversifizieren.

Tipps

- Verwenden Sie einen Chrome- oder Firefox-Browser (stellen Sie sicher, dass Ihr Browser auf dem neuesten Stand ist).
- Trennen Sie die Verbindung zum VPN, falls Sie eines verwenden.

- Schließen Sie alle zusätzlichen Tabs.

Voraussetzungen

- Twitch-Konto - LIVE-Video-Streaming-Dienst.
- Twitch-Stream-Key — teilt der Software mit, wohin dein Video gesendet werden soll.
- Open Broadcaster Software (OBS) — Kostenlose Open-Source-Software für Videoaufnahmen und LIVE-Streaming.
- (Optional) VDO Ninja (ehemals OBS Ninja) — Tool zum Hinzufügen und Umschalten zu und von zusätzlichen Video-Feeds, wenn Sie Kommentatoren und Interviewpartner einbeziehen möchten.

Um ein öffentliches LIVE DeepRacer AWS-Event durchzuführen

1. Richten Sie ein Twitch-Konto ein, indem Sie den Schritten unter So melden Sie sich für ein Twitch-Konto an.
2. Finde deinen Twitch-Stream-Schlüssel. Erfahre, wie du deinen Twitch-Steam-Schlüssel findest.
3. Laden Sie die Open Broadcaster Software (OBS) herunter.
4. Erfahren Sie, wie Sie mit OBS Ihre Szenen verwalten können. Richte sie im Voraus ein. Wir empfehlen, deine Ausrüstung mindestens eine Woche vor deinem Rennen vorzubereiten:
    a. Laden Sie die mitgelieferten DeepRacer AWS-Szenenvorlagen herunter.
    b. Laden Sie Szenen und ändern Sie sie.
    c. Aktualisiere die Quelle mit deiner Renn-URL.
    d. Überprüfe deine Kameras.
    e. Weisen Sie Personen ihren Feeds zu.
5. Falls Kommentatoren und Interviewpartner Teil deiner Übertragung sind, kannst du optional VDO Ninja (ehemals OBS Ninja) verwenden, um mehrere Video-Feeds zu verwalten. Erfahre, wie du OBS Ninja verwendest.
6. Navigieren Sie zur DeepRacerAWS-Konsole, um ein Rennen zu erstellen.
7. Wählen Sie Community-Rennen aus.
8. Wähle auf der Seite Community-Rennen die Option Rennen erstellen aus.
9. Entscheide, an welchem Datum und zu welcher Uhrzeit du ein öffentliches LIVE-Community-Rennen veranstalten möchtest.

10. Bevor du den Schritten zur Erstellung eines LIVE-Community-Rennens folgst, überprüfe unter Renndatum, ob dieser Zeitrahmen verfügbar ist. LIVE-Community-Rennen haben eine Standarddauer von vier Stunden. Wenden Sie sich an den Kundensupport, um ein längeres Rennen zu vereinbaren. Wenn dein LIVE-Rennen weniger als vier Stunden dauert, kannst du nichts unternehmen.
11. Wenn Sie sich für ein verfügbares Datum und eine verfügbare Uhrzeit entschieden haben, erstellen Sie eine entsprechende Videokonferenz für die Organisatoren und Teilnehmer des Rennens.
12. Erstellen Sie als Nächstes eine weitere Videokonferenz für Ihre Sender.
13. Folgen Sie den Schritten, um ein LIVE-Community-Rennen einzurichten.

    a. Fügen Sie optional in Schritt 8 unter Beschreibung des Rennens den Link für Ihren LIVE-Stream hinzu, den die Rennfahrer mit ihren Familien und Freunden teilen können. Du kannst auch die Konferenzbrücke im Racer Room für Rennfahrer hinzufügen. Die Beschreibung wird in Ihren Bestenlistendetails angezeigt, sodass Sie leicht auf die Links zugreifen können.

    b. Wählen Sie optional in Schritt 12 neben der E-Mail-Vorlage für Vorschläge die Option Kopieren aus und erstellen Sie eine E-Mail für Rennfahrer und Rennorganisatoren. Geben Sie Ihre Preise, den Zeitrahmen für die Einreichung des Modells und den Link zur Konferenzbrücke ein, über den sich Ihre Rennfahrer treffen, um sich in die Warteschlange zu stellen und sich auf das Rennen vorzubereiten.

    c. Erstelle eine weitere E-Mail oder einen Chat für dein Organisationsteam.
14. Folgen Sie am Renntag den Anweisungen the section called “Führen Sie ein LIVE-Rennen durch”
15. Feiere Gewinner und Teilnehmer, verteile Preise, schreibe Blogs, tweete, poste und verbreite.

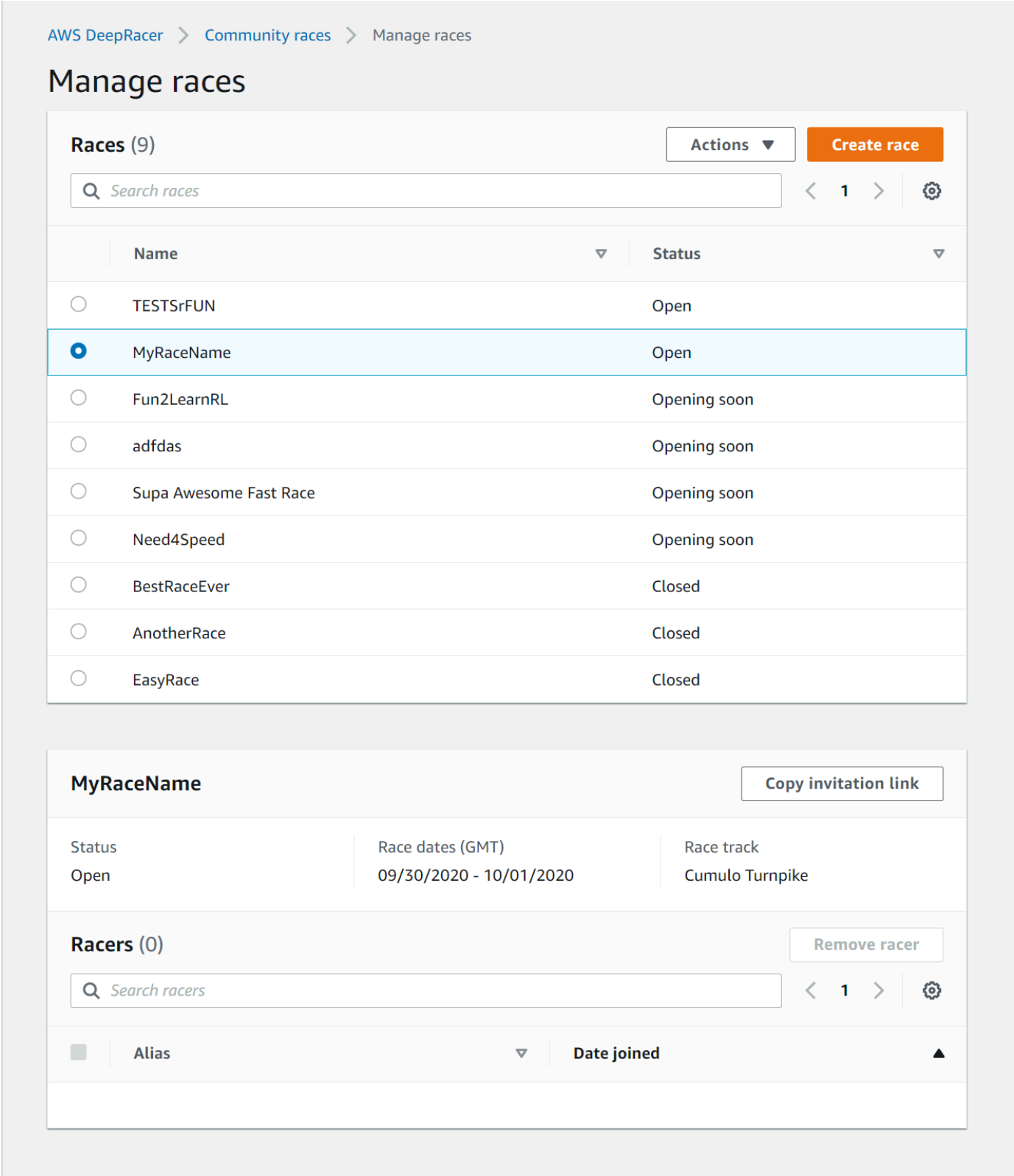
Managen Sie ein DeepRacer AWS-Community-Rennen

Alle Community-Rennen sind privat. Sie sind nur für Personen sichtbar, die über einen Einladungslink verfügen. Teilnehmer können Einladungslinks nach Belieben weiterleiten. Um an einem Rennen teilnehmen zu können, benötigen die Teilnehmer jedoch ein AWS Konto. Erstbenutzer müssen den Kontoerstellungsprozess abschließen, bevor sie am Rennen teilnehmen können.

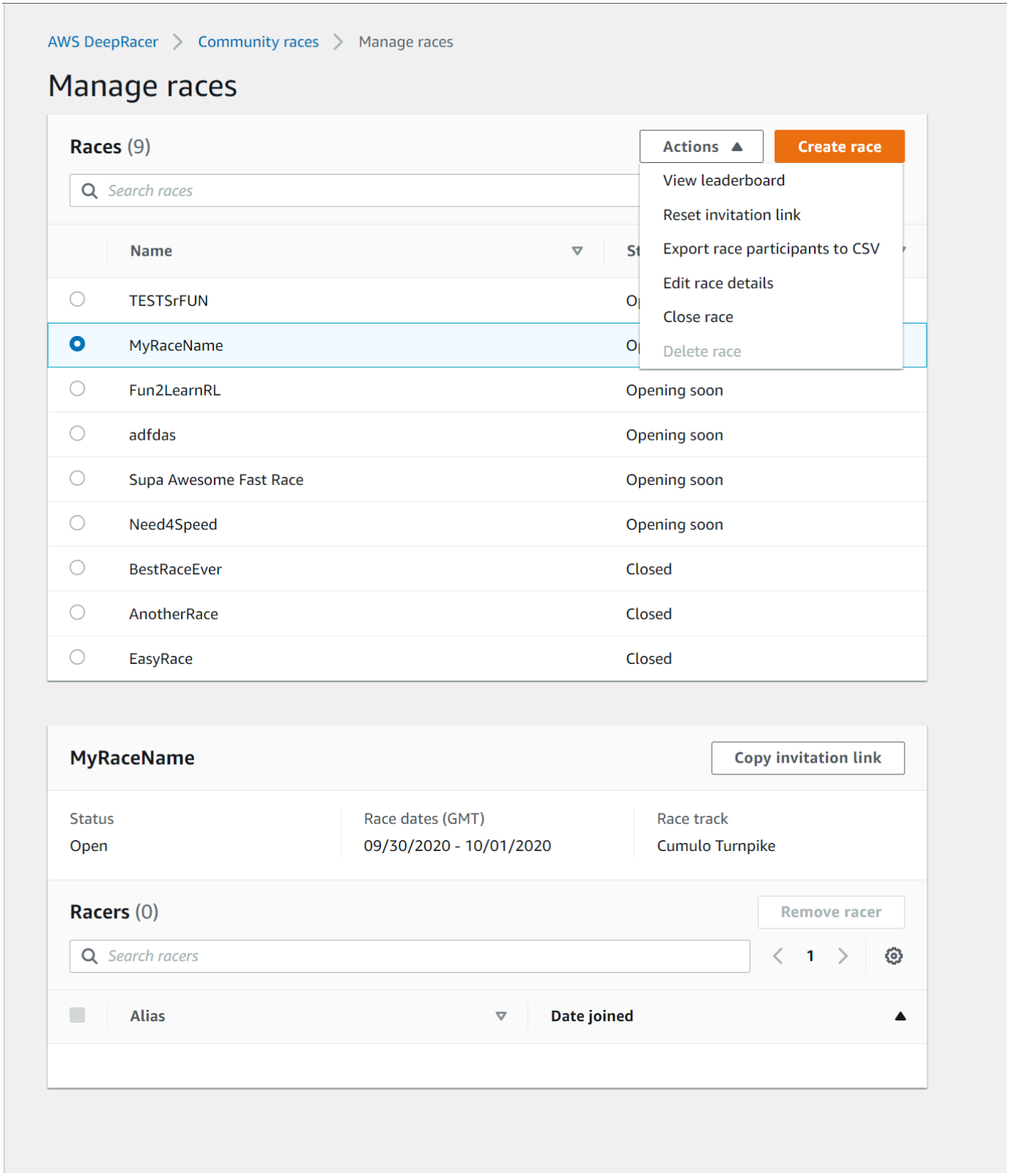
Als Organisator des Rennens können Sie die Renndetails, einschließlich der Start- und Enddaten, bearbeiten und Teilnehmer entfernen.

Um ein DeepRacer AWS-Community-Rennen zu organisieren

1. Melden Sie sich bei der DeepRacer AWS-Konsole an.
2. Wählen Sie Community-Rennen aus.
3. Wähle auf der Seite Rennen verwalten unter Rennen das Rennen aus, das du verwalten möchtest. Die Details des ausgewählten Rennens, einschließlich der Teilnehmerliste, werden angezeigt.



4. Um die Renndetails zu bearbeiten, wählen Sie unter Aktionen die Option Renndetails bearbeiten aus.



Folgen Sie den Anweisungen auf dem Bildschirm, um die Bearbeitung abzuschließen.

5. Um die Bestenliste der Veranstaltung einzusehen, wählen Sie unter Aktionen die Option Bestenliste anzeigen aus.
6. Um den Einladungslink des Events zurückzusetzen, wähle unter Aktionen die Option Einladungslink zurücksetzen aus. Durch das Zurücksetzen des Einladungslinks wird verhindert, dass Personen, die den ursprünglichen Link noch nicht ausgewählt haben, auf das Rennen zugreifen können. Alle Benutzer, die bereits auf den Link geklickt und ein Modell eingereicht haben, bleiben im Rennen.

   Du kannst den Link auch kopieren, um ihn mit eingeladenen Teilnehmern zu teilen.
7. Um ein offenes Rennen zu beenden, wähle unter Aktionen die Option Rennen schließen aus. Dadurch wird das Rennen unmittelbar vor dem angegebenen Schlusstermin beendet.
8. Um die Veranstaltung zu löschen, wählen Sie unter Aktionen die Option Rennen löschen. Dadurch werden dieses Rennen und die Details dauerhaft aus den Community-Rennen aller Teilnehmer entfernt.
9. Um einen Teilnehmer zu entfernen, wählen Sie einen oder mehrere Rennteilnehmer aus, wählen Sie Remove participants (Teilnehmer entfernen) und bestätigen Sie dann, dass der Teilnehmer entfernt werden soll.

   Das Entfernen eines Teilnehmers aus einer Veranstaltung widerruft die Berechtigung des Benutzers zum Zugriff auf die Rennveranstaltung.

Organisieren Sie eine DeepRacer AWS-Veranstaltung

Was ist eine DeepRacer AWS-Veranstaltung?

AWS DeepRacer ist ein Bildungsservice, der eine unterhaltsame Möglichkeit bietet, praktisches Lernen mit künstlicher Intelligenz und maschinellem Lernen (ganz AI/ML). AWS DeepRacer can help bridge the AI/ML talent gap for your organization and apply AI/ML nach Ihren Geschäftsanforderungen) zu erlernen.

AWS stellt DeepRacer nicht nur AI/ML skills to your team, it also allows you to host events to encourage team building and friendly competition. These events help both technical and non-technical participants learn the fundamentals of machine learning by providing hands-on experience with creating reinforcement learning models to race AWS DeepRacer cars in-person or virtually in the AWS DeepRacer League. AWS DeepRacer events also help leaders engage their teams to reach their organization's AI/ML Visionen und Ziele vor.

Dieser Leitfaden bietet Ressourcen, Tools und Beispiele, die Ihnen helfen, mit der Planung und Ausrichtung Ihrer eigenen virtuellen oder persönlichen DeepRacer AWS-Veranstaltungen zu beginnen. Wenn Sie Ihre DeepRacer AWS-Veranstaltung mit 50 Teilnehmern oder weniger planen möchten, gehen Sie zu Was Sie beachten sollten, bevor Sie beginnen. Wenn Sie eine größere Veranstaltung (mit mehr als 50 Teilnehmern) planen, empfehlen wir, mit Ihrem AWS Account-Team zusammenzuarbeiten und eine Veranstaltung zu beantragen.

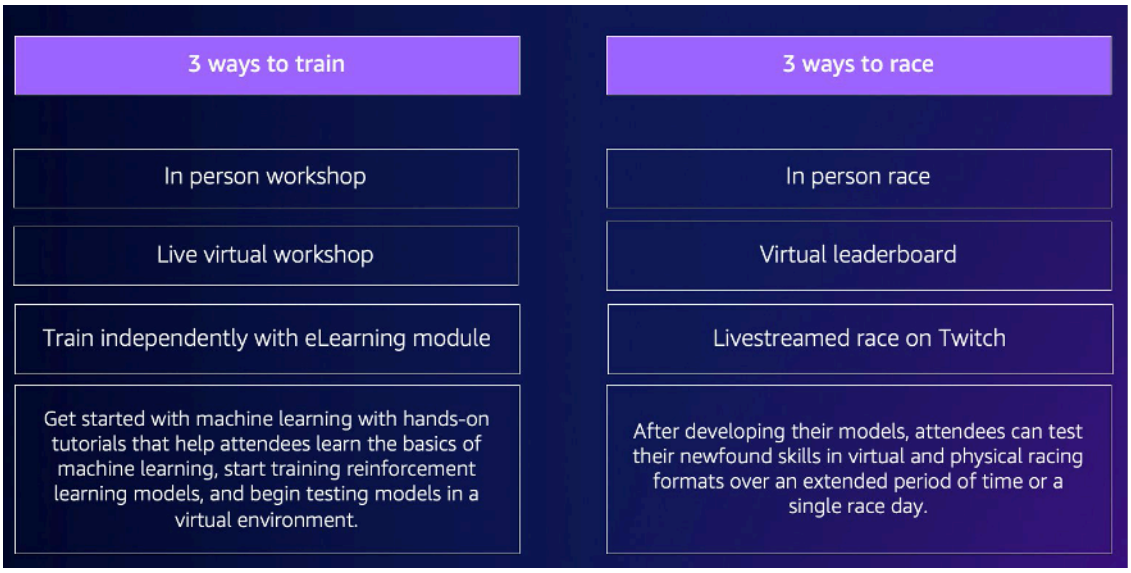
Weitere Informationen zu den Vorteilen von DeepRacer AWS-Veranstaltungen und Kundenreferenzen finden Sie unter AWS-Veranstaltungen für DeepRacer Unternehmen.

So funktionieren DeepRacer AWS-Veranstaltungen und was Sie erwartet

Ganz gleich, ob Sie Ihrem Team Schulungen und praktische Übungen mit Reinforcement-Learning anbieten, Ihr Unternehmen fördern möchten, um neue Talente anzuziehen, oder eine Kombination aus beidem möchten, dieser Leitfaden bietet die Tools und Ressourcen, mit denen Sie Ihre eigene DeepRacer AWS-Veranstaltung erstellen und anpassen können.

DeepRacer AWS-Veranstaltungen können flexibel an Ihre Bedürfnisse und Ziele angepasst werden. Die Grundformel für eine DeepRacer AWS-Veranstaltung umfasst jedoch:

- Eine pädagogische Komponente, z. B. ein DeepRacer Online-AWS-Reinforcement-Learning-Kurs oder ein persönlicher Workshop mit einem AWS DeepRacer Pit Crew-Experten.
- Praktische Modellschulung in der AWS Konsole und Zeit für die Teilnehmer, ihre Modelle zu erstellen und zu trainieren.
- Ein DeepRacer AWS-Rennen, z. B. ein einzelnes persönliches Rennen oder mehrere virtuelle Rennen, sodass die Teilnehmer ihre trainierten Modelle für maschinelles Lernen in Aktion sehen können. Weitere Informationen zu den Arten von Rennen, die Sie veranstalten können, finden Sie unter Arten von DeepRacer AWS-Rennen.
- Eine Zusammenfassung nach der Veranstaltung oder eine Mitteilung über die nächsten Schritte für die Veranstaltungsteilnehmer.



Weitere Informationen darüber, was Sie von DeepRacer AWS-Veranstaltungen und Kundenhighlights erwarten können, finden Sie im AWS DeepRacer YouTube Accelerate-Video.

Was ist zu beachten, bevor Sie beginnen

Der erste Schritt bei der Planung Ihrer Veranstaltung besteht darin, Ihre Geschäftsziele und -ziele für Ihr Unternehmen zu definieren und anschließend einen Projektplan zu entwickeln. Ein Beispiel für ein Ziel ist: „Ich möchte die Teambildung in meiner Organisation auf unterhaltsame und lehrreiche Weise fördern.“

Ihr Projektplan sollte die folgenden Fragen beantworten:

- Wer sind Ihre Eventleiter?

Finden Sie heraus, wer in Ihrer Organisation bei der Planung und Durchführung dieser Veranstaltung mitwirkt. Zu den Veranstaltungsleitern können Teams wie Einrichtungen, Personalabteilung, PR/Marketing, ein leitender Sponsor oder AWS ein Account-Team gehören

- Wer sind die Eigentümer der Veranstaltung?

  Identifizieren Sie, wem in Ihrer Organisation die Veranstaltung gehört. Diese Person oder dieses Team sollte der Hauptorganisator sein und bei der Planung dieser Veranstaltung an allen Entscheidungen beteiligt sein.

- Was ist das Veranstaltungstermin?

  Geben Sie an, wann Sie die Veranstaltung abhalten möchten. Wenn Sie eine große Veranstaltung (mit mehr als 100 Teilnehmern) in Betracht ziehen, müssen Sie Ihren Veranstaltungstermin später planen, um genügend Vorlaufzeit für die Planung und Durchführung der Veranstaltung zu haben. Ein Beispiel für einen Zeitplan finden Sie in den DeepRacer AWS-Veranstaltungsbeispielen.

- Wie hoch ist das geschätzte Budget?

  Schätzen Sie ein Budget für Ihre Veranstaltung. Es gibt zwei Kostenaspekte, die Sie bei einer Veranstaltung berücksichtigen sollten.

  - Die Kosten der Veranstaltung: Wenn Sie eine persönliche Veranstaltung veranstalten, kann dies alles umfassen, vom Veranstaltungsort über die Strecke und den Kauf von Geräten bis hin zu Speisen, Getränken und Veranstaltungspreisen.
  - Kosten für Modellschulungen: Diese Kosten entstehen, wenn Ihre Mitarbeiter Reinforcement-Learning-Modelle zur AWS Management Console Nutzung des DeepRacer AWS-Service schulen. Beispielsweise benötigt jeder Teilnehmer ungefähr 10 Stunden, um ein oder mehrere Modelle in der DeepRacer AWS-Konsole zu trainieren. Weitere Informationen zu den Kosten finden Sie unter DeepRacerAWS-Preise.

- Welcher Ort ist für die Veranstaltung erforderlich?

  Wenn Sie eine persönliche Veranstaltung abhalten, müssen Sie einen physischen Ort für den Workshop und das Rennen finden. Die Art der physischen Strecke, die Sie auswählen, muss ebenfalls zum Ort passen.

- Wie hoch ist die geschätzte Anzahl der Teilnehmer?

  Schätzen Sie die Anzahl der Teilnehmer, die an Ihrer Veranstaltung teilnehmen. Die Anzahl der Teilnehmer wird auch in Ihr geschätztes Budget einfließen.

- Planen Sie, nach der Veranstaltung eine Retrospektive abzuhalten, um den Erfolg zu messen?

Um die Teilnehmer zu ermutigen, ihre ML-Fähigkeiten weiter auszubauen und weiter zusammenzuarbeiten, sollten Sie erwägen, die nächsten Schritte mitzuteilen und die Teilnehmer um Feedback zu bitten. Senden Sie beispielsweise eine Umfrage an die Teilnehmer, um das Interesse zu messen und KI/ML-Führungskräfte innerhalb der Organisation zu rekrutieren. Dies kann auch bestimmen, wer an future DeepRacer AWS-Veranstaltungen beteiligt ist.

- Was ist die Kommunikationsstrategie, um die Veranstaltung in Ihrer Organisation bekannt zu machen?

  Die Kommunikation dieser Veranstaltung innerhalb Ihrer Organisation kann so einfach sein wie eine interne E-Mail oder eine Kalendereinladung.

- Wer sind die Stakeholder der Geschäftsleitung?

  Identifizieren Sie die Führungskräfte, die die Veranstaltung sponsern und die Vordenkerrolle und Zusammenarbeit innerhalb Ihres Unternehmens fördern.

Arten von DeepRacer AWS-Rennen

Nachdem Sie die wichtigsten Fragen für Ihren Projektplan beantwortet haben, entscheiden Sie, welche Art von Rennen Sie auf Ihrer Veranstaltung haben möchten. Die Art des Rennens, das Sie veranstalten, sollte auf Ihrem geschätzten Budget und der Anzahl der Teilnehmer für die Veranstaltung basieren. Sie können die folgenden zwei Arten von Rennen ausrichten.

Virtuelles Rennen

Virtuelle Rennen sind eine hervorragende Möglichkeit für Teams mit mehreren Regionen oder an entfernten Standorten, zusammenzuarbeiten und ihre trainierten Modelle in einer virtuellen Umgebung gegeneinander anzutreten. Da virtuelle Rennen keine physischen Geräte und Strecken erfordern, bieten sie Flexibilität bei der Gruppengröße und sind eine hervorragende Lösung für Unternehmen mit Standort- und Budgetbeschränkungen.

Es gibt zwei Arten von virtuellen Rennen: klassische Rennen und Live-Rennen. Der Hauptunterschied zwischen klassischen Rennen und Live-Rennen besteht darin, dass die Teilnehmer an klassischen Rennen ihre Modelle trainieren und sie in ihrer Freizeit einreichen können. Bei Live-Rennen fahren die Teilnehmer innerhalb einer bestimmten Zeit nacheinander mit ihren Modellen. Beide Arten von Rennen können privat oder öffentlich geteilt werden und verwenden Bestenlisten, um die Rangliste der Teilnehmer zu ermitteln. AWS bietet auch Tools, mit denen du deine Rennen auf Twitch übertragen kannst.

Virtuelle Rennen sind der schnellste Weg, um mit DeepRacer AWS-Rennen zu beginnen. Die Teilnehmer können den Prozess „Erste Schritte“, der eine Einführung in maschinelles Lernen und Reinforcement-Learning beinhaltet, in der Konsole durchführen. Die Konsole führt die Teilnehmer durch die Erstellung ihres ersten Modells. Weitere Informationen zum Einrichten eines virtuellen Rennens finden Sie unter Rennen erstellen.

Rennen vor Ort

Veranstalten Sie ein persönliches Rennen, um eine ansprechende Veranstaltung zu bieten, an der Ihr Team persönlich teilnehmen kann. Zu den Präsenzrennen gehören eine physische Strecke und DeepRacer AWS-Fahrzeuggeräte. AWS bietet Optionen für den Bau Ihrer Strecke und die Beschaffung von Fahrzeugen für Ihre Veranstaltung.

Weitere Informationen zu DeepRacer AWS-Geräten finden Sie unter Betrieb Ihres DeepRacer AWS-Fahrzeugs. Weitere Informationen zum Bau und zur Bestellung Ihres physischen Gleises und der Barrieren finden Sie unter Aufbau Ihres physischen Gleises.

Zusätzlich zu diesen beiden Arten von Rennen kannst du auch lehrreiche Online-Schulungen oder Workshops anbieten und deine Rennveranstaltungen auf Twitch live streamen.

Bewährte Methoden

Um Ihre DeepRacer AWS-Veranstaltungsplanung noch effizienter zu gestalten, empfehlen wir außerdem die folgenden Best Practices.

- Holen Sie sich Unterstützung von internen Führungskräften. Die Support durch Organisationsleiter fördert die Teilnahme des Teams und erhöht das Engagement und die allgemeine Zufriedenheit der Teilnehmer.
- Wenn Sie einen AWS Kundenbetreuer haben, kontaktieren Sie ihn, um ihn über Ihre Veranstaltung zu informieren und zu besprechen, ob Sie Unterstützung benötigen.
- Wenn Sie eine große Veranstaltung (mit mehr als 50 Teilnehmern) planen, wenden Sie sich frühzeitig und häufig an Ihre AWS Kundenbetreuer. Reichen Sie eine Anfrage ein, wenn Sie sich nicht sicher sind, an wen Sie sich in Ihrem AWS Account-Team wenden sollen. Informationen zum Einreichen einer Anfrage finden Sie unter Veranstaltung anfragen. Weitere Ressourcen der DeepRacer AWS-Community zu Veranstaltungen und Schulungen finden Sie im DeepRacer AWS-Slack-Kanal.
- Legen Sie ein Budget fest, das Ihren DeepRacer AWS-Veranstaltungszielen entspricht. Ihr Budget kann sich auf die Art der Veranstaltung auswirken, die Sie planen, aber AWS DeepRacer bietet Tools wie den Mehrbenutzermodus, mit denen Sie die Kosten begrenzen und verwalten können.

- Bieten Sie Ihren Teilnehmern Preise an. Egal, ob Ihr Team am traditionellen Meisterschaftspokal interessiert ist oder ob Ihre Preise attraktiv sind NFTs, um die Teilnahme zu fördern.

Erste Schritte mit Ihrer DeepRacer AWS-Veranstaltung

Sobald Sie die Ziele Ihrer Organisation definiert haben, können Sie anhand Ihres Projektplans beginnen, die Art der Veranstaltung, die Sie abhalten möchten, einzugrenzen. Die folgenden Beispielziele zeigen, wie Sie eine Veranstaltung auf der Grundlage Ihrer Anforderungen und der Vorteile, die Sie von AWS profitieren möchten, einrichten können DeepRacer.

Teambildung

Wenn Sie eine einmalige lokale Veranstaltung veranstalten möchten, die die Teambildung für kleinere Gruppen fördert, sollten Sie eine persönliche oder virtuelle Veranstaltung in Betracht ziehen. Ein Beispiel für die Art der Veranstaltung, mit der dieses Ziel erreicht wird, finden Sie unter Beispiele für virtuelle Veranstaltungen.

In KI/ML-Bildung investieren

Wenn Sie möchten, dass sich Ihre technischen und nicht-technischen Mitarbeiter besser mit maschinellem Lernen vertraut machen und diese Fähigkeiten anwenden, sollten Sie erwägen, mehr Unterstützung von Führungskräften in Anspruch zu nehmen und darüber nachzudenken, Ihre Veranstaltung zu einer teamübergreifenden Veranstaltung zu machen, an der mehr Teilnehmer teilnehmen, um eine größere organisatorische Wirkung zu erzielen. Diese Veranstaltung kann sowohl Präsenzveranstaltungen als auch virtuelle Veranstaltungen beinhalten, um eine Skalierung der Teilnehmer zu ermöglichen. Ein Beispiel für die Art der Veranstaltung, mit der dieses Ziel erreicht wird, finden Sie unter Beispiele für Präsenzveranstaltungen.

Förderung und Sensibilisierung der Organisation

Wenn Sie Ihr Unternehmen als innovativ und als Vordenker positionieren möchten, um mehr Talente anzuziehen und das allgemeine Bewusstsein in Ihrem Unternehmen zu fördern, sollten Sie erwägen, Ihre virtuelle oder persönliche Veranstaltung per Livestream zu streamen oder eine maßgeschneiderte Veranstaltung zu organisieren. Ein Beispiel für die Art der Veranstaltung, mit der dieses Ziel erreicht werden kann, finden Sie unter Beispiel für eine benutzerdefinierte Veranstaltung.

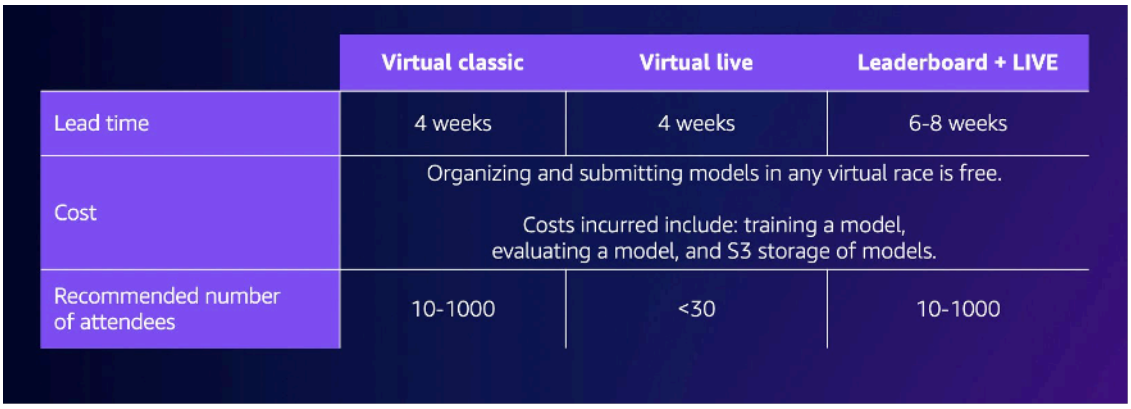
Beispiele für DeepRacer AWS-Ereignisse

In den folgenden Abschnitten finden Sie einige Beispiele für die verschiedenen Arten von Veranstaltungen, die Sie auf der Grundlage Ihrer Ziele und Anforderungen an Ihren Projektplan

erstellen können. Diese Veranstaltungszeitpläne werden auf der Grundlage der Bildungs- und Rennkomponenten geplant. Sie können Ihren Veranstaltungszeitplan jedoch für eine beliebige Anzahl von Veranstaltungskomponenten an Ihre organisatorischen Anforderungen anpassen.

Beispiele für virtuelle Veranstaltungen

Virtuelle Veranstaltungen sind eine hervorragende Möglichkeit für Organisationen an mehreren Standorten oder multiregionale Teams, sich auf bequeme und kostengünstige Weise zu treffen. Virtuelle Veranstaltungen sind praktischer und kostengünstiger, da sie weniger Abhängigkeiten aufweisen. Sie müssen beispielsweise nicht wie bei einer Präsenzveranstaltung physische Tracks, Geräte oder einen Ort in Betracht ziehen. Die folgenden Beispiele für virtuelle Veranstaltungen konzentrieren sich auf verschiedene Punkte im Projektplan, z. B. den Zeitplan und die Anzahl der Teilnehmer.



	Virtual classic	Virtual live	Leaderboard + LIVE
Lead time	4 weeks	4 weeks	6-8 weeks
Cost	Organizing and submitting models in any virtual race is free. Costs incurred include: training a model, evaluating a model, and S3 storage of models.		
Recommended number of attendees	10-1000	<30	10-1000

- Zweiwöchiges virtuelles klassisches Rennen:

  Das folgende Beispiel für einen zweiwöchigen Veranstaltungsplan ist eine hervorragende Option für Organisationen, die nach einer einfachen Möglichkeit suchen, eine Veranstaltung mit vielen Teilnehmern und Teams auszurichten, die am Rennen teilnehmen möchten, da es keine Begrenzung für die Einreichung klassischer Rennen und keine Zeitbeschränkungen für Live-Rennen gibt. Alle Teilnehmer können ihre Modelle jederzeit, auch gleichzeitig, innerhalb des Rennfensters einreichen. In diesem Beispiel für ein Event wird der Bereich „Community-Rennen“ in der AWS Konsole verwendet, um ein privates klassisches Rennen zu erstellen, und das Training und die Rennen werden in zwei verschiedenen Wochen geplant. Wie du ein Community-Rennen einrichtest, erfährst du unter Rennen verwalten.

  - Tag 1: Die Teilnehmer nehmen an einem kostenlosen virtuellen On-Demand-Workshop für alle Teilnehmer teil. Weitere Informationen zur Online-Workshop-Ressource finden Sie unter AWS DeepRacer: Driven by Reinforcement Learning. Sie können auch einen AWS DeepRacer Pit Crew-Experten mit der Durchführung eines virtuellen Workshops vereinbaren.

  - Tage 1—5: Die Teilnehmer trainieren, aktualisieren und testen ihre Modelle in der DeepRacer AWS-Konsole. Sie reichen ihre Modelle ein, um an den Tagen 1-5 an einem privaten virtuellen Rennen teilzunehmen. Informationen zum Einrichten der AWS Konsole für mehrere Teilnehmer unter einem AWS Konto finden Sie unter Mehrbenutzermodus.
  - Tage 5 — 10: Die zehn besten Gewinner werden am 5. Tag ermittelt und erhalten in der darauffolgenden Woche Zugang zu einem neuen privaten virtuellen Rennen auf einer neuen Strecke.
  - Tag 10: Die drei besten Gewinner werden am 10. Tag ermittelt und das Rennen und die Veranstaltung sind abgeschlossen.
- Einwöchiges virtuelles Live-Rennen:

  Das folgende Beispiel für ein virtuelles Live-Rennen ist eine großartige Möglichkeit, kleineren Teams die Spannung und den Spaß am Rennsport in Echtzeit näher zu bringen. Diese Art von Veranstaltung ist kostengünstig und ermöglicht es jedem, in Echtzeit Rennen zu fahren. Weitere Informationen zur Durchführung eines Live-Rennens findest du unter Ein LIVE-Rennen veranstalten.

