



Guia do usuário

# AWS CodeStar



# AWS CodeStar: Guia do usuário

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

As marcas comerciais e imagens comerciais da Amazon não podem ser usadas no contexto de nenhum produto ou serviço que não seja da Amazon, nem de qualquer maneira que possa gerar confusão entre os clientes ou que deprecie ou desprestige a Amazon. Todas as outras marcas comerciais que não pertencem à Amazon pertencem a seus respectivos proprietários, que podem ou não ser afiliados, patrocinados pela Amazon ou ter conexão com ela.

---

# Table of Contents

.....	viii
O que é AWS CodeStar? .....	1
O que posso fazer com AWS CodeStar? .....	1
Como faço para começar com AWS CodeStar? .....	2
Configuração .....	3
Etapa 1: Criar uma conta da .....	3
Inscreva-se para um Conta da AWS .....	3
Criar um usuário com acesso administrativo .....	4
Etapa 2: Criar a função AWS CodeStar de serviço .....	5
Etapa 3: configurar as permissões do IAM do usuário .....	5
Etapa 4: criar um par de EC2 chaves da Amazon para AWS CodeStar projetos .....	6
Etapa 5: abrir o AWS CodeStar console .....	6
Próximas etapas .....	7
Começando com AWS CodeStar .....	8
Etapa 1: criar um AWS CodeStar projeto .....	9
Etapa 2: Adicionar informações de exibição a seu perfil de usuário do AWS CodeStar .....	14
Etapa 3: Visualizar o projeto .....	15
Etapa 4: Confirmar uma alteração .....	16
Etapa 5: Adicionar mais membros à equipe .....	21
Etapa 6: Limpeza .....	23
Etapa 7: preparar o projeto para um ambiente de produção .....	24
Próximas etapas .....	25
Tutorial do projeto com tecnologia sem servidor .....	25
Visão geral .....	26
Etapa 1: criar o projeto .....	27
Etapa 2: Explorar os recursos do projeto .....	28
Etapa 3: Testar o serviço da web .....	31
Etapa 4: Configurar sua estação de trabalho local para editar o código do projeto .....	32
Etapa 5: Adicionar lógica ao serviço da web .....	33
Etapa 6: Testar o serviço da web aprimorado .....	35
Etapa 7: Adicionar um teste de unidade ao serviço da web .....	36
Etapa 8: Visualizar os resultados do teste de unidade .....	39
Etapa 9: Limpeza .....	39
Próximas etapas .....	40

AWS CLI Tutorial do projeto .....	41
Etapa 1: Fazer download e analisar o código-fonte de exemplo .....	42
Etapa 2: Fazer download do modelo da cadeia de ferramentas de exemplo .....	42
Etapa 3: Teste seu modelo de conjunto de ferramentas no AWS CloudFormation .....	44
Etapa 4: Fazer upload do código-fonte e do modelo da cadeia de ferramentas .....	44
Etapa 5: criar um projeto no AWS CodeStar .....	45
Projeto de habilidade do Alexa .....	48
Pré-requisitos .....	48
Etapa 1: Criar o projeto e conectar sua conta de desenvolvedor da Amazon .....	49
Etapa 2: Testar sua habilidade no Simulador do Alexa .....	51
Etapa 3: Explorar os recursos do projeto .....	51
Etapa 4: Fazer uma alteração na resposta da habilidade .....	51
Etapa 5: Configurar sua estação de trabalho local para se conectar ao repositório do projeto .....	52
Próximas etapas .....	53
Tutorial: Criar um projeto com um repositório GitHub de origem .....	53
Etapa 1: criar o projeto e criar seu GitHub repositório .....	53
Etapa 2: exibir seu código-fonte .....	57
Etapa 3: criar uma solicitação GitHub pull .....	57
Modelos de projeto .....	59
AWS CodeStar Arquivos e recursos do projeto .....	59
Primeiros passos: escolher um modelo de projeto .....	61
Escolha uma plataforma de computação de modelo .....	61
Escolha um tipo de aplicativo de modelo .....	62
Escolha um modelo de linguagem de programação .....	63
Como fazer alterações em seu AWS CodeStar projeto .....	63
Alterar o código-fonte do aplicativo e enviar alterações por push .....	64
Alterar recursos do aplicativo com o arquivo template.yml .....	64
.....	65
AWS CodeStar Melhores práticas .....	66
Melhores práticas de segurança para recursos do AWS CodeStar .....	66
Melhores práticas a fim de configurar versões para dependências .....	66
Monitoramento e registro em log das melhores práticas para recursos do AWS CodeStar .....	67
Como trabalhar com projetos do .....	68
Criar um projeto .....	70
Criar um projeto no AWS CodeStar (console) .....	70

Criar um projeto em AWS CodeStar (AWS CLI) .....	75
Use um IDE com AWS CodeStar .....	82
Use AWS Cloud9 com AWS CodeStar .....	83
Use o Eclipse com AWS CodeStar .....	91
Use o Visual Studio com AWS CodeStar .....	96
Alterar recursos do projeto .....	98
Alterações de recursos compatíveis .....	98
Adicionar um estágio ao AWS CodePipeline .....	100
Alterar configurações do AWS Elastic Beanstalk ambiente .....	101
Alterar uma AWS Lambda função no código-fonte .....	101
Habilitar rastreamento para um projeto .....	101
Adicionar um recurso a um projeto .....	104
Adicionar uma função do IAM a um projeto .....	110
Adicionar um estágio e uma endpoint de produção a um projeto .....	111
Use com segurança os parâmetros SSM em um projeto AWS CodeStar .....	120
Mudar tráfego para um projeto do AWS Lambda .....	122
Faça a transição do seu CodeStar projeto da AWS para produção .....	130
Criar um GitHub repositório .....	131
Trabalhar com tags de projetos .....	132
Adicionar uma tag a um projeto .....	132
Remover uma tag de um projeto .....	132
Obter uma lista de tags de um projeto .....	133
Excluir um projeto .....	133
Excluir um projeto no AWS CodeStar (console) .....	134
Excluir um projeto em AWS CodeStar (AWS CLI) .....	135
Como trabalhar com equipes do .....	137
Adicionar membros da equipe a um projeto .....	139
Adicionar um membro da equipe (Console) .....	141
Adicionar e exibir membros da equipe (AWS CLI) .....	143
Gerenciar permissões da equipe .....	144
Gerenciar permissões da equipe (console) .....	145
Gerenciar permissões da equipe (AWS CLI) .....	146
Remover membros da equipe de um projeto .....	146
Remover membros da equipe (console) .....	147
Remover membros da equipe (AWS CLI) .....	148
Trabalhando com seu perfil de AWS CodeStar usuário .....	149

Gerenciar informações de exibição .....	149
Gerenciar seu perfil de usuário (console) .....	150
Gerenciar perfis de usuário (AWS CLI) .....	151
Adicionar uma chave pública a seu perfil de usuário .....	154
Gerenciar a chave pública (console) .....	154
Gerenciar sua chave pública (AWS CLI) .....	155
Conecte-se à Amazon EC2 Instance com sua chave privada .....	156
Segurança .....	158
Proteção de dados .....	159
Criptografia de dados em AWS CodeStar .....	160
Gerenciamento de Identidade e Acesso .....	160
Público .....	161
Autenticação com identidades .....	161
Gerenciamento do acesso usando políticas .....	165
Como a AWS CodeStar trabalha com o IAM .....	167
AWS CodeStar Políticas e permissões em nível de projeto .....	178
Exemplos de políticas baseadas em identidade .....	185
Solução de problemas .....	216
Registrando chamadas de AWS CodeStar API com AWS CloudTrail .....	218
AWS CodeStar Informações em CloudTrail .....	219
Entendendo as entradas do arquivo de AWS CodeStar log .....	220
Validação de conformidade .....	221
Resiliência .....	221
Segurança da infraestrutura .....	222
Limites .....	223
Solução de problemas AWS CodeStar .....	225
Falha ao criar o projeto: um projeto não foi criado .....	225
Criação do projeto: vejo um erro quando tento editar a EC2 configuração da Amazon ao criar um projeto .....	226
Exclusão do projeto: um AWS CodeStar projeto foi excluído, mas os recursos ainda existem ..	227
Falha no gerenciamento da equipe: um usuário do IAM não pôde ser adicionado a uma equipe em um AWS CodeStar projeto .....	228
Falha de acesso: um usuário federado não pode acessar um projeto AWS CodeStar .....	229
Falha de acesso: um usuário federado não pode acessar ou criar um ambiente AWS Cloud9 ..	229
Falha de acesso: um usuário federado pode criar um AWS CodeStar projeto, mas não pode visualizar os recursos do projeto .....	229

Problema de função de serviço: a função de serviço não pôde ser criada .....	230
Problema de função de serviço: a função de serviço não é válida ou está ausente .....	230
Problema de função do projeto: as verificações do status de AWS Elastic Beanstalk saúde falham em instâncias em um AWS CodeStar projeto .....	231
Problema de função de projeto: uma função de projeto não é válida ou está ausente .....	232
Extensões de projetos: não é possível se conectar ao JIRA .....	232
GitHub: Não consigo acessar o histórico de commits, problemas ou código de um repositório .	232
AWS CloudFormation: criação da pilha revertida para permissões ausentes .....	233
AWS CloudFormation não está autorizado a realizar iam: PassRole na função de execução do Lambda .....	233
Não é possível criar a conexão para um GitHub repositório .....	234
Notas da versão .....	235
AWS Glossário .....	241

Em 31 de julho de 2024, a Amazon Web Services (AWS) interromperá o suporte para criação e visualização AWS CodeStar de projetos. Depois de 31 de julho de 2024, você não poderá mais acessar o AWS CodeStar console nem criar novos projetos. No entanto, os AWS recursos criados por AWS CodeStar, incluindo seus repositórios de origem, pipelines e compilações, não serão afetados por essa alteração e continuarão funcionando. AWS CodeStar As conexões e AWS CodeStar notificações não serão afetadas por essa descontinuação.

Se você deseja monitorar o trabalho, desenvolver código e criar, testar e implantar seus aplicativos, a Amazon CodeCatalyst fornece um processo de introdução simplificado e funcionalidades adicionais para gerenciar seus projetos de software. Saiba mais sobre a [funcionalidade](#) e [os preços](#) da Amazon CodeCatalyst.

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.

# O que é AWS CodeStar?

AWS CodeStar é um serviço baseado em nuvem para criar, gerenciar e trabalhar com projetos de desenvolvimento de software em AWS. Você pode desenvolver, criar e implantar aplicativos rapidamente em AWS um AWS CodeStar projeto. Um AWS CodeStar projeto cria e integra AWS serviços para sua cadeia de ferramentas de desenvolvimento de projetos. Dependendo do modelo de AWS CodeStar projeto escolhido, esse conjunto de ferramentas pode incluir controle de origem, criação, implantação, servidores virtuais ou recursos sem servidor e muito mais. AWS CodeStar também gerencia as permissões necessárias para os usuários do projeto (chamados de membros da equipe). Ao adicionar usuários como membros da equipe a um AWS CodeStar projeto, os proprietários do projeto podem, de forma rápida e simples, conceder a cada membro da equipe acesso adequado à função de um projeto e seus recursos.

## Tópicos

- [O que posso fazer com AWS CodeStar?](#)
- [Como faço para começar com AWS CodeStar?](#)

# O que posso fazer com AWS CodeStar?

Você pode usá-lo AWS CodeStar para ajudá-lo a configurar o desenvolvimento de aplicativos na nuvem e gerenciar seu desenvolvimento a partir de um painel único e centralizado. Especificamente, é possível:

- Inicie novos projetos de software AWS em minutos usando modelos para aplicativos web, serviços web e muito mais: AWS CodeStar inclui modelos de projeto para vários tipos de projetos e linguagens de programação. Como AWS CodeStar cuida da configuração, todos os recursos do seu projeto são configurados para trabalhar juntos.
- Gerenciar o acesso ao projeto para a equipe: o AWS CodeStar fornece um console central onde você pode atribuir membros da equipe do projeto às funções necessárias para acessar ferramentas e recursos. Essas permissões são aplicadas automaticamente em todos os AWS serviços usados em seu projeto, então você não precisa criar ou gerenciar políticas complexas do IAM.
- Visualize, opere e colabore em seus projetos em um só lugar: AWS CodeStar inclui um painel do projeto que fornece uma visão geral do projeto, de sua cadeia de ferramentas e de eventos importantes. Você pode monitorar as atividades mais recentes do projeto, como

confirmações recentes de código e acompanhar o status das alterações no código, criar resultados e implantações, tudo a partir da mesma página da web. Você pode monitorar o que está acontecendo no projeto a partir de um único painel e analisar os problemas a serem investigados.

- Iterar rapidamente com todas as ferramentas necessárias: o AWS CodeStar inclui uma cadeia de desenvolvimento integrada para o seu projeto. Os membros da equipe enviam código por push, e as alterações são implantadas automaticamente. A integração com o acompanhamento de problemas permite que os membros da equipe controlem o que precisa ser feito em seguida. Você e sua equipe podem trabalhar em conjunto de forma mais rápida e eficiente em todas as fases de entrega de código.

## Como faço para começar com AWS CodeStar?

Para começar com AWS CodeStar:

1. Prepare-se para usar AWS CodeStar seguindo as etapas em [Configuração AWS CodeStar](#).
2. Experimente AWS CodeStar seguindo as etapas do [Começando com AWS CodeStar](#) tutorial.
3. Compartilhe seu projeto com outros desenvolvedores seguindo as etapas em [Adicionar membros da equipe a um AWS CodeStar projeto](#).
4. Integre seu IDE favorito seguindo as etapas em [Use um IDE com AWS CodeStar](#).

# Configuração AWS CodeStar

Antes de começar a usar AWS CodeStar, você deve concluir as etapas a seguir.

## Tópicos

- [Etapa 1: Criar uma conta da](#)
- [Etapa 2: Criar a função AWS CodeStar de serviço](#)
- [Etapa 3: configurar as permissões do IAM do usuário](#)
- [Etapa 4: criar um par de EC2 chaves da Amazon para AWS CodeStar projetos](#)
- [Etapa 5: abrir o AWS CodeStar console](#)
- [Próximas etapas](#)

## Etapa 1: Criar uma conta da

### Inscreva-se para um Conta da AWS

Se você não tiver um Conta da AWS, conclua as etapas a seguir para criar um.

Para se inscrever em um Conta da AWS

1. Abra a <https://portal.aws.amazon.com/billing/inscrição>.
2. Siga as instruções online.

Parte do procedimento de inscrição envolve receber uma chamada telefônica e inserir um código de verificação no teclado do telefone.

Quando você se inscreve em um Conta da AWS, um Usuário raiz da conta da AWS é criado. O usuário-raiz tem acesso a todos os Serviços da AWS e recursos na conta. Como prática recomendada de segurança, atribua o acesso administrativo a um usuário e use somente o usuário-raiz para executar [tarefas que exigem acesso de usuário-raiz](#).

AWS envia um e-mail de confirmação após a conclusão do processo de inscrição. A qualquer momento, você pode visualizar a atividade atual da sua conta e gerenciar sua conta acessando <https://aws.amazon.com/e> escolhendo Minha conta.

## Criar um usuário com acesso administrativo

Depois de se inscrever em um Conta da AWS, proteja seu Usuário raiz da conta da AWS AWS IAM Identity Center, habilite e crie um usuário administrativo para que você não use o usuário root nas tarefas diárias.

### Proteja seu Usuário raiz da conta da AWS

1. Faça login [AWS Management Console](#) como proprietário da conta escolhendo Usuário raiz e inserindo seu endereço de Conta da AWS e-mail. Na próxima página, insira a senha.

Para obter ajuda ao fazer login usando o usuário-raiz, consulte [Fazer login como usuário-raiz](#) no Guia do usuário do Início de Sessão da AWS .

2. Habilite a autenticação multifator (MFA) para o usuário-raiz.

Para obter instruções, consulte [Habilitar um dispositivo de MFA virtual para seu usuário Conta da AWS raiz \(console\) no Guia](#) do usuário do IAM.

### Criar um usuário com acesso administrativo

1. Habilita o Centro de Identidade do IAM.

Para obter instruções, consulte [Habilitar o AWS IAM Identity Center](#) no Guia do usuário do AWS IAM Identity Center .

2. No Centro de Identidade do IAM, conceda o acesso administrativo a um usuário.

Para ver um tutorial sobre como usar o Diretório do Centro de Identidade do IAM como fonte de identidade, consulte [Configurar o acesso do usuário com o padrão Diretório do Centro de Identidade do IAM](#) no Guia AWS IAM Identity Center do usuário.

### Iniciar sessão como o usuário com acesso administrativo

- Para fazer login com o seu usuário do Centro de Identidade do IAM, use o URL de login enviado ao seu endereço de e-mail quando o usuário do Centro de Identidade do IAM foi criado.

Para obter ajuda para fazer login usando um usuário do IAM Identity Center, consulte [Como fazer login no portal de AWS acesso](#) no Guia Início de Sessão da AWS do usuário.

## Atribuir acesso a usuários adicionais

1. No Centro de Identidade do IAM, crie um conjunto de permissões que siga as práticas recomendadas de aplicação de permissões com privilégio mínimo.

Para obter instruções, consulte [Criar um conjunto de permissões](#) no Guia do usuário do AWS IAM Identity Center .

2. Atribua usuários a um grupo e, em seguida, atribua o acesso de autenticação única ao grupo.

Para obter instruções, consulte [Adicionar grupos](#) no Guia do usuário do AWS IAM Identity Center .

## Etapa 2: Criar a função AWS CodeStar de serviço

Crie uma [função de serviço](#) que seja usada para dar AWS CodeStar permissão para administrar AWS recursos e permissões do IAM em seu nome. Você precisa criar a função de serviço apenas uma vez.

### Important

Você deve estar conectado como um usuário administrativo do (ou conta raiz) para criar uma função de serviço. Para obter informações, consulte [Criando seu primeiro grupo e usuário do IAM](#).

1. Abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.
2. Escolha Iniciar projeto.

Se você não vir Start project (Iniciar projeto) e for direcionado para a página da lista de projetos, a função de serviço já terá sido criada.

3. Em Create service role (Criar função de serviço), escolha Yes, create role (Sim, criar função).
4. Saia do assistente. Você vai retornar a ela mais tarde.

## Etapa 3: configurar as permissões do IAM do usuário

Além do usuário administrativo, você pode usar AWS CodeStar como usuário do IAM, usuário federado, usuário raiz ou uma função assumida. Para obter informações sobre o que AWS CodeStar

pode fazer para usuários do IAM versus usuários federados, consulte [Funções CodeStar do AWS IAM](#).

Se você não tiver configurado usuários do IAM, consulte [Usuário do IAM](#).

Para conceder acesso, adicione as permissões aos seus usuários, grupos ou perfis:

- Usuários e grupos em AWS IAM Identity Center:

Crie um conjunto de permissões. Siga as instruções em [Criação de um conjunto de permissões](#) no Guia do usuário do AWS IAM Identity Center .

- Usuários gerenciados no IAM com provedor de identidades:

Crie um perfil para a federação de identidades. Siga as instruções em [Criando um perfil para um provedor de identidades de terceiros \(federação\)](#) no Guia do Usuário do IAM.

- Usuários do IAM:

- Crie um perfil que seu usuário possa assumir. Siga as instruções em [Criação de um perfil para um usuário do IAM](#) no Guia do usuário do IAM.

- (Não recomendado) Vincule uma política diretamente a um usuário ou adicione um usuário a um grupo de usuários. Siga as instruções em [Adição de permissões a um usuário \(console\)](#) no Guia do usuário do IAM.

## Etapa 4: criar um par de EC2 chaves da Amazon para AWS CodeStar projetos

Muitos AWS CodeStar projetos usam AWS CodeDeploy ou AWS Elastic Beanstalk implantam código em EC2 instâncias da Amazon. Para acessar EC2 as instâncias da Amazon associadas ao seu projeto, crie um par de EC2 chaves da Amazon para seu usuário do IAM. Seu usuário do IAM deve ter permissões para criar e gerenciar EC2 chaves da Amazon (por exemplo, permissão para realizar as `ec2:ImportKeyPair` ações `ec2:CreateKeyPair` e). Para obter mais informações, consulte [Amazon EC2 Key Pairs](#).

## Etapa 5: abrir o AWS CodeStar console

Faça login no e AWS Management Console, em seguida, abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.

## Próximas etapas

Parabéns, você concluiu a configuração! Para começar a trabalhar com AWS CodeStar, consulte [Começando com AWS CodeStar](#).

# Começando com AWS CodeStar

Neste tutorial, você usa AWS CodeStar para criar um aplicativo web. Este projeto inclui código de exemplo em um repositório de origem, uma cadeia de ferramentas de implantação e um painel do projeto onde você pode visualizar e monitorar o projeto.

Seguindo as etapas, você:

- Crie um projeto em AWS CodeStar.
- Explorar o projeto.
- Confirmar uma alteração no código.
- Ver a alteração no código implantada automaticamente.
- Adicionar outras pessoas para trabalhar no projeto.
- Limpar recursos do projeto quando não forem mais necessários.

## Note

Se você ainda não tiver feito isso, primeiro conclua as etapas em [Configuração AWS CodeStar](#), inclusive [Etapa 2: Criar a função AWS CodeStar de serviço](#). Você deve entrar usando uma conta que seja de um usuário administrativo no IAM. Para criar um projeto, você deve fazer login AWS Management Console usando um usuário do IAM que tenha a **AWSCodeStarFullAccess** política.

## Tópicos

- [Etapa 1: criar um AWS CodeStar projeto](#)
- [Etapa 2: Adicionar informações de exibição a seu perfil de usuário do AWS CodeStar](#)
- [Etapa 3: Visualizar o projeto](#)
- [Etapa 4: Confirmar uma alteração](#)
- [Etapa 5: Adicionar mais membros à equipe](#)
- [Etapa 6: Limpeza](#)
- [Etapa 7: preparar o projeto para um ambiente de produção](#)

- [Próximas etapas](#)
- [Tutorial: Criação e gerenciamento de um projeto com tecnologia sem servidor no AWS CodeStar](#)
- [Tutorial: Crie um projeto AWS CodeStar com o AWS CLI](#)
- [Tutorial: Crie um projeto Alexa Skill no AWS CodeStar](#)
- [Tutorial: Criar um projeto com um repositório GitHub de origem](#)

## Etapa 1: criar um AWS CodeStar projeto

Nesta etapa, você cria um projeto de desenvolvimento de software JavaScript (Node.js) para um aplicativo web. Você usa um modelo de AWS CodeStar projeto para criar o projeto.

### Note

O modelo de AWS CodeStar projeto usado neste tutorial usa as seguintes opções:

- Categoria do aplicativo: aplicativo Web
- Linguagem de programação: Node.js
- AWS Serviço: Amazon EC2

Se você escolher outras opções, sua experiência poderá não corresponder ao que é documentado neste tutorial.

Para criar um projeto no AWS CodeStar

1. Faça login no e AWS Management Console, em seguida, abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.

Certifique-se de estar conectado à AWS região em que deseja criar o projeto e seus recursos. Por exemplo, para criar um projeto no Leste dos EUA (Ohio), verifique se você selecionou essa AWS região. Para obter informações sobre AWS regiões onde AWS CodeStar está disponível, consulte [Regiões e endpoints](#) na Referência AWS geral.

2. Na página AWS CodeStar, escolha Criar um projeto.
3. Na página Escolha um modelo de projeto, escolha o tipo de projeto na lista de modelos de AWS CodeStar projeto. Você pode usar a barra de filtro estreitar suas opções. Por exemplo, para que

um projeto de aplicativo web escrito em Node.js seja implantado em EC2 instâncias da Amazon, marque as caixas de seleção Aplicativo Web, Node.js e Amazon EC2. Em seguida, escolha entre os modelos disponíveis para esse conjunto de opções.

Para obter mais informações, consulte [AWS CodeStar Modelos de projeto](#).

4. Escolha Próximo.
5. No campo de entrada de texto Nome do projeto, insira um nome para o projeto, como *My First Project*. Em ID do projeto, a ID para o projeto é derivada desse nome do projeto, mas é limitada a 15 caracteres.

Por exemplo, o ID padrão de um projeto chamado *My First Project* é *my-first-projec*. O ID do projeto é a base dos nomes de todos os recursos associados ao projeto. O AWS CodeStar usa esse ID do projeto como parte do URL para o repositório de códigos, bem como para os nomes de funções de acesso e políticas de segurança relacionados no IAM. Depois que o projeto for criado, o ID do projeto não pode ser alterado. Para editar a ID do projeto antes de criar o projeto, em ID do projeto, insira a ID que você deseja usar.

Para obter informações sobre os limites de nomes de projetos e projetos IDs, consulte [Limites em AWS CodeStar](#).

 Note

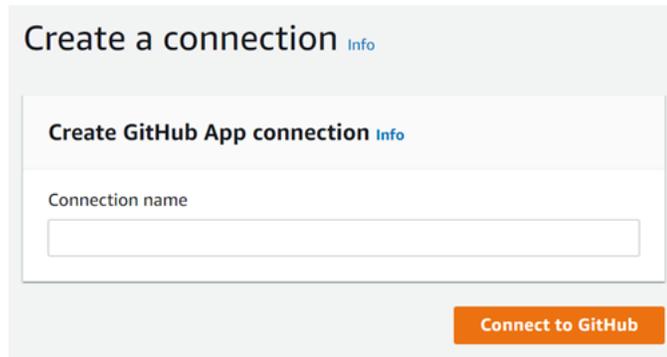
O projeto IDs deve ser exclusivo para sua AWS conta em uma AWS região.

6. Escolha o provedor do repositório AWS CodeCommit ou GitHub.
7. Se você escolher AWS CodeCommit, em Nome do repositório, aceite o nome padrão do AWS CodeCommit repositório ou insira um nome diferente. Em seguida, avance para a etapa 9.
8. Se você escolher GitHub, precisará escolher ou criar um recurso de conexão. Se você tiver uma conexão existente, escolha-a no campo de pesquisa. Caso contrário, crie uma nova conexão agora. Escolha Connect to GitHub.

A página Criar uma conexão é exibida.

 Note

Para criar uma conexão, você precisa ter uma GitHub conta. Se você estiver criando uma conexão para uma organização, você deve ser o proprietário da organização.



- a. Em Criar conexão com o GitHub aplicativo, no campo de texto de entrada do nome da conexão, insira um nome para sua conexão. Escolha Connect to GitHub.

A GitHub página Connect to exibe e mostra o campo GitHub Apps.

- b. Em GitHub Aplicativos, escolha uma instalação de aplicativo ou escolha Instalar um novo aplicativo para criar um.

**Note**

Você instala uma aplicação para todas as suas conexões com um provedor específico. Se você já instalou o GitHub aplicativo AWS Connector for, escolha-o e pule esta etapa.

- c. Na GitHub página Install AWS Connector for, escolha a conta na qual você deseja instalar o aplicativo.

**Note**

Se você instalou a aplicação anteriormente, poderá escolher Configure (Configurar) para prosseguir para uma página de modificação para a instalação da aplicação ou usar o botão Back (Voltar) para retornar ao console.

- d. Se a página Confirmar senha para continuar for exibida, digite sua GitHub senha e escolha Entrar.
- e. Na GitHub página Install AWS Connector for, mantenha os padrões e escolha Instalar.
- f. Na GitHub página Connect to, o ID de instalação da sua nova instalação aparece no campo de entrada de texto GitHub Apps.

Depois que a conexão é criada, na página de CodeStar criação do projeto, a mensagem Pronto para conectar é exibida.

### Note

É possível visualizar sua conexão em Configurações no console do Developer Tools. Para obter mais informações, consulte o tópico de [Conceitos básicos sobre conexões](#).

Select a repository provider

CodeCommit  
Use a new AWS CodeCommit repository for your project.

GitHub  
Use a new GitHub source repository for your project (requires an existing GitHub account).

**The GitHub repository provider now uses CodeStar Connections**  
To use a GitHub repository in CodeStar, create a connection. The connection will use GitHub Apps to access your repository. Use the following options to choose an existing connection or create a new one. [Learn more](#)

Connection  
Choose an existing connection or create a new one and then return to this task.

am:aws:codestar-connections:us-east- X or **Connect to GitHub**

**Ready to connect**  
Your Github connection is ready for use.

Repository owner  
The owner of the new repository. This can be a personal GitHub account or a GitHub organization.

Repository name  
The name of the new repository.

Repository description  
An optional description of the new repository.

Public

- g. Para Proprietário do repositório, escolha a GitHub organização ou sua GitHub conta pessoal.

- h. Em Nome do repositório, aceite o nome padrão do GitHub repositório ou insira um nome diferente.
- i. Escolha Público ou Privado.

 Note

Para usar AWS Cloud9 como seu ambiente de desenvolvimento, você deve escolher Público.

- j. (Opcional) Em Descrição do repositório, insira uma descrição para o GitHub repositório.

 Note

Se você escolher um modelo de projeto Alexa Skill, precisará conectar uma conta de desenvolvedor da Amazon. Para obter mais informações sobre como trabalhar com projetos do Alexa Skill, consulte o [Tutorial: Crie um projeto Alexa Skill no AWS CodeStar](#).

- 9. Se seu projeto estiver implantado em EC2 instâncias da Amazon e você quiser fazer alterações, configure suas EC2 instâncias da Amazon na EC2 Configuração da Amazon. Por exemplo, você pode escolher entre os tipos de instância disponíveis para o projeto.

 Note

Diferentes tipos de EC2 instância da Amazon fornecem níveis diferentes de poder computacional e podem ter custos associados diferentes. Para obter mais informações, consulte [Amazon EC2 Instance Types](#) e [Amazon EC2 Pricing](#).

Se tiver mais de uma nuvem privada virtual (VPC) ou várias sub-redes criadas no Amazon Virtual Private Cloud, você também poderá escolher a VPC e a sub-rede a serem utilizadas. No entanto, se você escolher um tipo de EC2 instância da Amazon que não seja compatível com instâncias dedicadas, não poderá escolher uma VPC cuja localização de instância esteja definida como Dedicada.

Para obter mais informações, consulte [O que é o Amazon VPC?](#) e [Noções básicas de instâncias dedicadas](#).

Em Par de chaves, escolha o par de EC2 chaves da Amazon em que você criou [Etapa 4: criar um par de EC2 chaves da Amazon para AWS CodeStar projetos](#). Selecione Eu reconheço que tenho acesso ao arquivo de chave privada.

10. Escolha Próximo.
11. Analise os recursos e os detalhes da configuração.
12. Escolha Avançar ou Criar projeto. (A opção exibida depende do modelo do projeto.)

Pode levar alguns minutos para criar o projeto, incluindo o repositório.

13. Depois que seu projeto tiver um repositório, você poderá usar a página Repositório para configurar o acesso a ele. Use os links nas Próximas etapas para configurar um IDE, configurar o controle de problemas ou adicionar membros da equipe ao seu projeto.

## Etapa 2: Adicionar informações de exibição a seu perfil de usuário do AWS CodeStar

Quando cria um projeto, você é adicionado à equipe do projeto como proprietário. Se for a primeira vez que você usa AWS CodeStar, será solicitado que você forneça:

- O nome de exibição a ser mostrado aos outros usuários.
- O endereço de e-mail a ser mostrado aos outros usuários.

Essas informações são usadas em seu perfil de AWS CodeStar usuário. Os perfis de usuário não são específicos do projeto, mas estão limitados a uma AWS região. Você deve criar um perfil de usuário em cada AWS região na qual você pertence aos projetos. Cada perfil pode conter informações diferentes, se você preferir.

Digite um nome de usuário e um endereço de e-mail e, em seguida, escolha Next (Avançar).

### Note

Esse nome de usuário e endereço de e-mail são usados em seu perfil de AWS CodeStar usuário. Se seu projeto usa recursos externos AWS (por exemplo, um GitHub repositório ou problemas no Atlassian JIRA), esses provedores de recursos podem ter seus próprios

perfis de usuário, com nomes de usuário e endereços de e-mail diferentes. Para obter mais informações, consulte a documentação do provedor do recurso.

## Etapa 3: Visualizar o projeto

A página AWS CodeStar do seu projeto é onde você e sua equipe visualizam o status dos recursos do seu projeto, incluindo as últimas confirmações do seu projeto, o estado do seu pipeline de entrega contínua e o desempenho de suas instâncias. Para ver mais informações sobre qualquer um desses recursos, escolha a página correspondente na barra de navegação.

Em seu novo projeto, a barra de navegação contém as seguintes páginas:

- A página Visão geral contém informações sobre a atividade do seu projeto, os recursos do projeto e o conteúdo do README do seu projeto.
- A página IDE é onde você conecta seu projeto a um ambiente de desenvolvimento integrado (IDE) para modificar, testar e enviar alterações de código-fonte por push. Ele contém instruções de configuração IDEs para AWS CodeCommit repositórios GitHub e informações sobre seus AWS Cloud9 ambientes.
- A página Repositório exibe os detalhes do seu repositório, incluindo nome, provedor, quando foi modificado pela última vez e clone. URLs Você também pode ver informações sobre o commit mais recente e visualizar e criar pull requests.
- A página Pipeline exibe informações de CI/CD sobre seu pipeline. Você pode visualizar detalhes do pipeline, como nome, ação mais recente e status. Você pode ver o histórico do pipeline e lançar uma alteração. Você também pode ver o status das etapas individuais do seu pipeline.
- A página de monitoramento exibe a Amazon EC2 ou AWS Lambda métricas, dependendo da configuração do seu projeto. Por exemplo, ele exibe a utilização da CPU de qualquer EC2 instância da Amazon implantada por AWS Elastic Beanstalk ou CodeDeploy recursos em seu pipeline. Em projetos que usam AWS Lambda, ele exibe métricas de invocação e erro para a função Lambda. Essas informações são exibidas por hora. Se você usou o modelo de AWS CodeStar projeto sugerido para este tutorial, deverá observar um aumento notável na atividade quando seu aplicativo for implantado pela primeira vez nessas instâncias. Você pode atualizar o monitoramento para ver as alterações na integridade da instância, o que pode ajudar a identificar problemas ou a necessidade de mais recursos.

- A página Problemas serve para integrar seu AWS CodeStar projeto a um projeto JIRA da Atlassian. A configuração desse bloco possibilitará que você e a equipe do projeto acompanhem problemas do JIRA no painel do projeto.

O painel de navegação no lado esquerdo do console é onde você pode navegar entre as páginas Projeto, Equipe e Configurações.

## Etapa 4: Confirmar uma alteração

Primeiro, examine o aplicativo de exemplo que foi incluído em seu projeto. Veja a aparência do aplicativo escolhendo Exibir aplicativo de qualquer lugar na navegação do seu projeto. O aplicativo web de exemplo será exibido em uma nova janela ou guia do navegador. Esse é o exemplo do projeto que AWS CodeStar foi criado e implantado.

Para examinar o próprio código, na barra de navegação, escolha Repositório. Escolha o link em Nome do repositório e o repositório do seu projeto será aberto em uma nova guia ou janela. Leia o conteúdo do arquivo Leiamos do repositório (README .md) e procure o conteúdo desses arquivos.

Nesta etapa, você altera o código e, em seguida, enviar essa alteração para o seu repositório. É possível fazer isso de algumas formas:

- Se o código do projeto estiver armazenado em um GitHub repositório CodeCommit or, você poderá usá-lo AWS Cloud9 para trabalhar com o código diretamente do seu navegador, sem instalar nenhuma ferramenta. Para obter mais informações, consulte [Crie um AWS Cloud9 ambiente para um projeto](#).
- Se o código do projeto estiver armazenado em um CodeCommit repositório e você tiver o Visual Studio ou o Eclipse instalado, poderá usar o AWS Toolkit for Visual Studio ou AWS Toolkit for Eclipse para se conectar mais facilmente ao código. Para obter mais informações, consulte [Use um IDE com AWS CodeStar](#). Se você não tiver o Visual Studio nem o Eclipse, instale um cliente do Git e siga as instruções mais adiante nesta etapa.
- Se o código do projeto estiver armazenado em um GitHub repositório, você poderá usar as ferramentas do seu IDE para se conectar a. GitHub
  - Para o Visual Studio, você pode usar ferramentas como a GitHub extensão do Visual Studio. Para obter mais informações, consulte a página [Visão geral](#) no site GitHub Extension for Visual Studio e [Getting Started with GitHub for Visual Studio](#) no GitHub site.
  - Para o Eclipse, você pode usar uma ferramenta como EGit para o Eclipse. Para obter mais informações, consulte a [EGit documentação](#) no EGit site.

- Para outras informações IDEs, consulte a documentação do seu IDE.
- Para outros tipos de repositórios de código, consulte a documentação do provedor do repositório.

As instruções a seguir mostram como fazer uma alteração secundária na amostra.

Para configurar o computador para confirmar alterações (usuário do IAM)

#### Note

Neste procedimento, presumimos que o código de seu projeto esteja armazenado em um repositório do CodeCommit. Para outros tipos de repositório de código, consulte a documentação do provedor do repositório e, em seguida, passe para o próximo procedimento, "[Para clonar o repositório do projeto e fazer uma alteração](#)".

Se o código estiver armazenado CodeCommit e você já estiver usando CodeCommit ou tiver usado o AWS CodeStar console para criar um ambiente de AWS Cloud9 desenvolvimento para o projeto, não precisará de mais configurações. Passe para o próximo procedimento, [Para clonar o repositório do projeto e fazer uma alteração](#).

1. [Instale o Git](#) no computador local.
2. Faça login no AWS Management Console e abra o console do IAM em <https://console.aws.amazon.com/iam/>.

Faça login como o usuário do IAM que usará as credenciais do Git para conexões com o repositório AWS CodeStar do seu projeto. CodeCommit

3. No painel de navegação do console do IAM, selecione Usuários e, na lista de usuários, escolha o seu usuário do IAM.
4. Na página de detalhes do usuário, escolha a guia Credenciais de segurança e, em Credenciais HTTPS Git CodeCommit para, escolha Gerar.

#### Note

Você não pode escolher suas próprias credenciais de login para as credenciais do Git. Para obter mais informações, consulte [Usar credenciais do Git e HTTPS com CodeCommit](#)

5. Copie as credenciais de login que o IAM gerou para você. Você pode escolher Mostrar e, em seguida, copiar e colar essas informações em um arquivo seguro no computador local ou escolher Fazer download das credenciais para fazer download dessas informações como um arquivo .CSV. Essas informações são necessárias para você se conectar ao CodeCommit.

Após salvar suas credenciais, selecione Close.

 Important

Essa é sua única chance de salvar as credenciais de login. Se você não salvá-los, poderá copiar o nome de usuário do console do IAM, mas não poderá pesquisar a senha. Você deverá reiniciar a senha e salvá-la.

Para configurar o computador para confirmar alterações (usuário federado)

Você pode usar o console para fazer upload dos arquivos para o repositório ou usar o Git para se conectar pelo computador local. Se você estiver usando acesso federado, siga estas etapas para usar o Git para se conectar e clonar o repositório do computador local.

 Note

Neste procedimento, presumimos que o código de seu projeto esteja armazenado em um repositório do CodeCommit. Para outros tipos de repositório de código, consulte a documentação do provedor do repositório e, em seguida, passe para o próximo procedimento, "[Para clonar o repositório do projeto e fazer uma alteração](#)".

1. [Instale o Git](#) no computador local.
2. [Instale AWS CLI](#) o.
3. Configure as credenciais de segurança temporárias para um usuário federado. Para obter informações, consulte [Acesso temporário aos CodeCommit repositórios](#). As credenciais temporárias consistem em:
  - AWS chave de acesso
  - AWS chave secreta
  - Token de sessão

Para obter mais informações sobre credenciais temporárias, consulte [Permissões para GetFederationToken](#).

4. Conecte-se ao seu repositório usando o auxiliar de AWS CLI credenciais. Para obter informações, consulte [Etapas de configuração para conexões HTTPS com CodeCommit repositórios no Linux, macOS ou Unix com o auxiliar de credenciais da AWS CLI](#) ou [Etapas de configuração para conexões HTTPS com CodeCommit repositórios no Windows com o auxiliar de credenciais](#) da CLI AWS
5. O exemplo a seguir mostra como se conectar a um CodeCommit repositório e enviar um commit para ele.

Exemplo: para clonar o repositório do projeto e fazer uma alteração

 Note

Este procedimento mostra como clonar o repositório de código do projeto em seu computador, fazer uma alteração no arquivo `index.html` do projeto e, em seguida, enviar as suas alterações para o repositório remoto. Nesse procedimento, presumimos que o código do seu projeto esteja armazenado em um CodeCommit repositório e que você esteja usando um cliente Git na linha de comando. Para outros tipos de repositórios de código ou ferramentas, consulte a documentação do provedor sobre como clonar o repositório, alterar o arquivo e, em seguida, enviar o código.

1. Se você usou o AWS CodeStar console para criar um ambiente de AWS Cloud9 desenvolvimento para o projeto, abra o ambiente de desenvolvimento e, em seguida, vá para a etapa 3 deste procedimento. Para abrir o ambiente de desenvolvimento, consulte [Abra um AWS Cloud9 ambiente para um projeto](#).

Com seu projeto aberto no AWS CodeStar console, na barra de navegação, escolha Repositório. Em Clone URL, escolha o protocolo para o tipo de conexão que você configurou e, em seguida CodeCommit, copie o link. Por exemplo, se você seguiu as etapas do procedimento anterior para configurar as credenciais do Git CodeCommit, escolha HTTPS.

2. No computador local, abra um terminal ou janela de linha de comando e altere os diretórios para um diretório temporário. Execute o comando `git clone` para clonar o repositório no computador. Cole o link que você copiou. Por exemplo, para CodeCommit usar HTTPS:

```
git clone https://git-codecommit.us-east-2.amazonaws.com/v1/repos/my-first-projec
```

Na primeira vez em que se conectar, será solicitado que você forneça as credenciais do repositório. Para CodeCommit, insira as credenciais de login do Git que você baixou no procedimento anterior.

3. Navegue até o diretório clonado no computador e examine o conteúdo.
4. Abra o arquivo `index.html` (na pasta pública) e faça uma alteração no arquivo. Por exemplo, adicione um parágrafo após a tag `<H2>`, como:

```
<P>Hello, world!</P>
```

Salve o arquivo.

5. No terminal ou no prompt de comando, adicione o arquivo alterado, confirme e envie a alteração por push:

```
git add index.html
git commit -m "Making my first change to the web app"
git push
```

6. Na página Repositório, veja as alterações em andamento. Você verá que o histórico de confirmação do repositório é atualizado com sua confirmação incluindo a mensagem de confirmação. Na página pipeline, você pode ver o pipeline coletar sua alteração no repositório e começar a compilar e implantar. Depois que seu aplicativo de web for implementado, você pode escolher Exibir aplicativo para visualizar sua alteração.

#### Note

Se Failed (Com falha) for exibido para qualquer um dos estágios de pipeline, consulte o seguinte para obter ajuda na solução do problema:

- No estágio Origem, consulte [Solucionar problemas do AWS CodeCommit](#) no Guia do usuário do AWS CodeCommit .

- No estágio Criar, consulte [Solucionar problemas do AWS CodeBuild](#) no Guia do usuário do AWS CodeBuild .
- No estágio Implantar, consulte [Solucionar problemas do AWS CloudFormation](#) no Guia do usuário do AWS CloudFormation .
- Para outros problemas, consulte [Solução de problemas AWS CodeStar](#).

## Etapa 5: Adicionar mais membros à equipe

Cada AWS CodeStar projeto já está configurado com três AWS CodeStar funções. Cada função oferece seu próprio nível de acesso ao projeto e a seus recursos:

- Proprietário: pode adicionar e remover membros da equipe, alterar o painel do projeto e excluir o projeto.
- Colaborador: pode alterar o painel do projeto e contribuir com o código se o código estiver armazenado CodeCommit, mas não pode adicionar ou remover membros da equipe nem excluir o projeto. Essa é a função que você deve escolher para a maioria dos membros da equipe em um AWS CodeStar projeto.
- Visualizador: pode visualizar o painel do projeto, o código do projeto CodeCommit, se o código estiver armazenado, e o estado do projeto, mas não pode mover, adicionar ou remover blocos do painel do projeto.

### Important

Se seu projeto usa recursos externos AWS (por exemplo, um GitHub repositório ou problemas no Atlassian JIRA), o acesso a esses recursos é controlado pelo provedor de recursos, não. AWS CodeStar Para obter mais informações, consulte a documentação do provedor do recurso.

Qualquer pessoa que tenha acesso a um AWS CodeStar projeto pode usar o AWS CodeStar console para acessar recursos externos AWS , mas relacionados ao projeto.

AWS CodeStar não permite que os membros da equipe do projeto participem de nenhum ambiente de AWS Cloud9 desenvolvimento relacionado a um projeto. Para permitir que um

membro da equipe participe de um ambiente compartilhado, consulte [Compartilhe um AWS Cloud9 ambiente com um membro da equipe do projeto](#).

Para obter mais informações sobre equipes e funções de projeto, consulte [Trabalhando com AWS CodeStar equipes](#).

Para adicionar um membro da equipe a um AWS CodeStar projeto (console)

1. Abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.
2. Selecione Projetos no painel de navegação e escolha seu projeto.
3. No painel lateral de navegação do projeto, escolha Equipe.
4. Na página Membros da equipe, escolha Adicionar membro da equipe.
5. Em Escolher usuário, proceda de uma das seguintes maneiras:
  - Se um usuário do IAM já existir para a pessoa que você deseja adicionar, escolha o usuário IAM na lista.

 Note

Os usuários que já foram adicionados a outro AWS CodeStar projeto aparecem na lista de AWS CodeStar usuários existentes.

Em Papel do projeto, escolha o AWS CodeStar papel (Proprietário, Colaborador ou Visualizador) para esse usuário. Essa é uma função em nível de projeto do AWS CodeStar que só pode ser alterada por um proprietário do projeto. Quando aplicado a um usuário do IAM, o papel fornece todas as permissões necessárias para acessar os recursos AWS CodeStar do projeto. Ele aplica as políticas necessárias para criar e gerenciar credenciais do Git para código armazenado CodeCommit no IAM ou fazer o upload das chaves EC2 SSH da Amazon para o usuário no IAM.

 Important

Você não pode fornecer ou alterar as informações de nome de exibição ou de e-mail de um usuário do IAM, a menos que você tenha feito login no console com esse

usuário. Para obter mais informações, consulte [Gerencie as informações de exibição AWS CodeStar do seu perfil de usuário](#).

Escolha Adicionar membro da equipe.

- Se um usuário do IAM não existir para a pessoa que você deseja adicionar ao projeto, escolha Criar novo usuário do IAM;. Você será redirecionado para o console do IAM, onde poderá criar um novo usuário do IAM. Consulte [Como criar usuários do IAM](#) no guia do usuário do IAM para obter mais informações. Depois de criar seu usuário do IAM, retorne ao AWS CodeStar console, atualize a lista de usuários e escolha o usuário do IAM que você criou na lista suspensa. Insira o nome de AWS CodeStar exibição, o endereço de e-mail e a função do projeto que você deseja aplicar a esse novo usuário e escolha Adicionar membro da equipe.

 Note

Para facilitar o gerenciamento, pelo menos um usuário deve receber a função Proprietário do projeto.

6. Envie ao novo membro da equipe as seguintes informações:

- Informações de conexão para seu AWS CodeStar projeto.
- Se o código-fonte estiver armazenado em CodeCommit, [instruções para configurar o acesso com credenciais do Git](#) ao CodeCommit repositório a partir de seus computadores locais.
- Informações sobre como o usuário pode gerenciar seu nome de exibição, endereço de e-mail e chave EC2 SSH pública da Amazon, conforme descrito em [Trabalhando com seu perfil de AWS CodeStar usuário](#).
- Uma senha avulsa e informações de conexão, se o usuário for novo no AWS e você tiver criado um usuário do IAM para essa pessoa. A senha expira na primeira vez em que o usuário entra. O usuário deve escolher uma nova senha.

## Etapa 6: Limpeza

Parabéns! Você concluiu o tutorial. Se você não quiser continuar usando esse projeto e seus recursos, exclua-o para evitar possíveis cobranças contínuas em sua AWS conta.

## Para excluir um projeto no AWS CodeStar

1. Abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.
2. Selecione Projetos no painel de navegação.
3. Selecione o projeto que deseja excluir e selecione Excluir.

Ou abra o projeto e escolha Settings do painel de navegação na esquerda do console. Na página de detalhes do projeto, escolha Excluir projeto.

4. Na Página de confirmação de exclusão, insira excluir. Mantenha a opção Excluir recursos selecionada se quiser excluir recursos do projeto. Escolha Excluir.

A exclusão de um projeto pode demorar vários minutos. Depois de excluído, o projeto não aparece mais na lista de projetos no AWS CodeStar console.

### Important

Se seu projeto usa recursos externos AWS (por exemplo, um GitHub repositório ou problemas no Atlassian JIRA), esses recursos não serão excluídos, mesmo que você marque a caixa de seleção.

Seu projeto não pode ser excluído se alguma política AWS CodeStar gerenciada tiver sido anexada manualmente a funções que não sejam usuários do IAM. Se tiver anexado as políticas gerenciadas do projeto a uma função do usuário federado, você deverá desanexar a política para excluir o projeto. Para obter mais informações, consulte [???](#).

## Etapa 7: preparar o projeto para um ambiente de produção

Depois que tiver criado o projeto, você estará pronto para criar, testar e implantar código. Analise as seguintes considerações para manter o projeto em um ambiente de produção:

- Aplicar patches regularmente e revise as melhores práticas de segurança para as dependências usadas pelo aplicativo. Para obter mais informações, consulte [Melhores práticas de segurança para recursos do AWS CodeStar](#).
- Monitore regularmente as configurações de ambiente sugeridas pela linguagem de programação para o projeto.

## Próximas etapas

Aqui estão alguns outros recursos para ajudar você a aprender sobre AWS CodeStar:

- O [Tutorial: Criação e gerenciamento de um projeto com tecnologia sem servidor no AWS CodeStar](#) usa um projeto que cria e implanta um serviço web usando lógica AWS Lambda e pode ser chamado por uma API no Amazon API Gateway.
- [AWS CodeStar Modelos de projeto](#) descreve outros tipos de projeto que você pode criar.
- [Trabalhando com AWS CodeStar equipes](#) fornece informações sobre como habilitar outras pessoas para ajudá-lo a trabalhar em seus projetos.

## Tutorial: Criação e gerenciamento de um projeto com tecnologia sem servidor no AWS CodeStar

Neste tutorial, você usa AWS CodeStar para criar um projeto que usa o AWS Serverless Application Model (AWS SAM) para criar e gerenciar AWS recursos para um serviço web hospedado em AWS Lambda

AWS CodeStar usa o AWS SAM, do qual depende AWS CloudFormation, para fornecer uma forma simplificada de criar e gerenciar AWS recursos compatíveis, incluindo Amazon API Gateway APIs, AWS Lambda funções e tabelas do Amazon DynamoDB. (Esse projeto não usa nenhuma tabela do Amazon DynamoDB.)

Para obter mais informações, consulte [AWS Serverless Application Model \(AWS SAM\)](#) em GitHub

Pré-requisito: concluir as etapas em [Configuração AWS CodeStar](#).

