



AWS 클라우드에서 애플리케이션을 현대화하기 위한 단계별 접근 방식

AWS 권장 가이드



AWS 권장 가이드: AWS 클라우드에서 애플리케이션을 현대화하기 위한 단계별 접근 방식

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon의 상표 및 트레이드 드레스는 Amazon 외 제품 또는 서비스와 함께, Amazon 브랜드 이미지를 떨어뜨리거나 고객에게 혼동을 일으킬 수 있는 방식으로 사용할 수 없습니다. Amazon이 소유하지 않은 기타 모든 상표는 Amazon과 제휴 관계이거나 관련이 있거나 후원 관계와 관계없이 해당 소유자의 자산입니다.

Table of Contents

소개	1
목표 비즈니스 성과	2
현대화 프로세스	3
1단계. 애플리케이션 평가	3
현대화 진단 플레이북 사용합니다	4
지표 식별	5
2단계. 작게 시작해서 추진력을 키우세요	6
우선 순위 드라이버 검증	7
세부 사항 마무리하세요	7
기본 플랫폼 서비스 구축 및 애플리케이션 현대화합니다	8
3단계. 확장 가능한 현대화 로드맵을 만드세요	8
현대화 준비 요인	10
코드	10
빌드 및 테스트	10
릴리스	11
운영	11
최적화	11
준비 상태	12
다음 단계	13
리소스	14
관련 가이드	14
AWS 리소스	14
문서 기록	15
.....	xvi

AWS 클라우드에서 애플리케이션을 현대화하기 위한 단계별 접근 방식

Vijay Thumma와 Ashish Ameta, Amazon Web Services(AWS)

2023년 5월([문서 기록](#))

현대화를 위해서는 새로운 기술을 채택 및 소비하고, 포트폴리오, 애플리케이션 및 인프라 가치를 더 빠르게 제공하고, 최적의 가격으로 규모를 조정할 수 있는 조직을 포지셔닝하기 위한 다차원적 접근 방식이 필요합니다. 여기에는 현대화된 모델에서 중단 없이 애플리케이션을 최적화하고 유지 관리하고 운영하는 것이 포함되며, 이를 위해서는 비즈니스 운영, 아키텍처 및 전반적인 엔지니어링 관행을 단순화해야 합니다.

현대화는 애플리케이션에만 국한되는 것이 아닙니다. 이를 위해서는 안전하고 유연한 운영 프레임워크를 제공하는 최신 인프라가 필요합니다. 비즈니스 프로세스 품질, 가용성 및 민첩성 측면에서 애플리케이션과 인프라는 분리할 수 없습니다. 인프라를 고려하지 않고 애플리케이션을 현대화하면 전체 비용이 증가하고 성능과 품질에 부정적인 영향을 미칩니다. 최신 애플리케이션은 새로운 아키텍처 패턴, 운영 모델 및 소프트웨어 제공 프로세스의 조합으로 구축됩니다. 사용자를 0명에서 수백만 명으로 확장 및 축소하고, 테라바이트(페타바이트는 아니더라도)의 데이터를 관리하고, 전 세계에서 사용할 수 있으며, 밀리초 내에 응답합니다. Amazon Web Services(AWS) Cloud에서 관리하는 워크로드 포트폴리오를 현대화하면 컨테이너, 서버리스 기술, 전용 데이터 스토어 및 소프트웨어 자동화를 사용하여 이러한 워크로드를 리플랫폼, 리팩터링 또는 교체하여 AWS에서 제공하는 최대한의 민첩성과 총비용 최적화(TCO) 혜택을 누릴 수 있습니다.

이 가이드는 애플리케이션 소유자, 비즈니스 소유자, 설계자, 기술 책임자 및 프로젝트 매니저를 위한 것입니다. 현대화 평가 단계에서 선택한 애플리케이션의 기본 기능을 개발하는 방법과 단계적 접근 방식을 사용하여 현대화 노력을 가속화하는 방법에 대해 설명합니다.

이 가이드는에서 권장하는 애플리케이션 현대화 접근 방식을 다루는 콘텐츠 시리즈의 일부입니다 AWS. 이 시리즈에는 다음 내용도 포함됩니다.