  - Tag 1: Die Teilnehmer nehmen an einem virtuellen Workshop mit einem AWS DeepRacer Pit Crew-Experten teil. Weitere Informationen zu Workshops finden Sie unter DeepRacerAWS-Veranstaltungen.
  - Tage 1—5: Die Teilnehmer trainieren, aktualisieren und testen ihre Modelle an den Tagen 1—5 in der DeepRacer AWS-Konsole.
  - Tag 5: Die Teilnehmer treffen sich für 1—2 Stunden online, um ihre Modelle einzureichen und an Tag 5 an einem virtuellen Live-Rennen teilzunehmen.
- Virtuelles klassisches Rennen und Live-Rennveranstaltung:

  Das folgende Beispiel für eine Veranstaltung ist eine hervorragende Option für Organisationen, die viele multiregionale Teilnehmer oder Teilnehmer zusammenbringen möchten, die aus der Ferne in mehreren Büros online arbeiten. Bei dieser Art von Veranstaltung haben Ihre Teilnehmer mehr Möglichkeiten, das Training ihrer Models und Rennen zu üben, da die Veranstaltung über zwei Wochen verteilt ist. Wir empfehlen außerdem, bei Ihren Live-Rennveranstaltungen einen Moderator zu haben, um Ihre Veranstaltung für die Teilnehmer, die den Livestream verfolgen, spannender zu gestalten. Schaut euch den Twitch-Stream zum Finale der Pro Division an, um zu sehen, wie die Finale-Übertrager das Rennen noch spannender machen.

- Tag 1: Die Teilnehmer nehmen an einem virtuellen Workshop mit einem AWS DeepRacer Pit Crew-Experten teil. Weitere Informationen zu Workshops finden Sie unter DeepRacerAWS-Veranstaltungen.
- Tage 1-5: Die Teilnehmer trainieren, aktualisieren und testen ihre Modelle in der DeepRacer AWS-Konsole und reichen dann Modelle ein, um an den Tagen 1-5 an einem privaten Ligarennen teilzunehmen.
- Tag 5: Die zehn besten Gewinner werden ermittelt und erhalten Zugang zu einem neuen privaten Ligarennen auf einer neuen Strecke.
- Tage 5 — 10: In der darauffolgenden Woche werden die zehn besten Gewinner ermittelt. Sie nehmen an den Rennen teil, während das Event live auf Twitch gestreamt wird. Weitere Informationen findest du unter Live-Übertragung eines Rennens.

Beispiele für Veranstaltungen vor Ort

Für Unternehmen, die ihre Teilnehmer an einem einzigen physischen Ort versammeln können, gibt es keine bessere Möglichkeit, AWS zu erleben DeepRacer als eine persönliche Veranstaltung. Nichts geht über den Nervenkitzel und die Aufregung, an der Strecke zu stehen und zu sehen, wie Ihr Modell auf einem echten Auto funktioniert. Im Allgemeinen erfordern persönliche Veranstaltungen mehr Ressourcen und sind teurer als virtuelle Veranstaltungen. Für jede Organisation mit mehr als 50 Teilnehmern und dem erforderlichen Budget empfehlen wir dringend, die Vorteile der DeepRacer AWS-Erfahrung vor Ort zu nutzen. Wir empfehlen außerdem, bei diesen Veranstaltungen einen Moderator anwesend zu haben, um Ihre Rennen noch spannender zu gestalten, da die Wettbewerbsergebnisse in Echtzeit für die Teilnehmer bekannt gegeben werden, die sie von der Rennstrecke aus verfolgen können.

	One-day workshop + race	Two-day workshop + race	Custom format
Lead time	6+ weeks	6+ weeks	10+ weeks
Recommended number of attendees	51–100	100–500	500–1,000

- Eintägige Veranstaltung vor Ort:

  Das folgende Beispiel einer Präsenzveranstaltung eignet sich hervorragend für lokale Teams und stellt sicher, dass alle Teilnehmer ihr erstes Modell an einem Tag trainieren AWS Management

Console und an einem Tag teilnehmen können. Eine typische eintägige Präsenzveranstaltung besteht aus einem persönlichen Workshop und einem Rennen für alle Teilnehmer. Der Workshop wird in der Regel von einem AWS DeepRacer Pit Crew-Experten geleitet, der sich mit den Grundlagen des Reinforcement-Learnings befasst und den Teilnehmern die Möglichkeit gibt, ihr erstes Modell zu trainieren. Im Anschluss an den Workshop können die Teilnehmer ihre Modelle einreichen und auf ein DeepRacer AWS-Gerät hochladen und am selben Tag Rennen fahren.

  - Die Teilnehmer nehmen an einem persönlichen DeepRacer AWS-Workshop teil, der von einem AWS DeepRacer Pit Crew-Experten abgehalten wird (90-120 Minuten). Weitere Informationen zur Anfrage von DeepRacer AWS-Workshops finden Sie unter Veranstaltung anfragen.
  - Persönliches Rennen im Anschluss an den Workshop (120 Minuten oder ~5 Minuten pro Rennen).
- Zweitägige Veranstaltung vor Ort:

  Das folgende Beispiel für eine zweitägige Präsenzveranstaltung ähnelt der eintägigen Veranstaltung, mit dem Unterschied, dass der Workshop und das eigentliche Rennen auf verschiedene Tage verteilt werden, sodass die Teilnehmer mehr Zeit haben, ihre Modelle zu trainieren und auf den neuesten Stand zu bringen, während sie sich auf das Rennen vorbereiten. Es ist üblich, dass Organisationen diese Veranstaltungen im Abstand von einigen Tagen oder sogar Wochen veranstalten, um den Teilnehmern mehr Zeit zu geben, ihre Modelle zu trainieren und zu verfeinern. Wenn mehr Zeit zwischen Workshops und Rennen zur Verfügung steht, können die Teilnehmer einen wettbewerbsfähigeren Renntag haben.

  - Tag 1: Die Teilnehmer nehmen an einem persönlichen DeepRacer AWS-Workshop teil, der von einem AWS DeepRacer Pit Crew-Experten abgehalten wird (90-120 Minuten). Weitere Informationen zur Anfrage für persönliche Workshops finden Sie unter Veranstaltung anfragen.
  - Tag 2: Die Teilnehmer nehmen im Anschluss an den Workshop an einem persönlichen Rennen teil (120 Minuten oder ungefähr 5 Minuten pro Rennen).

Beispiel für ein benutzerdefiniertes Ereignis

Wie bei virtuellen Veranstaltungen sind benutzerdefinierte Veranstaltungen eine hervorragende Option für größere Organisationen (100 Teilnehmer oder mehr), die eine Veranstaltung für Teams an mehreren Standorten veranstalten müssen. Mit benutzerdefinierten Veranstaltungen können Sie bei Schulungen, Workshops und Rennen flexibler vorgehen, da es keine zeitlichen und formalen Beschränkungen gibt. Sie können sowohl virtuelle als auch persönliche Rennen in diese Veranstaltungen einbeziehen. Diese Rennen können sich über mehrere Wochen erstrecken, damit die Teilnehmer aus aller Welt mehr Zeit damit verbringen können, ihre Modelle zu trainieren und

zusammenzuarbeiten. Diese Art von maßgeschneiderter Veranstaltung ist erfolgreicher, wenn Sie zuvor einige kleinere Veranstaltungen durchführen, um sich auf mögliche logistische Probleme vorzubereiten. Diese Art von Veranstaltung oder Veranstaltungsreihe trägt auch dazu bei, ein Team von Verfechtern des maschinellen Lernens in Ihrer eigenen Organisation aufzubauen.

- Im folgenden Beispiel erstreckt sich die benutzerdefinierte Veranstaltung über drei Monate, um Mitarbeitern aus mehreren Regionen gerecht zu werden.
  - Monat 1: Weltweite Workshops an mehreren Standorten.
  - Monat 2: Einmonatige Qualifikationsrennen für virtuelle Ligen, die Teilnehmern in mehreren Regionen zur Verfügung stehen.
  - Monat 3: Persönliches Rennen und virtuelles Championship-Cup-Rennen. Dieses persönliche Rennen kann live gestreamt werden, sodass Ihre weltweiten Teams es verfolgen können.

Wenn Sie daran interessiert sind, eine maßgeschneiderte Veranstaltung auszurichten, wenden Sie sich an, um Unterstützung bei der Veranstaltung AWS zu erhalten. Weitere Informationen finden Sie unter Veranstaltung anfragen.

Weitere Ressourcen

Weitere Ressourcen zu DeepRacer AWS-Veranstaltungen finden Sie in der folgenden Liste:

- DeepRacer AWS-Blog
- DeepRacer AWS-Liga
- DeepRacer Slack-Kanal der AWS-Community
- Schulung zum maschinellen Lernen
- Zertifizierung für maschinelles Lernen
- DeepRacer AWS-Schulung
- DeepRacer GitHub AWS-Repository
- Tipps zum Rennsport
- DeepRacer YouTube AWS-Kanal
- DeepRacer AWS-Storefront

Mehrbenutzermodus

Die Kontoeinrichtung im Mehrbenutzermodus bietet Unternehmen eine spannende Möglichkeit, mehrere DeepRacer AWS-Teilnehmer unter einem AWS Konto zu sponsern. Für gesponserte Teilnehmer fallen keine eigenen Kosten an. Stattdessen werden ihre Schulungsstunden und Lagerkosten dem Sponsorkonto in Rechnung gestellt. AWS Mit der Einrichtung eines Kontos im Mehrbenutzermodus können DeepRacer AWS-Veranstalter Budgets festlegen und Ausgaben überwachen und kontrollieren, indem sie die Standardkontingente für Trainingszeiten und -modelle für einzelne Teilnehmer, Gruppen oder für alle Teilnehmer aktualisieren.

In den folgenden Abschnitten wird beschrieben, wie Sie den DeepRacer AWS-Mehrbenutzermodus entweder als Administrator oder als Teilnehmer nutzen können.

> **Note**
>
> Der Mehrbenutzermodus mit Kontosponsoring ist nur im Service verfügbar. AWS DeepRacer

Themen

- Mehrbenutzermodus einrichten (admin)
- DeepRacer AWS-Mehrbenutzererlebnis (Teilnehmer)

Mehrbenutzermodus einrichten (admin)

Bei der Einrichtung eines Mehrbenutzerkontos können Organisatoren (z. B. Kontoadministratoren) Teilnehmern unter ihrer Konto-ID Zugriff auf den DeepRacer AWS-Service gewähren. Sie können auch Nutzungskontingente für die Trainingsstunden der Teilnehmer festlegen, die Ausgaben für Schulung und Speicherplatz überwachen, Schulungen starten und beenden sowie Modelle für jeden Benutzer in ihrem Konto von der DeepRacer AWS-Konsole aus anzeigen und verwalten.

Der Mehrbenutzermodus ist besonders nützlich für große Veranstaltungen mit mehreren Teilnehmern, die keine individuellen Konten haben. AWS Anstatt Konten für jeden Teilnehmer einer Veranstaltung zu erstellen und zu verwalten, kann ein DeepRacer AWS-Administrator alle seine gesponserten Teilnehmer über ein einziges AWS Konto hosten.

Im Mehrbenutzermodus können gesponserte Teilnehmer an Wettkämpfen teilnehmen und trainieren, ohne dass ihnen eigene Kosten entstehen. Ihre Schulungs- und Lagergebühren werden mit

der Abrechnung des Mehrbenutzerkontos der Sponsoren in Rechnung gestellt. AWS Wenn ein Administrator aufhört, die Nutzung der Teilnehmer zu sponsern, behalten die Teilnehmer ihre Racer-Aliase und -Profile.

Stakeholder, die mehrere Benutzer verwenden

Diese exemplarische Vorgehensweise bezieht sich auf die folgenden typischen Mehrbenutzer-Stakeholder für die Einrichtung und Verwendung des Mehrbenutzermodus.

- AWS Administrator für die IAM/SSO-Konfiguration. Der AWS Administrator für die IAM/SSO-Konfiguration richtet IAM oder SSO ein, damit der DeepRacer AWS-Administrator und die Teilnehmer den Mehrbenutzermodus verwenden können. Der AWS Administrator für IAM/SSO verfügt über IAM- und SSO-Administratorrechte. Informationen zum Erstellen von IAM-Benutzern finden Sie unter Einen IAM-Benutzer in Ihrem AWS-Konto erstellen.
- DeepRacer AWS-Administrator. Der DeepRacer AWS-Administrator verwaltet das Sponsoring DeepRacer der AWS-Teilnehmer und kann das Sponsoring unterbrechen und wieder aufnehmen, Modelle und Artefakte löschen, virtuelle Rennen konfigurieren und veranstalten sowie den Mehrbenutzermodus aktivieren und deaktivieren. Der DeepRacer AWS-Administrator hat AWSDeepRacerAccountAdminAccessBerechtigungen.
- DeepRacer AWS-Teilnehmer. DeepRacer AWS-Teilnehmer sind eingeladen, unter einem AWS Administratorkonto im Mehrbenutzermodus an Veranstaltungen teilzunehmen. Die Teilnehmer sind AWSDeepRacerDefaultMultiUserAccessberechtigt, Modelle im Konto des Sponsors zu trainieren, zu testen und zu speichern. Die Teilnehmer konfigurieren auch ihr Rennfahrerprofil, nehmen an virtuellen Rennen teil und laden ihre Modelle für den Einsatz auf einem physischen DeepRacer AWS-Fahrzeug herunter.

In dieser exemplarischen Vorgehensweise führen Sie die folgenden Schritte aus:

- Schritt 1. Führen Sie die Voraussetzungen aus.
- Schritt 2. Aktivieren Sie den Mehrbenutzermodus auf Ihrem DeepRacer AWS-Konto.
- Schritt 3. Laden Sie Teilnehmer ein.
- Schritt 4. Legen Sie Nutzungskontingente fest.
- Schritt 5. Überwachen Sie die Nutzung Ihrer gesponserten Teilnehmer.

Schritt 1. Voraussetzungen für den DeepRacer AWS-Mehrbenutzermodus

Erfüllen Sie die folgenden Voraussetzungen für den Mehrbenutzermodus

- Richten Sie Ihr Konto mit DeepRacer AWS-Administratorberechtigungen für mehrere Benutzer ein. Wenn Sie ein Rennen im Mehrbenutzermodus organisieren und typische DeepRacer AWS-Administratoraufgaben ausführen, müssen Sie Ihr Konto als DeepRacer AWS-Administrator mit entsprechender AWSDeepRacerAccountAdminAccessGenehmigung einrichten.
- AWS Gewähren Sie den Teilnehmern, die Sie sponsern möchten, Zugriff auf die Konsole und die Autorisierungsberechtigungen für Rennfahrer.

Richten Sie Ihr Konto mit DeepRacer AWS-Administratorberechtigungen für mehrere Benutzer ein

Um sich als DeepRacer AWS-Administrator für den Mehrbenutzermodus einzurichten, muss Ihrem Benutzer, AWSDeepRacerAccountAdminAccessIhrer Gruppe oder Rolle die DeepRacer IAM-AWS-Administratorrichtlinie zugewiesen sein. Abhängig von Ihrer Organisation können Sie die Administratorrichtlinie selbst einrichten, indem Sie über die Konsole einen Benutzer oder eine Rolle erstellen und die erforderliche IAM-Richtlinie anhängen, oder Sie können sie von Ihrem IT-Administrator bereitstellen lassen. Informationen zu den erforderlichen Administratorrichtlinien finden Sie unter. AWSDeepRacerAccountAdminAccess Weitere Informationen zu IAM-Richtlinien finden Sie unter Access Management im IAM-Benutzerhandbuch.

Bieten Sie Ihren gesponserten Teilnehmern AWS Konsolenzugriff

Um Rennfahrern, die Sie sponsern, Zugriff auf die DeepRacer AWS-Konsole zu gewähren, empfehlen wir die Verwendung von AWS Standardautorisierungsprotokollen wie AWS IAM Identity Centeroder AWS Identity and Access Management. Sie können den Zugriff auch über das bereits bestehende SSO Ihrer Organisation gewähren. Wenn sich Teilnehmer mit den von Ihnen angegebenen Anmeldeinformationen an der DeepRacer AWS-Konsole anmelden, werden sie aufgefordert, ein AWS Spielerkonto zu erstellen, um sich anzumelden und unter Ihrem AWS Konto auf die DeepRacer AWS-Konsole zuzugreifen. Weitere Informationen zu AWS Spielerkonten finden Sie unter AWS Spielerkonten.

Bieten Sie gesponserten Teilnehmern mithilfe von IAM AWS Konsolenzugriff

1. Erstellen Sie für jeden Teilnehmer einen IAM-Benutzernamen und ein Passwort. Weitere Informationen finden Sie unter Einen IAM-Benutzer in Ihrem AWS-Konto erstellen.

2. Erteilen Sie jedem Teilnehmer die Berechtigungen in AWSDeepRacerDefaultMultiUserAccess. Weitere Informationen finden Sie unter AWS Verwaltete Richtlinien für AWS DeepRacer.
3. Senden Sie den Teilnehmern eine E-Mail mit IAM-Benutzernamen und -Passwörtern sowie einem Link zur Konsole. Über den bereitgestellten Link und die Eingabe ihrer IAM-Benutzernamen und -Passwörter können die Teilnehmer auf die Konsole zugreifen. Informationen zum Erstellen von IAM-Benutzern finden Sie unter Einen IAM-Benutzer in Ihrem AWS-Konto erstellen.

Bieten Sie gesponserten Teilnehmern mithilfe von IAM Identity Center AWS Konsolenzugriff

1. Öffnen Sie die IAM Identity Center-Konsole unter https://console.aws.amazon.com/singlesignon/, erstellen Sie einen benutzerdefinierten Berechtigungssatz und weisen Sie dem Konto Benutzer zu. Weitere Informationen finden Sie unter Berechtigungssätze.
2. Geben Sie beim Erstellen des benutzerdefinierten Berechtigungssatzes die folgenden Werte an:

- Relaisstatus: `https://console.aws.amazon.com/deepracer/home?region=us-east-1#getStarted`

> **Note**
>
> Der Relay-Status leitet Teilnehmer innerhalb des Kontos an eine angegebene URL weiter. In diesem Fall werden sie zur DeepRacer AWS-Konsole weitergeleitet.

- AWS verwaltete Richtlinien: `AWS DeepRacerDefaultMultiUserAccess`

Nachdem Sie die Voraussetzungen erfüllt haben, können Sie den Mehrbenutzermodus aktivieren und die Teilnehmer über Ihr Konto zum Rennen einladen.

Schritt 2: Aktiviere den Mehrbenutzer-Kontomodus

Nachdem Sie Ihr DeepRacer AWS-Administratorkonto eingerichtet und Ihren gesponserten Teilnehmern Konsolenzugriff und Berechtigungen gewährt haben, können Sie den Mehrbenutzermodus in Ihrem DeepRacer AWS-Konto aktivieren.

> **Note**
>
> Standardmäßig gibt es Kontingente für Konten, die Teilnehmer im Mehrbenutzermodus sponsern. Weitere Informationen finden Sie im Abschnitt über Kontokontingente unter Nutzung überwachen.

1. Navigieren Sie im linken Navigationsbereich zur Mehrbenutzerverwaltung und zur Einrichtungsseite.
2. Aktivieren Sie unter Mehrbenutzerkontenmodus aktivieren die Option Mehrbenutzermodus aktivieren.
3. Wählen Sie im Dialogfeld Mehrbenutzermodus aktivieren die Kontrollkästchen aus, um zu bestätigen, dass Ihre gesponserten Teilnehmer Zugriff und Berechtigungen benötigen.
4. Wählen Sie Mehrbenutzermodus aktivieren aus.

   Wenn Sie die Voraussetzungen erfüllen und den Mehrbenutzermodus aktivieren, kann jeder Ihrer gesponserten Teilnehmer Rennen und Zugmodelle erstellen, wobei alle Schulungs- und Lagergebühren dem Konto des Administrators in Rechnung gestellt werden. AWS Standardmäßig hat ein Teilnehmer ein Kontingent von 3 gleichzeitigen Modellen und kann bis zu 10 offene oder future Rennen gleichzeitig verwalten (einschließlich LIVE-, Classic- und Student-Rennen).

Deaktivieren Sie den Mehrbenutzerkontenmodus

Durch die Deaktivierung des Mehrbenutzermodus wird sichergestellt, dass unter Ihrem Administratorkonto keine neuen Profile erstellt werden können und dass die Profile von zuvor gesponserten Teilnehmern nicht mehr auf dem Administratorkonto sichtbar sind. Die Teilnehmer werden nicht mehr aufgefordert, sich mit ihren AWS Spielerkonten anzumelden, und können nicht auf Modelle zugreifen oder diese trainieren, die unter dem Administratorkonto erstellt wurden.

Der Administrator kann die Modelle der gesponserten Teilnehmer herunterladen, speichern und importieren.

1. Navigieren Sie zur Mehrbenutzerverwaltung und zur Einrichtungsseite.
2. Wählen Sie unter Mehrbenutzerkontenmodus deaktivieren die Option Mehrbenutzermodus deaktivieren aus.

3. Aktivieren Sie im Dialogfeld „Mehrbenutzermodus deaktivieren“ das Kontrollkästchen, um zu bestätigen, dass Sie den Mehrbenutzermodus deaktivieren möchten. Wählen Sie Mehrbenutzermodus deaktivieren.

   Der Mehrbenutzermodus ist deaktiviert.

   > **Note**
   >
   > Alle Modelle, die im Rahmen eines AWS Mehrbenutzerkontos als Sponsoring erstellt wurden, bleiben bestehen, und die Kosten für die Modellspeicherung bleiben auf dem AWS Konto bestehen, bis die Modelle gelöscht werden.

Schritt 3: Laden Sie die Teilnehmer ein, gesponsert zu werden

Mithilfe der bereitgestellten E-Mail-Vorlage kannst du Teilnehmer dazu einladen, als gesponserte Teilnehmer zu trainieren und Rennen zu fahren.

Um Teilnehmer einzuladen

1. Navigieren Sie im linken Navigationsbereich zur Mehrbenutzerverwaltung und zur Einrichtungsseite. Wählen Sie im Abschnitt Benutzer einladen unter Mehrbenutzermodus einrichten die Option Einladungsvorlage anzeigen aus.
2. Kopieren Sie die angezeigte E-Mail-Vorlage in Ihre E-Mail-Client-Anwendung und erstellen Sie daraus eine E-Mail, die Sie an die Teilnehmer senden können, die Sie zum Sponsern einladen möchten. Wenn Sie das bestehende SSO Ihres Unternehmens verwenden, können Sie eine SSO-URL angeben, die Ihre Teilnehmer verwenden können. Alternativ können Sie IAM-Anmeldeinformationen angeben, mit denen sich die Teilnehmer an der AWS Konsole anmelden können.

Schritt 4: Legen Sie Nutzungskontingente fest

Nachdem Ihre gesponserten Teilnehmer ihre Einladungs-E-Mail erhalten und ihre Profile unter Ihrem Konto erstellt haben, werden sie in der Liste der gesponserten Nutzer im Bildschirm „Nutzung überwachen“ angezeigt. Auf diesem Bildschirm können Sie dann Nutzungskontingente für die Anzahl der verfügbaren Trainingsstunden und Modelle für gesponserte Teilnehmer festlegen. Durch die Festlegung von Kontingenten können Sie die Kosten pro Teilnehmer in Ihrem Konto kontrollieren und sicherstellen, dass die Teilnehmer ihr Nutzungskontingent nicht überschreiten können. Sie können

die Nutzungsquoten auch nach Bedarf erhöhen oder verringern, um gesponserten Teilnehmern die Stunden zur Verfügung zu stellen, die sie benötigen, um ein DeepRacer AWS-Modell effektiv zu trainieren.

> **Note**
>
> Standardmäßig erhalten gesponserte Teilnehmer im Mehrbenutzermodus 5 Stunden Schulungszeit.

Um Nutzungskontingente für gesponserte Rennfahrer zu bearbeiten

1. Navigieren Sie im linken Navigationsbereich zu Mehrbenutzerverwaltung und zum Bildschirm Nutzung überwachen. Wählen Sie im Bildschirm Nutzung überwachen unter Gesponserte Benutzer die Teilnehmer aus, für die Sie Kontingente festlegen möchten. Wählen Sie Aktionen aus, um die Drop-down-Liste zu öffnen, und wählen Sie Nutzungskontingente festlegen.
2. Geben Sie im Popup-Fenster „Nutzungskontingente festlegen“ die maximale Anzahl an Trainingsstunden und die maximale Modellanzahl für die ausgewählten Teilnehmer ein. Wählen Sie Bestätigen, um Ihre Änderungen beizubehalten, oder Abbrechen, um sie zu verwerfen.

Schritt 5: Nutzung überwachen

Sie können die Nutzung Ihrer gesponserten Teilnehmer überwachen, einschließlich der geschätzten Ausgaben und der Stunden des Schulungsmodells. Sie können auch das Sponsoring von Teilnehmern unterbrechen, Modelle löschen und Zusammenfassungen der Nutzung einsehen. Sie führen alle Aufgaben im Zusammenhang mit der Überwachung der Nutzung in der DeepRacer AWS-Mehrbenutzerverwaltung auf der Seite Nutzung überwachen aus.

Alle Informationen über die Ausgaben für gesponserte Rennfahrer sind nur Schätzungen und sollten nicht für Budgetierungs- oder Kostenrechnungszwecke verwendet werden. Die Schätzungen sind in USD angegeben und beinhalten keine Sonderpreise. Weitere Informationen über die Preise finden Sie unter – Preise.

Kontokontingente für den Mehrbenutzermodus

Standardmäßig hat ein Sponsoring-Konto im Mehrbenutzermodus die folgenden Kontingente, die auf alle gesponserten Profile aufgeteilt werden:

- 100 gleichzeitige Schulungsjobs

- 100 gleichzeitige Bewertungsjobs
- 100 offene oder future Rennen (einschließlich LIVE-, Classic- und Studentenrennen)
- 1000 Autos
- 50 private Bestenlisten

Um diese Kontingente anzupassen, wenden Sie sich an den Kundendienst.

Um eine Schätzung der Ausgaben einzusehen

Auf der Seite Nutzung überwachen können Sie unter Nutzung überwachen eine geschätzte Zusammenfassung der Nutzung durch Ihre Teilnehmer einsehen.

So richten Sie Abrechnungsbenachrichtigungen ein

Sie können Abrechnungsbenachrichtigungen für Ihr Konto einrichten. Abrechnungsbenachrichtigungen helfen Ihnen, über Ausgaben auf dem Laufenden zu bleiben. Weitere Informationen finden Sie unter Abrechnung.

Um das Sponsoring zu unterbrechen

Sie können das Sponsoring eines einzelnen Teilnehmers, mehrerer Teilnehmer oder aller Teilnehmer unterbrechen. Wenn du das Sponsoring pausierst, können deine gesponserten Teilnehmer in deinem Konto keine neuen Modelle oder Trainingsmodelle erstellen. Laufende Schulungen werden bis zum Abschluss abgeschlossen und sind in den Ausgabenschätzungen enthalten. Sie können die Patenschaft jederzeit wieder aufnehmen. Teilnehmer, deren Mehrbenutzerzugriff unterbrochen wurde, können ihre Modelle weiterhin ansehen und Modelle in Bestenlisten posten, aber sie können keine kostenverursachenden Aktivitäten durchführen.

1. Wählen Sie auf der Seite Nutzung überwachen unter Nutzung überwachen im Bereich Gesponserte Nutzer die Nutzer aus, für die Sie das Sponsoring unterbrechen möchten.
2. Wählen Sie Sponsoring pausieren aus.
3. Wählen Sie im Dialogfeld Sponsoring pausieren die Option Sponsoring pausieren aus, um das Sponsoring zu unterbrechen. Wählen Sie Abbrechen, wenn Sie entscheiden, dass Sie das Sponsoring nicht unterbrechen möchten.

Um das Sponsoring wieder aufzunehmen

Sie können das Sponsoring von Rennfahrern wieder aufnehmen, für die Sie das Sponsoring unterbrochen haben.

1. Wählen Sie auf der Seite Nutzung überwachen unter Nutzung überwachen im Bereich Gesponserte Nutzer die Rennfahrer aus, für die Sie das Sponsoring wieder aufnehmen möchten.
2. Wähle Sponsoring fortsetzen aus.

Um sich die Modelle der Rennfahrer anzusehen

- Auf der Seite „Meine Modelle“ können Sie sich unter Modelle Ihre Modelle und die Modelle Ihrer Benutzer ansehen.

Nächste Schritte

Nachdem Sie den Mehrbenutzermodus eingerichtet und aktiviert haben, können Sie die folgenden Schritte ausführen:

- Erstelle ein Community-Rennen.
- Fordern Sie einen DeepRacer AWS-Workshop an.

Richten Sie ein Community-Rennen ein

Community-Rennen bieten Ihren gesponserten Teilnehmern eine spannende Möglichkeit, Reinforcement-Learning zu erleben.

Sie können Community-Rennen erstellen und Ihre gesponserten Teilnehmer einladen.

Weitere Informationen finden Sie unter the section called “Erstelle einen Schnellstart für ein Rennen”.

Fordere einen Workshop an

Sie können einen Workshop anfragen, um mehr über AWS DeepRacer mit einem 60-minütigen Online- oder Präsenzworkshop zu erfahren.

Weitere Informationen finden Sie unter Workshop.

DeepRacer AWS-Mehrbenutzererlebnis (Teilnehmer)

In dieser exemplarischen Vorgehensweise wird die Erfahrung eines einzelnen Teilnehmers veranschaulicht, dessen Profil vom Konto einer Organisation im Mehrbenutzermodus gesponsert wird.

AWS DeepRacer bietet Ihnen eine spannende Möglichkeit, Reinforcement Learning (RL) zu erleben, indem Sie DeepRacer AWS-Modelle trainieren und testen. Ihre Organisation bietet Ihnen möglicherweise die Möglichkeit, Ihr Profil unter ihrem AWS Konto sponsern zu lassen. Alle Gebühren, die Sie generieren, einschließlich Schulung, Evaluierung und Speicherung von Modellen, werden dem AWS Konto in Rechnung gestellt, mit dem Sie sich angemeldet haben. Der Administrator des AWS Accounts, der dein Profil sponsert, kann deine Modelle, Fahrzeuge und Bestenlisten einsehen, deine Trainingszeiten pausieren, deine Trainingszeiten und Speicherkontingente anpassen und dein Profil nicht mehr sponsern.

Im Rahmen der Registrierung deines gesponserten Rennfahrers erstellst du ein AWS Spielerkonto. Das Konto ist ein portables Profil, das Sie behalten und mit einer Reihe anderer AWS Dienste verwenden können. Weitere Informationen finden Sie unter AWS Spielerkonten.

Voraussetzungen

Der Veranstaltungskoordinator Ihrer Organisation teilt Ihnen eine Einladung zur DeepRacer AWS-Mitgliedschaft mit, die auch Anmeldeinformationen für die AWS Konsole enthält. Verwenden Sie diese Anmeldeinformationen, um sich an der Konsole anzumelden. Im Rahmen Ihres Setups erstellen Sie auch ein Rennfahrerprofil und ein AWS Spielerkonto.

Diese exemplarische Vorgehensweise umfasst die folgenden Schritte:

- Melden Sie sich mit den Anmeldeinformationen des Sponsoring-Kontos bei der AWS Konsole an.
- Erstelle ein AWS Spielerkonto oder melde dich bei einem an.
- Passen Sie Ihr Profil an.
- Trainieren von -Modellen.
- Gesponserte Nutzung anzeigen.
- (Optional) Fordern Sie zusätzliche gesponserte Stunden an.

Schritt 1. Melden Sie sich mit den AWS Anmeldeinformationen des Sponsoring-Kontos bei der Konsole an

Um DeepRacer als gesponserter Teilnehmer mit AWS zu beginnen, melden Sie sich mit den Anmeldeinformationen in der Einladung, die Sie vom Veranstaltungskoordinator erhalten haben, bei der Konsole an.

Um sich als gesponserter Teilnehmer bei der AWS Konsole anzumelden

1. Verwenden Sie die Anmeldeinformationen auf der Einladung, die Sie vom Veranstaltungskoordinator erhalten haben.
2. Navigieren Sie in der Konsole zu AWS DeepRacer.

   Die Seite mit dem AWS Spielerkonto wird angezeigt.

Schritt 2. Erstelle ein AWS Spielerkonto oder melde dich an

1. Erstellen Sie auf der Seite AWS Spielerkonto ein vorhandenes Spielerkonto oder melden Sie sich bei einem vorhandenen AWS Spielerkonto an.
   - Wenn Sie noch kein Konto haben, wählen Sie Konto erstellen, geben Sie Ihre E-Mail-Adresse und ein Passwort ein und wählen Sie dann Konto erstellen.
   - Wenn du bereits ein AWS Spielerkonto hast, gib deine E-Mail-Adresse und dein Passwort ein und wähle Anmelden.
2. Eine Nachricht wird an die E-Mail-Adresse gesendet, die Sie angegeben haben, um die Kontoeinrichtung zu bestätigen.
3. Geben Sie im Feld Bestätigungscode den Code ein, den Sie in der E-Mail erhalten haben, und wählen Sie Registrierung bestätigen.

   > **Note**
   >
   > Bleiben Sie auf der aktuellen Seite, bis Sie Ihren Bestätigungscode eingegeben haben.

   Sie sind jetzt als gesponserter Teilnehmer bei der DeepRacer AWS-Konsole angemeldet.
4. Fahren Sie mit Schritt 3 fort, um Ihr Rennfahrerprofil anzupassen.

Schritt 3. Passen Sie Ihr Profil an

Passen Sie Ihr Profil an, indem Sie Ihr Profilbild bearbeiten und einen Rennfahrernamen hinzufügen. Du kannst dein Rennfahrerprofil jederzeit aktualisieren und ändern. Sie können auch Ihr Wohnsitzland und eine Kontakt-E-Mail für den Empfang von Mitteilungen über in der AWS DeepRacer League gewonnene Preise hinzufügen. Wenn Sie für Ihre Leistung in der AWS DeepRacer League Erfolge erzielen, können Sie diese außerdem auf der Profilseite „Ihr Rennfahrer“ in sozialen Netzwerken teilen.

> **Note**
>
> Um an Rennveranstaltungen der AWS DeepRacer League teilzunehmen und Modelle zu trainieren, müssen Sie einen Rennfahrernamen erstellen und Ihr Wohnsitzland hinzufügen. Ihr Rennfahrername muss weltweit einzigartig sein. Sobald du dein Wohnsitzland ausgewählt hast, ist es für die Rennsaison gesperrt.

So passen Sie Ihr Rennfahrer-Profilbild an

1. Navigiere im linken Navigationsbereich zur Profilseite „Dein Rennfahrer“.
2. Wähle auf der Seite Dein Rennfahrer-Profil die Option Bearbeiten aus.
3. Passen Sie im Dialogfeld Ihr Rennfahrer-Profil Ihr Rennfahrer-Profilbild an, indem Sie Elemente aus den Drop-down-Listen auswählen.
4. Wählen Sie Save (Speichern) aus.

So passen Sie den Namen Ihres Rennfahrers an

1. Navigiere im linken Navigationsbereich zur Profilseite „Dein Rennfahrer“.
2. Wähle auf der Seite Dein Rennfahrer-Profil die Option Bearbeiten aus.
3. Wählen Sie im Dialogfeld „Ihr Rennfahrerprofil“ die Option „Namen Ihres Rennfahrers ändern“ und geben Sie einen Namen für Ihr Profil ein.
4. Wählen Sie Save (Speichern) aus.

Schritt 4. Modelle trainieren

Wenn Sie Ihr Profil angepasst haben, können Sie mit dem Training von Modellen beginnen. Weitere Informationen finden Sie unter DeepRacer AWS-Modelle trainieren und auswerten.

Schritt 5. Gesponserte Nutzung anzeigen

Du solltest den Überblick über deine gesponserten Stunden und Modelle behalten, damit du das Beste aus ihnen herausholen kannst.

Um die Nutzung gesponserter Stunden und die gespeicherten Modelle einzusehen

- Auf der Profilseite deines Rennfahrers findest du unter Gesponserte Nutzung die Gesamtzahl der genutzten Stunden und die Anzahl der gespeicherten Modelle.