### Note

Sua AWS conta pode ser cobrada pelos custos relacionados a este tutorial, incluindo os custos dos AWS serviços usados pelo AWS CodeStar. Para obter mais informações, consulte [Preços do AWS CodeStar](#).

### Tópicos

- [Visão geral](#)

- [Etapa 1: criar o projeto](#)
- [Etapa 2: Explorar os recursos do projeto](#)
- [Etapa 3: Testar o serviço da web](#)
- [Etapa 4: Configurar sua estação de trabalho local para editar o código do projeto](#)
- [Etapa 5: Adicionar lógica ao serviço da web](#)
- [Etapa 6: Testar o serviço da web aprimorado](#)
- [Etapa 7: Adicionar um teste de unidade ao serviço da web](#)
- [Etapa 8: Visualizar os resultados do teste de unidade](#)
- [Etapa 9: Limpeza](#)
- [Próximas etapas](#)

## Visão geral

Neste tutorial, você:

1. Use AWS CodeStar para criar um projeto que usa o AWS SAM para criar e implantar um serviço web baseado em Python. Esse serviço web está hospedado AWS Lambda e pode ser acessado por meio do Amazon API Gateway.
2. Explore os recursos principais do projeto, que incluem:
  - O AWS CodeCommit repositório em que o código-fonte do projeto é armazenado. Esse código-fonte inclui a lógica do serviço Web e define os recursos relacionados da AWS .
  - O AWS CodePipeline pipeline que automatiza a construção do código-fonte. Esse pipeline usa o AWS SAM para criar e implantar uma função AWS Lambda, criar uma API relacionada no Amazon API Gateway e conectar a API à função.
  - A função que é implantada em. AWS Lambda
  - A API criada no Amazon API Gateway.
3. Teste o serviço da Web para confirmar se o serviço da Web AWS CodeStar foi criado e implantado conforme o esperado.
4. Configure sua estação de trabalho local para trabalhar com o código-fonte do projeto.
5. Altere o código-fonte do projeto usando sua estação de trabalho local. Quando você adiciona uma função ao projeto e, em seguida, envia as suas alterações para o código-fonte, o AWS CodeStar recompila e reimplanta o serviço da web.

6. Teste o serviço web novamente para confirmar se ele AWS CodeStar foi reconstruído e reimplantado conforme o esperado.
7. Escreva um teste de unidade usando sua estação de trabalho local para substituir alguns de seus testes manuais por um teste automatizado. Quando você envia o teste unitário, AWS CodeStar reconstrói e reimplanta o serviço web e executa o teste unitário.
8. Visualize os resultados dos testes de unidade.
9. Limpe o projeto. Essa etapa ajuda a evitar cobranças AWS em sua conta por custos relacionados a este tutorial.

## Etapa 1: criar o projeto

Nesta etapa, você usa o AWS CodeStar console para criar um projeto.

1. Faça login no AWS Management Console e abra o AWS CodeStar console, em <https://console.aws.amazon.com/codestar/>.

### Note

Você deve fazer login AWS Management Console usando as credenciais associadas ao usuário do IAM que você criou ou identificou. [Configuração AWS CodeStar](#) Esse usuário deve ter a política gerenciada **AWSCodeStarFullAccess** anexada.

2. Escolha a AWS região em que você deseja criar o projeto e seus recursos.

Para obter informações sobre AWS regiões onde AWS CodeStar está disponível, consulte [Regiões e endpoints](#) na Referência AWS geral.

3. Escolha Criar projeto.
4. Na página Choose a project template (Escolher um modelo de projeto):
  - Em Tipo de aplicação, selecione Serviço de web.
  - Em Linguagem de programação, selecione Python.
  - Em Serviço do AWS , selecione AWS Lambda.
5. Escolha a caixa que contém suas seleções. Escolha Próximo.
6. Em Project name (Nome do projeto), digite o nome do projeto (por exemplo, **My SAM Project**). Se você usar um nome diferente do exemplo, certifique-se de usá-lo ao longo de todo este tutorial.

Em ID do projeto, AWS CodeStar escolhe um identificador relacionado para esse projeto (por exemplo, my-sam-project). Se vir outro ID de projeto, certifique-se de usá-lo durante todo este tutorial.

Deixe AWS CodeCommit selecionado e não altere o valor Repository name (Nome do repositório).

7. Escolha Próximo.
8. Revise suas configurações e escolha Criar projeto.

Se for a primeira vez que você usa nesta AWS região, AWS CodeStar em Nome de exibição e E-mail, insira o nome de exibição e o endereço de e-mail que você deseja usar AWS CodeStar para seu usuário do IAM. Escolha Próximo.

9. Espere enquanto AWS CodeStar cria o projeto. Isso pode demorar vários minutos. Não continue até ver o banner provisionado do projeto ao atualizar.

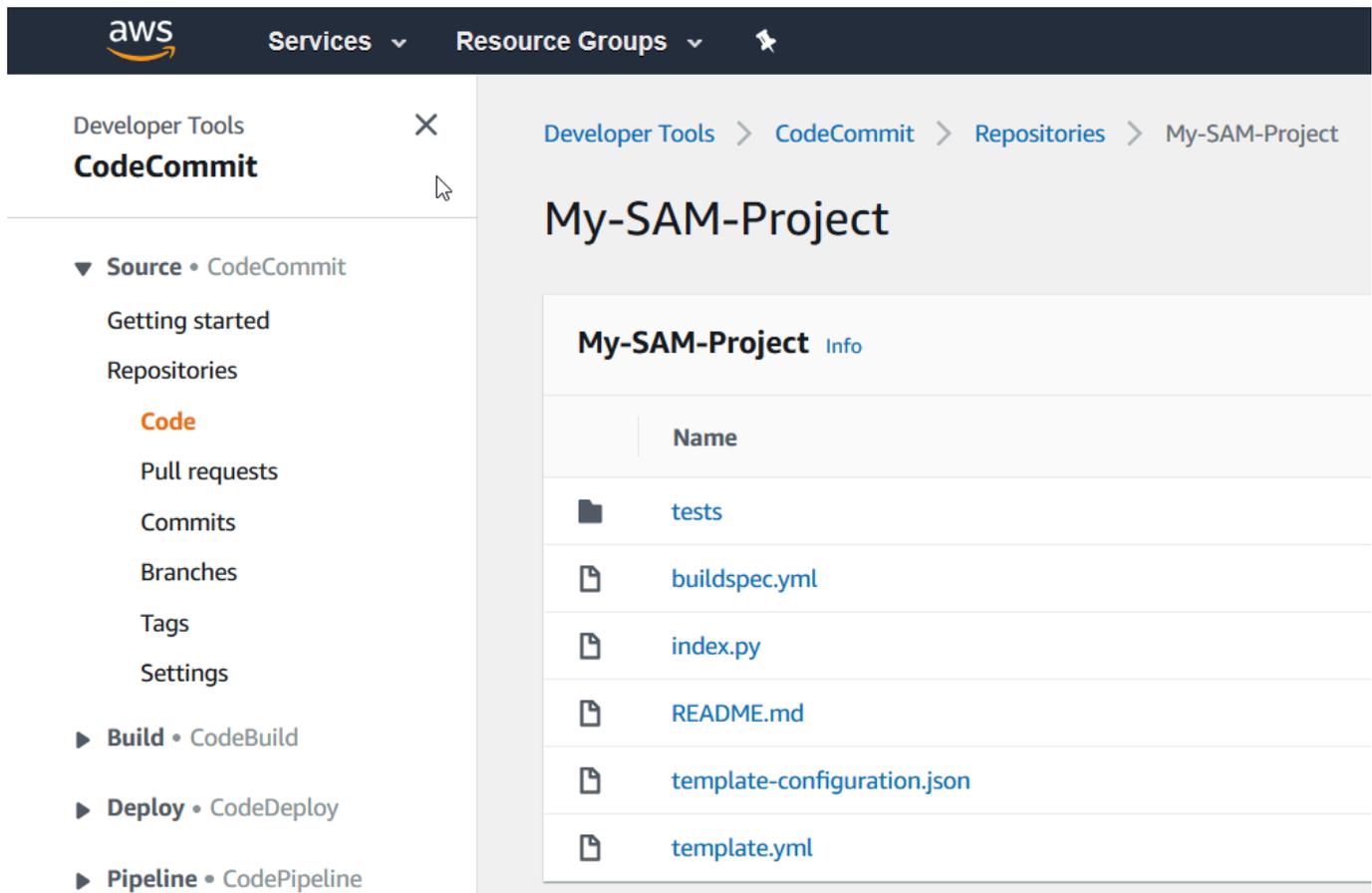
## Etapa 2: Explorar os recursos do projeto

Nesta etapa, você explora quatro dos AWS recursos do projeto para entender como o projeto funciona:

- O AWS CodeCommit repositório em que o código-fonte do projeto é armazenado. AWS CodeStar dá ao repositório o nome my-sam-project, onde my-sam-project está o nome do projeto.
- O AWS CodePipeline pipeline que usa CodeBuild um AWS SAM para automatizar a criação e a implantação da função Lambda e da API do serviço web no API Gateway. AWS CodeStar dá ao pipeline o nome my-sam-project--Pipeline, onde my-sam-project está o ID do projeto.
- A função Lambda que contém a lógica do serviço web. AWS CodeStar dá à função o nome awscodestar-my-sam-project-lambda- HelloWorld -**RANDOM\_ID**, onde:
  - my-sam-project é o ID do projeto.
  - HelloWorld é o ID da função conforme especificado no `template.yaml` arquivo no AWS CodeCommit repositório. Você explorará esse arquivo mais tarde.
  - **RANDOM\_ID** é uma ID aleatória que o AWS SAM atribui à função para ajudar a garantir a exclusividade.
- A API no API Gateway que facilita a chamada da função Lambda. AWS CodeStar dá à API o nome awscodestar-my-sam-project--lambda, onde my-sam-project está o ID do projeto.

## Para explorar o repositório de código-fonte em CodeCommit

1. Com seu projeto aberto no AWS CodeStar console, na barra de navegação, escolha Repositório.
2. Escolha o link para seu CodeCommit repositório (**My-SAM-Project**) em Detalhes do repositório.
3. No CodeCommit console, na página Código, os arquivos de código-fonte do projeto são exibidos:
  - `buildspec.yml`, que CodePipeline instrui CodeBuild o uso durante a fase de construção, para empacotar o serviço web usando o AWS SAM.
  - `index.py`, que contém a lógica para a função do Lambda Essa função simplesmente gera a string `Hello World` e um carimbo de data e hora no formato ISO.
  - `README.md`, que contém informações gerais sobre o repositório.
  - `template-configuration.json`, que contém o ARN do projeto com espaços reservados usados para marcar recursos com o ID do projeto.
  - `template.yml`, que o AWS SAM usa para empacotar o serviço web e criar a API no API Gateway.



Como visualizar o conteúdo de um arquivo, escolha-o na lista.

Para obter mais informações sobre como usar o CodeCommit console, consulte o [Guia AWS CodeCommit do usuário](#).

Para explorar o pipeline em CodePipeline

1. Para exibir informações sobre o pipeline, com seu projeto aberto no console do AWS CodeStar na barra de navegação lateral, selecione Pipeline e verá que o pipeline contém:
  - Uma fase Source (Origem) para obter o código-fonte do CodeCommit.
  - A fase Build (Compilar) para compilar o código-fonte com o CodeBuild.
  - Um estágio de implantação para implantar o código-fonte e os AWS recursos criados com o AWS SAM.

2. Para ver mais informações sobre o pipeline, em Detalhes do pipeline, escolha seu pipeline para abri-lo no CodePipeline console.

Para obter informações sobre como usar o CodePipeline console, consulte o [Guia AWS CodePipeline do usuário](#).

Para explorar a atividade do projeto e os recursos de AWS serviço na página Visão geral

1. Abra seu projeto no AWS CodeStar console e, na barra de navegação, escolha Visão geral.
2. Analise as listas Atividade do projeto e Recursos do projeto.

Para explorar a função no Lambda

1. Com seu projeto aberto no AWS CodeStar console, na barra de navegação lateral, escolha Visão geral.
2. Em Recursos do projeto, na coluna ARN, escolha o link da função do Lambda.

O código da função é exibido no console do Lambda.

Para obter informações sobre o uso do console Lambda, consulte o [Manual do desenvolvedor do AWS Lambda](#).

Para explorar o API no API Gateway

1. Com seu projeto aberto no AWS CodeStar console, na barra de navegação lateral, escolha Visão geral.
2. Em Recursos do projeto, na coluna ARN, escolha o link da API do Amazon API Gateway.

Os recursos da API são exibidos no console do API Gateway.

Para obter informações sobre o uso do console API Gateway, consulte o [Guia do desenvolvedor](#).

## Etapa 3: Testar o serviço da web

Nesta etapa, você testa o serviço Web que AWS CodeStar acabou de ser criado e implantado.

1. Com o seu projeto ainda aberto da etapa anterior, na barra de navegação, selecione Pipeline.

2. No bloco Implantação contínua, verifique se Bem-sucedido é exibido para as fases Origem, Criar e Implantar antes de continuar. Isso pode demorar vários minutos.

 Note

Se Failed (Com falha) for exibido para qualquer uma das fases, consulte o seguinte para obter ajuda na solução do problema:

- No estágio Origem, consulte [Solucionar problemas do AWS CodeCommit](#) no Guia do usuário do AWS CodeCommit .
- No estágio Criar, consulte [Solucionar problemas do AWS CodeBuild](#) no Guia do usuário do AWS CodeBuild .
- No estágio Implantar, consulte [Solucionar problemas do AWS CloudFormation](#) no Guia do usuário do AWS CloudFormation .
- Para outros problemas, consulte [Solução de problemas AWS CodeStar](#).

3. Escolha Novo aplicativo

Na nova guia que é aberta no navegador da web, o serviço da web exibe a seguinte saída de resposta:

```
{"output": "Hello World", "timestamp": "2017-08-30T15:53:42.682839"}
```

## Etapa 4: Configurar sua estação de trabalho local para editar o código do projeto

Nesta etapa, você configura a estação de trabalho local para editar o código-fonte no projeto do AWS CodeStar . Sua estação de trabalho local pode ser um computador físico ou virtual que execute macOS, Windows ou Linux.

1. Com o projeto ainda aberto da etapa anterior:
  - Na barra de navegação, escolha IDE e, em seguida, expanda Acesse o código do seu projeto.

- Escolha Exibir instruções abaixo da interface da linha de comando.

Se você tiver o Visual Studio ou o Eclipse instalado, escolha Exibir as instruções sob Visual Studio, ou Eclipse, siga as instruções e, em seguida, vá para [Etapa 5: Adicionar lógica ao serviço da web](#).

2. Siga as instruções para concluir as tarefas a seguir:
  - a. Configure o Git na sua estação de trabalho local.
  - b. Use o console do IAM para gerar credenciais do Git para o seu usuário do IAM.
  - c. Clone o CodeCommit repositório do projeto em sua estação de trabalho local.
3. No painel de navegação esquerdo, escolha Projeto para retornar à visão geral do projeto.

## Etapa 5: Adicionar lógica ao serviço da web

Nesta etapa, você usa a estação de trabalho local para adicionar lógica ao serviço da web. Especificamente, você adiciona uma função do Lambda e, em seguida, a conecta à API no API Gateway.

1. Na estação de trabalho local, vá até o diretório que contém o repositório de código-fonte clonado.
2. Nesse diretório, crie um arquivo chamado `hello.py`. Adicione o seguinte código e salve-o em seguida:

```
import json

def handler(event, context):
    data = {
        'output': 'Hello ' + event["pathParameters"]["name"]
    }
    return {
        'statusCode': 200,
        'body': json.dumps(data),
        'headers': {'Content-Type': 'application/json'}
    }
```

O código anterior gera a string Hello e a string que o chamador enviar para a função

3. No mesmo diretório, abra o arquivo `template.yml`. Adicione o seguinte código ao final do arquivo e salve-o em seguida:

```
Hello:
  Type: AWS::Serverless::Function
  Properties:
    FunctionName: !Sub 'awscodestar-${ProjectId}-lambda-Hello'
    Handler: hello.handler
    Runtime: python3.7
    Role:
      Fn::GetAtt:
        - LambdaExecutionRole
        - Arn
    Events:
      GetEvent:
        Type: Api
        Properties:
          Path: /hello/{name}
          Method: get
```

AWS O SAM usa esse código para criar uma função no Lambda, adicionar um novo método e caminho à API no API Gateway e, em seguida, conectar esse método e o caminho à nova função.

#### Note

O recuo do código anterior é importante. Se você não adicionar o código exatamente como for mostrado, o projeto poderá não ser criado corretamente.

4. Execute `git add .` para adicionar as alterações do arquivo à área de preparação do repositório clonado. Não se esqueça do ponto (`.`), que adiciona todos os arquivos alterados.

#### Note

Se você estiver usando o Visual Studio ou o Eclipse em vez da linha de comando, as instruções para o uso do Git podem ser diferentes. Consulte a documentação do Eclipse ou do Visual Studio.

5. Executar `git commit -m "Added hello.py and updated template.yaml."` para confirmar os arquivos preparados no repositório clonado
6. Execute `git push` para enviar a confirmação para o repositório remoto por push.

**Note**

Você pode ser solicitado a fornecer as credenciais de login geradas anteriormente. Para que não sejam solicitados toda vez que você interagir com o repositório remoto, considere instalar e configurar um gerenciador de credenciais do Git. Por exemplo, no macOS ou no Linux, você pode executar `git config credential.helper 'cache --timeout 900'` no terminal para serem solicitados a cada 15 minutos. Ou você pode executar `git config credential.helper 'store --file ~/.git-credentials'` para nunca serem solicitados novamente. O Git armazena suas credenciais em texto nítido em um arquivo sem formatação em seu diretório inicial. Para obter mais informações, consulte [Ferramentas do GiT – armazenamento de credenciais](#) no site do Git.

Depois de AWS CodeStar detectar o push, ele instrui o AWS SAM CodePipeline a reconstruir CodeBuild e reimplantar o serviço web. Você pode acompanhar o progresso da implantação na página Pipeline.

AWS O SAM dá à nova função o nome `awscodestar-my-sam-project-Lambda-hello-RANDOM_ID`, onde:

- `my-sam-project` é o ID do projeto.
- `Hello` é o ID da função conforme especificado no arquivo `template.yaml`.
- **`RANDOM_ID`** é um ID aleatório que o AWS SAM atribui à função para fins de exclusividade.

## Etapa 6: Testar o serviço da web aprimorado

Nesta etapa, você testa o serviço Web aprimorado que AWS CodeStar foi criado e implantado, com base na lógica que você adicionou na etapa anterior.

1. Com seu projeto ainda aberto no AWS CodeStar console, na barra de navegação, escolha Pipeline.
2. Assegure-se de ter executado o pipeline novamente e de que Bem-sucedido apareça nos estágios de Origem, Criação, e Implantação antes de continuar. Isso pode demorar vários minutos.

**Note**

Se Failed (Com falha) for exibido para qualquer uma das fases, consulte o seguinte para obter ajuda na solução do problema:

- No estágio Origem, consulte [Solucionar problemas do AWS CodeCommit](#) no Guia do usuário do AWS CodeCommit .
- No estágio Criar, consulte [Solucionar problemas do AWS CodeBuild](#) no Guia do usuário do AWS CodeBuild .
- No estágio Implantar, consulte [Solucionar problemas do AWS CloudFormation](#) no Guia do usuário do AWS CloudFormation .
- Para outros problemas, consulte [Solução de problemas AWS CodeStar](#).

### 3. Escolha Novo aplicativo

Na nova guia que é aberta no navegador da web, o serviço da web exibe a seguinte saída de resposta:

```
{"output": "Hello World", "timestamp": "2017-08-30T15:53:42.682839"}
```

4. Na caixa de endereço da guia, adicione o caminho **/hello/** e seu primeiro nome ao final da URL (por exemplo, `https://API_ID.execute-api.REGION_ID.amazonaws.com/Prod/hello/YOUR_FIRST_NAME`) e, em seguida, pressione Enter.

Se o seu nome for Mary, o serviço da web exibirá a saída de resposta a seguir:

```
{"output": "Hello Mary"}
```

## Etapa 7: Adicionar um teste de unidade ao serviço da web

Nesta etapa, você usa sua estação de trabalho local para adicionar um teste AWS CodeStar executado no serviço Web. Esse teste substitui os testes manuais que você fez anteriormente.

1. Na estação de trabalho local, vá até o diretório que contém o repositório de código-fonte clonado.
2. Nesse diretório, crie um arquivo chamado `hello_test.py`. Adicione o seguinte código e salve-o em seguida.

```
from hello import handler

def test_hello_handler():

    event = {
        'pathParameters': {
            'name': 'testname'
        }
    }

    context = {}

    expected = {
        'body': '{"output": "Hello testname"}',
        'headers': {
            'Content-Type': 'application/json'
        },
        'statusCode': 200
    }

    assert handler(event, context) == expected
```

Esse teste verifica se o resultado da função do Lambda está no formato esperado. Se a resposta for sim, o teste será bem-sucedido. Caso contrário, o teste falhará.

3. No mesmo diretório, abra o arquivo `buildspec.yml`. Substitua o conteúdo do arquivo pelo seguinte código e, em seguida, salve o arquivo.

```
version: 0.2

phases:
  install:
    runtime-versions:
      python: 3.7

    commands:
      - pip install pytest
```

```
# Upgrade AWS CLI to the latest version
- pip install --upgrade awscli

pre_build:
  commands:
    - pytest

build:
  commands:
    # Use AWS SAM to package the application by using AWS CloudFormation
    - aws cloudformation package --template template.yml --s3-bucket
    $S3_BUCKET --output-template template-export.yml

    # Do not remove this statement. This command is required for AWS CodeStar
    projects.
    # Update the AWS Partition, AWS Region, account ID and project ID in the
    project ARN on template-configuration.json file so AWS CloudFormation can tag
    project resources.
    - sed -i.bak 's/\${PARTITION}\$/'${PARTITION}'/g;s/\${AWS_REGION}
    \$/'${AWS_REGION}'/g;s/\${ACCOUNT_ID}\$/'${ACCOUNT_ID}'/g;s/\${PROJECT_ID}\
    $/'${PROJECT_ID}'/g' template-configuration.json

artifacts:
  type: zip
  files:
    - template-export.yml
    - template-configuration.json
```

Essa especificação de compilação instrui CodeBuild a instalação do pytest, a estrutura de teste do Python, em seu ambiente de compilação. CodeBuild usa pytest para executar o teste de unidade. O resto da especificação de criação é o mesmo de antes.

#### 4. Use o Git para enviar essas alterações para o repositório remoto.

```
git add .

git commit -m "Added hello_test.py and updated buildspec.yml."

git push
```

## Etapa 8: Visualizar os resultados do teste de unidade

Nesta etapa, você verá se o teste de unidade foi bem-sucedido ou não.

1. Com seu projeto ainda aberto no AWS CodeStar console, na barra de navegação, escolha Pipeline.
2. Certifique-se de que o pipeline tenha funcionado novamente antes de continuar. Isso pode demorar vários minutos.

Se o teste de unidade foi bem-sucedido, Succeeded (Bem-sucedido) será exibido para a fase Build (Criar).

3. Para ver os detalhes do resultado do teste unitário, no estágio de criação, escolha o CodeBuildlink.
4. No CodeBuild console, na my-sam-project página Build Project:, em Histórico de compilação, escolha o link na coluna Build run da tabela.
5. Na **BUILD\_ID** página my-sam-project:, em Criar registros, escolha o link Exibir registro inteiro.
6. No console do Amazon CloudWatch Logs, procure na saída do log um resultado de teste semelhante ao seguinte. No resultado do teste a seguir, o teste foi bem-sucedido:

```
...
===== test session starts =====
platform linux2 -- Python 2.7.12, pytest-3.2.1, py-1.4.34, pluggy-0.4.0
rootdir: /codebuild/output/src123456789/src, inifile:
collected 1 item

hello_test.py .

===== 1 passed in 0.01 seconds =====
...
```

Se o teste falhou, deve haver detalhes na saída do log para ajudá-lo a solucionar a falha.

## Etapa 9: Limpeza

Nesta etapa, você limpa o projeto para evitar custos contínuos desse projeto.

Se quiser continuar usando esse projeto, você pode pular esta etapa, mas sua AWS conta pode continuar sendo cobrada.

1. Com seu projeto ainda aberto no AWS CodeStar console, na barra de navegação, escolha Configurações.
2. Em Detalhes do projeto, Selecione Excluir projeto.
3. Insira **delete**, mantenha a caixa Excluir recursos selecionada e escolha Excluir.

 Important

Se você desmarcar essa caixa, o registro do projeto será excluído AWS CodeStar, mas muitos dos AWS recursos do projeto serão retidos. Sua AWS conta pode continuar sendo cobrada.

Se ainda houver um bucket do Amazon S3 AWS CodeStar criado para esse projeto, siga estas etapas para excluí-lo. :

1. Abra o console do Amazon S3, em. <https://console.aws.amazon.com/s3/>
2. Na lista de buckets, escolha o ícone ao lado de aws-codestar- **REGION\_ID** - - **ACCOUNT\_ID** -- pipe, onde: my-sam-project
  - **REGION\_ID** é o ID da AWS região do projeto que você acabou de excluir.
  - **ACCOUNT\_ID** é o ID AWS da sua conta.
  - my-sam-project é o ID do projeto que você acabou de excluir.
3. Escolha Empty Bucket (Esvaziar bucket). Digite o nome do bucket e escolha Confirm (Confirmar).
4. Escolha Excluir bucket. Digite o nome do bucket e escolha Confirm (Confirmar).

## Próximas etapas

Agora que você concluiu este tutorial, sugerimos analisar os seguintes recursos:

- O [Começando com AWS CodeStar](#) tutorial usa um projeto que cria e implanta um aplicativo web baseado em Node.js executado em uma instância da Amazon. EC2
- [AWS CodeStar Modelos de projeto](#) descreve outros tipos de projeto que você pode criar.
- [Trabalhando com AWS CodeStar equipes](#) mostra como outras pessoas podem ajudá-lo a trabalhar em seus projetos.

# Tutorial: Crie um projeto AWS CodeStar com o AWS CLI

Este tutorial mostra como usar o AWS CLI para criar um AWS CodeStar projeto com um exemplo de código-fonte e um exemplo de modelo de conjunto de ferramentas. AWS CodeStar provisiona a AWS infraestrutura e os recursos do IAM especificados em um modelo AWS CloudFormation de cadeia de ferramentas. O projeto gerencia os recursos da cadeia de ferramentas para criar e implantar o código-fonte.

AWS CodeStar usa AWS CloudFormation para criar e implantar seu código de amostra. Esse código de exemplo cria um serviço web que está hospedado AWS Lambda e pode ser acessado por meio do Amazon API Gateway.

Pré-requisitos:

- Siga as etapas em [Configuração AWS CodeStar](#).
- Você deve ter criado um bucket de armazenamento do Amazon S3. Neste tutorial, você faz upload do código-fonte de exemplo e do modelo da cadeia de ferramentas para esse local.

## Note

Sua AWS conta pode ser cobrada pelos custos relacionados a este tutorial, incluindo AWS serviços usados pela AWS CodeStar. Para obter mais informações, consulte [Preços do AWS CodeStar](#).

Tópicos

- [Etapa 1: Fazer download e analisar o código-fonte de exemplo](#)
- [Etapa 2: Fazer download do modelo da cadeia de ferramentas de exemplo](#)
- [Etapa 3: Teste seu modelo de conjunto de ferramentas no AWS CloudFormation](#)
- [Etapa 4: Fazer upload do código-fonte e do modelo da cadeia de ferramentas](#)
- [Etapa 5: criar um projeto no AWS CodeStar](#)

## Etapa 1: Fazer download e analisar o código-fonte de exemplo

Para este tutorial, existe um arquivo zip disponível para download. Ele contém o código-fonte de exemplo de um [aplicativo de exemplo](#) Node.js na plataforma de computação Lambda. Quando o código-fonte é colocado no repositório, a pasta e os arquivos são exibidos conforme mostrado:

```
tests/  
app.js  
buildspec.yml  
index.js  
package.json  
README.md  
template.yml
```

Os seguintes elementos do projeto são representados no código-fonte de exemplo:

- `tests/`: testes de unidade configurados para o projeto CodeBuild do projeto. Essa pasta está incluída no código de exemplo, mas não é obrigatória para criar um projeto.
- `app.js`: código-fonte do aplicativo do projeto.
- `buildspec.yml`: as instruções de compilação para o estágio de compilação do recurso CodeBuild. Esse arquivo é obrigatório para um modelo de cadeia de ferramentas com um recurso CodeBuild .
- `package.json`: as informações de dependências do código-fonte do aplicativo.
- `README.md`: o arquivo readme do projeto incluído em todos os projetos AWS CodeStar . Esse arquivo está incluído no código de exemplo, mas não é obrigatório para criar um projeto.
- `template.yml`: o arquivo de modelo de infraestrutura ou arquivo de modelo SAM incluído em todos os AWS CodeStar projetos. Isso é diferente de `template.yml` da cadeia de ferramentas que você enviará mais adiante neste tutorial. Esse arquivo está incluído no código de exemplo, mas não é obrigatório para criar um projeto.

## Etapa 2: Fazer download do modelo da cadeia de ferramentas de exemplo

O modelo de conjunto de ferramentas de amostra fornecido para este tutorial cria um repositório (CodeCommit), um pipeline (CodePipeline) e um container de construção (CodeBuild) e é usado AWS CloudFormation para implantar seu código-fonte em uma plataforma Lambda. Além desses recursos, também há funções do IAM que você pode usar para definir o escopo das permissões

do seu ambiente de execução, um bucket do Amazon S3 que CodePipeline usa para armazenar seus artefatos de implantação e uma regra de CloudWatch eventos que é usada para acionar implantações de pipeline quando você envia código para seu repositório. Para se alinhar com [melhores práticas do IAM da AWS](#), restrinja as políticas das funções da cadeia de ferramentas definidas neste exemplo.

Baixe e descompacte o AWS CloudFormation modelo de amostra no formato [YAML](#).

Quando você executar o comando create-project posteriormente no tutorial, este modelo criará os recursos personalizados da cadeia de ferramentas no AWS CloudFormation. Para obter mais informações sobre os recursos criados neste tutorial, consulte os seguintes tópicos no Guia do usuário do AWS CloudFormation :

- O [AWS::CodeCommit::Repository](#) AWS CloudFormation recurso cria um CodeCommit repositório.
- O [AWS::CodeBuild::Project](#) AWS CloudFormation recurso cria um projeto de CodeBuild construção.
- O [AWS::CodeDeploy::Application](#) AWS CloudFormation recurso cria um CodeDeploy aplicativo.
- O [AWS::CodePipeline::Pipeline](#) AWS CloudFormation recurso cria um CodePipeline pipeline.
- O [AWS::S3::Bucket](#) AWS CloudFormation recurso cria o repositório de artefatos do seu pipeline.
- O [AWS::S3::BucketPolicy](#) AWS CloudFormation recurso cria a política de compartimento de artefatos para o compartimento de artefatos do seu pipeline.
- O [AWS::IAM::Role](#) AWS CloudFormation recurso cria a função de trabalhador CodeBuild do IAM que dá AWS CodeStar permissões para gerenciar seu projeto de CodeBuild compilação.
- O [AWS::IAM::Role](#) AWS CloudFormation recurso cria a função de trabalhador CodePipeline do IAM que dá AWS CodeStar permissões para criar seu pipeline.
- O [AWS::IAM::Role](#) AWS CloudFormation recurso cria a função de trabalhador AWS CloudFormation do IAM que dá AWS CodeStar permissões para criar sua pilha de recursos.
- O [AWS::IAM::Role](#) AWS CloudFormation recurso cria a função de trabalhador AWS CloudFormation do IAM que dá AWS CodeStar permissões para criar sua pilha de recursos.
- O [AWS::IAM::Role](#) AWS CloudFormation recurso cria a função de trabalhador AWS CloudFormation do IAM que dá AWS CodeStar permissões para criar sua pilha de recursos.
- O [AWS::Events::Rule](#) AWS CloudFormation recurso cria a regra de CloudWatch eventos que monitora seu repositório em busca de eventos push.
- O [AWS::IAM::Role](#) AWS CloudFormation recurso cria a função CloudWatch Events IAM.

## Etapa 3: Teste seu modelo de conjunto de ferramentas no AWS CloudFormation

Para fazer upload do modelo da cadeia de ferramentas, você pode testar o modelo da cadeia de ferramentas no AWS CloudFormation e solucionar todos os erros.

1. Salve o modelo atualizado no computador local e abra o AWS CloudFormation console. Selecione Create Stack (Criar pilha). Seus novos recursos devem ser exibidos na lista.
2. Veja a pilha em busca de eventuais erros de criação.
3. Depois que o teste for concluído, exclua a pilha.

### Note

Certifique-se de excluir sua pilha e todos os recursos criados em AWS CloudFormation. Caso contrário, ao criar o projeto, você poderá encontrar erros para nomes de recursos já em uso.

## Etapa 4: Fazer upload do código-fonte e do modelo da cadeia de ferramentas

Para criar um AWS CodeStar projeto, você deve primeiro empacotar seu código-fonte em um arquivo.zip e colocá-lo no Amazon S3. AWS CodeStar inicializa seu repositório com esses conteúdos. Você especifica esse local no arquivo de entrada ao executar o comando para criar o projeto na AWS CLI.

Você também deve fazer upload do arquivo `toolchain.yml` e colocá-lo no Amazon S3. Você especifica esse local em seu arquivo de entrada ao executar o comando para criar seu projeto no AWS CLI

Para fazer upload do código-fonte e do modelo da cadeia de ferramentas

1. A estrutura do arquivo de exemplo a seguir mostra os arquivos de origem e o modelo da cadeia de ferramentas prontos para serem compactados e enviados. O código de exemplo inclui o arquivo `template.yml`. Lembre-se de que esse arquivo é diferente do arquivo `toolchain.yml`.

```
ls
```

```
src toolchain.yml

ls src/
README.md    app.js        buildspec.yml  index.js      package.json
template.yml  tests
```

2. Crie o .zip dos arquivos de código-fonte.

```
cd src; zip -r "../src.zip" *; cd ../
```

3. Use o comando cp e inclua os arquivos como parâmetros.

Os comandos a seguir fazem upload do arquivo .zip e toolchain.yml para o Amazon S3.

```
aws s3 cp src.zip s3://MyBucket/src.zip
aws s3 cp toolchain.yml s3://MyBucket/toolchain.yml
```

Para configurar o bucket do Amazon S3 a fim de compartilhar o código-fonte

- Como você está armazenando seu código-fonte e sua cadeia de ferramentas no Amazon S3, você pode usar as políticas e ACLs objetos do bucket do Amazon S3 para garantir que outros usuários AWS ou contas do IAM possam criar projetos a partir de suas amostras. AWS CodeStar garante que qualquer usuário que crie um projeto personalizado tenha acesso ao conjunto de ferramentas e à fonte que deseja usar.

Para permitir que qualquer pessoa use a amostra, execute os seguintes comandos:

```
aws s3api put-object-acl --bucket MyBucket --key toolchain.yml --acl public-read
aws s3api put-object-acl --bucket MyBucket --key src.zip --acl public-read
```

## Etapa 5: criar um projeto no AWS CodeStar

Use estas etapas para criar o projeto.

### Important

Certifique-se de configurar a AWS região preferida em AWS CLI. Seu projeto é criado na AWS Região configurada no AWS CLI.

## 1. Execute o comando create-project e inclua o parâmetro --generate-cli-skeleton:

```
aws codestar create-project --generate-cli-skeleton
```

Os dados formatados JSON aparecem na saída. Copie os dados para um arquivo (por exemplo, *input.json*) em um local no computador local ou na instância em que o AWS CLI esteja instalado. Modifique os dados copiados da maneira a seguir e salve os resultados. Esse arquivo de entrada é configurado para um projeto chamado MyProject com um nome de bucket myBucket.

- Certifique-se de fornecer o parâmetro roleArn. Para modelos personalizados, como o exemplo de modelo deste tutorial, é necessário fornecer uma função. Essa função deve ter permissões para criar todos os recursos especificados em [Etapa 2: Fazer download do modelo da cadeia de ferramentas de exemplo](#).
- Verifique se você forneceu o parâmetro ProjectId em stackParameters. O modelo de exemplo fornecido para este tutorial exige esse parâmetro.

```
{
  "name": "MyProject",
  "id": "myproject",
  "description": "Sample project created with the CLI",
  "sourceCode": [
    {
      "source": {
        "s3": {
          "bucketName": "MyBucket",
          "bucketKey": "src.zip"
        }
      },
      "destination": {
        "codeCommit": {
          "name": "myproject"
        }
      }
    }
  ],
  "toolchain": {
    "source": {
      "s3": {
```

```

        "bucketName": "MyBucket",
        "bucketKey": "toolchain.yml"
    }
},
"roleArn": "role_ARN",
"stackParameters": {
    "ProjectId": "myproject"
}
}
}

```

2. Alterne para o diretório que contenha o arquivo que você acabou de salvar e execute o comando `create-project` novamente. Inclua o parâmetro `--cli-input-json`.

```
aws codestar create-project --cli-input-json file://input.json
```

3. Se bem-sucedido, dados semelhantes aos seguintes serão exibidos na saída:

```

{
  "id": "project-ID",
  "arn": "arn"
}

```

- A saída contém informações sobre o novo projeto:
    - O valor `id` representa o ID do projeto.
    - O valor `arn` representa o ARN do projeto.
4. Use o comando `describe-project` para verificar o status da criação do projeto. Inclua o parâmetro `--id`.

```
aws codestar describe-project --id <project_ID>
```

Dados semelhantes aos seguintes são exibidos na saída:

```

{
  "name": "MyProject",
  "id": "myproject",
  "arn": "arn:aws:codestar:us-east-1:account_ID:project/myproject",
  "description": "",
  "createdTimeStamp": 1539700079.472,

```

```
"stackId": "arn:aws:cloudformation:us-east-1:account_ID:stack/awscodestar-myproject/stack-ID",
  "status": {
    "state": "CreateInProgress"
  }
}
```

- A saída contém informações sobre o novo projeto:
  - O valor `id` representa o ID do projeto exclusivo.
  - O valor `state` representa o status da criação do projeto, como `CreateInProgress` ou `CreateComplete`.

Embora o projeto esteja sendo criado, você pode [adicionar membros à equipe](#) ou [configurar o acesso](#) ao repositório do projeto na linha de comando ou de seu IDE favorito.

## Tutorial: Crie um projeto Alexa Skill no AWS CodeStar

AWS CodeStar é um serviço de desenvolvimento baseado em nuvem AWS que fornece as ferramentas de que você precisa para desenvolver, criar e implantar aplicativos rapidamente. Com AWS CodeStar, você pode configurar todo o seu conjunto de ferramentas de entrega contínua em minutos, permitindo que você comece a liberar o código mais rapidamente. Os modelos de projeto de habilidades da Alexa AWS CodeStar permitem que você crie uma habilidade simples da Alexa Hello World a partir da sua AWS conta com apenas alguns cliques. Os modelos também criam um pipeline de implantação básico para você começar a usar um fluxo de integração contínua (CI) para o desenvolvimento de habilidades.

Os principais benefícios de criar habilidades da Alexa a partir do Alexa AWS CodeStar são que você pode começar a desenvolver habilidades AWS e conectar sua conta de desenvolvedor da Amazon ao projeto para implantar habilidades no estágio de desenvolvimento diretamente de AWS. Você também pode obter um pipeline de implantação (CI) pronto para uso com um repositório com todo o código-fonte para o projeto. Você pode configurar esse repositório com seu IDE preferencial para criar habilidades com as ferramentas que já conhece.

### Pré-requisitos

- Crie uma conta de desenvolvedor da Amazon acessando <https://developer.amazon.com>. O cadastro é gratuito. Esta conta é proprietária de suas habilidades do Alexa.

- Se você não tiver uma AWS conta, use o procedimento a seguir para criar uma.

Para se inscrever em AWS

1. Abra <https://aws.amazon.com/>e escolha Criar uma AWS conta.

 Note

Se você já fez login no AWS Management Console usando Usuário raiz da conta da AWS credenciais, escolha Entrar em uma conta diferente. Se você fez login no console usando as credenciais do IAM, selecione Fazer login usando credenciais da Usuário raiz da conta da AWS . Em seguida, escolha Criar uma nova conta do AWS .

2. Siga as instruções online.

 Important

Depois de criar o projeto de habilidade do Alexa, faça todas as edições somente no repositório do projeto. Recomendamos que você não edite essa habilidade diretamente usando nenhuma outra ferramenta do Alexa Skills Kit (ASK), como a CLI do ASK ou o console do desenvolvedor do ASK. Essas ferramentas não estão integradas ao repositório do projeto. Usá-las fará com que a habilidade e o código de repositório percam a sincronia.

## Etapa 1: Criar o projeto e conectar sua conta de desenvolvedor da Amazon

Neste tutorial, você cria uma habilidade usando o Node.js em execução no AWS Lambda. A maioria das etapas são as mesmas para outros idiomas, embora o nome da habilidade seja diferente. Consulte o arquivo README.md no repositório do projeto para obter detalhes específicos do modelo de projeto que você escolher.

1. Faça login no e AWS Management Console, em seguida, abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.
2. Escolha a AWS região em que você deseja criar o projeto e seus recursos. O tempo de execução de habilidades da Alexa está disponível nas seguintes AWS regiões:
  - Ásia-Pacífico (Tóquio)
  - UE (Irlanda)

- Leste dos EUA (N. da Virgínia)
  - Oeste dos EUA (Oregon)
3. Escolha Criar projeto.
  4. Na página Choose a project template (Escolher um modelo de projeto):
    - a. Em Categoria de aplicativo, selecione Alexa Skill.
    - b. Em Linguagem de programação, selecione Node.js.
  5. Escolha a caixa que contém suas seleções.
  6. Em Project name (Nome do projeto), digite o nome do projeto (por exemplo, **My Alexa Skill**). Se você usar um nome diferente, não se esqueça de usá-lo ao longo deste tutorial. AWS CodeStar escolhe um identificador relacionado para esse projeto para o ID do projeto (por exemplo, my-alexas-skill). Se vir outro ID de projeto, certifique-se de usá-lo durante todo este tutorial.
  7. Escolha AWS CodeCommit para o repositório neste tutorial e não altere o valor do nome do repositório.
  8. Selecione Connect Amazon developer account (Conectar conta de desenvolvedor da Amazon) para vincular sua conta de desenvolvedor da Amazon para hospedar a habilidade. Se você não tiver uma conta de desenvolvedor da Amazon, crie uma conta e conclua primeiro o registro com a [Amazon Developers](#).
  9. Faça login com suas credenciais de desenvolvedor da Amazon. Escolha Permitir e, em seguida, escolha Confirmar para concluir a conexão.
  10. Se você tiver vários fornecedores IDs associados à sua conta de desenvolvedor da Amazon, escolha aquele que deseja usar para este projeto. Certifique-se de usar uma conta com a função de administrador ou desenvolvedor atribuída.
  11. Escolha Próximo.
  12. (Opcional) Se for a primeira vez que você usa AWS CodeStar nesta AWS região, insira o nome de exibição e o endereço de e-mail que você deseja usar AWS CodeStar para seu usuário do IAM. Escolha Próximo.
  13. Aguarde enquanto AWS CodeStar cria o projeto. Isso pode demorar vários minutos. Não continue até ver o banner provisionado do projeto.

## Etapa 2: Testar sua habilidade no Simulador do Alexa

Na primeira etapa, AWS CodeStar criei uma habilidade para você e a implantei no estágio de desenvolvimento de habilidades da Alexa. Em seguida, você testará a habilidade no Simulador do Alexa.

1. Em seu projeto no AWS CodeStar console, escolha Exibir aplicativo. Uma nova guia será aberta no Simulador do Alexa.
2. Faça login com suas credenciais de desenvolvedor da Amazon da conta com a qual você se conectou ao seu projeto na Etapa 1.
3. Em Test (Testar), selecione Development (Desenvolvimento) para habilitar os testes.
4. Digite `ask hello node hello`. O nome de invocação padrão para a habilidade é `hello node`.
5. Sua habilidade deve responder `Hello World!`.

Quando a habilidade estiver habilitada no Simulador do Alexa, você também poderá invocá-la em um dispositivo habilitado para o Alexa que estiver registrado em sua conta de desenvolvedor da Amazon. Para testar a habilidade em um dispositivo, diga Alexa, peça ao nó "hello" para dizer "hello".

Para obter mais informações sobre o Simulador do Alexa, consulte [Testar sua habilidade no console do desenvolvedor](#).

## Etapa 3: Explorar os recursos do projeto

Como parte da criação do projeto, AWS CodeStar também criou AWS recursos em seu nome. Esses recursos incluem o uso de um repositório de projetos CodeCommit, o uso de um pipeline de implantação CodePipeline e uma AWS Lambda função. Você pode acessar esses recursos na barra de navegação. Por exemplo, escolher Repositório mostra detalhes sobre o CodeCommit repositório. Você pode ver o status de implantação do pipeline na página Pipeline. Você pode ver uma lista completa dos AWS recursos criados como parte do seu projeto escolhendo Visão geral na barra de navegação. Essa lista inclui links para cada recurso.

## Etapa 4: Fazer uma alteração na resposta da habilidade

Nesta etapa, faça uma pequena alteração na resposta de sua habilidade para compreender o ciclo de iteração.

1. Na barra de navegação, escolha Repositório. Escolha o link em Nome do repositório e o repositório do seu projeto será aberto em uma nova guia ou janela. Esse repositório contém a especificação de compilação (buildspec.yml), a pilha de aplicativos do AWS CloudFormation (template.yml), o arquivo readme e o código-fonte de sua habilidade no [formato do pacote da habilidade \(estrutura do projeto\)](#).
2. Navegue até o arquivo lambda > personalizar > index.js (no caso do Node.js.). Esse arquivo contém o código de manuseio da solicitação, que usa o [ASK SDK](#).
3. Selecione Editar.
4. Substitua a string Hello World! na linha 24 pela string Hello. How are you?.
5. Role para baixo até o final do arquivo. Insira o nome do autor e o endereço de e-mail, bem como uma mensagem opcional de confirmação.
6. Selecione Commit changes (Confirmar alterações) para confirmar as alterações ao repositório.
7. Retorne ao projeto em AWS CodeStar e confira a página do Pipeline. Você agora deverá ver a implantação do pipeline.
8. Quando a implantação do pipeline for concluída, teste a habilidade de novamente no Simulador do Alexa. A habilidade deve responder Hello. How are you?.

## Etapa 5: Configurar sua estação de trabalho local para se conectar ao repositório do projeto

Anteriormente, você fez uma pequena alteração no código-fonte diretamente do CodeCommit console. Nesta etapa, você configura o repositório do projeto com sua estação de trabalho local, para que possa editar e gerenciar o código na linha de comando ou no seu IDE favorito. As etapas a seguir explicam como configurar as ferramentas de linha de comando.

1. Navegue até o painel do projeto em AWS CodeStar, se necessário.
2. Na barra de navegação, escolha IDE.
3. Em Acessar o código do seu projeto, exibir as instruções abaixo da interface da linha de comando.
4. Siga as instruções para concluir as tarefas a seguir:
  - a. Instale o Git em sua estação de trabalho local a partir de um site, como [Git Downloads](#).
  - b. Instale a AWS CLI. Para obter informações, consulte [Instalando a interface de linha de AWS comando](#).
  - c. Configure a AWS CLI com sua chave de acesso de usuário e chave secreta do IAM. Para obter informações, consulte [Configurando a AWS CLI](#).

- d. Clone o CodeCommit repositório do projeto em sua estação de trabalho local. Para obter mais informações, consulte [Connect to a CodeCommit Repository](#).

## Próximas etapas

Este tutorial mostrou como começar com uma habilidade básica. Para continuar sua jornada de desenvolvimento de habilidades, consulte os recursos a seguir.

- Entenda os fundamentos de uma habilidade assistindo [Como funcionam as habilidades da Alexa](#) e outros vídeos no canal Alexa Developers. YouTube
- Entenda os vários componentes da sua habilidade, examinando a documentação do [formato de pacote da habilidade](#), os [esquemas do manifesto da habilidade](#), e os [esquemas do modelo de interação](#).
- [Transforme sua ideia em uma habilidade revisando a documentação do Alexa Skills Kit e do ASK SDKs](#)

## Tutorial: Criar um projeto com um repositório GitHub de origem

Com AWS CodeStar, você pode configurar seu repositório para criar, revisar e mesclar pull requests com sua equipe de projeto.

Neste tutorial, você cria um projeto com um exemplo de código-fonte do aplicativo web em um GitHub repositório, um pipeline que implanta suas alterações e EC2 instâncias em que seu aplicativo está hospedado na nuvem. Depois que seu projeto for criado, este tutorial mostra como criar e GitHub mesclar uma pull request que faz uma alteração na página inicial do seu aplicativo web.

### Tópicos

- [Etapa 1: criar o projeto e criar seu GitHub repositório](#)
- [Etapa 2: exibir seu código-fonte](#)
- [Etapa 3: criar uma solicitação GitHub pull](#)

## Etapa 1: criar o projeto e criar seu GitHub repositório

Nesta etapa, use o console para criar seu projeto e criar uma conexão com seu novo GitHub repositório. Para acessar seu GitHub repositório, você cria um recurso de conexão que AWS

CodeStar usa para gerenciar a autorização com GitHub. Quando o projeto é criado, seus recursos adicionais são provisionados para você.

1. Faça login no e AWS Management Console, em seguida, abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.
2. Escolha a AWS região em que você deseja criar o projeto e seus recursos.
3. Na página AWS CodeStar, escolha Criar um projeto.
4. Na página Escolha um modelo de projeto, marque as caixas de seleção Aplicativo Web, Node.js e Amazon EC2. Em seguida, escolha entre os modelos disponíveis para esse conjunto de opções.

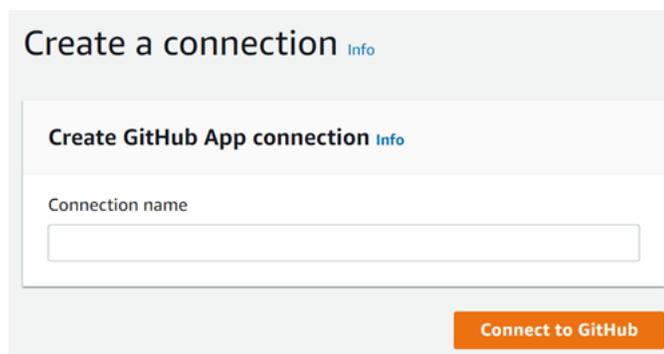
Para obter mais informações, consulte [AWS CodeStar Modelos de projeto](#).

5. Escolha Próximo.
6. Em Project name (Nome do projeto), digite o nome do projeto (por exemplo, **MyTeamProject**). Se você escolher outro nome, certifique-se de usá-lo durante todo este tutorial.
7. Em Repositório do projeto, escolha GitHub.
8. Se você escolher GitHub, precisará escolher ou criar um recurso de conexão. Se você tiver uma conexão existente, escolha-a no campo de pesquisa. Caso contrário, você criará uma nova conexão aqui. Escolha Connect to GitHub.