- [AWS 클라우드에서 애플리케이션을 현대화하기 위한 전략](#)
- [AWS 클라우드의 애플리케이션에 대한 현대화 준비 상태 평가](#)
- [마이크로서비스로 모놀리식 유형 분해](#)
- [AWS 서버리스 서비스를 사용하여 마이크로서비스 통합](#)

목표 비즈니스 성과

애플리케이션 현대화에 대한 단계별 접근 방식을 통해 다음과 같은 결과를 기대할 수 있습니다.

- 구축 및 검증 접근 방식과 마이크로서비스와 같은 클라우드 네이티브 아키텍처를 사용하여 조직의 역량과 능력을 강화하여 더 빠르게 혁신할 수 있습니다.
- 교육 및 도구 개선을 통해 조직의 준비 상태를 구축하는 변경 관리 및 운영 모델입니다.
- 12주 이내에 초기 결과를 도출하는 데 도움이 되는 팀 접근 방식은 경험적 학습을 제공하고 독립적이고 지속적인 고객 성공을 가능하게 합니다.
- 마이크로서비스, API, 재사용 가능한 구성 요소 및 컨테이너화를 기반으로 하는 구성 가능한 애플리케이션 아키텍처.
- 일부 전략적 애플리케이션을 위한 확장 가능한 현대화 로드맵으로, 여기에는 스플릿 앤 시드 모델에서 작동하는 권장 가이드가 포함되어 있습니다. 이 모델에서는 현대화 기능과 서비스가 비즈니스 성과에 초점을 맞춘 여러 엔지니어링 팀에 걸쳐 규모가 조정됩니다. 새로운 최소 기능 제품(MVP)이 정의되면 초기 팀원들이 흩어져 새로운 제품 팀을 구성합니다.

현대화 프로세스

현대화에 대한 단계별 접근 방식의 목표는 현대화 차원을 사용하여 가치를 높이고, 이를 비즈니스를 차별화하는 핵심 애플리케이션의 일부에 적용하고, 최신 기술 채택을 가속화하는 것입니다. 단계별 접근 방식은 다음 세 단계로 구성됩니다.

1. 현대화 진단 플레이북을 사용하여 애플리케이션의 성숙도를 평가하세요. 포괄적인 접근 방식을 채택하고 의도한 비즈니스 성과에 맞는 프로세스를 개발하세요.
2. 소규모로 시작하여 MVP를 제공할 준비를 하고 비즈니스 성과를 조기에 점진적으로 추진하여 추진력을 확보하세요. 이 단계는 일반적으로 12~16주가 소요됩니다.
3. 스플릿 앤 시드 모델에서 사용할 수 있는 확장 가능한 현대화 로드맵을 만드세요.

다음 섹션에서는 이러한 단계에 대해 자세히 설명합니다.

주제

- [1단계. 애플리케이션 평가](#)
- [2단계. 작게 시작해서 추진력을 키우세요](#)
- [3단계. 확장 가능한 현대화 로드맵을 만드세요](#)

1단계. 애플리케이션 평가

이 단계의 목표는 다음과 같습니다.

- 애플리케이션 환경을 철저히 이해하고 최신 데이터 플랫폼에 맞게 애플리케이션을 준비하여 비즈니스에 영향을 주지 않으면서 가치 창출 시간을 가속화하고 현대화, 최적화 및 확장할 수 있습니다.
- 애플리케이션 환경을 프로파일링하여 변화에 따른 이점, 위험 및 비용을 식별합니다.
- 전략 및 계획, 구현, 마이그레이션 및 애플리케이션 현대화, 지속적인 지원에 이르기까지 E2E 서비스 세트를 제공합니다.
- 지속적인 비즈니스 가치를 제공하기 위한 재사용 가능한 관행과 도구를 제공하는 정책, 권장 사항 및 제어를 구축합니다.

평가 단계에서 애플리케이션 소유자와 설계자는 현대화 진단 플레이북을 사용하여 현대화 목표와 우선 순위를 검증합니다.

현대화 진단 플레이북 사용합니다

현대화 진단 플레이북은 기업이 현재 상태에서 미래 상태로 전환하는 데 따르는 가치를 결정하는 프로세스를 제공합니다. 여기에는 현대화에 수반되는 기술 변화가 포함됩니다.

진단 플레이북을 사용하여 클라우드 현대화를 위한 애플리케이션 또는 애플리케이션 제품군의 우선순위를 결정하고 현대화 중에 해결해야 할 구성 요소를 식별합니다.