Schritt 6: (Optional) Fordere zusätzliche gesponserte Stunden an

Als gesponserter Teilnehmer erhalten Sie fünf Stunden kostenlose Trainingszeit. Wenn Ihre kostenlosen gesponserten Stunden aufgebraucht sind, können Sie bei Ihrem Kontoadministrator oder dem Veranstalter zusätzliche Stunden beantragen. Wenn Sie keinen Zugriff auf zusätzliche gesponserte Stunden haben, können Sie Ihre Reise mit AWS alternativ fortsetzen, DeepRacer indem Sie Ihr eigenes DeepRacer AWS-Konto erstellen. Informationen zu Schulungs- und Lagerkosten finden Sie unter Preise.

Tools für Lehrkräfte für AWS-Studierende DeepRacer

In diesem Abschnitt finden Sie Informationen und Ressourcen, um den DeepRacer AWS-Machine-Learning-Lehrplan in den Unterricht zu integrieren, praktische AWS-Übungen für DeepRacer Schüler abzuhalten und Community-Rennen für Schüler zu organisieren.

Integrieren Sie AWS DeepRacer Student in den Unterricht

Wenn Sie ein Pädagoge sind, der gerade erst mit AWS anfängt DeepRacer, empfehlen wir Ihnen, die AWS DeepRacer Student Educator Playbooks zu lesen.

Playbook zum Lehrplan

Das AWS DeepRacer Student Curriculum Playbook beschreibt den Überblick über die einzelnen DeepRacer AWS-Studentenmodule, Lernziele, Lernergebnisse, Schlüsselkonzepte, unterstützendes Material sowie Bewertungs- und Aktivitätsvorschläge.

Playbook für Schülerlabore

Das AWS DeepRacer Student Labs Playbook bietet die Informationen und Ressourcen für Lehrkräfte, um praktische DeepRacer AWS-Student-Labore abzuhalten. Praktische Übungen bestehen aus virtuellen Veranstaltungen wie AWS DeepRacer Student League-Rennen, privaten Community-Rennen, virtuellen Live-Rennen und Präsenzveranstaltungen mit einer physischen Strecke und einem DeepRacer AWS-Gerät.

Erstellen Sie Community-Rennen für Schüler

Nachdem Sie mit den Playbooks für Lehrkräfte begonnen haben, verwenden Sie Community-Rennen in der DeepRacer AWS-Konsole, um Rennen für Schüler in der AWS DeepRacer Student League zu erstellen. Teilen Sie einen Link zur Einladung zu einem Rennen, um Teilnehmer des Studentenrennens einzuladen.

Lehrkräfte benötigen ein AWS Konto, um sich bei der DeepRacer AWS-Konsole anzumelden, um Rennen zu erstellen und zu organisieren. Studierende benötigen jedoch nur eine E-Mail-Adresse, um sich bei der AWS DeepRacer Student League anzumelden, ihr Profil zu aktualisieren, kostenlose Kurse zu belegen und DeepRacer AWS-Modelle zu erstellen. Lehrkräfte können auch eine E-Mail-Adresse verwenden, um ein Konto in der AWS DeepRacer Student League zu erstellen, um eine

Vorschau des Lehrplans zu erhalten, das Rennerlebnis auszuprobieren und die Fortschritte Ihrer Schüler zu überwachen.

Fahren Sie mit einem der folgenden Themen fort, um ein virtuelles DeepRacer AWS-Studenten-Rennen zu erstellen oder zu verwalten.

Themen

- the section called “Erstellen Sie ein Studentenrennen”
- the section called “Passen Sie ein Studentenrennen an”
- the section called “Organisieren Sie ein Studentenrennen”

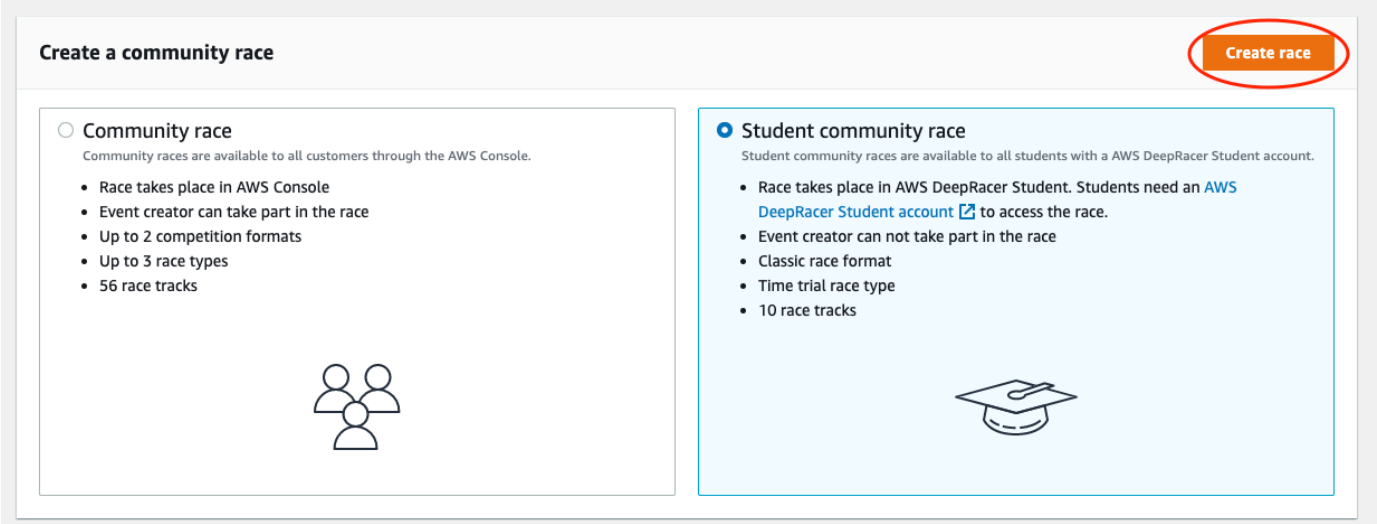
Ein AWS-Community-Rennen für DeepRacer Studierende erstellen

Mit den Standardeinstellungen für das Rennen in der Studentengemeinschaft können Sie schnell ein virtuelles Rennen einrichten.

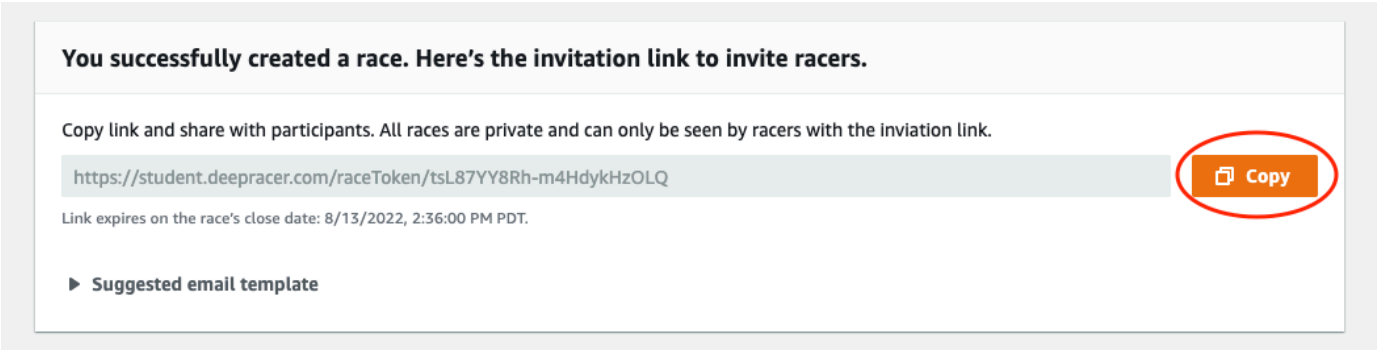
Community-Rennen für Schüler sind asynchrone Ereignisse, für die keine Interaktion in Echtzeit erforderlich ist. Die Teilnehmer müssen einen Einladungslink erhalten, um ein Modell für das Rennen einzureichen und sich die Bestenliste anzusehen. Rennfahrer können innerhalb eines bestimmten Zeitraums jederzeit eine unbegrenzte Anzahl von Modellen einreichen, um in der Bestenliste aufzusteigen. Ergebnisse und Videos für klassische Rennen können für eingereichte Modelle auf der Bestenlisten-Seite eingesehen werden, sobald das Rennen gestartet wird.

Um mit der Erstellung eines Community-Rennens für Studenten zu beginnen

1. Öffnen Sie die DeepRacer AWS-Konsole.
2. Wählen Sie auf der Seite Community Races die Option Student Community Race aus.
3. Wähle Rennen erstellen aus.



4. Geben Sie einen originellen, aussagekräftigen Namen für das Rennen ein.
5. Geben Sie das Startdatum und die Uhrzeit der Veranstaltung im 24-Stunden-Format an. Die DeepRacer AWS-Konsole erkennt Ihre Zeitzone automatisch. Geben Sie auch ein Enddatum und eine Endzeit ein.
6. Um die Standard-Renneinstellungen zu verwenden, wählen Sie Weiter. Wenn du bereit bist, mehr über all deine Optionen zu erfahren, gehe zuthe section called “Passen Sie ein Studentenrennen an”.
7. Überprüfe auf der Seite mit den Renndetails die Rennspezifikationen. Um Änderungen vorzunehmen, wähle Bearbeiten oder Zurück, um zur Seite mit den Renndetails zurückzukehren. Wenn du bereit bist, den Einladungslink zu erhalten, wähle Absenden.
8. Um dein Rennen mit anderen zu teilen, wähle den Link kopieren und füge ihn in die vorgeschlagene E-Mail-Vorlage, Textnachrichten und deine bevorzugten Social-Media-Anwendungen ein. Alle Rennen können nur von Rennfahrern mit einem Einladungslink gesehen werden. Der Link läuft am Schlusstag des Rennens ab.



9. Wenn der Zeitraum für dein Studentenrennen abläuft, notiere dir unter Rennfahrer auf der Seite Rennen verwalten, wer ein Modell eingereicht hat und wer das noch tun muss.

Wähle Rennen verwalten, um die gewählte Rennstrecke zu ändern, eine Rennbeschreibung hinzuzufügen, eine Rangierungsmethode auszuwählen, zu entscheiden, wie viele Rennfahrer zurückgesetzt werden dürfen, die Mindestanzahl an Runden festzulegen, die ein RL-Modell absolvieren muss, um sich für dein Rennen zu qualifizieren, die Off-Track-Strafe festzulegen und weitere Renndetails anzupassen.

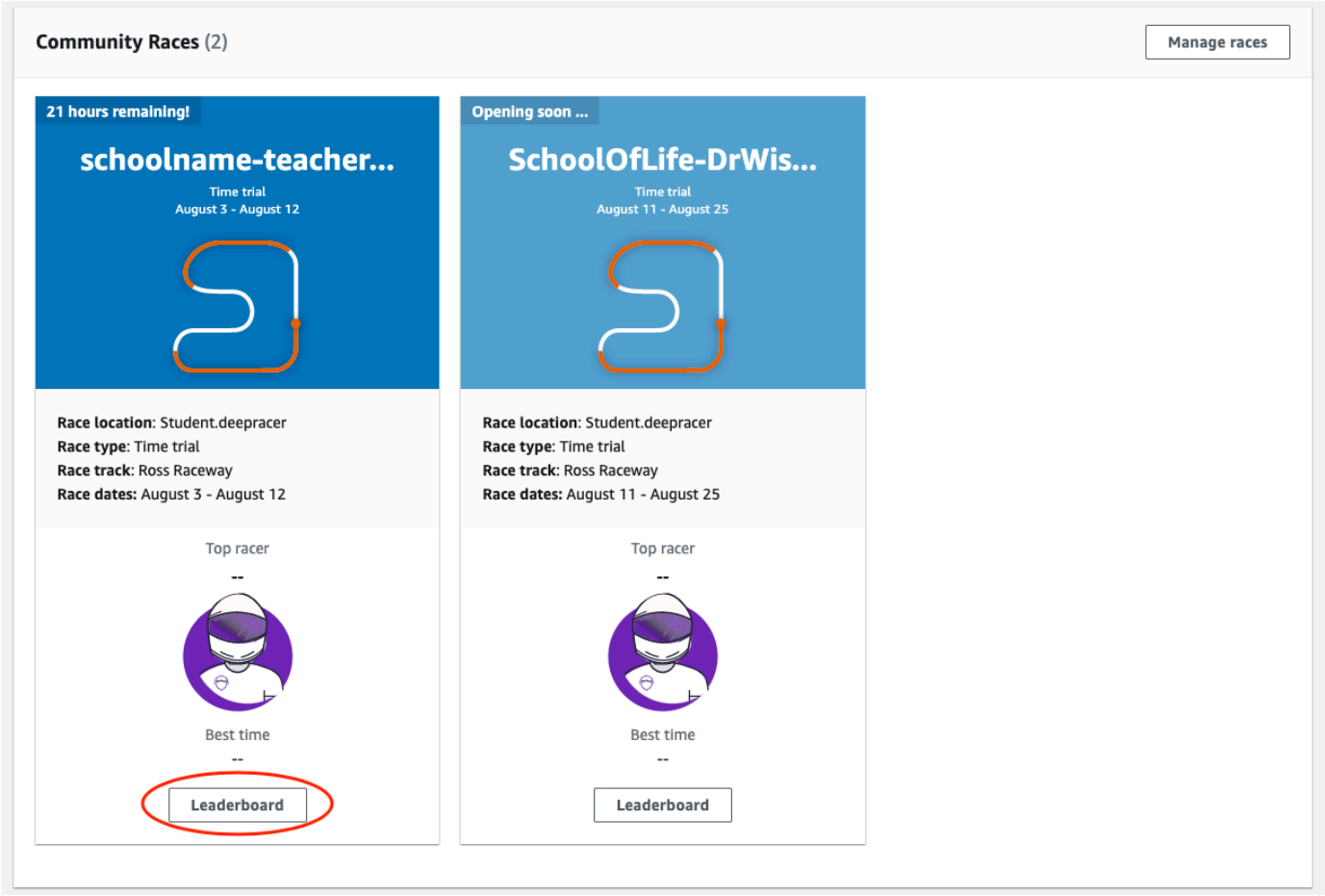
> **Note**
>
> Du siehst nur die Aliasnamen deiner Schüler auf dem Reiter Rennfahrer und in der Bestenliste. Notiere dir also, welcher Alias welchem Schüler zugeordnet ist.

Passen Sie ein AWS-Community-Rennen für DeepRacer Studenten an

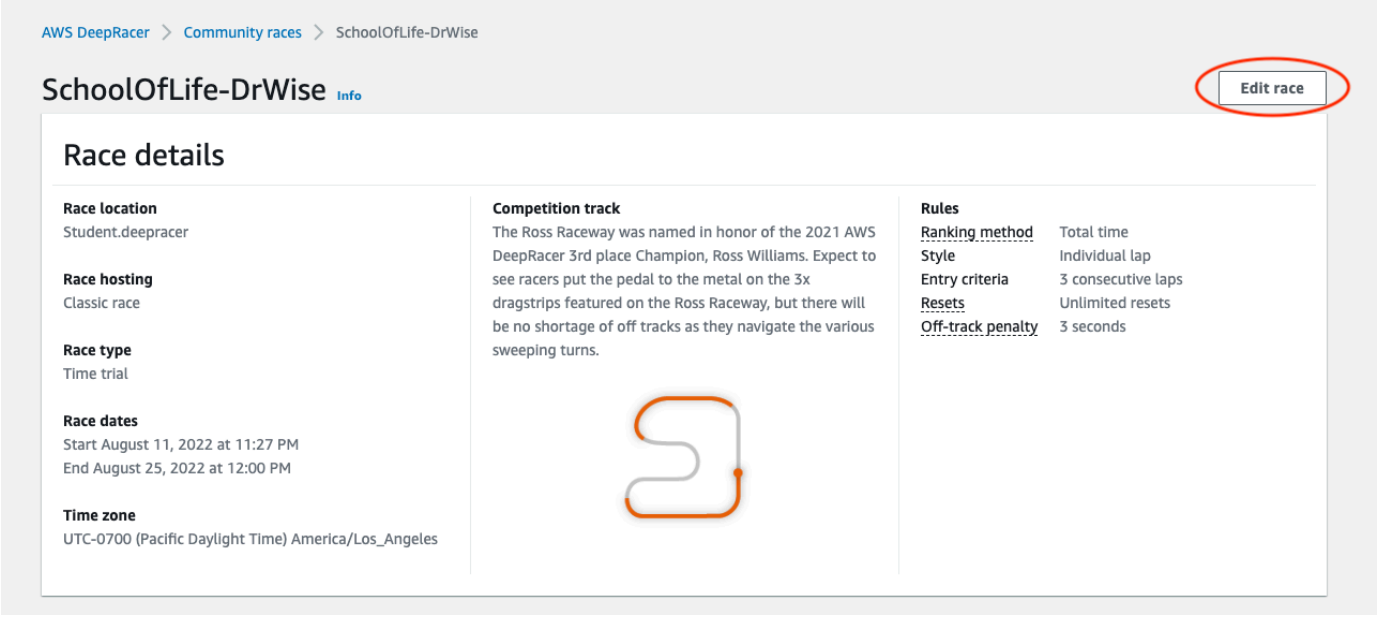
Um ein Rennen zu erstellen, das auf Ihre Gruppe zugeschnitten ist, fügen Sie Anpassungen hinzu, die die Komplexität und Herausforderung eines Rennens erhöhen oder verringern.

Um ein Studentenrennen individuell anzupassen

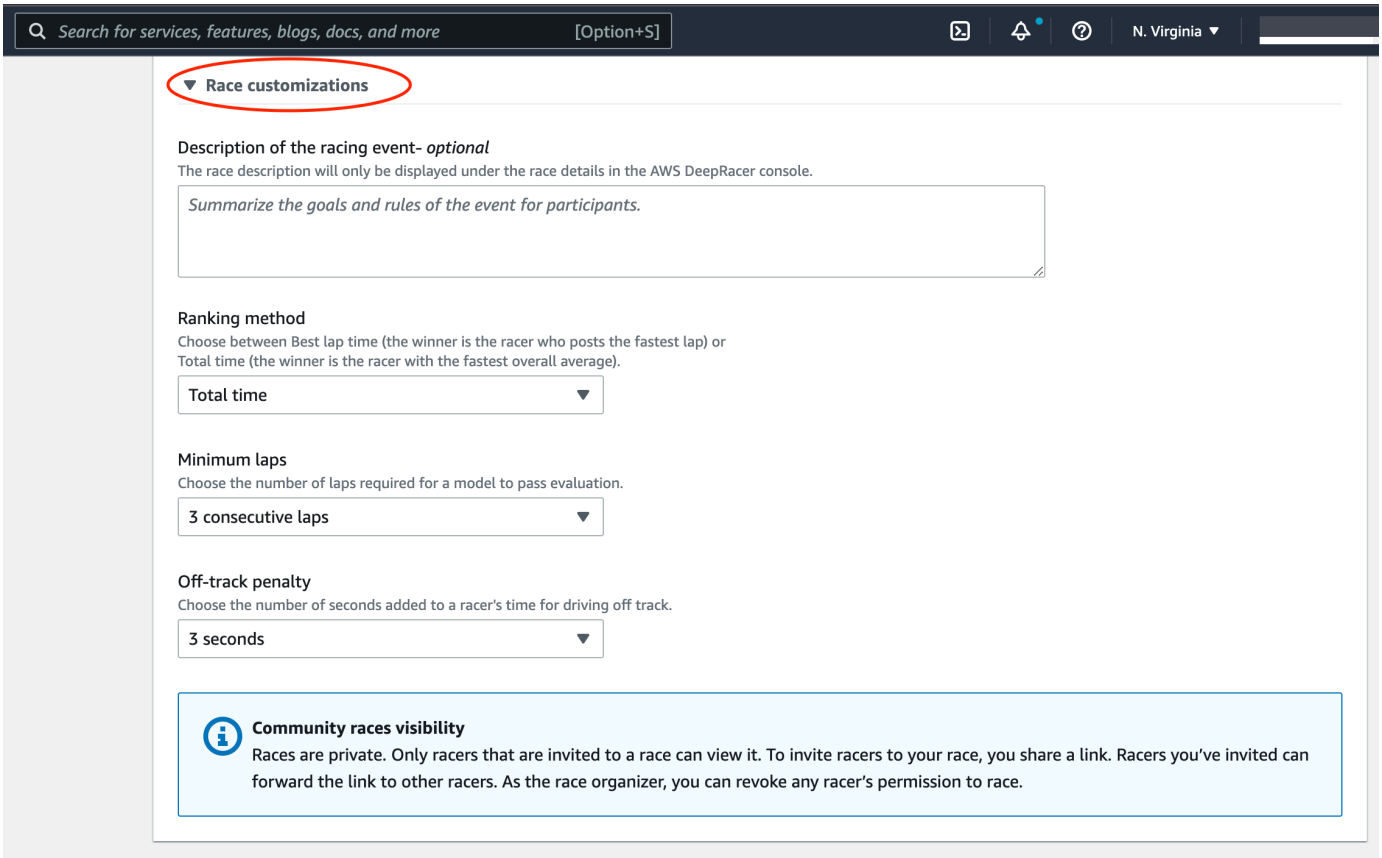
1. Öffnen Sie die DeepRacer AWS-Konsole.
2. Wählen Sie Community-Rennen.
3. Wähle auf der Seite „Community-Rennen“ die Bestenliste für das Rennen aus, das du anpassen möchtest.



4. Wähle auf der Seite mit den Renndetails die Option Rennen bearbeiten aus.



5. Erweitere die Renn-Anpassungen.



6. Schreiben Sie optional eine Beschreibung für Ihr Rennen, in der die Ziele und Regeln der Veranstaltung für die Teilnehmer zusammengefasst sind. Die Beschreibung wird in den Details deiner Bestenliste angezeigt.
7. Wählen Sie für die Ranglistenmethode für ein klassisches Rennen zwischen der besten Rundenzeit, bei der der Gewinner der Fahrer ist, der die schnellste Runde gefahren hat, der Durchschnittszeit, bei der nach mehreren Versuchen innerhalb des Zeitrahmens der Rennfahrer mit der besten Durchschnittszeit der Gewinner ist, oder Gesamtzeit, bei der der Fahrer mit dem schnellsten Gesamtdurchschnitt der Gewinner ist.
8. Wählen Sie einen Wert für „Mindestrunden“. Dies ist die Anzahl der aufeinanderfolgenden Runden, die ein Rennfahrer absolvieren muss, um sich für die Einreichung des Ergebnisses in die Bestenliste des Rennens zu qualifizieren. Wählen Sie für ein Anfängerrennen eine kleinere Zahl. Für fortgeschrittene Benutzer wählen Sie eine größere Anzahl.
9. Wählen Sie für die Strafe außerhalb der Strecke die Anzahl der Sekunden aus, die zur Zeit eines Rennfahrers hinzugefügt werden soll, wenn sein RL-Modell von der Strecke abkommt.
10. Sie haben jetzt alle Anpassungsoptionen für Ihr Community-Rennen für Schüler abgeschlossen. Wähle Weiter, um die Renndetails zu überprüfen.

11. Überprüfe auf der Seite „Renndetails überprüfen“ die Rennspezifikationen. Um Änderungen vorzunehmen, wähle Bearbeiten oder Zurück, um zur Seite mit den Renndetails zurückzukehren. Wenn du bereit bist, den Einladungslink zu erhalten, wähle Absenden.
12. Wählen Sie Erledigt aus. Die Seite Rennen verwalten wird angezeigt.

Unter Community-Rennen verwalten erfährst du, wie du unsere E-Mail-Vorlage verwenden kannst, um neue Rennfahrer einzuladen, Rennfahrer aus deinem Rennen zu entfernen, den Status der eingereichten Modelle von Rennfahrern zu überprüfen und mehr.

Organisieren Sie ein AWS-Community-Rennen für DeepRacer Studenten

Alle Community-Rennen für Studenten sind nur für Personen sichtbar, die einen Einladungslink erhalten haben. Die Teilnehmer können Einladungslinks frei weiterleiten. Um an einem Rennen teilnehmen zu können, benötigen die Teilnehmer jedoch ein DeepRacer AWS-Studentenkonto. Erstbenutzer müssen den Kontoerstellungsprozess abschließen, bevor sie am Rennen teilnehmen können. Die Schüler benötigen lediglich eine E-Mail-Adresse, um ein Konto einzurichten.

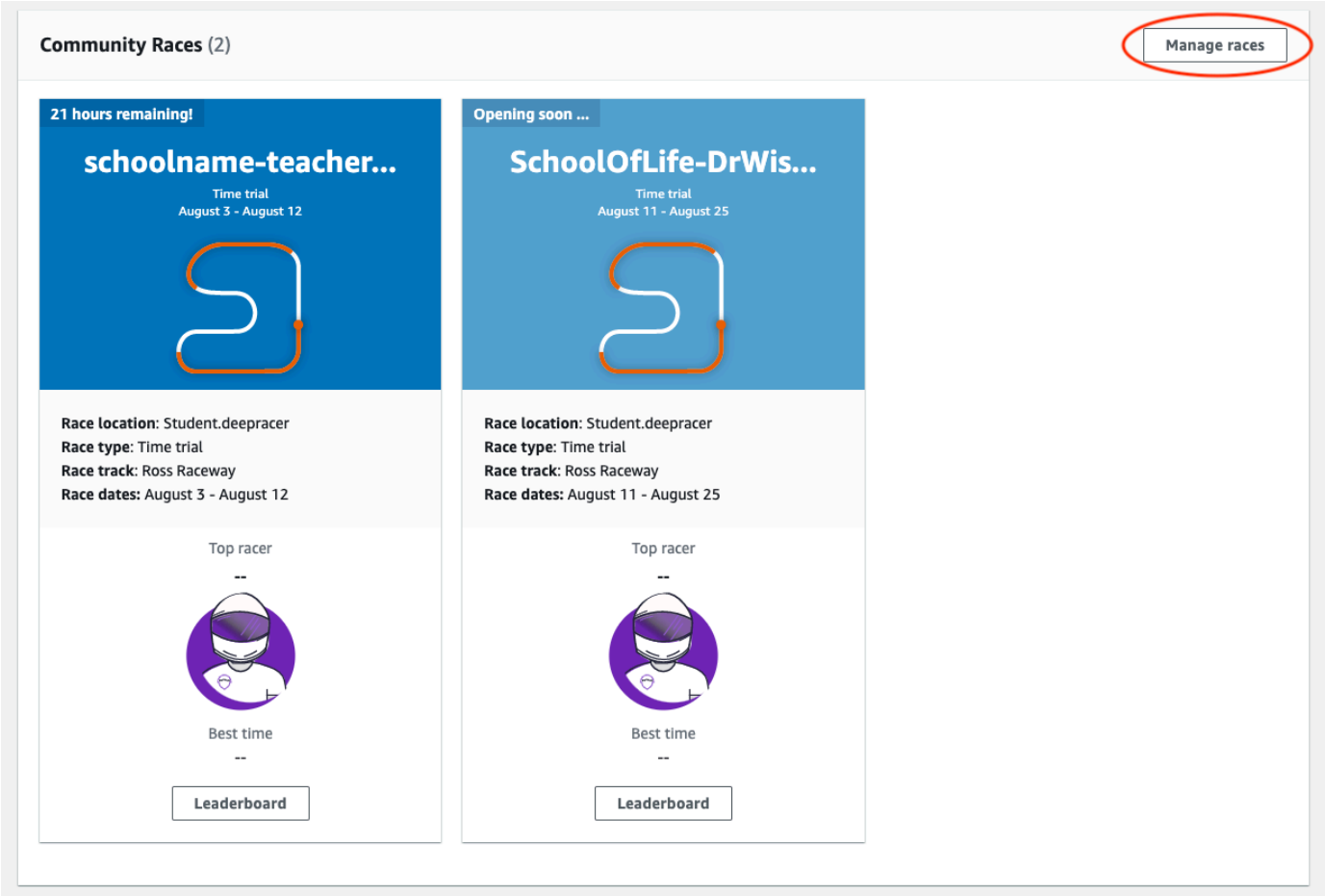
Als Organisator des Rennens können Sie:

- Renndetails bearbeiten (einschließlich Start- und Enddatum)
- Teilnehmer entfernen
- Rennen beenden
- Rennen löschen

> **Note**
>
> Du siehst nur die Aliase deiner Schüler im Reiter Rennfahrer und in der Bestenliste. Notiere dir also, welcher Alias welchem Schüler zugeordnet ist.

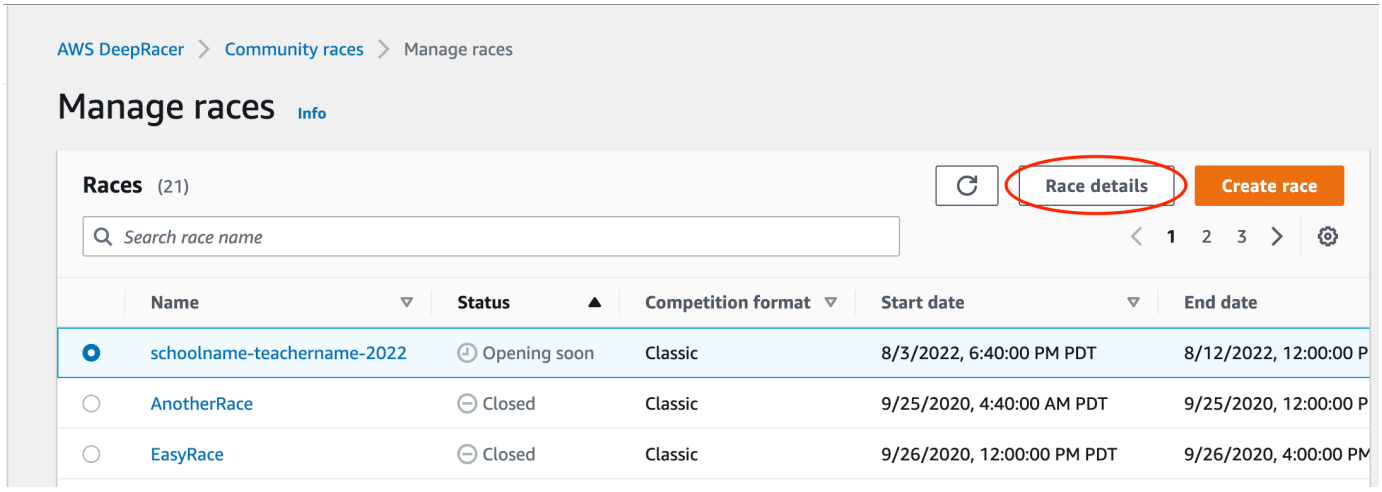
So verwalten Sie ein AWS-Community-Rennen für DeepRacer Studenten

1. Melden Sie sich bei der DeepRacer AWS-Konsole an.
2. Wählen Sie Community-Rennen.
3. Wähle Rennen verwalten aus.



4. Wählen Sie auf der Seite Rennen verwalten das Rennen aus, das Sie verwalten möchten.

5. Wähle Renndetails und dann Bearbeiten.

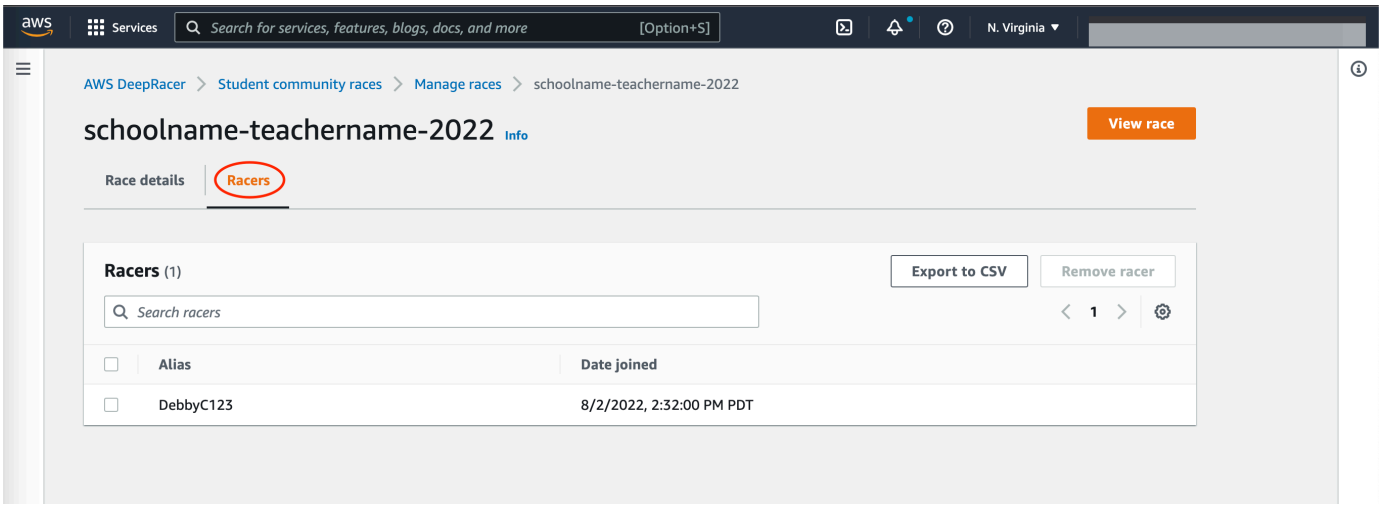


6. Um die Bestenliste des Events einzusehen, wähle Rennen anzeigen.

7. Um den Einladungslink des Events zurückzusetzen, wähle Einladungslink zurücksetzen. Durch das Zurücksetzen des Einladungslinks wird verhindert, dass Spieler, die den ursprünglichen

Link noch nicht ausgewählt haben, aus dem Rennen teilnehmen. Das Zurücksetzen des Einladungslinks hat keine Auswirkungen auf bestehende Teilnehmer des Rennens.

8. Um ein Rennen zu beenden, wähle Rennen beenden. Dadurch wird das Rennen sofort beendet.
9. Um das Event zu löschen, wähle Rennen löschen. Dadurch wird dieses Rennen dauerhaft von der AWS Konsole und von AWS DeepRacer Student entfernt.
10. Um einen Teilnehmer zu entfernen, wählen Sie die Registerkarte Rennfahrer, wählen Sie einen oder mehrere Teilnehmer aus und wählen Sie Rennfahrer entfernen aus. Wenn ein Teilnehmer von einer Veranstaltung entfernt wird, kann er nicht am Rennen teilnehmen.



Sicherheit für AWS DeepRacer

DeepRacer Um AWS für das Training und die Evaluierung von Reinforcement-Learning verwenden zu können, muss Ihr AWS Konto über entsprechende Sicherheitsberechtigungen für den Zugriff auf abhängige AWS Ressourcen verfügen, darunter Amazon VPC für die Ausführung von Trainingsjobs und einen Amazon S3 S3-Bucket zum Speichern trainierter Modellartefakte.

Die DeepRacer AWS-Konsole bietet Ihnen die Möglichkeit, die erforderlichen Sicherheitseinstellungen für die abhängigen Services einzurichten. In diesem Abschnitt werden die AWS Dienste, DeepRacer von denen AWS abhängt, sowie die IAM-Rollen und -Richtlinien dokumentiert, die die erforderlichen Berechtigungen für den Zugriff auf die abhängigen Dienste definieren.

Themen

- Datenschutz in AWS DeepRacer
- DeepRacerAWS-abhängige Services AWS
- Erforderliche IAM-Rollen für AWS DeepRacer zum Aufrufen abhängiger Services AWS
- AWS Identity and Access Management für AWS DeepRacer

Datenschutz in AWS DeepRacer

AWS DeepRacer entspricht dem Modell der AWS gemeinsamen Verantwortung, dem der , das Vorschriften und Richtlinien für den Datenschutz umfasst. AWS ist für den Schutz der globalen Infrastruktur verantwortlich, auf der alle AWS Dienste ausgeführt werden. AWS behält die Kontrolle über die auf dieser Infrastruktur gehosteten Daten, einschließlich der Sicherheitskonfigurationskontrollen für den Umgang mit Kundeninhalten und personenbezogenen Daten. AWS Kunden und APN-Partner, die entweder als Datenverantwortliche oder als Datenverarbeiter agieren, sind für alle personenbezogenen Daten verantwortlich, die sie in die AWS Cloud stellen.

Aus Datenschutzgründen empfehlen wir Ihnen, die AWS Kontoanmeldedaten zu schützen und individuelle Benutzerkonten mit AWS Identity and Access Management (IAM) einzurichten, sodass jeder Benutzer nur die Berechtigungen erhält, die für die Erfüllung seiner Aufgaben erforderlich sind. Außerdem empfehlen wir, die Daten mit folgenden Methoden schützen:

- Verwenden Sie für jedes Konto die Multi-Faktor-Authentifizierung (MFA).

- Verwenden Sie SSL/TLS, um mit Ressourcen zu kommunizieren. AWS
- Richten Sie die API und die Protokollierung von Benutzeraktivitäten mit ein. AWS CloudTrail
- Verwenden Sie AWS Verschlüsselungslösungen zusammen mit allen Standardsicherheitskontrollen innerhalb der AWS Dienste.
- Verwenden Sie erweiterte verwaltete Sicherheitsservices wie Amazon Macie, die dabei helfen, in Amazon S3 gespeicherte persönliche Daten zu erkennen und zu sichern.