A página Criar uma conexão é exibida.

#### Note

Para criar uma conexão, você precisa ter uma GitHub conta. Se você estiver criando uma conexão para uma organização, você deve ser o proprietário da organização.



Create a connection [Info](#)

Create GitHub App connection [Info](#)

Connection name

[Connect to GitHub](#)

- a. Em Criar conexão com o GitHub aplicativo, em Nome da conexão, insira um nome para sua conexão. Escolha Connect to GitHub.

A GitHub página Connect to exibe e mostra o campo GitHub Apps.

- b. Em GitHub Aplicativos, escolha uma instalação de aplicativo ou escolha Instalar um novo aplicativo para criar um.

 Note

Você instala uma aplicação para todas as suas conexões com um provedor específico. Se você já instalou o GitHub aplicativo AWS Connector for, escolha-o e pule esta etapa.

- c. Na GitHub página Install AWS Connector for, escolha a conta na qual você deseja instalar o aplicativo.

 Note

Se você instalou a aplicação anteriormente, poderá escolher Configure (Configurar) para prosseguir para uma página de modificação para a instalação da aplicação ou usar o botão Back (Voltar) para retornar ao console.

- d. Se a página Confirmar senha para continuar for exibida, digite sua GitHub senha e escolha Entrar.
- e. Na GitHub página Install AWS Connector for, deixe os padrões e escolha Instalar.
- f. Na GitHub página Connect to, o ID de instalação da sua nova instalação aparece em GitHubApps.

Depois que a conexão for criada com sucesso, na página de CodeStar criação do projeto, a mensagem Pronto para conectar é exibida.

 Note

É possível visualizar sua conexão em Configurações no console do Developer Tools. Para obter mais informações, consulte o tópico de [Conceitos básicos sobre conexões](#).

Select a repository provider

CodeCommit

Use a new AWS CodeCommit repository for your project.



GitHub

Use a new GitHub source repository for your project (requires an existing GitHub account).



**The GitHub repository provider now uses CodeStar Connections**

To use a GitHub repository in CodeStar, create a connection. The connection will use GitHub Apps to access your repository. Use the following options to choose an existing connection or create a new one. [Learn more](#)

Connection

Choose an existing connection or create a new one and then return to this task.

or

 **Ready to connect**

Your Github connection is ready for use.

Repository owner

The owner of the new repository. This can be a personal GitHub account or a GitHub organization.

[Redacted]
▼

Repository name

The name of the new repository.

cs-dk-gh

Repository description

An optional description of the new repository.

Public

- g. Para Proprietário do repositório, escolha a GitHub organização ou sua GitHub conta pessoal.
- h. Em Nome do repositório, aceite o nome padrão do GitHub repositório ou insira um nome diferente.
- i. Escolha Público ou Privado.

**Note**

Se quiser usar AWS Cloud9 como seu ambiente de desenvolvimento, você deve escolher um repositório público.

- j. (Opcional) Em Descrição do repositório, insira uma descrição para o GitHub repositório.

9. Configure suas EC2 instâncias da Amazon na EC2 Configuração da Amazon se seu projeto estiver implantado em EC2 instâncias da Amazon e você quiser fazer alterações. Por exemplo, você pode escolher entre os tipos de instância disponíveis para o projeto.

Em Par de chaves, escolha o par de EC2 chaves da Amazon em que você criou [Etapa 4: criar um par de EC2 chaves da Amazon para AWS CodeStar projetos](#). Selecione **Eu reconheço que tenho acesso ao arquivo de chave privada**.

10. Escolha **Próximo**.
11. Analise os recursos e os detalhes da configuração.
12. Escolha **Avançar** ou **Criar projeto**. (A opção exibida depende do modelo do projeto.)

Aguarde alguns minutos enquanto seu projeto é criado.

13. Depois que seu projeto for criado, escolha **Exibir aplicativo** para visualizar seu aplicativo web.

## Etapa 2: exibir seu código-fonte

Nesta etapa, você visualiza seu código-fonte e as ferramentas que você pode usar para seu repositório de origem.

1. Na barra de navegação de seu projeto, escolha **Repositório**.

Para ver uma lista de confirmações em GitHub, escolha **Exibir confirmações**. Isso abre seu histórico de commits em GitHub.

Para ver os problemas, escolha a guia **Problemas** do seu projeto. Para criar um novo problema em GitHub, escolha **Criar GitHub problema**. Isso abre o formulário de problema do seu repositório em GitHub.

2. Na guia **Repositório**, escolha o link em **Nome do repositório** e o repositório do seu projeto será aberto em uma nova guia ou janela. Esse repositório contém o código-fonte do seu projeto.

## Etapa 3: criar uma solicitação GitHub pull

Nesta etapa, você faz uma pequena alteração no código-fonte e cria uma pull request.

1. Em GitHub, crie uma nova ramificação de recursos no seu repositório. Escolha o campo **suspensão da ramificação principal** e insira uma nova ramificação no campo chamado **feature-branch**. Escolha **Criar nova ramificação**. A ramificação é criada e verificada para você.

2. Dentro GitHub, faça uma mudança na `feature-branch` filial. Abra a pasta pública e abra o arquivo `index.html`.
3. No AWS CodeStar console, em Pull requests, para criar uma pull request GitHub, escolha Create pull request. Isso abre o formulário de pull request do seu repositório em GitHub. Em GitHub, escolha o ícone de lápis para editar o arquivo.

Depois de Congratulations!, adicione a string Well done, <name>! e substitua <name> pelo seu nome. Escolha Commit changes (Confirmar alterações). A alteração está comprometida com sua ramificação de atributos.

4. No AWS CodeStar console, escolha seu projeto. Escolha a guia Repositório. Em Solicitações pull, escolha Criar solicitação pull.

O formulário é aberto em GitHub. Deixe o galho principal no galho base. Em Comparar com, escolha sua ramificação de atributos. Exiba a diferença.

5. Em GitHub, escolha Criar pull request. Uma pull request chamada Update index.html é criada.
6. No AWS CodeStar console, visualize o novo pull request. Escolha Mesclar alterações para confirmar as alterações no repositório e mesclar a pull request com a ramificação principal do seu repositório.
7. Retorne ao projeto em AWS CodeStar e confira a página do Pipeline. Você agora deverá ver a implantação do pipeline.
8. Depois que seu projeto for criado, escolha Exibir aplicativo para visualizar seu aplicativo web.

# AWS CodeStar Modelos de projeto

AWS CodeStar os modelos de projeto permitem que você comece com um aplicativo de amostra e o implante usando AWS recursos criados para apoiar seu projeto de desenvolvimento. Quando você escolhe um modelo de AWS CodeStar projeto, o tipo de aplicativo, a linguagem de programação e a plataforma computacional são provisionados para você. Depois de criar projetos com aplicativos web, serviços web, habilidades Alexa e páginas estáticas da Web, você poderá substituir o aplicativo de exemplo pelos próprios.

Depois de AWS CodeStar criar seu projeto, você pode modificar os AWS recursos que dão suporte à entrega do seu aplicativo. AWS CodeStar trabalha com AWS CloudFormation para permitir que você use código para criar serviços de suporte e servidores/plataformas sem servidor na nuvem. AWS CloudFormation permite modelar toda a sua infraestrutura em um arquivo de texto.

## Tópicos

- [AWS CodeStar Arquivos e recursos do projeto](#)
- [Primeiros passos: escolher um modelo de projeto](#)
- [Como fazer alterações em seu AWS CodeStar projeto](#)

## AWS CodeStar Arquivos e recursos do projeto

Um AWS CodeStar projeto é uma combinação do código-fonte e dos recursos criados para implantar o código. A coleção de recursos que ajudam a compilar, lançar e implantar o código são chamados de recursos da cadeia de ferramentas. Na criação do projeto, um AWS CloudFormation modelo provisiona os recursos do seu conjunto de ferramentas em um pipeline integration/continuous deployment (CI/CD (contínuo)).

Você pode usar AWS CodeStar para criar projetos de duas maneiras, dependendo do seu nível de experiência com a criação AWS de recursos:

- Quando você usa o console para criar um projeto, AWS CodeStar cria seus recursos de conjunto de ferramentas, incluindo seu repositório, e preenche seu repositório com exemplos de código de aplicativo e arquivos de projeto. Use o console para configurar rapidamente projetos de amostra com base em um conjunto de opções de projeto pré-configurado.
- Ao usar a CLI para criar um projeto, você fornece o AWS CloudFormation modelo que cria os recursos da cadeia de ferramentas e o código-fonte do aplicativo. Use a CLI para permitir que você

crie seu projeto AWS CodeStar a partir do seu modelo e, em seguida, preencha seu repositório com seu código de amostra.

Um AWS CodeStar projeto fornece um único ponto de gerenciamento. Você pode usar o assistente Criar projeto no console para configurar um projeto de exemplo. Em seguida, você pode usá-lo como uma plataforma de colaboração para gerenciar permissões e recursos para a equipe. Para obter mais informações, consulte [O que é AWS CodeStar?](#). Quando você usa o console para criar um projeto, o código-fonte é fornecido como um código de exemplo e os recursos da cadeia de ferramentas CI/CD são criados para você

Ao criar um projeto no console, AWS CodeStar provisione os seguintes recursos:

- Um repositório de código em GitHub ou CodeCommit.
- No repositório do projeto, um arquivo README .md que dá detalhes dos arquivos e diretórios.
- No repositório do projeto, um arquivo `template.yml` que armazena a definição da pilha do tempo de execução do aplicativo. Você usa esse arquivo para adicionar ou modificar recursos do projeto que não são recursos do conjunto de ferramentas, como AWS recursos usados para notificações, suporte de banco de dados, monitoramento e rastreamento.
- AWS serviços e recursos criados em conexão com seu pipeline, como o bucket de artefatos do Amazon S3, Amazon CloudWatch Events e funções de serviço relacionadas.
- Um aplicativo de exemplo funcional com código-fonte completo e um endpoint HTTP público.
- Um recurso AWS computacional, com base no tipo de modelo do AWS CodeStar projeto:
  - Uma função Lambda.
  - Uma EC2 instância da Amazon.
  - Um AWS Elastic Beanstalk ambiente.
- A partir de 6 de dezembro de 2018 (horário do Pacífico):
  - Um limite de permissões que é uma política do IAM especializada para controlar o acesso aos recursos do projeto. Por padrão, o limite de permissões é anexado às funções no projeto de exemplo. Para obter mais informações, consulte [Limite de permissões do IAM para funções de operador](#).
  - Uma função AWS CloudFormation do IAM para criar recursos do projeto usando AWS CloudFormation isso inclui permissões para todos os recursos AWS CloudFormation compatíveis, incluindo funções do IAM.
  - Uma função do IAM para cadeia de ferramentas.

- Funções de execução do Lambda definidas na pilha de aplicativos, que você pode modificar.
- Antes de 6 de dezembro de 2018 (horário do Pacífico):
  - Uma função AWS CloudFormation do IAM para criar recursos do projeto com suporte para um conjunto limitado de AWS CloudFormation recursos.
  - Uma função do IAM para criar um CodePipeline recurso.
  - Uma função do IAM para criar um CodeBuild recurso.
  - Uma função do IAM para criar um CodeDeploy recurso, se aplicável ao seu tipo de projeto.
  - Uma função do IAM para criar o aplicativo EC2 web da Amazon, se aplicável ao seu tipo de projeto.
  - Uma função do IAM para criar um recurso de CloudWatch eventos.
  - Uma função de execução para o Lambda modificada dinamicamente para incluir um conjunto parcial de recursos.

O projeto inclui páginas de detalhes que mostram o status e contêm links para o gerenciamento da equipe, links para IDEs instruções de configuração do seu repositório e um histórico de confirmação das alterações do código-fonte no repositório. Você também pode selecionar ferramentas para se conectar com ferramentas de rastreamento de problemas externos, como o Jira.

## Primeiros passos: escolher um modelo de projeto

Ao escolher um AWS CodeStar projeto no console, você está escolhendo entre um conjunto de opções pré-configuradas com exemplos de código e recursos para começar rapidamente. Essas opções são chamadas de modelos de projeto. Cada modelo de AWS CodeStar projeto consiste em uma linguagem de programação, tipo de aplicativo e plataforma computacional. A combinação que você seleciona determina o modelo de projeto.

## Escolha uma plataforma de computação de modelo

Cada modelo configura um dos seguintes tipos de plataforma de computação:

- Ao escolher um AWS Elastic Beanstalk projeto, você implanta em um AWS Elastic Beanstalk ambiente nas instâncias do Amazon Elastic Compute Cloud na nuvem.
- Quando você escolhe um EC2 projeto da Amazon, AWS CodeStar cria EC2 instâncias Linux para hospedar seu aplicativo na nuvem. Os membros da sua equipe de projeto podem acessar as instâncias, e sua equipe usa o par de chaves que você fornece ao SSH em suas EC2 instâncias

da Amazon. AWS CodeStar também tem um SSH gerenciado que usa permissões de membros da equipe para gerenciar conexões de key pair.

- Quando você escolhe AWS Lambda, AWS CodeStar cria um ambiente sem servidor acessado por meio do Amazon API Gateway, sem instâncias ou servidores para manter.

## Escolha um tipo de aplicativo de modelo

Cada modelo configura um dos seguintes tipos de aplicativo:

- Serviço web

Um serviço web é usado para tarefas executadas em segundo plano, como chamadas APIs. Depois de AWS CodeStar criar seu projeto de serviço web de amostra, você pode escolher a URL do endpoint para ver a saída do Hello World, mas o uso principal desse tipo de aplicativo não é como interface de usuário (UI). Os modelos de AWS CodeStar projeto nessa categoria oferecem suporte ao desenvolvimento em Ruby, Java, ASP.NET, PHP, Node.js e muito mais.

- Aplicativo Web

Um aplicativo web tem uma interface do usuário. Depois de AWS CodeStar criar seu projeto de aplicativo web de amostra, você pode escolher o URL do endpoint para ver um aplicativo web interativo. Os modelos de AWS CodeStar projeto nessa categoria oferecem suporte ao desenvolvimento em Ruby, Java, ASP.NET, PHP, Node.js e muito mais.

- Página da web estática

Escolha esse modelo se você quiser um projeto para um site HTML. Os modelos de AWS CodeStar projeto nesta categoria oferecem suporte ao desenvolvimento em HTML5.

- Habilidade do Alexa

Selecione esse modelo se quiser um projeto para uma habilidade do Alexa com uma função do AWS Lambda . Quando você cria o projeto de habilidades, a AWS CodeStar retorna um Amazon Resource Name (ARN) que você pode usar como um endpoint de serviço. Para obter mais informações, consulte [Hospedar uma habilidade personalizada como uma função AWS Lambda](#).

**Note**

As funções do Lambda para as habilidades do Alexa são compatíveis somente nas regiões Leste dos EUA (Norte da Virgínia), Oeste dos EUA (Oregon), UE (Irlanda) e Ásia-Pacífico (Tóquio).

- Regra do Config

Escolha esse modelo se quiser um projeto para uma AWS Config regra que permita automatizar regras em todos AWS os recursos da sua conta. A função retorna um ARN que você pode usar como um endpoint de serviço para a regra.

## Escolha um modelo de linguagem de programação

Ao escolher um modelo de projeto, você seleciona uma linguagem de programação, como Ruby, Java, ASP.NET, PHP, Node.js e muito mais.

## Como fazer alterações em seu AWS CodeStar projeto

Você pode atualizar o projeto modificando:

- Código de exemplo e recursos de linguagem de programação para o aplicativo.
- Os recursos que compõem a infraestrutura na qual o aplicativo é armazenado e implantado (sistemas operacionais, aplicativos e serviços de suporte, parâmetros de implantação, e a plataforma de computação em nuvem). Você pode modificar recursos de aplicativos no arquivo `template.yml`. Este é o arquivo AWS CloudFormation que modela o ambiente de tempo de execução do aplicativo.

**Note**

Se você estiver trabalhando com um AWS CodeStar projeto do Alexa Skills, não poderá fazer alterações na habilidade fora do repositório de AWS CodeStar origem (CodeCommit ou GitHub). Se você editar a habilidade no portal do desenvolvedor do Alexa, talvez a alteração não esteja visível no repositório de origem e as duas versões ficarão fora de sincronia.

## Alterar o código-fonte do aplicativo e enviar alterações por push

Para modificar o código-fonte de exemplo, scripts e outros arquivos de origem do aplicativo, edite os arquivos no repositório de origem por:

- Usando o modo Editar em CodeCommit ou GitHub.
- Abrindo o projeto em um IDE, como AWS Cloud9.
- Clonar o repositório localmente e confirmar e enviar as alterações. Para ter mais informações, consulte [Etapa 4: Confirmar uma alteração](#).

## Alterar recursos do aplicativo com o arquivo `template.yml`

Em vez de modificar manualmente um recurso de infraestrutura, use-o AWS CloudFormation para modelar e implantar os recursos de tempo de execução do seu aplicativo.

Você pode modificar ou adicionar um recurso de aplicativo, como uma função do Lambda, na pilha de tempo de execução, editando o arquivo `template.yml` no repositório do projeto. Você pode adicionar qualquer recurso que esteja disponível como um recurso do AWS CloudFormation .

Para alterar o código ou as configurações de uma AWS Lambda função, consulte [Adicionar um recurso a um projeto](#).

Modifique o `template.yml` arquivo no repositório do seu projeto para adicionar o tipo de AWS CloudFormation recursos que são recursos do aplicativo. Quando você adiciona um recurso do aplicativo à `Resources` seção do `template.yml` arquivo AWS CloudFormation e AWS CodeStar cria o recurso para você. Para obter uma lista de AWS CloudFormation recursos e suas propriedades necessárias, consulte [Referência de tipos de AWS recursos](#). Para obter mais informações, consulte este exemplo em [Etapa 1: editar a função do CloudFormation trabalhador no IAM](#).

AWS CodeStar permite que você implemente as melhores práticas configurando e modelando o ambiente de execução do seu aplicativo.

## Como gerenciar permissões para alterar recursos do aplicativo

Quando você usa AWS CloudFormation para adicionar recursos de aplicativos em tempo de execução, como uma função Lambda, a função de AWS CloudFormation trabalho pode usar as permissões que já tem. Para alguns recursos de aplicativo em tempo de execução, você deve ajustar

manualmente as permissões da função de operador do AWS CloudFormation antes de editar o arquivo `template.yml`.

Para obter um exemplo de alteração das permissões da função de AWS CloudFormation trabalhador, consulte [Etapa 5: adicionar permissões de recursos com uma política em linha](#).

# AWS CodeStar Melhores práticas

AWS CodeStar está integrado a vários produtos e serviços. As seções a seguir descrevem as melhores práticas para AWS CodeStar esses produtos e serviços relacionados.

## Tópicos

- [Melhores práticas de segurança para recursos do AWS CodeStar](#)
- [Melhores práticas a fim de configurar versões para dependências](#)
- [Monitoramento e registro em log das melhores práticas para recursos do AWS CodeStar](#)

## Melhores práticas de segurança para recursos do AWS CodeStar

Você deve aplicar patches regularmente e revisar as melhores práticas de segurança para as dependências usadas pelo aplicativo. Use essas melhores práticas de segurança para atualizar o código de exemplo e manter o projeto em um ambiente de produção:

- Acompanhe anúncios de segurança e atualizações contínuas para a estrutura.
- Para implantar o projeto, siga as melhores práticas desenvolvidas para a estrutura.
- Analise as dependências da estrutura regularmente e atualize conforme necessário.
- Cada AWS CodeStar modelo contém instruções de configuração para sua linguagem de programação. Consulte o arquivo README .md no repositório de origem do projeto.
- Como prática recomendada para isolar os recursos do projeto, gerencie o acesso com privilégio mínimo aos recursos do AWS usando uma estratégia de várias contas, conforme apresentado em [Segurança em AWS CodeStar](#).

## Melhores práticas a fim de configurar versões para dependências

O código-fonte de amostra em seu AWS CodeStar projeto usa dependências listadas no package .json arquivo em seu repositório de origem. Como melhores práticas, defina sempre as dependências a fim de apontar para uma versão específica. Isso é conhecido como fixar a versão. Não recomendamos definir a versão como latest porque isso pode introduzir alterações que podem travar o aplicativo sem aviso prévio.

# Monitoramento e registro em log das melhores práticas para recursos do AWS CodeStar

Você pode usar os recursos de login AWS para determinar as ações que os usuários realizaram em sua conta e os recursos que foram usados. Os arquivos de log mostram:

- A data e hora das ações.
- O endereço IP de origem de uma ação.
- As ações que falharam devido a permissões inadequadas.

AWS CloudTrail pode ser usado para registrar chamadas de AWS API e eventos relacionados feitos por ou em nome de uma AWS conta. Para obter mais informações, consulte [Registrando chamadas de AWS CodeStar API com AWS CloudTrail](#).

# Trabalhando com projetos em AWS CodeStar

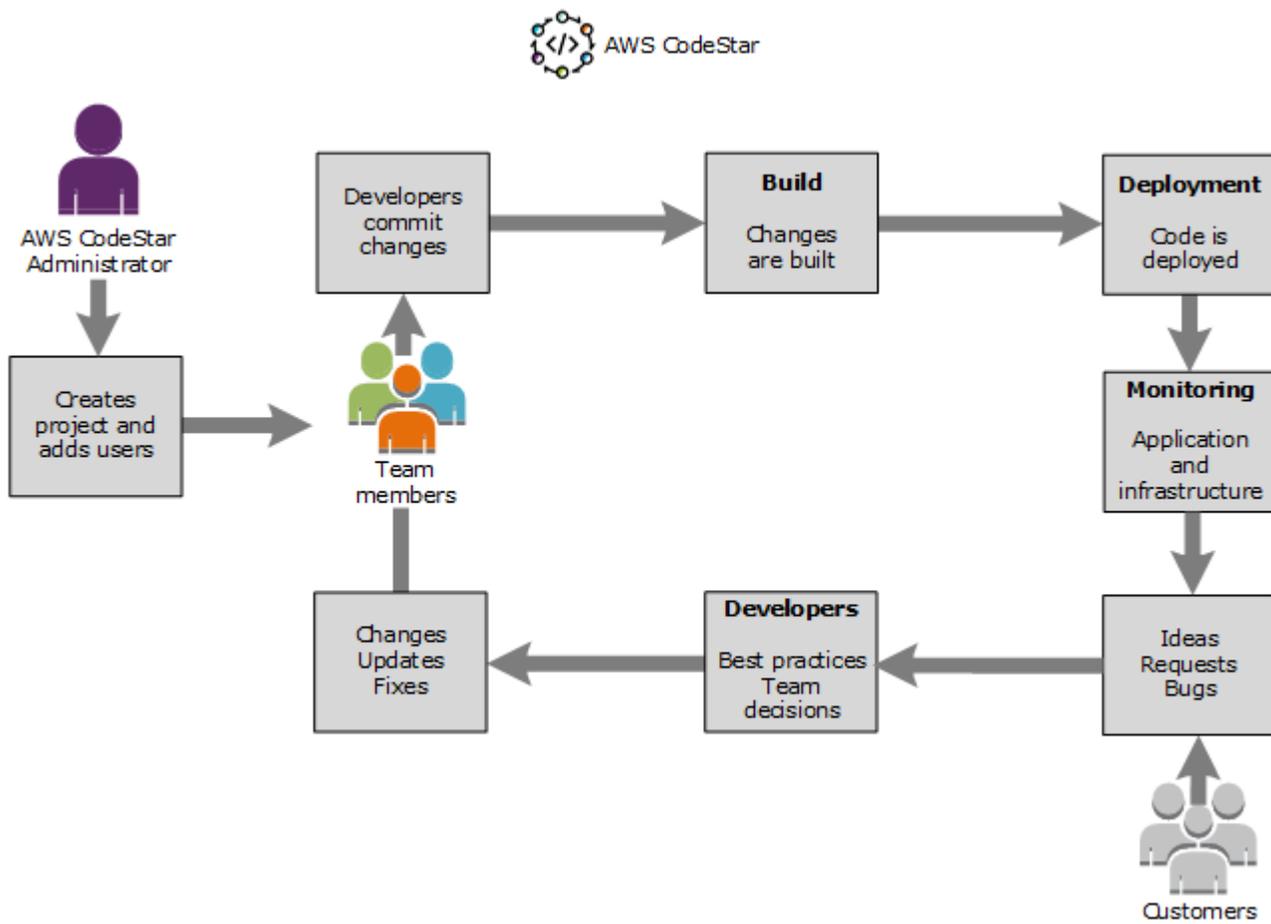
Ao usar um modelo de AWS CodeStar projeto, você pode criar rapidamente um projeto que já esteja configurado com os recursos necessários, incluindo:

- Repositório de código-fonte
- Ambiente de compilação
- Recursos de hospedagem e compilação
- Linguagem de programação

O modelo inclui até exemplos de código-fonte para que você possa começar a trabalhar com o projeto imediatamente.

Depois que você tiver um projeto, você poderá adicionar ou remover recursos, personalizar o painel do projeto e monitorar o andamento.

O diagrama a seguir mostra um fluxo de trabalho básico em um AWS CodeStar projeto.



O fluxo de trabalho básico no diagrama mostra um desenvolvedor com a política do `AWSCodeStarFullAccess` aplicada que cria um projeto e adiciona os membros da equipe ao projeto. Juntos, eles escrevem, compilam, testam e implantam código. O painel do projeto fornece ferramentas que podem ser usadas em tempo real para visualizar a atividade do aplicativo e monitorar as compilações, o fluxo do código por meio do pipeline de implantação e muito mais. A equipe usa o bloco de wiki da equipe para compartilhar informações, práticas recomendadas e links. Eles integram o software de rastreamento de problemas do software para ajudar a acompanhar o andamento e as tarefas. À medida que os clientes fornecem solicitações e comentários, a equipe adiciona essas informações ao projeto e as integra no planejamento e no desenvolvimento do projeto. À medida que o projeto cresce, a equipe adiciona mais membros à equipe para oferecer suporte à base de código.

# Crie um projeto em AWS CodeStar

Você usa o AWS CodeStar console para criar um projeto. Se você usar um modelo de projeto, ele configurará os recursos necessários para você. O modelo também inclui código de exemplo que você pode usar para começar a codificação.

Para criar um projeto, faça login no AWS Management Console com um usuário do IAM que tenha a `AWSCodeStarFullAccess` política ou permissões equivalentes. Para obter mais informações, consulte [Configuração AWS CodeStar](#).

## Note

Você deve concluir as etapas no [Configuração AWS CodeStar](#) antes que possa concluir os procedimentos neste tópico.

## Tópicos

- [Criar um projeto no AWS CodeStar \(console\)](#)
- [Criar um projeto em AWS CodeStar \(AWS CLI\)](#)

## Criar um projeto no AWS CodeStar (console)

Use o AWS CodeStar console para criar um projeto.

Para criar um projeto no AWS CodeStar

1. Faça login no e AWS Management Console, em seguida, abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.

Certifique-se de estar conectado à AWS região em que deseja criar o projeto e seus recursos. Por exemplo, para criar um projeto no Leste dos EUA (Ohio), verifique se você selecionou essa AWS região. Para obter informações sobre AWS regiões onde AWS CodeStar está disponível, consulte [Regiões e endpoints](#) na Referência AWS geral.

2. Na página AWS CodeStar, escolha Criar um projeto.
3. Na página Escolha um modelo de projeto, escolha o tipo de projeto na lista de modelos de AWS CodeStar projeto. Você pode usar a barra de filtro estreitar suas opções. Por exemplo, para que um projeto de aplicativo web escrito em Node.js seja implantado em EC2 instâncias da Amazon,

marque as caixas de seleção Aplicativo Web, Node.js e Amazon EC2. Em seguida, escolha entre os modelos disponíveis para esse conjunto de opções.

Para obter mais informações, consulte [AWS CodeStar Modelos de projeto](#).

4. Escolha Próximo.
5. No campo de entrada de texto Nome do projeto, insira um nome para o projeto, como *My First Project*. Em ID do projeto, a ID para o projeto é derivada desse nome do projeto, mas é limitada a 15 caracteres.

Por exemplo, o ID padrão de um projeto chamado *My First Project* é *my-first-projec*. O ID do projeto é a base dos nomes de todos os recursos associados ao projeto. O AWS CodeStar usa esse ID do projeto como parte do URL para o repositório de códigos, bem como para os nomes de funções de acesso e políticas de segurança relacionados no IAM. Depois que o projeto for criado, o ID do projeto não pode ser alterado. Para editar a ID do projeto antes de criar o projeto, em ID do projeto, insira a ID que você deseja usar.

Para obter informações sobre os limites de nomes de projetos e projetos IDs, consulte [Limites em AWS CodeStar](#).

 Note

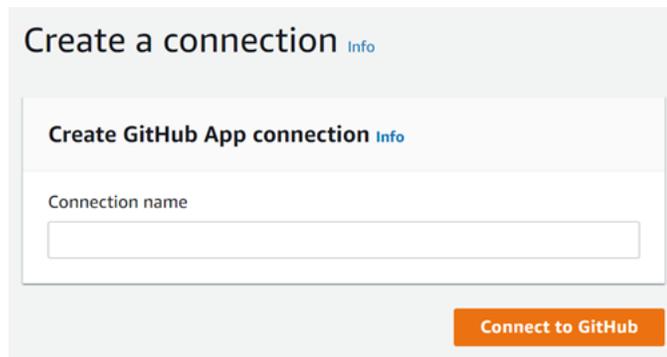
O projeto IDs deve ser exclusivo para sua AWS conta em uma AWS região.

6. Escolha o provedor do repositório AWS CodeCommit ou GitHub.
7. Se você escolher AWS CodeCommit, em Nome do repositório, aceite o nome padrão do AWS CodeCommit repositório ou insira um nome diferente. Em seguida, avance para a etapa 9.
8. Se você escolher GitHub, precisará escolher ou criar um recurso de conexão. Se você tiver uma conexão existente, escolha-a no campo de pesquisa. Caso contrário, crie uma nova conexão agora. Escolha Connect to GitHub.

A página Criar uma conexão é exibida.

 Note

Para criar uma conexão, você precisa ter uma GitHub conta. Se você estiver criando uma conexão para uma organização, você deve ser o proprietário da organização.



- a. Em Criar conexão com o GitHub aplicativo, no campo de texto de entrada do nome da conexão, insira um nome para sua conexão. Escolha Connect to GitHub.

A GitHub página Connect to exibe e mostra o campo GitHub Apps.

- b. Em GitHub Aplicativos, escolha uma instalação de aplicativo ou escolha Instalar um novo aplicativo para criar um.

**Note**

Você instala uma aplicação para todas as suas conexões com um provedor específico. Se você já instalou o GitHub aplicativo AWS Connector for, escolha-o e pule esta etapa.

- c. Na GitHub página Install AWS Connector for, escolha a conta na qual você deseja instalar o aplicativo.

**Note**

Se você instalou a aplicação anteriormente, poderá escolher Configure (Configurar) para prosseguir para uma página de modificação para a instalação da aplicação ou usar o botão Back (Voltar) para retornar ao console.

- d. Se a página Confirmar senha para continuar for exibida, digite sua GitHub senha e escolha Entrar.
- e. Na GitHub página Install AWS Connector for, mantenha os padrões e escolha Instalar.
- f. Na GitHub página Connect to, o ID de instalação da sua nova instalação aparece no campo de entrada de texto GitHub Apps.

Depois que a conexão é criada, na página de CodeStar criação do projeto, a mensagem Pronto para conectar é exibida.

### Note

É possível visualizar sua conexão em Configurações no console do Developer Tools. Para obter mais informações, consulte o tópico de [Conceitos básicos sobre conexões](#).

Select a repository provider

CodeCommit  
Use a new AWS CodeCommit repository for your project.

GitHub  
Use a new GitHub source repository for your project (requires an existing GitHub account).

**The GitHub repository provider now uses CodeStar Connections**  
To use a GitHub repository in CodeStar, create a connection. The connection will use GitHub Apps to access your repository. Use the following options to choose an existing connection or create a new one. [Learn more](#)

Connection  
Choose an existing connection or create a new one and then return to this task.

am:aws:codestar-connections:us-east- X or **Connect to GitHub**

**Ready to connect**  
Your Github connection is ready for use.

Repository owner  
The owner of the new repository. This can be a personal GitHub account or a GitHub organization.

Repository name  
The name of the new repository.

Repository description  
An optional description of the new repository.

Public

- g. Para Proprietário do repositório, escolha a GitHub organização ou sua GitHub conta pessoal.

- h. Em Nome do repositório, aceite o nome padrão do GitHub repositório ou insira um nome diferente.
- i. Escolha Público ou Privado.

 Note

Para usar AWS Cloud9 como seu ambiente de desenvolvimento, você deve escolher Público.

- j. (Opcional) Em Descrição do repositório, insira uma descrição para o GitHub repositório.

 Note

Se você escolher um modelo de projeto Alexa Skill, precisará conectar uma conta de desenvolvedor da Amazon. Para obter mais informações sobre como trabalhar com projetos do Alexa Skill, consulte o [Tutorial: Crie um projeto Alexa Skill no AWS CodeStar](#).

9. Se seu projeto estiver implantado em EC2 instâncias da Amazon e você quiser fazer alterações, configure suas EC2 instâncias da Amazon na EC2 Configuração da Amazon. Por exemplo, você pode escolher entre os tipos de instância disponíveis para o projeto.

 Note

Diferentes tipos de EC2 instância da Amazon fornecem níveis diferentes de poder computacional e podem ter custos associados diferentes. Para obter mais informações, consulte [Amazon EC2 Instance Types](#) e [Amazon EC2 Pricing](#).

Se tiver mais de uma nuvem privada virtual (VPC) ou várias sub-redes criadas no Amazon Virtual Private Cloud, você também poderá escolher a VPC e a sub-rede a serem utilizadas. No entanto, se você escolher um tipo de EC2 instância da Amazon que não seja compatível com instâncias dedicadas, não poderá escolher uma VPC cuja localização de instância esteja definida como Dedicada.

Para obter mais informações, consulte [O que é o Amazon VPC?](#) e [Noções básicas de instâncias dedicadas](#).

Em Par de chaves, escolha o par de EC2 chaves da Amazon em que você criou [Etapa 4: criar um par de EC2 chaves da Amazon para AWS CodeStar projetos](#). Selecione Eu reconheço que tenho acesso ao arquivo de chave privada.

10. Escolha Próximo.
11. Analise os recursos e os detalhes da configuração.
12. Escolha Avançar ou Criar projeto. (A opção exibida depende do modelo do projeto.)

Pode levar alguns minutos para criar o projeto, incluindo o repositório.

13. Depois que seu projeto tiver um repositório, você poderá usar a página Repositório para configurar o acesso a ele. Use os links nas Próximas etapas para configurar um IDE, configurar o controle de problemas ou adicionar membros da equipe ao seu projeto.

Embora o projeto esteja sendo criado, você pode [adicionar membros à equipe](#) ou [configurar o acesso](#) ao repositório do projeto na linha de comando ou de seu IDE favorito.

## Criar um projeto em AWS CodeStar (AWS CLI)

Um AWS CodeStar projeto é uma combinação do código-fonte e dos recursos criados para implantar o código. A coleção de recursos que ajudam a compilar, lançar e implantar o código são chamados de recursos da cadeia de ferramentas. Na criação do projeto, um AWS CloudFormation modelo provisiona os recursos do seu conjunto de ferramentas em um pipeline integration/continuous deployment (CI/CD (contínuo)).

Quando você usa o console para criar um projeto, o modelo da cadeia de ferramentas é criado para você. Ao usar o AWS CLI para criar um projeto, você cria o modelo do conjunto de ferramentas que cria os recursos do conjunto de ferramentas.

Uma cadeia de ferramentas completa requer os seguintes recursos recomendados:

1. Um GitHub repositório CodeCommit ou que contém seu código-fonte.
2. Um CodePipeline pipeline configurado para ouvir as alterações no seu repositório.
  - a. Quando você usa CodeBuild para executar testes de unidade ou integração, recomendamos que você adicione um estágio de construção ao seu pipeline para criar artefatos de construção.
  - b. Recomendamos que você adicione um estágio de implantação ao seu pipeline que use CodeDeploy ou AWS CloudFormation implante seu artefato de construção e código-fonte em sua infraestrutura de tempo de execução.

 Note

Como CodePipeline requer pelo menos dois estágios em um pipeline, e o primeiro estágio deve ser o estágio de origem, adicione um estágio de construção ou implantação como segundo estágio.

AWS CodeStar os conjuntos de ferramentas são definidos como um [CloudFormation modelo](#).

Para um tutorial que aborda essa tarefa e configura recursos de exemplo, consulte [Tutorial: Crie um projeto AWS CodeStar com o AWS CLI](#).

Pré-requisitos:

Ao criar um projeto, você fornece os parâmetros a seguir em um arquivo de entrada. Se o seguinte não for fornecido, AWS CodeStar cria um projeto vazio.

- Código-fonte. Se esse parâmetro estiver incluído na solicitação, você também deverá incluir um modelo da cadeia de ferramentas.
  - O código-fonte deve incluir o código do aplicativo necessários para executar o projeto.
  - Seu código-fonte deve incluir todos os arquivos de configuração necessários, como um `buildspec.yml` para um CodeBuild projeto ou um `appspect.yml` para uma implantação. CodeDeploy
  - Você pode incluir itens opcionais em seu código-fonte, como um README ou um `template.yml` para recursos que não sejam do conjunto de ferramentas. AWS
- Modelo da cadeia de ferramentas. Seu modelo de cadeia de ferramentas provisiona AWS os recursos e as funções do IAM a serem gerenciados para seu projeto.
- Locais de origem. Se especificar o código-fonte e um modelo da cadeia de ferramentas para o projeto, você deverá fornecer um local. Faça upload dos arquivos de origem e o modelo da cadeia de ferramentas para o bucket do Amazon S3. O AWS CodeStar recupera os arquivos e os utiliza para criar o projeto.

 Important

Certifique-se de configurar a AWS região preferida no AWS CLI. Seu projeto é criado na AWS Região configurada no AWS CLI.

1. Execute o comando `create-project` e inclua o parâmetro `--generate-cli-skeleton`:

```
aws codestar create-project --generate-cli-skeleton
```

Os dados formatados JSON aparecem na saída. Copie os dados para um arquivo (por exemplo, `input.json`) em um local no computador local ou na instância em que o AWS CLI esteja instalado. Modifique os dados copiados da maneira a seguir e salve os resultados.

```
{
  "name": "project-name",
  "id": "project-id",
  "description": "description",
  "sourceCode": [
    {
      "source": {
        "s3": {
          "bucketName": "s3-bucket-name",
          "bucketKey": "s3-bucket-object-key"
        }
      },
      "destination": {
        "codeCommit": {
          "name": "codecommit-repository-name"
        },
        "gitHub": {
          "name": "github-repository-name",
          "description": "github-repository-description",
          "type": "github-repository-type",
          "owner": "github-repository-owner",
          "privateRepository": true,
          "issuesEnabled": true,
          "token": "github-personal-access-token"
        }
      }
    }
  ],
  "toolchain": {
    "source": {
      "s3": {
        "bucketName": "s3-bucket-name",
        "bucketKey": "s3-bucket-object-key"
      }
    }
  }
}
```

```

    },
    "roleArn": "service-role-arn",
    "stackParameters": {
      "KeyName": "key-name"
    }
  },
  "tags": {
    "KeyName": "key-name"
  }
}

```

Substitua o seguinte:

- *project-name*: obrigatório O nome amigável para este AWS CodeStar projeto.
- *project-id*: obrigatório O ID do projeto para esse AWS CodeStar projeto.

#### Note

Você deve ter um ID de projeto exclusivo ao criar um projeto. Você receberá um erro se enviar um arquivo de entrada com um ID de projeto já existente.

- *description*: opcional. A descrição desse AWS CodeStar projeto.
- *sourceCode*: opcional. As informações de configuração do código-fonte fornecido para o projeto. Atualmente, somente um objeto `sourceCode` é compatível. Cada `sourceCode` objeto contém informações sobre o local pelo qual o código-fonte é recuperado AWS CodeStar e o destino em que o código-fonte é preenchido.
- *source*: obrigatório Ele define o local onde você fez upload do código-fonte. A única fonte compatível é o Amazon S3. AWS CodeStar recupera o código-fonte e o inclui no repositório após a criação do projeto.
  - *S3*: opcional. O local do Amazon S3 do código-fonte.
    - *bucket-name*: o bucket que contém seu código-fonte.
    - *bucket-key*: o prefixo do bucket e a chave do objeto que apontam para o arquivo.zip que contém seu código-fonte (por exemplo,src.zip).
- *destination*: opcional. Os locais de destino onde o código-fonte deve ser preenchido quando o projeto é criado. Os destinos suportados para seu código-fonte são CodeCommit GitHub e.

**Você só pode fornecer uma destas duas opções:**

- ***codeCommit***: o único atributo obrigatório é o nome do CodeCommit repositório que deve conter seu código-fonte. Esse repositório deve estar no modelo da cadeia de ferramentas.

 Note

Para isso CodeCommit, você deve fornecer o nome do repositório que você definiu na pilha do conjunto de ferramentas. AWS CodeStar inicializa esse repositório com o código-fonte que você forneceu no Amazon S3.

- ***github***: esse objeto representa as informações necessárias para criar o GitHub repositório e alimentá-lo com o código-fonte. Se você escolher um GitHub repositório, os valores a seguir serão obrigatórios.

 Note

Pois GitHub, você não pode especificar um GitHub repositório existente. AWS CodeStar cria um para você e preenche esse repositório com o código-fonte que você enviou para o Amazon S3. AWS CodeStar usa as informações a seguir para criar seu repositório no GitHub.

- ***name***: obrigatório O nome do seu GitHub repositório.
- ***description***: obrigatório A descrição do seu GitHub repositório.
- ***type***: obrigatório O tipo de GitHub repositório. Os valores válidos são User ou Organization.
- ***owner***: obrigatório O nome de GitHub usuário do proprietário do seu repositório. Se o repositório for de propriedade de uma GitHub organização, forneça o nome da organização.
- ***privateRepository***: obrigatório Se você quiser que esse repositório seja privado ou público. Os valores válidos são true ou false.
- ***issuesEnabled***: obrigatório Se você deseja habilitar problemas GitHub com este repositório. Os valores válidos são true ou false.
- ***token***: opcional. Esse é um token de acesso pessoal AWS CodeStar usado para acessar sua GitHub conta. Esse token deve conter os seguintes escopos: repo, user e admin:repo\_hook. Para recuperar um token de acesso pessoal de GitHub, consulte [Criação de um token de acesso pessoal para a linha de comando](#) no GitHub site.

**Note**

Se você usa a CLI para criar um projeto com um repositório de GitHub origem, AWS CodeStar use seu token para acessar o repositório por meio de aplicativos. OAuth Se você usa o console para criar um projeto com um repositório de GitHub origem, AWS CodeStar usa um recurso de conexão, que acessa o repositório com aplicativos. GitHub

- **toolchain**: Informações sobre o conjunto de ferramentas de CI/CD a ser configurado quando o projeto for criado. Isso inclui o local em que você fez upload do modelo da cadeia de ferramentas. O modelo cria uma pilha do AWS CloudFormation que contém seus recursos da cadeia de ferramentas. Isso também inclui qualquer substituição de parâmetro para referência e AWS CloudFormation a função a ser usada para criar a pilha. AWS CodeStar recupera o modelo e o usa AWS CloudFormation para executar o modelo.
- **source**: obrigatório A localização do seu modelo de conjunto de ferramentas. O Amazon S3 é o único local de origem compatível.
- **S3**: opcional. O local do Amazon S3 onde você fez upload do modelo da cadeia de ferramentas.
  - **bucket-name**: O nome do bucket do Amazon S3.
  - **bucket-key**: o prefixo do bucket e a chave do objeto que apontam para o arquivo .yaml ou .json que contém seu modelo de conjunto de ferramentas (por exemplo,). files/toolchain.yaml
- **stackParameters**: opcional. Contém pares de chave-valor a serem passados para o AWS CloudFormation. Esses serão os parâmetros, se houver, para o modelo da cadeia de ferramentas fazer referência na configuração.
- **role**: opcional. A função usada para criar os recursos da cadeia de ferramentas na conta. A função é necessária da seguinte maneira:
  - Se a função não for fornecida, AWS CodeStar usa a função de serviço padrão criada para sua conta se o conjunto de ferramentas for um modelo de AWS CodeStar início rápido. Se a função de serviço não existir na conta, você poderá criar uma. Para ter mais informações, consulte [Etapa 2: Criar a função AWS CodeStar de serviço](#).
  - Você deverá fornecer a função se estiver carregando e usando seu próprio modelo de cadeia de ferramentas personalizado. Você pode criar uma função com base na

função de serviço e na declaração de política do AWS CodeStar . Para obter um exemplo dessa declaração de política, consulte [AWSCodeStarServiceRole Política](#).

- **tags**: opcional. As tags anexadas ao seu AWS CodeStar projeto.

 Note

Essas tags não são anexadas a recursos contidos no projeto.

2. Alterne para o diretório que contenha o arquivo que você acabou de salvar e execute o comando `create-project` novamente. Inclua o parâmetro `--cli-input-json`.

```
aws codestar create-project --cli-input-json file://input.json
```

3. Se bem-sucedido, dados semelhantes aos seguintes serão exibidos na saída:

```
{
  "id": "project-ID",
  "arn": "arn"
}
```

- A saída contém informações sobre o novo projeto:

- O valor `id` representa o ID do projeto.
- O valor `arn` representa o ARN do projeto.

4. Use o comando `describe-project` para verificar o status da criação do projeto. Inclua o parâmetro `--id`.

```
aws codestar describe-project --id <project_ID>
```

Dados semelhantes aos seguintes são exibidos na saída:

```
{
  "name": "MyProject",
  "id": "myproject",
  "arn": "arn:aws:codestar:us-east-1:account_ID:project/myproject",
  "description": "",
  "createdTimeStamp": 1539700079.472,
  "stackId": "arn:aws:cloudformation:us-east-1:account_ID:stack/awscodestar-  
myproject/stack-ID",
  "status": {
```

```
    "state": "CreateInProgress"  
  }  
}
```

- A saída contém informações sobre o novo projeto:
  - O valor `state` representa o status da criação do projeto, como `CreateInProgress` ou `CreateComplete`.

Embora o projeto esteja sendo criado, você pode [adicionar membros à equipe](#) ou [configurar o acesso](#) ao repositório do projeto na linha de comando ou de seu IDE favorito.

## Use um IDE com AWS CodeStar

Ao integrar um IDE com AWS CodeStar, você pode continuar escrevendo e desenvolvendo código em seu ambiente preferido. As alterações que você faz são incluídas no AWS CodeStar projeto sempre que você confirma e envia seu código.

The screenshot shows an IDE window with a code editor on the left and a Git commit interface on the right. The code editor displays the following HTML code:

```

48     <nav class="website-nav">
49         <ul>
50             <li><a class="home-link" href="https://aws.amazon.com/">
51             <li><a href="https://aws.amazon.com/what-is-cloud-comput
52             <li><a href="https://aws.amazon.com/solutions/">Services
53             <li><a href="https://aws.amazon.com/contact-us/">Contact
54         </ul>
55     </nav>
56 </header>
57
58     <div class="message">
59         <a class="twitter-link" href="http://twitter.com/home/?status=I
60         <div class="text">
61             <h1>Congratulations!</h1>
62             <h2>You just created a Node.js web application</h2>
63             <h3>And I made a change in Eclipse!</h3>
64         </div>
65     </div>
66 </div>
67
68 <footer>
69     <p class="footer-contents">Designed and developed with <a href="http

```

The Git commit interface shows the following details:

- Unstaged Changes (1):** .project
- Staged Changes (1):** index.html - public
- Commit Message:** Updated index.html with a new h3
- Author:** Mary Major <mary\_major@example.com>
- Committer:** Mary Major <mary\_major@example.com>
- Buttons:** Commit and Push..., Commit

## Tópicos

- [Use AWS Cloud9 com AWS CodeStar](#)
- [Use o Eclipse com AWS CodeStar](#)
- [Use o Visual Studio com AWS CodeStar](#)

## Use AWS Cloud9 com AWS CodeStar

Você pode usar AWS Cloud9 para fazer alterações no código e desenvolver software em um AWS CodeStar projeto. AWS Cloud9 é um IDE on-line, que você acessa por meio do seu navegador da web. O IDE oferece uma experiência de edição de código completa com suporte para várias

linguagens de programação e depuradores de tempo de execução, bem como um terminal integrado. Em segundo plano, uma EC2 instância da Amazon hospeda um ambiente de AWS Cloud9 desenvolvimento. Esse ambiente fornece o AWS Cloud9 IDE e acesso aos arquivos de código do AWS CodeStar projeto. Para obter mais informações, consulte o Guia do usuário do [AWS Cloud9](#).

Você pode usar o AWS CodeStar console ou o AWS Cloud9 console para criar ambientes de AWS Cloud9 desenvolvimento para projetos que armazenam seu código em CodeCommit. Para AWS CodeStar projetos que armazenam seu código GitHub, você só pode usar o AWS Cloud9 console. Este tópico descreve como usar ambos os consoles.

Para usar AWS Cloud9, você precisa:

- Um usuário do IAM que foi adicionado como membro da equipe a um AWS CodeStar projeto.
- Se o AWS CodeStar projeto armazenar seu código-fonte em CodeCommit, AWS credenciais para o usuário do IAM.

## Tópicos

- [Crie um AWS Cloud9 ambiente para um projeto](#)
- [Abra um AWS Cloud9 ambiente para um projeto](#)
- [Compartilhe um AWS Cloud9 ambiente com um membro da equipe do projeto](#)
- [Excluir um AWS Cloud9 ambiente de um projeto](#)
- [Use GitHub com AWS Cloud9](#)
- [Recursos adicionais](#)

## Crie um AWS Cloud9 ambiente para um projeto

Siga estas etapas para criar um ambiente de AWS Cloud9 desenvolvimento para um AWS CodeStar projeto.

1. Siga as etapas em [Criar um projeto](#) se deseja criar um novo projeto.
2. Abra o projeto no AWS CodeStar console. Na barra de navegação, escolha IDE. Escolha Criar ambiente e use as etapas a seguir.

**⚠ Important**

Se o projeto estiver em uma AWS região sem suporte, você não verá AWS Cloud9 as opções na guia IDE na barra de navegação. AWS Cloud9 No entanto, você pode usar o AWS Cloud9 console para criar um ambiente de desenvolvimento, abrir o novo ambiente e conectá-lo ao AWS CodeCommit repositório do projeto. Ignore as etapas a seguir e consulte [Criar um ambiente](#), [Abrir um ambiente](#) e [Exemplo do AWS CodeCommit](#) no Guia do usuário do AWS Cloud9 . Para ver a lista de AWS regiões suportadas, consulte [AWS Cloud9](#) no Referência geral da Amazon Web Services.

Em Criar AWS Cloud9 ambiente, personalize os padrões do projeto.