진단 차원

현대화 진단 플레이북은 애플리케이션 또는 애플리케이션 그룹의 현재 및 대상 (마이그레이션 후) 상태의 다음 차원을 이해하는 데 도움이 됩니다.

- 애플리케이션 그룹화 – 현대화를 위해 애플리케이션 (예: 기술 또는 운영 모델별)을 그룹화할 이유가 있습니까?
- 시퀀싱 – 종속성에 따라 애플리케이션을 현대화해야 하는 순서가 있습니까?
- 기술 – 기술 범주 (예: 미들웨어, 데이터베이스, 메시징)는 무엇입니까?
- 종속성 – 애플리케이션이 다른 시스템이나 미들웨어에 대한 주요 종속성을 가지고 있습니까?
- 환경 – 개발, 테스트 및 프로덕션 환경이 얼마나 많이 사용됩니까?
- 스토리지 – 스토리지 요구 사항은 무엇입니까 (예: 테스트 데이터의 사본 수)?
- 운영 모델 – 애플리케이션의 모든 구성 요소가 지속적 통합 및 지속적 전달 (CI/CD) 파이프라인을 채택할 수 있습니까?
 - 그렇다면 애플리케이션 팀과 누구에게 어떤 인프라 책임을 배분해야 할까요?
 - 그렇지 않다면 운영팀에 맡겨야 할 인프라 책임은 (예: 패치 작업) 무엇입니까?
- 전송 모델:
 - 애플리케이션 또는 애플리케이션 그룹을 기반으로 플랫폼 변경, 리팩터링, 재작성, 대체 중 무엇을 해야 할까요?
 - 현대화의 어느 부분에서 클라우드 네이티브 서비스를 사용해야 할까요?
- 기술 세트 – 어떤 전문 지식이 필요한가요? 예제:
 - 처음부터 컨테이너 및 서버리스 기술을 사용하여 모듈식 아키텍처로 애플리케이션을 구축하기 위한 클라우드 애플리케이션 배경.
 - 오픈 소스와 AWS 도구 및 서비스를 사용하여 CI/CD 프로세스, 코드형 인프라, 자동화 또는 애플리케이션 관찰성 영역에서 솔루션을 개발하는 DevOps 전문 지식입니다.
- 현대화 접근 방식 – 애플리케이션의 현재 상태, 클라우드 기술 선택, 현재의 기술 부채, CI/CD, 모니터링, 기술 및 운영 모델을 고려할 때 수행해야 할 기술적 마이그레이션 작업은 무엇입니까?

- 현대화 타이밍 – 현대화 시기에 영향을 미칠 수 있는 비즈니스 포트폴리오 타이밍 고려 사항 또는 기타 계획된 작업 고려 사항은 무엇입니까?
- 인프라의 단위 및 총 비용 - 경제 분석을 AWS기반으로 온프레미스에서 워크로드를 유지 관리하는 데 드는 연간 비용은 얼마입니까?

이러한 차원을 기준으로 애플리케이션을 평가하면 클라우드로 현대화를 추진하면서 비즈니스, 기술 및 경제성을 지속적으로 파악하는 데 도움이 됩니다.

빌딩 블록

애플리케이션을 현대화할 때 관찰 결과를 비즈니스 민첩성, 조직 민첩성, 엔지니어링 효율성의 세 가지 구성 요소로 분류할 수 있습니다.

- 비즈니스 민첩성 - 비즈니스 요구 사항을 요구 사항으로 전환하기 위한 비즈니스 내 효율성과 관련된 사례. 전달 조직이 비즈니스 요청에 얼마나 대응하고 있는지, 기능을 프로덕션 환경에 릴리스하는 데 있어 비즈니스가 어느 정도 통제할 수 있는지.
- 조직 민첩성 – 전달 프로세스를 정의하는 관행. 예를 들어 애자일 방법론과 DevOps 행사, 역할 할당 및 명확성, 조직 전반의 전반적인 협업, 커뮤니케이션 및 지원 등이 있습니다.
- 엔지니어링 효율성 – 품질 보증, 테스트, CI/CD, 구성 관리, 애플리케이션 설계, 소스 코드 관리와 관련된 개발 사례.