Wir empfehlen dringend, in Freitextfeldern wie z. B. im Feld Name keine sensiblen, identifizierenden Informationen wie Kontonummern von Kunden einzugeben. Dies gilt auch, wenn Sie mit AWS DeepRacer oder anderen AWS Services über die Konsole AWS CLI, API oder arbeiten AWS SDKs. Alle Daten, die Sie in AWS DeepRacer oder andere Services eingeben, werden möglicherweise zur Aufnahme in Diagnoseprotokolle aufgenommen. Wenn Sie eine URL für einen externen Server bereitstellen, schließen Sie keine Anmeldeinformationen zur Validierung Ihrer Anforderung an den betreffenden Server in die URL ein.

Weitere Informationen zum Datenschutz enthält der Blog-Beitrag AWS Shared Responsibility Model and GDPR im AWS -Sicherheitsblog.

DeepRacerAWS-abhängige Services AWS

AWS DeepRacer verwendet die folgenden AWS-Services, um die erforderlichen Ressourcen zu verwalten:

Amazon Simple Storage Service

Um trainierte Modellartefakte in einem Amazon S3 S3-Bucket zu speichern.

AWS Lambda

Zum Erstellen und Ausführen der Belohnungsfunktionen.

AWS CloudFormation

Um Schulungsjobs für DeepRacer AWS-Modelle zu erstellen.

SageMaker KI

Um die DeepRacer AWS-Modelle zu trainieren.

Die abhängigen AWS Lambda AWS CloudFormation, und die SageMaker KI wiederum nutzen andere AWS Dienste wie Amazon CloudWatch und Amazon CloudWatch Logs.

Die folgende Tabelle zeigt AWS Services DeepRacer, die von AWS direkt oder indirekt genutzt werden.

AWS Services, die AWS direkt oder indirekt DeepRacer nutzt

AWS Principal des Dienstes	Kommentare
application-autoscaling	• Indirekt von SageMaker KI aufgerufen, um ihre Abläufe automatisch zu skalieren.
cloudformation	• Wird direkt von AWS aufgerufen DeepRacer , um Kontoressourcen zu erstellen.
cloudwatch	• Wird direkt von AWS aufgerufen DeepRacer , um seine Operationen zu protokollieren. • Indirekt von SageMaker KI aufgerufen, um ihre Operationen zu protokollieren.
ec2	• Indirekt von AWS CloudFormation und SageMaker KI zur Schaffung und Durchführung von Ausbildungsjobs aufgerufen.
kinesisvideo	• Wird direkt von AWS aufgerufen DeepRacer , um zwischengespeicherte Trainingsstreams anzuzeigen.
lambda	• Wird direkt von AWS aufgerufen DeepRacer , um die Belohnungsfunktionen zu erstellen und auszuführen.
logs	• Wird direkt von AWS aufgerufen DeepRacer , um seine Operationen zu protokollieren. • Wird indirekt von aufgerufen AWS Lambda , um seine Operationen zu protokollieren.

AWS Principal des Dienstes	Kommentare
s3	• Indirekt von SageMaker KI aufgerufen, um SageMaker KI-spezifische Speichero perationen durchzuführen. • Wird direkt von AWS aufgerufen, DeepRacer um Buckets zu erstellen, aufzulisten und zu löschen, deren Namen mit "" beginnen`deepracer` . Wird auch aufgerufe n, um Objekte aus den Buckets herunterz uladen, Objekte in die Buckets hochzuladen oder Objekte aus den Buckets zu löschen.
`sagemaker`	• Wird direkt von AWS aufgerufen DeepRacer , um Reinforcement-Learning-Modelle zu trainieren.

Um diese Services mithilfe von AWS aufrufen DeepRacer zu können, müssen Sie über die entsprechenden IAM-Rollen verfügen, denen die erforderlichen Richtlinien zugeordnet sind. Sie erfahren mehr über die Details zu diesen Richtlinien und Rollen unter Erforderliche IAM-Rollen für AWS DeepRacer zum Aufrufen abhängiger Services AWS.

Erforderliche IAM-Rollen für AWS DeepRacer zum Aufrufen abhängiger Services AWS

Bevor Sie ein Modell erstellen, verwenden Sie die DeepRacer AWS-Konsole, um Ressourcen für Ihr Konto einzurichten. Dabei erstellt die DeepRacer AWS-Konsole die folgenden IAM-Rollen:

AWSDeepRacerServiceRole

Ermöglicht AWS DeepRacer , die erforderlichen Ressourcen zu erstellen und AWS Services in Ihrem Namen aufzurufen.

AWSDeepRacerSageMakerAccessRole

Ermöglicht Amazon SageMaker AI, die erforderlichen Ressourcen zu erstellen und AWS Dienste in Ihrem Namen anzurufen.

AWSDeepRacerLambdaAccessRole

Ermöglicht AWS Lambda Funktionen, AWS Dienste in Ihrem Namen aufzurufen.

AWSDeepRacerCloudFormationAccessRole

Ermöglicht AWS CloudFormation das Erstellen und Verwalten von AWS Stacks und Ressourcen in Ihrem Namen.

Folgen Sie den Links, um detaillierte Zugriffsberechtigungen in der AWS IAM-Konsole einzusehen.

AWS Identity and Access Management für AWS DeepRacer

AWS Identity and Access Management (IAM) hilft einem Administrator AWS-Service , den Zugriff auf Ressourcen sicher zu AWS kontrollieren. IAM-Administratoren kontrollieren, wer authentifiziert (angemeldet) und autorisiert werden kann (über Berechtigungen verfügt), um Ressourcen zu verwenden. DeepRacer IAM ist ein Programm AWS-Service , das Sie ohne zusätzliche Kosten nutzen können.

Themen

- Zielgruppe
- Authentifizierung mit Identitäten
- Zugriff mithilfe von Richtlinien verwalten
- Wie AWS DeepRacer funktioniert mit IAM
- Beispiele für identitätsbasierte Richtlinien für AWS DeepRacer
- AWS verwaltete Richtlinien für AWS DeepRacer
- Serviceübergreifende Confused-Deputy-Prävention
- Problembehandlung bei AWS DeepRacer Identität und Zugriff

Zielgruppe

Die Art und Weise, wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von der Arbeit ab, in der Sie tätig sind. DeepRacer

Dienstbenutzer — Wenn Sie den DeepRacer Dienst für Ihre Arbeit verwenden, stellt Ihnen Ihr Administrator die erforderlichen Anmeldeinformationen und Berechtigungen zur Verfügung.

Wenn Sie für Ihre Arbeit mehr DeepRacer Funktionen verwenden, benötigen Sie möglicherweise zusätzliche Berechtigungen. Wenn Sie die Funktionsweise der Zugriffskontrolle nachvollziehen, wissen Sie bereits, welche Berechtigungen Sie von Ihrem Administrator anfordern müssen. Unter Problembehandlung bei AWS DeepRacer Identität und Zugriff finden Sie nützliche Informationen für den Fall, dass Sie keinen Zugriff auf eine Feature in DeepRacer haben.

Serviceadministrator — Wenn Sie in Ihrem Unternehmen für die DeepRacer Ressourcen verantwortlich sind, haben Sie wahrscheinlich vollen Zugriff auf DeepRacer. Es ist Ihre Aufgabe, zu bestimmen, auf welche DeepRacer Funktionen und Ressourcen Ihre Servicebenutzer zugreifen sollen. Anschließend müssen Sie Anforderungen an Ihren IAM-Administrator senden, um die Berechtigungen der Servicebenutzer zu ändern. Lesen Sie die Informationen auf dieser Seite, um die Grundkonzepte von IAM nachzuvollziehen. Weitere Informationen darüber, wie Ihr Unternehmen IAM nutzen kann DeepRacer, finden Sie unterWie AWS DeepRacer funktioniert mit IAM.

IAM-Administrator: Wenn Sie als IAM-Administrator fungieren, sollten Sie Einzelheiten dazu kennen, wie Sie Richtlinien zur Verwaltung des Zugriffs auf DeepRacer verfassen können. Beispiele für DeepRacer identitätsbasierte Richtlinien, die Sie in IAM verwenden können, finden Sie unter. Beispiele für identitätsbasierte Richtlinien für AWS DeepRacer

Authentifizierung mit Identitäten

Authentifizierung ist die Art und Weise, wie Sie sich AWS mit Ihren Identitätsdaten anmelden. Sie müssen als IAM-Benutzer authentifiziert (angemeldet AWS) sein oder eine IAM-Rolle annehmen. Root-Benutzer des AWS-Kontos

Sie können sich AWS als föderierte Identität anmelden, indem Sie Anmeldeinformationen verwenden, die über eine Identitätsquelle bereitgestellt wurden. AWS IAM Identity Center (IAM Identity Center) -Benutzer, die Single Sign-On-Authentifizierung Ihres Unternehmens und Ihre Google- oder Facebook-Anmeldeinformationen sind Beispiele für föderierte Identitäten. Wenn Sie sich als Verbundidentität anmelden, hat der Administrator vorher mithilfe von IAM-Rollen einen Identitätsverbund eingerichtet. Wenn Sie über den Verbund darauf zugreifen AWS , übernehmen Sie indirekt eine Rolle.

Je nachdem, welcher Benutzertyp Sie sind, können Sie sich beim AWS Management Console oder beim AWS Zugangsportal anmelden. Weitere Informationen zur Anmeldung finden Sie AWS unter So melden Sie sich bei Ihrem an AWS-Konto im AWS-Anmeldung Benutzerhandbuch.

Wenn Sie AWS programmgesteuert darauf zugreifen, AWS stellt es ein Software Development Kit (SDK) und eine Befehlszeilenschnittstelle (CLI) bereit, mit denen Sie Ihre Anfragen mithilfe Ihrer

Anmeldeinformationen kryptografisch signieren können. Wenn Sie keine AWS Tools verwenden, müssen Sie Anfragen selbst signieren. Weitere Informationen zur Verwendung der empfohlenen Methode für die Selbstsignierung von Anforderungen finden Sie unter AWS Signature Version 4 für API-Anforderungen im IAM-Benutzerhandbuch.

Unabhängig von der verwendeten Authentifizierungsmethode müssen Sie möglicherweise zusätzliche Sicherheitsinformationen bereitstellen. AWS Empfiehlt beispielsweise, die Multi-Faktor-Authentifizierung (MFA) zu verwenden, um die Sicherheit Ihres Kontos zu erhöhen. Weitere Informationen finden Sie unter Multi-Faktor-Authentifizierung im AWS IAM Identity Center - Benutzerhandbuch und AWS Multi-Faktor-Authentifizierung (MFA) in IAM im IAM-Benutzerhandbuch.

AWS-Konto Root-Benutzer

Wenn Sie einen erstellen AWS-Konto, beginnen Sie mit einer Anmeldeidentität, die vollständigen Zugriff auf alle AWS-Services Ressourcen im Konto hat. Diese Identität wird als AWS-Konto Root-Benutzer bezeichnet. Der Zugriff erfolgt, indem Sie sich mit der E-Mail-Adresse und dem Passwort anmelden, mit denen Sie das Konto erstellt haben. Wir raten ausdrücklich davon ab, den Root-Benutzer für Alltagsaufgaben zu verwenden. Schützen Sie Ihre Root-Benutzer-Anmeldeinformationen. Verwenden Sie diese nur, um die Aufgaben auszuführen, die nur der Root-Benutzer ausführen kann. Eine vollständige Liste der Aufgaben, für die Sie sich als Root-Benutzer anmelden müssen, finden Sie unter Aufgaben, die Root-Benutzer-Anmeldeinformationen erfordern im IAM-Benutzerhandbuch.

IAM-Benutzer und -Gruppen

Ein IAM-Benutzer ist eine Identität innerhalb von Ihnen AWS-Konto , die über spezifische Berechtigungen für eine einzelne Person oder Anwendung verfügt. Wenn möglich, empfehlen wir, temporäre Anmeldeinformationen zu verwenden, anstatt IAM-Benutzer zu erstellen, die langfristige Anmeldeinformationen wie Passwörter und Zugriffsschlüssel haben. Bei speziellen Anwendungsfällen, die langfristige Anmeldeinformationen mit IAM-Benutzern erfordern, empfehlen wir jedoch, die Zugriffsschlüssel zu rotieren. Weitere Informationen finden Sie unter Regelmäßiges Rotieren von Zugriffsschlüsseln für Anwendungsfälle, die langfristige Anmeldeinformationen erfordern im IAM-Benutzerhandbuch.

Eine IAM-Gruppe ist eine Identität, die eine Sammlung von IAM-Benutzern angibt. Sie können sich nicht als Gruppe anmelden. Mithilfe von Gruppen können Sie Berechtigungen für mehrere Benutzer gleichzeitig angeben. Gruppen vereinfachen die Verwaltung von Berechtigungen, wenn es zahlreiche Benutzer gibt. Sie könnten beispielsweise eine Gruppe benennen IAMAdminsund dieser Gruppe Berechtigungen zur Verwaltung von IAM-Ressourcen erteilen.

Benutzer unterscheiden sich von Rollen. Ein Benutzer ist einer einzigen Person oder Anwendung eindeutig zugeordnet. Eine Rolle kann von allen Personen angenommen werden, die sie benötigen. Benutzer besitzen dauerhafte Anmeldeinformationen. Rollen stellen temporäre Anmeldeinformationen bereit. Weitere Informationen finden Sie unter Anwendungsfälle für IAM-Benutzer im IAM-Benutzerhandbuch.

IAM-Rollen

Eine IAM-Rolle ist eine Identität innerhalb von Ihnen AWS-Konto , die über bestimmte Berechtigungen verfügt. Sie ist einem IAM-Benutzer vergleichbar, jedoch nicht mit einer bestimmten Person verknüpft. Um vorübergehend eine IAM-Rolle in der zu übernehmen AWS Management Console, können Sie von einer Benutzer- zu einer IAM-Rolle (Konsole) wechseln. Sie können eine Rolle übernehmen, indem Sie eine AWS CLI oder AWS API-Operation aufrufen oder eine benutzerdefinierte URL verwenden. Weitere Informationen zu Methoden für die Verwendung von Rollen finden Sie unter Methoden für die Übernahme einer Rolle im IAM-Benutzerhandbuch.

IAM-Rollen mit temporären Anmeldeinformationen sind in folgenden Situationen hilfreich:

- Verbundbenutzerzugriff – Um einer Verbundidentität Berechtigungen zuzuweisen, erstellen Sie eine Rolle und definieren Berechtigungen für die Rolle. Wird eine Verbundidentität authentifiziert, so wird die Identität der Rolle zugeordnet und erhält die von der Rolle definierten Berechtigungen. Informationen zu Rollen für den Verbund finden Sie unter Erstellen von Rollen für externe Identitätsanbieter (Verbund) im IAM-Benutzerhandbuch. Wenn Sie IAM Identity Center verwenden, konfigurieren Sie einen Berechtigungssatz. Wenn Sie steuern möchten, worauf Ihre Identitäten nach der Authentifizierung zugreifen können, korreliert IAM Identity Center den Berechtigungssatz mit einer Rolle in IAM. Informationen zu Berechtigungssätzen finden Sie unter Berechtigungssätze im AWS IAM Identity Center -Benutzerhandbuch.
- Temporäre IAM-Benutzerberechtigungen – Ein IAM-Benutzer oder eine -Rolle kann eine IAM-Rolle übernehmen, um vorübergehend andere Berechtigungen für eine bestimmte Aufgabe zu erhalten.
- Kontoübergreifender Zugriff – Sie können eine IAM-Rolle verwenden, um einem vertrauenswürdigen Prinzipal in einem anderen Konto den Zugriff auf Ressourcen in Ihrem Konto zu ermöglichen. Rollen stellen die primäre Möglichkeit dar, um kontoübergreifendem Zugriff zu gewähren. Bei einigen können Sie AWS-Services jedoch eine Richtlinie direkt an eine Ressource anhängen (anstatt eine Rolle als Proxy zu verwenden). Informationen zu den Unterschieden zwischen Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

- Serviceübergreifender Zugriff — Einige AWS-Services verwenden Funktionen in anderen AWS-Services. Wenn Sie beispielsweise in einem Service einen Anruf tätigen, ist es üblich, dass dieser Service Anwendungen in Amazon ausführt EC2 oder Objekte in Amazon S3 speichert. Ein Dienst kann dies mit den Berechtigungen des aufrufenden Prinzipals mit einer Servicerolle oder mit einer serviceverknüpften Rolle tun.
  - Forward Access Sessions (FAS) — Wenn Sie einen IAM-Benutzer oder eine IAM-Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FAS verwendet die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, kombiniert mit der Anfrage, Anfragen an AWS-Service nachgelagerte Dienste zu stellen. FAS-Anfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS-Services oder Ressourcen erforderlich sind. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien für FAS-Anfragen finden Sie unter Zugriffssitzungen weiterleiten.
  - Servicerolle – Eine Servicerolle ist eine IAM-Rolle, die ein Service übernimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM-Administrator kann eine Servicerolle innerhalb von IAM erstellen, ändern und löschen. Weitere Informationen finden Sie unter Erstellen einer Rolle zum Delegieren von Berechtigungen an einen AWS-Service im IAM-Benutzerhandbuch.
  - Dienstbezogene Rolle — Eine dienstbezogene Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS-Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Servicebezogene Rollen erscheinen in Ihrem Dienst AWS-Konto und gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.
- Auf Amazon ausgeführte Anwendungen EC2 — Sie können eine IAM-Rolle verwenden, um temporäre Anmeldeinformationen für Anwendungen zu verwalten, die auf einer EC2 Instance ausgeführt werden und AWS API-Anfragen stellen AWS CLI . Dies ist dem Speichern von Zugriffsschlüsseln innerhalb der EC2 Instance vorzuziehen. Um einer EC2 Instanz eine AWS Rolle zuzuweisen und sie allen ihren Anwendungen zur Verfügung zu stellen, erstellen Sie ein Instanzprofil, das an die Instanz angehängt ist. Ein Instanzprofil enthält die Rolle und ermöglicht Programmen, die auf der EC2 Instanz ausgeführt werden, temporäre Anmeldeinformationen abzurufen. Weitere Informationen finden Sie im IAM-Benutzerhandbuch unter Verwenden einer IAM-Rolle, um Berechtigungen für Anwendungen zu gewähren, die auf EC2 Amazon-Instances ausgeführt werden.

Zugriff mithilfe von Richtlinien verwalten

Sie kontrollieren den Zugriff, AWS indem Sie Richtlinien erstellen und diese an AWS Identitäten oder Ressourcen anhängen. Eine Richtlinie ist ein Objekt, AWS das, wenn es einer Identität oder Ressource zugeordnet ist, deren Berechtigungen definiert. AWS wertet diese Richtlinien aus, wenn ein Prinzipal (Benutzer, Root-Benutzer oder Rollensitzung) eine Anfrage stellt. Die Berechtigungen in den Richtlinien legen fest, ob eine Anforderung zugelassen oder abgelehnt wird. Die meisten Richtlinien werden AWS als JSON-Dokumente gespeichert. Weitere Informationen zu Struktur und Inhalten von JSON-Richtliniendokumenten finden Sie unter Übersicht über JSON-Richtlinien im IAM-Benutzerhandbuch.

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Standardmäßig haben Benutzer, Gruppen und Rollen keine Berechtigungen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen. Der Administrator kann dann die IAM-Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen annehmen.

IAM-Richtlinien definieren Berechtigungen für eine Aktion unabhängig von der Methode, die Sie zur Ausführung der Aktion verwenden. Angenommen, es gibt eine Richtlinie, die Berechtigungen für die `iam:GetRole`-Aktion erteilt. Ein Benutzer mit dieser Richtlinie kann Rolleninformationen von der AWS Management Console AWS CLI, der oder der AWS API abrufen.

Identitätsbasierte Richtlinien

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität anfügen können, wie z. B. IAM-Benutzern, -Benutzergruppen oder -Rollen. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien im IAM-Benutzerhandbuch.

Identitätsbasierte Richtlinien können weiter als Inline-Richtlinien oder verwaltete Richtlinien kategorisiert werden. Inline-Richtlinien sind direkt in einen einzelnen Benutzer, eine einzelne Gruppe oder eine einzelne Rolle eingebettet. Verwaltete Richtlinien sind eigenständige Richtlinien, die Sie mehreren Benutzern, Gruppen und Rollen in Ihrem System zuordnen können AWS-Konto.

Zu den verwalteten Richtlinien gehören AWS verwaltete Richtlinien und vom Kunden verwaltete Richtlinien. Informationen dazu, wie Sie zwischen einer verwalteten Richtlinie und einer Inline-Richtlinie wählen, finden Sie unter Auswählen zwischen verwalteten und eingebundenen Richtlinien im IAM-Benutzerhandbuch.

Ressourcenbasierte Richtlinien

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele für ressourcenbasierte Richtlinien sind IAM-Rollen-Vertrauensrichtlinien und Amazon-S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie einen Prinzipal angeben. Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS-Services

Ressourcenbasierte Richtlinien sind Richtlinien innerhalb dieses Diensts. Sie können AWS verwaltete Richtlinien von IAM nicht in einer ressourcenbasierten Richtlinie verwenden.

Zugriffskontrolllisten () ACLs

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLs ähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

Amazon S3 und Amazon VPC sind Beispiele für Dienste, die Unterstützung ACLs bieten. AWS WAF Weitere Informationen finden Sie unter Übersicht über ACLs die Zugriffskontrollliste (ACL) im Amazon Simple Storage Service Developer Guide.

Weitere Richtlinientypen

AWS unterstützt zusätzliche, weniger verbreitete Richtlinientypen. Diese Richtlinientypen können die maximalen Berechtigungen festlegen, die Ihnen von den häufiger verwendeten Richtlinientypen erteilt werden können.

- Berechtigungsgrenzen – Eine Berechtigungsgrenze ist ein erweitertes Feature, mit der Sie die maximalen Berechtigungen festlegen können, die eine identitätsbasierte Richtlinie einer IAM-Entität (IAM-Benutzer oder -Rolle) erteilen kann. Sie können eine Berechtigungsgrenze für eine Entität festlegen. Die daraus resultierenden Berechtigungen sind der Schnittpunkt der

identitätsbasierten Richtlinien einer Entität und ihrer Berechtigungsgrenzen. Ressourcenbasierte Richtlinien, die den Benutzer oder die Rolle im Feld `Principal` angeben, werden nicht durch Berechtigungsgrenzen eingeschränkt. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen über Berechtigungsgrenzen finden Sie unter Berechtigungsgrenzen für IAM-Entitäten im IAM-Benutzerhandbuch.

- Dienststeuerungsrichtlinien (SCPs) — SCPs sind JSON-Richtlinien, die die maximalen Berechtigungen für eine Organisation oder Organisationseinheit (OU) in festlegen. AWS Organizations AWS Organizations ist ein Dienst zur Gruppierung und zentralen Verwaltung mehrerer Objekte AWS-Konten , die Ihrem Unternehmen gehören. Wenn Sie alle Funktionen in einer Organisation aktivieren, können Sie Richtlinien zur Servicesteuerung (SCPs) auf einige oder alle Ihre Konten anwenden. Das SCP schränkt die Berechtigungen für Entitäten in Mitgliedskonten ein, einschließlich der einzelnen Root-Benutzer des AWS-Kontos Entitäten. Weitere Informationen zu Organizations und SCPs finden Sie unter Richtlinien zur Servicesteuerung im AWS Organizations Benutzerhandbuch.
- Ressourcenkontrollrichtlinien (RCPs) — RCPs sind JSON-Richtlinien, mit denen Sie die maximal verfügbaren Berechtigungen für Ressourcen in Ihren Konten festlegen können, ohne die IAM-Richtlinien aktualisieren zu müssen, die jeder Ressource zugeordnet sind, deren Eigentümer Sie sind. Das RCP schränkt die Berechtigungen für Ressourcen in Mitgliedskonten ein und kann sich auf die effektiven Berechtigungen für Identitäten auswirken, einschließlich der Root-Benutzer des AWS-Kontos, unabhängig davon, ob sie zu Ihrer Organisation gehören. Weitere Informationen zu Organizations RCPs, einschließlich einer Liste AWS-Services dieser Support-Leistungen RCPs, finden Sie unter Resource Control Policies (RCPs) im AWS Organizations Benutzerhandbuch.
- Sitzungsrichtlinien – Sitzungsrichtlinien sind erweiterte Richtlinien, die Sie als Parameter übergeben, wenn Sie eine temporäre Sitzung für eine Rolle oder einen verbundenen Benutzer programmgesteuert erstellen. Die resultierenden Sitzungsberechtigungen sind eine Schnittmenge der auf der Identität des Benutzers oder der Rolle basierenden Richtlinien und der Sitzungsrichtlinien. Berechtigungen können auch aus einer ressourcenbasierten Richtlinie stammen. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen finden Sie unter Sitzungsrichtlinien im IAM-Benutzerhandbuch.

Mehrere Richtlinientypen

Wenn mehrere auf eine Anforderung mehrere Richtlinientypen angewendet werden können, sind die entsprechenden Berechtigungen komplizierter. Informationen darüber, wie AWS bestimmt wird, ob eine Anfrage zulässig ist, wenn mehrere Richtlinientypen betroffen sind, finden Sie im IAM-Benutzerhandbuch unter Bewertungslogik für Richtlinien.

Wie AWS DeepRacer funktioniert mit IAM

Bevor Sie IAM zur Verwaltung des Zugriffs auf verwenden, sollten Sie sich darüber informieren DeepRacer, mit welchen IAM-Funktionen Sie arbeiten können. DeepRacer

IAM-Funktionen, die Sie mit verwenden können AWS DeepRacer

IAM-Feature	DeepRacer Unterstützung
Identitätsbasierte Richtlinien	Ja
Ressourcenbasierte Richtlinien	Nein
Richtlinienaktionen	Ja
Richtlinienressourcen	Ja
Bedingungsschlüssel für die Richtlinie	Ja
ACLs	Nein
ABAC (Tags in Richtlinien)	Ja
Temporäre Anmeldeinformationen	Ja
Prinzipalberechtigungen	Ja
Servicerollen	Ja
Service-verknüpfte Rollen	Nein

Einen allgemeinen Überblick darüber, wie DeepRacer und andere AWS Dienste mit den meisten IAM-Funktionen funktionieren, finden Sie im IAM-Benutzerhandbuch unter AWS Dienste, die mit IAM funktionieren.

Identitätsbasierte Richtlinien für DeepRacer

Unterstützt Richtlinien auf Identitätsbasis: Ja

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität anfügen können, wie z. B. IAM-Benutzern, -Benutzergruppen oder -Rollen. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien im IAM-Benutzerhandbuch.

Mit identitätsbasierten IAM-Richtlinien können Sie angeben, welche Aktionen und Ressourcen zugelassen oder abgelehnt werden. Darüber hinaus können Sie die Bedingungen festlegen, unter denen Aktionen zugelassen oder abgelehnt werden. Sie können den Prinzipal nicht in einer identitätsbasierten Richtlinie angeben, da er für den Benutzer oder die Rolle gilt, dem er zugeordnet ist. Informationen zu sämtlichen Elementen, die Sie in einer JSON-Richtlinie verwenden, finden Sie in der IAM-Referenz für JSON-Richtlinienelemente im IAM-Benutzerhandbuch.

Beispiele für identitätsbasierte Richtlinien für DeepRacer

Beispiele für DeepRacer identitätsbasierte Richtlinien finden Sie unter. Beispiele für identitätsbasierte Richtlinien für AWS DeepRacer

Ressourcenbasierte Richtlinien in DeepRacer

Unterstützt ressourcenbasierte Richtlinien: Nein

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele für ressourcenbasierte Richtlinien sind IAM-Rollen-Vertrauensrichtlinien und Amazon-S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie einen Prinzipal angeben. Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS-Services

Um kontoübergreifenden Zugriff zu ermöglichen, können Sie ein gesamtes Konto oder IAM-Entitäten in einem anderen Konto als Prinzipal in einer ressourcenbasierten Richtlinie angeben. Durch das Hinzufügen eines kontoübergreifenden Auftraggebers zu einer ressourcenbasierten Richtlinie ist nur die halbe Vertrauensbeziehung eingerichtet. Wenn sich der Prinzipal und die Ressource unterscheiden AWS-Konten, muss ein IAM-Administrator des vertrauenswürdigen Kontos auch der Prinzipalentität (Benutzer oder Rolle) die Berechtigung zum Zugriff auf die Ressource erteilen. Sie erteilen Berechtigungen, indem Sie der juristischen Stelle eine identitätsbasierte Richtlinie anfügen.

Wenn jedoch eine ressourcenbasierte Richtlinie Zugriff auf einen Prinzipal in demselben Konto gewährt, ist keine zusätzliche identitätsbasierte Richtlinie erforderlich. Weitere Informationen finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

Richtlinienaktionen für DeepRacer

Unterstützt Richtlinienaktionen: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das Element `Action` einer JSON-Richtlinie beschreibt die Aktionen, mit denen Sie den Zugriff in einer Richtlinie zulassen oder verweigern können. Richtlinienaktionen haben normalerweise denselben Namen wie der zugehörige AWS API-Vorgang. Es gibt einige Ausnahmen, z. B. Aktionen, die nur mit Genehmigung durchgeführt werden können und für die es keinen passenden API-Vorgang gibt. Es gibt auch einige Operationen, die mehrere Aktionen in einer Richtlinie erfordern. Diese zusätzlichen Aktionen werden als abhängige Aktionen bezeichnet.

Schließen Sie Aktionen in eine Richtlinie ein, um Berechtigungen zur Durchführung der zugeordneten Operation zu erteilen.

Eine Liste der DeepRacer Aktionen finden Sie unter Aktionen definiert von AWS DeepRacer in der Serviceautorisierungsreferenz.

Bei Richtlinienaktionen wird vor der Aktion das folgende Präfix DeepRacer verwendet:

```
deepracer
```

Um mehrere Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie mit Kommata:

```
"Action": [
      "deepracer:action1",
      "deepracer:action2"
         ]
```

Beispiele für DeepRacer identitätsbasierte Richtlinien finden Sie unter. Beispiele für identitätsbasierte Richtlinien für AWS DeepRacer

Politische Ressourcen für DeepRacer

Unterstützt Richtlinienressourcen: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das JSON-Richtlinienelement `Resource` gibt die Objekte an, auf welche die Aktion angewendet wird. Anweisungen müssen entweder ein – `Resource`oder ein `NotResource`-Element enthalten. Als bewährte Methode geben Sie eine Ressource mit dem zugehörigen Amazon-Ressourcennamen (ARN) an. Sie können dies für Aktionen tun, die einen bestimmten Ressourcentyp unterstützen, der als Berechtigungen auf Ressourcenebene bezeichnet wird.

Verwenden Sie für Aktionen, die keine Berechtigungen auf Ressourcenebene unterstützen, z. B. Auflistungsoperationen, einen Platzhalter (*), um anzugeben, dass die Anweisung für alle Ressourcen gilt.

```
"Resource": "*"
```

Eine Liste der DeepRacer Ressourcentypen und ihrer ARNs Eigenschaften finden Sie unter Resources defined by AWS DeepRacer in der Service Authorization Reference. Informationen zu den Aktionen, mit denen Sie den ARN einzelner Ressourcen angeben können, finden Sie unter Von AWS definierte AktionenDeepRacer.

Beispiele für DeepRacer identitätsbasierte Richtlinien finden Sie unter. Beispiele für identitätsbasierte Richtlinien für AWS DeepRacer

Richtlinien-Bedingungsschlüssel für DeepRacer

Unterstützt servicespezifische Richtlinienbedingungsschlüssel: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer auf was Zugriff hat. Das heißt, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Das Element `Condition` (oder `Condition` block) ermöglicht Ihnen die Angabe der Bedingungen, unter denen eine Anweisung wirksam ist. Das Element `Condition` ist optional. Sie können bedingte Ausdrücke erstellen, die Bedingungsoperatoren verwenden, z. B. ist gleich oder kleiner als, damit die Bedingung in der Richtlinie mit Werten in der Anforderung übereinstimmt.

Wenn Sie mehrere `Condition`-Elemente in einer Anweisung oder mehrere Schlüssel in einem einzelnen `Condition`-Element angeben, wertet AWS diese mittels einer logischen `AND`-Operation aus. Wenn Sie mehrere Werte für einen einzelnen Bedingungsschlüssel angeben, AWS wertet die Bedingung mithilfe einer logischen `OR` Operation aus. Alle Bedingungen müssen erfüllt werden, bevor die Berechtigungen der Anweisung gewährt werden.

Sie können auch Platzhaltervariablen verwenden, wenn Sie Bedingungen angeben. Beispielsweise können Sie einem IAM-Benutzer die Berechtigung für den Zugriff auf eine Ressource nur dann gewähren, wenn sie mit dessen IAM-Benutzernamen gekennzeichnet ist. Weitere Informationen finden Sie unter IAM-Richtlinienelemente: Variablen und Tags im IAM-Benutzerhandbuch.

AWS unterstützt globale Bedingungsschlüssel und dienstspezifische Bedingungsschlüssel. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter Kontextschlüssel für AWS globale Bedingungen im IAM-Benutzerhandbuch.

Eine Liste der DeepRacer Bedingungsschlüssel finden Sie unter Bedingungsschlüssel für AWS DeepRacer im IAM-Benutzerhandbuch in der Service Authorization Reference. Informationen zu den Aktionen und Ressourcen, mit denen Sie einen Bedingungsschlüssel verwenden können, finden Sie unter Aktionen definiert von AWS DeepRacer.

Beispiele für DeepRacer identitätsbasierte Richtlinien finden Sie unter. Beispiele für identitätsbasierte Richtlinien für AWS DeepRacer

Zugriffskontrolllisten () ACLs in DeepRacer

Unterstützt ACLs: Nein

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLs ähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

Attributbasierte Zugriffskontrolle (ABAC) mit DeepRacer

Unterstützt ABAC (Tags in Richtlinien): Ja

Die attributbasierte Zugriffskontrolle (ABAC) ist eine Autorisierungsstrategie, bei der Berechtigungen basierend auf Attributen definiert werden. In werden AWS diese Attribute als Tags bezeichnet. Sie können Tags an IAM-Entitäten (Benutzer oder Rollen) und an viele AWS Ressourcen anhängen. Das Markieren von Entitäten und Ressourcen ist der erste Schritt von ABAC. Anschließend entwerfen Sie ABAC-Richtlinien, um Operationen zuzulassen, wenn das Tag des Prinzipals mit dem Tag der Ressource übereinstimmt, auf die sie zugreifen möchten.

ABAC ist in Umgebungen hilfreich, die schnell wachsen, und unterstützt Sie in Situationen, in denen die Richtlinienverwaltung mühsam wird.

Um den Zugriff auf der Grundlage von Tags zu steuern, geben Sie im Bedingungselement einer Richtlinie Tag-Informationen an, indem Sie die Schlüssel `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`, oder Bedingung `aws:TagKeys` verwenden.

Wenn ein Service alle drei Bedingungsschlüssel für jeden Ressourcentyp unterstützt, lautet der Wert für den Service Ja. Wenn ein Service alle drei Bedingungsschlüssel für nur einige Ressourcentypen unterstützt, lautet der Wert Teilweise.

Weitere Informationen zu ABAC finden Sie unter Definieren von Berechtigungen mit ABAC-Autorisierung im IAM-Benutzerhandbuch. Um ein Tutorial mit Schritten zur Einstellung von ABAC anzuzeigen, siehe Attributbasierte Zugriffskontrolle (ABAC) verwenden im IAM-Benutzerhandbuch.

Temporäre Anmeldeinformationen verwenden mit DeepRacer

Unterstützt temporäre Anmeldeinformationen: Ja

Einige funktionieren AWS-Services nicht, wenn Sie sich mit temporären Anmeldeinformationen anmelden. Weitere Informationen, einschließlich Informationen, die mit temporären Anmeldeinformationen AWS-Services funktionieren AWS-Services , finden Sie im IAM-Benutzerhandbuch unter Diese Option funktioniert mit IAM.

Sie verwenden temporäre Anmeldeinformationen, wenn Sie sich mit einer anderen AWS Management Console Methode als einem Benutzernamen und einem Passwort anmelden. Wenn Sie beispielsweise AWS über den Single Sign-On-Link (SSO) Ihres Unternehmens darauf zugreifen, werden bei diesem Vorgang automatisch temporäre Anmeldeinformationen erstellt. Sie erstellen auch automatisch temporäre Anmeldeinformationen, wenn Sie sich als Benutzer bei der Konsole anmelden und dann die Rollen wechseln. Weitere Informationen zum Wechseln von Rollen finden Sie unter Wechseln von einer Benutzerrolle zu einer IAM-Rolle (Konsole) im IAM-Benutzerhandbuch.