1. Para alterar o tipo padrão de EC2 instância da Amazon para hospedar o ambiente, em Tipo de instância, escolha o tipo de instância.
2. AWS Cloud9 usa a Amazon Virtual Private Cloud (Amazon VPC) em sua AWS conta para se comunicar com a instância. Dependendo de como a Amazon VPC está configurada em sua AWS conta, faça o seguinte.

A conta tem uma VPC com, pelo menos, uma sub-rede nessa VPC?	A VPC que você deseja usar é AWS Cloud9 a VPC padrão na conta?	A VPC possui uma única sub-rede?	Faça o seguinte
Não	—	—	Se não houver nenhuma VPC, crie uma. Expanda Configurações de rede. Em Network (VPC) (Rede [VPC]), escolha Create VPC (Criar VPC) e siga as instruções na página. Para obter mais informações, consulte <a href="#">Criar um Amazon VPC para o AWS Cloud9</a> no Guia do usuário do AWS Cloud9 .

A conta tem uma VPC com, pelo menos, uma sub-rede nessa VPC?	A VPC que você deseja usar é AWS Cloud9 a VPC padrão na conta?	A VPC possui uma única sub-rede?	Faça o seguinte
			Se existe uma VPC mas não há sub-rede, crie uma. Expanda Configurações de rede. Em Network (VPC) (Rede [VPC]), escolha Create subnet (Criar sub-rede) e siga as instruções. Para obter mais informações, consulte <a href="#">Criar uma sub-rede para o AWS Cloud9</a> no Guia do usuário AWS Cloud9 .
Sim	Sim	Sim	Vá para a etapa 4 deste procedimento. (AWS Cloud9 usa a VPC padrão com sua única sub-rede.)
Sim	Sim	Não	Em Subnet (Sub-rede), selecione a sub-rede que você deseja que o AWS Cloud9 use na VPC padrão pré-selecionada.
Sim	Não	Yes ou No	Em Rede (VPC), escolha a VPC que você deseja usar. AWS Cloud9 Em Sub-rede, escolha a sub-rede que você deseja AWS Cloud9 usar nessa VPC.

Para obter mais informações, consulte Configurações [da Amazon VPC para ambientes de AWS Cloud9 desenvolvimento](#) no Guia do AWS Cloud9 usuário.

- Insira um nome de ambiente e, opcionalmente, adicione uma descrição do ambiente.

**Note**

Os nomes do ambiente devem ser exclusivos por usuário.

4. Para alterar o período de tempo padrão após o qual o ambiente AWS Cloud9 é desligado quando ele não foi usado, expanda Configurações de economia de custos e, em seguida, altere a configuração.
5. Selecione Criar ambiente.

Para abrir o ambiente, consulte [Abra um AWS Cloud9 ambiente para um projeto](#).

Você pode usar essas etapas para criar mais de um ambiente para um projeto. Por exemplo, convém usar um ambiente para trabalhar em uma parte do código e outro ambiente para trabalhar na mesma parte do código com configurações diferentes.

## Abra um AWS Cloud9 ambiente para um projeto

Siga estas etapas para abrir um ambiente de AWS Cloud9 desenvolvimento que você criou para um AWS CodeStar projeto.

1. Com o projeto aberto no AWS CodeStar console, na barra de navegação, escolha IDE.

### Important

Se o código-fonte do projeto estiver armazenado GitHub, você não verá o IDE na barra de navegação. No entanto, você pode usar o AWS Cloud9 console para abrir um ambiente existente. Ignore o restante deste procedimento e consulte [Abertura de um ambiente](#) no Guia do usuário do AWS Cloud9 e [Use GitHub com AWS Cloud9](#).

2. Para Seus AWS Cloud9 ambientes ou AWS Cloud9 Ambientes compartilhados, escolha Open IDE para o ambiente que você deseja abrir.

Você pode usar o AWS Cloud9 IDE para começar a trabalhar com código no AWS CodeCommit repositório do projeto imediatamente. Para obter mais informações, consulte [A janela do ambiente, O editor, as guias e os painéis](#) e [O terminal](#) no Guia do usuário do AWS Cloud9 e [Comandos do Git básicos](#) no Guia do usuário do AWS CodeCommit .

## Compartilhe um AWS Cloud9 ambiente com um membro da equipe do projeto

Depois de criar um ambiente de AWS Cloud9 desenvolvimento para um AWS CodeStar projeto, você pode convidar outros usuários da sua AWS conta, incluindo membros da equipe do projeto, para acessar o mesmo ambiente. Isso é especialmente útil na programação de pares, em que

dois programadores aproveitam a codificação e dão orientações sobre o mesmo código por meio do compartilhamento de tela ou ainda sentados na mesma estação de trabalho. Os membros do ambiente podem usar o AWS Cloud9 IDE compartilhado para ver as alterações de código de cada membro destacadas no editor de código e conversar por texto com outros membros durante a codificação.

Adicionar um membro da equipe a um projeto não permite automaticamente que esse membro participe de nenhum ambiente de AWS Cloud9 desenvolvimento relacionado ao projeto. Para convidar um membro da equipe do projeto para acessar um ambiente para um projeto, você precisa determinar a função correta de acesso do membro do ambiente, aplicar políticas AWS gerenciadas ao usuário e convidá-lo para o seu ambiente. Para obter mais informações, consulte [Sobre funções de acesso de membro do ambiente](#) e [Convidar um usuário do IAM para o ambiente](#) no Guia do usuário do AWS Cloud9 .

Quando você convida um membro da equipe do projeto para acessar um ambiente de um projeto, o console do AWS CodeStar exibe o ambiente para esse membro da equipe. O ambiente é exibido na lista Ambientes compartilhados na guia IDE no AWS CodeStar console do projeto. Para exibir essa lista, peça para o membro da equipe abrir o projeto no console e escolher IDE na barra de navegação.

#### Important

Se o código-fonte do projeto estiver armazenado GitHub, você não verá o IDE na barra de navegação. No entanto, você pode usar o AWS Cloud9 console para convidar outros usuários da sua AWS conta, incluindo membros da equipe do projeto, para acessar um ambiente. Para isso, consulte [Use GitHub com AWS Cloud9](#) neste guia e consulte [Sobre funções de acesso de membro do ambiente](#) e [Convidar um usuário do IAM para o ambiente](#) no Guia do usuário do AWS Cloud9 .

Você também pode convidar um usuário que não seja um membro da equipe do projeto para acessar um ambiente. Por exemplo, convém que um usuário trabalhe no código de um projeto, mas não tenha acesso a esse projeto. Para convidar esse tipo de usuário, consulte [Sobre funções de acesso de membro do ambiente](#) e [Convidar um usuário do IAM para o ambiente](#) no Guia do usuário do AWS Cloud9 . Quando você convida um usuário que não seja um membro da equipe do projeto para acessar um ambiente de um projeto, esse usuário pode usar o console do AWS Cloud9 para acessar o ambiente. Para obter mais informações, consulte [Abrir um ambiente](#) no Guia do usuário do AWS Cloud9 .

## Excluir um AWS Cloud9 ambiente de um projeto

Quando você exclui um projeto e todos os seus AWS recursos AWS CodeStar, todos os ambientes de AWS Cloud9 desenvolvimento relacionados que foram criados com o AWS CodeStar console também são excluídos e não podem ser recuperados. Você pode excluir um ambiente de desenvolvimento de um projeto sem excluir o projeto.

1. Com o projeto aberto no AWS CodeStar console, na barra de navegação, escolha IDE.

### Important

Se o código-fonte do projeto estiver armazenado GitHub, você não verá o IDE na barra de navegação. No entanto, você pode usar o AWS Cloud9 console para excluir um ambiente de desenvolvimento. Ignore o restante deste procedimento e consulte [Exclusão de um ambiente](#) no Guia do usuário do AWS Cloud9 .

2. Escolha o ambiente que você deseja excluir nos ambientes do Cloud9 e escolha Excluir
3. Insira **delete** para confirmar a exclusão para o ambiente de desenvolvimento e então selecione Excluir.

### Warning

Você não poderá recuperar um ambiente de desenvolvimento depois que ele tiver sido excluído. Todas as alterações de código não confirmadas no ambiente são perdidas.

## Use GitHub com AWS Cloud9

Para AWS CodeStar projetos que têm seu código-fonte armazenado GitHub, o AWS CodeStar console não suporta trabalhar diretamente com ambientes de AWS Cloud9 desenvolvimento. No entanto, você pode usar o AWS Cloud9 console para trabalhar com o código-fonte nos GitHub repositórios.

1. Use o AWS Cloud9 console para criar um ambiente AWS Cloud9 de desenvolvimento. Para obter mais informações, consulte [Criação de um ambiente](#) no Guia do usuário do AWS Cloud9 .
2. Use o AWS Cloud9 console para abrir o ambiente de desenvolvimento. Para obter mais informações, consulte [Abertura de um ambiente](#) no Guia do usuário do AWS Cloud9 .

3. No IDE, use uma sessão de terminal para se conectar ao GitHub repositório (um processo conhecido como clonagem). Se uma sessão de terminal não estiver em execução, na barra de menus no IDE, escolha Window, New Terminal (Janela, novo terminal). Para ver os comandos a serem usados para clonar o GitHub repositório, consulte [Clonando um repositório](#) no site de Ajuda. GitHub

Para navegar até a página principal do GitHub repositório, com o projeto aberto no AWS CodeStar console, na barra de navegação lateral, escolha Código.

4. Use a janela Environment (Ambiente) e as guias do editor no IDE para exibir, alterar e salvar o código. Para obter mais informações, consulte [A janela de ambiente](#) e [O editor, as guias e os painéis](#) no Guia do usuário do AWS Cloud9 .
5. Use o Git na sessão de terminal no IDE para enviar alterações feitas no código para o repositório e obter alterações feitas no código periodicamente de outros do repositório. Para obter mais informações, consulte [Enviando para um repositório remoto e Buscando um repositório remoto](#) no site de Ajuda. GitHub Para comandos do Git, consulte [Git Cheatsheet no site](#) de Ajuda. GitHub

#### Note

Para evitar que o Git solicite suas credenciais de GitHub login toda vez que você enviar ou extrair código do repositório, você pode usar um auxiliar de credenciais. Para obter mais informações, consulte [Armazenando sua GitHub senha em cache no Git](#) GitHub no site de Ajuda.

## Recursos adicionais

Para obter mais informações sobre o uso AWS Cloud9, consulte o seguinte no Guia AWS Cloud9 do usuário:

- [Tutorial](#)
- [Trabalhar com ambientes](#)
- [Trabalhar com a IDE](#)
- [Amostras](#)

## Use o Eclipse com AWS CodeStar

Você pode usar o Eclipse para fazer alterações no código e desenvolver software em um AWS CodeStar projeto. Você pode editar o código AWS CodeStar do seu projeto com o Eclipse e, em seguida, confirmar e enviar suas alterações para o repositório de origem do AWS CodeStar projeto.

### Note

As informações neste tópico se aplicam somente aos AWS CodeStar projetos que armazenam seu código-fonte em CodeCommit. Se seu AWS CodeStar projeto armazena seu código-fonte em GitHub, você pode usar uma ferramenta como EGit para o Eclipse. Para obter mais informações, consulte a [EGit documentação](#) no EGit site.

Se o AWS CodeStar projeto armazenar seu código-fonte em CodeCommit, você deverá instalar uma versão do AWS Toolkit for Eclipse que suporte AWS CodeStar. Você também deve ser membro da equipe do AWS CodeStar projeto com a função de proprietário ou colaborador.

Para usar o Eclipse, você também precisa:

- Um usuário do IAM que foi adicionado a um AWS CodeStar projeto como membro da equipe.
- Se o AWS CodeStar projeto armazenar seu código-fonte nas CodeCommit credenciais do [Git \(credenciais de login\)](#) para o usuário do IAM.
- Permissões suficientes para instalar o Eclipse e o AWS Toolkit for Eclipse em seu computador local.

### Tópicos

- [Etapa 1: instalar AWS Toolkit for Eclipse](#)
- [Etapa 2: importar seu AWS CodeStar projeto para o Eclipse](#)
- [Etapa 3: Editar o código AWS CodeStar do projeto no Eclipse](#)

## Etapa 1: instalar AWS Toolkit for Eclipse

O Toolkit for Eclipse é um pacote de software que você pode adicionar ao Eclipse. Ele é instalado e gerenciado da mesma forma como outros pacotes de software no Eclipse. O AWS CodeStar kit de ferramentas está incluído como parte do Toolkit for Eclipse.

## Para instalar o Toolkit for Eclipse com o módulo AWS CodeStar

1. Instale o Eclipse no seu computador local. As versões suportadas do Eclipse incluem Luna, Mars e Neon.
2. Baixe e instale o Toolkit do Eclipse. Para obter mais informações, consulte o [AWS Toolkit for Eclipse Guia de conceitos básicos do](#).
3. No Eclipse, escolha Ajuda e, em seguida, escolha Instalar novo software.
4. Em Software disponível, escolha Adicionar.
5. Em Adicionar repositório, escolha Arquivo, navegue para o local onde você salvou o arquivo .zip e abra o arquivo. Deixe o Nome em branco e, em seguida, clique em OK.
6. Em Available Software (Software disponível), escolha Select all (Selecionar tudo) para selecionar AWS Core Management Tools e Developer Tools (Ferramentas de desenvolvedor), e escolha Next (Avançar).
7. Em Detalhes da instalação, escolha Avançar.
8. Em Analisar licenças, analise os contratos de licença. Escolha I accept the terms of the license agreement (Aceito os termos do contrato de licença) e escolha Finish (Concluir). Reinicie o Eclipse.

## Etapa 2: importar seu AWS CodeStar projeto para o Eclipse

Depois de instalar o Toolkit for Eclipse, você AWS CodeStar pode importar projetos e editar, confirmar e enviar código do IDE.

### Note

Você pode adicionar vários AWS CodeStar projetos a um único espaço de trabalho no Eclipse, mas deve atualizar as credenciais do projeto ao mudar de um projeto para outro.

## Para importar um AWS CodeStar projeto

1. AWS No menu, escolha Importar AWS CodeStar projeto. Como alternativa, escolha Arquivo e, em seguida, escolha Importar. Em Selecionar, expanda AWS e escolha AWS CodeStar Projeto.

Escolha Próximo.

2. Em Seleção de AWS CodeStar projetos, escolha seu AWS perfil e a AWS região onde o AWS CodeStar projeto está hospedado. Se você não tiver um AWS perfil configurado com uma chave de acesso e uma chave secreta no seu computador, escolha Configurar AWS contas e siga as instruções.

Em Selecionar AWS CodeStar projeto e repositório, escolha seu AWS CodeStar projeto. Em Configurar credenciais do Git, insira as credenciais de login que você gerou para acessar o repositório do projeto. (Se você não tiver as credenciais do Git, consulte [Conceitos básicos](#).) Escolha Próximo.

**AWS CodeStar Project Selection**

Select the AWS CodeStar project you want to checkout from the remote host.

Select AWS account and region:

Select Account:  [Configure AWS accounts...](#)

Select Region:

Select AWS CodeStar project and repository:

Project Name	Project ID	Project Description
My First Project	my-first-projec	AWS CodeStar created project

Select repository:

Configure Git credentials:

You can manually copy and paste Git credentials for AWS CodeCommit below. Alternately, you can import them from a downloaded .csv file. To learn how to generate Git credentials, see [Create Git Credentials for HTTPS Connections to AWS CodeCommit](#).

User name:

Password:

Show password

3. Por padrão, todas as ramificações do repositório do projeto estão selecionadas. Se você não desejar importar uma ou mais ramificações, limpe as caixas e, em seguida, escolha Avançar.
4. No Local Destination (Destino local), escolha um destino onde o assistente de importação cria o repositório local em seu computador e, em seguida, escolha Finish (Concluir).
5. No Project Explorer, expanda a árvore do projeto para procurar os arquivos no AWS CodeStar projeto.

## Etapa 3: Editar o código AWS CodeStar do projeto no Eclipse

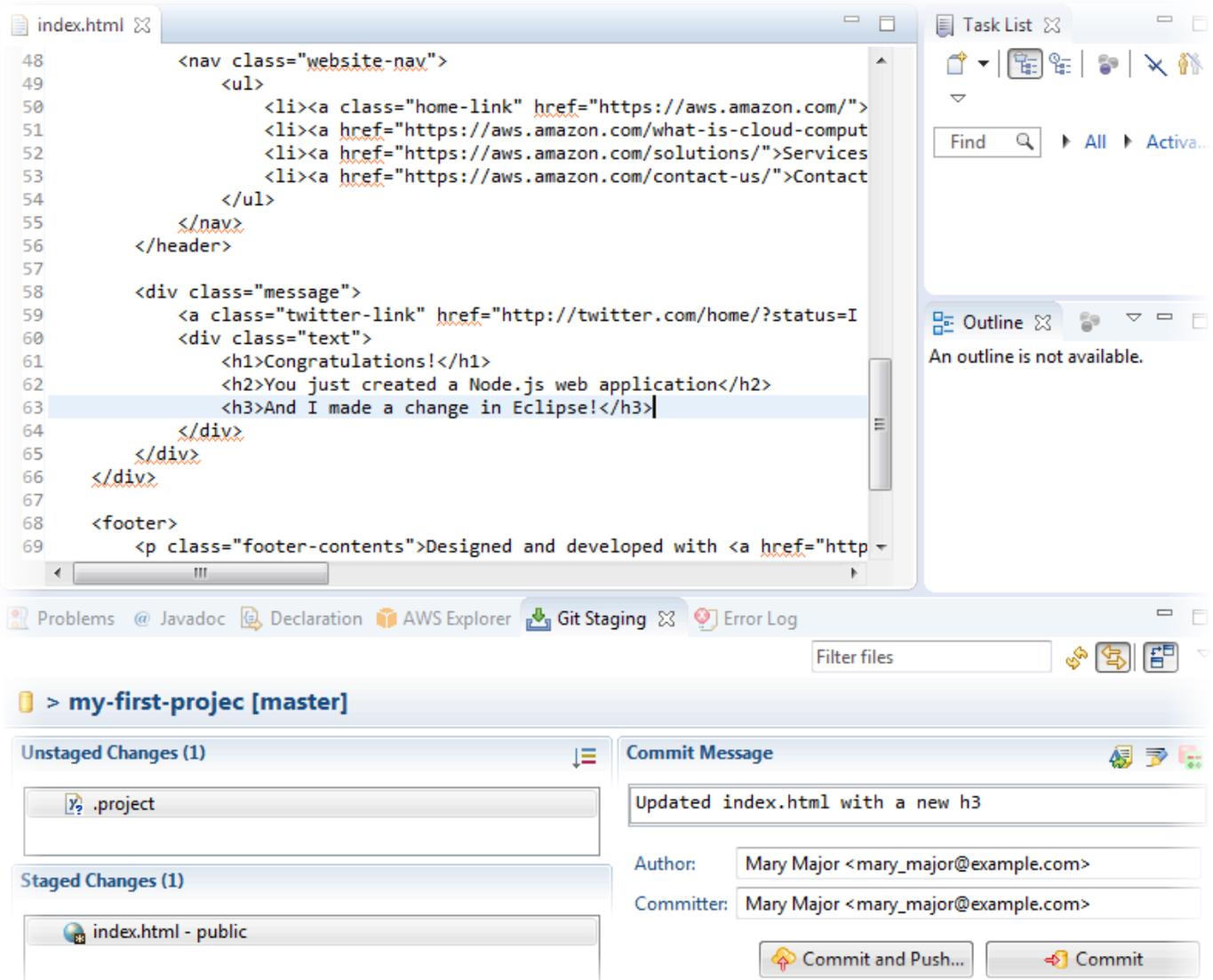
Depois de importar um AWS CodeStar projeto para um espaço de trabalho do Eclipse, você pode editar o código do projeto, salvar suas alterações, confirmar e enviar seu código para o repositório de origem do projeto. Esse é o mesmo processo que você segue para qualquer repositório Git usando o EGit plug-in para Eclipse. Para obter mais informações, consulte o [Guia EGit do usuário](#) no site do Eclipse.

Para editar o código do projeto e fazer sua primeira confirmação no repositório de origem de um AWS CodeStar projeto

1. No Project Explorer, expanda a árvore do projeto para procurar os arquivos no AWS CodeStar projeto.
2. Edite um ou mais arquivos de código e salve as alterações.
3. Quando estiver pronto para confirmar as alterações, abra o menu de contexto desse arquivo, escolha Equipe e, em seguida, escolha Confirmar.

Você poderá ignorar essa etapa, se a janela Git Staging (Preparação do Git) já estiver aberta na visualização do projeto.

4. Em Git Staging (Preparação do Git), prepare as alterações movendo os arquivos alterados para Staged Changes (Alterações preparadas). Digite uma mensagem de confirmação em Commit Message (Mensagem de confirmação) e, em seguida, escolha Commit and Push (Confirmar e enviar por push).



Para visualizar a implantação das alterações do código, volte para o painel de controle de seu projeto. Para obter mais informações, consulte [Etapa 3: Visualizar o projeto](#).

## Use o Visual Studio com AWS CodeStar

Você pode usar o Visual Studio para fazer alterações no código e desenvolver software em um AWS CodeStar projeto.

### Note

O Visual Studio para Mac não oferece suporte ao AWS Toolkit, portanto, ele não pode ser usado com AWS CodeStar.

As informações neste tópico se aplicam somente aos AWS CodeStar projetos que armazenam seu código-fonte em CodeCommit. Se seu AWS CodeStar projeto armazena seu código-fonte em GitHub, você pode usar uma ferramenta como a GitHub extensão para Visual Studio. Para obter mais informações, consulte a página [Visão geral](#) no site da GitHub Extensão para Visual Studio e [Introdução ao GitHub Visual Studio](#) no GitHub site.

Para usar o Visual Studio para editar código no repositório de origem de um AWS CodeStar projeto, você deve instalar uma versão do AWS Toolkit for Visual Studio que ofereça suporte AWS CodeStar. Você deve ser membro da equipe de projeto do AWS CodeStar com a função de proprietário ou colaborador.

Para usar o Visual Studio, você também precisa:

- Um usuário do IAM que foi adicionado a um AWS CodeStar projeto como membro da equipe.
- AWS credenciais para seu usuário do IAM (por exemplo, sua chave de acesso e chave secreta).
- Permissões suficientes para instalar o Visual Studio e o AWS Toolkit for Visual Studio em seu computador local.

O Toolkit for Visual Studio é um pacote de software que você pode adicionar ao Visual Studio. Ele é instalado e gerenciado da mesma forma como outros pacotes de software no Visual Studio.

Para instalar o Toolkit for Visual Studio com AWS CodeStar o módulo e configurar o acesso ao repositório do projeto

1. Instale o Visual Studio no computador local.
2. Faça o download e instale o Toolkit for Visual Studio e salve o arquivo .zip em uma pasta ou diretório local. Na página Introdução à AWS Toolkit for Visual Studio página, insira ou importe suas AWS credenciais e escolha Salvar e fechar.
3. No Visual Studio, abra o Team Explorer. Em Hosted Service Providers (Provedores de serviços hospedados), localize o CodeCommit e escolha Connect (Conectar).
4. Em Gerenciar conexões, escolha Clonar. Escolha o repositório do seu projeto e a pasta no seu computador local na qual você deseja clonar o repositório e escolha OK.
5. Se você for solicitado a criar credenciais do Git, escolha Yes. O toolkit tenta criar credenciais em seu nome. Salve o arquivo de credenciais em um local seguro. Essa é a única oportunidade que você terá de salvar essas credenciais. Se o kit de ferramentas não puder criar credenciais em

seu nome, ou se você escolher Não, você deve criar e fornecer suas próprias credenciais do Git. Para obter mais informações, consulte [Para configurar o computador para confirmar alterações \(usuário do IAM\)](#) ou siga as instruções online.

Quando terminar de clonar o projeto, você estará pronto para começar a editar seu código no Visual Studio e confirmar e enviar suas alterações para o repositório do seu projeto em CodeCommit.

## Alterar AWS recursos em um AWS CodeStar projeto

Depois de criar um projeto no AWS CodeStar, você pode alterar o conjunto padrão de AWS recursos que é AWS CodeStar adicionado ao projeto.

### Alterações de recursos compatíveis

A tabela a seguir lista as alterações suportadas nos AWS recursos padrão em um AWS CodeStar projeto.

Alteração	Observações
Adicione um palco AWS CodePipeline a.	Consulte <a href="#">Adicionar um estágio ao AWS CodePipeline</a> .
Alterar as configurações do ambiente do Elastic Beanstalk.	Consulte <a href="#">Alterar configurações do AWS Elastic Beanstalk ambiente</a> .
Altere o código ou as configurações de uma AWS Lambda função, sua função do IAM ou sua API no Amazon API Gateway.	Consulte <a href="#">Alterar uma AWS Lambda função no código-fonte</a> .
Adicione um recurso a um AWS Lambda projeto e expanda as permissões para criar e acessar o novo recurso.	Consulte <a href="#">Adicionar um recurso a um projeto</a> .
Adicione a mudança de tráfego com CodeDeploy para uma AWS Lambda função.	Consulte <a href="#">Mudar tráfego para um projeto do AWS Lambda</a> .

Alteração	Observações
Adicionar AWS X-Ray suporte	Consulte <a href="#">Habilitar rastreamento para um projeto</a> .
Edite o arquivo buildspec.yml do seu projeto para adicionar uma fase de compilação de teste de unidade para execução. AWS CodeBuild	Consulte <a href="#">Etapa 7: Adicionar um teste de unidade ao serviço da web</a> no tutorial de Projeto sem servidor.
Adicione sua própria função do IAM ao projeto.	Consulte <a href="#">Adicionar uma função do IAM a um projeto</a> .
Alterar uma definição de perfil do IAM.	Para funções definidas na pilha de aplicativos. Você não pode alterar as funções definidas no conjunto de ferramentas ou nas AWS CloudFormation pilhas.
Modifique o projeto do Lambda para adicionar um endpoint.	
Modifique seu EC2 projeto para adicionar um endpoint.	
Modifique o projeto do Elastic Beanstalk para adicionar um endpoint.	
Edite o projeto para adicionar um estágio de produção e um endpoint.	Consulte <a href="#">Adicionar um estágio e uma endpoint de produção a um projeto</a> .
Use com segurança os parâmetros SSM em um projeto. AWS CodeStar	Consulte <a href="#">the section called “Use com segurança os parâmetros SSM em um projeto AWS CodeStar”</a> .

As seguintes alterações não são compatíveis.

- Mude para um destino de implantação diferente (por exemplo, implante AWS Elastic Beanstalk em vez de AWS CodeDeploy).

- Adicione um nome amigável de endpoint da web.
- Altere o nome CodeCommit do repositório (para um AWS CodeStar projeto conectado a CodeCommit).
- Para um AWS CodeStar projeto conectado a GitHub, desconecte o GitHub repositório e reconecte o repositório a esse projeto ou conecte qualquer outro repositório a esse projeto. Você pode usar o CodePipeline console (não o AWS CodeStar console) para se desconectar e reconectar GitHub no estágio de origem de um pipeline. No entanto, se você reconectar o estágio de origem a um GitHub repositório diferente, no AWS CodeStar painel do projeto, as informações nos blocos Repositório e Problemas podem estar erradas ou desatualizadas. Desconectar o GitHub repositório não remove as informações desse repositório do histórico de confirmações e dos blocos de GitHub problemas no painel do AWS CodeStar projeto. Para remover essas informações, use o GitHub site para desativar o acesso ao GitHub AWS CodeStar projeto. Para revogar o acesso, no GitHub site, use a seção OAuth Aplicativos autorizados da página de configurações do perfil da sua GitHub conta.
- Desconecte o CodeCommit repositório (para um AWS CodeStar projeto conectado a CodeCommit) e reconecte o repositório a esse projeto ou conecte qualquer outro repositório a esse projeto.

## Adicionar um estágio ao AWS CodePipeline

Você pode adicionar um novo estágio a um pipeline AWS CodeStar criado em um projeto. Para obter mais informações, consulte [Editar um pipeline no AWS CodePipeline](#) no Guia do usuário do AWS CodePipeline .

### Note

Se o novo estágio depender de algum AWS recurso que AWS CodeStar não tenha sido criado, o pipeline poderá falhar. Isso ocorre porque a função do IAM AWS CodeStar criada para AWS CodePipeline pode não ter acesso a esses recursos por padrão.

Para tentar dar AWS CodePipeline acesso a AWS recursos que AWS CodeStar não foram criados, talvez você queira alterar a função do IAM AWS CodeStar criada. Isso não é suportado porque AWS CodeStar pode remover suas alterações de função do IAM ao realizar verificações regulares de atualização no projeto.

## Alterar configurações do AWS Elastic Beanstalk ambiente

Você pode alterar as configurações de um AWS CodeStar ambiente do Elastic Beanstalk criado em um projeto. Por exemplo, talvez você queira alterar o ambiente padrão do Elastic Beanstalk AWS CodeStar em seu projeto de Instância Única para Carga Balanceada. Para fazer isso, edite o arquivo `template.yml` no repositório do projeto. Você também pode precisar alterar permissões para as funções de trabalho do projeto. Depois de enviar o modelo, AWS CodeStar altere e AWS CloudFormation provisione os recursos para você.

Para obter mais informações sobre como editar o arquivo `template.yml`, consulte [Alterar recursos do aplicativo com o arquivo `template.yml`](#). Para obter mais informações sobre como personalizar ambientes do Elastic Beanstalk, consulte [Console de Gerenciamento do Ambiente do AWS Elastic Beanstalk](#) no Guia do Desenvolvedor do AWS Elastic Beanstalk .

## Alterar uma AWS Lambda função no código-fonte

Você pode alterar o código ou as configurações de uma função Lambda, ou sua função IAM ou API Gateway, que é AWS CodeStar criada em um projeto. Para fazer isso, recomendamos que você use o AWS Serverless Application Model (AWS SAM) junto com o `template.yaml` arquivo no repositório do CodeCommit seu projeto. Esse arquivo `template.yaml` define o nome da função, o handler, o runtime, o perfil do IAM e a API no Gateway API. Para obter mais informações, consulte [Como criar aplicativos sem servidor usando o AWS SAM](#) no GitHub site.

## Habilitar rastreamento para um projeto

AWS X-Ray oferece rastreamento, que você pode usar para analisar o comportamento de desempenho de aplicativos distribuídos (por exemplo, latências nos tempos de resposta). Depois de adicionar rastreamentos ao seu AWS CodeStar projeto, você pode usar o AWS X-Ray console para visualizar as visualizações do aplicativo e os tempos de resposta.

### Note

Você pode usar estas etapas para os seguintes projetos, criados com as seguintes alterações de suporte do projeto:

- Qualquer projeto do Lambda.
- Para projetos da Amazon EC2 ou do Elastic Beanstalk criados após 3 de agosto de AWS CodeStar 2018, `/template.yml` provisionei um arquivo no repositório do projeto.

Cada AWS CodeStar modelo inclui um AWS CloudFormation arquivo que modela as dependências de AWS tempo de execução do seu aplicativo, como tabelas de banco de dados e funções Lambda. Esse arquivo é armazenado no repositório de origem no arquivo `/template.yml`.

Você pode modificar esse arquivo para adicionar rastreamento adicionando o AWS X-Ray recurso à Resources seção. Em seguida, você modifica as permissões do IAM do seu projeto AWS CloudFormation para permitir a criação do recurso. Para obter mais informações sobre elementos de modelo e formatação, consulte [Referência de tipos de recurso do AWS](#).

Essas são as etapas de alto nível a serem seguidas para personalizar o modelo.

1. [Etapa 1: Editar a função de operador no IAM para rastreamento](#)
2. [Etapa 2: modificar o arquivo `template.yml` para rastreamento](#)
3. [Etapa 3: confirmar e enviar a alteração do modelo para rastreamento](#)
4. [Etapa 4: monitorar a atualização da pilha do AWS CloudFormation para rastreamento](#)

## Etapa 1: Editar a função de operador no IAM para rastreamento

Você deve entrar como administrador para realizar as etapas 1 e 4. Essa etapa mostra um exemplo de permissões de edição para um projeto do Lambda.

### Note

Você pode ignorar essa etapa se seu projeto tiver sido provisionado com uma política de limite de permissões.

Para projetos criados após o PDT de 6 de dezembro de 2018, AWS CodeStar provisionou seu projeto com uma política de limite de permissões.

1. Faça login no AWS Management Console e abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.
2. Crie um projeto ou escolha um projeto existente com um `template.yml` file e abra a página Project resources (Recursos do projeto).
3. Em Recursos do projeto, localize a função do IAM criada para a função CodeStarWorker / Lambda na lista de recursos. O nome da função segue este formato: `role/CodeStarWorker-Project_name-lambda-Function_name`. Escolha o ARN da função.

4. A função é aberta no console do IAM. Escolha Anexar políticas. Procure a política `AWSXrayWriteOnlyAccess`, marque a caixa ao lado dela e escolha Attach Policy (Anexar política).

## Etapa 2: modificar o arquivo `template.yml` para rastreamento

1. Abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.
2. Escolha o projeto com tecnologia sem servidor e abra a página Code (Código). No nível mais alto do repositório, localize e edite o arquivo `template.yml`. Em Resources, cole o recurso na seção Properties.

Tracing: Active

Este exemplo mostra um modelo modificado:

```
Resources:
  GetHelloWorld:
    Type: AWS::Serverless::Function
    Properties:
      Handler: index.get
      Runtime: nodejs1.3
      Tracing: Active # Enable X-Ray tracing for the function
    Role:
      Fn::ImportValue:
        !Join ['-', [!Ref 'ProjectId', !Ref 'AWS::Region', 'LambdaTrustRole']]
    Events:
      GetEvent:
        Type: Api
        Properties:
          Path: /
          Method: get
```

## Etapa 3: confirmar e enviar a alteração do modelo para rastreamento

- Confirme e envie as alterações no arquivo `template.yml`.

### Note

Isso inicia o pipeline. Se você confirmar as alterações antes de atualizar as permissões do IAM, seu pipeline será iniciado, a atualização da AWS CloudFormation pilha encontrará erros e a atualização da pilha será revertida. Se isso acontecer, corrija as permissões e reinicie o pipeline.

## Etapa 4: monitorar a atualização da pilha do AWS CloudFormation para rastreamento

1. A atualização da AWS CloudFormation pilha começa quando o pipeline do seu projeto inicia a fase de implantação. Para ver o status da atualização da pilha, em seu AWS CodeStar painel, escolha o AWS CloudFormation estágio em seu funil.

Se a atualização da pilha AWS CloudFormation retornar erros, consulte as diretrizes de solução de problemas em [AWS CloudFormation: criação da pilha revertida para permissões ausentes](#). Caso as permissões da função de operador não sejam encontradas, edite a política anexada à função de operador Lambda do projeto. Consulte [Etapa 1: Editar a função de operador no IAM para rastreamento](#).

2. Use o painel para visualizar a conclusão bem-sucedida do pipeline. O rastreamento já está habilitado no aplicativo.
3. Verifique se o rastreamento está habilitado exibindo os detalhes da função no console do Lambda.
4. Escolha o endpoint do aplicativo do projeto. Essa interação com o aplicativo é rastreada. Você pode visualizar informações de rastreamento no console do AWS X-Ray .

Trace list					
ID	Age	Method	Response	Response time	URL
...315e2d41	4.7 min		200	270 ms	
...88c0c37c	12.8 sec		200	23.0 ms	

## Adicionar um recurso a um projeto

Cada AWS CodeStar modelo para todos os projetos vem com um AWS CloudFormation arquivo que modela as dependências de AWS tempo de execução do seu aplicativo, como tabelas de banco de dados e funções Lambda. Isso é armazenado no repositório de origem no arquivo `/template.yml`.

### Note

Você pode usar estas etapas para os seguintes projetos, criados com as seguintes alterações de suporte do projeto:

- Qualquer projeto do Lambda.

- Para projetos da Amazon EC2 ou do Elastic Beanstalk criados após 3 de agosto de AWS CodeStar 2018, `/template.yml` provisionei um arquivo no repositório do projeto.

Você pode modificar esse arquivo adicionando AWS CloudFormation recursos à `Resources` seção. A modificação do `template.yml` arquivo permite AWS CodeStar AWS CloudFormation adicionar o novo recurso ao seu projeto. Alguns recursos exigem que você adicione outras permissões à política para a função de CloudFormation trabalhador do seu projeto. Para obter mais informações sobre elementos de modelo e formatação, consulte [Referência de tipos de recurso do AWS](#).

Depois de determinar quais recursos você deve adicionar ao projeto, estas serão as etapas de alto nível a serem seguidas para personalizar um modelo. Para obter uma lista de AWS CloudFormation recursos e suas propriedades necessárias, consulte [Referência de tipos de AWS recursos](#).

1. [Etapa 1: editar a função do CloudFormation trabalhador no IAM](#) (se necessário)
2. [Etapa 2: modificar o arquivo `template.yml`](#)
3. [Etapa 3: confirmar e enviar a alteração do modelo](#)
4. [Etapa 4: monitorar a atualização da pilha do AWS CloudFormation](#)
5. [Etapa 5: adicionar permissões de recursos com uma política em linha](#)

Use as etapas desta seção para modificar seu modelo de AWS CodeStar projeto para adicionar um recurso e, em seguida, expandir as permissões da função de CloudFormation trabalhador do projeto no IAM. Neste exemplo, o `AWS::SQS::Queue` recurso é adicionado ao `template.yml` arquivo. A alteração inicia uma resposta automática AWS CloudFormation que adiciona uma fila do Amazon Simple Queue Service ao seu projeto.

## Etapa 1: editar a função do CloudFormation trabalhador no IAM

Você deve entrar como administrador para realizar as etapas 1 e 5.

### Note

Você pode ignorar essa etapa se seu projeto tiver sido provisionado com uma política de limite de permissões.

Para projetos criados após o PDT de 6 de dezembro de 2018, AWS CodeStar provisionou seu projeto com uma política de limite de permissões.

1. Faça login no AWS Management Console e abra o AWS CodeStar console, em <https://console.aws.amazon.com/codestar/>.
2. Crie um projeto ou escolha um projeto existente com um `template.yml` file e abra a página Project resources (Recursos do projeto).
3. Em Recursos do projeto, localize a função do IAM criada para a AWS CloudFormation função CodeStarWorker/na lista de recursos. O nome da função segue este formato: `role/CodeStarWorker-Project_name-CloudFormation`.
4. A função é aberta no console do IAM. Na guia Permissions (Permissões), em Inline Policies (Políticas em linha), expanda a linha da política da função de serviço e escolha Edit Policy (Editar política).
5. Escolha a guia JSON para editar a política.

 Note

A política anexada à função de operador é `CodeStarWorkerCloudFormationRolePolicy`.

6. No campo JSON, adicione a declaração de política a seguir no elemento Statement.

```
{
  "Action": [
    "sqs:CreateQueue",
    "sqs>DeleteQueue",
    "sqs:GetQueueAttributes",
    "sqs:SetQueueAttributes",
    "sqs:ListQueues",
    "sqs:GetQueueUrl"
  ],
  "Resource": [
    "*"
  ],
  "Effect": "Allow"
}
```

7. Escolha Review policy (Revisar política) para garantir que a política não contenha erros e selecione Save changes (Salvar alterações).

## Etapa 2: modificar o arquivo template.yml

1. Abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.
2. Escolha o projeto com tecnologia sem servidor e abra a página Code (Código). No nível superior do repositório, anote o local de template.yml.
3. Use um IDE, o console ou a linha de comando no repositório local para editar o arquivo template.yml no repositório. Cole o recurso na seção Resources. Neste exemplo, quando o texto a seguir é copiado, ele adiciona a seção Resources.

```
Resources:
  TestQueue:
    Type: AWS::SQS::Queue
```

Este exemplo mostra um modelo modificado:

```
Resources:
  HelloWorld:
    Type: AWS::Serverless::Function
    Properties:
      Handler: index.handler
      Runtime: python3.6
      Role:
        Fn::ImportValue:
          !Join ['-', [!Ref 'ProjectId', !Ref 'AWS::Region', 'LambdaTrustRole']]
    Events:
      GetEvent:
        Type: Api
        Properties:
          Path: /
          Method: get
      PostEvent:
        Type: Api
        Properties:
          Path: /
          Method: post
  TestQueue:
    Type: AWS::SQS::Queue
```

## Etapa 3: confirmar e enviar a alteração do modelo

- Confirme e envie as alterações no arquivo template.yml salvas na etapa 2.

### Note

Isso inicia o pipeline. Se você confirmar as alterações antes de atualizar as permissões do IAM, seu pipeline será iniciado e a atualização da AWS CloudFormation pilha encontrará erros, o que fará com que a atualização da pilha seja revertida. Se isso acontecer, corrija as permissões e reinicie o pipeline.

## Etapa 4: monitorar a atualização da pilha do AWS CloudFormation

- Quando o pipeline do seu projeto inicia a fase de implantação, a atualização da AWS CloudFormation pilha é iniciada. Você pode escolher o AWS CloudFormation estágio do seu funil no AWS CodeStar painel para ver a atualização da pilha.

### Solucionar problemas:

A atualização da pilha falhará se as permissões de recursos necessárias não forem encontradas. Visualize o status da falha na visualização AWS CodeStar do painel do pipeline do seu projeto.

Escolha o CloudFormationlink no estágio de implantação do seu pipeline para solucionar a falha no AWS CloudFormation console. No console, na lista Events (Eventos), escolha o projeto para visualizar detalhes de criação da pilha. Há uma mensagem com os detalhes da falha. Neste exemplo, a permissão `sqs:CreateQueue` não foi encontrada.

08:37:11 UTC-0700	UPDATE_ROLLBACK_COMPLETE	AWS::CloudFormation::Stack	awscodestar-dk-sqs-red-lambda	
08:37:11 UTC-0700	DELETE_COMPLETE	AWS::SQS::Queue	TestQueue	
08:37:09 UTC-0700	UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS	AWS::CloudFormation::Stack	awscodestar-dk-sqs-red-lambda	
08:37:06 UTC-0700	UPDATE_COMPLETE	AWS::Lambda::Function	HelloWorld	
08:37:03 UTC-0700	UPDATE_ROLLBACK_IN_PROGRESS	AWS::CloudFormation::Stack	awscodestar-dk-sqs-red-lambda	The following resource(s) failed to create: [TestQueue]. The following resource(s) failed to update: [HelloWorld].
08:37:02 UTC-0700	UPDATE_FAILED	AWS::Lambda::Function	HelloWorld	Resource update cancelled
08:37:01 UTC-0700	CREATE_FAILED	AWS::SQS::Queue	TestQueue	API: sqs:CreateQueue Access to the resource https://sqs.us-west-2.amazonaws.com/ is denied.
08:37:01 UTC-0700	CREATE_IN_PROGRESS	AWS::SQS::Queue	TestQueue	

Adicione todas as permissões ausentes editando a política anexada à função de AWS CloudFormation trabalhador do seu projeto. Consulte [Etapa 1: editar a função do CloudFormation trabalhador no IAM](#).

- Após a execução bem-sucedida do pipeline, os recursos são criados na pilha do AWS CloudFormation. Na lista Recursos em AWS CloudFormation, visualize o recurso criado para seu projeto. Neste exemplo, a TestQueue fila está listada na seção Recursos.

O URL da fila está disponível em AWS CloudFormation. O URL da fila segue este formato:

```
https://{REGION_ENDPOINT}/queue. |api-domain|/{YOUR_ACCOUNT_NUMBER}/  
{YOUR_QUEUE_NAME}
```

Para obter mais informações, consulte [Enviar uma mensagem do Amazon SQS](#), [Receber uma mensagem de uma fila do Amazon SQS](#) e [Excluir uma mensagem de uma fila do Amazon SQS](#).

## Etapa 5: adicionar permissões de recursos com uma política em linha

Conceda a membros da equipe acesso ao novo recurso adicionando a política em linha à função do usuário. Nem todos os recursos exigem que você adicione permissões. Para realizar as etapas a seguir, você deve entrar no console como um usuário raiz, um usuário do administrador na conta ou um usuário do IAM ou um usuário federado com a política gerenciada AdministratorAccess ou equivalente.

Para usar o editor de políticas JSON para criar uma política

1. Faça login no AWS Management Console e abra o console do IAM em <https://console.aws.amazon.com/iam/>.
2. No painel de navegação à esquerda, escolha Políticas.

Se essa for a primeira vez que você escolhe Políticas, a página Bem-vindo às políticas gerenciadas será exibida. Escolha Começar.

3. Na parte superior da página, escolha Criar política.
4. Na seção Editor de políticas, escolha a opção JSON.
5. Insira o seguinte documento de política JSON:

```
{  
  "Action": [  
    "sqs:CreateQueue",  
    "sqs>DeleteQueue",  
    "sqs:GetQueueAttributes",  
    "sqs:SetQueueAttributes",  
    "sqs:ListQueues",  
    "sqs:GetQueueUrl"  
  ],  
  "Resource": [  

```

```
    "*"
  ],
  "Effect": "Allow"
}
```

## 6. Escolha Próximo.

### Note

É possível alternar entre as opções de editor Visual e JSON a qualquer momento. Porém, se você fizer alterações ou escolher Próximo no editor Visual, o IAM poderá reestruturar a política a fim de otimizá-la para o editor visual. Para obter mais informações, consulte [Reestruturação de política](#) no Guia do usuário do IAM.

7. Na página Revisar e criar, insira um Nome de política e uma Descrição (opcional) para a política que você está criando. Revise Permissões definidas nessa política para ver as permissões que são concedidas pela política.
8. Escolha Criar política para salvar sua nova política.

## Adicionar uma função do IAM a um projeto

Desde 6 de dezembro de 2018, horário do Pacífico você pode definir suas próprias funções e políticas na pilha do aplicativo (template.yml). Para reduzir os riscos de escalonamento de privilégios e ações destrutivas, você deve definir o limite de permissões específicas ao projeto para cada entidade do IAM que criar. Se você tiver um projeto do Lambda com várias funções, uma melhor prática é criar um perfil do IAM para cada função.

Para adicionar uma função do IAM ao projeto

1. Edite o arquivo `template.yml` de seu projeto.
2. Na seção `Resources:`, adicione o recurso do IAM usando o formato no exemplo a seguir:

```
SampleRole:
  Description: Sample Lambda role
  Type: AWS::IAM::Role
  Properties:
    AssumeRolePolicyDocument:
      Statement:
```

```

- Effect: Allow
Principal:
  Service: [lambda.amazonaws.com]
Action: sts:AssumeRole
ManagedPolicyArns:
  - arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole
PermissionsBoundary: !Sub 'arn:${AWS::Partition}:iam::${AWS::AccountId}:policy/CodeStar_${ProjectId}_PermissionsBoundary'

```

3. Libere suas alterações por meio do pipeline e verifique o sucesso.

## Adicionar um estágio e uma endpoint de produção a um projeto

Use os procedimentos desta seção para adicionar um novo estágio de produção (Prod) ao pipeline e um estágio de aprovação manual entre os estágios de implantação e produção do pipeline. Isso cria uma pilha de recursos adicionais quando o pipeline do projeto é executado.

### Note

Você pode usar esses procedimentos se:

- Para projetos criados após 3 de agosto de 2018, AWS CodeStar provisionou seu projeto Amazon EC2, Elastic Beanstalk ou Lambda com um arquivo no repositório do projeto. /`template.yml`
- Para projetos criados após o PDT de 6 de dezembro de 2018, AWS CodeStar provisionou seu projeto com uma política de limite de permissões.

Todos os AWS CodeStar projetos usam um arquivo AWS CloudFormation de modelo que modela as dependências de AWS tempo de execução do seu aplicativo, como instâncias Linux e funções Lambda. O arquivo /`template.yml` é armazenado em seu repositório de origem.

No arquivo /`template.yml`, use o parâmetro `Stage` para adicionar uma pilha de recursos para um novo estágio no pipeline do projeto.

`Stage:`

`Type:` String

`Description:` The name for a project pipeline stage, such as Staging or Prod, for which resources are provisioned and deployed.

```
Default: ''
```

O parâmetro `Stage` é aplicado a todos os recursos nomeados com o ID de projeto referenciado no recurso. Por exemplo, o seguinte nome de função é um recurso nomeado no modelo:

```
RoleName: !Sub 'CodeStar-${ProjectId}-WebApp${Stage}'
```

## Pré-requisitos

Use as opções de modelo no AWS CodeStar console para criar um projeto.

Certifique-se de que o usuário do IAM tenha as permissões a seguir:

- `iam:PassRoles` sobre a AWS CloudFormation função do projeto.
- `iam:PassRole` na função da cadeia de ferramentas do projeto.
- `cloudformation:DescribeStacks`
- `cloudformation:ListChangeSets`

Somente para projetos do Elastic Beanstalk EC2 ou da Amazon:

- `codedeploy:CreateApplication`
- `codedeploy:CreateDeploymentGroup`
- `codedeploy:GetApplication`
- `codedeploy:GetDeploymentConfig`
- `codedeploy:GetDeploymentGroup`
- `elasticloadbalancing:DescribeTargetGroups`

## Tópicos

- [Etapa 1: criar um novo grupo de implantação em CodeDeploy \(somente EC2 projetos da Amazon\)](#)
- [Etapa 2: Adicionar um novo estágio de pipeline para o estágio de produção](#)
- [Etapa 3: Adicionar um estágio de aprovação manual](#)
- [Etapa 4: Fazer uma alteração e monitorar a atualização da AWS CloudFormation pilha](#)

## Etapa 1: criar um novo grupo de implantação em CodeDeploy (somente EC2 projetos da Amazon)

Você escolhe seu CodeDeploy aplicativo e, em seguida, adiciona um novo grupo de implantação associado à nova instância.

### Note

Se seu projeto for um projeto do Lambda ou do Elastic Beanstalk, você poderá ignorar essa etapa.

1. Abra o CodeDeploy console em <https://console.aws.amazon.com/codedeploy>.
2. Escolha o CodeDeploy aplicativo que foi gerado para seu projeto quando ele foi criado em AWS CodeStar.
3. Em Deployment groups, escolha Create deployment group.
4. Em Nome do grupo de implantação, insira **<project-id>-prod-Env**.
5. Em Função de serviço, escolha a função de trabalhador da cadeia de ferramentas para seu AWS CodeStar projeto.
6. Em Deployment type (Tipo de implantação), selecione In-place (No local).
7. Em Configuração do ambiente, escolha a guia Amazon EC2 Instances.
8. No grupo de tags, em Key (Chave), escolha `aws:cloudformation:stack-name`. Em Valor, escolha `awscodestar-<projectid>-infrastructure-prod` (a pilha a ser criada para a GenerateChangeSetação).
9. Em Deployment settings (Configurações da implantação), escolha `CodeDeployDefault.AllAtOnce`.
10. Limpe Choose a load balancer (Escolher um load balancer).
11. Selecione Criar grupo de implantação.

Agora, o segundo grupo de implantação foi criado.

## Etapa 2: Adicionar um novo estágio de pipeline para o estágio de produção

Adicione um estágio com o mesmo conjunto de ações de implantação como o estágio de implantação do projeto. Por exemplo, o novo estágio Prod de um EC2 projeto da Amazon deve ter as mesmas ações do estágio Deploy criado para o projeto.