지표 식별

고객에게 중요한 것을 제공하고 있는지 알아보려면 개선을 주도하고 제공을 가속화하는 조치를 구현해야 합니다. 목표, 질문, 지표 (GQM) 는 측정값이 이러한 기준을 충족하는지 확인하기 위한 효과적인 프레임워크를 제공합니다. 이 프레임워크를 사용하여 다음 단계에 따라 목표를 달성할 수 있습니다.

1. 수행 중인 목표 또는 결과를 파악하세요.
2. 목표 달성 여부를 판단하기 위해 반드시 답해야 할 질문을 도출하세요.
3. 질문에 적절하게 답하기 위해 무엇을 측정해야 하거나 측정할 수 있는지 결정하세요. 측정에는 다음 2가지 카테고리가 있습니다.
 - 제품 메트릭은 고객에게 중요한 것을 제공하고 있는지 확인하는 데 도움이 됩니다.
 - 소프트웨어 제공 라이프사이클을 개선하는 데 도움이 되는 운영 메트릭입니다.

제품 지표

제품 지표는 비즈니스 성과에 초점을 맞추므로 새로운 작업 범위에 대한 투자수익률 (ROI) 이 결정될 때 설정해야 합니다. 제품 메트릭을 설정하는 데 유용한 기법 중 하나는 새로운 작업 범위가 구현되면 비즈니스에 어떤 변화가 생길지 묻는 것입니다. 이러한 생각을 현대화 기능이 제공되었을 때 무엇이 사실인지에 초점을 맞춘 테스트 형식으로 공식화하는 것이 도움이 됩니다.

예를 들어, 기존 시스템에서 트랜잭션을 마이그레이션하면 클라이언트를 온보딩할 수 있는 새로운 기회가 열릴 것이라고 생각한다면 어떤 점이 개선될까요? 다음 클라이언트를 온보딩하려면 얼마나 많은 용량을 확보해야 할까요? 그 결과를 검증하기 위한 테스트는 어떻게 구성될까요? 이 시나리오의 경우 제품 지표에는 다음과 같은 항목이 포함됩니다.

- 비즈니스 가치 테스트 또는 가설을 식별하세요 (예: x퍼센트의 트랜잭션 용량을 확보하면 신규 비즈니스의 y퍼센트를 확보할 수 있음).
- 기준을 정하세요 (예: x개 거래의 현재 용량이 y명의 고객을 지원할 수 있음).
- 결과 검증 (예를 들어, 용량이 x퍼센트 개선되었는데 이제 y퍼센트의 신규 비즈니스를 도입할 수 있을까요?)

운영 지표

소프트웨어 제공 라이프사이클을 개선하고 현대화를 가속화하고 있는지 판단하려면 소프트웨어 제공에 소요되는 리드 타임과 구현 시간을 알아야 합니다. 즉, 비즈니스 요구 사항을 프로덕션 환경에서 기능으로 얼마나 빨리 전환할 수 있을까요?

유용한 운영 지표는 다음과 같습니다.

- 리드 타임 – 요청에서 생산까지 작업 범위가 이동하는 데 시간이 얼마나 걸립니까?
- 주기 시간 – 처음부터 끝까지 작업 범위를 구현하는 데 시간이 얼마나 걸립니까?
- 배포 빈도 – 변경 내용을 프로덕션에 얼마나 자주 배포합니까?
- 서비스 복원 시간 – 장애 복구에 걸리는 시간 (평균 복구 시간 또는 MTTR로 측정) 은 얼마나 걸립니까?
- 변경 실패율 – 평균 장애 간격 (MTBF)은 얼마입니까?

2단계. 작게 시작해서 추진력을 키우세요

이 단계의 목표는 초기 최소 실행 가능 제품 (MVP)을 제공하여 모멘텀을 확보하는 것입니다. 이 접근 방식을 통해 비즈니스 성과를 조기에 그리고 점진적으로 이끌어낼 수 있습니다.

우선 순위 드라이버 검증

애플리케이션 팀과 함께 현대화 작업을 시작하기 전에 앞서 결정한 우선 순위 요인을 검증하는 것이 좋습니다. 다음 단계를 따릅니다.