Mithilfe der AWS API AWS CLI oder können Sie temporäre Anmeldeinformationen manuell erstellen. Sie können diese temporären Anmeldeinformationen dann für den Zugriff verwenden AWS. AWS empfiehlt, temporäre Anmeldeinformationen dynamisch zu generieren, anstatt langfristige Zugriffsschlüssel zu verwenden. Weitere Informationen finden Sie unter Temporäre Sicherheitsanmeldeinformationen in IAM.

Serviceübergreifende Prinzipal-Berechtigungen für DeepRacer

Unterstützt Forward Access Sessions (FAS): Ja

Wenn Sie einen IAM-Benutzer oder eine IAM-Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FAS verwendet die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, kombiniert mit der Anforderung, Anfragen an nachgelagerte Dienste AWS-Service zu stellen. FAS-Anfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS-Services oder Ressourcen erforderlich sind. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien für FAS-Anfragen finden Sie unter Zugriffssitzungen weiterleiten.

Servicerollen für DeepRacer

Unterstützt Servicerollen: Ja

Eine Servicerolle ist eine IAM-Rolle, die ein Service annimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM-Administrator kann eine Servicerolle innerhalb von IAM erstellen, ändern und löschen. Weitere Informationen finden Sie unter Erstellen einer Rolle zum Delegieren von Berechtigungen an einen AWS-Service im IAM-Benutzerhandbuch.

> **Warning**
>
> Durch das Ändern der Berechtigungen für eine Servicerolle kann die DeepRacer Funktionalität beeinträchtigt werden. Bearbeiten Sie Servicerollen nur, wenn DeepRacer eine Anleitung dazu gibt.

Mit Diensten verknüpfte Rollen für DeepRacer

Unterstützt serviceverknüpfte Rollen: Ja

Eine dienstbezogene Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS-Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Dienstbezogene Rollen werden in Ihrem Dienst angezeigt AWS-Konto und gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.

Details zum Erstellen oder Verwalten von serviceverknüpften Rollen finden Sie unter AWS -Services, die mit IAM funktionieren. Suchen Sie in der Tabelle nach einem Service mit einem `Yes` in der Spalte Service-linked role (Serviceverknüpfte Rolle). Wählen Sie den Link Yes (Ja) aus, um die Dokumentation für die serviceverknüpfte Rolle für diesen Service anzuzeigen.

Beispiele für identitätsbasierte Richtlinien für AWS DeepRacer

Benutzer und Rollen haben standardmäßig nicht die Berechtigung, DeepRacer-Ressourcen zu erstellen oder zu ändern. Sie können auch keine Aufgaben mithilfe der AWS Management Console, AWS Command Line Interface (AWS CLI) oder AWS API ausführen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen. Der Administrator kann dann die IAM-Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen annehmen.

Informationen dazu, wie Sie unter Verwendung dieser beispielhaften JSON-Richtliniendokumente eine identitätsbasierte IAM-Richtlinie erstellen, finden Sie unter Erstellen von IAM-Richtlinien (Konsole) im IAM-Benutzerhandbuch.

Einzelheiten zu Aktionen und Ressourcentypen, die von definiert wurden DeepRacer, einschließlich des Formats von ARNs für jeden der Ressourcentypen, finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für AWS DeepRacer in der Service Authorization Reference.

Themen

- Bewährte Methoden für Richtlinien
- Verwenden der Konsole DeepRacer
- Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

Bewährte Methoden für Richtlinien

Identitätsbasierte Richtlinien legen fest, ob jemand DeepRacer Ressourcen in Ihrem Konto erstellen, darauf zugreifen oder sie löschen kann. Dies kann zusätzliche Kosten für Ihr verursachen AWS-Konto. Befolgen Sie beim Erstellen oder Bearbeiten identitätsbasierter Richtlinien die folgenden Anleitungen und Empfehlungen:

- Beginnen Sie mit AWS verwalteten Richtlinien und wechseln Sie zu Berechtigungen mit den geringsten Rechten — Verwenden Sie die AWS verwalteten Richtlinien, die Berechtigungen für viele gängige Anwendungsfälle gewähren, um Ihren Benutzern und Workloads zunächst Berechtigungen zu gewähren. Sie sind in Ihrem verfügbar. AWS-Konto Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie vom AWS Kunden verwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind. Weitere Informationen finden Sie unter AWS -verwaltete Richtlinien oder AWS -verwaltete Richtlinien für Auftrags-Funktionen im IAM-Benutzerhandbuch.

- Anwendung von Berechtigungen mit den geringsten Rechten – Wenn Sie mit IAM-Richtlinien Berechtigungen festlegen, gewähren Sie nur die Berechtigungen, die für die Durchführung einer Aufgabe erforderlich sind. Sie tun dies, indem Sie die Aktionen definieren, die für bestimmte Ressourcen unter bestimmten Bedingungen durchgeführt werden können, auch bekannt als die geringsten Berechtigungen. Weitere Informationen zur Verwendung von IAM zum Anwenden von Berechtigungen finden Sie unter Richtlinien und Berechtigungen in IAM im IAM-Benutzerhandbuch.
- Verwenden von Bedingungen in IAM-Richtlinien zur weiteren Einschränkung des Zugriffs – Sie können Ihren Richtlinien eine Bedingung hinzufügen, um den Zugriff auf Aktionen und Ressourcen zu beschränken. Sie können beispielsweise eine Richtlinienbedingung schreiben, um festzulegen, dass alle Anforderungen mithilfe von SSL gesendet werden müssen. Sie können auch Bedingungen verwenden, um Zugriff auf Serviceaktionen zu gewähren, wenn diese für einen bestimmten Zweck verwendet werden AWS-Service, z. AWS CloudFormation B. Weitere Informationen finden Sie unter IAM-JSON-Richtlinienelemente: Bedingung im IAM-Benutzerhandbuch.
- Verwenden von IAM Access Analyzer zur Validierung Ihrer IAM-Richtlinien, um sichere und funktionale Berechtigungen zu gewährleisten – IAM Access Analyzer validiert neue und vorhandene Richtlinien, damit die Richtlinien der IAM-Richtliniensprache (JSON) und den bewährten IAM-Methoden entsprechen. IAM Access Analyzer stellt mehr als 100 Richtlinienprüfungen und umsetzbare Empfehlungen zur Verfügung, damit Sie sichere und funktionale Richtlinien erstellen können. Weitere Informationen finden Sie unter Richtlinienvalidierung mit IAM Access Analyzer im IAM-Benutzerhandbuch.
- Multi-Faktor-Authentifizierung (MFA) erforderlich — Wenn Sie ein Szenario haben, das IAM-Benutzer oder einen Root-Benutzer in Ihrem System erfordert AWS-Konto, aktivieren Sie MFA für zusätzliche Sicherheit. Um MFA beim Aufrufen von API-Vorgängen anzufordern, fügen Sie Ihren Richtlinien MFA-Bedingungen hinzu. Weitere Informationen finden Sie unter Sicherer API-Zugriff mit MFA im IAM-Benutzerhandbuch.

Weitere Informationen zu bewährten Methoden in IAM finden Sie unter Bewährte Methoden für die Sicherheit in IAM im IAM-Benutzerhandbuch.

Verwenden der Konsole DeepRacer

Um auf die AWS DeepRacer Konsole zugreifen zu können, benötigen Sie ein Mindestmaß an Berechtigungen. Diese Berechtigungen müssen es Ihnen ermöglichen, Details zu den DeepRacer Ressourcen in Ihrem aufzulisten und anzuzeigen AWS-Konto. Wenn Sie eine identitätsbasierte

Richtlinie erstellen, die strenger ist als die mindestens erforderlichen Berechtigungen, funktioniert die Konsole nicht wie vorgesehen für Entitäten (Benutzer oder Rollen) mit dieser Richtlinie.

Sie müssen Benutzern, die nur die API AWS CLI oder die AWS API aufrufen, keine Mindestberechtigungen für die Konsole gewähren. Stattdessen sollten Sie nur Zugriff auf die Aktionen zulassen, die der API-Operation entsprechen, die die Benutzer ausführen möchten.

Um sicherzustellen, dass Benutzer und Rollen die DeepRacer Konsole weiterhin verwenden können, fügen Sie den Entitäten auch die DeepRacer `ConsoleAccess` oder die `ReadOnly` AWS verwaltete Richtlinie hinzu. Weitere Informationen finden Sie unter Hinzufügen von Berechtigungen zu einem Benutzer im IAM-Benutzerhandbuch.

Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

In diesem Beispiel wird gezeigt, wie Sie eine Richtlinie erstellen, die IAM-Benutzern die Berechtigung zum Anzeigen der eingebundenen Richtlinien und verwalteten Richtlinien gewährt, die ihrer Benutzeridentität angefügt sind. Diese Richtlinie umfasst Berechtigungen zum Ausführen dieser Aktion auf der Konsole oder programmgesteuert mithilfe der AWS CLI AWS OR-API.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
```

```
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS verwaltete Richtlinien für AWS DeepRacer

Eine AWS verwaltete Richtlinie ist eine eigenständige Richtlinie, die von erstellt und verwaltet wird AWS. AWS Verwaltete Richtlinien dienen dazu, Berechtigungen für viele gängige Anwendungsfälle bereitzustellen, sodass Sie damit beginnen können, Benutzern, Gruppen und Rollen Berechtigungen zuzuweisen.

Beachten Sie, dass AWS verwaltete Richtlinien für Ihre speziellen Anwendungsfälle möglicherweise keine Berechtigungen mit den geringsten Rechten gewähren, da sie allen AWS Kunden zur Verfügung stehen. Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie vom Kunden verwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind.

Sie können die in AWS verwalteten Richtlinien definierten Berechtigungen nicht ändern. Wenn die in einer AWS verwalteten Richtlinie definierten Berechtigungen AWS aktualisiert werden, wirkt sich das Update auf alle Prinzidentitäten (Benutzer, Gruppen und Rollen) aus, denen die Richtlinie zugeordnet ist. AWS aktualisiert eine AWS verwaltete Richtlinie höchstwahrscheinlich, wenn eine neue Richtlinie eingeführt AWS-Service wird oder neue API-Operationen für bestehende Dienste verfügbar werden.

Weitere Informationen finden Sie unter Von AWS verwaltete Richtlinien im IAM-Benutzerhandbuch.

Die folgenden AWS verwalteten Richtlinien gelten speziell für die Verwendung des DeepRacer AWS-Mehrbenutzermodus, um mehrere Teilnehmer unter Ihrem AWS Konto zu sponsern.

- `AWSDeepRacerAccountAdminAccess`Erteilt die erforderlichen DeepRacer AWS-Berechtigungen für den Administrator von Mehrbenutzerkonten.
- `AWSDeepRacerDefaultMultiUserAccess`Erteilt die erforderlichen DeepRacer AWS-Berechtigungen zur Verwendung der DeepRacer AWS-Konsole.

Themen

- AWSDeepRacerAccountAdminAccess verwaltete Richtlinie für DeepRacer AWS-Administratoren
- AWSDeepRacerDefaultMultiUserAccess verwaltete Richtlinie für DeepRacer AWS-Mehrbenutzer-Rennfahrer
- DeepRacer AWS-Updates für AWS verwaltete Richtlinien

AWSDeepRacerAccountAdminAccess verwaltete Richtlinie für DeepRacer AWS-Administratoren

Um mehreren Profilen die Verwendung Ihrer AWS Konto-ID und Rechnungsinformationen bei AWS zu ermöglichen DeepRacer, fügen Sie die `AWSDeepRacerAccountAdminAccess` Richtlinie bei.

Sie können die `AWSDeepRacerAccountAdminAccess` Richtlinie an die IAM-Identität anhängen, die Sie für das Sponsoring anderer Rennfahrer verwenden möchten.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "DeepRacerAdminAccessStatement",
            "Effect": "Allow",
            "Action": [
                "deepracer:*"
            ],
            "Resource": [
                "*"
            ],
            "Condition":
            {
                "Null": {
                    "deepracer:UserToken": "true"
                }
            }
        }
    ]
}
```

AWSDeepRacerDefaultMultiUserAccess verwaltete Richtlinie für DeepRacer AWS-Mehrbenutzer-Rennfahrer

Die Richtlinie gewährt DeepRacer AWS-Rennfahrern Zugriff auf alle DeepRacer AWS-Aktionen mit Ausnahme von Administratoraktionen für Mehrbenutzerkonten. `AWSDeepRacerDefaultMultiUserAccess`

Sie können die `AWSDeepRacerDefaultMultiUserAccess` Richtlinie mit den IAM-Identitäten der Teilnehmer verknüpfen, die Sie unter Ihrem Konto sponsern möchten.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": [
    "deepracer:Add*",
    "deepracer:Remove*",
    "deepracer:Create*",
    "deepracer:Perform*",
    "deepracer:Clone*",
    "deepracer:Get*",
    "deepracer:List*",
    "deepracer:Edit*",
    "deepracer:Start*",
    "deepracer:Set*",
    "deepracer:Update*",
    "deepracer:Delete*",
    "deepracer:Stop*",
    "deepracer:Import*",
    "deepracer:Tag*",
    "deepracer:Untag*"
   ],
   "Resource": [
    "*"
   ],
   "Condition": {
    "Null": {
     "deepracer:UserToken": "false"
    },
    "Bool": {
     "deepracer:MultiUser": "true"
    }
```

```
    }
   },
   {
    "Effect": "Allow",
    "Action": [
     "deepracer:GetAccountConfig",
     "deepracer:GetTrack",
     "deepracer:ListTracks",
     "deepracer:TestRewardFunction"
    ],
    "Resource": [
     "*"
    ]
   },
   {
    "Effect": "Deny",
    "Action": [
     "deepracer:Admin*"
    ],
    "Resource": [
     "*"
    ]
   }
  ]
}
```

DeepRacer AWS-Updates für AWS verwaltete Richtlinien

Sehen Sie sich Details zu Aktualisierungen der AWS verwalteten Richtlinien für AWS an, DeepRacer seit dieser Service begonnen hat, diese Änderungen zu verfolgen. Abonnieren Sie den RSS-Feed auf der Seite DeepRacer AWS-Dokumentenverlauf, um automatische Benachrichtigungen über Änderungen an dieser Seite zu erhalten.

Änderung	Beschreibung	Datum
`AWSDeepRacerAccoun tAdminAccess` und `AWSDeepRacerDefaul`	Neue verwaltete Richtlinien wurden hinzugefügt, sodass Sie im Mehrbenutzermodus mehrere Teilnehmer unter	26. Oktober 2021

Änderung	Beschreibung	Datum
`tMultiUserAccess` Richtlinien wurden hinzugefügt	einem DeepRacer AWS-Konto sponsern können.	
AWS DeepRacer hat damit begonnen, Richtlinienänderungen zu verfolgen.	AWS DeepRacer hat damit begonnen, Änderungen für seine von AWS verwalteten Richtlinien nachzuverfolgen.	26. Oktober 2021

Serviceübergreifende Confused-Deputy-Prävention

Das Confused-Deputy-Problem ist ein Sicherheitsproblem, bei dem eine juristische Stelle, die nicht über die Berechtigung zum Ausführen einer Aktion verfügt, eine privilegiertere juristische Stelle zwingen kann, die Aktion auszuführen. In AWS, dienstübergreifender Identitätswechsel kann zum Problem des verwirrten Stellvertreters führen. Ein dienstübergreifender Identitätswechsel kann auftreten, wenn ein Dienst (der Anruf-Dienst) einen anderen Dienst anruft (den aufgerufenen Dienst). Der aufrufende Service kann manipuliert werden, um seine Berechtigungen zu verwenden, um Aktionen auf die Ressourcen eines anderen Kunden auszuführen, für die er sonst keine Zugriffsberechtigung haben sollte. Um dies zu verhindern, bietet AWS Tools, mit denen Sie Ihre Daten für alle Services mit Serviceprinzipalen schützen können, die Zugriff auf Ressourcen in Ihrem Konto erhalten haben.

Wir empfehlen, die Kontextschlüssel `aws:SourceArn`und die `aws:SourceAccount`globalen Bedingungsschlüssel in Ressourcenrichtlinien zu verwenden, um die Berechtigungen einzuschränken, die der AWSDeep RacerLong Ressource einen anderen Dienst gewähren. Wenn Sie beide globalen Bedingungskontextschlüssel verwenden, müssen der `aws:SourceAccount`-Wert und das Konto im `aws:SourceArn`-Wert dieselbe Konto-ID verwenden, wenn sie in derselben Richtlinienanweisung verwendet werden.

Der Wert von `aws:SourceArn` muss s3::: seinyour-bucket-name.

Der effektivste Weg, um sich vor dem Confused-Deputy-Problem zu schützen, ist die Verwendung des globalen Bedingungskontext-Schlüssels `aws:SourceArn` mit dem vollständigen ARN der Ressource. Wenn Sie den vollständigen ARN der Ressource nicht kennen oder wenn Sie mehrere Ressourcen angeben, verwenden Sie den globalen Bedingungskontext-Schlüssel `aws:SourceArn` mit Platzhaltern (`*`) für die unbekannten Teile des ARN. Beispiel, `arn:aws:servicename::123456789012:*`.

Das folgende Beispiel zeigt, wie Sie die Kontextschlüssel `aws:SourceArn` und die `aws:SourceAccount` globalen Bedingungsschlüssel in AWSDeep Racer verwenden können, um das Problem des verwirrten Stellvertreters zu verhindern.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Stmt1586917903457",
            "Effect": "Allow",
            "Principal": {
                "Service": "deepracer.amazonaws.com"
                },
            "Action": [
                "s3:GetObjectAcl",
                "s3:GetObject",
                "s3:PutObject",
                "s3:PutObjectAcl"
            ],
            "Resource": [
               "arn:aws:s3:::your-bucket-name",
               "arn:aws:s3:::your-bucket-name/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:SourceArn": "arn:${Partition}:deepracer:${Region}:
${Account}:model/reinforcement_learning/${ResourceId}"
                }
            }
        }
    ]
}
```

Wenn Sie eine benutzerdefinierte Ressource AWS Key Management Service (KMS) für diesen Bucket verwenden, fügen Sie die AWS KMS Ressourcenrichtlinie hinzu:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "statement identifier",
    "Effect": "effect",
    "Principal": "principal",
```

```
    "Action": "action",
    "Resource": "resource",
    "Condition": {
        "StringEquals": {
            "aws:SourceArn": "arn:${Partition}:deepracer:${Region}:${Account}:model/
reinforcement_learning/${ResourceId}"
        }
    }
  }]
}
```

Problembehandlung bei AWS DeepRacer Identität und Zugriff

Verwenden Sie die folgenden Informationen, um häufig auftretende Probleme zu diagnostizieren und zu beheben, die bei der Arbeit mit DeepRacer und IAM auftreten können.

Themen

- Ich erhalte einen Autorisierungsfehler im DeepRacer Mehrbenutzerkontenmodus
- Ich bin nicht autorisiert, eine Aktion in DeepRacer auszuführen.
- Ich bin nicht berechtigt, iam auszuführen: PassRole
- Ich möchte meine Zugriffsschlüssel anzeigen
- Ich bin Administrator und möchte anderen Zugriff auf DeepRacer gewähren.
- Ich möchte Personen außerhalb von mir den Zugriff auf meine Ressourcen ermöglichen AWS-Konto DeepRacer

Ich erhalte einen Autorisierungsfehler im DeepRacer Mehrbenutzerkontenmodus

Wenn Sie ein Administrator mit einer AWSDeepRacerAccountAdminAccessRichtlinie sind, wird möglicherweise ein Autorisierungsfehler angezeigt, wenn Ihrer Sitzung ein Benutzertoken zugeordnet ist. Administratoren sollten keine Benutzertoken mit einer Sitzung verknüpft haben. Um dieses Problem zu beheben, löschen Sie Ihre Cookies.

Wenn sich das Konto im Mehrbenutzermodus befindet und Sie ein Rennfahrer mit einer AWSDeepRacerDefaultMultiUserAccessRichtlinie sind, wird möglicherweise ein Autorisierungsfehler angezeigt, wenn Ihrer Richtlinie kein Benutzertoken zugeordnet ist. Um dieses Problem zu lösen, müssen Sie sich bei Ihrem AWS-Player-Profil authentifizieren, bevor Sie die Nutzung AWS DeepRacer fortsetzen können.

Wenn sich das Konto im Einzelbenutzermodus befindet und Sie ein Rennfahrer mit einer bestimmten `AWSDeepRacerDefaultMultiUserAccess` Richtlinie sind, wird möglicherweise ein Autorisierungsfehler angezeigt. Um dieses Problem zu beheben, wenden Sie sich an Ihren AWS Kontoadministrator, da ein Benutzer mit einer `AWSDeepRacerDefaultMultiUserAccess` Richtlinie im Einzelbenutzermodus keinen Zugriff darauf hat. AWS DeepRacer

Ich bin nicht autorisiert, eine Aktion in DeepRacer auszuführen.

Wenn Ihnen AWS Management Console mitgeteilt wird, dass Sie nicht berechtigt sind, eine Aktion durchzuführen, müssen Sie sich an Ihren Administrator wenden, um Unterstützung zu erhalten. Ihr Administrator ist die Person, die Ihnen Ihren Benutzernamen und Ihr Passwort bereitgestellt hat.