Para copiar parâmetros e campos do estágio de implantação

1. No painel do seu AWS CodeStar projeto, escolha Detalhes do funil para abrir seu funil no CodePipeline console.
2. Selecione Editar.
3. No estágio de implantação, escolha Edit stage (Editar estágio).
4. Escolha o ícone de edição na GenerateChangeSetação. Anote os valores dos seguintes campos. Você usa esses valores ao criar a nova ação.
  - Nome da stack
  - Change set name (Nome do conjunto de alterações)
  - Modelo
  - Configuração do modelo
  - Input artifacts (Artefatos de entrada)
5. Expanda Advanced (Avançado) e, em Parameters (Parâmetros), copie os parâmetros de seu projeto. Você cola esses parâmetros em sua nova ação. Por exemplo, copie os parâmetros mostrados aqui em formato JSON:

- Projetos Lambda:

```
{
  "ProjectId": "MyProject"
}
```

- EC2 Projetos da Amazon:

```
{
  "ProjectId": "MyProject",
  "InstanceType": "t2.micro",
  "WebAppInstanceProfile": "awscodestar-MyProject-WebAppInstanceProfile-EXAMPLEY5VSFS",
}
```

```
"ImageId": "ami-EXAMPLE1",  
"KeyPairName": "my-keypair",  
"SubnetId": "subnet-EXAMPLE",  
"VpcId": "vpc-EXAMPLE1"  
}
```

- Projetos do Elastic Beanstalk

```
{  
  "ProjectId": "MyProject",  
  "InstanceType": "t2.micro",  
  "KeyPairName": "my-keypair",  
  "SubnetId": "subnet-EXAMPLE",  
  "VpcId": "vpc-EXAMPLE",  
  "SolutionStackName": "64bit Amazon Linux 2018.03 v3.0.5 running Tomcat 8 Java  
8",  
  "EBTrustRole": "CodeStarWorker-myproject-EBService",  
  "EBInstanceProfile": "awscodestar-myproject-EBInstanceProfile-11111EXAMPLE"  
}
```

6. No painel de edição do estágio, escolha Cancel (Cancelar).

Para criar uma GenerateChangeSet ação em seu novo estágio de produção

**Note**

Depois de adicionar a nova ação e enquanto ainda estiver no modo de edição, se você reabrir a nova ação para edição, alguns dos campos poderão não ser exibidos. Você também poderá ver o seguinte: A pilha stack-name não existe  
Esse erro não impede que você salve o pipeline. No entanto, para restaurar os campos ausentes, é necessário excluir a nova ação e adicioná-la novamente. Depois de salvar e executar o pipeline, a pilha é reconhecida e o erro não aparece novamente.

1. Se seu funil ainda não estiver exibido, no painel do AWS CodeStar projeto, escolha Detalhes do funil para abrir seu funil no console.
2. Selecione Editar.
3. Na parte inferior do diagrama, escolha + Add stage (+ Adicionar estágio).

4. Insira um nome de estágio (por exemplo, **Prod**) e escolha + Add action group (+ Adicionar grupo de ações).
5. Em Action name (Nome da ação), informe um nome (por exemplo, **GenerateChangeSet**).
6. Em Provedor de ação, selecione AWS CloudFormation.
7. Em Action mode (Modo de ação), escolha Create or replace a change set (Criar ou substitui um conjunto de alterações).
8. Em Nome da pilha, insira um novo nome para a AWS CloudFormation pilha que será criada por essa ação. Comece com um nome idêntico ao nome da pilha de implantação e, em seguida, adicione **-prod**:
  - Projetos Lambda: `awscodestar-<project_name>-lambda-prod`
  - Projetos da Amazon EC2 e do Elastic Beanstalk: `awscodestar-<project_name>-infrastructure-prod`

 Note

Para que não haja falha, o nome da pilha deve começar exatamente com **awscodestar-<project\_name>-**.

9. Em Change set name (Nome do conjunto de alterações), insira o mesmo nome do conjunto de alterações fornecido no estágio de implantação existente (por exemplo, **pipeline-changeset**).
10. Em Input artifacts (Artefatos de entrada), escolha o artefato de criação.
11. Em Template (Modelo), insira o mesmo nome do modelo de alterações fornecido no estágio de implantação existente (por exemplo, **<project-ID>-BuildArtifact::template.yml**).
12. Em Template configuration (Configuração de modelo), insira o mesmo arquivo de configuração do modelo de alterações fornecido no estágio de implantação existente (por exemplo, **<project-ID>-BuildArtifact::template-configuration.json**).
13. Em Capabilities (Recursos), escolha `CAPABILITY_NAMED_IAM`.
14. Em Role name (Nome da função), escolha o nome da função de operador do AWS CloudFormation de seu projeto.
15. Expanda Advanced (Avançado) e, em Parameters (Parâmetros), cole os parâmetros de seu projeto. Inclua o Stage parâmetro, mostrado aqui no formato JSON, para um EC2 projeto da Amazon:

```
{  
  
  "ProjectId": "MyProject",  
  "InstanceType": "t2.micro",  
  "WebAppInstanceProfile": "awscodestar-MyProject-WebAppInstanceProfile-  
EXAMPLEY5VSFS",  
  "ImageId": "ami-EXAMPLE1",  
  "KeyPairName": "my-keypair",  
  "SubnetId": "subnet-EXAMPLE",  
  "VpcId": "vpc-EXAMPLE1",  
  "Stage": "Prod"  
}
```

 Note

Certifique-se de colar todos os parâmetros do projeto, não apenas os novos parâmetros ou os parâmetros que você deseja alterar.

16. Escolha Salvar.

17. No AWS CodePipeline painel, escolha Salvar alteração do pipeline e, em seguida, escolha Salvar alteração.

 Note

Pode ser exibida uma mensagem notificando você sobre a exclusão e adição de recursos de detecção de alterações. Reconheça a mensagem e continue na próxima etapa deste tutorial.

Visualize seu pipeline atualizado.

Para criar uma ExecuteChangeSet ação em seu novo estágio de produção

1. Se você ainda não está visualizando seu funil, no painel do AWS CodeStar projeto, escolha Detalhes do funil para abrir seu funil no console.
2. Selecione Editar.

3. Em seu novo estágio Prod, após a nova GenerateChangeSetação, escolha + Adicionar grupo de ação.
4. Em Action name (Nome da ação), informe um nome (por exemplo, **ExecuteChangeSet**).
5. Em Provedor de ação, selecione AWS CloudFormation.
6. Em Action mode (Modo de ação), escolha Execute a change set (Executar um conjunto de alterações).
7. Em Nome da pilha, insira o novo nome da AWS CloudFormation pilha que você inseriu na GenerateChangeSet ação (por exemplo, **awscodestar-`<project-ID>`-infrastructure-prod**).
8. Em Alterar nome do conjunto, insira o mesmo nome do conjunto usado no estágio de implantação (por exemplo, **pipeline-changeset**).
9. Selecione Concluído.
10. No AWS CodePipeline painel, escolha Salvar alteração do pipeline e, em seguida, escolha Salvar alteração.

 Note

Pode ser exibida uma mensagem notificando você sobre a exclusão e adição de recursos de detecção de alterações. Reconheça a mensagem e continue na próxima etapa deste tutorial.

Visualize seu pipeline atualizado.

Para criar uma ação de CodeDeploy implantação em seu novo estágio de produção (somente EC2 projetos da Amazon)

1. Depois das novas ações no estágio de produção, selecione + Action (+ Ação).
2. Em Action name (Nome da ação), informe um nome (por exemplo, **Deploy**).
3. Em Provedor de ação, selecione AWS CodeDeploy.
4. Em Nome do aplicativo, escolha o nome do CodeDeploy aplicativo para seu projeto.
5. Em Deployment group (Grupo de implantação), escolha o nome do novo grupo de implantação do CodeDeploy que você criou na etapa 2.

6. Em Input artifacts (Artefatos de entrada), escolha o mesmo artefato de criação usado no estágio existente.
7. Selecione Concluído.
8. No AWS CodePipeline painel, escolha Salvar alteração do pipeline e, em seguida, escolha Salvar alteração. Visualize seu pipeline atualizado.

### Etapa 3: Adicionar um estágio de aprovação manual

Como uma melhor prática, adicione um estágio de aprovação manual na frente do novo estágio de produção.

1. No canto superior esquerdo, selecione Edit (Editar).
2. No diagrama do pipeline, entre os estágios de implantação e de produção, escolha + Add stage (+ Adicionar estágio).
3. Em Edit stage (Editar estágio), digite um nome de estágio (por exemplo, **Approval**) e escolha + Add action group (+ Adicionar grupo de ação).
4. Em Action name (Nome da ação), informe um nome (por exemplo, **Approval**).
5. Em Tipo de aprovação, selecione Aprovação manual.
6. (Opcional) Em Configuration (Configuração), em SNS Topic ARN (ARN do tópico do SNS), escolha o tópico do SNS que você criou e no qual se inscreveu.
7. Escolha Add Action.
8. No AWS CodePipeline painel, escolha Salvar alteração do pipeline e, em seguida, escolha Salvar alteração. Visualize seu pipeline atualizado.
9. Para enviar suas alterações e iniciar uma compilação do pipeline, selecione Liberar alteração e, depois, Liberar.

### Etapa 4: Fazer uma alteração e monitorar a atualização da AWS CloudFormation pilha

1. Enquanto seu pipeline está em execução, você pode usar as etapas aqui para acompanhar a criação da pilha e do endpoint para sua nova etapa.
2. Quando o pipeline inicia o estágio de implantação, a atualização da AWS CloudFormation pilha é iniciada. Você pode escolher o AWS CloudFormation estágio do seu funil no AWS CodeStar painel para ver a notificação de atualização da pilha. Para visualizar detalhes de criação da pilha, no console, escolha o projeto na lista Events (Eventos).

3. Após a conclusão bem-sucedida do seu pipeline, os recursos são criados em sua AWS CloudFormation pilha. No AWS CloudFormation console, escolha a pilha de infraestrutura para seu projeto. Os nomes de pilhas seguem este formato:
  - Projetos Lambda: `awscodestar-<project_name>-lambda-prod`
  - Projetos da Amazon EC2 e do Elastic Beanstalk: `awscodestar-<project_name>-infrastructure-prod`

Na lista Recursos no AWS CloudFormation console, visualize o recurso criado para seu projeto. Neste exemplo, a nova EC2 instância da Amazon aparece na seção Recursos.

4. Acesse o endpoint do estágio de produção:
  - Para um projeto do Elastic Beanstalk, abra a nova pilha no console e expanda Resources AWS CloudFormation . Escolha o aplicativo do Elastic Beanstalk. O link abre no console do Elastic Beanstalk. Escolha Environments (Ambientes). Escolha a URL em URL para abrir o endpoint em um navegador.
  - Para um projeto Lambda, abra a nova pilha no AWS CloudFormation console e expanda Recursos. Escolha o recurso do API Gateway. O link é aberto no console do API Gateway. Escolha Estágios. Escolha a URL em Invoke URL (Invocar URL) para abrir o endpoint em um navegador.
  - Para um EC2 projeto da Amazon, escolha a nova EC2 instância da Amazon na lista de recursos do projeto no AWS CodeStar console. O link é aberto na página Instância do EC2 console da Amazon. Escolha a guia Descrição, copie a URL no DNS público (IPv4) e abra a URL em um navegador.
5. Verifique se a alteração foi implantada.

## Use com segurança os parâmetros SSM em um projeto AWS CodeStar

Vários clientes armazenam segredos, como credenciais, em parâmetros do [Repositório de parâmetros do Gerenciador de Sistemas](#). Agora você pode usar esses parâmetros com segurança em um AWS CodeStar projeto. Por exemplo, talvez você queira usar parâmetros SSM em sua especificação de compilação para CodeBuild ou ao definir recursos do aplicativo em sua pilha de conjuntos de ferramentas (template.yml).

Para usar parâmetros de SSM em um CodeStar projeto da AWS, você deve marcar manualmente os parâmetros com o ARN do CodeStar projeto da AWS. Você também deve fornecer permissões

apropriadas à função de trabalhador da CodeStar cadeia de ferramentas da AWS para acessar os parâmetros que você marcou.

## Antes de começar

- [Crie um novo](#) ou identifique um parâmetro do Systems Manager existente que contenha as informações que deseja acessar.
- Identifique qual CodeStar projeto da AWS você deseja usar ou [crie um novo projeto](#).
- Anote o ARN do CodeStar projeto. Ele é semelhante a: `arn:aws:codestar:region-id:account-id:project/project-id`.

## Marque um parâmetro com o ARN do CodeStar projeto da AWS

Consulte [Marcação de parâmetros do gerenciador de sistemas](#) para obter instruções passo a passo.

1. Em Chave, insira `awscodestar:projectArn`.
2. Em Valor, insira o ARN do projeto de CodeStar: `arn:aws:codestar:region-id:account-id:project/project-id`
3. Escolha Salvar.

Agora você pode fazer referência ao parâmetro SSM em seu arquivo `template.yml`. Se quiser usá-lo com uma função de operador da cadeia de ferramentas, será necessário conceder permissões adicionais.

## Conceda permissões para usar parâmetros marcados em seu conjunto de ferramentas de CodeStar projetos da AWS

### Note

Estas etapas se aplicam somente a projetos criados após 6 de dezembro de 2018, horário do Pacífico.

1. Abra o painel do CodeStar projeto da AWS para o projeto que você deseja usar.
2. Clique em Project (Projeto) para visualizar a lista de recursos criados e localize a função de operador da cadeia de ferramentas. É um recurso do IAM com um nome no formato: `role/CodeStarWorker-project-id-ToolChain`.

3. Clique no ARN para abri-lo no console do IAM.
4. Localize o ToolChainWorkerPolicy e expanda-o, se necessário.
5. Clique em Edit Policy (Editar política).
6. Em Action:, adicione a seguinte linha:

```
ssm:GetParameter*
```

7. Clique em Review policy (Revisar política) e clique em Save changes (Salvar alterações).

Para projetos criados antes de 6 de dezembro de 2018, horário do Pacífico será necessário adicionar as seguintes permissões às funções de operador para cada serviço.

```
{
  "Action": [
    "ssm:GetParameter*"
  ],
  "Resource": "*",
  "Effect": "Allow",
  "Condition": {
    "StringEquals": {
      "ssm:ResourceTag/awscodestar:projectArn": "arn:aws:codestar:region-id:account-id:project/project-id"
    }
  }
}
```

## Mudar tráfego para um projeto do AWS Lambda

AWS CodeDeploy suporta implantações de versões de AWS Lambda funções para funções em seus projetos AWS CodeStar sem servidor. Uma AWS Lambda implantação transfere o tráfego de entrada de uma função Lambda existente para uma versão atualizada da função Lambda. Convém testar uma função do Lambda atualizada implantando uma versão separada e revertendo a implantação para a primeira versão, se necessário.

Use as etapas desta seção para modificar seu modelo de AWS CodeStar projeto e atualizar as permissões do IAM de suas CodeStarWorker funções. Essa tarefa inicia uma resposta automática

AWS CloudFormation que cria AWS Lambda funções com alias e, em seguida, instrui AWS CodeDeploy a transferir o tráfego para um ambiente atualizado.

 Note

Conclua essas etapas somente se você criou seu CodeStar projeto da AWS antes de 12 de dezembro de 2018.

AWS CodeDeploy tem três opções de implantação que permitem transferir o tráfego para versões de sua AWS Lambda função em seu aplicativo:

- **Canário:** o tráfego é deslocado em dois incrementos. Você pode escolher entre opções do canary predefinidas que especificam a porcentagem de tráfego deslocada para a versão da função do Lambda atualizada no primeiro incremento e no intervalo, em minutos, antes que o tráfego restante seja deslocado no segundo incremento.
- **Linear:** o tráfego é deslocado em incrementos iguais com um número igual de minutos entre cada incremento. Você pode escolher entre opções lineares predefinidas que especificam a porcentagem de tráfego deslocado em cada incremento e o número de minutos entre cada incremento. O tráfego é deslocado em incrementos iguais com um número igual de minutos entre cada incremento. Você pode escolher entre opções lineares predefinidas que especificam a porcentagem de tráfego deslocado em cada incremento e o número de minutos entre cada incremento.
- **Rll-at-once:** Todo o tráfego é transferido da função Lambda original para a versão atualizada da função Lambda de uma só vez.

#### Tipo de preferência de implantação

Canary10Percent30Minutes

Canary10Percent5Minutes

Canary10Percent10Minutes

Canary10Percent15Minutes

PercentEveryLinear 10 10 minutos

## Tipo de preferência de implantação

PercentEveryLinear 10 1 minuto

PercentEveryLinear 10 2 minutos

PercentEveryLinear 10 3 minutos

AllAtOnce

Para obter mais informações sobre AWS CodeDeploy implantações em uma plataforma de AWS Lambda computação, consulte [Implantações em uma plataforma de computação Lambda AWS](#).

Para obter mais informações sobre o AWS SAM, consulte [AWS Serverless Application Model \(AWS SAM\)](#) em GitHub

### Pré-requisitos:

Quando você cria um projeto com tecnologia sem servidor, selecione qualquer modelo com a plataforma de computação Lambda. Você deve entrar como administrador para realizar as etapas de 4 a 6.

Etapa 1: Modificar o modelo do SAM para adicionar parâmetros de implantação da AWS Lambda versão

1. Abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.
2. Crie um projeto ou escolha um projeto existente com um arquivo `template.yml` e abra a página Code (Código). No nível mais alto do repositório, observe o local do modelo SAM chamado `template.yml` a ser modificado.
3. Abra o arquivo `template.yml` no IDE ou o repositório local. Copie o texto a seguir para adicionar uma seção `Globals` ao arquivo. O texto de exemplo neste tutorial escolhe a opção `Canary10Percent5Minutes`.

```
Globals:
  Function:
    AutoPublishAlias: live
    DeploymentPreference:
      Enabled: true
      Type: Canary10Percent5Minutes
```

Este exemplo mostra um modelo modificado após a adição da seção `Globals`:

```
AWSTemplateFormatVersion: 2010-09-09
Transform:
- AWS::Serverless-2016-10-31
- AWS::CodeStar

Parameters:
  ProjectId:
    Type: String
    Description: CodeStar projectId used to associate new resources to team members

Globals:
  Function:
    AutoPublishAlias: live
    DeploymentPreference:
      Enabled: true
      Type: Canary10Percent5Minutes

Resources:
  HelloWorld:
    Type: AWS::Serverless::Function
    Properties:
      Handler: index.handler
      Runtime: python3.6
      Role:
        Fn::ImportValue:
          !Join ['-', [!Ref 'ProjectId', !Ref 'AWS::Region', 'LambdaTrustRole']]
      Events:
```

Para obter mais informações, consulte o guia de referência [Seções globais](#) dos modelos SAM.

Etapa 2: editar a AWS CloudFormation função para adicionar permissões

1. Faça login no AWS Management Console e abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.

#### Note

Você deve fazer login AWS Management Console usando as credenciais associadas ao usuário do IAM que você criou ou identificou. [Configuração AWS CodeStar](#) Esse usuário deve ter a política AWS gerenciada chamada **AWSCodeStarFullAccess** anexada.

2. Escolha o projeto com tecnologia sem servidor existente e abra a página Project resources (Recursos do projeto).
3. Em Recursos, escolha a função do IAM criada para a AWS CloudFormation função CodeStarWorker /. A função é aberta no console do IAM.
4. Na aba Permissions, em Inline Policies, na linha da sua política da função de serviço, selecione Edit Policy. Escolha a guia JSON para editar a política no formato JSON.

 Note

Sua função de serviço é chamada `CodeStarWorkerCloudFormationRolePolicy`.

5. No campo JSON, adicione as declarações de política dentro do elemento `Statement`. Substitua os *id* espaços reservados *region* e por sua região e ID da conta.

```
{
  "Action": [
    "s3:GetObject",
    "s3:GetObjectVersion",
    "s3:GetBucketVersioning"
  ],
  "Resource": "*",
  "Effect": "Allow"
},
{
  "Action": [
    "s3:PutObject"
  ],
  "Resource": [
    "arn:aws:s3:::codepipeline*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "lambda:*"
  ],
  "Resource": [
    "arn:aws:lambda:region:id:function:*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "apigateway:*"
  ],
  "Resource": [
    "arn:aws:apigateway:region::*"
  ],
  "Effect": "Allow"
}
```

```
"Effect": "Allow"
},
{
  "Action": [
    "iam:GetRole",
    "iam:CreateRole",
    "iam>DeleteRole",
    "iam:PutRolePolicy"
  ],
  "Resource": [
    "arn:aws:iam::id:role/*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "iam:AttachRolePolicy",
    "iam>DeleteRolePolicy",
    "iam:DetachRolePolicy"
  ],
  "Resource": [
    "arn:aws:iam::id:role/*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "iam:PassRole"
  ],
  "Resource": [
    "*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "codedeploy:CreateApplication",
    "codedeploy>DeleteApplication",
    "codedeploy:RegisterApplicationRevision"
  ],
  "Resource": [
    "arn:aws:codedeploy:region:id:application:*"
  ],
  "Effect": "Allow"
}
```

```

},
{
  "Action": [
    "codedeploy:CreateDeploymentGroup",
    "codedeploy:CreateDeployment",
    "codedeploy>DeleteDeploymentGroup",
    "codedeploy:GetDeployment"
  ],
  "Resource": [
    "arn:aws:codedeploy:region:id:deploymentgroup:*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "codedeploy:GetDeploymentConfig"
  ],
  "Resource": [
    "arn:aws:codedeploy:region:id:deploymentconfig:*"
  ],
  "Effect": "Allow"
}

```

6. Selecione Review policy (Revisar política) para garantir que a política não contenha erros. Quando a política não tiver erros, selecione Save changes (Salvar alterações).

Etapa 3: confirme e envie sua alteração de modelo para iniciar a mudança de AWS Lambda versão

1. Confirme e envie as alterações no arquivo `template.yml` salvas na etapa 1.

#### Note

Isso inicia o pipeline. Se você confirmar as alterações antes de atualizar as permissões do IAM, seu pipeline será iniciado e a atualização da AWS CloudFormation pilha encontrará erros que reverterem a atualização da pilha. Se isso acontecer, reinicie o pipeline depois que as permissões forem corrigidas.

2. A atualização da AWS CloudFormation pilha começa quando o pipeline do seu projeto inicia a fase de implantação. Para ver a notificação de atualização da pilha quando a implantação começar, em seu AWS CodeStar painel, selecione o AWS CloudFormation estágio em seu pipeline.

Durante a atualização da pilha, atualiza AWS CloudFormation automaticamente os recursos do projeto da seguinte forma:

- AWS CloudFormation processa o `template.yml` arquivo criando funções, ganchos de eventos e recursos do Lambda com alias.
- AWS CloudFormation chama o Lambda para criar a nova versão da função.
- AWS CloudFormation cria um AppSpec arquivo e chama AWS CodeDeploy para mudar o tráfego.

Para obter mais informações sobre como publicar funções do Lambda com o alias no SAM, consulte a referência de modelo [Modelo de Aplicação de tecnologia sem servidor do AWS](#). Para obter mais informações sobre ganchos de eventos e recursos no AWS CodeDeploy AppSpec arquivo, consulte a [seção AppSpec 'recursos' \(somente implantações AWS Lambda\)](#) [AppSpec e a seção 'ganchos' para](#) uma implantação Lambda. AWS

3. Após a conclusão bem-sucedida do pipeline, os recursos são criados na pilha do AWS CloudFormation . Na página Projeto, na lista Recursos do Projeto, visualize o AWS CodeDeploy aplicativo, o grupo de AWS CodeDeploy implantação e os recursos da função de AWS CodeDeploy serviço criados para seu projeto.
4. Para criar uma nova versão, faça uma alteração na função do Lambda no repositório. A nova implantação começa e migra o tráfego de acordo com o tipo de implantação indicado no modelo do SAM. Para visualizar o status do tráfego que está sendo migrado para a nova versão, na página Project (Projeto), na lista Project Resources (Recursos do projeto), escolha a implantação do AWS CodeDeploy .
5. Para ver detalhes sobre cada revisão, em Revisões, escolha o link para o grupo de AWS CodeDeploy implantação.
6. Em seu diretório de trabalho local, você pode fazer alterações em sua AWS Lambda função e confirmar a alteração no repositório do projeto. AWS CloudFormation suporta AWS CodeDeploy o gerenciamento da próxima revisão da mesma forma. Para obter mais informações sobre a reimplantação, interrupção ou reversão de uma implantação do Lambda, consulte [Implantações em uma AWS](#) plataforma de computação Lambda.

# Faça a transição do seu CodeStar projeto da AWS para produção

Depois de criar seu aplicativo usando um CodeStar projeto da AWS e ver o que a AWS CodeStar fornece, talvez você queira fazer a transição do seu projeto para uso em produção. Uma forma de fazer isso é replicar os AWS recursos do seu aplicativo fora da AWS CodeStar. Você ainda precisará de um repositório, um projeto de construção, um pipeline e uma implantação, mas em vez de fazer com que a AWS os CodeStar crie para você, você os recriará usando AWS CloudFormation

## Note

Pode ser útil criar ou visualizar um projeto semelhante usando primeiro um dos CodeStar Quick Starts da AWS e usá-lo como modelo para seu próprio projeto para garantir que você inclua os recursos e as políticas de que precisa.

Um CodeStar projeto da AWS é uma combinação do código-fonte e dos recursos criados para implantar o código. A coleção de recursos que ajudam a compilar, lançar e implantar o código são chamados de recursos da cadeia de ferramentas. Na criação do projeto, um AWS CloudFormation modelo provisiona os recursos do seu conjunto de ferramentas em um pipeline integration/continuous deployment (CI/CD (contínuo)).

Quando você usa o console para criar um projeto, o modelo da cadeia de ferramentas é criado para você. Ao usar o AWS CLI para criar um projeto, você cria o modelo do conjunto de ferramentas que cria os recursos do conjunto de ferramentas.

Uma cadeia de ferramentas completa requer os seguintes recursos recomendados:

1. Um GitHub repositório CodeCommit ou que contém seu código-fonte.
2. Um CodePipeline pipeline configurado para ouvir as alterações no seu repositório.
  - a. Quando você usa CodeBuild a AWS para executar testes unitários ou de integração, recomendamos que você adicione um estágio de construção ao seu pipeline para criar artefatos de construção.
  - b. Recomendamos que você adicione um estágio de implantação ao seu pipeline que use CodeDeploy ou AWS CloudFormation implante seu artefato de construção e código-fonte em sua infraestrutura de tempo de execução.

**Note**

Como CodePipeline requer pelo menos dois estágios em um pipeline, e o primeiro estágio deve ser o estágio de origem, adicione um estágio de construção ou implantação como segundo estágio.

## Tópicos

- [Criar um GitHub repositório](#)

## Criar um GitHub repositório

Você cria um GitHub repositório definindo-o em seu modelo de conjunto de ferramentas. Certifique-se de que já tenha criado um local para um arquivo ZIP contendo o código-fonte, para que o código possa ser carregado no repositório. Além disso, você já deve ter criado um token de acesso pessoal para que GitHub AWS possa se conectar GitHub em seu nome. Além do token de acesso pessoal GitHub, você também deve ter `s3:GetObject` permissão para o Code objeto que você entregou.

Para especificar um GitHub repositório público, adicione um código como o seguinte ao seu modelo de conjunto de ferramentas em `AWS CloudFormation`

```
GitHubRepo:
  Condition: CreateGitHubRepo
  Description: GitHub repository for application source code
  Properties:
    Code:
      S3:
        Bucket: MyCodeS3Bucket
        Key: MyCodeS3BucketKey
    EnableIssues: true
    IsPrivate: false
    RepositoryAccessToken: MyGitHubPersonalAccessToken
    RepositoryDescription: MyAppCodeRepository
    RepositoryName: MyAppSource
    RepositoryOwner: MyGitHubUserName
  Type: AWS::CodeStar::GitHubRepository
```

Esse código especifica as seguintes informações:

- O local do código que você deseja incluir, que deve ser um bucket do Amazon S3.
- Se você deseja habilitar problemas no GitHub repositório.
- Se o GitHub repositório é privado.
- O token de acesso GitHub pessoal que você criou.
- Uma descrição, um nome e um proprietário para o repositório que você está criando.

Para obter detalhes completos sobre quais informações especificar, consulte [AWS::CodeStar::GitHubRepositório](#) no Guia do AWS CloudFormation Usuário.

## Trabalhando com tags de projeto em AWS CodeStar

Você pode associar tags com projetos no AWS CodeStar. As tags podem ajudá-lo a gerenciar seus projetos. Por exemplo, convém adicionar uma tag com uma chave de Release e um valor de Beta a qualquer projeto no qual sua organização esteja trabalhando para uma versão beta.

### Adicionar uma tag a um projeto

1. Com o projeto aberto no AWS CodeStar console, no painel de navegação lateral, escolha Configurações.
2. Em Tags, escolha Editar.
3. Em Chave, insira um nome para a tag. Em Value (Valor), digite o valor da tag.
4. Opcional: Escolha Adicionar tag para adicionar mais tags.
5. Quando terminar de adicionar tags, escolha Salvar.

### Remover uma tag de um projeto

1. Com o projeto aberto no AWS CodeStar console, no painel de navegação lateral, escolha Configurações.
2. Em Tags, escolha Editar.
3. Em Tags, encontre a tag que deseja remover e selecione Remover tag.
4. Escolha Salvar.

## Obter uma lista de tags de um projeto

Use o AWS CLI para executar o AWS CodeStar `list-tags-for-project` comando, especificando o nome do projeto:

```
aws codestar list-tags-for-project --id my-first-projec
```

Se bem-sucedido, uma lista de tags aparecerá na saída, semelhante à seguinte:

```
{
  "tags": {
    "Release": "Beta"
  }
}
```

## Excluir um AWS CodeStar projeto

Se você não precisar mais de um projeto, você pode excluir o projeto e seus recursos, para que ele não seja mais cobrado na AWS. Quando você exclui um projeto, todos os membros da equipe são removidos desse projeto. Seus papéis de projeto são removidos dos usuários do IAM, mas seus perfis de usuário não AWS CodeStar são alterados. Você pode usar o AWS CodeStar console ou AWS CLI excluir um projeto. A exclusão de um projeto requer a função AWS CodeStar de serviço, `aws-codestar-service-role`, que não deve ser modificada e pode ser assumida por. AWS CodeStar

### Important

A exclusão de um projeto em AWS CodeStar não pode ser desfeita. Por padrão, todos os AWS recursos do projeto são excluídos AWS da sua conta, incluindo:

- O CodeCommit repositório do projeto junto com qualquer coisa armazenada nesse repositório.
- As funções do AWS CodeStar projeto e as políticas associadas do IAM configuradas para o projeto e seus recursos.
- Qualquer EC2 instância da Amazon criada para o projeto.
- O aplicativo de implantação e os recursos associados, como:
  - Um CodeDeploy aplicativo e grupos de implantação associados.

- Uma AWS Lambda função e o API Gateway associado APIs.
- Um AWS Elastic Beanstalk aplicativo e um ambiente associado.
- O pipeline de implantação contínua do projeto em CodePipeline.
- As AWS CloudFormation pilhas associadas ao projeto.
- Qualquer ambiente de AWS Cloud9 desenvolvimento criado com o AWS CodeStar console. Todas as alterações de código não confirmadas nos ambientes são perdidas.

Para excluir todos os recursos do projeto junto com o projeto, marque a caixa de seleção Excluir recursos. Se você desmarcar essa opção, o projeto será excluído AWS CodeStar e os papéis do projeto que permitiram o acesso a esses recursos serão excluídos no IAM, mas todos os outros recursos serão mantidos. Você pode continuar cobrando por esses recursos em AWS. Se decidir que não deseja mais um ou mais desses recursos, você deverá excluí-los manualmente. Para obter mais informações, consulte [Exclusão do projeto: um AWS CodeStar projeto foi excluído, mas os recursos ainda existem](#).

Se você decidir manter os recursos ao excluir um projeto, como uma prática recomendada, copie a lista de recursos na página de detalhes do projeto. Dessa forma, você tem um registro de todos os recursos que você manteve, embora o projeto não exista mais.

## Tópicos

- [Excluir um projeto no AWS CodeStar \(console\)](#)
- [Excluir um projeto em AWS CodeStar \(AWS CLI\)](#)

## Excluir um projeto no AWS CodeStar (console)

Você pode usar o AWS CodeStar console para excluir um projeto.

Para excluir um projeto no AWS CodeStar

1. Abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.
2. Selecione Projetos no painel de navegação.
3. Selecione o projeto que deseja excluir e selecione Excluir.

Ou abra o projeto e escolha Settings do painel de navegação na esquerda do console. Na página de detalhes do projeto, escolha Excluir projeto.

4. Na Página de confirmação de exclusão, insira excluir. Mantenha a opção Excluir recursos selecionada se quiser excluir recursos do projeto. Escolha Excluir.

A exclusão de um projeto pode demorar vários minutos. Depois de excluído, o projeto não aparece mais na lista de projetos no AWS CodeStar console.

 Important

Se seu projeto usa recursos externos AWS (por exemplo, um GitHub repositório ou problemas no Atlassian JIRA), esses recursos não serão excluídos, mesmo que você marque a caixa de seleção.

Seu projeto não pode ser excluído se alguma política AWS CodeStar gerenciada tiver sido anexada manualmente a funções que não sejam usuários do IAM. Se tiver anexado as políticas gerenciadas do projeto a uma função do usuário federado, você deverá desanexar a política para excluir o projeto. Para obter mais informações, consulte [???](#).

## Excluir um projeto em AWS CodeStar (AWS CLI)

Você pode usar o AWS CLI para excluir um projeto.

Para excluir um projeto no AWS CodeStar

1. Em um terminal (Linux, macOS, ou Unix) ou prompt de comando (Windows), execute o comando `delete-project` incluindo o nome do projeto. Por exemplo, para excluir um projeto com o ID `my-2nd-project`:

```
aws codestar delete-project --id my-2nd-project
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "projectArn": "arn:aws:codestar:us-east-2:111111111111:project/my-2nd-project"
}
```

Os projetos não são excluídos imediatamente.

2. Execute o comando `describe-project`, incluindo o nome do projeto. Por exemplo, para verificar o status de um projeto com o ID `my-2nd-project`:

```
aws codestar describe-project --id my-2nd-project
```

se o projeto ainda não foi excluído, esse comando retorna uma saída semelhante à seguinte:

```
{
  "name": "my project",
  "id": "my-2nd-project",
  "arn": "arn:aws:codestar:us-west-2:123456789012:project/my-2nd-project",
  "description": "My second CodeStar project.",
  "createdTimeStamp": 1572547510.128,
  "status": {
    "state": "CreateComplete"
  }
}
```

Se o projeto for excluído, esse comando retornará uma saída semelhante à seguinte:

```
An error occurred (ProjectNotFoundException) when calling the DescribeProject
operation: The project ID was not found: my-2nd-project. Make sure that the
project ID is correct and then try again.
```

3. Execute o comando `list-projects` e verifique se o projeto excluído não aparece mais na lista de projetos associados à sua conta da AWS .

```
aws codestar list-projects
```

## Trabalhando com AWS CodeStar equipes

Depois de criar um projeto de desenvolvimento, conceda acesso a outros usuários, para que vocês possam trabalhar em conjunto. Em AWS CodeStar, cada projeto tem uma equipe de projeto. Um usuário pode pertencer a vários AWS CodeStar projetos e ter AWS CodeStar funções diferentes (e, portanto, permissões diferentes) em cada um. No AWS CodeStar console, os usuários veem todos os projetos associados à sua AWS conta, mas só podem ver e trabalhar nos projetos nos quais são membros da equipe.

Os membros da equipe podem escolher um nome amigável para si mesmos. Eles também podem adicionar um endereço de e-mail para que outros membros da equipe possam entrar em contato com eles. Os membros da equipe que não são proprietários não podem alterar sua função do AWS CodeStar para o projeto.

Cada projeto AWS CodeStar tem três funções:

### Funções e permissões em um AWS CodeStar projeto

Função	Visualizar o painel e o status do projeto	Add/Remove/AccessRecursos do projeto	Adicionar /remover membros da equipe	Excluir projeto
Proprietário	x	x	x	x
Colaborador	x	x		
Visualizador	x			

- **Proprietário:** pode adicionar e remover outros membros da equipe, contribuir com código para um repositório do projeto se o código estiver armazenado CodeCommit, conceder ou negar a outros membros da equipe acesso remoto a qualquer EC2 instância da Amazon executando Linux associada ao projeto, configurar o painel do projeto e excluir o projeto.
- **Colaborador:** pode adicionar e remover recursos do painel, como um quadro do JIRA, contribuir com código para o repositório do projeto, se o código estiver armazenado CodeCommit, e interagir totalmente com o painel. Não pode adicionar ou remover membros da equipe, conceder ou negar acesso remoto a recursos ou excluir o projeto. Essa é a função que você deve escolher para a maioria dos membros da equipe.

- Visualizador: pode visualizar o painel do projeto, o código em que está armazenado e CodeCommit, nos blocos do painel, o estado do projeto e seus recursos.

#### Important

Se seu projeto usa recursos externos AWS (por exemplo, um GitHub repositório ou problemas no Atlassian JIRA), o acesso a esses recursos é controlado pelo provedor de recursos, não. AWS CodeStar Para obter mais informações, consulte a documentação do provedor do recurso.

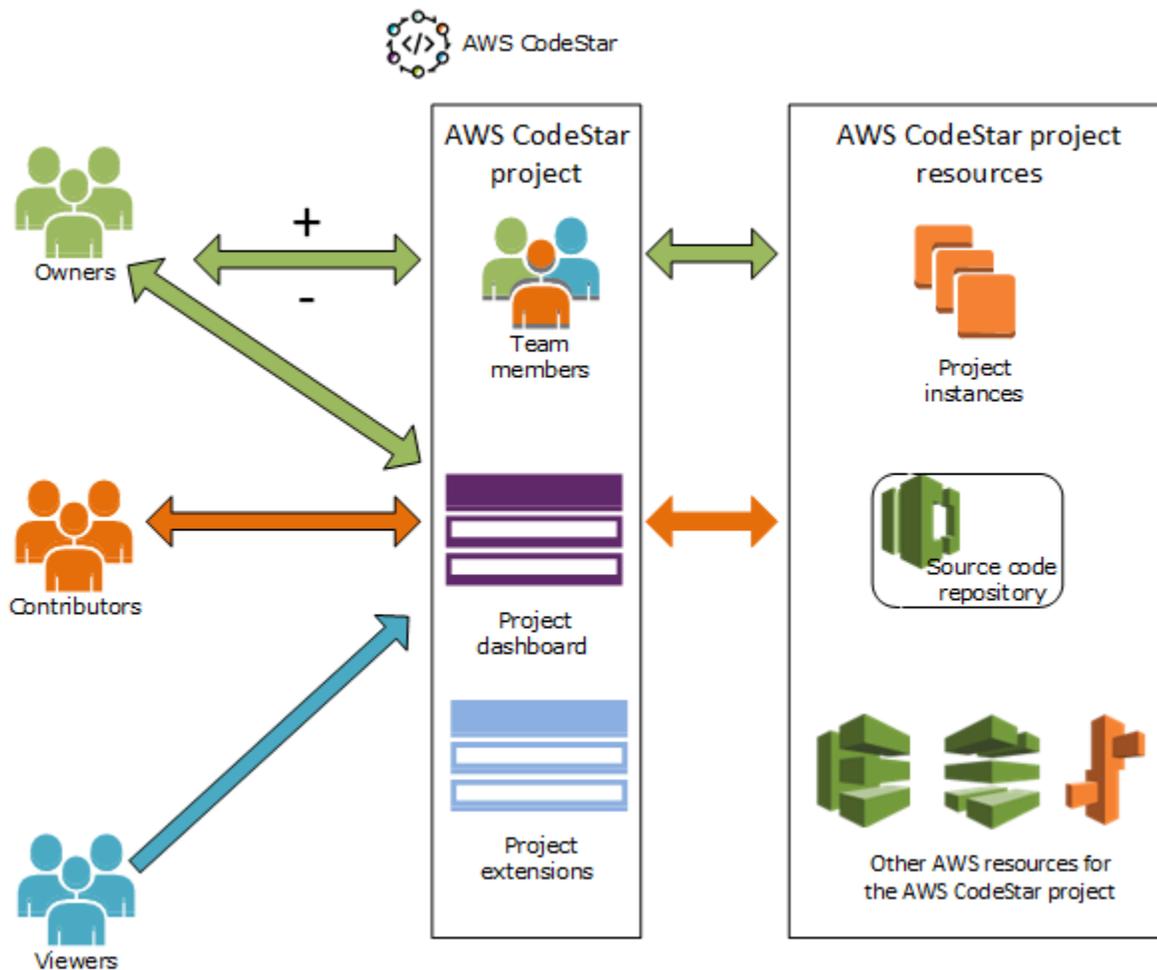
Qualquer pessoa que tenha acesso a um AWS CodeStar projeto pode usar o AWS CodeStar console para acessar recursos externos AWS , mas relacionados ao projeto.

AWS CodeStar não permite automaticamente que os membros da equipe do projeto participem de nenhum ambiente de AWS Cloud9 desenvolvimento relacionado a um projeto.

Para permitir que um membro da equipe participe de um ambiente compartilhado, consulte [Compartilhe um AWS Cloud9 ambiente com um membro da equipe do projeto](#).

Uma política do IAM é associada a cada função do projeto. Esta política é personalizada para o projeto para refletir seus recursos. Para obter mais informações sobre essas políticas, consulte [Exemplos de políticas CodeStar baseadas em identidade da AWS](#).

O diagrama a seguir mostra a relação entre cada função e um projeto do AWS CodeStar .



## Tópicos

- [Adicionar membros da equipe a um AWS CodeStar projeto](#)
- [Gerenciar permissões para membros AWS CodeStar da equipe](#)
- [Remover membros da equipe de um AWS CodeStar projeto](#)

## Adicionar membros da equipe a um AWS CodeStar projeto

Se você tem o papel de proprietário em um AWS CodeStar projeto ou tem a `AWSCodeStarFullAccess` política aplicada ao seu usuário do IAM, você pode adicionar outros usuários do IAM à equipe do projeto. Esse é um processo simples que aplica uma AWS CodeStar função (proprietário, colaborador ou visualizador) ao usuário. Essas funções são por projeto e personalizadas. Por exemplo, um membro da equipe colaborador no projeto A pode ter permissões para recursos diferentes das de um membro da equipe colaborador no projeto B. Um membro da

equipe pode ter apenas uma função em um projeto. Depois que você adiciona um membro da equipe, ele ou ela pode interagir imediatamente com o projeto no nível definido pela função.

Os benefícios das AWS CodeStar funções e da participação na equipe incluem:

- Você não precisa configurar manualmente as permissões no IAM para os membros da equipe.
- Pode alterar facilmente o nível de acesso de um membro da equipe a um projeto.
- Os usuários podem acessar projetos no AWS CodeStar console somente se forem membros da equipe.
- O acesso do usuário a um projeto é definido pela função.

Para obter mais informações sobre equipes e AWS CodeStar funções, consulte [Trabalhando com AWS CodeStar equipes](#) [Trabalhando com seu perfil de AWS CodeStar usuário](#) e.

Para adicionar um membro da equipe a um projeto, você deve ter o papel de AWS CodeStar proprietário do projeto ou da `AWSCodeStarFullAccess` política.

#### Important

Adicionar um membro da equipe não afeta o acesso desse membro a recursos externos AWS (por exemplo, um GitHub repositório ou problemas no Atlassian JIRA). Essas permissões de acesso são controladas pelo provedor de recursos, não AWS CodeStar. Para obter mais informações, consulte a documentação do provedor do recurso.

Qualquer pessoa que tenha acesso a um AWS CodeStar projeto pode usar o AWS CodeStar console para acessar recursos que estão fora desse projeto AWS, mas relacionados a ele.

Adicionar um membro da equipe a um projeto não permite automaticamente que esse membro participe de nenhum ambiente de AWS Cloud9 desenvolvimento relacionado ao projeto. Para permitir que um membro da equipe participe de um ambiente compartilhado, consulte [Compartilhe um AWS Cloud9 ambiente com um membro da equipe do projeto](#).

A concessão de acesso de usuário federado a um projeto envolve anexar manualmente a política de proprietário, colaborador ou visualizador do AWS CodeStar gerenciada à função assumida pelo usuário federado. Para obter mais informações, consulte [Acesso de usuário federado a AWS CodeStar](#).

## Tópicos

- [Adicionar um membro da equipe \(Console\)](#)

- [Adicionar e exibir membros da equipe \(AWS CLI\)](#)

## Adicionar um membro da equipe (Console)

Você pode usar o AWS CodeStar console para adicionar um membro da equipe ao seu projeto. Se um usuário do IAM já existir para a pessoa que você deseja adicionar, você poderá adicionar o usuário do IAM. Do contrário, você poderá criar um usuário do IAM para essa pessoa ao adicioná-los ao projeto.

Para adicionar um membro da equipe a um AWS CodeStar projeto (console)

1. Abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.
2. Selecione Projetos no painel de navegação e escolha seu projeto.
3. No painel lateral de navegação do projeto, escolha Equipe.
4. Na página Membros da equipe, escolha Adicionar membro da equipe.
5. Em Escolher usuário, proceda de uma das seguintes maneiras:
  - Se um usuário do IAM já existir para a pessoa que você deseja adicionar, escolha o usuário IAM na lista.

### Note

Os usuários que já foram adicionados a outro AWS CodeStar projeto aparecem na lista de AWS CodeStar usuários existentes.

Em Papel do projeto, escolha o AWS CodeStar papel (Proprietário, Colaborador ou Visualizador) para esse usuário. Essa é uma função em nível de projeto do AWS CodeStar que só pode ser alterada por um proprietário do projeto. Quando aplicado a um usuário do IAM, o papel fornece todas as permissões necessárias para acessar os recursos AWS CodeStar do projeto. Ele aplica as políticas necessárias para criar e gerenciar credenciais do Git para código armazenado CodeCommit no IAM ou fazer o upload das chaves EC2 SSH da Amazon para o usuário no IAM.

 Important

Você não pode fornecer ou alterar as informações de nome de exibição ou de e-mail de um usuário do IAM, a menos que você tenha feito login no console com esse usuário. Para obter mais informações, consulte [Gerencie as informações de exibição AWS CodeStar do seu perfil de usuário](#).

Escolha Adicionar membro da equipe.

- Se um usuário do IAM não existir para a pessoa que você deseja adicionar ao projeto, escolha Criar novo usuário do IAM;. Você será redirecionado para o console do IAM, onde poderá criar um novo usuário do IAM. Consulte [Como criar usuários do IAM](#) no guia do usuário do IAM para obter mais informações. Depois de criar seu usuário do IAM, retorne ao AWS CodeStar console, atualize a lista de usuários e escolha o usuário do IAM que você criou na lista suspensa. Insira o nome de AWS CodeStar exibição, o endereço de e-mail e a função do projeto que você deseja aplicar a esse novo usuário e escolha Adicionar membro da equipe.

 Note

Para facilitar o gerenciamento, pelo menos um usuário deve receber a função Proprietário do projeto.

6. Envie ao novo membro da equipe as seguintes informações:

- Informações de conexão para seu AWS CodeStar projeto.
- Se o código-fonte estiver armazenado em CodeCommit, [instruções para configurar o acesso com credenciais do Git](#) ao CodeCommit repositório a partir de seus computadores locais.
- Informações sobre como o usuário pode gerenciar seu nome de exibição, endereço de e-mail e chave EC2 SSH pública da Amazon, conforme descrito em [Trabalhando com seu perfil de AWS CodeStar usuário](#).
- Uma senha avulsa e informações de conexão, se o usuário for novo no AWS e você tiver criado um usuário do IAM para essa pessoa. A senha expira na primeira vez em que o usuário entra. O usuário deve escolher uma nova senha.

## Adicionar e exibir membros da equipe (AWS CLI)

Você pode usar o AWS CLI para adicionar membros da equipe à sua equipe de projeto. Você também pode visualizar informações sobre todos os membros da equipe do projeto.

Para adicionar um membro da equipe

1. Abra uma janela de terminal ou de comando.
2. Execute o comando `associate-team-member` com os parâmetros `--project-id`, `-user-arn` e `--project-role`. Você também pode especificar se o usuário tem acesso remoto às instâncias do projeto incluindo os parâmetros `--remote-access-allowed` ou `--no-remote-access-allowed`. Por exemplo:

```
aws codestar associate-team-member --project-id my-first-projec --user-arn
arn:aws:iam:111111111111:user/Jane_Doe --project-role Contributor --remote-access-
allowed
```

Esse comando não retorna uma saída.

Para visualizar todos os membros da equipe (AWS CLI)

1. Abra uma janela de terminal ou de comando.
2. Execute o comando `list-team-members` com o parâmetro `--project-id`. Por exemplo:

```
aws codestar list-team-members --project-id my-first-projec
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "teamMembers":[
    {"projectRole":"Owner","remoteAccessAllowed":true,"userArn":"arn:aws:iam::111111111111:use
Mary_Major"},
    {"projectRole":"Contributor","remoteAccessAllowed":true,"userArn":"arn:aws:iam::1111111111
Jane_Doe"},
    {"projectRole":"Contributor","remoteAccessAllowed":true,"userArn":"arn:aws:iam::1111111111
John_Doe"},
```

```
{ "projectRole": "Viewer", "remoteAccessAllowed": false, "userArn": "arn:aws:iam::111111111111:u
John_Stiles" }
]
```

## Gerenciar permissões para membros AWS CodeStar da equipe

Você altera as permissões dos membros da equipe alterando suas AWS CodeStar funções. Cada membro da equipe pode ser atribuído a apenas uma função em um AWS CodeStar projeto, mas muitos usuários podem ser atribuídos à mesma função. Você pode usar o AWS CodeStar console ou AWS CLI gerenciar permissões.

### Important

Para alterar a função de um membro da equipe, você deve ter a função de AWS CodeStar proprietário desse projeto ou ter a `AWSCodeStarFullAccess` política aplicada.

Alterar as permissões de um membro da equipe não afeta o acesso desse membro a nenhum recurso externo AWS (por exemplo, um GitHub repositório ou problemas no Atlassian JIRA). Essas permissões de acesso são controladas pelo provedor de recursos, não pelo AWS CodeStar. Para obter mais informações, consulte a documentação do provedor do recurso.

Qualquer pessoa que tenha acesso a um AWS CodeStar projeto pode usar o AWS CodeStar console para acessar recursos que estão fora desse projeto AWS, mas relacionados a ele. Alterar a função de um membro da equipe em um projeto não permite ou impede automaticamente que esse membro participe de qualquer ambiente de AWS Cloud9 desenvolvimento do projeto. Para permitir ou evitar que um membro da equipe participe de um ambiente compartilhado, consulte [Compartilhe um AWS Cloud9 ambiente com um membro da equipe do projeto](#).