1. 진단 플레이북에서 필요한 정보를 컴파일하세요.
 - 우선순위 애플리케이션 목록에서 우선순위 동인과 타당성 평가를 수집하세요.
 - 애플리케이션에 대한 전환 및 목표 상태 성향을 수집하세요.
 - 클라우드 현대화 계획에서 애플리케이션 소유자, 설계자 및 이해 관계자를 식별하세요.
 - 종속성 또는 애플리케이션 스위트 시퀀싱 (알고 있는 경우) 에 대한 정보를 요청하세요.
 - 인벤토리 항목이 종속성 또는 애플리케이션 스위트 그룹과 어떤 관련이 있는지 확인하세요. 애플리케이션에는 다른 구성 요소와 긴밀하게 연결되거나 종속되는 개별 구성 요소가 있을 수 있으며 이러한 구성 요소를 함께 현대화할 수도 있습니다.
2. 1단계 직원들과 한 시간 또는 두 시간 회의를 예약하여 우선 순위가 높은 요인을 검증하세요.
 - 솔루션 엔지니어 또는 설계자별로 여러 애플리케이션 (최대 3~4개)을 그룹화하고 애플리케이션 종속성 또는 애플리케이션 제품군 정보를 기반으로 한 회의에서 논의해 보세요.
 - 예정된 이번 회의에서 각 팀원의 역할과 기대치를 결정하세요.
3. 회의를 진행하세요.

세부 사항 마무리하세요

이전 섹션의 프로세스에 따라 우선 순위 요인을 검증한 후 세부 정보를 수집하여 현대화 접근 방식과 시기를 결정할 수 있습니다.

이 단계에서 코어 팀은 2일간의 짧은 스프린트를 통해 애플리케이션 팀과 협력하여 AWS 클라우드에서 애플리케이션의 미래 상태를 설계합니다. 활동에는 제품 정의, 제품 발견, 스토리 작성, 가치 흐름 매핑, CI/CD 프로세스 설계 등이 포함됩니다. 다음은 몇 가지 아이디어입니다.

- 애플리케이션의 각 개별 구성 요소 (예: 네트워크 구성, 스토리지 구성, 데이터베이스, 서버, 애플리케이션이 서버에 배포되는 방식)를 모델링합니다.
- 컨테이너나 서버리스 기술과 같은 도구를 사용하여 해당 모델을 다양한 구성 요소 및 구성으로 분해할 수 있습니다.
- 애플리케이션 기능을 기본 인프라에 대한 종속성과 분리합니다. 애플리케이션의 기능을 소스 코드를 변경하지 않고 이동할 수 있는 구성 요소로 추상화합니다.
- CI/CD 도구 및 메커니즘을 사용하여 DevOps와 긴밀하게 통합합니다.

기본 플랫폼 서비스 구축 및 애플리케이션 현대화합니다

이 12주 단계에서는 풀스택 팀의 지원을 받아 우선 순위가 지정된 비즈니스 사용 사례를 제공합니다. 이 작업은 여러 조의 피자 두 판 팀이 수행합니다. 예를 들어, 기본 플랫폼 서비스를 개발하기 위해 플랫폼 엔지니어링 팀을 구성하고 새로운 비즈니스 성과를 제공하기 위해 제품 팀을 구성합니다.

- 플랫폼 엔지니어링 팀은 클라우드 기반, 개발자 워크플로우 및 데이터 분석 기능을 지원하는 AWS 서비스를 구성, 통합 및 사용자 지정합니다. 규모가 크고 복잡한 엔터프라이즈에는 이러한 각 기능을 지원하는 팀이 여러 개 있을 수 있습니다.
- 제품 팀은 초기 단계에서 우선순위를 정한 비즈니스 성과를 위해 새로운 서비스와 경험을 개발합니다. 제품 팀은 새로운 서비스를 개발하면서 핵심 비즈니스 역량도 현대화합니다.

플랫폼 엔지니어링 및 제품 팀은 고객이 평가할 수 있는 최소 실행 가능 제품 (MVP) 을 제공합니다. 초기 MVP가 성공하면 새로운 애플리케이션을 식별하고 초기 팀 구성원을 분할하여 새 제품 팀을 구성하는 스플릿 앤 시드 방식을 사용하여 현대화 프로그램을 확장할 수 있습니다.