Der folgende Beispielfehler tritt auf, wenn der `mateojackson` IAM-Benutzer versucht, die Konsole zum Anzeigen von Details zu einer fiktiven *`my-example-widget`*-Ressource zu verwenden, jedoch nicht über `deepracer:`*`GetWidget`*-Berechtigungen verfügt.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 deepracer:GetWidget on resource: my-example-widget
```

In diesem Fall bittet Mateo seinen Administrator um die Aktualisierung seiner Richtlinien, um unter Verwendung der Aktion *`my-example-widget`* auf die Ressource `deepracer:`*`GetWidget`* zugreifen zu können.

Ich bin nicht berechtigt, iam auszuführen: PassRole

Wenn Sie die Fehlermeldung erhalten, dass Sie nicht zum Durchführen der `iam:PassRole`-Aktion autorisiert sind, müssen Ihre Richtlinien aktualisiert werden, um eine Rolle an DeepRacer übergeben zu können.

Einige AWS-Services ermöglichen es Ihnen, eine bestehende Rolle an diesen Dienst zu übergeben, anstatt eine neue Servicerolle oder eine dienstverknüpfte Rolle zu erstellen. Hierzu benötigen Sie Berechtigungen für die Übergabe der Rolle an den Dienst.

Der folgende Beispielfehler tritt auf, wenn ein IAM-Benutzer mit dem Namen `marymajor` versucht, die Konsole zu verwenden, um eine Aktion in DeepRacer auszuführen. Die Aktion erfordert jedoch, dass der Service über Berechtigungen verfügt, die durch eine Servicerolle gewährt werden. Mary besitzt keine Berechtigungen für die Übergabe der Rolle an den Dienst.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

In diesem Fall müssen die Richtlinien von Mary aktualisiert werden, um die Aktion `iam:PassRole` ausführen zu können.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen odzur Verfügung gestellt.

Ich möchte meine Zugriffsschlüssel anzeigen

Nachdem Sie Ihre IAM-Benutzerzugriffsschlüssel erstellt haben, können Sie Ihre Zugriffsschlüssel-ID jederzeit anzeigen. Sie können Ihren geheimen Zugriffsschlüssel jedoch nicht erneut anzeigen. Wenn Sie den geheimen Zugriffsschlüssel verlieren, müssen Sie ein neues Zugriffsschlüsselpaar erstellen.

Zugriffsschlüssel bestehen aus zwei Teilen: einer Zugriffsschlüssel-ID (z. B. `AKIAIOSFODNN7EXAMPLE`) und einem geheimen Zugriffsschlüssel (z. B. `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`). Ähnlich wie bei Benutzernamen und Passwörtern müssen Sie die Zugriffsschlüssel-ID und den geheimen Zugriffsschlüssel zusammen verwenden, um Ihre Anforderungen zu authentifizieren. Verwalten Sie Ihre Zugriffsschlüssel so sicher wie Ihren Benutzernamen und Ihr Passwort.

> **Important**
>
> Geben Sie Ihre Zugriffsschlüssel nicht an Dritte weiter, auch nicht für die Suche nach Ihrer kanonischen Benutzer-ID. Auf diese Weise können Sie jemandem dauerhaften Zugriff auf Ihre gewähren AWS-Konto.

Während der Erstellung eines Zugriffsschlüsselpaars werden Sie aufgefordert, die Zugriffsschlüssel-ID und den geheimen Zugriffsschlüssel an einem sicheren Speicherort zu speichern. Der geheime Zugriffsschlüssel ist nur zu dem Zeitpunkt verfügbar, an dem Sie ihn erstellen. Wenn Sie Ihren geheimen Zugriffsschlüssel verlieren, müssen Sie Ihrem IAM-Benutzer neue Zugriffsschlüssel hinzufügen. Sie können maximal zwei Zugriffsschlüssel besitzen. Wenn Sie bereits zwei Zugriffschlüssel besitzen, müssen Sie ein Schlüsselpaar löschen, bevor Sie ein neues erstellen. Anweisungen hierfür finden Sie unter Verwalten von Zugriffsschlüsseln im IAM-Benutzerhandbuch.

Ich bin Administrator und möchte anderen Zugriff auf DeepRacer gewähren.

Um anderen den Zugriff zu ermöglichen DeepRacer, müssen Sie den Personen oder Anwendungen, die Zugriff benötigen, die Erlaubnis erteilen. Wenn Sie Personen und Anwendungen verwalten, weisen Sie Benutzern oder Gruppen Berechtigungssätze zu, um deren Zugriffsebene zu definieren.

AWS IAM Identity Center Mit Berechtigungssätzen werden automatisch IAM-Richtlinien erstellt und den IAM-Rollen zugewiesen, die der Person oder Anwendung zugeordnet sind. Weitere Informationen finden Sie im AWS IAM Identity Center Benutzerhandbuch unter Berechtigungssätze.

Wenn Sie IAM Identity Center nicht verwenden, müssen Sie IAM-Entitäten (Benutzer oder Rollen) für die Personen oder Anwendungen erstellen, die Zugriff benötigen. Anschließend müssen Sie der Entität eine Richtlinie anfügen, die dieser die korrekten Berechtigungen in DeepRacer gewährt. Nachdem die Berechtigungen erteilt wurden, stellen Sie dem Benutzer oder Anwendungsentwickler die Anmeldeinformationen zur Verfügung. Sie werden diese Anmeldeinformationen für den Zugriff verwenden AWS. Weitere Informationen zum Erstellen von IAM-Benutzern, -Gruppen, -Richtlinien und -Berechtigungen finden Sie im IAM-Benutzerhandbuch unter IAM-Identitäten sowie Richtlinien und Berechtigungen in IAM.

Ich möchte Personen außerhalb von mir den Zugriff auf meine Ressourcen ermöglichen AWS-Konto DeepRacer

Sie können eine Rolle erstellen, die Benutzer in anderen Konten oder Personen außerhalb Ihrer Organisation für den Zugriff auf Ihre Ressourcen verwenden können. Sie können festlegen, wem die Übernahme der Rolle anvertraut wird. Für Dienste, die ressourcenbasierte Richtlinien oder Zugriffskontrolllisten (ACLs) unterstützen, können Sie diese Richtlinien verwenden, um Personen Zugriff auf Ihre Ressourcen zu gewähren.

Weitere Informationen dazu finden Sie hier:

- Informationen darüber, ob diese Funktionen DeepRacer unterstützt werden, finden Sie unter. Wie AWS DeepRacer funktioniert mit IAM
- Informationen dazu, wie Sie Zugriff auf Ihre Ressourcen gewähren können, AWS-Konten die Ihnen gehören, finden Sie im IAM-Benutzerhandbuch unter Gewähren des Zugriffs auf einen IAM-Benutzer in einem anderen AWS-Konto , den Sie besitzen.
- Informationen dazu, wie Sie Dritten Zugriff auf Ihre Ressourcen gewähren können AWS-Konten, finden Sie AWS-Konten im IAM-Benutzerhandbuch unter Gewähren des Zugriffs für Dritte.
- Informationen dazu, wie Sie über einen Identitätsverbund Zugriff gewähren, finden Sie unter Gewähren von Zugriff für extern authentifizierte Benutzer (Identitätsverbund) im IAM-Benutzerhandbuch.
- Informationen zum Unterschied zwischen der Verwendung von Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

Tagging

Ein Tag ist eine benutzerdefinierte Attributbezeichnung, die Sie einer AWS Ressource AWS zuweisen oder zuweisen. Jedes AWS Tag besteht aus zwei Teilen:

- Ein Tag-Schlüssel (zum Beispiel`companyname`,`costcenter`,`environment`,`project`, oder`secret`). Bei Tag-Schlüsseln wird zwischen Groß- und Kleinschreibung unterschieden.
- Ein optionales Feld, das als Tag-Wert bezeichnet wird. Ein nicht angegebener Tag-Wert entspricht einer leeren Zeichenfolge. Wie bei Tag-Schlüsseln wird auch bei Tag-Werten zwischen Groß- und Kleinschreibung unterschieden.

Zusammen werden sie als Schlüssel-Wert-Paare bezeichnet.

Im DeepRacer AWS-Service können Sie Autos, RL-Modellen und Bestenlisten für Community-Rennen Tags zuweisen. Kennzeichnen Sie diese und andere AWS Ressourcen, die Tagging unterstützen, um darauf hinzuweisen, dass die Ressourcen miteinander verknüpft sind. Neben der Identifizierung und Organisation Ihrer Modelle und Bestenlisten mit Stichwörtern können Sie Tags auch verwenden, um die Kostenzuweisung nachzuverfolgen und in IAM-Richtlinien zu kontrollieren, wer Ihre Ressourcen einsehen und mit ihnen interagieren kann. Verwenden Sie die DeepRacer AWS-Konsole oder die, AWS CLI um Tags hinzuzufügen, zu verwalten und zu entfernen.

Weitere Informationen zur Verwendung von Tags finden Sie im Whitepaper Bewährte Methoden für die Markierung.

Tag zur Nachverfolgung der Kostenzuweisung

AWS Cost Explorer und Cost and Usage Report unterstützen die Möglichkeit, die AWS-Kosten nach Tags aufzuschlüsseln. Geschäftstags wie `cost centerbusinessunit`, oder `project` können verwendet werden, um AWS-Kosten den typischen Kategorien der Finanzberichterstattung einer Organisation zuzuordnen. Ein Kostenzuordnungsbericht kann jedoch jedes beliebige Schlagwort enthalten, sodass Sie die Kosten auf einfache Weise technischen oder sicherheitstechnischen Kategorien zuordnen können, z. B. bestimmten Anwendungen, Umgebungen oder Compliance-Programmen. In einer Organisation haben ausschließlich Verwaltungskonten und einzelne Konten, die nicht Mitglied einer Organisation sind, Zugriff auf den Manager für Kostenzuordnungs-Tags in der Konsole für Fakturierung und Kostenmanagement. Weitere Informationen zur Verwendung von Tags zur Nachverfolgung der Kostenzuweisung finden Sie unter Benutzerdefinierte Kostenzuordnungs-Tags.

Tag zur Zugriffsverwaltung

Sie können auch IAM-Benutzer und -Rollen taggen, um den Zugriff auf Ihre Modelle und Bestenlisten der Community-Rennen zu verwalten. Informationen zum Taggen von IAM-Benutzern und -Rollen finden Sie unter Taggen von IAM-Benutzern und -Rollen. Ein Tutorial zum Erstellen und Testen einer Richtlinie, die es IAM-Rollen mit Prinzipal-Tags ermöglicht, auf Ressourcen mit passenden Tags zuzugreifen, finden Sie unter IAM-Tutorial: Berechtigungen für den Zugriff AWS auf Ressourcen auf der Grundlage von Tags definieren. Weitere Informationen zur Verwendung von Tags zur Steuerung des Zugriffs auf Ihre AWS Ressourcen, die Tagging unterstützen, finden Sie unter Steuern des Zugriffs auf AWS Ressourcen mithilfe von Ressourcen-Tags.

Themen

- Hinzufügen, Anzeigen und Bearbeiten von Tags für eine neue Ressource
- Fügen Sie Tags für eine vorhandene Ressource hinzu, zeigen Sie sie an und bearbeiten Sie sie

Hinzufügen, Anzeigen und Bearbeiten von Tags für eine neue Ressource

Wenn Sie Tags zu einem neuen Auto, einem RL-Modell oder einer Bestenliste für Gemeinschaftsrennen hinzufügen, können Sie diese Ressourcen leichter identifizieren, organisieren, die Kostenzuweisung verfolgen und den Zugriff auf diese Ressourcen verwalten. Fügen Sie einem Modell oder einer Bestenliste ein oder mehrere Tags (Schlüssel-Wert-Paare) hinzu. Für jede Ressource muss jeder Tag-Schlüssel einzigartig sein, und jeder Tag-Schlüssel kann nur einen Wert haben, aber eine Ressource kann bis zu 50 Tags haben.

Erstellen und wenden Sie die Tags ressourcenweise in der DeepRacer AWS-Konsole an oder verwenden Sie den Tag-Editor, um mehrere Ressourcen gleichzeitig hinzuzufügen, zu bearbeiten oder zu löschen.

> **Important**
>
> Die Bearbeitung von Tags für ein RL-Modell oder die Bestenliste für Community-Rennen kann sich auf den Zugriff auf diese Ressourcen auswirken. Bevor Sie den Namen (Schlüssel) oder den Wert eines Tags bearbeiten, sollten Sie alle IAM-Richtlinien überprüfen, die den Schlüssel oder Wert für ein Tag verwenden könnten, um den Zugriff auf diese Ressourcen zu kontrollieren.

Um Tags für ein neues RL-Modell hinzuzufügen, anzuzeigen und zu bearbeiten

Verwenden Sie die DeepRacer AWS-Konsole, um Tags zu einem neuen RL-Modell hinzuzufügen, anzuzeigen und zu bearbeiten.

1. Wählen Sie unter Ihre Modelle die Option Modell erstellen aus.
2. Erweitern Sie auf der Seite Modell erstellen nach dem Ausfüllen der Trainingsdetails die Überschrift Tags.
3. Wählen Sie unter der Überschrift Tags die Option Neues Tag hinzufügen aus.
4. Geben Sie für Schlüssel einen Namen für das Tag ein. Sie können einen optionalen Wert für das Tag unter Wert hinzufügen. Weitere Informationen zur Benennung von Tags finden Sie im Thema Bewährte Methoden für die Benennung von Tags und Ressourcen im Whitepaper Best Practices für Tagging.
5. (Optional) Zum Hinzufügen eines weiteren Tags wählen Sie Neues Tag hinzufügen erneut aus.
6. (Optional) Um einen einzelnen Schlüssel oder Wert zu entfernen, wählen Sie das X neben dem Schlüssel oder Wert aus.
7. (Optional) Um ein Schlüssel-Wert-Paar zu entfernen, wählen Sie Entfernen aus.
8. Wenn Sie mit dem Hinzufügen von Tags fertig sind, wählen Sie unter Umgebungssimulation eine Spur aus und klicken Sie auf Weiter.

Nachdem Sie ein neues Modell markiert und zum Training eingereicht haben, können Sie die zugehörigen Tags während oder nach dem Training und der Evaluierung unter der Überschrift „Tags“ unten auf der Seite verwalten.

1. Wählen Sie Tags verwalten aus.
2. Im Popup-Feld „Schlagworte verwalten“ können Sie ein von Ihnen erstelltes Tag entfernen, indem Sie neben dem Tag, das Sie entfernen möchten, auf die Schaltfläche Entfernen klicken oder Neues Tag hinzufügen wählen, um ein neues Tag hinzuzufügen.
3. Wenn Sie ein neues Tag hinzufügen möchten, geben Sie im Feld Schlüssel einen Namen für das Tag ein. Sie können einen optionalen Wert für das Tag unter Wert hinzufügen. Weitere Informationen zur Benennung von Tags finden Sie im Thema Bewährte Methoden für die Benennung von Tags und Ressourcen im Whitepaper Best Practices für Tagging.
4. Wenn Sie mit dem Entfernen und Hinzufügen von Tags fertig sind, wählen Sie Absenden.

Um Tags für eine neue Bestenliste für Community-Rennen hinzuzufügen, anzusehen und zu bearbeiten

Verwenden Sie die DeepRacer AWS-Konsole, um Stichwörter zu einer neuen Bestenliste für Community-Rennen hinzuzufügen, anzuzeigen und zu bearbeiten.

1. Wählen Sie unter Community-Rennen die Option Rennen erstellen aus.
2. Erweitere auf der Seite mit den Renndetails die Überschrift Tags.
3. Wähle unter der Überschrift „Schlagworte“ die Option Neues Tag hinzufügen aus.
4. Geben Sie für Schlüssel einen Namen für das Tag ein. Sie können einen optionalen Wert für das Tag unter Wert hinzufügen. Weitere Informationen zur Benennung von Tags finden Sie im Thema Bewährte Methoden für die Benennung von Tags und Ressourcen im Whitepaper Best Practices für Tagging.
5. (Optional) Zum Hinzufügen eines weiteren Tags wählen Sie Neues Tag hinzufügen erneut aus.
6. (Optional) Um einen einzelnen Schlüssel oder Wert zu entfernen, wählen Sie das X neben dem Schlüssel oder Wert aus.
7. (Optional) Um ein Schlüssel-Wert-Paar zu entfernen, wählen Sie Entfernen aus.
8. Wenn Sie mit dem Hinzufügen von Tags fertig sind, wählen Sie unter Umgebungssimulation eine Spur aus und klicken Sie auf Weiter.

Fügen Sie Tags für eine vorhandene Ressource hinzu, zeigen Sie sie an und bearbeiten Sie sie

Das Hinzufügen von Tags zu einem bestehenden AWS DeepRacer RL-Modell oder einer Bestenliste für Community-Rennen kann Ihnen helfen, diese Ressourcen zu identifizieren, zu organisieren, die Kostenzuweisung zu verfolgen und den Zugriff auf diese Ressourcen zu verwalten. Fügen Sie einem Modell oder einer Bestenliste ein oder mehrere Tags (Schlüssel-Wert-Paare) hinzu. Für jede Ressource muss jeder Tag-Schlüssel einzigartig sein, und jeder Tag-Schlüssel kann nur einen Wert haben, aber eine Ressource kann bis zu 50 Tags haben.

Erstellen und wenden Sie die Tags ressourcenweise in der DeepRacer AWS-Konsole an oder verwenden Sie den Tag-Editor, um mehrere Ressourcen gleichzeitig hinzuzufügen, zu bearbeiten oder zu löschen.

> **Important**
>
> Die Bearbeitung von Tags für ein RL-Modell oder die Bestenliste für Community-Rennen kann sich auf den Zugriff auf diese Ressourcen auswirken. Bevor Sie den Namen (Schlüssel) oder den Wert eines Tags bearbeiten, sollten Sie alle IAM-Richtlinien überprüfen, die den Schlüssel oder Wert für ein Tag verwenden könnten, um den Zugriff auf diese Ressourcen zu kontrollieren.

Um Tags für ein vorhandenes RL-Modell hinzuzufügen, anzuzeigen und zu bearbeiten

Sie können die DeepRacer AWS-Konsole verwenden, um Tags für ein vorhandenes RL-Modell hinzuzufügen, anzuzeigen oder zu bearbeiten.

1. Wählen Sie unter Ihre Modelle ein Modell aus der Liste aus, indem Sie seinen Namen wählen.
2. Wählen Sie Aktionen aus.
3. Wählen Sie in der Drop-down-Liste die Option Tags verwalten aus.
4. Im Popup-Feld „Tags verwalten“ können Sie Tags anzeigen, hinzufügen oder entfernen:
    a. Um einen Tag hinzuzufügen, wählen Sie Add new tag (Neuen Tag hinzufügen). Geben Sie für Schlüssel einen Namen für das Tag ein. Sie können einen optionalen Wert für das Tag unter Wert hinzufügen. Weitere Informationen zur Benennung von Tags finden Sie im Thema Bewährte Methoden für die Benennung von Tags und Ressourcen im Whitepaper Best Practices für Tagging.
    b. (Optional) Zum Hinzufügen eines weiteren Tags wählen Sie Add new tag (Neues Tag hinzufügen) erneut aus.
    c. Um einen einzelnen Schlüssel oder Wert zu entfernen, wählen Sie das X neben dem Schlüssel oder Wert aus.
    d. Um ein Schlüssel-Wert-Paar zu entfernen, wählen Sie Entfernen.
5. Wenn Sie mit dem Anzeigen, Hinzufügen und Entfernen von Tags fertig sind, wählen Sie Absenden.

Um Stichwörter für eine bestehende Bestenliste für Community-Rennen hinzuzufügen, anzusehen und zu bearbeiten

1. Wähle unter Community-Rennen die Option Rennen verwalten aus.

2. Wähle auf der Seite „Rennen verwalten“ ein Rennen aus.
3. Wählen Sie Aktionen aus.
4. Wähle in der Drop-down-Liste die Option Schlagworte verwalten aus.
5. Im Popup-Feld „Tags verwalten“ können Sie Tags anzeigen, hinzufügen oder entfernen:

    a. Um einen Tag hinzuzufügen, wählen Sie Add new tag (Neuen Tag hinzufügen). Geben Sie für Schlüssel einen Namen für das Tag ein. Sie können einen optionalen Wert für das Tag unter Wert hinzufügen. Weitere Informationen zur Benennung von Tags finden Sie im Thema Bewährte Methoden für die Benennung von Tags und Ressourcen im Whitepaper Best Practices für Tagging.

    b. (Optional) Zum Hinzufügen eines weiteren Tags wählen Sie Add new tag (Neues Tag hinzufügen) erneut aus.

    c. Um einen einzelnen Schlüssel oder Wert zu entfernen, wählen Sie das X neben dem Schlüssel oder Wert aus.

    d. Um ein Schlüssel-Wert-Paar zu entfernen, wählen Sie Entfernen.
6. Wenn Sie mit dem Anzeigen, Hinzufügen und Entfernen von Tags fertig sind, wählen Sie Absenden.

Behebung häufiger DeepRacer AWS-Probleme

Hier finden Sie Tipps zur Fehlerbehebung bei häufig gestellten Fragen sowie aktuelle Fehlerkorrekturen.

Themen

- So lösen Sie häufig auftretende Probleme mit AWS DeepRacer LIVE
- Warum kann ich über eine USB-Verbindung zwischen meinem Computer und dem Fahrzeug keine Verbindung zur Gerätekonsole herstellen?
- So schalten Sie die Stromquelle des DeepRacer AWS-Rechenmoduls von der Batterie auf eine Steckdose um
- So verwenden Sie ein USB-Flash-Laufwerk, um AWS mit Ihrem Wi-Fi-Netzwerk DeepRacer zu verbinden
- So laden Sie den Akku des DeepRacer AWS-Laufwerksmoduls auf
- So laden Sie den Akku des DeepRacer AWS-Rechenmoduls auf
- Mein Akku ist geladen, aber mein DeepRacer AWS-Fahrzeug bewegt sich nicht
- Fehlerbehebung bei der Sperrung der DeepRacer AWS-Fahrzeugbatterie
- So wickeln Sie ein Dell-Akkuanschlusskabel bei der Installation eines LiDAR-Sensors ein
- So halten Sie die Wi-Fi-Verbindung Ihres Fahrzeugs aufrecht
- So erhalten Sie die Mac-Adresse Ihres DeepRacer AWS-Geräts
- So stellen Sie das Standardkennwort Ihrer DeepRacer AWS-Gerätekonsole wieder her
- So aktualisieren Sie Ihr DeepRacer AWS-Gerät manuell
- So diagnostizieren und lösen Sie häufig auftretende DeepRacer AWS-Betriebsprobleme

So lösen Sie häufig auftretende Probleme mit AWS DeepRacer LIVE

Ich kann das Rennvideo auf der LIVE-Rennseite nicht sehen

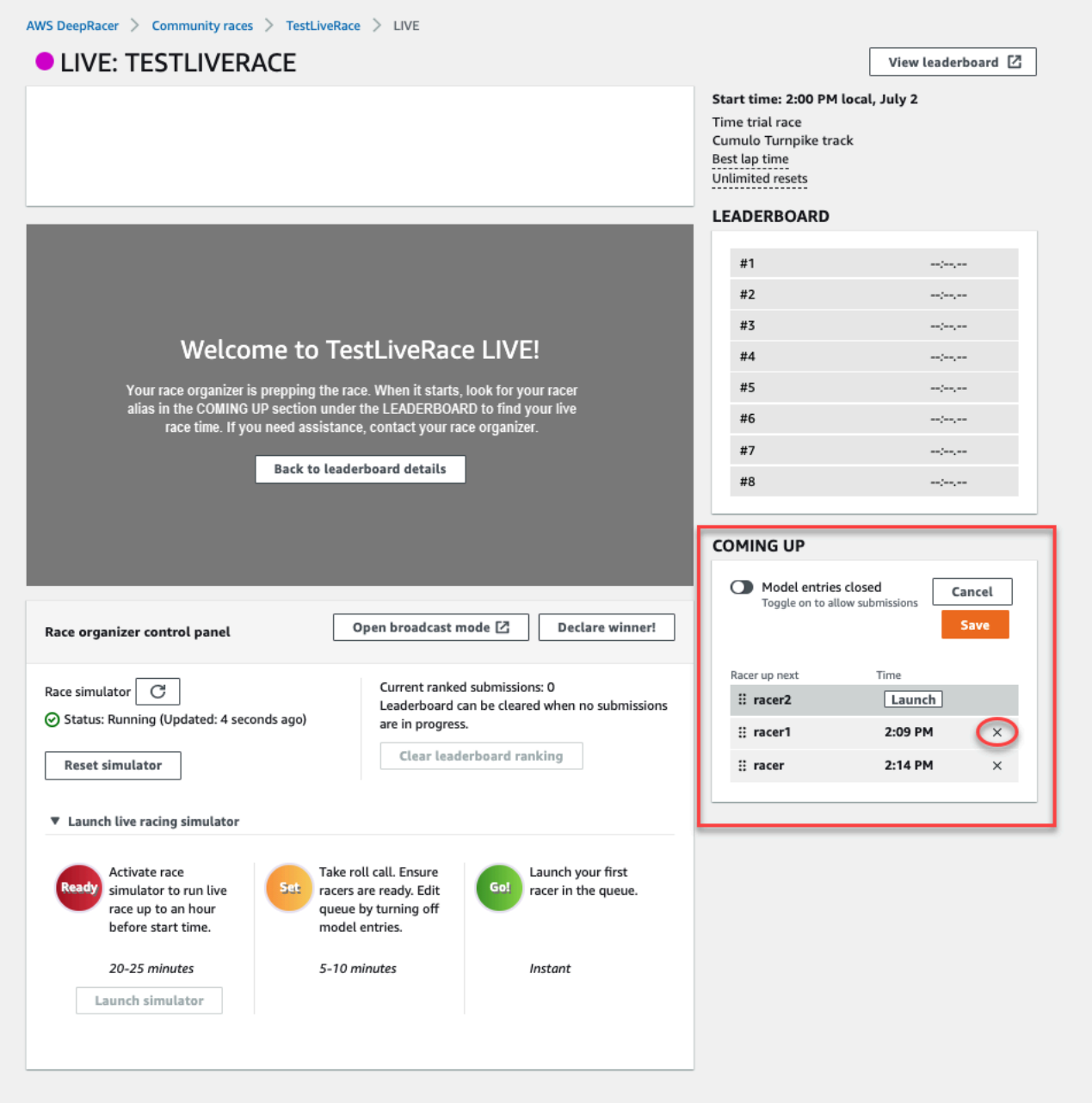
- Wenn du ein virtuelles privates Netzwerk (VPN) verwendest, vergewissere dich, dass es während der Rennveranstaltung unterbrochen wurde.

- Wenn auf Ihrem Gerät ein Werbeblocker läuft, vergewissern Sie sich, dass die Verbindung während der Rennveranstaltung unterbrochen wurde.
- Wenn in deinem Heimnetzwerk ein Werbeblocker läuft, vergewissere dich, dass die Verbindung während der Rennveranstaltung unterbrochen wurde.

Der Name eines Rennfahrers in der Rennwarteschlange ist rot

Wenn der Name eines Rennfahrers im Bereich „Deming up“ der LIVE: <Your Race Name>-Seite rot markiert ist, bedeutet das, dass bei der Einreichung des Modells des Rennfahrers ein Fehler aufgetreten ist.

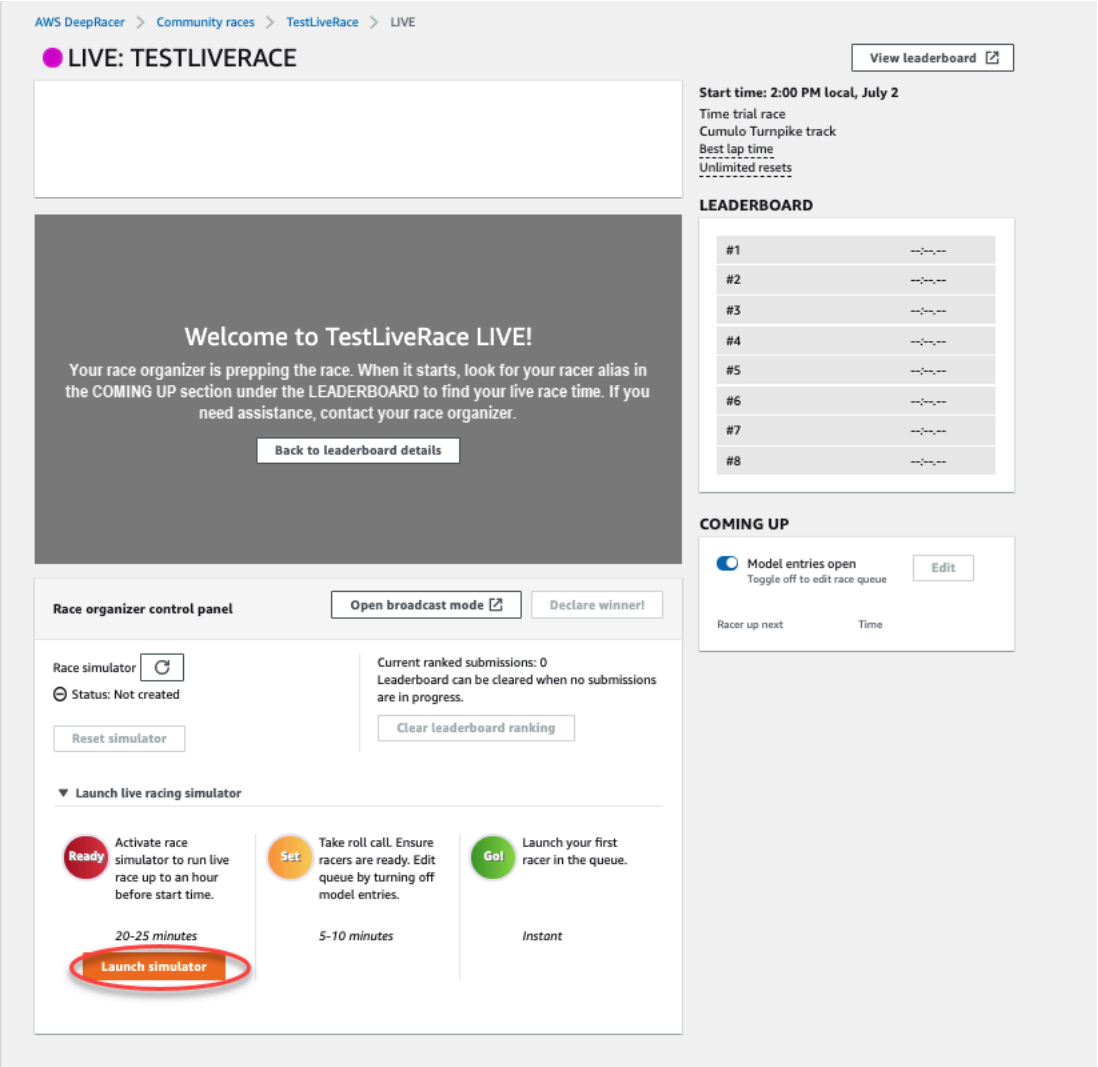
- Wenn Sie ein Rennorganisator sind, <Your Race Name>wählen Sie im Bereich „Demnächst“ der Seite „LIVE:“ die Option „Bearbeiten“ aus, um das eingereichte Modell des Rennfahrers zu löschen, indem Sie in der Zeile, die den Namen des Rennfahrers enthält, auf X klicken. Wählen Sie als Nächstes „Speichern“. Hilfe the section called “Führen Sie ein LIVE-Rennen durch” zum Neuordnen Ihrer Warteschlange finden Sie in Schritt 11 von.



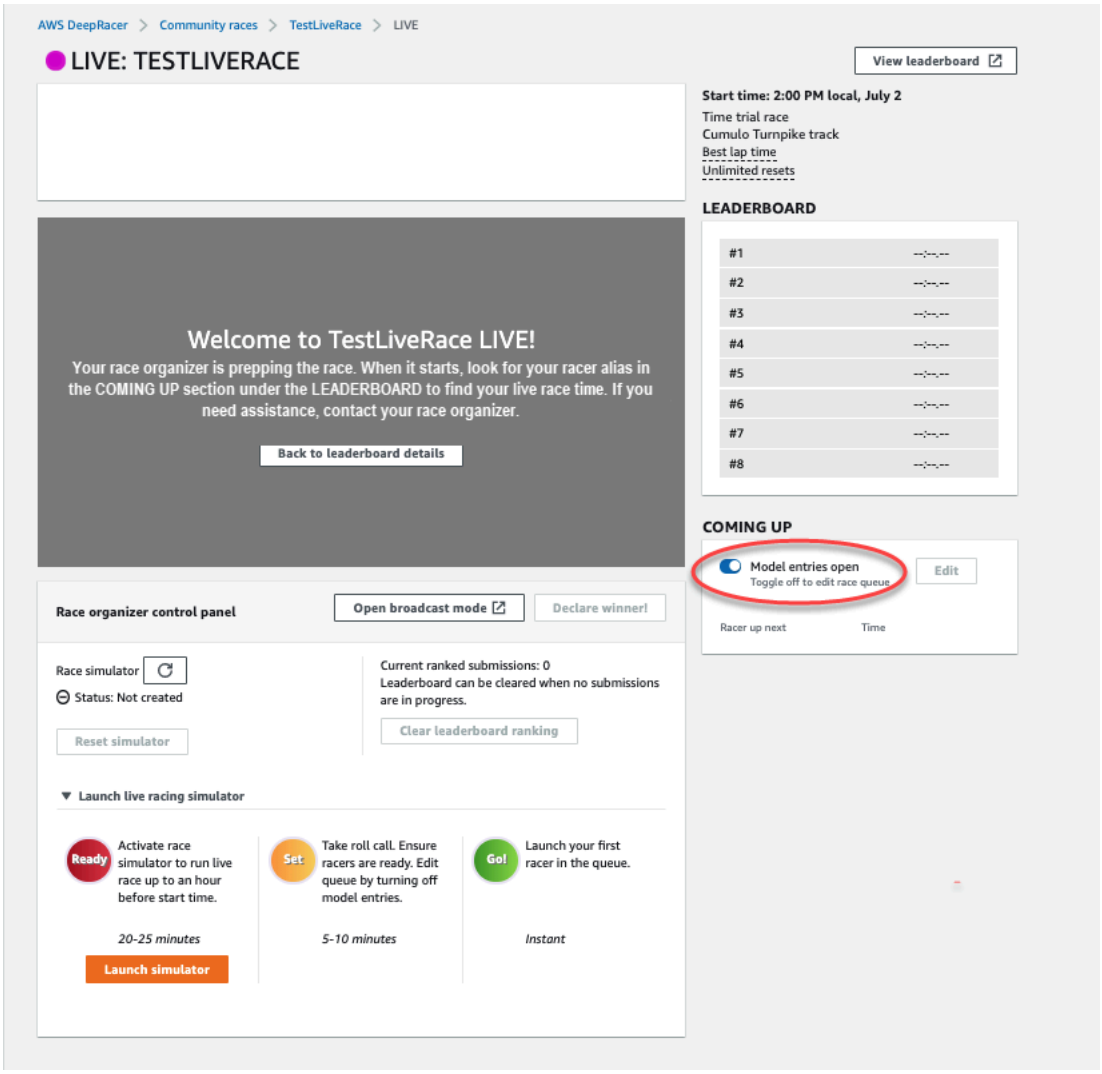
- Wenn Sie ein Rennfahrer sind, reichen Sie Ihr Modell erneut für das Rennen ein. Gehe zu the section called “Führen Sie ein LIVE-Rennen durch” und wähle An einem LIVE-Rennen teilnehmen, um Hilfe zu erhalten.

Ich führe ein LIVE-Rennen durch und kann die Rennfahrer nicht starten

- Vergewissere dich, dass du auf der LIVE: -Seite im Bereich Live-Rennsimulator starten die Option Simulator starten ausgewählt hast. <Your Race Name> Weitere Hilfe finden Sie in Schritt zwei vonthe section called “Führen Sie ein LIVE-Rennen durch”.



- Vergewissern Sie sich, dass Sie unter DEMNÄCHST auf der <Your Race Name>Seite LIVE: die Option Modelleinträge öffnen und schließen deaktiviert haben. Weitere Hilfe finden Sie in Schritt drei von. the section called “Führen Sie ein LIVE-Rennen durch”



Ich verwende einen Chrome- oder Firefox-Browser, habe aber immer noch Probleme, das LIVE-Rennen zu sehen

- Vergewissern Sie sich, dass Sie die neueste Version des Chrome- oder Firefox-Browsers haben. Wenn nicht, aktualisieren Sie Ihren Browser auf die neueste Version und versuchen Sie erneut, das Rennen anzusehen.
- Wenn Sie ein virtuelles privates Netzwerk (VPN) verwenden, stellen Sie sicher, dass die Verbindung unterbrochen ist.
- Wenn auf Ihrem Gerät ein Werbeblocker läuft, stellen Sie sicher, dass die Verbindung während des Rennens unterbrochen wurde.

- Wenn in deinem Heimnetzwerk ein Werbeblocker läuft, vergewissere dich, dass die Verbindung während der Rennveranstaltung unterbrochen wurde.
- Wenn WebRTC in Ihrem Internetbrowser ausgeschaltet ist, schalten Sie es während der Rennveranstaltung ein.

Warum kann ich über eine USB-Verbindung zwischen meinem Computer und dem Fahrzeug keine Verbindung zur Gerätekonsole herstellen?

Wenn Sie Ihr Fahrzeug zum ersten Mal einrichten, stellen Sie möglicherweise fest, dass es die Gerätekonsole (auch bekannt als Geräte-Webserver, auf dem Fahrzeug gehostet) nicht öffnen kann`https://deepracer.aws`, nachdem Sie Ihr DeepRacer AWS-Fahrzeug mit einem Micro-USB/USB-Kabel (USB wird auch als USB-A bezeichnet) an Ihren Computer angeschlossen haben.

Dies kann mehrere Ursachen haben. In der Regel können Sie das Problem mit der folgenden einfachen Lösung beheben.

Um das Netzwerk Ihres Geräts zu aktivieren USB-over-Ethernet

1. Deaktivieren Sie WLAN auf Ihrem Computer und ziehen Sie alle damit verbundenen Ethernet-Kabel ab.
2. Drücken Sie die RESET-Taste am Fahrzeug, um das Gerät neu zu starten.
3. Öffnen Sie die Gerätekonsole, indem Sie auf Ihrem Computer in einem Webbrowser zu `https://deepracer.aws` navigieren.

Wenn das vorherige Verfahren nicht funktioniert, können Sie die Netzwerkeinstellungen Ihres Computers überprüfen, um sicherzustellen, dass sie so konfiguriert sind, dass der Computer eine Verbindung mit dem Netzwerk des Geräts mit dem Netzwerknamen `Deepracer` herstellen kann. Führen Sie dazu die Schritte im folgenden Verfahren aus.

> **Note**
>
> In den folgenden Anweisungen wird davon ausgegangen, dass Sie mit einem MacOS-Computer arbeiten. Informationen zu anderen Computersystemen finden Sie in der

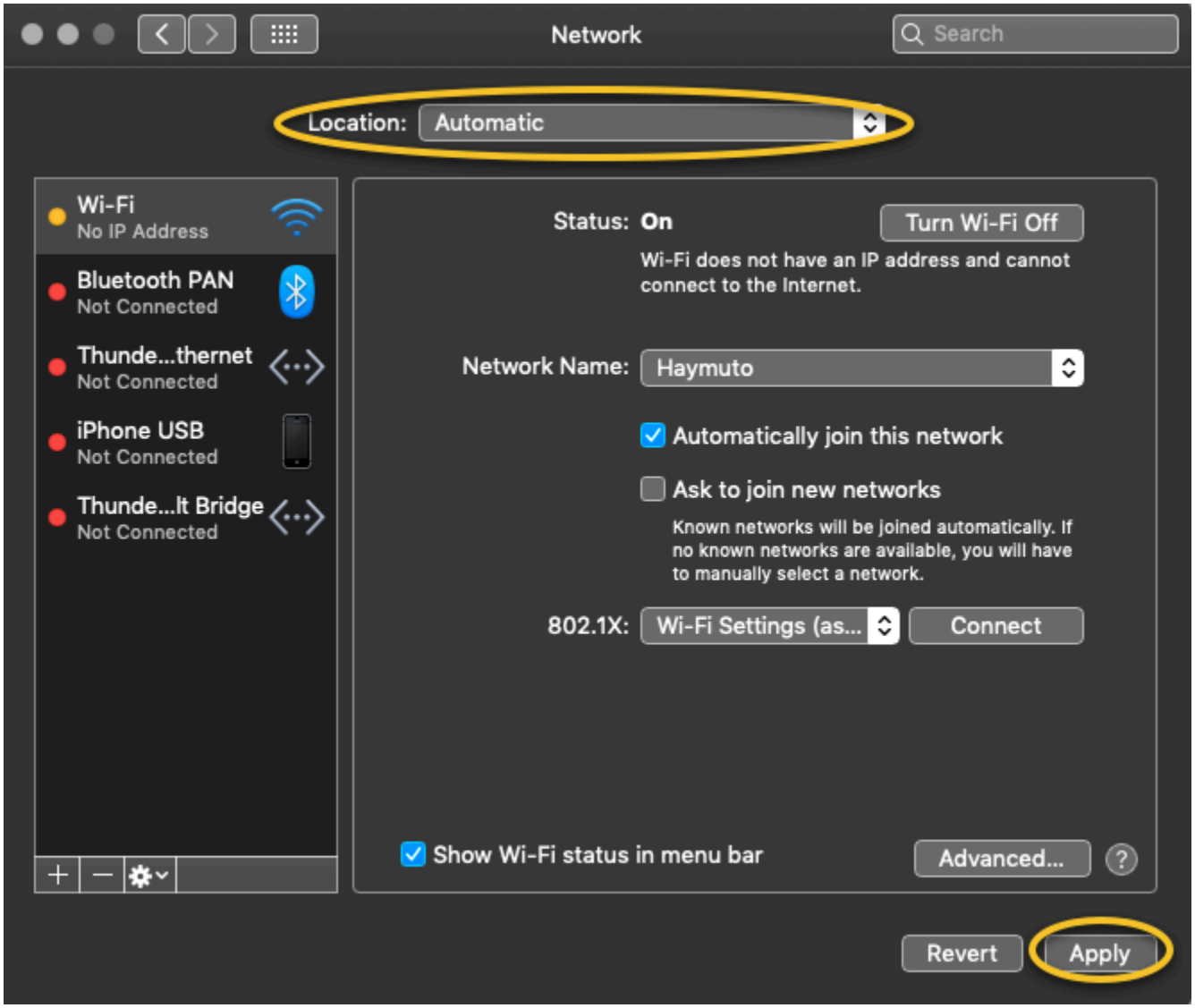
Dokumentation zu Netzwerkeinstellungen für das jeweilige Betriebssystem. Verwenden Sie die nachfolgenden Anweisungen als allgemeine Anleitung.

So aktivieren Sie das USB-over-ethernet Gerätenetzwerk auf Ihrem macOS-Computer

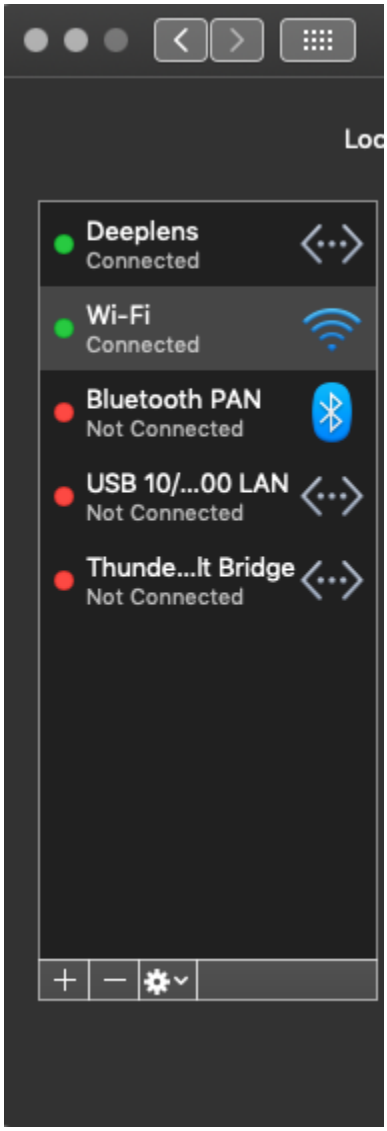
1. Wählen Sie das Netzwerksymbol (oben rechts in der Anzeige), um die Netzwerkeinstellungen zu öffnen.

   Alternativ können Sie cmd+Leertaste, wählen, Netzwerk eingeben und dann Netzwerksystemeinstellungen auswählen.

2. Prüfen Sie, ob Deepracer als Connected (Verbunden) aufgelistet ist. Wenn DeepRaceres aufgeführt, aber nicht angeschlossen ist, vergewissern Sie sich, dass das Micro-USB/USB-Kabel fest zwischen dem Fahrzeug und Ihrem Computer angeschlossen ist.
3. Wenn das Deepracer-Netzwerk dort nicht aufgeführt ist oder zwar aufgeführt, aber nicht angeschlossen ist, wenn das USB-Kabel angeschlossen ist, wählen Sie in der Standorteinstellung „Automatisch“ und dann „Anwenden“.



4. Stellen Sie sicher, dass das DeepRacer AWS-Netzwerk als Verbunden betriebsbereit ist und läuft.



5. Wenn Ihr Computer mit dem Deepracer-Netzwerk verbunden ist, aktualisieren Sie die Seite `https://deepracer.aws` im Browser und fahren mit den restlichen Anweisungen im Handbuch „Erste Schritte“ unter Connect to Wi-Fi (Mit dem WLAN verbinden) fort.
6. Wenn das Deepracer-Netzwerk nicht verbunden ist, trennen Sie Ihren Computer vom DeepRacer AWS-Fahrzeug und verbinden Sie ihn dann erneut. Wenn das Deepracer-Netzwerk in den Status Connected (Verbunden) wechselt, fahren Sie mit den Anweisungen im Handbuch „Erste Schritte“ fort.
7. Wenn das Deepracer-Netzwerk auf dem Gerät immer noch nicht verbunden ist, starten Sie Ihren Computer und Ihr DeepRacer AWS-Fahrzeug neu und wiederholen Sie gegebenenfalls Schritt 1 dieses Verfahrens.

Wenn die oben genannte Lösung das Problem immer noch nicht behebt, ist möglicherweise das Gerätezertifikat beschädigt. Gehen Sie wie folgt vor, um ein neues Zertifikat für Ihr DeepRacer AWS-Fahrzeug zu generieren, um die beschädigte Datei zu reparieren.

Um ein neues Zertifikat für das DeepRacer AWS-Fahrzeug zu generieren

1. Trennen Sie die USB-Verbindung zwischen Ihrem Computer und Ihrem DeepRacer AWS-Fahrzeug, indem Sie das Micro-USB/USB-Kabel abziehen.
2. Connect Sie Ihr DeepRacer AWS-Fahrzeug mit einem Monitor (mit einem HDMI-to-HDMI Kabel) und mit einer USB-Tastatur und -Maus.
3. Melden Sie sich beim DeepRacer AWS-Betriebssystem an. Wenn dies die erste Anmeldung beim Betriebssystem des Geräts ist, verwenden Sie `deepracer` als Passwort, wenn Sie zur Eingabe aufgefordert werden. Ändern Sie dann das Passwort bei Bedarf und verwenden Sie das aktualisierte Passwort für die nachfolgenden Anmeldungen.
4. Öffnen Sie ein Terminalfenster, und geben Sie den folgenden Shell-Befehl ein. Sie können die Verknüpfung Terminal unter Anwendungen – > System-Tools auf dem Desktop auswählen, um ein Terminalfenster zu öffnen. Oder Sie verwenden den Dateibrowser, navigieren zum Ordner `/usr/bin` und wählen gnome-terminal aus, um es zu öffnen.

   ```
   sudo /opt/aws/deepracer/nginx/nginx_install_certs.sh && sudo reboot
   ```

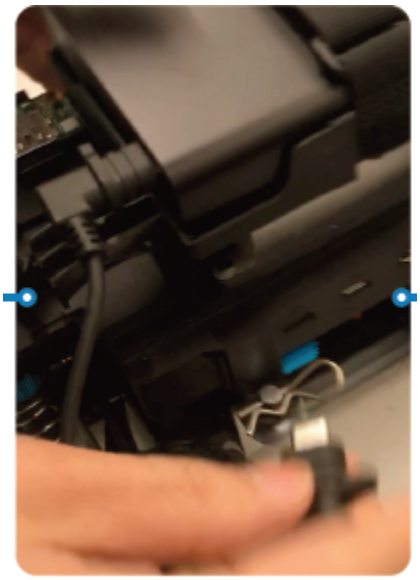
   Geben Sie das Passwort ein, das Sie im vorherigen Schritt verwendet oder aktualisiert haben, wenn Sie dazu aufgefordert werden.

   Mit dem Befehl oben wird ein neues Zertifikat installiert und das Gerät neu gestartet. Außerdem wird das Passwort der Gerätekonsole auf den Standardwert zurückgesetzt, der auf der Unterseite des DeepRacer AWS-Fahrzeugs aufgedruckt ist.
5. Trennen Sie den Monitor, die Tastatur und die Maus vom Fahrzeug und verbinden Sie ihn mit dem (Micro-)USB-Kabel wieder mit Ihrem Computer.
6. Führen Sie das zweite Verfahren in diesem Thema durch, um zu überprüfen, ob Ihr Computer tatsächlich mit dem Gerätenetzwerk verbunden ist, bevor Sie die Gerätekonsole (`https://deepracer.aws`) erneut öffnen. Fahren Sie dann mit den Anweisungen Herstellen einer Verbindung mit dem WLAN-Netzwerk im Handbuch zu den ersten Schritten fort.

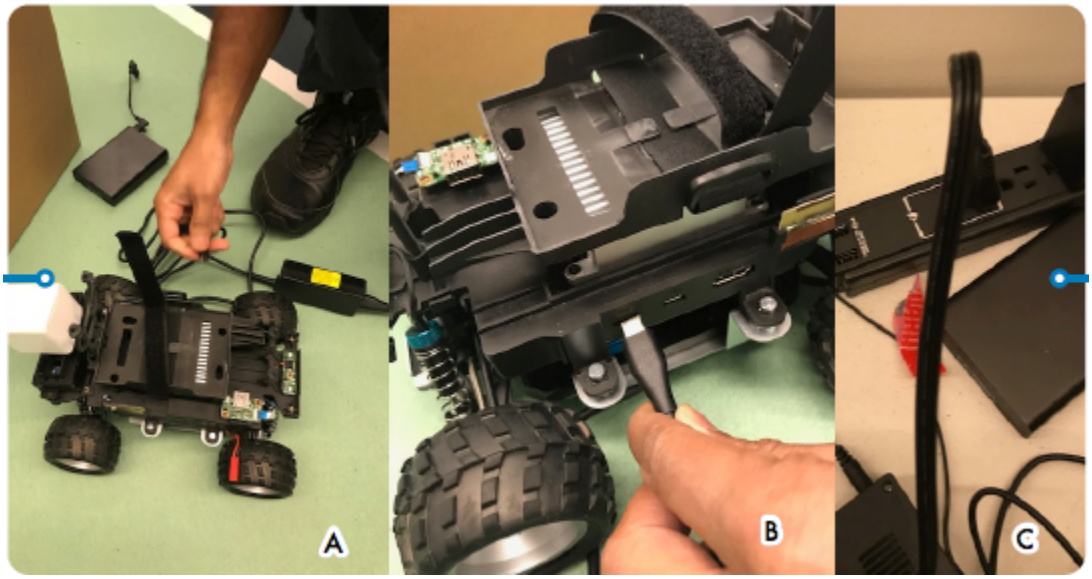
So schalten Sie die Stromquelle des DeepRacer AWS-Rechenmoduls von der Batterie auf eine Steckdose um

Wenn der Akkuladestand des Rechenmoduls bei der ersten Einrichtung Ihres AWS DeepRacer niedrig ist, gehen Sie wie folgt vor, um die Stromversorgung des Rechenmoduls von der Batterie auf eine Steckdose umzuschalten:

1. Ziehen Sie das USB-C-Kabel aus dem Rechenmodul-Netzanschluss des Fahrzeugs.



2. Schließen Sie das Netzkabel und das USB-C-Kabel an das Netzteil (A) des Rechenmoduls an. Schließen Sie das Netzkabel an eine Steckdose (C) an und schließen Sie das USB-C-Kabel an den Stromanschluss (B) des Rechenmoduls des Fahrzeugs an.



So verwenden Sie ein USB-Flash-Laufwerk, um AWS mit Ihrem Wi-Fi-Netzwerk DeepRacer zu verbinden

Um ein DeepRacer AWS-Fahrzeug über ein USB-Flash-Laufwerk mit Ihrem Wi-Fi-Netzwerk zu Hause oder im Büro zu verbinden, benötigen Sie Folgendes:

- Ein USB-Flash-Laufwerk
- Der Name (SSID) und das Passwort für das Wi-Fi-Netzwerk, dem Sie beitreten möchten

> **Note**
>
> AWS unterstützt DeepRacer keine Wi-Fi-Netzwerke, für die eine aktive Captcha-Überprüfung für die Benutzeranmeldung erforderlich ist.

So verbinden Sie ein DeepRacer AWS-Fahrzeug über ein USB-Flash-Laufwerk mit einem Wi-Fi-Netzwerk

1. Schließen Sie das USB-Flash-Laufwerk an Ihren Computer an.

2. Öffnen Sie einen Webbrowser auf Ihrem Computer und navigieren Sie zu https://aws.amazon.com/deepracer/usbwifi. Dieser Link öffnet eine Textdatei mit dem Namen `wifi-creds.txt` Hosted on GitHub.