Você também pode conceder permissões para que os usuários acessem remotamente qualquer instância do Amazon EC2 Linux associada ao projeto. Depois de conceder essa permissão, o usuário deve carregar uma chave pública SSH associada ao seu perfil de AWS CodeStar usuário em todos os projetos da equipe. Para conectar-se com êxito às instâncias do Linux, o usuário deve ter o SSH configurado e a chave privada no computador local.

## Tópicos

- [Gerenciar permissões da equipe \(console\)](#)
- [Gerenciar permissões da equipe \(AWS CLI\)](#)

## Gerenciar permissões da equipe (console)

Você pode usar o AWS CodeStar console para gerenciar as funções dos membros da equipe. Você também pode gerenciar se os membros da equipe têm acesso remoto às EC2 instâncias da Amazon associadas ao seu projeto.

Para alterar a função de um membro da equipe

1. Abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.
2. Selecione Projetos no painel de navegação e escolha seu projeto.
3. No painel lateral de navegação do projeto, escolha Equipe.
4. Na página Membros da equipe, selecione o membro da equipe e escolha Editar.
5. No Papel do projeto, escolha o AWS CodeStar papel (proprietário, colaborador ou visualizador) que você deseja conceder a esse usuário.

Para obter mais informações sobre AWS CodeStar funções e suas permissões, consulte [Trabalhando com AWS CodeStar equipes](#).

Escolha Editar membro da equipe.

Para conceder a um membro da equipe permissões de acesso remoto às EC2 instâncias da Amazon

1. Abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.
2. Selecione Projetos no painel de navegação e escolha seu projeto.
3. No painel lateral de navegação do projeto, escolha Equipe.
4. Na página Membros da equipe, selecione o membro da equipe e escolha Editar.
5. Selecione Permitir acesso SSH a instâncias do projeto e, em seguida, escolha Editar membro da equipe.
6. (Opcional) Notifique os membros da equipe de que eles devem fazer o upload de uma chave pública SSH para seus AWS CodeStar usuários, caso ainda não tenham feito isso. Para obter

mais informações, consulte [Adicione uma chave pública ao seu perfil de AWS CodeStar usuário](#)

## Gerenciar permissões da equipe (AWS CLI)

Você pode usar o AWS CLI para gerenciar a função do projeto atribuída a um membro da equipe. Você pode usar os mesmos AWS CLI comandos para gerenciar se esse membro da equipe tem acesso remoto às EC2 instâncias da Amazon associadas ao seu projeto.

Para gerenciar as permissões para um membro da equipe

1. Abra uma janela de terminal ou de comando.
2. Execute o comando `update-team-member` com os parâmetros `--project-id`, `-user-arn` e `--project-role`. Você também pode especificar se o usuário tem acesso remoto às instâncias do projeto incluindo os parâmetros `--remote-access-allowed` ou `--no-remote-access-allowed`. Por exemplo, para atualizar a função do projeto de um usuário do IAM chamado `John_Doe` e alterar suas permissões de visualizador sem acesso remoto às instâncias do projeto da Amazon: EC2

```
aws codestar update-team-member --project-id my-first-projec --user-arn
arn:aws:iam:111111111111:user/John_Doe --project-role Viewer --no-remote-access-
allowed
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "projectRole": "Viewer",
  "remoteAccessAllowed": false,
  "userArn": "arn:aws:iam::111111111111:user/John_Doe"
}
```

## Remover membros da equipe de um AWS CodeStar projeto

Depois de remover um usuário de um AWS CodeStar projeto, o usuário ainda aparece no histórico de confirmações do repositório do projeto, mas não tem mais acesso ao CodeCommit repositório ou a nenhum outro recurso do projeto, como o pipeline do projeto. (A exceção a essa regra é um usuário do IAM que tem outras políticas aplicadas que concedem acesso a esses recursos.) O

usuário não pode acessar o painel do projeto, e o projeto não aparece mais na lista de projetos que o usuário vê no AWS CodeStar painel. Você pode usar o AWS CodeStar console ou AWS CLI para remover membros da equipe da sua equipe de projeto.

### Important

Embora a remoção de um membro da equipe de um projeto negue o acesso remoto às EC2 instâncias do projeto da Amazon, ela não fecha nenhuma das sessões SSH ativas do usuário.

A remoção de um integrante da equipe não afeta o acesso desse membro a nenhum recurso externo AWS (por exemplo, um GitHub repositório ou problemas no Atlassian JIRA). Essas permissões de acesso são controladas pelo provedor de recursos, não AWS CodeStar. Para obter mais informações, consulte a documentação do provedor do recurso.

A remoção de um membro da equipe de um projeto não exclui automaticamente os ambientes de AWS Cloud9 desenvolvimento relacionados desse membro da equipe nem impede que esse membro participe de qualquer ambiente de AWS Cloud9 desenvolvimento relacionado para o qual tenha sido convidado. Para excluir um ambiente de desenvolvimento, consulte [Excluir um AWS Cloud9 ambiente de um projeto](#). Para evitar que um membro da equipe participe de um ambiente compartilhado, consulte [Compartilhe um AWS Cloud9 ambiente com um membro da equipe do projeto](#).

Para remover um membro da equipe de um projeto, você deve ter o papel de AWS CodeStar proprietário desse projeto ou ter a `AWSCodeStarFullAccess` política aplicada à sua conta.

### Tópicos

- [Remover membros da equipe \(console\)](#)
- [Remover membros da equipe \(AWS CLI\)](#)

## Remover membros da equipe (console)

Você pode usar o AWS CodeStar console para remover membros da equipe do seu projeto.

Remover um membro da equipe de um projeto

1. Abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.
2. Selecione Projetos no painel de navegação e escolha seu projeto.

3. No painel lateral de navegação do projeto, escolha Equipe.
4. Na página Membros da equipe, selecione o membro da equipe e escolha Remover.

## Remover membros da equipe (AWS CLI)

Você pode usar o AWS CLI para remover membros da equipe do seu projeto.

Para remover um membro da equipe

1. Abra uma janela de terminal ou de comando.
2. Execute o comando `disassociate-team-member` com os parâmetros `--project-id` e `-user-arn`. Por exemplo:

```
aws codestar disassociate-team-member --project-id my-first-projec --user-arn
arn:aws:iam:111111111111:user/John_Doe
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "projectId": "my-first-projec",
  "userArn": "arn:aws:iam::111111111111:user/John_Doe"
}
```

# Trabalhando com seu perfil de AWS CodeStar usuário

Seu perfil de AWS CodeStar usuário está associado ao seu usuário do IAM. Esse perfil contém um nome de exibição e endereço de e-mail que são usados em todos os AWS CodeStar projetos aos quais você pertence. Você pode fazer upload de uma chave pública SSH a ser associada a seu perfil. Essa chave pública faz parte do par de chaves público-privadas SSH que você usa quando se conecta às EC2 instâncias da Amazon associadas aos AWS CodeStar projetos aos quais você pertence.

## Note

As informações nesses tópicos abrangem somente seu perfil de AWS CodeStar usuário. Se seu projeto usa recursos externos AWS (por exemplo, um GitHub repositório ou problemas no Atlassian JIRA), esses provedores de recursos podem usar seus próprios perfis de usuário, que podem ter configurações diferentes. Para obter mais informações, consulte a documentação do provedor do recurso.

## Tópicos

- [Gerencie as informações de exibição AWS CodeStar do seu perfil de usuário](#)
- [Adicione uma chave pública ao seu perfil de AWS CodeStar usuário](#)

## Gerencie as informações de exibição AWS CodeStar do seu perfil de usuário

Você pode usar o AWS CodeStar console ou AWS CLI alterar o nome de exibição e o endereço de e-mail em seu perfil de usuário. Um perfil de usuário não é específico do projeto. Ele está associado ao seu usuário do IAM e é aplicado aos AWS CodeStar projetos aos quais você pertence em uma AWS região. Se você pertence a projetos em mais de uma AWS região, você tem perfis de usuário separados.

Você só pode gerenciar seu próprio perfil de usuário no AWS CodeStar console. Se você tiver a `AWSCodeStarFullAccess` política, poderá usá-la AWS CLI para visualizar e gerenciar outros perfis.

 Note

As informações neste tópico abrangem somente seu perfil de AWS CodeStar usuário. Se seu projeto usa recursos externos AWS (por exemplo, um GitHub repositório ou problemas no Atlassian JIRA), esses provedores de recursos podem usar seus próprios perfis de usuário, que podem ter configurações diferentes. Para obter mais informações, consulte a documentação do provedor do recurso.

## Tópicos

- [Gerenciar seu perfil de usuário \(console\)](#)
- [Gerenciar perfis de usuário \(AWS CLI\)](#)

## Gerenciar seu perfil de usuário (console)

Você pode gerenciar seu perfil de usuário no AWS CodeStar console navegando até qualquer projeto em que seja membro da equipe e alterando as informações do seu perfil. Como os perfis de usuário são específicos do usuário, não do projeto, as alterações do seu perfil de usuário aparecem em todos os projetos em uma AWS região em que você é membro da equipe.

 Important

Para usar o console a fim de alterar as informações de exibição de um usuário, você deve fazer login como esse usuário do IAM. Nenhum outro usuário, mesmo aqueles com a função de AWS CodeStar proprietário de um projeto ou com a `AWSCodeStarFullAccess` política aplicada, pode alterar suas informações de exibição.

Para alterar suas informações de exibição em todos os projetos em uma AWS região

1. Abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.
2. Escolha Projetos no painel de navegação e escolha um projeto em que você seja membro da equipe.
3. No painel lateral de navegação do projeto, escolha Equipe.
4. Na página Membros da equipe, escolha o usuário do IAM e, em seguida, escolha Editar.

5. Edite o nome de exibição, o endereço de e-mail, ou os dois, e, em seguida, escolha Editar membro da equipe.

 Note

Um nome de exibição e um endereço de e-mail são necessários. Para obter mais informações, consulte [Limites em AWS CodeStar](#).

## Gerenciar perfis de usuário (AWS CLI)

Você pode usar o AWS CLI para criar e gerenciar seu perfil de usuário no AWS CodeStar. Você também pode usar o AWS CLI para visualizar as informações do seu perfil de usuário e ver todos os perfis de usuário configurados para sua AWS conta em uma AWS região.

Certifique-se de que seu AWS perfil esteja configurado para a região em que você deseja criar, gerenciar ou visualizar perfis de usuário.

Para criar um perfil de usuário

1. Abra uma janela de terminal ou de comando.
2. Execute o comando `create-user-profile` com os parâmetros `user-arn`, `display-name` e `email-address`. Por exemplo:

```
aws codestar create-user-profile --user-arn arn:aws:iam:111111111111:user/John_Stiles --display-name "John Stiles" --email-address "john_stiles@example.com"
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "createdTimestamp":1.491439687681E9,
  "displayName":"John Stiles",
  "emailAddress":"john.stiles@example.com",
  "lastModifiedTimestamp":1.491439687681E9,
  "userArn":"arn:aws:iam::111111111111:user/Jane_Doe"
}
```

## Para visualizar informações de exibição

1. Abra uma janela de terminal ou de comando.
2. Execute o comando `describe-user-profile` com o parâmetro `user-arn`. Por exemplo:

```
aws codestar describe-user-profile --user-arn arn:aws:iam:111111111111:user/
Mary_Major
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "createdTimestamp":1.490634364532E9,
  "displayName":"Mary Major",
  "emailAddress":"mary.major@example.com",
  "lastModifiedTimestamp":1.491001935261E9,
  "sshPublicKey":"EXAMPLE=",
  "userArn":"arn:aws:iam::111111111111:user/Mary_Major"
}
```

## Para alterar suas informações de exibição

1. Abra uma janela de terminal ou de comando.
2. Execute o comando `update-user-profile` com o parâmetro `user-arn` e os parâmetros de perfil que você deseja alterar, como os parâmetros `display-name` ou `email-address`. Por exemplo, se um usuário com o nome de exibição Jane Doe quiser alterar seu nome de exibição para Jane Mary Doe:

```
aws codestar update-user-profile --user-arn arn:aws:iam:111111111111:user/Jane_Doe
--display-name "Jane Mary Doe"
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "createdTimestamp":1.491439687681E9,
  "displayName":"Jane Mary Doe",
  "emailAddress":"jane.doe@example.com",
  "lastModifiedTimestamp":1.491442730598E9,
  "sshPublicKey":"EXAMPLE1",
  "userArn":"arn:aws:iam::111111111111:user/Jane_Doe"
```

```
}
```

Para listar todos os perfis de usuário em uma AWS região em sua AWS conta

1. Abra uma janela de terminal ou de comando.
2. Execute o comando `aws codestar list-user-profiles`. Por exemplo:

```
aws codestar list-user-profiles
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "userProfiles": [
    {
      "displayName": "Jane Doe",
      "emailAddress": "jane.doe@example.com",
      "sshPublicKey": "EXAMPLE1",
      "userArn": "arn:aws:iam::111111111111:user/Jane_Doe"
    },
    {
      "displayName": "John Doe",
      "emailAddress": "john.doe@example.com",
      "sshPublicKey": "EXAMPLE2",
      "userArn": "arn:aws:iam::111111111111:user/John_Doe"
    },
    {
      "displayName": "Mary Major",
      "emailAddress": "mary.major@example.com",
      "sshPublicKey": "EXAMPLE=",
      "userArn": "arn:aws:iam::111111111111:user/Mary_Major"
    },
    {
      "displayName": "John Stiles",
      "emailAddress": "john.stiles@example.com",
      "sshPublicKey": "",
      "userArn": "arn:aws:iam::111111111111:user/John_Stiles"
    }
  ]
}
```

# Adicione uma chave pública ao seu perfil de AWS CodeStar usuário

Você pode fazer upload de uma chave SSH pública como parte do par de chaves pública/privada que você cria e gerencia. Você usa esse par de chaves SSH público-privadas para acessar EC2 instâncias da Amazon executando Linux. Se um proprietário de projeto concedeu a você permissão de acesso remoto, você poderá acessar somente as instâncias associadas ao projeto. Você pode usar o AWS CodeStar console ou AWS CLI gerenciar sua chave pública.

## Important

O proprietário AWS CodeStar do projeto pode conceder aos proprietários, colaboradores e visualizadores do projeto acesso SSH às EC2 instâncias da Amazon para o projeto, mas somente o indivíduo (proprietário, colaborador ou visualizador) pode definir a chave SSH. Para isso, o usuário deve fazer login como o proprietário, colaborador ou o visualizador individual.

AWS CodeStar não gerencia chaves SSH para AWS Cloud9 ambientes.

## Tópicos

- [Gerenciar a chave pública \(console\)](#)
- [Gerenciar sua chave pública \(AWS CLI\)](#)
- [Conecte-se à Amazon EC2 Instance com sua chave privada](#)

## Gerenciar a chave pública (console)

Embora você não possa gerar um par de chaves pública-privada no console, você pode criar um localmente e depois adicioná-lo ou gerenciá-lo como parte do seu perfil de usuário por meio do AWS CodeStar console.

Para gerenciar a chave SSH pública

1. Em um terminal ou janela do emulador Bash, execute o comando `ssh-keygen` para gerar um par de chaves SSH pública/privada no computador local. Você pode gerar uma chave em qualquer formato permitido pela Amazon EC2. Para obter informações sobre formatos aceitáveis, consulte [Importando sua própria chave pública para a Amazon EC2](#). De forma ideal, gere uma chave que

seja SSH-2 RSA, no formato OpenSSH e contenha 2048 bits. A chave pública é armazenada em um arquivo com a extensão .pub.

2. Abra o AWS CodeStar console em <https://console.aws.amazon.com/codestar/>.

Escolha um projeto em que você é membro da equipe.

3. No painel de navegação, escolha Equipe.
4. Na página Membros da equipe, localize o nome de seu usuário do IAM e então escolha Editar.
5. Na página Editar membro da equipe, em Acesso remoto, habilite Permitir acesso SSH às instâncias do projeto.
6. Na caixa Chave pública SSH, cole a chave pública e escolha Editar membro da equipe.

#### Note

Você pode alterar sua chave pública excluindo a chave antiga nesse campo e colando a nova chave. Você pode excluir uma chave pública excluindo o conteúdo desse campo e escolhendo Editar membro da equipe.

Quando altera ou exclui uma chave pública você está alterando seu perfil de usuário. Essa não é uma alteração por projeto. Como sua chave está associada a seu perfil, ela é alterada (ou é excluída) em todos os projetos em que você tiver recebido acesso remoto.

Excluir sua chave pública remove seu acesso às EC2 instâncias da Amazon que executam Linux em todos os projetos nos quais você recebeu acesso remoto. No entanto, ela não fecha nenhuma sessão SSH aberta que usa essa chave. Feche todas as sessões abertas.

## Gerenciar sua chave pública (AWS CLI)

Você pode usar o AWS CLI para gerenciar sua chave pública SSH como parte do seu perfil de usuário.

Para gerenciar a chave pública

1. Em um terminal ou janela do emulador Bash, execute o comando `ssh-keygen` para gerar um par de chaves SSH pública/privada no computador local. Você pode gerar uma chave em qualquer formato permitido pela Amazon EC2. Para obter informações sobre formatos aceitáveis, consulte [Importando sua própria chave pública para a Amazon EC2](#). De forma ideal, gere uma chave que

seja SSH-2 RSA, no formato OpenSSH e contenha 2048 bits. A chave pública é armazenada em um arquivo com a extensão .pub.

2. Para adicionar ou alterar sua chave pública SSH em seu perfil de AWS CodeStar usuário, execute o `update-user-profile` comando com o `--ssh-public-key` parâmetro. Por exemplo:

```
aws codestar update-user-profile --user-arn arn:aws:iam:111111111111:user/Jane_Doe
--ssh-key-id EXAMPLE1
```

Esse comando retorna uma saída semelhante à seguinte:

```
{
  "createdTimestamp":1.491439687681E9,
  "displayName":"Jane Doe",
  "emailAddress":"jane.doe@example.com",
  "lastModifiedTimestamp":1.491442730598E9,
  "sshPublicKey":"EXAMPLE1",
  "userArn":"arn:aws:iam::111111111111:user/Jane_Doe"
}
```

## Conecte-se à Amazon EC2 Instance com sua chave privada

Certifique-se de ter criado um par de EC2 chaves da Amazon. Adicione sua chave pública ao seu perfil de usuário em AWS CodeStar. Para criar um par de chaves, consulte [Etapa 4: criar um par de EC2 chaves da Amazon para AWS CodeStar projetos](#) Para adicionar a chave pública a seu perfil de usuário, consulte as instruções anteriormente neste tópico.

Para se conectar a uma instância Amazon EC2 Linux usando sua chave privada

1. Com seu projeto aberto no AWS CodeStar console, no painel de navegação, escolha Projeto.
2. Em Recursos do projeto, escolha o link ARN na linha em que Type é Amazon EC2 e Name começa com instance.
3. No EC2 console da Amazon, escolha Connect.
4. Siga as instruções na caixa de diálogo Connect To Your Instance (Conectar à sua instância).

Para o nome do usuário, use `ec2-user`. Se usar o nome de usuário incorreto, você não poderá conectar-se à instância.

---

Para obter mais informações, consulte os seguintes recursos no Guia do EC2 usuário da Amazon.

- [Conexão à sua instância do Linux utilizando SSH](#)
- [Conexão com a instância do Linux no Windows utilizando PuTTY](#)
- [Conectando-se à sua instância Linux usando MindTerm](#)

# Segurança em AWS CodeStar

A segurança na nuvem AWS é a maior prioridade. Como AWS cliente, você se beneficia de uma arquitetura de data center e rede criada para atender aos requisitos das organizações mais sensíveis à segurança.

A segurança é uma responsabilidade compartilhada entre você AWS e você. O [modelo de responsabilidade compartilhada](#) descreve isso como segurança da nuvem e segurança na nuvem:

- **Segurança da nuvem** — AWS é responsável por proteger a infraestrutura que executa AWS os serviços na AWS nuvem. AWS também fornece serviços que você pode usar com segurança. Auditores terceirizados testam e verificam regularmente a eficácia de nossa segurança como parte dos Programas de Conformidade Programas de [AWS](#) de . Para saber mais sobre os programas de conformidade aplicáveis AWS CodeStar, consulte [AWS Services in Scope by Compliance Program AWS](#) .
- **Segurança na nuvem** — Sua responsabilidade é determinada pelo AWS serviço que você usa. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da empresa e as leis e regulamentos aplicáveis.

Esta documentação ajuda você a entender como aplicar o modelo de responsabilidade compartilhada ao usar AWS CodeStar. Os tópicos a seguir mostram como configurar para atender AWS CodeStar aos seus objetivos de segurança e conformidade. Você também aprenderá a usar outros AWS serviços que ajudam a monitorar e proteger seus AWS CodeStar recursos.

Ao criar políticas personalizadas e usar limites de permissão em AWS CodeStar, garanta o acesso com privilégios mínimos concedendo somente as permissões necessárias para realizar uma tarefa e definindo o escopo das permissões para os recursos específicos. Para impedir que membros de outros projetos acessem recursos em seu projeto, conceda aos membros da organização permissões separadas para cada AWS CodeStar projeto. Como prática recomendada, crie uma conta de projeto para cada membro e, em seguida, atribua acesso baseado em funções a essa conta.

Por exemplo, você pode usar um serviço como o AWS Control Tower with AWS Organizations para provisionar contas para cada função de desenvolvedor em um DevOps grupo. Em seguida, você pode atribuir permissões a essas contas. As permissões gerais se aplicam à conta, mas o usuário tem acesso limitado aos recursos fora do projeto.

Para obter mais informações sobre como gerenciar o acesso com privilégios mínimos aos AWS recursos usando uma estratégia de várias contas, consulte a estratégia de [várias contas da AWS para sua landing zone no Guia do usuário do Control Tower](#).AWS

## Tópicos

- [Proteção de dados em AWS CodeStar](#)
- [Identity and Access Management para AWS CodeStar](#)
- [Registrando chamadas de AWS CodeStar API com AWS CloudTrail](#)
- [Validação de conformidade para AWS CodeStar](#)
- [Resiliência em AWS CodeStar](#)
- [Segurança de infraestrutura em AWS CodeStar](#)

## Proteção de dados em AWS CodeStar

O [modelo de responsabilidade AWS compartilhada](#) de se aplica à proteção de dados na AWS CodeStar. Conforme descrito neste modelo, AWS é responsável por proteger a infraestrutura global que executa todos os Nuvem AWS. Você é responsável por manter o controle sobre o conteúdo hospedado nessa infraestrutura. Você também é responsável pelas tarefas de configuração e gerenciamento de segurança dos Serviços da AWS que usa. Para obter mais informações sobre a privacidade de dados, consulte as [Data Privacy FAQ](#). Para obter mais informações sobre a proteção de dados na Europa, consulte a postagem do blog [AWS Shared Responsibility Model and RGPD](#) no Blog de segurança da AWS .

Para fins de proteção de dados, recomendamos que você proteja Conta da AWS as credenciais e configure usuários individuais com AWS IAM Identity Center ou AWS Identity and Access Management (IAM). Dessa maneira, cada usuário receberá apenas as permissões necessárias para cumprir suas obrigações de trabalho. Recomendamos também que você proteja seus dados das seguintes formas:

- Use uma autenticação multifator (MFA) com cada conta.
- Use SSL/TLS para se comunicar com os recursos. AWS Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Configure a API e o registro de atividades do usuário com AWS CloudTrail.
- Use soluções de AWS criptografia, juntamente com todos os controles de segurança padrão Serviços da AWS.

- Use serviços gerenciados de segurança avançada, como o Amazon Macie, que ajuda a localizar e proteger dados sigilosos armazenados no Amazon S3.
- Se você precisar de módulos criptográficos validados pelo FIPS 140-3 ao acessar AWS por meio de uma interface de linha de comando ou de uma API, use um endpoint FIPS. Para obter mais informações sobre os endpoints FIPS disponíveis, consulte [Federal Information Processing Standard \(FIPS\) 140-3](#).

É altamente recomendável que nunca sejam colocadas informações confidenciais ou sigilosas, como endereços de e-mail de clientes, em tags ou campos de formato livre, como um campo Nome. Isso inclui quando você trabalha com CodeStar ou Serviços da AWS usa o console, a API ou AWS SDKs. AWS CLI Quaisquer dados inseridos em tags ou em campos de texto de formato livre usados para nomes podem ser usados para logs de faturamento ou de diagnóstico. Se você fornecer um URL para um servidor externo, é fortemente recomendável que não sejam incluídas informações de credenciais no URL para validar a solicitação nesse servidor.

## Criptografia de dados em AWS CodeStar

Por padrão, AWS CodeStar criptografa as informações que armazena sobre seu projeto. Tudo além do ID do projeto é criptografado em repouso, como o nome do projeto, a descrição e os e-mails do usuário. Evite colocar informações pessoais em seu projeto IDs. AWS CodeStar também criptografa as informações em trânsito por padrão. Nenhuma ação por parte do cliente é necessária para a criptografia em repouso ou a criptografia em trânsito.

## Identity and Access Management para AWS CodeStar

AWS Identity and Access Management (IAM) é uma ferramenta AWS service (Serviço da AWS) que ajuda o administrador a controlar com segurança o acesso aos AWS recursos. Os administradores do IAM controlam quem pode ser autenticado (conectado) e autorizado (tem permissões) para usar os recursos da AWS CodeStar . O IAM é um AWS service (Serviço da AWS) que você pode usar sem custo adicional.

### Tópicos

- [Público](#)
- [Autenticação com identidades](#)
- [Gerenciamento do acesso usando políticas](#)
- [Como a AWS CodeStar trabalha com o IAM](#)

- [AWS CodeStar Políticas e permissões em nível de projeto](#)
- [Exemplos de políticas CodeStar baseadas em identidade da AWS](#)
- [Solução de problemas de CodeStar identidade e acesso da AWS](#)

## Público

A forma como você usa AWS Identity and Access Management (IAM) é diferente, dependendo do trabalho que você faz na AWS CodeStar.

**Usuário do serviço** — Se você usa o CodeStar serviço da AWS para fazer seu trabalho, seu administrador fornecerá as credenciais e as permissões de que você precisa. À medida que você usa mais CodeStar recursos da AWS para fazer seu trabalho, você pode precisar de permissões adicionais. Compreenda como o acesso é gerenciado pode ajudar a solicitar as permissões corretas ao administrador. Se você não conseguir acessar um recurso na AWS CodeStar, consulte [Solução de problemas de CodeStar identidade e acesso da AWS](#).

**Administrador de serviços** — Se você é responsável pelos CodeStar recursos da AWS em sua empresa, provavelmente tem acesso total à AWS CodeStar. É seu trabalho determinar quais CodeStar recursos e recursos da AWS seus usuários do serviço devem acessar. Envie as solicitações ao administrador do IAM para alterar as permissões dos usuários de serviço. Revise as informações nesta página para compreender os conceitos básicos do IAM. Para saber mais sobre como sua empresa pode usar o IAM com a AWS CodeStar, consulte [Como a AWS CodeStar trabalha com o IAM](#).

**Administrador do IAM** — Se você for administrador do IAM, talvez queira saber detalhes sobre como criar políticas para gerenciar o acesso à AWS CodeStar. Para ver exemplos de políticas CodeStar baseadas em identidade da AWS que você pode usar no IAM, consulte [Exemplos de políticas CodeStar baseadas em identidade da AWS](#)

## Autenticação com identidades

A autenticação é como você faz login AWS usando suas credenciais de identidade. Você deve estar autenticado (conectado AWS) como o Usuário raiz da conta da AWS, como usuário do IAM ou assumindo uma função do IAM.

Você pode entrar AWS como uma identidade federada usando credenciais fornecidas por meio de uma fonte de identidade. AWS IAM Identity Center Usuários (IAM Identity Center), a autenticação

de login único da sua empresa e suas credenciais do Google ou do Facebook são exemplos de identidades federadas. Quando você faz login como identidade federada, o administrador já configurou anteriormente a federação de identidades usando perfis do IAM. Ao acessar AWS usando a federação, você está assumindo indiretamente uma função.

Dependendo do tipo de usuário que você é, você pode entrar no AWS Management Console ou no portal de AWS acesso. Para obter mais informações sobre como fazer login em AWS, consulte [Como fazer login Conta da AWS](#) no Guia do Início de Sessão da AWS usuário.

Se você acessar AWS programaticamente, AWS fornece um kit de desenvolvimento de software (SDK) e uma interface de linha de comando (CLI) para assinar criptograficamente suas solicitações usando suas credenciais. Se você não usa AWS ferramentas, você mesmo deve assinar as solicitações. Para obter mais informações sobre como usar o método recomendado para assinar solicitações por conta própria, consulte [Assinatura de solicitações de AWS API](#) no Guia do usuário do IAM.

Independente do método de autenticação usado, também pode ser necessário fornecer informações adicionais de segurança. Por exemplo, AWS recomenda que você use a autenticação multifator (MFA) para aumentar a segurança da sua conta. Para saber mais, consulte [Autenticação Multifator](#) no Guia do Usuário do AWS IAM Identity Center . [Usar a autenticação multifator \(MFA\) na AWS](#) no Guia do Usuário do IAM.

## Conta da AWS usuário root

Ao criar uma Conta da AWS, você começa com uma identidade de login que tem acesso completo a todos Serviços da AWS os recursos da conta. Essa identidade é chamada de usuário Conta da AWS raiz e é acessada fazendo login com o endereço de e-mail e a senha que você usou para criar a conta. É altamente recomendável não usar o usuário-raiz para tarefas diárias. Proteja as credenciais do usuário-raiz e use-as para executar as tarefas que somente ele puder executar. Para obter a lista completa das tarefas que exigem login como usuário-raiz, consulte [Tarefas que exigem credenciais de usuário-raiz](#) no Guia do Usuário do IAM.

## Grupos e usuários do IAM

Um [usuário do IAM](#) é uma identidade dentro da sua Conta da AWS que tem permissões específicas para uma única pessoa ou aplicativo. Sempre que possível, é recomendável contar com credenciais temporárias em vez de criar usuários do IAM com credenciais de longo prazo, como senhas e chaves de acesso. No entanto, se você tiver casos de uso específicos que exijam credenciais de longo prazo com usuários do IAM, é recomendável alternar as chaves de acesso. Para obter mais

informações, consulte [Alternar as chaves de acesso regularmente para casos de uso que exijam credenciais de longo prazo](#) no Guia do Usuário do IAM.

Um [grupo do IAM](#) é uma identidade que especifica uma coleção de usuários do IAM. Não é possível fazer login como um grupo. É possível usar grupos para especificar permissões para vários usuários de uma vez. Os grupos facilitam o gerenciamento de permissões para grandes conjuntos de usuários. Por exemplo, você pode ter um grupo chamado IAMAdminse conceder a esse grupo permissões para administrar recursos do IAM.

Usuários são diferentes de perfis. Um usuário é exclusivamente associado a uma pessoa ou a uma aplicação, mas um perfil pode ser assumido por qualquer pessoa que precisar dele. Os usuários têm credenciais permanentes de longo prazo, mas os perfis fornecem credenciais temporárias. Para saber mais, consulte [Quando criar um usuário do IAM \(em vez de um perfil\)](#) no Guia do usuário do IAM.

## Perfis do IAM

Uma [função do IAM](#) é uma identidade dentro da sua Conta da AWS que tem permissões específicas. Ele é semelhante a um usuário do IAM, mas não está associado a uma pessoa específica. Você pode assumir temporariamente uma função do IAM no AWS Management Console [trocando de funções](#). Você pode assumir uma função chamando uma operação de AWS API AWS CLI ou usando uma URL personalizada. Para obter mais informações sobre métodos para o uso de perfis, consulte [Utilizar perfis do IAM](#) no Guia do usuário do IAM.

Funções do IAM com credenciais temporárias são úteis nas seguintes situações:

- **Acesso de usuário federado:** para atribuir permissões a identidades federadas, é possível criar um perfil e definir permissões para ele. Quando uma identidade federada é autenticada, essa identidade é associada ao perfil e recebe as permissões definidas por ele. Para obter mais informações sobre perfis para federação, consulte [Criar um perfil para um provedor de identidades de terceiros](#) no Guia do usuário do IAM. Se usar o Centro de Identidade do IAM, configure um conjunto de permissões. Para controlar o que suas identidades podem acessar após a autenticação, o Centro de Identidade do IAM correlaciona o conjunto de permissões a um perfil no IAM. Para obter informações sobre conjuntos de permissões, consulte [Conjuntos de Permissões](#) no Guia do Usuário do AWS IAM Identity Center .
- **Permissões temporárias para usuários do IAM:** um usuário ou um perfil do IAM pode presumir um perfil do IAM para obter temporariamente permissões diferentes para uma tarefa específica.
- **Acesso entre contas:** é possível usar um perfil do IAM para permitir que alguém (uma entidade principal confiável) em outra conta acesse recursos em sua conta. Os perfis são a principal forma

de conceder acesso entre contas. No entanto, com alguns Serviços da AWS, você pode anexar uma política diretamente a um recurso (em vez de usar uma função como proxy). Para conhecer a diferença entre perfis e políticas baseadas em recurso para acesso entre contas, consulte [Acesso a recursos entre contas no IAM](#) no Guia do usuário do IAM.

- Acesso entre serviços — Alguns Serviços da AWS usam recursos em outros Serviços da AWS. Por exemplo, quando você faz uma chamada em um serviço, é comum que esse serviço execute aplicativos na Amazon EC2 ou armazene objetos no Amazon S3. Um serviço pode fazer isso usando as permissões da entidade principal da chamada, usando um perfil de serviço ou um perfil vinculado ao serviço.
- Sessões de acesso direto (FAS) — Quando você usa um usuário ou uma função do IAM para realizar ações em AWS, você é considerado principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. O FAS usa as permissões do diretor chamando um AWS service (Serviço da AWS), combinadas com a solicitação AWS service (Serviço da AWS) para fazer solicitações aos serviços posteriores. As solicitações do FAS são feitas somente quando um serviço recebe uma solicitação que requer interações com outros Serviços da AWS ou com recursos para ser concluída. Nesse caso, você precisa ter permissões para executar ambas as ações. Para obter detalhes da política ao fazer solicitações de FAS, consulte [Sessões de acesso direto](#).
- Perfil de serviço: um perfil de serviço é um [perfil do IAM](#) que um serviço assume para executar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para obter mais informações, consulte [Criar um perfil para delegar permissões a um AWS service \(Serviço da AWS\)](#) no Guia do usuário do IAM.
- Função vinculada ao serviço — Uma função vinculada ao serviço é um tipo de função de serviço vinculada a um AWS service (Serviço da AWS). O serviço pode presumir o perfil de executar uma ação em seu nome. As funções vinculadas ao serviço aparecem em você Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não editar as permissões para perfis vinculados ao serviço.
- Aplicativos em execução na Amazon EC2 — Você pode usar uma função do IAM para gerenciar credenciais temporárias para aplicativos que estão sendo executados em uma EC2 instância e fazendo solicitações AWS CLI de AWS API. Isso é preferível a armazenar chaves de acesso na EC2 instância. Para atribuir uma AWS função a uma EC2 instância e disponibilizá-la para todos os aplicativos, você cria um perfil de instância anexado à instância. Um perfil de instância contém a função e permite que programas em execução na EC2 instância recebam credenciais temporárias. Para obter mais informações, consulte [Como usar uma função do IAM para conceder permissões a aplicativos executados em EC2 instâncias da Amazon](#) no Guia do usuário do IAM.

Para saber se deseja usar perfis do IAM, consulte [Quando criar um perfil do IAM \(em vez de um usuário\)](#) no Guia do usuário do IAM.

## Gerenciamento do acesso usando políticas

Você controla o acesso AWS criando políticas e anexando-as a AWS identidades ou recursos. Uma política é um objeto AWS que, quando associada a uma identidade ou recurso, define suas permissões. AWS avalia essas políticas quando um principal (usuário, usuário raiz ou sessão de função) faz uma solicitação. As permissões nas políticas determinam se a solicitação será permitida ou negada. A maioria das políticas é armazenada AWS como documentos JSON. Para obter mais informações sobre a estrutura e o conteúdo de documentos de políticas JSON, consulte [Visão geral das políticas JSON](#) no Guia do usuário do IAM.

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

Por padrão, usuários e perfis não têm permissões. Para conceder permissão aos usuários para executar ações nos recursos que eles precisam, um administrador do IAM pode criar políticas do IAM. O administrador pode então adicionar as políticas do IAM aos perfis e os usuários podem assumir os perfis.

As políticas do IAM definem permissões para uma ação independentemente do método usado para executar a operação. Por exemplo, suponha que você tenha uma política que permite a ação `iam:GetRole`. Um usuário com essa política pode obter informações de função da AWS Management Console AWS CLI, da ou da AWS API.

### Políticas baseadas em identidade

As políticas baseadas em identidade são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como usuário, grupo de usuários ou perfil do IAM. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte [Criando políticas do IAM](#) no Guia do Usuário do IAM.

As políticas baseadas em identidade podem ser categorizadas ainda adicionalmente como políticas em linha ou políticas gerenciadas. As políticas em linha são anexadas diretamente a um único usuário, grupo ou perfil. As políticas gerenciadas são políticas autônomas que você pode associar a vários usuários, grupos e funções em seu Conta da AWS. As políticas AWS gerenciadas incluem políticas gerenciadas e políticas gerenciadas pelo cliente. Para saber como escolher entre uma

política gerenciada ou uma política em linha, consulte [Escolher entre políticas gerenciadas e políticas em linha](#) no Guia do Usuário do IAM.

## Políticas baseadas em recurso

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e as políticas de bucket do Amazon S3. Em serviços compatíveis com políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o atributo ao qual a política está anexada, a política define quais ações uma entidade principal especificado pode executar nesse atributo e em que condições. Você deve [especificar uma entidade principal](#) em uma política baseada em recursos. Os diretores podem incluir contas, usuários, funções, usuários federados ou. Serviços da AWS

Políticas baseadas em recursos são políticas em linha localizadas nesse serviço. Você não pode usar políticas AWS gerenciadas do IAM em uma política baseada em recursos.

## Listas de controle de acesso (ACLs)

As listas de controle de acesso (ACLs) controlam quais diretores (membros da conta, usuários ou funções) têm permissões para acessar um recurso. ACLs são semelhantes às políticas baseadas em recursos, embora não usem o formato de documento de política JSON.

O Amazon S3 e o AWS WAF Amazon VPC são exemplos de serviços que oferecem suporte. ACLs Para saber mais ACLs, consulte a [visão geral da lista de controle de acesso \(ACL\)](#) no Guia do desenvolvedor do Amazon Simple Storage Service.

## Outros tipos de política

AWS oferece suporte a tipos de políticas adicionais menos comuns. Esses tipos de política podem definir o máximo de permissões concedidas a você pelos tipos de política mais comuns.

- **Limites de permissões:** um limite de permissões é um recurso avançado no qual você define o máximo de permissões que uma política baseada em identidade pode conceder a uma entidade do IAM (usuário ou perfil do IAM). É possível definir um limite de permissões para uma entidade. As permissões resultantes são a interseção das políticas baseadas em identidade de uma entidade com seus limites de permissões. As políticas baseadas em recurso que especificam o usuário ou o perfil no campo `Principal` não são limitadas pelo limite de permissões. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais

informações sobre limites de permissões, consulte [Limites de permissões para identidades do IAM](#) no Guia do usuário do IAM.

- Políticas de controle de serviço (SCPs) — SCPs são políticas JSON que especificam as permissões máximas para uma organização ou unidade organizacional (OU) em AWS Organizations. AWS Organizations é um serviço para agrupar e gerenciar centralmente várias Contas da AWS que sua empresa possui. Se você habilitar todos os recursos em uma organização, poderá aplicar políticas de controle de serviço (SCPs) a qualquer uma ou a todas as suas contas. O SCP limita as permissões para entidades nas contas dos membros, incluindo cada uma Usuário raiz da conta da AWS. Para obter mais informações sobre Organizations e SCPs, consulte [Políticas de controle de serviços](#) no Guia AWS Organizations do Usuário.
- Políticas de sessão: são políticas avançadas que você transmite como um parâmetro quando cria de forma programática uma sessão temporária para um perfil ou um usuário federado. As permissões da sessão resultante são a interseção das políticas baseadas em identidade do usuário ou do perfil e das políticas de sessão. As permissões também podem ser provenientes de uma política baseada em recursos. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações, consulte [Políticas de sessão](#) no Guia do usuário do IAM.

## Vários tipos de política

Quando vários tipos de política são aplicáveis a uma solicitação, é mais complicado compreender as permissões resultantes. Para saber como AWS determinar se uma solicitação deve ser permitida quando vários tipos de políticas estão envolvidos, consulte [Lógica de avaliação de políticas](#) no Guia do usuário do IAM.

## Como a AWS CodeStar trabalha com o IAM

Antes de usar o IAM para gerenciar o acesso à AWS CodeStar, você deve entender quais recursos do IAM estão disponíveis para uso com a AWS CodeStar. Para ter uma visão de alto nível de como a AWS CodeStar e outros AWS serviços funcionam com o IAM, consulte [AWS Serviços que funcionam com o IAM](#) no Guia do usuário do IAM.

### Tópicos

- [Políticas baseadas em CodeStar identidade da AWS](#)
- [Políticas baseadas em CodeStar recursos da AWS](#)
- [Autorização baseada em CodeStar tags da AWS](#)

- [Funções CodeStar do AWS IAM](#)
- [Acesso do usuário do IAM ao AWS CodeStar](#)
- [Acesso de usuário federado a AWS CodeStar](#)
- [Usando credenciais temporárias com a AWS CodeStar](#)
- [Funções vinculadas ao serviço](#)
- [Perfis de serviço](#)

## Políticas baseadas em CodeStar identidade da AWS

Com as políticas baseadas em identidade do IAM, você pode especificar ações e recursos permitidos ou negados e as condições sob as quais as ações são permitidas ou negadas. AWS CodeStar cria várias políticas baseadas em identidade em seu nome, que permitem AWS CodeStar criar e gerenciar recursos dentro do escopo de um AWS CodeStar projeto. A AWS CodeStar oferece suporte a ações, recursos e chaves de condição específicos. Para conhecer todos os elementos usados em uma política JSON, consulte [Referência de elementos de política JSON do IAM](#) no Guia do usuário do IAM.

### Ações

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento `Action` de uma política JSON descreve as ações que podem ser usadas para permitir ou negar acesso em uma política. As ações de política geralmente têm o mesmo nome da operação de AWS API associada. Existem algumas exceções, como ações somente de permissão, que não têm uma operação de API correspondente. Algumas operações também exigem várias ações em uma política. Essas ações adicionais são chamadas de ações dependentes.

Incluem ações em uma política para conceder permissões para executar a operação associada.

As ações políticas na AWS CodeStar usam o seguinte prefixo antes da ação: `codestar:`. Por exemplo, para permitir que um usuário específico do IAM edite os atributos de um AWS CodeStar projeto, como a descrição do projeto, você pode usar a seguinte declaração de política:

```
{
  "Version": "2012-10-17",
  "Statement" : [
    {
```

```
    "Effect" : "Allow",
    "Action" : [
      "codestar:UpdateProject"
    ],
    "Resource" : "arn:aws:codestar:us-east-2:project/my-first-projec"
  }
]
```

As instruções de política devem incluir um elemento `Action` ou `NotAction`. A AWS CodeStar define seu próprio conjunto de ações que descrevem tarefas que você pode realizar com esse serviço.

Para especificar várias ações em uma única instrução, separe-as com vírgulas, como segue:

```
"Action": [
  "codestar:action1",
  "codestar:action2"
```

Você também pode especificar várias ações usando caracteres curinga (\*). Por exemplo, para especificar todas as ações que começam com a palavra `List`, inclua a seguinte ação:

```
"Action": "codestar:List*"
```

Para ver uma lista de CodeStar ações da AWS, consulte [Ações definidas pela AWS CodeStar](#) no Guia do usuário do IAM.

## Recursos

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento de política JSON `Resource` especifica o objeto ou os objetos aos quais a ação se aplica. As instruções devem incluir um elemento `Resource` ou `NotResource`. Como prática recomendada, especifique um recurso usando seu [nome do recurso da Amazon \(ARN\)](#). Isso pode ser feito para ações que oferecem compatibilidade com um tipo de recurso específico, conhecido como permissões em nível de recurso.

Para ações que não oferecem compatibilidade com permissões em nível de recurso, como operações de listagem, use um curinga (\*) para indicar que a instrução se aplica a todos os recursos.

```
"Resource": "*"
```

O recurso AWS CodeStar do projeto tem o seguinte ARN:

```
arn:aws:codestar:region:account:project/resource-specifier
```

Para obter mais informações sobre o formato de ARNs, consulte [Amazon Resource Names \(ARNs\) e AWS Service Namespaces](#).

Por exemplo, o seguinte especifica o AWS CodeStar projeto chamado *my-first-projec* registrado na AWS conta 111111111111 na AWS região *us-east-2*:

```
arn:aws:codestar:us-east-2:111111111111:project/my-first-projec
```

O seguinte especifica qualquer AWS CodeStar projeto que comece com o nome *my-proj* registrado na AWS conta 111111111111 na AWS região *us-east-2*:

```
arn:aws:codestar:us-east-2:111111111111:project/my-proj*
```

Algumas CodeStar ações da AWS, como listar projetos, não podem ser executadas em um recurso. Nesses casos, você deve utilizar o caractere curinga (\*).

```
"ListProjects": "*"
```

Para ver uma lista dos tipos de CodeStar recursos da AWS e seus ARNs, consulte [Recursos definidos pela AWS CodeStar](#) no Guia do usuário do IAM. Para saber com quais ações você pode especificar o ARN de cada recurso, consulte [Ações definidas pela AWS](#). CodeStar

## Chaves de condição

CodeStar A AWS não fornece nenhuma chave de condição específica do serviço, mas oferece suporte ao uso de algumas chaves de condição globais. Para ver todas as chaves de condição AWS globais, consulte [Chaves de contexto de condição AWS global](#) no Guia do usuário do IAM.

## Exemplos

Para ver exemplos de políticas CodeStar baseadas em identidade da AWS, consulte [Exemplos de políticas CodeStar baseadas em identidade da AWS](#)

## Políticas baseadas em CodeStar recursos da AWS

AWS CodeStar não oferece suporte a políticas baseadas em recursos.

## Autorização baseada em CodeStar tags da AWS

Você pode anexar tags a CodeStar projetos da AWS ou passar tags em uma solicitação para a AWS CodeStar. Para controlar o acesso baseado em tags, forneça informações sobre as tags no [elemento de condição](#) de uma política usando as `codestar:ResourceTag/key-name`, `aws:RequestTag/key-name` ou chaves de condição `aws:TagKeys`. Para obter mais informações sobre a marcação de CodeStar recursos da AWS, consulte [the section called “Trabalhar com tags de projetos”](#).

Para ver um exemplo de política baseada em identidade para limitar o acesso a um AWS CodeStar projeto com base nas tags desse projeto, consulte [Visualização de CodeStar projetos da AWS com base em tags](#)

## Funções CodeStar do AWS IAM

Uma [função do IAM](#) é uma entidade na sua AWS conta que tem permissões específicas.

Você pode usar AWS CodeStar como [usuário do IAM](#), usuário federado, usuário raiz ou função assumida. Todos os tipos de usuários com as permissões apropriadas podem gerenciar as permissões do projeto para seus AWS recursos, mas AWS CodeStar gerenciam as permissões do projeto automaticamente para usuários do IAM. As políticas e perfis do IAM concedem permissões e acesso a esse usuário com base na função do projeto. Você pode usar o console do IAM para criar outras políticas que atribuam AWS CodeStar e outras permissões a um usuário do IAM.

Por exemplo, você pode permitir que um usuário visualize, mas não altere, um projeto do AWS CodeStar. Nesse caso, você adiciona o usuário do IAM a um AWS CodeStar projeto com o papel de visualizador. Cada AWS CodeStar projeto tem um conjunto de políticas que ajudam você a controlar o acesso ao projeto. Além disso, você pode controlar a quais usuários têm acesso AWS CodeStar.

AWS CodeStar o acesso é tratado de forma diferente para usuários do IAM e usuários federados. Somente os usuários do IAM podem ser adicionados às equipes. Para conceder aos usuários do IAM permissões para projetos, adicione o usuário à equipe do projeto e atribua uma função a ele.

Para conceder permissões aos usuários federados para projetos, você anexa manualmente a política gerenciada da função do AWS CodeStar projeto à função do usuário federado.

Esta tabela resume as ferramentas disponíveis para cada tipo de acesso.

Recurso de permissões	IAM user (Usuário do IAM)	Usuário federado	Usuário raiz
Gerenciamento de chaves SSH para acesso remoto para projetos Amazon EC2 e Elastic Beanstalk	✓		
AWS CodeCommit Acesso SSH	✓		
Permissões de usuário do IAM gerenciadas por AWS CodeStar	✓		
Permissões de projeto gerenciadas manualmente		✓	✓
Os usuários podem ser adicionados ao projeto como membros da equipe	✓		

## Acesso do usuário do IAM ao AWS CodeStar

Quando você adiciona um usuário do IAM a um projeto e escolhe uma função para o usuário, o AWS CodeStar aplica a política adequada automaticamente ao usuário do IAM. Para os usuários do IAM, você não precisa anexar ou gerenciar diretamente políticas ou permissões no IAM. Para obter informações sobre como adicionar um usuário do IAM a um AWS CodeStar projeto, consulte [Adicionar membros da equipe a um AWS CodeStar projeto](#). Para obter informações sobre como remover um usuário do IAM de um AWS CodeStar projeto, consulte [Remover membros da equipe de um AWS CodeStar projeto](#).

### Anexar uma política em linha a um usuário do IAM

Quando você adiciona um usuário a um projeto, anexa AWS CodeStar automaticamente a política gerenciada do projeto que corresponde à função do usuário. Você não deve anexar manualmente uma política AWS CodeStar gerenciada de um projeto a um usuário do IAM. Com exceção de `AWSCodeStarFullAccess`, não recomendamos que você anexe políticas que alterem as

permissões de um usuário do IAM em um AWS CodeStar projeto. Se você decidir criar e anexar suas próprias políticas, consulte [Adicionar e remover permissões de identidade do IAM](#) no Guia do usuário do IAM.

## Acesso de usuário federado a AWS CodeStar

Em vez de criar um usuário do IAM ou usar o usuário raiz, você pode usar identidades de usuário do diretório de usuários corporativo AWS Directory Service, de um provedor de identidade da web ou de usuários do IAM assumindo funções. Estes são conhecidos como usuários federados.

Conceda aos usuários federados acesso ao seu AWS CodeStar projeto anexando manualmente as políticas gerenciadas descritas em Políticas [e permissões em AWS CodeStar nível de projeto](#) à função do IAM do usuário. Você anexa a política de proprietário, colaborador ou visualizador depois de AWS CodeStar criar os recursos do projeto e as funções do IAM.

Pré-requisitos:

- Você deve ter criado um provedor de identidade. Por exemplo, você pode configurar um provedor de identidade SAML e configurar a AWS autenticação por meio do provedor. Para obter mais informações sobre como configurar um provedor de identidade, consulte [Criar provedores de identidade do IAM](#). Para obter mais informações sobre a federação SAML, consulte [Sobre a federação baseada em SAML 2.0](#).
- Você deve ter criado uma função para um usuário federado a ser assumida quando o acesso for solicitado por meio de um [provedor de identidade](#). Uma política de confiança STS deve ser anexada à função que permite que os usuários federados assumam a função. Para obter mais informações, consulte [Usuários federados e funções](#) no Guia do usuário do IAM.
- Você deve ter criado seu AWS CodeStar projeto e saber o ID do projeto.