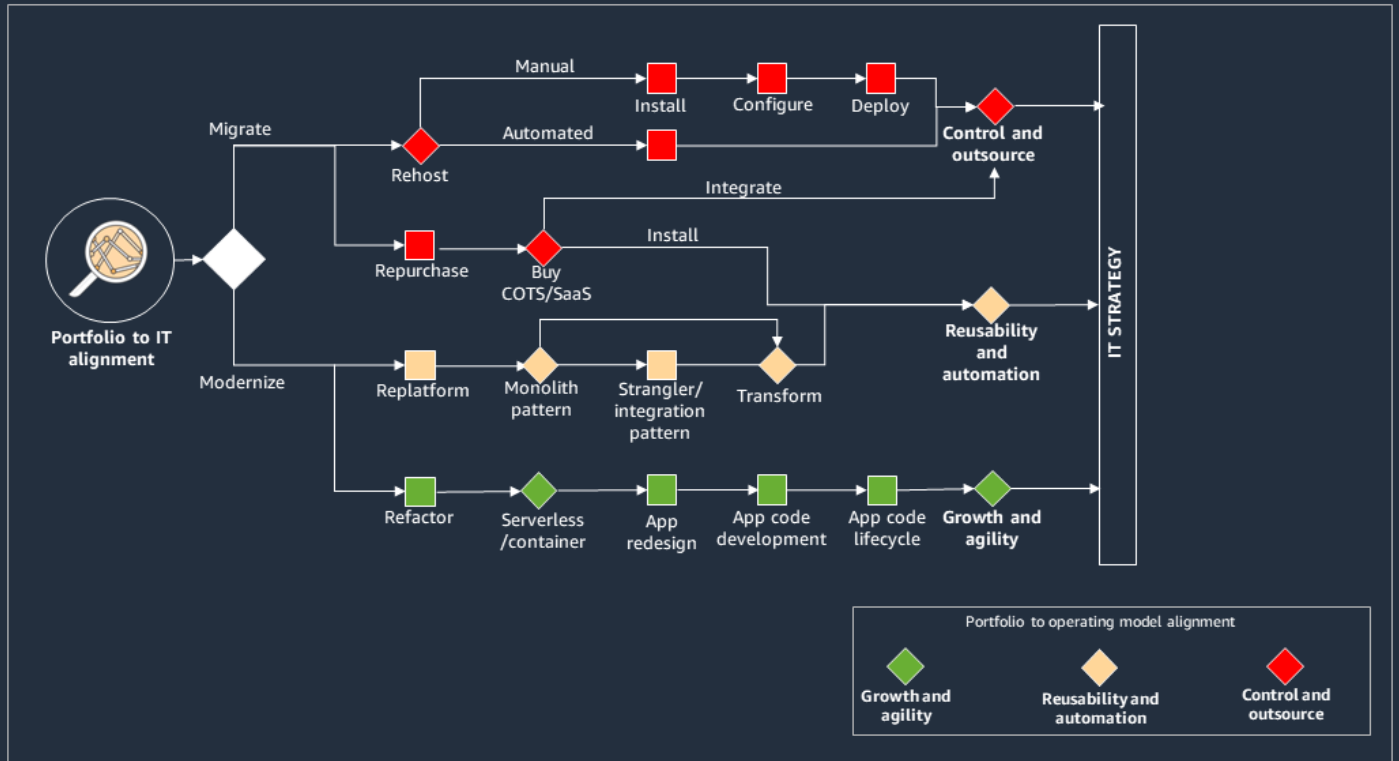
3단계. 확장 가능한 현대화 로드맵을 만드세요

우선 순위가 지정된 결과 및 애플리케이션의 초기 MVP 릴리스 후에는 현대화 노력을 확장 및 가속화하고, 개발자 생산성을 높이고, 빠르게 혁신하기 위한 로드맵을 개발하는 것이 좋습니다. 핵심 팀은 비즈니스 성과에 초점을 맞춘 여러 엔지니어링 팀 전반으로 조직의 역량과 서비스를 확장하기 위해 팀을 분할하고 새로운 팀을 파견합니다. 시간이 지남에 따라 스플릿 앤 시드 방식을 채택함으로써 조직은 더 많은 개발에 착수하고 현대화 속도를 가속화할 수 있습니다.

현대화 로드맵은 이벤트 기반, 스트랭글러, 도메인 기반 설계, 분해, 최신 데이터베이스 옵션 등과 같이 명확하게 정의된 패턴을 사용하여 애플리케이션 현대화에 대한 실용적이고 지속적인 접근 방식을 설명해야 합니다.