```
40 lines (39 sloc)  3.25 KB
```

```
1  ##############################################################################
2  #                    Connect the AWS DeepRacer vehicle to Wi-Fi               #
3  # File name: wifi-creds.txt                                                   #
4  #                                                                             #
5  # To connect the AWS DeepRacer vehicle to Wi-Fi, type your Wi-Fi name (SSID) and  #
6  # password in the appropriate field at the end of this file. Both values are case #
7  # sensitive.                                                                  #
8  #                                                                             #
9  # For example:                                                                #
10 #      ssid: 'Your-WiFi 100'                                                  #
11 #      password: 'Passwd1234'                                                 #
12 #                                                                             #
```

3. Speichern Sie `wifi-creds.txt` auf Ihrem USB-Flash-Laufwerk. Abhängig vom verwendeten Webbrowser wird die Textdatei möglicherweise auf Ihren Computer heruntergeladen und automatisch in Ihrem Standard-Editor für Code geöffnet. Falls der Download `wifi-creds.txt` nicht automatisch erfolgt, öffnen Sie das Kontextmenü (Rechtsklick) und wählen Sie Speichern unter, um die Textdatei auf Ihrem USB-Flash-Laufwerk zu speichern.

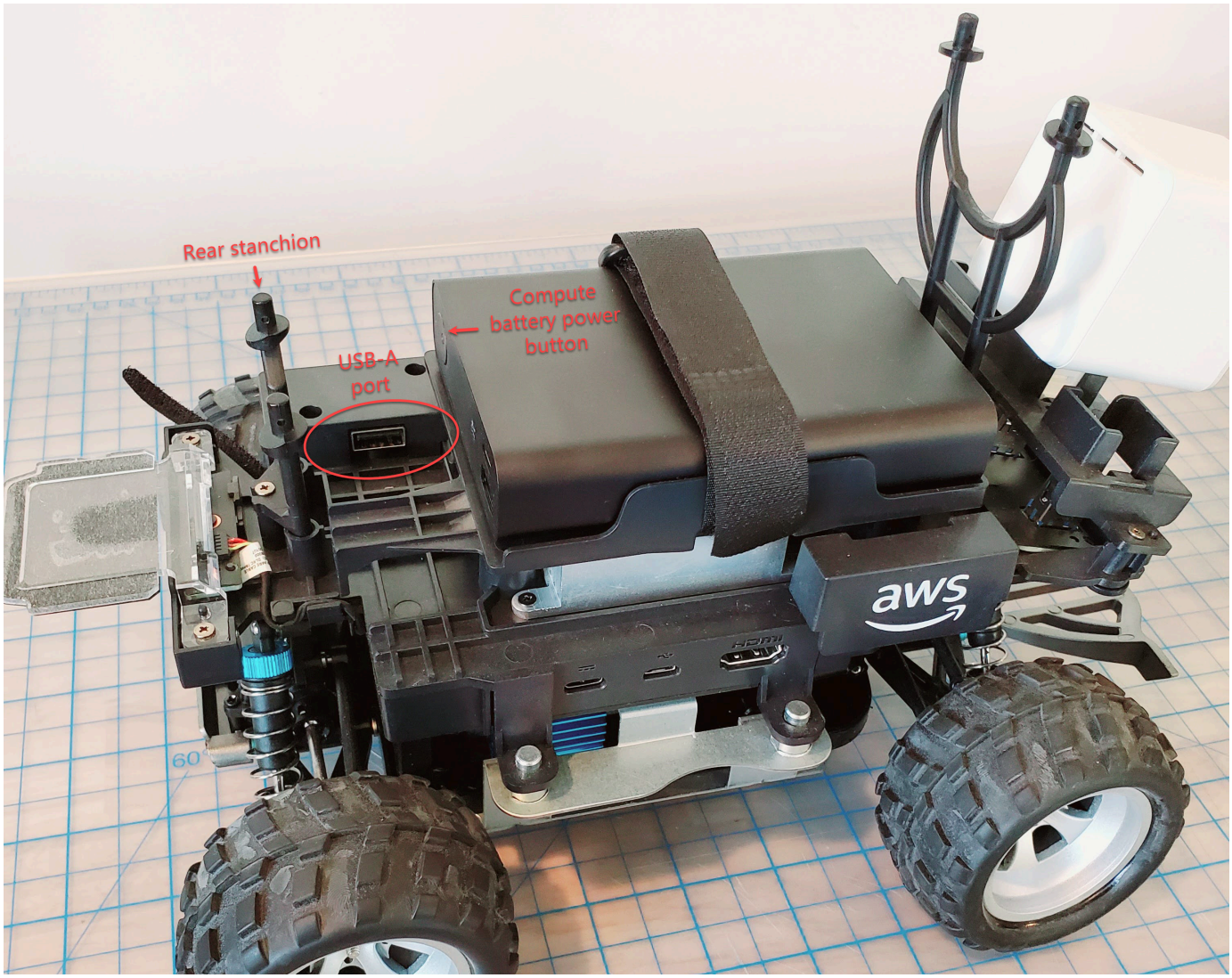
> **Warning**
>
> Ändern Sie nicht den Dateinamen.

4. Wenn es noch `wifi-creds.txt` nicht geöffnet ist, öffnen Sie es in einem Code-Editor im Nur-Text-Modus. Einige Texteditoren verwenden standardmäßig Rich-Text (.rtf) statt Nur-Text (.txt), wenn der Dateityp nicht angegeben ist. Wenn Sie also Probleme bei der Bearbeitung der Datei haben, überprüfen Sie Ihre Einstellungen. Wenn Sie Windows verwenden, können Sie auch versuchen, die Datei mit der Sublime Text-Anwendung zu öffnen, die Sie kostenlos herunterladen können, oder, wenn Sie einen Mac verwenden, die TextEdit Anwendung ausprobieren, die auf den meisten Mac-Geräten vorinstalliert ist und standardmäßig Klartext verwendet.
5. Geben Sie zwischen den einfachen Anführungszeichen am Ende der Datei den Namen (SSID) und das Passwort des Wi-Fi-Netzwerks ein, das Sie verwenden möchten. SSID steht für „Service Set Identifier“. Dies ist die technische Bezeichnung des Namens Ihres WLAN-Netzwerks.

> **Note**
>
> Wenn der Netzwerkname (SSID) oder das Passwort ein Leerzeichen enthalten, z. B. Your-Wi-Fi 100, geben Sie den Namen genau, einschließlich des Leerzeichens, innerhalb der Anführungszeichen („) ein. Wenn kein Leerzeichen vorhanden ist, ist die Verwendung von Anführungszeichen optional. Beispielsweise enthält das Wi-Fi-Passwort Passwd1234 kein Leerzeichen, sodass die Verwendung einfacher Anführungszeichen funktioniert, aber nicht erforderlich ist. Sowohl SSID als auch das Passwort unterscheiden zwischen Groß- und Kleinschreibung.

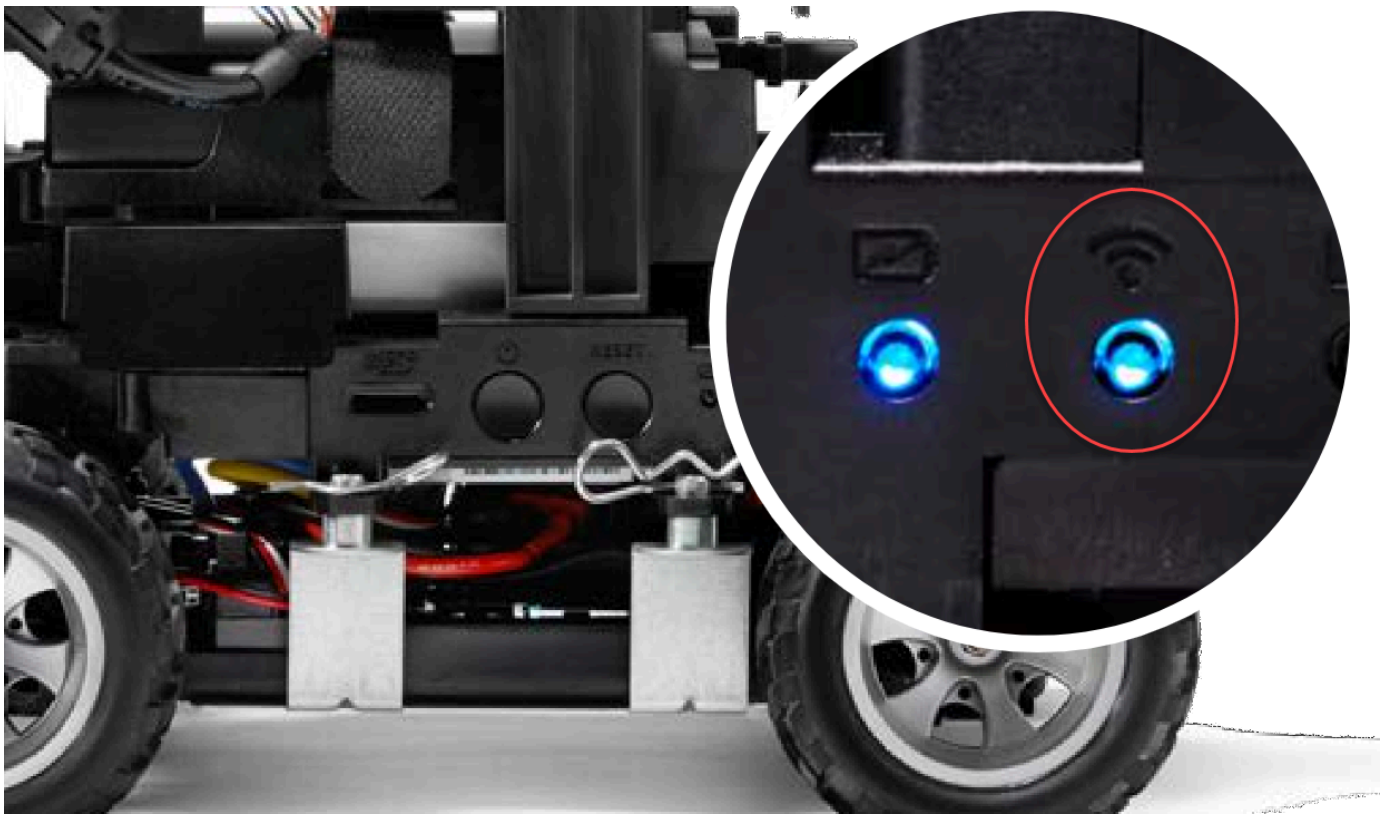
```
29  # If you have validated the Wi-Fi credentials but the Wi-Fi LED doesn't          #
30  # turn solid blue, try restarting the vehicle by pressing the reset button.      #
31  # When the power LED turns blue, plug the USB drive in again.                    #
32  #                                                                                #
33  # To finish setting up, follow the instructions on https://docs.aws.amazon.com/  #
34  # deepracer/latest/developerguide/deepracer-troubleshooting-wifi-connection-first #
35  # -time.                                                                         #
36  ##################################################################################
37
38  # Enter your Wi-Fi name (SSID) and password:
39  ssid: ''
40  password: ''
```

6. Speichern Sie die Datei auf Ihrem USB-Flash-Laufwerk.
7. Werfen Sie das USB-Laufwerk aus Ihrem Computer und stecken Sie es in den USB-A-Anschluss auf der Rückseite des DeepRacer AWS-Fahrzeugs zwischen dem Netzschalter für die Rechenbatterie und der hinteren Stütze.



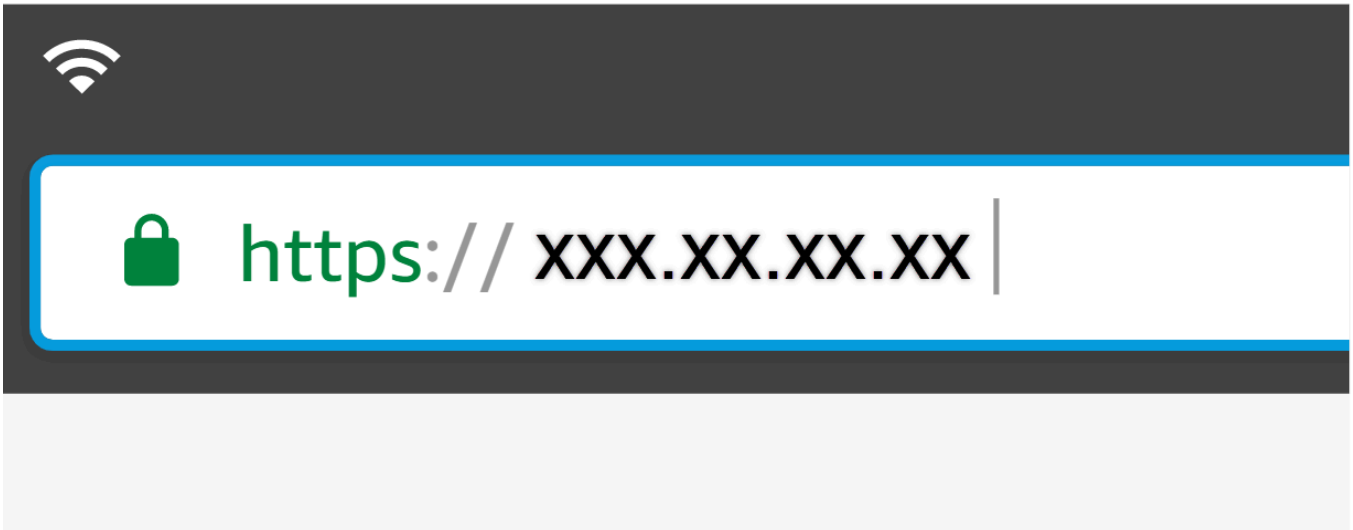
8. Stellen Sie sicher, dass AWS eingeschaltet DeepRacer ist.
9. Beobachten Sie die WLAN-LED am Fahrzeug. Wenn es blinkt und dann von weiß zu blau wechselt, ist das Fahrzeug mit dem Wi-Fi-Netzwerk verbunden. Trennen Sie das USB-Laufwerk und fahren Sie mit Schritt 11 fort.

> **Note**
>
> Wenn das USB-Laufwerk an das Fahrzeug angeschlossen war, bevor Sie versucht haben, das Fahrzeug mit einem Wi-Fi-Netzwerk zu verbinden, wird automatisch eine Liste der verfügbaren Wi-Fi-Netzwerke in einer `wifi-creds.txt` Datei auf Ihrem Flash-Laufwerk angezeigt. Entfernen Sie das Kommentarzeichen für das Netzwerk, zu dem Sie eine Verbindung herstellen möchten, indem Sie das Rautezeichen entfernen.



10. Wenn die Wi-Fi-LED nach dem Blinken rot leuchtet, trennen Sie das USB-Laufwerk vom Fahrzeug und schließen Sie es wieder an Ihren Computer an. Überprüfen Sie den Wi-Fi-Namen und das Passwort, die Sie in die Textdatei eingegeben haben, auf Tippfehler, falsche Leerzeichen, falsche Groß- und Kleinschreibung oder fehlende oder falsch verwendete einfache Anführungszeichen. Korrigieren Sie Fehler, speichern Sie die Datei erneut und wiederholen Sie die Schritte 7-9.
11. Wenn die WLAN-LED des Fahrzeugs blau leuchtet, ziehen Sie das USB-Laufwerk vom Fahrzeug ab und schließen es an Ihren Computer an.
12. Öffnen Sie die `wifi-creds.txt` Datei. Suchen Sie am Ende der Textdatei die IP-Adresse Ihres Fahrzeugs und kopieren Sie sie.
13. Vergewissern Sie sich, dass sich Ihr Computer im selben Netzwerk wie das Fahrzeug befindet, und fügen Sie dann die IP-Adresse in Ihren Webbrowser ein.

> **Note**
>
> Wenn Sie macOS Catalina verwenden, verwenden Sie den Firefox-Webbrowser. Chrome wird nicht unterstützt.



14. Wenn Ihnen die Meldung angezeigt wird, dass die Verbindung nicht privat oder sicher ist, akzeptieren Sie die Sicherheitswarnung und fahren zur Host-Seite fort.

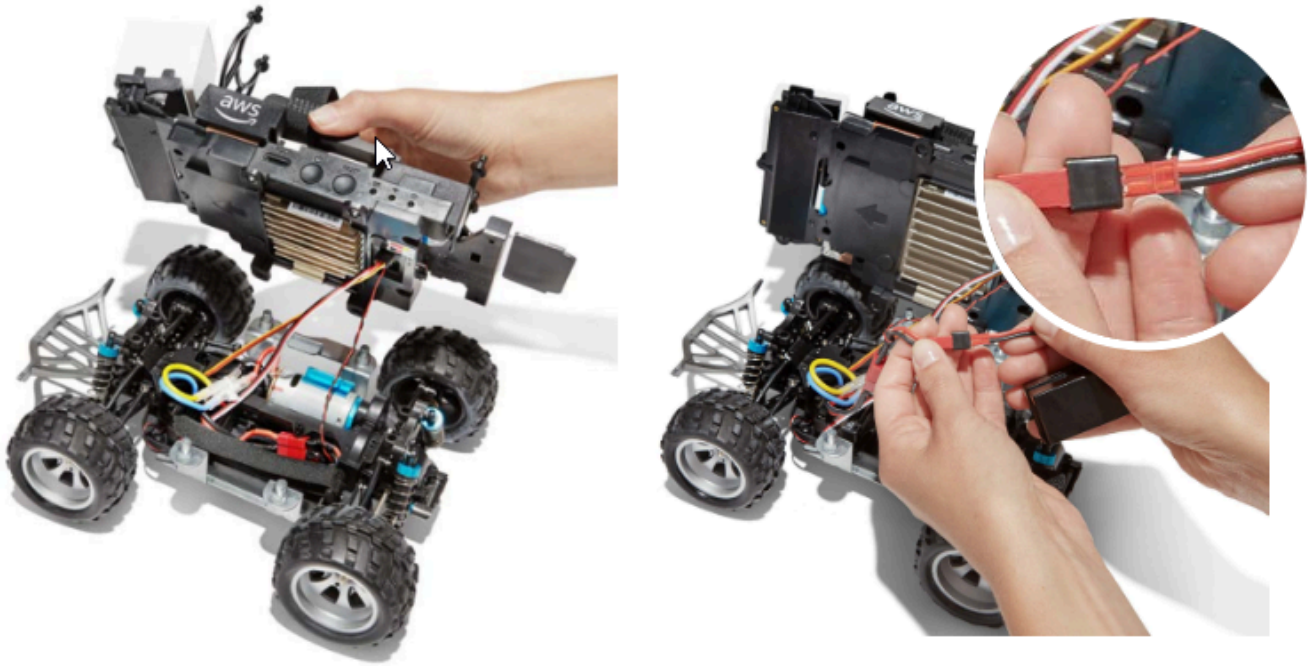
Ihr AWS DeepRacer ist jetzt mit WLAN verbunden.

So laden Sie den Akku des DeepRacer AWS-Laufwerksmoduls auf

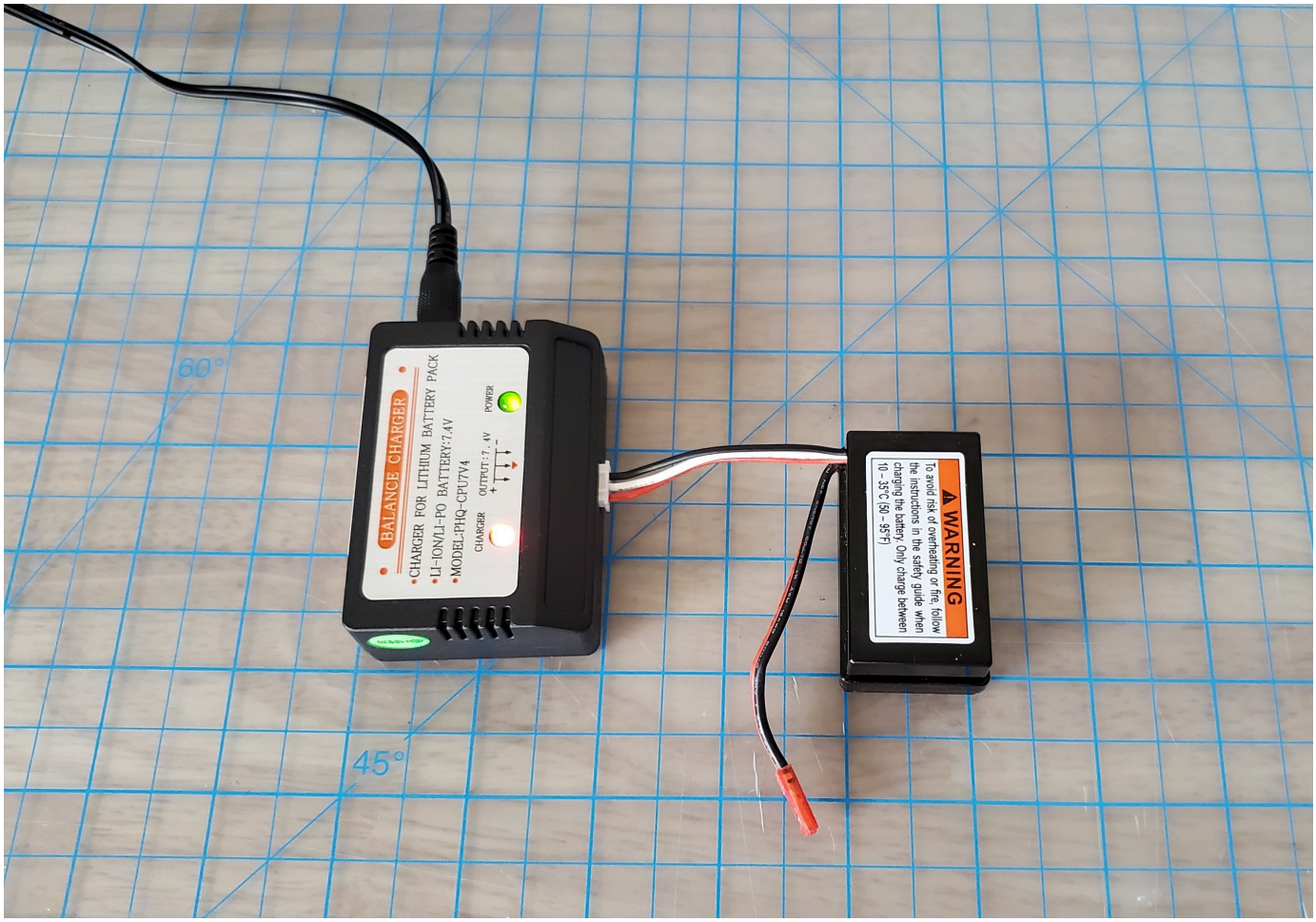
Die Batterie des DeepRacer AWS-Antriebsmoduls besteht aus zwei Kabelsätzen mit zwei unterschiedlich farbigen JST-Steckern, weiß und rot. Der weiße 3-polige Stecker am Ende der schwarzen, roten und weißen Kabel verbindet die Batterie des Fahrzeugmoduls mit dem Batterieladegerät. Der rote 2-polige Stecker am Ende der schwarzen und roten Kabel verbindet die Batterie mit dem Fahrzeugantriebsstrang.

Gehen Sie wie folgt vor, um den Akku Ihres DeepRacer AWS-Drive-Moduls aufzuladen:

1. Um auf die Batterie des Antriebsmoduls zuzugreifen, sofern diese an das Fahrzeug angeschlossen ist, heben Sie das Rechenmodul an. Achten Sie dabei darauf, dass sich die Kabel, die es mit dem Antriebsstrang verbinden, nicht lockern.



2. Um optional die Batterie des Antriebsmoduls aus dem Fahrzeug zu entfernen, trennen Sie den roten 2-poligen Batteriestecker vom schwarz-roten Antriebsstranganschluss und lösen Sie den Klettverschluss.
3. Verbinden Sie die Batterie mit dem Batterieladegerät, indem Sie den weißen 3-poligen Stecker der Batterie mit dem Ladeanschluss verbinden.



Rotes Licht + grünes Licht = nicht vollständig aufgeladen

4. Stecken Sie das Netzkabel des Akkuladegeräts in eine Steckdose. Wenn nur das grüne Licht leuchtet, ist der Akku vollständig geladen.
5. Trennen Sie den weißen 3-poligen Stecker der aufgeladenen Fahrzeugbatterie vom Ladeadapter. Wenn Sie die Batterie herausgenommen haben, um sie aufzuladen (optional), achten Sie darauf, den roten 2-poligen Stecker wieder mit dem Fahrzeugantriebsstrang zu verbinden und die Batterie mit dem Klettband am Fahrzeug zu befestigen.
6. Schalten Sie den Fahrzeugantrieb ein, indem Sie seinen Schalter in die Stellung „Ein“ bringen. Achten Sie darauf, dass die Anzeige ertönt (zwei kurze Signaltöne), um den erfolgreichen Ladevorgang zu bestätigen. Wenn Sie keine zwei Signaltöne hören, versuchen Sie, die Fahrzeugbatterie zu entsperren

Die Batterie Ihres DeepRacer AWS-Antriebsmoduls ist jetzt einsatzbereit.

So laden Sie den Akku des DeepRacer AWS-Rechenmoduls auf

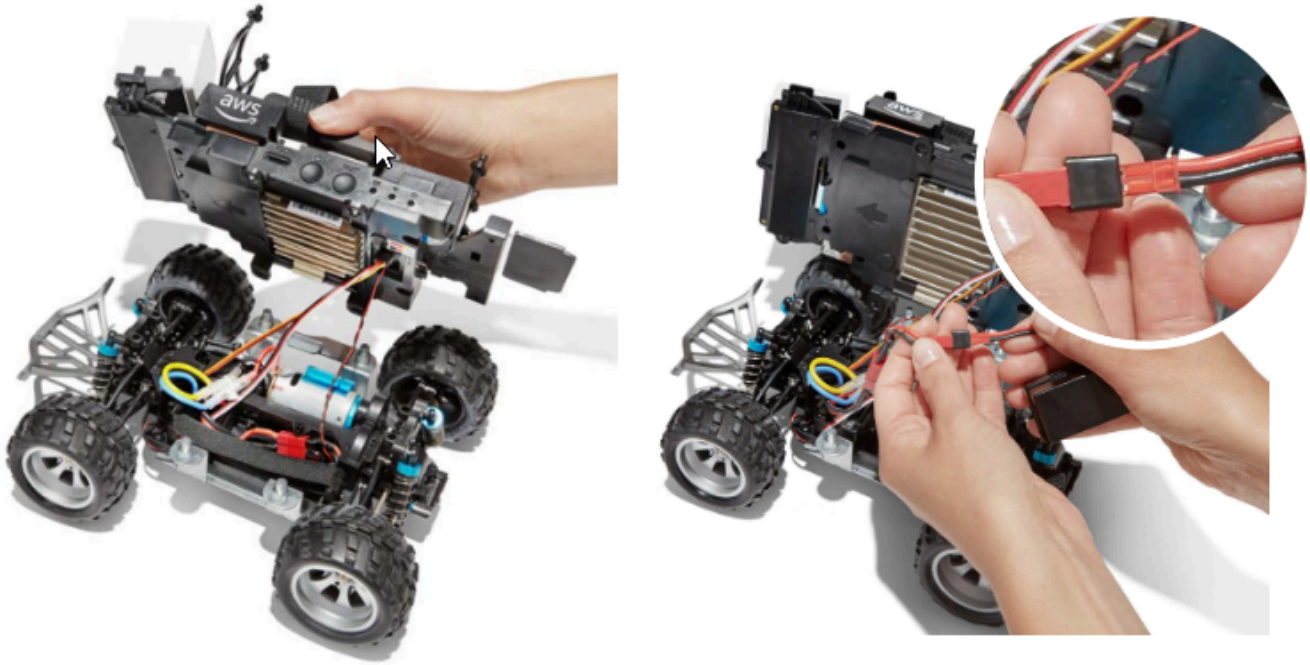
Gehen Sie wie folgt vor, um den Akku Ihres DeepRacer AWS-Rechenmoduls aufzuladen:

1. Entfernen Sie optional den Akku des Rechenmoduls aus dem Fahrzeug.
2. Schließen Sie das Ladegerät des Rechenmoduls an den Akku des Rechenmoduls an.
3. Stecken Sie das Netzkabel des Ladegeräts des Rechenmoduls in eine Steckdose.

Mein Akku ist geladen, aber mein DeepRacer AWS-Fahrzeug bewegt sich nicht

Gehen Sie wie folgt vor, wenn Ihre DeepRacer AWS-Konsole eingerichtet, Ihre Rechenbatterie geladen und Ihr WLAN verbunden ist, Ihr Fahrzeug sich aber immer noch nicht bewegt:

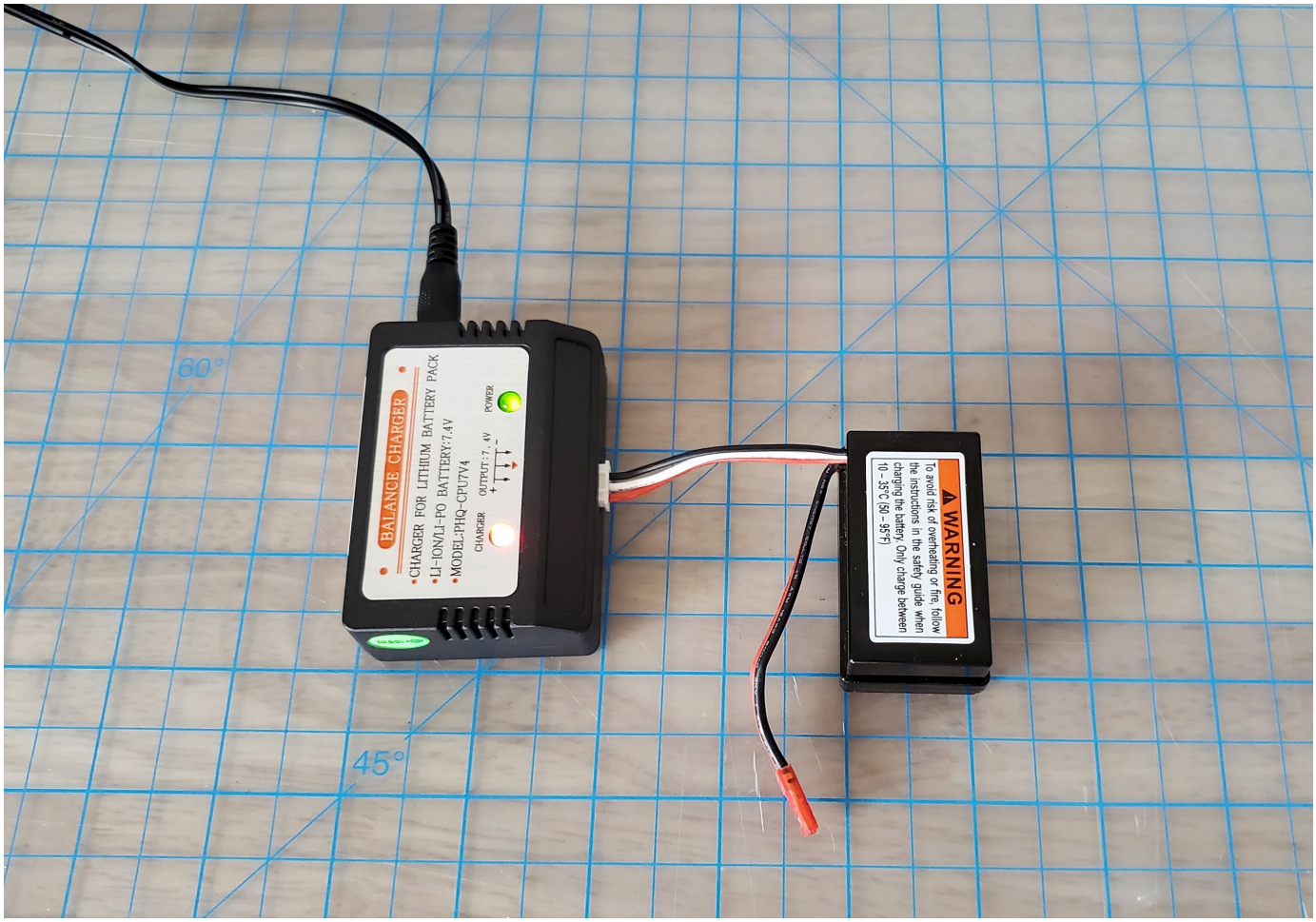
1. Heben Sie das Datenverarbeitungsmodul an und achten Sie darauf, dass sich die Kabel, die es mit dem Antrieb verbinden, nicht lösen. Vergewissern Sie sich, dass die Fahrzeugbatterie auf der Unterseite richtig angeschlossen ist, roter 2-poliger Stecker an schwarzen und roten Antriebsstranganschluss.



2. Schalten Sie den Fahrzeugantrieb ein, indem Sie den Schalter in die Stellung „Ein“ bringen. Achten Sie darauf, dass die Anzeige ertönt (zwei kurze Signaltöne), um zu bestätigen, dass das

Fahrzeug geladen ist. Wenn das Fahrzeug erfolgreich eingeschaltet wird, fahren Sie mit Schritt 4 fort.

3. Wenn Sie beim Einschalten der Fahrzeugbatterie keine zwei Signaltöne hören, stellen Sie sicher, dass die Batterie vollständig geladen ist. Stecken Sie das weiße Anschlusskabel des Fahrzeugakkus in den Ladeadapter, der durch die roten und grünen LED-Kontrollleuchten vom Adapter des Rechenmoduls unterschieden werden kann. Schließen Sie den Adapter an sein Ladekabel an und stecken Sie ihn in eine Steckdose. Wenn die rote und die grüne Lampe am Fahrzeugladeadapter leuchten, zeigt dies an, dass der Akku noch weiter aufgeladen werden muss.



Rotes Licht + grünes Licht = nicht vollständig aufgeladen

Wenn nur die grüne Lampe leuchtet, ist Ihr Akku vollständig geladen und einsatzbereit. Trennen Sie den weißen Stecker der Autobatterie vom Ladeadapter und schließen Sie den roten Stecker wieder an das Fahrzeug an. Wenn Sie den Akku zum Aufladen (optional) entfernt haben, müssen Sie ihn wieder mit dem Klettband am Antriebsstrang befestigen. Schalten Sie den Fahrzeugantrieb ein, indem Sie seinen Schalter in die Stellung „Ein“ bringen. Wenn Sie immer noch keine zwei Signaltöne hören, versuchen Sie, die Fahrzeugbatterie zu entsperren.

4. Connect Sie Ihr Fahrzeug mit WLAN und öffnen Sie die DeepRacer AWS-Konsole in Ihrem Browser. Fahren Sie Ihr Fahrzeug manuell mit dem Touch-Joystick, um festzustellen, ob es sich bewegen kann.

ERINNERUNG: Um das Meiste aus Ihrer Fahrzeugbatterie herauszuholen, stellen Sie sicher, dass Sie den Fahrzeugantrieb ausschalten oder die Batterie abklemmen, wenn Sie Ihr AWS DeepRacer nicht verwenden.

Wenn sich Ihr Fahrzeug immer noch nicht bewegt, wenden Sie sich an AWSDeepRacer-Help @amazon .com.

Fehlerbehebung bei der Sperrung der DeepRacer AWS-Fahrzeugbatterie

> **Important**
>
> Diese Batterie ist nur für die Verwendung mit dem DeepRacer Auto bestimmt. Diese Batterie muss ordnungsgemäß gehandhabt werden, um Brandgefahr, Explosion oder andere Sicherheitsaspekte zu vermeiden. Folgen Sie allen Anweisungen und beachten Sie alle Warnungen im DeepRacer AWS-Gerätesicherheitsleitfaden.

Nutzungsbedingungen, Garantien und Hinweise für DeepRacer AWS-Geräte

- Nutzungsbedingungen für DeepRacer AWS-Geräte
- Einjährige beschränkte Garantie für AWS-Geräte DeepRacer
- AWS-Leitfaden zur DeepRacer Gerätesicherheit

Themen

- So verhindern Sie die Sperrung der DeepRacer AWS-Fahrzeugbatterie
- So entsperren Sie eine DeepRacer AWS-Fahrzeugbatterie nach der Sperrung

So verhindern Sie die Sperrung der DeepRacer AWS-Fahrzeugbatterie

Erfahren Sie, wie Sie eine Sperrung der DeepRacer AWS-Fahrzeugbatterie verhindern können.

Um den Batteriezustand zu erhalten, wechselt die DeepRacer AWS-Fahrzeugbatterie in den Sperrzustand. In diesem Fall versorgt die Batterie Ihr Fahrzeug nicht mit Strom, auch wenn es noch teilweise geladen ist. Um zu verhindern, dass Ihre Autobatterie in den Aussperrzustand wechselt, gehen Sie wie folgt vor:

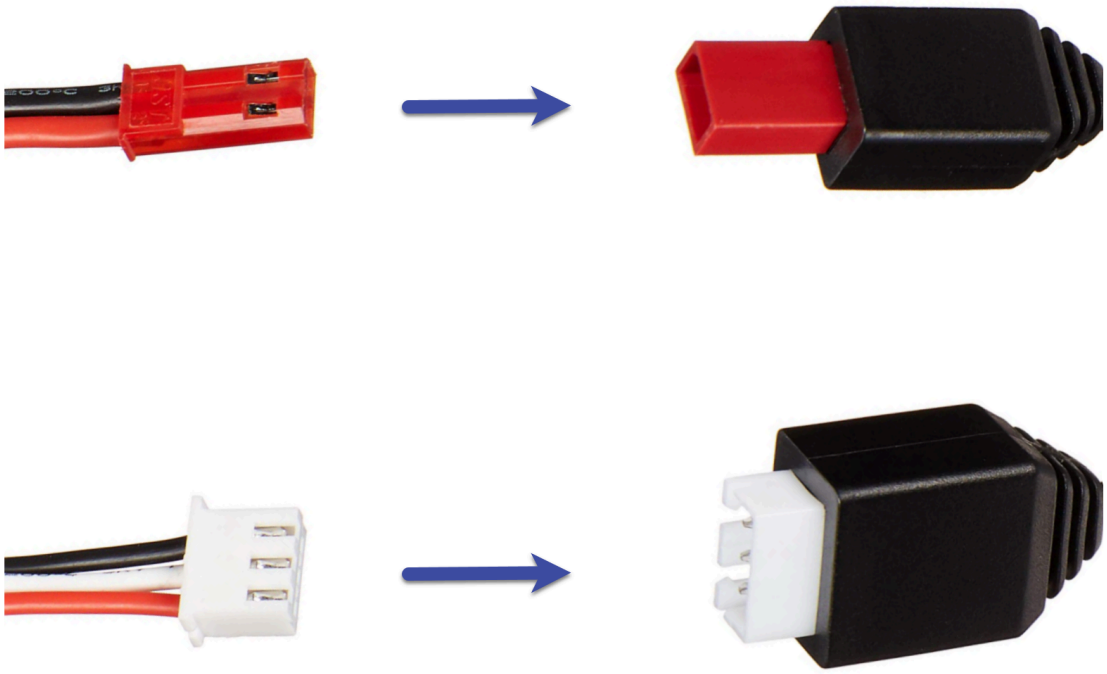
- Wenn Sie Ihr AWS nicht mehr verwenden DeepRacer, schalten Sie das Fahrzeug aus, um die Ladung der Batterie aufrechtzuerhalten.
- Wenn die Gerätekonsole Sie darüber informiert, dass der Stand Ihrer Fahrzeugbatterie niedrig ist, laden Sie sie so schnell wie möglich auf.
- Wenn Sie denken, dass Sie AWS DeepRacer für eine Weile nicht verwenden werden, trennen Sie die Batterie vom Fahrzeug und laden Sie sie vollständig auf. Wir empfehlen Ihnen, Ihre Fahrzeugbatterie mindestens einmal im Jahr aufzuladen, um sie zu schützen und eine Aussperrung zu verhindern.

> **Note**
>
> Alle Lithium-Polymer-Batterien (LiPo) entladen sich im Laufe der Zeit langsam, auch wenn sie nicht verwendet werden.

So entsperren Sie eine DeepRacer AWS-Fahrzeugbatterie nach der Sperrung

Verwenden Sie das Entsperrkabel, um Ihre DeepRacer AWS-Batterie nach dem Sperren zu entsperren:

1. Stecken Sie die Batteriestecker in die passenden farbigen Kabelstecker (rot bis rot und weiß bis weiß).



2. Trennen Sie die Batterie vom Kabel.

3. Ihre DeepRacer AWS-Fahrzeugbatterie ist sofort einsatzbereit. Schließen Sie den roten 2-poligen Stecker wieder an den Stecker des Fahrzeugantriebsstrangs an und befestigen Sie die Batterie mit dem Klettband am Fahrzeug.
4. Schalten Sie den Fahrzeugantrieb ein, indem Sie seinen Schalter in die Stellung „Ein“ bringen. Achten Sie darauf, dass die Anzeige ertönt (zwei kurze Signaltöne), um zu bestätigen, dass die Batterie erfolgreich entsperrt wurde.

So wickeln Sie ein Dell-Akkuanschlusskabel bei der Installation eines LiDAR-Sensors ein

Die Montage der Evo-Schale über einem LiDAR-Sensor, der mit einem DeepRacer AWS-Fahrzeug verbunden ist, mithilfe des extra langen USB-C-auf-abgewinkelten USB-C-Anschlusskabels von Dell erfordert eine spezielle Kabelumwickelungstechnik.

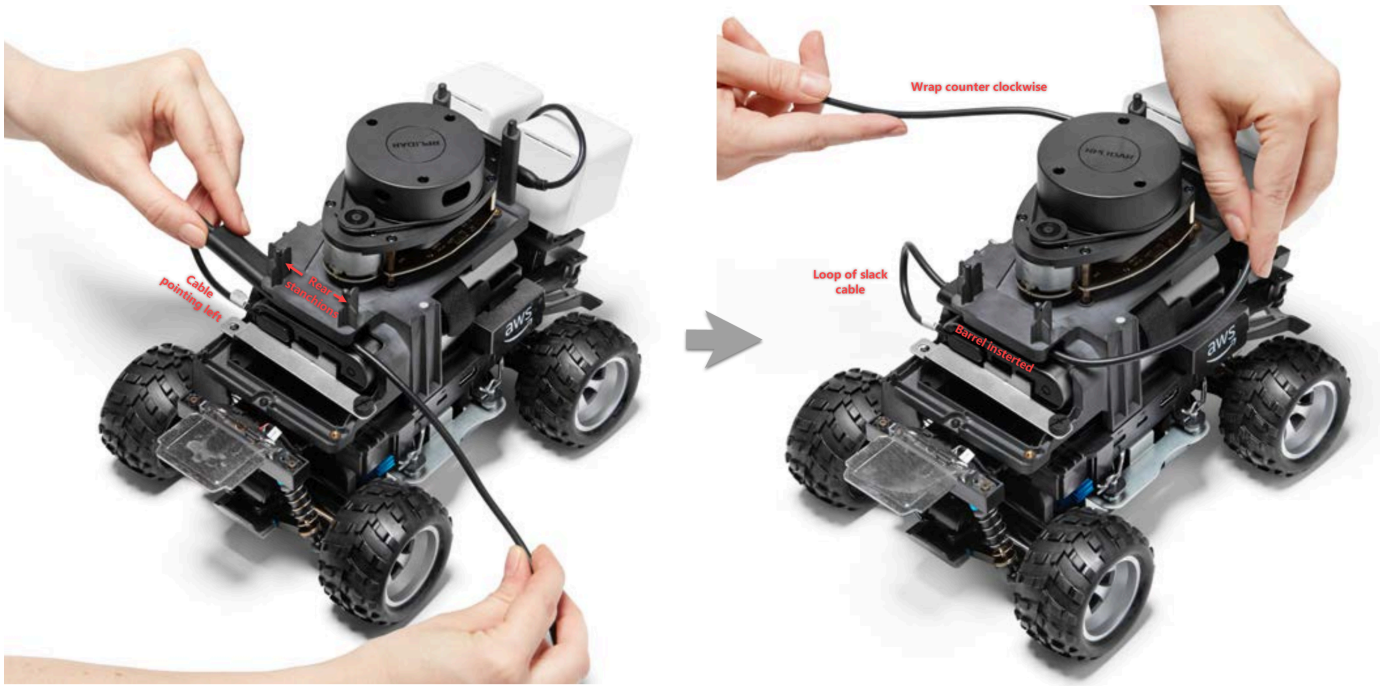
Ein Video zu diesem Vorgang finden Sie unter AWS DeepRacer: Installieren Sie den LiDAR-Sensor und wickeln Sie das Kabel des Dell Compute-Batterieanschlusses ein. YouTube Das Video beginnt mit der Installation des LiDAR-Sensors auf dem DeepRacer AWS-Fahrzeug. Die Technik zum Einwickeln von Akkus bei Dell beginnt um 00:01:27 Sekunden.

> **Note**
>
> Das Akku-Anschlusskabel von Dell hat ein Gehäuse, ein Standard-USB-C-Ende und ein abgewinkeltes USB-C-Ende.

Um ein Dell-Batteriekabel um einen LiDAR-Sensor zu wickeln, um das Evo-Gehäuse aufzunehmen

1. Stecken Sie das abgewinkelte Ende des Compute Battery-Anschlusskabels mit Blick auf die Rückseite des DeepRacer AWS-Fahrzeugs in den USB-C-Anschluss für die Compute Battery, sodass das Verbindungskabel nach links zeigt.

2. Wenn Sie das Fahrzeug leicht nach links drehen, finden Sie die Öffnung zum Raum zwischen dem LiDAR-Halter und der Rechenbatterie direkt unter den hinteren Stützen und führen Sie das Kabel durch. Hören Sie auf, das Kabel durchzuziehen, wenn der Zylinder in diesen Raum eingeführt ist. Links vom USB-C-Anschluss sollte sich eine Kabelschlaufe befinden.



3. Wickeln Sie das Kabel mit Blick auf die Rückseite des DeepRacer AWS-Fahrzeugs gegen den Uhrzeigersinn um die Basis des LiDAR-Sensors und befestigen Sie das Kabel mit den Kabelklemmen an sich selbst, um einen festen Sitz zu gewährleisten.
4. Drehen Sie das Fahrzeug leicht nach rechts und stecken Sie das Standard-USB-C-Ende des Kabels in den USB-C-Anschluss.

5. Platzieren Sie die Evo-Schale auf Ihrem DeepRacer AWS-Fahrzeug und befestigen Sie sie mit Stecknadeln, um die Passform zu testen. Wenn die Schale richtig sitzt, ist der LiDAR-Sensor durch die Aussparung in der Schale vollständig sichtbar, und Sie haben Zugriff auf die Nadellöcher an der Oberseite der Rungen. Entfernen Sie das Gehäuse und passen Sie das Kabel nach Bedarf an.

Ihr LiDAR-Sensor ist angeschlossen. Sie sind bereit, Ihr Fahrzeug einzuschalten, zu fahren und zu experimentieren.

So halten Sie die Wi-Fi-Verbindung Ihres Fahrzeugs aufrecht

Die folgende Anleitung zur Fehlerbehebung gibt Ihnen Tipps zur Aufrechterhaltung der Verbindung Ihres Fahrzeugs.

So beheben Sie Probleme mit der Wi-Fi-Verbindung, wenn die Wi-Fi-LED-Anzeige Ihres Fahrzeugs blau blinkt, dann zwei Sekunden lang rot leuchtet und schließlich erlischt

Überprüfen Sie die folgenden Punkte, um sicherzustellen, dass Sie über die gültigen Einstellungen für die WLAN-Verbindung verfügen.

- Vergewissern Sie sich, dass das USB-Laufwerk nur eine Festplattenpartition hat, auf der sich nur eine Datei namens wifi-creds.txt befindet. Wenn mehrere Dateien namens wifi-creds.txt gefunden werden, werden alle in der Reihenfolge verarbeitet, in der sie gefunden wurden, was zu unvorhersehbarem Verhalten führen kann.
- Vergewissern Sie sich, dass die SSID und das Passwort des WLAN-Netzwerks in der Datei wifi-creds.txt korrekt angegeben sind. Ein Beispiel für diese Datei ist wie folgt dargestellt:

```
##################################################################################
#                                  AWS DeepRacer                                 #
# File name: wifi-creds.txt                                                      #
#                                                                                #
# ...                                                                            #
##################################################################################

# Provide your SSID and password below
ssid: ' MyHomeWi-Fi'
password: myWiFiPassword
```

- Überprüfen Sie, dass die Feldnamen, also „`ssid`“ und „`password`“, in der Datei wifi-creds.txt in Kleinbuchstaben angegeben wurden.
- Stellen Sie sicher, dass alle Feldnamen und Werte durch einen Doppelpunkt (:) getrennt sind. Zum Beispiel. `ssid : ' MyHomeWi-Fi'`
- Stellen Sie sicher, dass Feldwerte, die Leerzeichen enthalten, von einem Paar einfacher Anführungszeichen eingeschlossen sind. Auf dem Mac TextEdit oder einem anderen Texteditor werden einfache Anführungszeichen in der Form '...' angezeigt, aber nicht in der Form '...'. Wenn der Feldwert keine Leerzeichen enthält, kann der Wert ohne einfache Anführungszeichen angegeben werden.

Was bedeutet es, wenn die WLAN- oder Betriebs-LED-Anzeige des Fahrzeugs blau blinkt?

Wenn der USB-Stick eine Datei namens wifi-creds.txt enthält, blinkt die WLAN-LED-Anzeige blau, während das Fahrzeug versucht, sich mit dem in der Datei angegebenen WLAN-Netzwerk zu verbinden.

Wenn der USB-Stick ein Verzeichnis `models` enthält, blinkt die Power-LED blau, während das Fahrzeug versucht, die Modelldateien innerhalb des Verzeichnisses zu laden.

Wenn das USB-Laufwerk sowohl die Datei wifi-creds.txt als auch ein Verzeichnis `models` enthält, verarbeitet das Fahrzeug die beiden nacheinander, beginnend mit einem Versuch, sich mit dem WLAN zu verbinden, und dann werden die Modelle geladen.

Die WLAN-LED kann auch für zwei Sekunden rot leuchten, wenn der Versuch einer WLAN-Verbindung fehlschlägt.

Wie kann ich über den Hostnamen eine Verbindung zur Gerätekonsole des Fahrzeugs herstellen?

Wenn Sie über den Hostnamen eine Verbindung zur Gerätekonsole des Fahrzeugs herstellen, stellen Sie sicher, dass Sie Folgendes eingeben: `https://`*`hostname`*`.local` in den Browser, wo der *`hostname`* Wert (des `AMSS-`*`1234`* Formats) auf der Unterseite des DeepRacer AWS-Fahrzeugs gedruckt wird.)

Wie stelle ich über die IP-Adresse eine Verbindung zur Gerätekonsole des Fahrzeugs her

Stellen Sie sicher, dass zum Verbinden mit der Gerätekonsole über die IP-Adresse, wie in der Datei device-status.txt (auf dem USB-Laufwerk) angegeben, die folgenden Bedingungen erfüllt sind.

- Vergewissern Sie sich, dass sich Ihr Laptop oder Ihre Mobilgeräte im selben Netzwerk wie das DeepRacer AWS-Fahrzeug befinden.
- Überprüfen Sie, ob Sie sich mit einem VPN verbunden haben; wenn ja, trennen Sie zuerst die Verbindung.
- Probieren Sie ein anderes WLAN-Netzwerk aus. Schalten Sie beispielsweise den persönlichen Hotspot auf Ihrem Handy ein.

So erhalten Sie die Mac-Adresse Ihres DeepRacer AWS-Geräts

Folgen Sie den nachstehenden Anweisungen, um die Mac-Adresse Ihres DeepRacer AWS-Geräts abzurufen:

1. Stellen Sie sicher, dass Ihr DeepRacer AWS-Gerät nur mit einem Wi-Fi-Netzwerk verbunden ist.
2. Connect Sie Ihr DeepRacer AWS-Gerät mit einem Monitor. Sie benötigen ein HDMI-to-HDMI HDMI-to-DVI oder ein ähnliches Kabel und stecken ein Ende des Kabels in den HDMI-Anschluss am Fahrzeuggehäuse und das andere Ende in einen unterstützten Display-Port am Monitor.
3. Connect nach dem Booten des Rechenmoduls des Geräts eine USB-Tastatur über den USB-Anschluss an Ihr AWS DeepRacer an.
4. Geben Sie `deepracer` das Eingabefeld Benutzername ein.
5. Geben Sie im Eingabefeld Password (Passwort) das SSH-Passwort für das Gerät ein.

   Wenn Sie sich zum ersten Mal am Gerät anmelden, geben Sie das Passwort **`deepracer`** in das Eingabefeld ein. Setzen Sie das Passwort bei Bedarf zurück, bevor Sie zum nächsten Schritt übergehen. Sie verwenden das neue Passwort für zukünftige Anmeldungen. Verwenden Sie aus Sicherheitsgründen ein komplexes oder starkes Passwort.
6. Öffnen Sie nach der Anmeldung ein Terminalfenster.

   Sie können die Terminalanwendung über die Suchen-Schaltfläche finden.
7. Geben Sie den folgenden Ubuntu-Shell-Befehl im Terminalfenster ein:

   ```
   ifconfig | grep HWaddr
   ```

   Der Befehl erzeugt eine Ausgabe ähnlich der folgenden:

   ```
   mlan0    Link encap:Ethernet   HWaddr   01:2a:34:b5:c6:de
   ```

   Die hexadezimalen Zahlen sind die MAC-Adresse des Geräts.

So stellen Sie das Standardkennwort Ihrer DeepRacer AWS-Gerätekonsole wieder her

Um das Standardkennwort Ihrer DeepRacer AWS-Gerätekonsole wiederherzustellen, müssen Sie das Standardkennwort abrufen oder zurücksetzen. Das Standardkennwort ist auf der Unterseite des Geräts aufgedruckt, wie in der folgenden Abbildung dargestellt.

Folgen Sie den Anweisungen im folgenden Verfahren, um das Passwort für Ihren DeepRacer AWS-Geräte-Webserver mithilfe eines Ubuntu-Terminalfensters wiederherzustellen.

1. Connect Sie Ihr DeepRacer AWS-Gerät mit einem Monitor. Sie benötigen ein HDMI-to-HDMI HDMI-to-DVI oder ein ähnliches Kabel und stecken ein Ende des Kabels in den HDMI-Anschluss am Fahrzeuggehäuse und das andere Ende in einen unterstützten Display-Port am Monitor.
2. Connect nach dem Booten des Rechenmoduls des Geräts eine USB-Tastatur über den USB-Anschluss an Ihr AWS DeepRacer an.
3. Geben `deepracer` Sie im Feld Nutzername ein.
4. Geben Sie unter Password (Passwort) das SSH-Passwort des Geräts ein.

Wenn Sie sich zum ersten Mal am Gerät anmelden, geben Sie `deepracer` Passwort ein. Setzen Sie das Passwort bei Bedarf zurück, bevor Sie zum nächsten Schritt übergehen. Sie verwenden das neue Passwort für zukünftige Anmeldungen. Verwenden Sie aus Sicherheitsgründen ein komplexes oder starkes Passwort.

5. Öffnen Sie nach Ihrer Anmeldung ein Terminalfenster.

   Sie können die Suchschaltfläche verwenden, um die Terminalfenster-Anwendung zu finden.

6. Um das Standardkonsolenpasswort für das Gerät zu erhalten, geben Sie den folgenden Befehl im Terminalfenster ein:

   ```
   $cat /sys/class/dmi/id/chassis_asset_tag
   ```

   Der Befehl gibt als Ergebnis das Standardpasswort aus.

7. Um das Passwort für die Gerätekonsole auf die Standardeinstellung zurückzusetzen, führen Sie das folgende Python-Skript im Terminalfenster aus:

   ```
   sudo python /opt/aws/deepracer/nginx/reset_default_password.py
   ```

So aktualisieren Sie Ihr DeepRacer AWS-Gerät manuell

Jüngste Änderungen am DeepRacer AWS-Service haben dazu geführt, dass bestimmte ältere Geräte, wie die, die auf der AWS re:Invent 2018 vertrieben wurden, nicht automatisch aktualisiert werden können. Führen Sie die folgenden Schritte aus, um ein solches Gerät manuell zu aktualisieren.

Um ein DeepRacer AWS-Gerät manuell zu aktualisieren

1. Laden Sie dieses manuell aktualisierte DeepRacer AWS-Geräteskript auf Ihren Computer herunter und entpacken Sie es.

   Der Standardname der unkomprimierten Datei für dieses Skript lautet `deepracer-device-manual-update.sh`. In diesem Thema wird davon ausgegangen, dass Sie den Standard-Skriptdateinamen verwenden.

2. Kopieren Sie die heruntergeladene und unkomprimierte Skriptdatei (`deepracer-device-manual-update.sh`) von Ihrem Computer auf ein USB-Laufwerk.

3. Schließen Sie das Gerät über ein HDMI-HDMI-Kabel an einen Monitor an, verbinden Sie es mit einer USB-Tastatur und einer USB-Maus.
4. Schalten Sie das Gerät ein und melden Sie sich beim Betriebssystem an, nachdem das Gerät hochgefahren wurde.

   Sie müssen das neue Betriebssystem-Passwort festlegen, wenn dies Ihre erste Anmeldung am Gerät ist.
5. Schließen Sie das USB-Laufwerk an das Gerät an und kopieren Sie die Skriptdatei in einen Ordner (zum Beispiel: ~/Desktop) auf dem Gerät.
6. Geben Sie in einem Terminal auf dem Gerät den folgenden Befehl ein, um zum Ordner der Skriptdatei zu wechseln und der Skriptdatei Ausführungsberechtigung hinzuzufügen:

   ```
   cd ~/Desktop
   chmod +x deepracer-device-manual-update.sh
   ```
7. Geben Sie den folgenden Shell-Befehl ein, um das Skript auszuführen:

   ```
   sudo -H ./deepracer-device-manual-update.sh
   ```
8. Wenn Sie mit der Aktualisierung des Geräts fertig sind, öffnen Sie einen Webbrowser auf Ihrem Computer oder auf einem Mobilgerät und navigieren Sie zur IP-Adresse des Geräts, z. B. 192.168.1.11 in einem Heimnetzwerk oder 10.56.101.13 in einem Büronetzwerk.

   Stellen Sie sicher, dass Ihr Gerät mit Ihrem WLAN-Netzwerk verbunden ist, und verwenden Sie einen Browser in demselben Netzwerk ohne Tunneling über ein VPN.
9. Geben Sie in der Gerätekonsole das Passwort für die Gerätekonsole ein, um sich anzumelden. Warten Sie, bis der Aktualisierungsbildschirm angezeigt wird. Wenn Sie zu weiteren Aktualisierungen aufgefordert werden, befolgen Sie die Anweisungen dort.

So diagnostizieren und lösen Sie häufig auftretende DeepRacer AWS-Betriebsprobleme

Wenn Sie Reinforcement-Learning mit Ihrem DeepRacer AWS-Fahrzeug ausprobieren, funktioniert das Gerät möglicherweise nicht mehr. Die folgenden Themen zur Fehlerbehebung helfen Ihnen, Probleme zu diagnostizieren und zu beheben.

Themen

- Warum zeigt der Videoplayer auf der Gerätekonsole den Videostream von der Kamera meines Fahrzeugs nicht an?
- Warum bewegt sich mein DeepRacer AWS-Fahrzeug nicht?
- Warum sehe ich das neueste Geräte-Update nicht? Wie erhalte ich das neueste Update?
- Warum ist mein DeepRacer AWS-Fahrzeug nicht mit meinem Wi-Fi-Netzwerk verbunden?
- Warum dauert das Laden der DeepRacer AWS-Gerätekonsolenseite lange?
- Warum funktioniert ein Modell nicht gut, wenn es in einem DeepRacer AWS-Fahrzeug eingesetzt wird?

Warum zeigt der Videoplayer auf der Gerätekonsole den Videostream von der Kamera meines Fahrzeugs nicht an?

Nachdem Sie sich bei der DeepRacer AWS-Gerätekonsole angemeldet haben, sehen Sie im Videoplayer in Device Controls kein Live-Video, das von der am DeepRacer AWS-Fahrzeug montierten Kamera gestreamt wurde. Folgendes könnte dieses Problem verursachen:

- Die Verbindung der Kamera zum USB-Anschluss ist möglicherweise locker. Ziehen Sie das Kameramodul vom Fahrzeug ab, schließen Sie es wieder an den USB-Anschluss an, schalten Sie das Gerät aus und dann wieder ein, um es neu zu starten.
- Die Kamera ist möglicherweise fehlerhaft. Verwenden Sie eine bekannte funktionierende Kamera eines anderen DeepRacer AWS-Fahrzeugs, falls verfügbar, um zu testen, ob dies die Ursache ist.

Warum bewegt sich mein DeepRacer AWS-Fahrzeug nicht?

Sie haben Ihr DeepRacer AWS-Fahrzeug eingeschaltet, können es aber nicht bewegen. Folgendes könnte dieses Problem verursachen:

- Die Power-Bank des Fahrzeugs ist nicht eingeschaltet oder nicht mit dem Fahrzeug verbunden. Stellen Sie sicher, dass das mitgelieferte USB-C-to-USB C-Kabel zwischen dem USB-C-Anschluss an der Powerbank und dem USB-C-Anschluss am Fahrzeuggehäuse angeschlossen ist. Stellen Sie sicher, dass die LED-Anzeigen, die den Ladezustand der Powerbank anzeigen, leuchten. Ist dies nicht der Fall, drücken Sie den Netzschalter an der Power-Bank und dann den Netzschalter am Fahrzeugchassis, um das Gerät hochzufahren. Das Gerät ist hochgefahren, wenn die Rücklichter leuchten.

- Wenn die Power-Bank eingeschaltet und das Fahrzeug hochgefahren ist, das Fahrzeug sich jedoch weder im manuellen noch im autonomen Fahrmodus bewegt, überprüfen Sie, ob die Batterie des Fahrzeugs unter dem Fahrzeugchassis geladen und eingeschaltet ist. Wenn nicht, laden Sie die Fahrzeugbatterie wieder auf und schalten Sie es ein, nachdem die Batterie vollständig geladen ist.
- Die Kabelstecker der Fahrzeugbatterie ist nicht vollständig an den Stromkabelanschluss des Antriebsmoduls des Geräts angeschlossen. Stellen Sie sicher, dass die Kabelstecker fest angeschlossen sind.
- Die Batteriekabel sind defekt. Testen Sie die Batterie wenn möglich an einem anderen funktionierenden Fahrzeug, um herauszufinden, ob dies der Fall ist.
- Der Netzschalter der Fahrzeugbatterie ist nicht eingeschaltet. Schalten Sie den Netzschalter ein und stellen Sie sicher, dass Sie zwei Signaltöne gefolgt von einem langen Signalton hören.

Warum sehe ich das neueste Geräte-Update nicht? Wie erhalte ich das neueste Update?

Warum ist die Software meines DeepRacer AWS-Fahrzeugs veraltet?

- In letzter Zeit wurde keine automatische Aktualisierung auf dem Gerät durchgeführt. Möglicherweise müssen Sie eine manuelle Aktualisierung durchführen.
- Das Fahrzeug ist nicht mit dem Internet verbunden. Stellen Sie sicher, dass das Fahrzeug mit einem WLAN- oder Ethernet-Netzwerk mit Internetzugang verbunden ist.

Warum ist mein DeepRacer AWS-Fahrzeug nicht mit meinem Wi-Fi-Netzwerk verbunden?

Wenn ich den Netzwerkstatus auf dem Betriebssystem des Fahrzeugs überprüfe, sehe ich nicht, dass das DeepRacer AWS-Fahrzeug mit einem Wi-Fi-Netzwerk verbunden ist. Dies könnte auf folgende Probleme zurückzuführen sein:

- Für das DeepRacer AWS-Fahrzeug wurde kein WLAN konfiguriert. Befolgen Sie diese Einrichtungsanweisung, um das WLAN-Netzwerk für Ihr Fahrzeug einzurichten.
- Das Fahrzeug befindet sich außerhalb des aktiven Netzwerksignalbereichs. Stellen Sie sicher, dass Sie das Fahrzeug innerhalb des gewählten WLAN-Netzwerkbereichs betreiben.

- Das vorkonfigurierte WLAN-Netzwerk des Fahrzeugs stimmt nicht mit dem verfügbaren WLAN-Netzwerk überein. Befolgen Sie die Einrichtungsanweisung, um das WLAN-Netzwerk einzurichten, für das kein aktiver CAPTCHA erforderlich ist.

Warum dauert das Laden der DeepRacer AWS-Gerätekonsolenseite lange?

Als ich versucht habe, die Gerätekonsole meines DeepRacer AWS-Fahrzeugs zu öffnen, scheint das Laden der Gerätekonsolenseite lange zu dauern.

- Ihr Fahrzeug ist heruntergefahren oder ausgeschaltet. Stellen Sie sicher, dass das Fahrzeug eingeschaltet ist, wenn die Rücklichter eingeschaltet sind.
- Die IP-Adresse Ihres Fahrzeugs wurde geändert, höchstwahrscheinlich vom DHCP-Server Ihres Netzwerks. Um die neue IP-Adresse des Fahrzeugs herauszufinden, folgen Sie diesen Anweisungen zur Einrichtung, um sich über die USB-USB-Kabelverbindung zwischen Ihrem Computer und dem Fahrzeug an der Gerätekonsole anzumelden. Zeigen Sie die neue IP-Adresse unter Settings (Einstellungen) an. Alternativ können Sie die Liste der mit Ihrem Netzwerk verbundenen Geräte durchsuchen, um die neue IP-Adresse zu ermitteln. Wenn Sie kein Netzwerkadministrator sind, bitten Sie den Administrator, dies für Sie zu untersuchen.

Warum funktioniert ein Modell nicht gut, wenn es in einem DeepRacer AWS-Fahrzeug eingesetzt wird?

Nach dem Training eines Modells und der Bereitstellung seiner Artefakte auf Ihrem DeepRacer AWS-Fahrzeug funktioniert das Fahrzeug manchmal nicht wie erwartet. Was ist schiefgegangen?

Im Allgemeinen ist die Optimierung eines trainierten Modells für die Übertragung auf ein physisches DeepRacer AWS-Fahrzeug ein schwieriger Lernprozess. Häufig ist ein iterativer Prozess durch Ausprobieren erforderlich. Allgemeine Richtlinien zu bewährten Methoden finden Sie unter Optimieren Sie DeepRacer AWS-Schulungsmodelle für reale Umgebungen.

Im Folgenden sind einige wahrscheinliche allgemeine Faktoren aufgeführt, die sich auf die Modellleistung in Ihrem DeepRacer AWS-Fahrzeug auswirken:

- Ihr Modell ist im Training nicht konvergiert. Klonen Sie das Modell, um das Training fortzusetzen oder das Modell für einen längeren Zeitraum neu zu schulen. Stellen Sie sicher, dass der Agent in der Simulation Runden kontinuierlich beendet, d. h. der Prozess weist gegen Ende der Schulung 100 % auf.

- Ihr Modell wurde übermäßig geschult (d. h. übermäßig ausgestattet). Es passt zu gut zu den Schulungsdaten, weist aber keine verallgemeinertes Verhalten in unbekannten Situationen auf. Schulen Sie das Modell mit einer flexibleren oder besser geeigneten Belohnungsfunktionneu oder erhöhen Sie die Granularität des Aktionsraums. Sie sollten auch ein geschultes Modell auf verschiedenen Strecken auswerten, um zu sehen, ob das Modell gut verallgemeinert wird.
- Ihr DeepRacer AWS-Fahrzeug wurde möglicherweise nicht richtig kalibriert. Um dies zu testen, wechseln Sie zum manuellen Fahren und prüfen Sie, ob das Fahrzeug wie erwartet fährt. Wenn nicht, kalibrieren Sie das Fahrzeug.
- Sie fahren das Fahrzeug autonom auf einer Strecke, die nicht den Anforderungen entspricht. Streckenanforderungen finden Sie unter Erstellen Sie Ihren physischen Track für AWS DeepRacer.
- Es gibt zu viele Objekte in der Nähe der physischen Strecke, wodurch sich die Strecke deutlich von der simulierten Umgebung unterscheidet. Räumen Sie die Umgebung der Strecke frei, damit die physische Strecke der simulierten möglichst ähnlich ist.
- Eine Spiegelung auf der Streckenoberfläche oder einem Objekt in der Nähe kann Blendeffekte erzeugen und die Kamera beeinträchtigen. Passen Sie die Beleuchtung an und vermeiden Sie Strecken auf glatten Betonböden oder anderen glänzenden Materialien.

Dokumentenverlauf für den DeepRacer AWS-Entwicklerhandbuch

In der folgenden Tabelle werden die wichtigen Änderungen an der Dokumentation seit der letzten Version von AWS beschrieben DeepRacer.

Änderung	Beschreibung	Datum
Updates für die AWS DeepRacer League 2023	Es wurden mehrere Themen aktualisiert, die sich auf die AWS DeepRacer League beziehen. Weitere Informationen zur DeepRacer AWS-League-Saison 2023 finden Sie in den Allgemeinen Geschäftsbedingungen.	1. März 2023
Das Thema „ DeepRacer AWS-Modelle mithilfe von SageMaker KI-Notebooks trainieren und evaluieren“ wurde vorübergehend entfernt	Das Thema Trainieren und Evaluieren von DeepRacer AWS-Modellen mithilfe von SageMaker KI-Notebooks wurde aus entferntModelle trainieren und auswerten. Derzeit werden die Verfahren für die Verwendung eines AWS SageMaker KI-Notebooks mit AWS DeepRacer aktualisiert.	1. November 2022
Aktualisierungen der von IAM verwalteten Richtlinien für die Mehrbenutzerfunktion	Neue verwaltete Richtlinien `AWSDeepRacerAccountAdminAccess` und wurden hinzugefügt`AWSDeepRacerDefaultMultiUserAccess` , sodass Sie im Mehrbenut	26. Oktober 2021

	zermodus mehrere Teilnehmer unter einem DeepRacer AWS-Konto sponsern können, siehethe section called “AWSDeepRacerAccountAdminAccess”.	
Updates für die Mehrbenutzerfunktion	AWS unterstützt DeepRacer jetzt die Mehrbenutzerfunktion, mit der ein AWS Konto mehrere Teilnehmer für Rennen und Training sponsern kann. Weitere Informationen finden Sie unter Mehrbenutzermodus.	26. Oktober 2021
Updates für Rennen mit mehreren Fahrzeugen und Hindernisvermeidung	AWS unterstützt DeepRacer jetzt neue Sensortypen von Stereokameras und LIDAR, die Rennen mit mehreren Fahrzeugen und die Vermeidung von Hindernissen ermöglichen. Weitere Informationen finden Sie unter the section called “Renntypen verstehen und Sensoren aktivieren”.	02. Dezember 2019

Updates für Community-Rennen	AWS ermöglicht es DeepRacer AWS-Benutzern DeepRacer jetzt, ihre eigenen Rennveranstaltungen, sogenannte Community-Rennen, zu organisieren. Private Bestenlisten stehen nur eingeladenen Benutzern offen. Weitere Informationen finden Sie unter Nehmen Sie an einem Rennen teil.	02. Dezember 2019
Updates für die allgemeine Verfügbarkeit	AWS bietet DeepRacer jetzt robustere Methoden zum Trainieren und Evaluieren von Deep-Learning-Modellen. Die Benutzeroberfläche wurde aktualisiert und erklärt. Weitere Optionen und präzise Daten stehen zum Erstellen eigener physischer Strecken zur Verfügung. Informationen zur Fehlerbehebung sind verfügbar.	29. April 2019

DeepRacer AWS-Entwicklerhandbuch für Erstveröffentlichung	Erste Veröffentlichung der Dokumentation, die dem DeepRacer AWS-Benutzer helfen soll, Reinforcement-Learning zu erlernen und seine Anwendungen für autonome Rennen mithilfe der DeepRacer AWS-Konsole, des DeepRacer AWS-Simulators und eines DeepRacer AWS-Scale-Modellfahrzeugs zu erkunden.	28. November 2018

AWS Glossar

Die neueste AWS Terminologie finden Sie im AWS Glossar in der AWS-Glossar Referenz.

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.