Para obter mais informações sobre como criar uma função para provedores de identidade, consulte [Criar uma função para um provedor de identidade de terceiros \(federação\)](#).

Anexar a política `AWSCodeStarFullAccess` gerenciada à função do usuário federado

Conceda a um usuário federado permissões para criar um projeto anexando a política gerenciada `AWSCodeStarFullAccess`. Para realizar essas etapas, você deve entrar no console como um usuário raiz, um usuário do administrador da conta ou um usuário do IAM ou um usuário federado com a política gerenciada `AdministratorAccess` associada ou equivalente.

**Note**

Depois que você criar o projeto, as permissões de proprietário do projeto não serão aplicadas automaticamente. Usando uma função com permissões administrativas para a conta, anexe a política gerenciada do proprietário, conforme descrito em [Anexe a política AWS CodeStar Viewer/Contributor/Owner gerenciada do seu projeto à função do usuário federado](#).

1. Abra o console do IAM. No painel de navegação, selecione Políticas.
2. Digite `AWSCodeStarFullAccess` no campo de pesquisa. O nome da política é exibido, com um tipo de política de Gerenciado pelo AWS . Você pode expandir a política para ver as permissões na declaração de política.
3. Marque o círculo ao lado da política e, em Policy actions (Ações da política), escolha Attach (Anexar).
4. Na página Summary (Resumo), selecione a guia Attached entities (Entidades anexadas). Escolha Anexar.
5. Na página Attach Policy (Anexar política), filtre a função do usuário federado no campo de pesquisa. Selecione a caixa ao lado do nome da função e escolha Attach policy (Anexar política). A guia Attached entities (Entidades anexadas) mostrará o novo anexo.

## Anexe a política AWS CodeStar Viewer/Contributor/Owner gerenciada do seu projeto à função do usuário federado

Conceda aos usuários federados acesso ao projeto anexando a política gerenciada de proprietário, colaborador ou visualizador do apropriada à função do usuário. A política gerenciada oferece o nível apropriado de permissões. Diferentemente dos usuários do IAM, você precisa anexar manualmente e desanexar políticas gerenciadas para usuários federados. Isso equivale a atribuir permissões do projeto aos membros da equipe em AWS CodeStar. Para realizar essas etapas, você deve entrar no console como um usuário raiz, um usuário do administrador do na conta ou um usuário do IAM ou um usuário federado com a política gerenciada `AdministratorAccess` associada ou equivalente.

### Pré-requisitos:

- Você deve ter criado uma função ou ter uma função existente assumida pelo usuário federado.
- Você deve saber qual nível de permissões deseja conceder. As políticas gerenciadas anexadas às funções de proprietário, colaborador e visualizador fornecem permissões baseadas na função para o projeto.

- Seu AWS CodeStar projeto deve ter sido criado. A política gerenciada não está disponível no IAM até a criação do projeto.
1. Abra o console do IAM. No painel de navegação, selecione Políticas.
  2. Digite o ID do projeto no campo de pesquisa. O nome da política correspondente ao projeto é exibido, com um tipo de política de Customer managed (Gerenciado pelo cliente). Você pode expandir a política para ver as permissões na declaração de política.
  3. Escolha uma dessas políticas gerenciadas. Marque o círculo ao lado da política e, em Policy actions (Ações da política), escolha Attach (Anexar).
  4. Na página Summary (Resumo), selecione a guia Attached entities (Entidades anexadas). Escolha Anexar.
  5. Na página Attach Policy (Anexar política), filtre a função do usuário federado no campo de pesquisa. Selecione a caixa ao lado do nome da função e escolha Attach policy (Anexar política). A guia Attached entities (Entidades anexadas) mostrará o novo anexo.

### Separar uma política AWS CodeStar gerenciada da função do usuário federado

Antes de excluir seu AWS CodeStar projeto, você deve separar manualmente todas as políticas gerenciadas anexadas à função de um usuário federado. Para realizar essas etapas, você deve entrar no console como um usuário raiz, um usuário do administrador da conta ou um usuário do IAM ou um usuário federado com a política gerenciada `AdministratorAccess` associada ou equivalente.

1. Abra o console do IAM. No painel de navegação, selecione Políticas.
2. Digite o ID do projeto no campo de pesquisa.
3. Marque o círculo ao lado da política e, em Policy actions (Ações da política), escolha Attach (Anexar).
4. Na página Summary (Resumo), selecione a guia Attached entities (Entidades anexadas).
5. Filtre a função do usuário federado no campo de pesquisa. Escolha Desassociar.

### Anexar uma política AWS Cloud9 gerenciada à função do usuário federado

Se você estiver usando um ambiente de AWS Cloud9 desenvolvimento, conceda aos usuários federados acesso a ele anexando a política `AWSCloud9User` gerenciada à função do usuário. Diferentemente dos usuários do IAM, você precisa anexar manualmente e desanexar políticas

gerenciadas para usuários federados. Para realizar essas etapas, você deve entrar no console como um usuário raiz, um usuário do administrador da conta ou um usuário do IAM ou um usuário federado com a política gerenciada `AdministratorAccess` associada ou equivalente.

Pré-requisitos:

- Você deve ter criado uma função ou ter uma função existente assumida pelo usuário federado.
- Você deve saber qual nível de permissões deseja conceder:
  - A política gerenciada `AWSCloud9User` permite que o usuário faça o seguinte:
    - Crie seus próprios ambientes AWS Cloud9 de desenvolvimento.
    - Obtenha informações sobre os ambientes.
    - Altere as configurações para os ambientes.
  - A política gerenciada `AWSCloud9Administrator` permite que o usuário faça o seguinte para eles ou outros:
    - Crie ambientes.
    - Obtenha informações sobre ambientes.
    - Exclua ambientes.
    - Altere as configurações de ambientes.

1. Abra o console do IAM. No painel de navegação, selecione Políticas.
2. Digite o nome da política no campo de pesquisa. A política gerenciada é exibida, com um tipo de política de Gerenciado pelo AWS . Você pode expandir a política para ver as permissões na declaração de política.
3. Escolha uma dessas políticas gerenciadas. Marque o círculo ao lado da política e, em Policy actions (Ações da política), escolha Attach (Anexar).
4. Na página Summary (Resumo), selecione a guia Attached entities (Entidades anexadas). Escolha Anexar.
5. Na página Attach Policy (Anexar política), filtre a função do usuário federado no campo de pesquisa. Escolha a caixa ao lado do nome da função e Attach policy (Anexar política). A guia Attached entities (Entidades anexadas) mostrará o novo anexo.

## Separar uma política AWS Cloud9 gerenciada da função do usuário federado

Se você estiver usando um ambiente de AWS Cloud9 desenvolvimento, poderá remover o acesso de um usuário federado a ele desanexando a política que concede acesso. Para realizar essas etapas, você deve entrar no console como um usuário raiz, um usuário do administrador da conta ou um usuário do IAM ou um usuário federado com a política gerenciada `AdministratorAccess` associada ou equivalente.

1. Abra o console do IAM. No painel de navegação, selecione Políticas.
2. Digite o nome do projeto no campo de pesquisa.
3. Marque o círculo ao lado da política e, em Policy actions (Ações da política), escolha Attach (Anexar).
4. Na página Summary (Resumo), selecione a guia Attached entities (Entidades anexadas).
5. Filtre a função do usuário federado no campo de pesquisa. Escolha Desassociar.

## Usando credenciais temporárias com a AWS CodeStar

É possível usar credenciais temporárias para fazer login com federação, assumir um perfil do IAM ou assumir um perfil entre contas. Você obtém credenciais de segurança temporárias chamando operações de AWS STS API, como [AssumeRole](#) ou [GetFederationToken](#).

A AWS CodeStar suporta o uso de credenciais temporárias, mas a funcionalidade dos membros da AWS CodeStar equipe não funciona para acesso federado. A funcionalidade de membro da equipe só dá suporte à adição de um usuário do IAM como membro da equipe.

## Funções vinculadas ao serviço

[As funções vinculadas ao serviço](#) permitem que AWS os serviços acessem recursos em outros serviços para concluir uma ação em seu nome. Os perfis vinculados a um serviço aparecem em sua conta do IAM e são de propriedade do serviço. Um administrador pode visualizar, mas não editar, as permissões de funções vinculadas ao serviço.

AWS CodeStar não oferece suporte a funções vinculadas a serviços.

## Perfis de serviço

Esse atributo permite que um serviço assuma um [perfil de serviço](#) em seu nome. O perfil permite que o serviço acesse recursos em outros serviços para concluir uma ação em seu nome. Os perfis

de serviço aparecem em sua conta do IAM e são de propriedade da conta. Isso significa que um administrador do pode alterar as permissões para essa função. Porém, fazer isso pode alterar a funcionalidade do serviço.

A AWS CodeStar oferece suporte a funções de serviço. AWS CodeStar usa uma função de serviço `aws-codestar-service-role`, quando cria e gerencia os recursos do seu projeto. Para obter mais informações, consulte [Termos e conceitos de funções](#) no Guia do usuário do IAM.

#### Important

Você deve estar conectado como um usuário administrativo do ou conta raiz para criar essa função de serviço. Para obter mais informações, consulte [Somente primeiro acesso: as credenciais de usuário raiz](#) e [Criar o primeiro usuário e grupo de administrador do IAM](#) no Guia do usuário do IAM.

Essa função é criada para você na primeira vez que você cria um projeto em AWS CodeStar. A função de serviço atua em seu nome para:

- Criar os recursos escolhidos ao criar um projeto.
- Exiba informações sobre esses recursos no painel do AWS CodeStar projeto.

Ela também atua em seu nome quando você gerencia os recursos de um projeto. Para obter um exemplo dessa declaração de política, consulte [AWSCodeStarServiceRole Política](#).

Além disso, AWS CodeStar cria várias funções de serviço específicas do projeto, dependendo do tipo de projeto. AWS CloudFormation e as funções do conjunto de ferramentas são criadas para cada tipo de projeto.

- AWS CloudFormation as funções permitem acessar AWS CodeStar AWS CloudFormation para criar e modificar pilhas para seu AWS CodeStar projeto.
- As funções do conjunto de ferramentas AWS CodeStar permitem acessar outros AWS serviços para criar e modificar recursos para seu AWS CodeStar projeto.

## AWS CodeStar Políticas e permissões em nível de projeto

Ao criar um projeto, AWS CodeStar cria as funções e políticas do IAM necessárias para gerenciar os recursos do seu projeto. As políticas se encaixam em três categorias:

- Políticas do IAM para membros da equipe do projeto.
- Políticas do IAM para funções de operador.
- Políticas do IAM para uma função de execução em tempo de execução.

## Políticas do IAM para membros de equipe

Ao criar um projeto, AWS CodeStar cria três políticas gerenciadas pelo cliente para acesso do proprietário, do colaborador e do espectador ao projeto. Todos os AWS CodeStar projetos contêm políticas do IAM para esses três níveis de acesso. Esses níveis de acesso são específicos do projeto e definidos por uma política gerenciada do IAM com um nome padrão, onde *project-id* está o ID do AWS CodeStar projeto (por exemplo, *my-first-projec*):

- CodeStar\_*project-id*\_Owner
- CodeStar\_*project-id*\_Contributor
- CodeStar\_*project-id*\_Viewer

### Important

Essas políticas estão sujeitas a alterações por AWS CodeStar. Elas não devem ser editadas manualmente. Se você deseja adicionar ou alterar permissões, anexe políticas adicionais ao usuário do IAM.

À medida que você adiciona membros da equipe (usuários do IAM) ao projeto e escolhe seus níveis de acesso, a política correspondente é anexada ao usuário do IAM, concedendo ao usuário o conjunto de permissões apropriado para atuar em recursos do projeto. Na maioria das circunstâncias, você não precisa anexar ou gerenciar diretamente políticas ou permissões no IAM. Não é recomendável anexar manualmente uma política de nível de AWS CodeStar acesso a um usuário do IAM. Se for absolutamente necessário, como complemento de uma política de nível de AWS CodeStar acesso, você pode criar suas próprias políticas gerenciadas ou em linha para aplicar seu próprio nível de permissões a um usuário do IAM.

As políticas têm um escopo rígido para recursos do projeto e ações específicas. À medida que novos recursos são adicionados à pilha de infraestrutura, AWS CodeStar tenta atualizar as políticas dos membros da equipe para incluir permissões para acessar o novo recurso, se eles forem um dos tipos de recursos compatíveis.

**Note**

As políticas para níveis de acesso em um AWS CodeStar projeto se aplicam somente a esse projeto. Isso ajuda a garantir que os usuários só possam ver e interagir com os AWS CodeStar projetos para os quais têm permissões, no nível determinado por sua função. Somente usuários que criam AWS CodeStar projetos devem ter uma política aplicada que permita o acesso a todos os AWS CodeStar recursos, independentemente do projeto.

Todas as políticas de nível de AWS CodeStar acesso variam, dependendo dos AWS recursos associados ao projeto ao qual os níveis de acesso estão associados. Ao contrário de outros serviços da AWS, essas políticas são personalizadas quando o projeto é criado e atualizado à medida que os recursos do projeto mudam. Portanto, não existe uma política gerenciada de proprietário, colaborador ou visualizador canônica.

#### AWS CodeStar Política de função do proprietário

A política gerenciada pelo `CodeStar_project-id_Owner` cliente permite que o usuário execute todas as ações no AWS CodeStar projeto sem restrições. Essa é a única política que permite que um usuário adicione ou remova membros da equipe. O conteúdo da política varia, dependendo dos recursos associados ao projeto. Consulte [AWS CodeStar Política de função do proprietário](#) para ver um exemplo.

Um usuário do IAM com essa política pode realizar todas as AWS CodeStar ações no projeto, mas, diferentemente de um usuário do IAM com a `AWSCodeStarFullAccess` política, o usuário não pode criar projetos. O escopo da `codestar:*` permissão é limitado a um recurso específico (o AWS CodeStar projeto associado a esse ID do projeto).

#### AWS CodeStar Política de função do colaborador

A política gerenciada pelo cliente `CodeStar_project-id_Contributor` permite que um usuário colabore com o projeto e altere o painel do projeto, mas não permite que um usuário adicione ou remova membros da equipe. O conteúdo da política varia, dependendo dos recursos associados ao projeto. Consulte [Política da função Colaborador do AWS CodeStar](#) para ver um exemplo.

#### AWS CodeStar Política de função do espectador

A política gerenciada pelo cliente `CodeStar_project-id_Viewer` permite que um usuário visualize um projeto no AWS CodeStar, mas não altere seus recursos ou adicione ou remova

membros da equipe. O conteúdo da política varia, dependendo dos recursos associados ao projeto. Consulte [AWS CodeStar Política de função do espectador](#) para ver um exemplo.

## Políticas do IAM para funções de operador

Se você criar seu AWS CodeStar projeto após o PDT de 6 de dezembro de 2018, a AWS CodeStar criará duas funções de trabalhador, `CodeStar-project-id-ToolChain` e `CodeStar-project-id-CloudFormation`. Um papel de trabalhador é um papel do IAM específico do projeto que é AWS CodeStar criado para ser passado para um serviço. Ele concede permissões para que o serviço possa criar recursos e executar ações no contexto do seu AWS CodeStar projeto. A função de trabalhador da cadeia de ferramentas tem uma relação de confiança estabelecida com serviços da cadeia de ferramentas CodeBuild, como CodeDeploy, e CodePipeline. Os membros da equipe do projeto (proprietários e colaboradores) recebem acesso para passar a função de operador para serviços downstream confiáveis. Para obter um exemplo da declaração da política em linha para essa função, consulte [AWS CodeStar Política de funções dos trabalhadores da Toolchain \(após 6 de dezembro de 2018 PDT\)](#).

A função de CloudFormation trabalhador inclui permissões para recursos selecionados suportados por AWS CloudFormation, bem como permissões para criar usuários, funções e políticas do IAM em sua pilha de aplicativos. Também tem uma relação de confiança estabelecida com AWS CloudFormation. Para reduzir os riscos de escalonamento de privilégios e ações destrutivas, a política de AWS CloudFormation função inclui uma condição que exige o limite de permissões específico do projeto para cada entidade do IAM (usuário ou função) criada na pilha de infraestrutura. Para obter um exemplo da declaração da política em linha para essa função, consulte [AWS CloudFormation Política de função do trabalhador](#).

Para CodeStar projetos da AWS criados antes de 6 de dezembro de 2018, o PDT AWS CodeStar cria funções de trabalhadores individuais para recursos da cadeia de ferramentas CodePipeline CodeBuild, como, e CloudWatch Eventos, e também cria uma função de trabalhador AWS CloudFormation que suporta um conjunto limitado de recursos. Cada uma dessas funções tem uma relação de confiança estabelecida com o serviço correspondente. Os membros da equipe do projeto (proprietários e colaboradores) e algumas das outras funções de operador recebem acesso para passar a função para serviços downstream confiáveis. As permissões para as funções de operador são definidas em uma política em linha que é delimitada para um conjunto básico de ações que a função pode executar em um conjunto de recursos do projeto. Essas permissões são estáticas. Elas incluem permissões para recursos incluídos no projeto durante a criação, mas não são atualizadas quando novos recursos são adicionados ao projeto. Para obter exemplos dessas declarações de políticas, consulte.

- [AWS CloudFormation Política de funções do trabalhador \(antes de 6 de dezembro de 2018 PDT\)](#)
- [AWS CodePipeline Política de funções do trabalhador \(antes de 6 de dezembro de 2018 PDT\)](#)
- [AWS CodeBuild Política de funções do trabalhador \(antes de 6 de dezembro de 2018 PDT\)](#)
- [Política de função de funcionários da Amazon CloudWatch Events \(antes de 6 de dezembro de 2018 PDT\)](#)

## Política do IAM para a função de execução

Para projetos criados após o PDT de 6 de dezembro de 2018, a AWS CodeStar cria uma função de execução genérica para o projeto de amostra em sua pilha de aplicativos. A função é limitada aos recursos do projeto usando a política de limite de permissões. À medida que você expande o projeto de amostra, você pode criar funções adicionais do IAM, e a política de AWS CloudFormation funções exige que essas funções sejam definidas usando o limite de permissão para evitar o aumento de privilégios. Para obter mais informações, consulte [Adicionar uma função do IAM a um projeto](#).

Para projetos Lambda criados antes de 6 de dezembro de 2018, o PDT cria AWS CodeStar uma função de execução do Lambda que tem uma política embutida anexada com permissões para atuar nos recursos na pilha de projetos. À medida que novos recursos são adicionados ao modelo do SAM, AWS CodeStar tenta atualizar a política da função de execução do Lambda para incluir permissões para o novo recurso, caso sejam um dos tipos de recursos compatíveis.

## Limite de permissões do IAM

Depois do PDT de 6 de dezembro de 2018, quando você cria um projeto, a AWS CodeStar cria uma política gerenciada pelo cliente e atribui essa política como [limite de permissões do IAM](#) às funções do IAM no projeto. A AWS CodeStar exige que todas as entidades do IAM criadas na pilha de aplicativos tenham um limite de permissões. Um limite de permissões controla as permissões máximas que a função pode ter, mas não fornece a função com nenhuma permissão. As políticas de permissões definem as permissões para a função. Isso significa que, independentemente de quantas permissões extras forem adicionadas a uma função, qualquer pessoa que use a função não poderá executar mais do que as ações incluídas no limite de permissões. Para obter informações sobre como as políticas de permissões e os limites de permissões são avaliados, consulte [Lógica de avaliação de políticas](#) no Guia do usuário do IAM.

A AWS CodeStar usa um limite de permissões específico do projeto para evitar a escalação de privilégios para recursos fora do projeto. O limite de permissões da AWS CodeStar inclui ARNs dos recursos do projeto. Para obter um exemplo dessa declaração de política, consulte [Política de limite de permissões da AWS CodeStar](#).

A CodeStar transformação da AWS atualiza essa política quando você adiciona ou remove um recurso suportado do projeto por meio da pilha de aplicativos (`template.yml`).

## Adicionar um limite de permissões do IAM a projetos existentes

Se você tem um CodeStar projeto da AWS que foi criado antes do PDT de 6 de dezembro de 2018, você deve adicionar manualmente um limite de permissão às funções do IAM no projeto. Como uma melhor prática, recomendamos usar um limite específico ao projeto que inclua apenas recursos no projeto para impedir o escalonamento de privilégios para recursos fora do projeto. Siga estas etapas para usar o limite de permissões CodeStar gerenciadas pela AWS que é atualizado à medida que o projeto evolui.

1. Faça login no AWS CloudFormation console e localize o modelo para a pilha do conjunto de ferramentas em seu projeto. Este modelo é chamado `awscodestar-project-id`.
2. Escolha o modelo, selecione Actions (Ações) e View/Edit template in Designer (Visualizar/editar modelo no Designer).
3. Localize a seção Resources e inclua o trecho de código a seguir na parte superior da seção.

```
PermissionsBoundaryPolicy:
  Description: Creating an IAM managed policy for defining the permissions boundary
for an AWS CodeStar project
  Type: AWS::IAM::ManagedPolicy
  Properties:
    ManagedPolicyName: !Sub 'CodeStar_${ProjectId}_PermissionsBoundary'
    Description: 'IAM policy to define the permissions boundary for IAM entities
created in an AWS CodeStar project'
    PolicyDocument:
      Version: '2012-10-17'
      Statement:
        - Sid: '1'
          Effect: Allow
          Action: ['*']
          Resource:
            - !Sub 'arn:${AWS::Partition}:cloudformation:${AWS::Region}:
${AWS::AccountId}:stack/awscodestar-${ProjectId}-*'

```

Talvez você precise de permissões adicionais do IAM para atualizar a pilha no console do AWS CloudFormation .

4. (Opcional) Para criar perfis do IAM específicas ao aplicativo, execute esta etapa. No console do IAM, atualize a política embutida anexada à AWS CloudFormation função do seu projeto para incluir o seguinte trecho. Talvez você precise recursos adicionais do IAM para atualizar a política.

```
{
  "Action": [
    "iam:PassRole"
  ],
  "Resource": "arn:aws:iam:::{AccountId}:role/CodeStar-{ProjectId}*",
  "Effect": "Allow"
},
{
  "Action": [
    "iam:CreateServiceLinkedRole",
    "iam:GetRole",
    "iam>DeleteRole",
    "iam>DeleteUser"
  ],
  "Resource": "*",
  "Effect": "Allow"
},
{
  "Action": [
    "iam:AttachRolePolicy",
    "iam:AttachUserPolicy",
    "iam:CreateRole",
    "iam:CreateUser",
    "iam>DeleteRolePolicy",
    "iam>DeleteUserPolicy",
    "iam:DetachUserPolicy",
    "iam:DetachRolePolicy",
    "iam:PutUserPermissionsBoundary",
    "iam:PutRolePermissionsBoundary"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "iam:PermissionsBoundary": "arn:aws:iam:::{AccountId}:policy/CodeStar_{ProjectId}_PermissionsBoundary"
    }
  },
  "Effect": "Allow"
}
```

```
}
```

5. Faça uma alteração no pipeline do seu projeto para que a AWS CodeStar atualize o limite de permissões com as permissões apropriadas.

Para obter mais informações, consulte [Adicionar uma função do IAM a um projeto](#).

## Exemplos de políticas CodeStar baseadas em identidade da AWS

Por padrão, os usuários e funções do IAM não têm permissão para criar ou modificar CodeStar recursos da AWS. Eles também não podem realizar tarefas usando a AWS API AWS Management Console AWS CLI, ou. Um administrador deve criar as políticas do IAM que concedam aos usuários e aos perfis permissões para executar operações de API específicas nos recursos especificados que precisam. O administrador deve anexar essas políticas aos usuários ou grupos do IAM que exigem essas permissões.

Para saber como criar uma política baseada em identidade do IAM usando esses exemplos de documentos de política JSON, consulte [Criar políticas na guia JSON](#) no Guia do usuário do IAM.

### Tópicos

- [Melhores práticas de políticas](#)
- [AWSCodeStarServiceRole Política](#)
- [AWSCodeStarFullAccess Política](#)
- [AWS CodeStar Política de função do proprietário](#)
- [Política da função Colaborador do AWS CodeStar](#)
- [AWS CodeStar Política de função do espectador](#)
- [AWS CodeStar Política de funções dos trabalhadores da Toolchain \(após 6 de dezembro de 2018 PDT\)](#)
- [AWS CloudFormation Política de função do trabalhador](#)
- [AWS CloudFormation Política de funções do trabalhador \(antes de 6 de dezembro de 2018 PDT\)](#)
- [AWS CodePipeline Política de funções do trabalhador \(antes de 6 de dezembro de 2018 PDT\)](#)
- [AWS CodeBuild Política de funções do trabalhador \(antes de 6 de dezembro de 2018 PDT\)](#)
- [Política de função de funcionários da Amazon CloudWatch Events \(antes de 6 de dezembro de 2018 PDT\)](#)

- [Política de limite de CodeStar permissões da AWS](#)
- [Listar recursos de um projeto](#)
- [Usando o CodeStar console da AWS](#)
- [Permitir que os usuários visualizem suas próprias permissões](#)
- [Atualizar um projeto do AWS CodeStar](#)
- [Adicionar um membro da equipe a um projeto](#)
- [Listando perfis de usuário associados a uma AWS conta](#)
- [Visualização de CodeStar projetos da AWS com base em tags](#)
- [AWS CodeStar atualizações nas políticas AWS gerenciadas](#)

## Melhores práticas de políticas

As políticas baseadas em identidade determinam se alguém pode criar, acessar ou excluir CodeStar recursos da AWS em sua conta. Essas ações podem incorrer em custos para sua Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece com as políticas AWS gerenciadas e avance para as permissões de privilégios mínimos — Para começar a conceder permissões aos seus usuários e cargas de trabalho, use as políticas AWS gerenciadas que concedem permissões para muitos casos de uso comuns. Eles estão disponíveis no seu Conta da AWS. Recomendamos que você reduza ainda mais as permissões definindo políticas gerenciadas pelo AWS cliente que sejam específicas para seus casos de uso. Para obter mais informações, consulte [Políticas gerenciadas pela AWS](#) ou [Políticas gerenciadas pela AWS para funções de trabalho](#) no Guia do usuário do IAM.
- Aplique permissões de privilégio mínimo: ao definir permissões com as políticas do IAM, conceda apenas as permissões necessárias para executar uma tarefa. Você faz isso definindo as ações que podem ser executadas em recursos específicos sob condições específicas, também conhecidas como permissões de privilégio mínimo. Para obter mais informações sobre como usar o IAM para aplicar permissões, consulte [Políticas e permissões no IAM](#) no Guia do usuário do IAM.
- Use condições nas políticas do IAM para restringir ainda mais o acesso: você pode adicionar uma condição às políticas para limitar o acesso a ações e recursos. Por exemplo, você pode escrever uma condição de política para especificar que todas as solicitações devem ser enviadas usando SSL. Você também pode usar condições para conceder acesso às ações de serviço se elas forem usadas por meio de uma ação específica AWS service (Serviço da AWS), como AWS CloudFormation. Para obter mais informações, consulte [Elementos da política JSON do IAM: condição](#) no Guia do usuário do IAM.

- Use o IAM Access Analyzer para validar suas políticas do IAM a fim de garantir permissões seguras e funcionais: o IAM Access Analyzer valida as políticas novas e existentes para que elas sigam a linguagem de política do IAM (JSON) e as práticas recomendadas do IAM. O IAM Access Analyzer oferece mais de 100 verificações de política e recomendações acionáveis para ajudá-lo a criar políticas seguras e funcionais. Para obter mais informações, consulte [Validação de políticas do IAM Access Analyzer](#) no Guia do usuário do IAM.
- Exigir autenticação multifator (MFA) — Se você tiver um cenário que exija usuários do IAM ou um usuário root, ative Conta da AWS a MFA para obter segurança adicional. Para exigir MFA quando as operações de API forem chamadas, adicione condições de MFA às suas políticas. Para obter mais informações, consulte [Configuração de acesso à API protegido por MFA](#) no Guia do Usuário do IAM.

Para obter mais informações sobre as práticas recomendadas do IAM, consulte [Práticas Recomendadas de Segurança no IAM](#) no Guia do usuário do IAM.

## AWSCodeStarServiceRole Política

A `aws-codestar-service-role` política é anexada à função de serviço que AWS CodeStar permite realizar ações com outros serviços. Na primeira vez que você faz login AWS CodeStar, você cria a função de serviço. É necessário criá-la somente uma vez. A política será anexada automaticamente à função de serviço depois que ela for criada.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ProjectEventRules",
      "Effect": "Allow",
      "Action": [
        "events:PutTargets",
        "events:RemoveTargets",
        "events:PutRule",
        "events>DeleteRule",
        "events:DescribeRule"
      ],
      "Resource": [
        "arn:aws:events:*:*:rule/awscodestar-*"
      ]
    },
  ],
}
```

```

    "Sid": "ProjectStack",
    "Effect": "Allow",
    "Action": [
        "cloudformation:*Stack*",
        "cloudformation:CreateChangeSet",
        "cloudformation:ExecuteChangeSet",
        "cloudformation>DeleteChangeSet",
        "cloudformation:GetTemplate"
    ],
    "Resource": [
        "arn:aws:cloudformation:*:*:stack/awscodestar-*",
        "arn:aws:cloudformation:*:*:stack/awseb-*",
        "arn:aws:cloudformation:*:*:stack/aws-cloud9-*",
        "arn:aws:cloudformation:*:aws:transform/CodeStar*"
    ]
},
{
    "Sid": "ProjectStackTemplate",
    "Effect": "Allow",
    "Action": [
        "cloudformation:GetTemplateSummary",
        "cloudformation:DescribeChangeSet"
    ],
    "Resource": "*"
},
{
    "Sid": "ProjectQuickstarts",
    "Effect": "Allow",
    "Action": [
        "s3:GetObject"
    ],
    "Resource": [
        "arn:aws:s3:::awscodestar-*/*"
    ]
},
{
    "Sid": "ProjectS3Buckets",
    "Effect": "Allow",
    "Action": [
        "s3:*"
    ],
    "Resource": [
        "arn:aws:s3:::aws-codestar-*",
        "arn:aws:s3:::elasticbeanstalk-*"
    ]
}

```

```

    ]
  },
  {
    "Sid": "ProjectServices",
    "Effect": "Allow",
    "Action": [
      "codestar:*",
      "codecommit:*",
      "codepipeline:*",
      "codedeploy:*",
      "codebuild:*",
      "autoscaling:*",
      "cloudwatch:Put*",
      "ec2:*",
      "elasticbeanstalk:*",
      "elasticloadbalancing:*",
      "iam:ListRoles",
      "logs:*",
      "sns:*",
      "cloud9:CreateEnvironmentEC2",
      "cloud9>DeleteEnvironment",
      "cloud9:DescribeEnvironment*",
      "cloud9:ListEnvironments"
    ],
    "Resource": "*"
  },
  {
    "Sid": "ProjectWorkerRoles",
    "Effect": "Allow",
    "Action": [
      "iam:AttachRolePolicy",
      "iam:CreateRole",
      "iam>DeleteRole",
      "iam>DeleteRolePolicy",
      "iam:DetachRolePolicy",
      "iam:GetRole",
      "iam:PassRole",
      "iam:GetRolePolicy",
      "iam:PutRolePolicy",
      "iam:SetDefaultPolicyVersion",
      "iam:CreatePolicy",
      "iam>DeletePolicy",
      "iam:AddRoleToInstanceProfile",
      "iam:CreateInstanceProfile",

```

```

        "iam:DeleteInstanceProfile",
        "iam:RemoveRoleFromInstanceProfile"
    ],
    "Resource": [
        "arn:aws:iam::*:role/CodeStarWorker*",
        "arn:aws:iam::*:policy/CodeStarWorker*",
        "arn:aws:iam::*:instance-profile/awscodestar-*"
    ]
},
{
    "Sid": "ProjectTeamMembers",
    "Effect": "Allow",
    "Action": [
        "iam:AttachUserPolicy",
        "iam:DetachUserPolicy"
    ],
    "Resource": "*",
    "Condition": {
        "ArnEquals": {
            "iam:PolicyArn": [
                "arn:aws:iam::*:policy/CodeStar_*"
            ]
        }
    }
},
{
    "Sid": "ProjectRoles",
    "Effect": "Allow",
    "Action": [
        "iam:CreatePolicy",
        "iam>DeletePolicy",
        "iam:CreatePolicyVersion",
        "iam>DeletePolicyVersion",
        "iam>ListEntitiesForPolicy",
        "iam>ListPolicyVersions",
        "iam:GetPolicy",
        "iam:GetPolicyVersion"
    ],
    "Resource": [
        "arn:aws:iam::*:policy/CodeStar_*"
    ]
},
{
    "Sid": "InspectServiceRole",

```

```

    "Effect": "Allow",
    "Action": [
      "iam:ListAttachedRolePolicies"
    ],
    "Resource": [
      "arn:aws:iam::*:role/aws-codestar-service-role",
      "arn:aws:iam::*:role/service-role/aws-codestar-service-role"
    ]
  },
  {
    "Sid": "IAMLinkRole",
    "Effect": "Allow",
    "Action": [
      "iam:CreateServiceLinkedRole"
    ],
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "iam:AWSServiceName": "cloud9.amazonaws.com"
      }
    }
  },
  {
    "Sid": "DescribeConfigRuleForARN",
    "Effect": "Allow",
    "Action": [
      "config:DescribeConfigRules"
    ],
    "Resource": [
      "*"
    ]
  },
  {
    "Sid": "ProjectCodeStarConnections",
    "Effect": "Allow",
    "Action": [
      "codestar-connections:UseConnection",
      "codestar-connections:GetConnection"
    ],
    "Resource": "*"
  },
  {
    "Sid": "ProjectCodeStarConnectionsPassConnections",
    "Effect": "Allow",

```

```

    "Action": "codestar-connections:PassConnection",
    "Resource": "*",
    "Condition": {
      "StringEqualsIfExists": {
        "codestar-connections:PassedToService":
"codepipeline.amazonaws.com"
      }
    }
  ]
}

```

## AWSCodeStarFullAccess Política

Nas instruções [Configuração AWS CodeStar](#), você anexou uma política chamada `AWSCodeStarFullAccess` ao usuário do IAM. Esta declaração de política permite que o usuário execute todas as ações disponíveis AWS CodeStar com todos os AWS CodeStar recursos disponíveis associados à AWS conta. Isso inclui a criação e a exclusão de projetos. O exemplo a seguir é um trecho de uma política `AWSCodeStarFullAccess` representativa. A política real difere dependendo do modelo selecionado ao iniciar um novo AWS CodeStar projeto.

A AWS CloudFormation exige `cloudformation::ListStacks` permissão ao fazer chamadas `cloudformation::DescribeStacks` sem uma pilha de destino.

### Detalhes das permissões

Esta política inclui as permissões para fazer o seguinte:

- `ec2`—Recupere informações sobre EC2 instâncias para criar um AWS CodeStar projeto.
- `cloud9`— Recupere informações sobre AWS Command Line Interface ambientes.
- `cloudformation`—Recupere informações sobre pilhas de AWS CodeStar projetos.
- `codestar`—Execute ações dentro de um AWS CodeStar projeto.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CodeStarEC2",
      "Effect": "Allow",
      "Action": [

```

```

    "codestar:*",
    "ec2:DescribeKeyPairs",
    "ec2:DescribeVpcs",
    "ec2:DescribeSubnets",
    "cloud9:DescribeEnvironment*"
  ],
  "Resource": "*"
},
{
  "Sid": "CodeStarCF",
  "Effect": "Allow",
  "Action": [
    "cloudformation:DescribeStack*",
    "cloudformation:ListStacks*",
    "cloudformation:GetTemplateSummary"
  ],
  "Resource": [
    "arn:aws:cloudformation:*:*:stack/awscodestar-*"
  ]
}
]
}

```

Talvez você não queira conceder todo esse acesso a todos os usuários. Em vez disso, você pode adicionar permissões em nível de projeto usando funções de projeto gerenciadas por AWS CodeStar. As funções concedem níveis específicos de acesso aos AWS CodeStar projetos e são nomeadas da seguinte forma:

- Proprietário
- Colaborador
- Visualizador

## AWS CodeStar Política de função do proprietário

A política de função de CodeStar proprietário da AWS permite que um usuário execute todas as ações em um CodeStar projeto da AWS sem restrições. A AWS CodeStar aplica a CodeStar\_*project-id*\_Owner política aos membros da equipe do projeto com o nível de acesso do proprietário.

...

```

{
  "Effect": "Allow",
  "Action": [
    ...
    "codestar:*",
    ...
  ],
  "Resource": [
    "arn:aws:codestar:us-east-2:111111111111:project/project-id",
    "arn:aws:iam::account-id:policy/CodeStar_project-id_Owner"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:DescribeUserProfile",
    "codestar:ListProjects",
    "codestar:ListUserProfiles",
    "codestar:VerifyServiceRole",
    ...
  ],
  "Resource": [
    "*"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:*UserProfile",
    ...
  ],
  "Resource": [
    "arn:aws:iam::account-id:user/user-name"
  ]
}
...

```

## Política da função Colaborador do AWS CodeStar

A política de função de CodeStar colaborador da AWS permite que um usuário contribua com o projeto e altere o painel do projeto. A AWS CodeStar aplica a CodeStar\_*project-id*\_Contributor política aos membros da equipe do projeto com o nível de acesso de

colaborador. Os usuários com acesso de colaborador podem colaborar com o projeto e alterar o painel do projeto, mas não podem adicionar ou remover membros da equipe.

```

...
{
  "Effect": "Allow",
  "Action": [
    ...
    "codestar:Describe*",
    "codestar:Get*",
    "codestar:List*",
    "codestar:PutExtendedAccess",
    ...
  ],
  "Resource": [
    "arn:aws:codestar:us-east-2:111111111111:project/project-id",
    "arn:aws:iam::account-id:policy/CodeStar_project-id_Contributor"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:DescribeUserProfile",
    "codestar:ListProjects",
    "codestar:ListUserProfiles",
    "codestar:VerifyServiceRole",
    ...
  ],
  "Resource": [
    "*"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:*UserProfile",
    ...
  ],
  "Resource": [
    "arn:aws:iam::account-id:user/user-name"
  ]
}
...

```

## AWS CodeStar Política de função do espectador

A política de função de CodeStar visualizador da AWS permite que um usuário visualize um projeto na AWS CodeStar. A AWS CodeStar aplica a CodeStar\_*project-id*\_Viewer política aos membros da equipe do projeto com o nível de acesso do espectador. Usuários com acesso de visualizador podem visualizar um projeto na AWS CodeStar, mas não podem alterar seus recursos nem adicionar ou remover membros da equipe.

```
...
{
  "Effect": "Allow",
  "Action": [
    ...
    "codestar:Describe*",
    "codestar:Get*",
    "codestar:List*",
    ...
  ],
  "Resource": [
    "arn:aws:codestar:us-east-2:111111111111:project/project-id",
    "arn:aws:iam:account-id:policy/CodeStar_project-id_Viewer"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:DescribeUserProfile",
    "codestar:ListProjects",
    "codestar:ListUserProfiles",
    "codestar:VerifyServiceRole",
    ...
  ],
  "Resource": [
    "*"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:*UserProfile",
    ...
  ],
  "Resource": [
```

```

    "arn:aws:iam::account-id:user/user-name"
  ]
}
...

```

## AWS CodeStar Política de funções dos trabalhadores da Toolchain (após 6 de dezembro de 2018 PDT)

Para AWS CodeStar projetos criados após o PDT de 6 de dezembro de 2018, a AWS CodeStar cria uma política embutida para uma função de trabalhador que cria recursos para seu projeto em outros AWS serviços. O conteúdo da política depende do tipo de projeto que está sendo criado. A política a seguir é um exemplo. Para obter mais informações, consulte [Políticas do IAM para funções de operador](#).

```

{
  "Statement": [
    {
      "Action": [
        "s3:GetObject",
        "s3:GetObjectVersion",
        "s3:GetBucketVersioning",
        "s3:PutObject*",
        "codecommit:CancelUploadArchive",
        "codecommit:GetBranch",
        "codecommit:GetCommit",
        "codecommit:GetUploadArchiveStatus",
        "codecommit:GitPull",
        "codecommit:UploadArchive",
        "codebuild:StartBuild",
        "codebuild:BatchGetBuilds",
        "codebuild:StopBuild",
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents",
        "cloudformation:DescribeStacks",
        "cloudformation:DescribeChangeSet",
        "cloudformation:CreateChangeSet",
        "cloudformation>DeleteChangeSet",
        "cloudformation:ExecuteChangeSet",
        "codepipeline:StartPipelineExecution",
        "lambda:ListFunctions",
        "lambda:InvokeFunction",

```

```

    "sns:Publish"
  ],
  "Resource": [
    "*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "iam:PassRole"
  ],
  "Resource": [
    "*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "kms:GenerateDataKey*",
    "kms:Encrypt",
    "kms:Decrypt"
  ],
  "Resource": [
    "*"
  ],
  "Effect": "Allow"
}
]
}

```

## AWS CloudFormation Política de função do trabalhador

Para AWS CodeStar projetos criados após o PDT de 6 de dezembro de 2018, a AWS CodeStar cria uma política embutida para uma função de trabalhador que cria AWS CloudFormation recursos para seu projeto da AWS CodeStar . O conteúdo da política depende do tipo de recursos necessários para o projeto. A política a seguir é um exemplo. Para obter mais informações, consulte [Políticas do IAM para funções de operador](#).

```

{
{
  "Statement": [
    {

```

```

    "Action": [
      "s3:PutObject",
      "s3:GetObject",
      "s3:GetObjectVersion"
    ],
    "Resource": [
      "arn:aws:s3:::aws-codestar-region-id-account-id-project-id",
      "arn:aws:s3:::aws-codestar-region-id-account-id-project-id/*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "apigateway:DELETE",
      "apigateway:GET",
      "apigateway:PATCH",
      "apigateway:POST",
      "apigateway:PUT",
      "codedeploy:CreateApplication",
      "codedeploy:CreateDeployment",
      "codedeploy:CreateDeploymentConfig",
      "codedeploy:CreateDeploymentGroup",
      "codedeploy>DeleteApplication",
      "codedeploy>DeleteDeployment",
      "codedeploy>DeleteDeploymentConfig",
      "codedeploy>DeleteDeploymentGroup",
      "codedeploy:GetDeployment",
      "codedeploy:GetDeploymentConfig",
      "codedeploy:GetDeploymentGroup",
      "codedeploy:RegisterApplicationRevision",
      "codestar:SyncResources",
      "config>DeleteConfigRule",
      "config:DescribeConfigRules",
      "config:ListTagsForResource",
      "config:PutConfigRule",
      "config:TagResource",
      "config:UntagResource",
      "dynamodb>CreateTable",
      "dynamodb>DeleteTable",
      "dynamodb:DescribeContinuousBackups",
      "dynamodb:DescribeTable",
      "dynamodb:DescribeTimeToLive",
      "dynamodb:ListTagsOfResource",
      "dynamodb:TagResource",

```

```
"dynamodb:UntagResource",
"dynamodb:UpdateContinuousBackups",
"dynamodb:UpdateTable",
"dynamodb:UpdateTimeToLive",
"ec2:AssociateIamInstanceProfile",
"ec2:AttachVolume",
"ec2:CreateSecurityGroup",
"ec2:createTags",
"ec2:DescribeIamInstanceProfileAssociations",
"ec2:DescribeInstances",
"ec2:DescribeSecurityGroups",
"ec2:DescribeSubnets",
"ec2:DetachVolume",
"ec2:DisassociateIamInstanceProfile",
"ec2:ModifyInstanceAttribute",
"ec2:ModifyInstanceCreditSpecification",
"ec2:ModifyInstancePlacement",
"ec2:MonitorInstances",
"ec2:ReplaceIamInstanceProfileAssociation",
"ec2:RunInstances",
"ec2:StartInstances",
"ec2:StopInstances",
"ec2:TerminateInstances",
"events:DeleteRule",
"events:DescribeRule",
"events:ListTagsForResource",
"events:PutRule",
"events:PutTargets",
"events:RemoveTargets",
"events:TagResource",
"events:UntagResource",
"kinesis:AddTagsToStream",
"kinesis:CreateStream",
"kinesis:DecreaseStreamRetentionPeriod",
"kinesis>DeleteStream",
"kinesis:DescribeStream",
"kinesis:IncreaseStreamRetentionPeriod",
"kinesis:RemoveTagsFromStream",
"kinesis:StartStreamEncryption",
"kinesis:StopStreamEncryption",
"kinesis:UpdateShardCount",
"lambda:CreateAlias",
"lambda:CreateFunction",
"lambda>DeleteAlias",
```

```
"lambda:DeleteFunction",
"lambda:DeleteFunctionConcurrency",
"lambda:GetFunction",
"lambda:GetFunctionConfiguration",
"lambda:ListTags",
"lambda:ListVersionsByFunction",
"lambda:PublishVersion",
"lambda:PutFunctionConcurrency",
"lambda:TagResource",
"lambda:UntagResource",
"lambda:UpdateAlias",
"lambda:UpdateFunctionCode",
"lambda:UpdateFunctionConfiguration",
"s3:CreateBucket",
"s3:DeleteBucket",
"s3:DeleteBucketWebsite",
"s3:PutAccelerateConfiguration",
"s3:PutAnalyticsConfiguration",
"s3:PutBucketAcl",
"s3:PutBucketCORS",
"s3:PutBucketLogging",
"s3:PutBucketNotification",
"s3:PutBucketPublicAccessBlock",
"s3:PutBucketVersioning",
"s3:PutBucketWebsite",
"s3:PutEncryptionConfiguration",
"s3:PutInventoryConfiguration",
"s3:PutLifecycleConfiguration",
"s3:PutMetricsConfiguration",
"s3:PutReplicationConfiguration",
"sns:CreateTopic",
"sns:DeleteTopic",
"sns:GetTopicAttributes",
"sns:ListSubscriptionsByTopic",
"sns:ListTopics",
"sns:SetSubscriptionAttributes",
"sns:Subscribe",
"sns:Unsubscribe",
"sqs:CreateQueue",
"sqs:DeleteQueue",
"sqs:GetQueueAttributes",
"sqs:GetQueueUrl",
"sqs:ListQueueTags",
"sqs:TagQueue",
```

```

        "sqs:UntagQueue"
    ],
    "Resource": "*",
    "Effect": "Allow"
  },
  {
    "Action": [
      "lambda:AddPermission",
      "lambda:RemovePermission"
    ],
    "Resource": [
      "arn:aws:lambda:region-id:account-id:function:awscodestar-*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "iam:PassRole"
    ],
    "Resource": [
      "arn:aws:iam::account-id:role/CodeStar-project-id*"
    ],
    "Effect": "Allow"
  },
  {
    "Condition": {
      "StringEquals": {
        "iam:PassedToService": "codedeploy.amazonaws.com"
      }
    },
    "Action": [
      "iam:PassRole"
    ],
    "Resource": [
      "arn:aws:iam::account-id:role/CodeStarWorker-project-id-CodeDeploy"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "cloudformation:CreateChangeSet"
    ],
    "Resource": [
      "arn:aws:cloudformation:region-id:aws:transform/Serverless-2016-10-31",

```

```

        "arn:aws:cloudformation:region-id:aws:transform/CodeStar"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "iam:CreateServiceLinkedRole",
        "iam:GetRole",
        "iam>DeleteRole",
        "iam>DeleteUser"
    ],
    "Resource": "*",
    "Effect": "Allow"
},
{
    "Condition": {
        "StringEquals": {
            "iam:PermissionsBoundary": "arn:aws:iam::account-id:policy/
CodeStar_project-id_PermissionsBoundary"
        }
    },
    "Action": [
        "iam:AttachRolePolicy",
        "iam:AttachUserPolicy",
        "iam:CreateRole",
        "iam:CreateUser",
        "iam>DeleteRolePolicy",
        "iam>DeleteUserPolicy",
        "iam:DetachUserPolicy",
        "iam:DetachRolePolicy",
        "iam:PutUserPermissionsBoundary",
        "iam:PutRolePermissionsBoundary"
    ],
    "Resource": "*",
    "Effect": "Allow"
},
{
    "Action": [
        "kms:CreateKey",
        "kms:CreateAlias",
        "kms>DeleteAlias",
        "kms:DisableKey",
        "kms:EnableKey",
        "kms:UpdateAlias",

```

```

        "kms:TagResource",
        "kms:UntagResource"
    ],
    "Resource": "*",
    "Effect": "Allow"
},
{
    "Condition": {
        "StringEquals": {
            "ssm:ResourceTag/awscodestar:projectArn":
"arn:aws:codestar:project-id:account-id:project/project-id"
        }
    },
    "Action": [
        "ssm:GetParameter*"
    ],
    "Resource": "*",
    "Effect": "Allow"
}
]
}

```

## AWS CloudFormation Política de funções do trabalhador (antes de 6 de dezembro de 2018 PDT)

Se seu CodeStar projeto da AWS foi criado antes do PDT de 6 de dezembro de 2018, a AWS CodeStar criou uma política em linha para uma função de AWS CloudFormation trabalhador. A declaração de política a seguir é um exemplo.

```

{
    "Statement": [
        {
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:GetObjectVersion"
            ],
            "Resource": [
                "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-pipe",
                "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-pipe/*"
            ],
            "Effect": "Allow"
        }
    ]
}

```

```
    },
    {
      "Action": [
        "codestar:SyncResources",
        "lambda:CreateFunction",
        "lambda>DeleteFunction",
        "lambda:AddPermission",
        "lambda:UpdateFunction",
        "lambda:UpdateFunctionCode",
        "lambda:GetFunction",
        "lambda:GetFunctionConfiguration",
        "lambda:UpdateFunctionConfiguration",
        "lambda:RemovePermission",
        "lambda:listTags",
        "lambda:TagResource",
        "lambda:UntagResource",
        "apigateway:*",
        "dynamodb:CreateTable",
        "dynamodb>DeleteTable",
        "dynamodb:DescribeTable",
        "kinesis:CreateStream",
        "kinesis>DeleteStream",
        "kinesis:DescribeStream",
        "sns:CreateTopic",
        "sns>DeleteTopic",
        "sns:ListTopics",
        "sns:GetTopicAttributes",
        "sns:SetTopicAttributes",
        "s3:CreateBucket",
        "s3>DeleteBucket",
        "config:DescribeConfigRules",
        "config:PutConfigRule",
        "config>DeleteConfigRule",
        "ec2:*",
        "autoscaling:*",
        "elasticloadbalancing:*",
        "elasticbeanstalk:*"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "iam:PassRole"
```

```

    ],
    "Resource": [
        "arn:aws:iam::account-id:role/CodeStarWorker-project-id-Lambda"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "cloudformation:CreateChangeSet"
    ],
    "Resource": [
        "arn:aws:cloudformation:us-east-1:aws:transform/Serverless-2016-10-31",
        "arn:aws:cloudformation:us-east-1:aws:transform/CodeStar"
    ],
    "Effect": "Allow"
}
]
}

```

## AWS CodePipeline Política de funções do trabalhador (antes de 6 de dezembro de 2018 PDT)

Se seu CodeStar projeto da AWS foi criado antes do PDT de 6 de dezembro de 2018, a AWS CodeStar criou uma política em linha para uma função de CodePipeline trabalhador. A declaração de política a seguir é um exemplo.

```

{
  "Statement": [
    {
      "Action": [
        "s3:GetObject",
        "s3:GetObjectVersion",
        "s3:GetBucketVersioning",
        "s3:PutObject"
      ],
      "Resource": [
        "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-pipe",
        "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-pipe/*"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [

```

```

        "codecommit:CancelUploadArchive",
        "codecommit:GetBranch",
        "codecommit:GetCommit",
        "codecommit:GetUploadArchiveStatus",
        "codecommit:UploadArchive"
    ],
    "Resource": [
        "arn:aws:codecommit:us-east-1:account-id:project-id"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "codebuild:StartBuild",
        "codebuild:BatchGetBuilds",
        "codebuild:StopBuild"
    ],
    "Resource": [
        "arn:aws:codebuild:us-east-1:account-id:project/project-id"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "cloudformation:DescribeStacks",
        "cloudformation:DescribeChangeSet",
        "cloudformation:CreateChangeSet",
        "cloudformation>DeleteChangeSet",
        "cloudformation:ExecuteChangeSet"
    ],
    "Resource": [
        "arn:aws:cloudformation:us-east-1:account-id:stack/awscodestar-project-id-lambda/*"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "iam:PassRole"
    ],
    "Resource": [
        "arn:aws:iam::account-id:role/CodeStarWorker-project-id-CloudFormation"
    ],
    "Effect": "Allow"
}

```

```

    }
  ]
}

```

## AWS CodeBuild Política de funções do trabalhador (antes de 6 de dezembro de 2018 PDT)

Se seu CodeStar projeto da AWS foi criado antes do PDT de 6 de dezembro de 2018, a AWS CodeStar criou uma política em linha para uma função de CodeBuild trabalhador. A declaração de política a seguir é um exemplo.

```

{
  "Statement": [
    {
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:GetObjectVersion"
      ],
      "Resource": [
        "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-pipe",
        "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-pipe/*",
        "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-app",
        "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-app/*"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "codecommit:GitPull"
      ],
      "Resource": [
        "arn:aws:codecommit:us-east-1:account-id:project-id"
      ],
    }
  ]
}

```

```

    "Effect": "Allow"
  },
  {
    "Action": [
      "kms:GenerateDataKey*",
      "kms:Encrypt",
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:us-east-1:account-id:alias/aws/s3"
    ],
    "Effect": "Allow"
  }
]
}

```

## Política de função de funcionários da Amazon CloudWatch Events (antes de 6 de dezembro de 2018 PDT)

Se seu CodeStar projeto da AWS foi criado antes do PDT de 6 de dezembro de 2018, a AWS CodeStar criou uma política em linha para uma função de trabalhador de CloudWatch eventos. A declaração de política a seguir é um exemplo.

```

{
  "Statement": [
    {
      "Action": [
        "codepipeline:StartPipelineExecution"
      ],
      "Resource": [
        "arn:aws:codepipeline:us-east-1:account-id:project-id-Pipeline"
      ],
      "Effect": "Allow"
    }
  ]
}

```

## Política de limite de CodeStar permissões da AWS

Se você criar um CodeStar projeto da AWS após o PDT de 6 de dezembro de 2018, a AWS CodeStar criará uma política de limite de permissões para seu projeto. Essa política impede o

escalonamento de privilégios a recursos fora do projeto. É uma política dinâmica que é atualizada conforme o projeto evolui. O conteúdo da política depende do tipo de projeto que está sendo criado. A política a seguir é um exemplo. Para obter mais informações, consulte [Limite de permissões do IAM](#).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "1",
      "Effect": "Allow",
      "Action": [
        "s3:GetObject"
      ],
      "Resource": [
        "arn:aws:s3::*/AWSLogs/*/Config/*"
      ]
    },
    {
      "Sid": "2",
      "Effect": "Allow",
      "Action": [
        "*"
      ],
      "Resource": [
        "arn:aws:codestar:us-east-1:account-id:project/project-id",
        "arn:aws:cloudformation:us-east-1:account-id:stack/awscodestar-project-id-lambda/eefbbf20-c1d9-11e8-8a3a-500c28b4e461",
        "arn:aws:cloudformation:us-east-1:account-id:stack/awscodestar-project-id/4b80b3f0-c1d9-11e8-8517-500c28b236fd",
        "arn:aws:codebuild:us-east-1:account-id:project/project-id",
        "arn:aws:codecommit:us-east-1:account-id:project-id",
        "arn:aws:codepipeline:us-east-1:account-id:project-id-Pipeline",
        "arn:aws:execute-api:us-east-1:account-id:7rlst5mrgi",
        "arn:aws:iam::account-id:role/CodeStarWorker-project-id-CloudFormation",
        "arn:aws:iam::account-id:role/CodeStarWorker-project-id-CloudWatchEventRule",
        "arn:aws:iam::account-id:role/CodeStarWorker-project-id-CodeBuild",
        "arn:aws:iam::account-id:role/CodeStarWorker-project-id-CodePipeline",
        "arn:aws:iam::account-id:role/CodeStarWorker-project-id-Lambda",
        "arn:aws:lambda:us-east-1:account-id:function:awscodestar-project-id-lambda-GetHelloWorld-KFKTXYNH9573",
        "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-app",
        "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-pipe"
      ]
    }
  ]
}
```

```

    ]
  },
  {
    "Sid": "3",
    "Effect": "Allow",
    "Action": [
      "apigateway:GET",
      "config:Describe*",
      "config:Get*",
      "config:List*",
      "config:Put*",
      "logs:CreateLogGroup",
      "logs:CreateLogStream",
      "logs:DescribeLogGroups",
      "logs:PutLogEvents"
    ],
    "Resource": [
      "*"
    ]
  }
]
}

```

## Listar recursos de um projeto

Neste exemplo, você quer conceder a um usuário do IAM especificado em sua AWS conta acesso para listar os recursos de um AWS CodeStar projeto.

```

{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "codestar:ListResources",
      ],
      "Resource" : "arn:aws:codestar:us-east-2:project/my-first-projec"
    }
  ]
}

```

## Usando o CodeStar console da AWS

Nenhuma permissão específica é necessária para acessar o CodeStar console da AWS, mas você não pode fazer nada útil a menos que tenha a `AWSCodeStarFullAccess` política ou uma das funções em AWS CodeStar nível de projeto: Proprietário, Colaborador ou Visualizador. Para obter mais informações sobre `AWSCodeStarFullAccess`, consulte [AWSCodeStarFullAccess Política](#). Para obter mais informações sobre as políticas de nível de projeto, consulte [Políticas do IAM para membros de equipe](#).

Você não precisa permitir permissões mínimas do console para usuários que estão fazendo chamadas somente para a API AWS CLI ou para a AWS API. Em vez disso, permita o acesso somente às ações que correspondem à operação da API que você está tentando executar.

### Permitir que os usuários visualizem suas próprias permissões

Este exemplo mostra como criar uma política que permita que os usuários do IAM visualizem as políticas gerenciadas e em linha anexadas a sua identidade de usuário. Essa política inclui permissões para concluir essa ação no console ou programaticamente usando a API AWS CLI ou AWS .

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsWithUser",
        "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": "*"
}
]
}

```

## Atualizar um projeto do AWS CodeStar

Neste exemplo, você quer conceder a um usuário do IAM especificado em sua AWS conta acesso para editar os atributos de um AWS CodeStar projeto, como a descrição do projeto.

```

{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "codestar:UpdateProject"
      ],
      "Resource" : "arn:aws:codestar:us-east-2:project/my-first-projec"
    }
  ]
}

```

## Adicionar um membro da equipe a um projeto

Neste exemplo, você quer conceder a um usuário do IAM específico a capacidade de adicionar membros da equipe a um AWS CodeStar projeto com o ID do projeto *my-first-projec*, mas negar explicitamente a esse usuário a capacidade de remover membros da equipe:

```

{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "codestar:AssociateTeamMember",

```

```

    ],
    "Resource" : "arn:aws:codestar:us-east-2:project/my-first-projec"
  },
  {
    "Effect" : "Deny",
    "Action" : [
      "codestar:DisassociateTeamMember",
    ],
    "Resource" : "arn:aws:codestar:us-east-2:project/my-first-projec"
  }
]
}

```

## Listando perfis de usuário associados a uma AWS conta

Neste exemplo, você permite que um usuário do IAM que tenha essa política anexada liste todos os perfis de AWS CodeStar usuário associados a uma AWS conta:

```

{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "codestar:ListUserProfiles",
      ],
      "Resource" : "*"
    }
  ]
}

```

## Visualização de CodeStar projetos da AWS com base em tags

Você pode usar condições em sua política baseada em identidade para controlar o acesso aos CodeStar projetos da AWS com base em tags. Este exemplo mostra como é possível criar uma política que permite visualizar um projeto. No entanto, a permissão será concedida somente se a tag Owner do projeto tiver o valor do nome de usuário desse usuário. Essa política também concede as permissões necessárias concluir essa ação no console.

```

{

```

```

"Version": "2012-10-17",
"Statement": [
  {
    "Sid": "ListProjectsInConsole",
    "Effect": "Allow",
    "Action": "codestar:ListProjects",
    "Resource": "*"
  },
  {
    "Sid": "ViewProjectIfOwner",
    "Effect": "Allow",
    "Action": "codestar:GetProject",
    "Resource": "arn:aws:codestar:*:*:project/*",
    "Condition": {
      "StringEquals": {"codestar:ResourceTag/Owner": "${aws:username}"}
    }
  }
]
}

```

É possível anexar essa política aos usuários do IAM na sua conta. Se um usuário chamado `richard-roe` tentar visualizar um CodeStar projeto da AWS, o projeto deverá ser marcado como `Owner=richard-roe` ou `owner=richard-roe`. Caso contrário, ele terá o acesso negado. A chave da tag de condição `Owner` corresponde a `Owner` e a `owner` porque os nomes das chaves de condição não fazem distinção entre maiúsculas e minúsculas. Para obter mais informações, consulte [IAM JSON Policy Elements: Condition](#) (Elementos da política JSON do IAM: Condição) no Guia do usuário do IAM.

## AWS CodeStar atualizações nas políticas AWS gerenciadas

Veja detalhes sobre as atualizações das políticas AWS gerenciadas da AWS CodeStar desde que esse serviço começou a monitorar essas mudanças. Para receber alertas automáticos sobre alterações nessa página, assine o feed RSS na página de [histórico de CodeStar documentos](#) da AWS.

Alteração	Descrição	Data
<a href="#">AWSCodeStarFullAccess Política</a> — Atualize a	A política de função de AWS CodeStar acesso foi atualizad	24 de março de 2023

Alteração	Descrição	Data
AWSCodeStarFullAccess política	a. O resultado da política é o mesmo, mas o cloudformation exige um ListStacks acréscimo ao DescribeStacks que já é necessário.	
<a href="#">AWSCodeStarService Role Política</a> — Atualize a AWSCodeStarServiceRole política	<p>A política da função de CodeStar serviço da AWS foi atualizada para corrigir ações redundantes na declaração de política.</p> <p>A política de função de serviço permite que o CodeStar serviço da AWS execute ações em seu nome.</p>	23 de setembro de 2021
A AWS CodeStar começou a monitorar as mudanças	CodeStar A AWS começou a monitorar as mudanças em suas políticas AWS gerenciadas.	23 de setembro de 2021

## Solução de problemas de CodeStar identidade e acesso da AWS

Use as informações a seguir para ajudá-lo a diagnosticar e corrigir problemas comuns que você pode encontrar ao trabalhar com a AWS CodeStar e o IAM.

### Tópicos

- [Não estou autorizado a realizar uma ação na AWS CodeStar](#)
- [Não estou autorizado a realizar iam: PassRole](#)
- [Quero permitir que pessoas fora da minha AWS conta acessem meus CodeStar recursos da AWS](#)

## Não estou autorizado a realizar uma ação na AWS CodeStar

Se isso AWS Management Console indicar que você não está autorizado a realizar uma ação, entre em contato com o administrador para obter ajuda. Seu administrador é a pessoa que forneceu suas credenciais de login.

O erro de exemplo a seguir ocorre quando o usuário `mateojackson` do IAM tenta usar o console para visualizar detalhes sobre um `widget`, mas não tem as permissões `codestar:GetWidget`.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
codestar:GetWidget on resource: my-example-widget
```

Neste caso, Mateo pede ao administrador para atualizar suas políticas para permitir a ele o acesso ao recurso `my-example-widget` usando a ação `codestar:GetWidget`.

## Não estou autorizado a realizar iam: PassRole

Se você receber um erro informando que não está autorizado a realizar a `iam:PassRole` ação, suas políticas devem ser atualizadas para permitir que você passe uma função para a AWS CodeStar.

Alguns Serviços da AWS permitem que você passe uma função existente para esse serviço em vez de criar uma nova função de serviço ou uma função vinculada ao serviço. Para fazer isso, é preciso ter permissões para passar o perfil para o serviço.

O exemplo de erro a seguir ocorre quando um usuário do IAM chamado `marymajor` tenta usar o console para realizar uma ação na AWS CodeStar. No entanto, a ação exige que o serviço tenha permissões concedidas por um perfil de serviço. Mary não tem permissões para passar o perfil para o serviço.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

Nesse caso, as políticas de Mary devem ser atualizadas para permitir que ela realize a ação `iam:PassRole`.

Se precisar de ajuda, entre em contato com seu AWS administrador. Seu administrador é a pessoa que forneceu suas credenciais de login.

## Quero permitir que pessoas fora da minha AWS conta acessem meus CodeStar recursos da AWS

É possível criar um perfil que os usuários de outras contas ou pessoas fora da sua organização podem usar para acessar seus recursos. É possível especificar quem é confiável para assumir o perfil. Para serviços que oferecem suporte a políticas baseadas em recursos ou listas de controle de acesso (ACLs), você pode usar essas políticas para conceder às pessoas acesso aos seus recursos.

Para saber mais, consulte:

- Para saber se a AWS CodeStar oferece suporte a esses recursos, consulte [Como a AWS CodeStar trabalha com o IAM](#).
- Para saber como fornecer acesso aos seus recursos em todas as Contas da AWS que você possui, consulte Como [fornecer acesso a um usuário do IAM em outra Conta da AWS que você possui](#) no Guia do usuário do IAM.
- Para saber como fornecer acesso aos seus recursos a terceiros Contas da AWS, consulte Como [fornecer acesso Contas da AWS a terceiros](#) no Guia do usuário do IAM.
- Para saber como conceder acesso por meio da federação de identidades, consulte [Conceder acesso a usuários autenticados externamente \(federação de identidades\)](#) no Guia do usuário do IAM.
- Para conhecer a diferença entre perfis e políticas baseadas em recurso para acesso entre contas, consulte [Acesso a recursos entre contas no IAM](#) no Guia do usuário do IAM.

## Registrando chamadas de AWS CodeStar API com AWS CloudTrail

AWS CodeStar é integrado com AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, função ou AWS serviço em AWS CodeStar. CloudTrail captura todas as chamadas de API AWS CodeStar como eventos. As chamadas capturadas incluem chamadas do AWS CodeStar console e chamadas de código para operações de AWS CodeStar API. Se você criar uma trilha, poderá habilitar a entrega contínua de CloudTrail eventos em um bucket do S3, incluindo eventos para AWS CodeStar. Se você não configurar uma trilha, ainda poderá ver os eventos mais recentes no CloudTrail console no Histórico de eventos. Usando as informações coletadas por CloudTrail, você pode determinar a solicitação que foi feita AWS CodeStar, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita e outros detalhes.

Para saber mais sobre isso CloudTrail, consulte o [Guia AWS CloudTrail do usuário](#).

## AWS CodeStar Informações em CloudTrail

CloudTrail é ativado em sua AWS conta quando você cria a conta. Quando a atividade ocorre em AWS CodeStar, essa atividade é registrada em um CloudTrail evento junto com outros eventos AWS de serviço no histórico de eventos. Você pode visualizar, pesquisar e baixar eventos recentes em sua AWS conta. Para obter mais informações, consulte [Visualização de eventos com histórico de CloudTrail eventos](#).

Para um registro contínuo dos eventos em sua AWS conta, incluindo eventos para AWS CodeStar, crie uma trilha. Por padrão, quando você cria uma trilha no console, a trilha se aplica a todas as AWS regiões. A trilha registra eventos de todas as regiões na AWS partição e entrega os arquivos de log ao bucket do S3 que você especificar. Você pode configurar outros AWS serviços para analisar e agir com base nos dados de eventos coletados nos CloudTrail registros. Para obter mais informações, consulte:

- [Visão Geral para Criar uma Trilha](#)
- [CloudTrail Serviços e integrações compatíveis](#)
- [Configurando notificações do Amazon SNS para CloudTrail](#)
- [Recebendo arquivos de CloudTrail log de várias regiões](#) e [recebendo arquivos de CloudTrail log de várias contas](#)

Todas AWS CodeStar as ações são registradas CloudTrail e documentadas na [Referência da AWS CodeStar API](#). Por exemplo, chamadas para as AssociateTeamMember ações DescribeProjectUpdateProject, e geram entradas nos arquivos de CloudTrail log.

Cada entrada de log ou evento contém informações sobre quem gerou a solicitação. As informações de identidade ajudam a determinar:

- Se a solicitação foi feita com credenciais de usuário-raiz ou usuário do IAM.
- Se a solicitação foi feita com credenciais de segurança temporárias de uma função ou de um usuário federado.
- Se a solicitação foi feita por outro AWS serviço.

Para obter mais informações, consulte [Elemento userIdentity do CloudTrail](#).

## Entendendo as entradas do arquivo de AWS CodeStar log

CloudTrail os arquivos de log contêm uma ou mais entradas de log. Um evento representa uma única solicitação de qualquer fonte e inclui informações sobre a ação solicitada, a data e a hora da ação, os parâmetros da solicitação e assim por diante. CloudTrail os arquivos de log não são um rastreamento de pilha ordenado das chamadas públicas de API, portanto, eles não aparecem em nenhuma ordem específica.

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra uma CreateProject operação sendo chamada: AWS CodeStar

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROAJLIN20F3UBEXAMPLE:role-name",
    "arn": "arn:aws:sts::account-ID:assumed-role/role-name/role-session-name",
    "accountId": "account-ID",
    "accessKeyId": "ASIAJ44LFQS5XEXAMPLE",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2017-06-04T23:56:57Z"
      },
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROAJLIN20F3UBEXAMPLE",
        "arn": "arn:aws:iam::account-ID:role/service-role/role-name",
        "accountId": "account-ID",
        "userName": "role-name"
      }
    },
    "invokedBy": "codestar.amazonaws.com"
  },
  "eventTime": "2017-06-04T23:56:57Z",
  "eventSource": "codestar.amazonaws.com",
  "eventName": "CreateProject",
  "awsRegion": "region-ID",
  "sourceIPAddress": "codestar.amazonaws.com",
  "userAgent": "codestar.amazonaws.com",
  "requestParameters": {
```

```

    "clientRequestToken": "arn:aws:cloudformation:region-ID:account-ID:stack/stack-name/additional-ID",
    "id": "project-ID",
    "stackId": "arn:aws:cloudformation:region-ID:account-ID:stack/stack-name/additional-ID",
    "description": "AWS CodeStar created project",
    "name": "project-name",
    "projectTemplateId": "arn:aws:codestar:region-ID::project-template/project-template-name"
  },
  "responseElements": {
    "projectTemplateId": "arn:aws:codestar:region-ID::project-template/project-template-name",
    "arn": "arn:aws:codestar:us-east-1:account-ID:project/project-ID",
    "clientRequestToken": "arn:aws:cloudformation:region-ID:account-ID:stack/stack-name/additional-ID",
    "id": "project-ID"
  },
  "requestID": "7d7556d0-4981-11e7-a3bc-dd5daEXAMPLE",
  "eventID": "6b0d6e28-7a1e-4a73-981b-c8fdbEXAMPLE",
  "eventType": "AwsApiCall",
  "recipientAccountId": "account-ID"
}

```

## Validação de conformidade para AWS CodeStar

AWS CodeStar não está no escopo de nenhum programa de AWS conformidade.

Para obter uma lista de AWS serviços no escopo de programas de conformidade específicos, consulte [AWS Serviços no escopo por programa de conformidade](#). Para obter informações gerais, consulte [Programas de conformidade da AWS](#).

Você pode baixar relatórios de auditoria de terceiros usando AWS Artifact. Para obter mais informações, consulte [Baixando relatórios no AWS Artifact](#).

## Resiliência em AWS CodeStar

A infraestrutura AWS global é construída em torno de AWS regiões e zonas de disponibilidade. AWS As regiões fornecem várias zonas de disponibilidade fisicamente separadas e isoladas, conectadas a redes de baixa latência, alta taxa de transferência e alta redundância. Com as zonas de disponibilidade, é possível projetar e operar aplicações e bancos de dados que executam

o failover automaticamente entre as zonas de disponibilidade sem interrupção. As zonas de disponibilidade são mais altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de data center tradicionais.

Para obter mais informações sobre AWS regiões e zonas de disponibilidade, consulte [Infraestrutura AWS global](#).

## Segurança de infraestrutura em AWS CodeStar

Como um serviço gerenciado, a AWS CodeStar é protegida pela segurança de rede AWS global. Para obter informações sobre serviços AWS de segurança e como AWS proteger a infraestrutura, consulte [AWS Cloud Security](#). Para projetar seu AWS ambiente usando as melhores práticas de segurança de infraestrutura, consulte [Proteção](#) de infraestrutura no Security Pillar AWS Well-Architected Framework.

Você usa chamadas de API AWS publicadas para acessar CodeStar pela rede. Os clientes devem oferecer compatibilidade com:

- Transport Layer Security (TLS). Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Conjuntos de criptografia com perfect forward secrecy (PFS) como DHE (Ephemeral Diffie-Hellman) ou ECDHE (Ephemeral Elliptic Curve Diffie-Hellman). A maioria dos sistemas modernos, como Java 7 e versões posteriores, comporta esses modos.

Além disso, as solicitações devem ser assinadas usando um ID da chave de acesso e uma chave de acesso secreta associada a uma entidade principal do IAM. Ou é possível usar o [AWS Security Token Service](#) (AWS STS) para gerar credenciais de segurança temporárias para assinar solicitações.

Por padrão, AWS CodeStar não isola o tráfego do serviço. Os projetos criados usando AWS CodeStar estão abertos à Internet pública, a menos que você modifique manualmente as configurações de acesso por meio da Amazon EC2, do API Gateway ou do Elastic Beanstalk. Isso é intencional. Você pode modificar as configurações de acesso na Amazon EC2, no API Gateway ou no Elastic Beanstalk na medida que desejar, inclusive impedindo todo o acesso à Internet.

AWS CodeStar não fornece suporte para VPC endpoints (AWS PrivateLink) por padrão, mas você pode configurar esse suporte diretamente nos recursos do projeto.

# Limites em AWS CodeStar

A tabela a seguir descreve os limites em AWS CodeStar. AWS CodeStar depende de outros AWS serviços para os recursos do projeto. Alguns desses limites de serviço podem ser alterados. Para obter informações sobre os limites que podem ser alterados, consulte [Limites de serviço da AWS](#).

Número de projetos	Máximo de 333 projetos em uma AWS conta. O limite real varia, dependendo do nível de outras dependências de serviço (por exemplo, o número máximo de pipelines CodePipeline permitido para sua AWS conta).
Número de AWS CodeStar projetos aos quais um usuário do IAM pode pertencer	Máximo de 10 por usuário do IAM individual.
Projeto IDs	<p>O projeto IDs deve ser exclusivo em uma AWS conta. O projeto IDs deve ter pelo menos 2 caracteres e não pode exceder 15 caracteres. Os caracteres permitidos incluem:</p> <p>Letras a a z inclusive.</p> <p>Números 0 a 9 inclusive.</p> <p>O caractere especial - (sinal de menos).</p> <p>Todos os outros caracteres, como letras maiúsculas, espaços . (ponto), @ (arroba) ou _ (sublinhado), não são permitidos.</p>
Nomes de projetos	Os nomes de projetos não podem ter mais que 100 caracteres de comprimento e não podem começar ou terminar com um espaço vazio.
Descrições de projetos	Qualquer combinação de caracteres entre 0 e 1.024 caracteres. As descrições de projetos são opcionais.

Membros da equipe em um AWS CodeStar projeto	100
Nome de exibição em um perfil de usuário	Qualquer combinação de caracteres entre 1 e 100 caracteres. Os nomes de exibição devem incluir pelo menos um caractere. Esse caractere não pode ser um espaço. Os nomes de exibição não podem começar ou terminar com um espaço.
Endereço de e-mail em um perfil de usuário	O endereço de e-mail deve incluir uma @ e terminar em uma extensão de domínio válida.
Acesso federado, acesso à conta root ou acesso temporário ao AWS CodeStar	AWS CodeStar oferece suporte a usuários federados e ao uso de credenciais de acesso temporário. Não é recomendável usar AWS CodeStar com uma conta root.
Perfis do IAM	Um máximo de 5.120 caracteres em qualquer política gerenciada que seja anexada a um perfil do IAM.

# Solução de problemas AWS CodeStar

As informações a seguir podem ajudar a solucionar problemas comuns no AWS CodeStar.

## Tópicos

- [Falha ao criar o projeto: um projeto não foi criado](#)
- [Criação do projeto: vejo um erro quando tento editar a EC2 configuração da Amazon ao criar um projeto](#)
- [Exclusão do projeto: um AWS CodeStar projeto foi excluído, mas os recursos ainda existem](#)
- [Falha no gerenciamento da equipe: um usuário do IAM não pôde ser adicionado a uma equipe em um AWS CodeStar projeto](#)
- [Falha de acesso: um usuário federado não pode acessar um projeto AWS CodeStar](#)
- [Falha de acesso: um usuário federado não pode acessar ou criar um ambiente AWS Cloud9](#)
- [Falha de acesso: um usuário federado pode criar um AWS CodeStar projeto, mas não pode visualizar os recursos do projeto](#)
- [Problema de função de serviço: a função de serviço não pôde ser criada](#)
- [Problema de função de serviço: a função de serviço não é válida ou está ausente](#)
- [Problema de função do projeto: as verificações do status de AWS Elastic Beanstalk saúde falham em instâncias em um AWS CodeStar projeto](#)
- [Problema de função de projeto: uma função de projeto não é válida ou está ausente](#)
- [Extensões de projetos: não é possível se conectar ao JIRA](#)
- [GitHub: Não consigo acessar o histórico de commits, problemas ou código de um repositório](#)
- [AWS CloudFormation: criação da pilha revertida para permissões ausentes](#)
- [AWS CloudFormation não está autorizado a realizar iam: PassRole na função de execução do Lambda](#)
- [Não é possível criar a conexão para um GitHub repositório](#)

## Falha ao criar o projeto: um projeto não foi criado

**Problema:** quando você tenta criar um projeto, você vê uma mensagem informando que houve falha na criação.

**Correções possíveis:** os motivos mais comuns para a falha são:

- Já existe um projeto com esse ID na sua AWS conta, possivelmente em uma AWS região diferente.
- O usuário do IAM que você usou para fazer login AWS Management Console não tem as permissões necessárias para criar um projeto.
- A função AWS CodeStar de serviço não tem uma ou mais permissões necessárias.
- Você atingiu o limite máximo de um ou mais recursos para um projeto (como o limite de políticas gerenciadas pelo cliente no IAM, buckets do Amazon S3 ou pipelines em). CodePipeline

Antes de criar um projeto, verifique se você tem a política `AWSCodeStarFullAccess` aplicada para o usuário do IAM. Para obter mais informações, consulte [AWSCodeStarFullAccess Política](#).

Ao criar um projeto, verifique se o ID é exclusivo e atende aos requisitos do AWS CodeStar . Certifique-se de marcar a caixa de seleção AWS CodeStar Gostaria de ter permissão para administrar AWS recursos em seu nome.

Para solucionar outros problemas, abra o AWS CloudFormation console, escolha a pilha do projeto que você tentou criar e escolha a guia Eventos. Pode haver mais de uma pilha para um projeto. Os nomes das pilhas começam com `awscodestar-` e são seguidos pelo ID do projeto. As pilhas podem estar sob a visualização de filtro Excluídas. Analise as mensagens de falha nos eventos da pilha e corrija o problema listado como a causa dessas falhas.

## Criação do projeto: vejo um erro quando tento editar a EC2 configuração da Amazon ao criar um projeto

**Problema:** Ao editar as opções de EC2 configuração da Amazon durante a criação do projeto, você vê uma mensagem de erro ou uma opção acinzentada e não pode continuar com a criação do projeto.

**Correções possíveis:** os motivos mais comuns para uma mensagem de erro são:

- A VPC no modelo do AWS CodeStar projeto (a VPC padrão ou a usada quando a EC2 configuração da Amazon foi editada) tem localização de instância dedicada, e o tipo de instância não é compatível com instâncias dedicadas. Escolha outro tipo de instância ou outro Amazon VPC.
- Sua AWS conta não tem Amazon VPCs. Você pode ter excluído a VPC padrão e não criou nenhuma outra. Abra o console da Amazon VPC em <https://console.aws.amazon.com/vpc/>, escolha Seu VPCs e verifique se você tem pelo menos uma VPC configurada. Se não tiver, crie

uma. Para obter mais informações, consulte [Visão geral do Amazon Virtual Private Cloud](#) no Guia de introdução do Amazon VPC.

- O Amazon VPC não tem nenhuma sub-rede. Escolha outra VPC ou crie uma sub-rede para a VPC. Para obter mais informações, consulte [Noções básicas da VPC e da sub-rede](#).

## Exclusão do projeto: um AWS CodeStar projeto foi excluído, mas os recursos ainda existem

Problema: um AWS CodeStar projeto foi excluído, mas os recursos criados para esse projeto ainda existem. Por padrão, AWS CodeStar exclui os recursos do projeto quando o projeto é excluído. Alguns recursos, como os buckets do Amazon S3, são mantidos, mesmo que o usuário selecione a caixa de seleção Excluir recursos, porque os buckets podem conter dados.

Possíveis correções: abra o [AWS CloudFormation console](#) e encontre uma ou mais AWS CloudFormation pilhas usadas para criar o projeto. Os nomes das pilhas começam com `awscodestar-` e são seguidos pelo ID do projeto. As pilhas podem estar sob a visualização de filtro Excluídas. Analise os eventos associados à pilha para descobrir os recursos criados para o projeto. Abra o console de cada um desses recursos na AWS região em que você criou o AWS CodeStar projeto e exclua manualmente os recursos.

Os recursos do projeto que podem permanecer incluem:

- Um ou mais buckets do projeto no Amazon S3. Ao contrário de outros recursos do projeto, os buckets do projeto no Amazon S3 não são excluídos quando a caixa de seleção Excluir recursos AWS associados junto AWS CodeStar com o projeto é marcada.

Abra o console do Amazon S3 em <https://console.aws.amazon.com/s3/>.

- Um repositório de origem para seu projeto em CodeCommit.

Abra o CodeCommit console em <https://console.aws.amazon.com/codecommit/>.

- Um pipeline para seu projeto em CodePipeline.

Abra o CodePipeline console em <https://console.aws.amazon.com/codepipeline/>.

- Um aplicativo e grupos de implantação associados no CodeDeploy.

Abra o CodeDeploy console em <https://console.aws.amazon.com/codedeploy/>.

- Um aplicativo e ambientes associados no AWS Elastic Beanstalk.

Abra o console do Elastic Beanstalk em <https://console.aws.amazon.com/elasticbeanstalk/>

- Uma função no AWS Lambda.

Abra o AWS Lambda console em <https://console.aws.amazon.com/lambda/>.

- Um ou mais APIs no API Gateway.

Abra o console do API Gateway em <https://console.aws.amazon.com/apigateway/>.

- Uma ou mais políticas ou perfis do IAM no IAM.

Faça login no AWS Management Console e abra o console do IAM em <https://console.aws.amazon.com/iam/>.

- Uma instância na Amazon EC2.

Abra o EC2 console da Amazon em <https://console.aws.amazon.com/ec2/>.

- Um ou mais ambientes de desenvolvimento em AWS Cloud9.

Para visualizar, acessar e gerenciar ambientes de desenvolvimento, abra o AWS Cloud9 console em <https://console.aws.amazon.com/cloud9/>.

Se seu projeto usa recursos externos AWS (por exemplo, um GitHub repositório ou problemas no Atlassian JIRA), esses recursos não são excluídos, mesmo que a caixa Excluir AWS recursos associados junto com o CodeStar projeto esteja selecionada.

## Falha no gerenciamento da equipe: um usuário do IAM não pôde ser adicionado a uma equipe em um AWS CodeStar projeto

Problema: quando você tenta adicionar um usuário a um projeto, você vê uma mensagem de erro informando que houve falha na adição.

Correções possíveis: o motivo mais comum para esse erro é que o usuário do atingiu o limite de políticas gerenciadas que podem ser aplicadas a um usuário no IAM. Você também pode receber esse erro se não tiver o papel de proprietário no AWS CodeStar projeto em que tentou adicionar o usuário ou se o usuário do IAM não existir ou tiver sido excluído.

Verifique se você está conectado como um usuário proprietário desse AWS CodeStar projeto. Para obter mais informações, consulte [Adicionar membros da equipe a um AWS CodeStar projeto](#) .

Para solucionar outros problemas, abra o console do IAM, escolha o usuário que você tentou adicionar e verifique quantas políticas gerenciadas estão aplicadas ao usuário do IAM.

Para obter mais informações, consulte [Limitações em entidades e objetos do IAM](#). Para obter os limites que podem ser alterados, consulte [Service Limits da AWS](#).

## Falha de acesso: um usuário federado não pode acessar um projeto AWS CodeStar

Problema: um usuário federado não consegue ver projetos no AWS CodeStar console.

Correções possíveis: se você estiver conectado como um usuário federado, verifique se tem a política gerenciada apropriada anexada à função que você assumir para fazer login. Para obter mais informações, consulte [Anexe a política AWS CodeStar Viewer/Contributor/Owner gerenciada do seu projeto à função do usuário federado](#).

Adicione usuários federados ao seu AWS Cloud9 ambiente anexando políticas manualmente. Consulte [Anexar uma política AWS Cloud9 gerenciada à função do usuário federado](#).

## Falha de acesso: um usuário federado não pode acessar ou criar um ambiente AWS Cloud9

Problema: um usuário federado não consegue ver ou criar um AWS Cloud9 ambiente no AWS Cloud9 console.

Correções possíveis: se você estiver conectado como um usuário federado, verifique se tem a política gerenciada apropriada anexada à função do usuário federado.

Você adiciona usuários federados ao seu AWS Cloud9 ambiente anexando manualmente as políticas à função do usuário federado. Consulte [Anexar uma política AWS Cloud9 gerenciada à função do usuário federado](#).

## Falha de acesso: um usuário federado pode criar um AWS CodeStar projeto, mas não pode visualizar os recursos do projeto

Problema: um usuário federado foi capaz de criar um projeto, mas não pode visualizar recursos do projeto, como o pipeline do projeto.

Possíveis correções: Se você anexou a política **AWSCodeStarFullAccess** gerenciada, você tem permissões para criar um projeto em AWS CodeStar. No entanto, para acessar todos os recursos do projeto, você deve anexar a política gerenciada do proprietário.

Depois de AWS CodeStar criar os recursos do projeto, as permissões do projeto para todos os recursos do projeto ficam disponíveis nas políticas gerenciadas pelo proprietário, colaborador e visualizador. Para acessar todos os recursos, você deve anexar manualmente a política de proprietário à função. Consulte [Etapa 3: configurar as permissões do IAM do usuário](#).

## Problema de função de serviço: a função de serviço não pôde ser criada

Problema: Ao tentar criar um projeto no AWS CodeStar, você vê uma mensagem solicitando que você crie a função de serviço. Quando você escolhe a opção para criá-la, você vê um erro.

Possíveis correções: o motivo mais comum para esse erro é que você está conectado AWS com uma conta que não tem permissões suficientes para criar a função de serviço. Para criar a função de AWS CodeStar serviço (`aws-codestar-service-role`), você deve estar conectado como usuário administrativo ou com uma conta raiz. Faça logout do console e faça login com um usuário do IAM que tenha a política gerenciada `AdministratorAccess` aplicada.

## Problema de função de serviço: a função de serviço não é válida ou está ausente

Problema: Ao abrir o AWS CodeStar console, você vê uma mensagem informando que a função AWS CodeStar de serviço está ausente ou não é válida.

Correções possíveis: o motivo mais comum para esse erro é que um usuário administrativo editou ou excluiu a função de serviço (`aws-codestar-service-role`). Se a função de serviço foi excluída, você é solicitado a criá-la. Você deve estar conectado como um usuário administrativo ou com uma conta root para criar a função. Se tiver sido editada, a função não será mais válida. Faça login no console do IAM como um usuário administrativo, localize o perfil de serviço na lista de funções e exclua-a. Mude para o AWS CodeStar console e siga as instruções para criar a função de serviço.

# Problema de função do projeto: as verificações do status de AWS Elastic Beanstalk saúde falham em instâncias em um AWS CodeStar projeto

**Problema:** Se você criou um AWS CodeStar projeto que inclui o Elastic Beanstalk antes de 22 de setembro de 2017, as verificações do status de saúde do Elastic Beanstalk podem falhar. Se você não tiver alterado a configuração do Elastic Beanstalk desde que criou o projeto, a verificação de status de integridade falhará e informará um estado cinza. Apesar da falha na verificação de integridade, o aplicativo ainda deve ser executado conforme o esperado. Se você não tiver alterado a configuração do Elastic Beanstalk desde que criou o projeto, a verificação de status de integridade falhará e o aplicativo não executará corretamente.

**Correção:** um ou mais perfis do IAM estão ausentes nas declarações de política do IAM obrigatórias. Adicione as políticas para as funções afetadas ausentes na sua conta da AWS .

1. Faça login no AWS Management Console e abra o console do IAM em <https://console.aws.amazon.com/iam/>.

(Se você não conseguir fazer isso, consulte o administrador AWS da conta para obter ajuda.)

2. No painel de navegação, selecione Perfis.
3. Na lista de funções, escolha CodeStarWorker- **Project-ID** -EB, onde **Project-ID** está o ID de um dos projetos afetados. (Se você não conseguir encontrar facilmente uma função na lista, insira todo ou parte do nome da função na caixa Search (Pesquisa).)
4. Na guia Permissions, escolha Attach Policy.
5. Na lista de políticas, selecione AWSElasticBeanstalkEnhancedHealthAWSElasticBeanstalkService. (Se você não conseguir encontrar facilmente uma política na lista, insira todo ou parte do nome da política na caixa de Pesquisa.)
6. Escolha Attach Policy.
7. Repita as etapas de 3 a 6 para cada função afetada que tenha um nome seguindo o padrão CodeStarWorker- **Project-ID** -EB.

## Problema de função de projeto: uma função de projeto não é válida ou está ausente

**Problema:** quando você tenta adicionar um usuário a um projeto, você vê uma mensagem de erro informando que houve falha na adição porque a política de uma função de projeto está ausente ou não é válida.

**Correções possíveis:** o motivo mais comum para esse erro é que uma ou mais políticas de projeto foram editadas ou excluídas do IAM. As políticas do projeto são exclusivas AWS CodeStar dos projetos e não podem ser recriadas. O projeto não pode ser usado. Crie um projeto em e AWS CodeStar, em seguida, migre os dados para o novo projeto. Clone o código do projeto no repositório do projeto não utilizável e envie esse código por push para o novo repositório do projeto. Copie as informações de wiki da equipe do projeto antigo para o novo projeto. Adicione usuários ao novo projeto. Quando tiver certeza de que migrou todos os dados e configurações, exclua o projeto inutilizável.

## Extensões de projetos: não é possível se conectar ao JIRA

**Problema:** Quando você usa a extensão Atlassian JIRA para tentar conectar um AWS CodeStar projeto a uma instância do JIRA, a seguinte mensagem é exibida: "A URL não é uma URL válida do JIRA. Verificar se a URL está correta."

**Correções possíveis:**

- Verifique se a URL do JIRA está correta e, em seguida, tente a conexão novamente.
- Sua instância auto-hospedada do JIRA pode não estar acessível na internet pública. Entre em contato com o administrador da rede para garantir que sua instância do JIRA possa ser acessada na internet pública e, em seguida, tente a conexão novamente.

## GitHub: Não consigo acessar o histórico de commits, problemas ou código de um repositório

**Problema:** No painel de um projeto que armazena seu código GitHub, os blocos Histórico de confirmações e GitHubProblemas exibem um erro de conexão, ou escolher Abrir em GitHub ou Criar problema nesses blocos exibe um erro.

**Causas possíveis:**

- Talvez o AWS CodeStar projeto não tenha mais acesso ao GitHub repositório.
- O repositório pode ter sido excluído ou renomeado em GitHub

## AWS CloudFormation: criação da pilha revertida para permissões ausentes

Depois de adicionar um recurso ao arquivo `template.yml`, visualize a atualização da pilha AWS CloudFormation para as mensagens de erro. A atualização da pilha falhará se determinados critérios não forem atendidos (por exemplo, quando as permissões de recursos necessários estão ausentes).

### Note

Em 2 de maio de 2019, atualizamos a política de funções do AWS CloudFormation trabalhador para todos os projetos existentes. Essa atualização reduz o escopo das permissões de acesso concedidas ao seu pipeline do projeto para maior segurança em seus projetos.

Para solucionar problemas, visualize o status da falha na visualização do AWS CodeStar painel do pipeline do seu projeto.

Em seguida, escolha o CloudFormationlink no estágio de implantação do seu pipeline para solucionar a falha no AWS CloudFormation console. Para visualizar os detalhes da criação da pilha, expanda a lista Events (Eventos) do projeto e visualize as mensagens de falha. A mensagem indica que a permissão está ausente. Corrija a política de função de operador do AWS CloudFormation e execute o pipeline novamente.

## AWS CloudFormation não está autorizado a realizar iam: PassRole na função de execução do Lambda

Se você tiver um projeto criado antes do PDT de 6 de dezembro de 2018 que cria funções Lambda, talvez veja AWS CloudFormation um erro como este:

```
User: arn:aws:sts::id:assumed-role/CodeStarWorker-project-id-CloudFormation/  
AWSCloudFormation is not authorized to perform: iam:PassRole on resource:
```

```
arn:aws:iam::id:role/CodeStarWorker-project-id-Lambda (Service: AWSLambdaInternal;  
Status Code: 403; Error Code: AccessDeniedException; Request ID: id)
```

Esse erro ocorre porque sua função de AWS CloudFormation trabalhador não tem permissão para passar uma função para provisionar sua nova função Lambda.

Para corrigir esse erro, você precisará atualizar sua política de função de AWS CloudFormation trabalhador com o seguinte trecho.

```
{  
    "Action": [ "iam:PassRole" ],  
    "Resource": [  
        "arn:aws:iam::account-id:role/CodeStarWorker-project-id-Lambda",  
    ],  
    "Effect": "Allow"  
}
```

Depois de atualizar a política, execute o pipeline novamente.

Como alternativa, é possível usar uma função personalizada para sua função do Lambda adicionando um limite de permissões ao seu projeto, conforme descrito em [Adicionar um limite de permissões do IAM a projetos existentes](#)

## Não é possível criar a conexão para um GitHub repositório

Problema:

Como uma conexão com um GitHub repositório usa o AWS Conector para GitHub, você precisa de permissões de proprietário da organização ou permissões de administrador no repositório para criar a conexão.

Possíveis correções: Para obter informações sobre os níveis de permissão de um GitHub repositório, consulte <https://docs.github.com/en/free-pro-team@latest/github/setting-up-and-managing-organizations-and-teams/permission-levels-for-an-organization>.

# AWS CodeStar Notas de versão do guia do usuário

A tabela a seguir descreve as mudanças importantes em cada versão do Guia AWS CodeStar do usuário. Para receber notificações sobre atualizações dessa documentação, é possível inscrever-se em um feed RSS.

Alteração	Descrição	Data
<a href="#">Atualizações de política de acesso</a>	A política de função de AWS CodeStar acesso foi atualizada. O resultado da política é o mesmo, mas o CloudFormation exige um ListStacks adicional ao DescribeStacks que já é necessário. Para consultar a política atualizada, consulte a <a href="#">AWSCodeStarFullAccess Política</a> .	24 de março de 2023
<a href="#">Atualizações da política de perfil de serviço</a>	A política AWS CodeStar de função de serviço foi atualizada. Para consultar a política atualizada, consulte a <a href="#">AWSCodeStarServiceRole Política</a> .	23 de setembro de 2021
<a href="#">Use um recurso de conexão para projetos com um repositório GitHub de origem</a>	Quando você usa o console para criar um projeto AWS CodeStar com um GitHub repositório, um recurso de conexão é usado para gerenciar suas GitHub ações. As conexões usam GitHub aplicativos, enquanto a GitHub autorização anterior é usada OAuth. Para ver um tutorial que mostra como criar um	27 de abril de 2021

projeto que usa uma conexão com GitHub, consulte [Tutorial: Criar um projeto com um repositório de GitHub origem](#). O tutorial também mostra como criar, revisar e mesclar uma pull request para o repositório de origem do seu projeto.

[AWS CodeStar suporta AWS Cloud9 na região Oeste dos EUA \(Norte da Califórnia\)](#)

AWS CodeStar agora suporta o uso AWS Cloud9 na região Oeste dos EUA (Norte da Califórnia). Para obter mais informações, consulte [Configurando o Cloud9](#).

16 de fevereiro de 2021

[Atualize a documentação para refletir a nova experiência do console](#)

Em 12 de agosto de 2020, o AWS CodeStar serviço mudou para uma nova experiência de usuário no AWS console. O guia do usuário foi atualizado para corresponder à nova experiência do console.

12 de agosto de 2020

[AWS CodeStar projetos podem ser criados com a AWS CodeStar CLI](#)

AWS CodeStar projetos podem ser criados com o comando CLI. AWS CodeStar cria seu projeto e sua infraestrutura usando o código-fonte e um modelo de conjunto de ferramentas fornecido por você. Consulte [Criar um projeto em AWS CodeStar \(AWS CLI\)](#).

24 de outubro de 2018

[Todos os modelos de AWS CodeStar projeto agora incluem AWS CloudFormation arquivos para atualizações de infraestrutura](#)

AWS CodeStar trabalha com AWS CloudFormation para permitir que você use código para criar serviços de suporte e servidores ou plataformas sem servidor na nuvem. O AWS CloudFormation arquivo agora está disponível para todos os tipos de modelos de AWS CodeStar projetos (modelos com a plataforma de computação Lambda ou Elastic Beanstalk). EC2 O arquivo é armazenado em `template.yml` no repositório de origem. Você pode visualizar e modificar o arquivo para adicionar recursos ao projeto. Consulte [Modelos de projetos](#).

3 de agosto de 2018

[AWS CodeStar Notificações de atualização do Guia do Usuário agora disponíveis por meio de RSS](#)

A versão HTML do Guia do AWS CodeStar Usuário agora oferece suporte a um feed RSS de atualizações que estão documentadas na página Notas de versão da atualização da documentação. O feed RSS inclui as atualizações feitas após 30 de junho de 2018. As atualizações anunciadas anteriormente ainda estão disponíveis na página Notas de release de atualizações da documentação. Use o botão RSS no menu superior do painel para assinar o feed.

30 de junho de 2018

A tabela a seguir descreve as mudanças importantes em cada versão do Guia do AWS CodeStar usuário antes de 30 de junho de 2018.

Alteração	Descrição	Alterado em
O Guia AWS CodeStar do Usuário agora está disponível em GitHub	Este guia já está disponível em GitHub. Você também pode usar GitHub para enviar feedback e solicitações de alteração do conteúdo deste guia. Para obter mais informações, escolha o GitHub ícone Editar na barra de navegação do guia ou consulte o aws-codestar-user-guide repositório <a href="#">awsdocs/</a> no site. GitHub	22 de fevereiro de 2018
AWS CodeStar agora está disponível na Ásia-Pacífico (Seul)	AWS CodeStar agora está disponível na região Ásia-Pacífico (Seul). Para ter mais informações, consulte <a href="#">AWS CodeStar</a> no Referência geral da Amazon Web Services.	14 de fevereiro de 2018

Alteração	Descrição	Alterado em
AWS CodeStar agora está disponível na Ásia-Pacífico (Tóquio) e Canadá (Central)	AWS CodeStar agora está disponível nas regiões Ásia-Pacífico (Tóquio) e Canadá (Central). Para ter mais informações, consulte <a href="#">AWS CodeStar</a> no Referência geral da Amazon Web Services.	20 de dezembro de 2017
AWS CodeStar agora suporta AWS Cloud9	AWS CodeStar agora suporta o uso AWS Cloud9 de um IDE on-line baseado em navegador da Web para trabalhar com o código do projeto. Para obter mais informações, consulte <a href="#">Use AWS Cloud9 com AWS CodeStar</a> .  Para obter uma lista das AWS regiões suportadas, consulte <a href="#">AWS Cloud9</a> no Referência geral da Amazon Web Services.	30 de novembro de 2017
AWS CodeStar agora suporta GitHub	AWS CodeStar agora suporta o armazenamento do código do projeto em GitHub. Para obter mais informações, acesse <a href="#">Criar um projeto</a> .	12 de outubro de 2017
AWS CodeStar agora disponível no Oeste dos EUA (Norte da Califórnia) e na Europa (Londres)	AWS CodeStar agora está disponível nas regiões Oeste dos EUA (Norte da Califórnia) e Europa (Londres). Para ter mais informações, consulte <a href="#">AWS CodeStar</a> no Referência geral da Amazon Web Services.	17 de agosto de 2017
AWS CodeStar agora disponível na Ásia-Pacífico (Sydney), Ásia-Pacífico (Cingapura) e Europa (Frankfurt)	AWS CodeStar agora está disponível nas regiões Ásia-Pacífico (Sydney), Ásia-Pacífico (Cingapura) e Europa (Frankfurt). Para ter mais informações, consulte <a href="#">AWS CodeStar</a> no Referência geral da Amazon Web Services.	25 de julho de 2017

Alteração	Descrição	Alterado em
AWS CloudTrail agora suporta AWS CodeStar	AWS CodeStar agora está integrado com CloudTrail, um serviço que captura chamadas de API feitas por ou em nome de AWS CodeStar sua AWS conta e entrega os arquivos de log para um bucket do Amazon S3 que você especificar. Para obter mais informações, consulte <a href="#">Registrando chamadas de AWS CodeStar API com AWS CloudTrail</a> .	14 de junho de 2017
Lançamento inicial	Esta é a primeira versão do Guia do usuário do AWS CodeStar .	19 de abril de 2017

# AWS Glossário

Para obter a AWS terminologia mais recente, consulte o [AWS glossário](#) na Glossário da AWS Referência.