로드맵에는 다음 다이어그램과 같이 애플리케이션 구성 요소를 식별하여 비즈니스 로직을 변경하지 않고 관리 서비스 (예: 데이터베이스 서비스)로 이동하거나 코드 수준을 변경하여 성능, 확장성, 관리 용이성, 안정성 및 리소스 사용을 개선하기 위한 의사 결정 트리 매트릭스가 포함되어야 합니다.

Migration and modernization decisions



현대화 준비 요인

애플리케이션을 현대화할 때는 다음과 같은 표준과 모범 사례를 준수해야 합니다.

주제

- [코드](#)
- [빌드 및 테스트](#)
- [릴리스](#)
- [운영](#)
- [최적화](#)
- [준비 상태](#)

코드

- 소프트웨어 기능을 문서화하는 코드 주석을 제공하고 이를 사용하여 문서를 생성하십시오.
- 작은 코드 체크인과 기능 요청 추적성을 지원하는 코드 관리 및 배포 프로세스를 따르십시오.
- 100% 코드 커버리지로 단위, 기능, 성능 및 임계 경로 테스트를 포함하는 테스트 스위트를 구축하십시오.
- 코드 재사용을 장려하여 코드베이스에 동일하거나 유사한 기능을 제공하십시오.
- 전체 코드 개발에 투자하기 전에 프로토타입을 개발하여 사용자와 함께 기능을 검증하십시오.

빌드 및 테스트

- 테스트를 기반으로 기능 완전성을 재정의하여 품질을 개선하고 반복되는 문제를 방지하십시오.
- 승인 테스트를 자동화하십시오.
- 모든 자동 테스트를 모니터링하고 장애를 처리하기 위한 프로세스를 마련하십시오.
- 프로덕션 및 비프로덕션 환경 모두에서 성능을 추적하고, 실제 트래픽 및 로드 테스트를 기반으로 서비스 레벨 목표(SLO)를 정의하고, 성능 요구 사항에 맞게 확장할 수 있는 기능을 제공합니다.
- 구성 파일에서 민감한 데이터를 추출하고 구성을 자동화하고 모니터링하는 도구를 제공합니다.

릴리스

- 종속성 (예: 데이터베이스 릴리스), 회귀 테스트 및 추적을 지원하여 배포를 자동화합니다.
- 빌드가 성공할 때마다 프로덕션 환경에 코드를 점진적으로 릴리스하십시오.
- 기능 플래그를 (토글) 효과적으로 관리합니다. 런타임 구성을 지원하고, 사용량을 모니터링하고, 개발 주기 전반에 걸쳐 플래그를 유지하고, 카테고리별로 소유자를 할당합니다.
- 빌드 파이프라인에 추적 기능을 제공하여 트리거, 실패 알림 및 성공적인 완료를 추적할 수 있습니다.
- 자동 배포 프로세스를 실행하고 지속적 배포에서 '제로 터치' 코드 업데이트를 테스트하십시오.
- 다운타임이 전혀 없는 완전 자동화된 블루/그린 배포 방법론을 사용하십시오.
- 데이터베이스 스키마 변경이 모든 개발 및 프로덕션 환경에서 일관되게 구현되도록 하십시오.

운영

- 알림 시스템과 통합된 DevOps 심사 런북을 만드십시오.
- 모니터링 및 알림 시스템이 서비스 레벨 목표(SLO)를 충족하고 임계값, 상태 점검, 비표준 HTTP 응답 및 예상치 못한 결과를 지원하는지 확인하십시오.
- 효과적인 위험 관리 및 재해 복구 프로세스를 수립하십시오.
- 비즈니스 및 법적 요구 사항을 충족하는 로그 순환 및 보존 전략을 개발하십시오.
- 제품 성능을 추적하고, 새로운 기능의 성공을 측정하고, 지표가 기대에 미치지 못할 경우 알림을 표시하는 대시보드를 개발하십시오.

최적화

- 성과 및 품질 측정을 기반으로 프로세스를 정기적으로 검토하고 개선하십시오.
- 근본 원인 분석 및 예방 프로세스를 구현하여 문제가 재발하지 않도록 하십시오.
- 제품 상태를 파악하는 데이터 기반 지표를 제공하고 모든 알림 및 조치가 이러한 지표를 기반으로 이루어지도록 하십시오.

준비 상태

- 다양한 기능을 갖춘 팀 (비즈니스 파트너, 개발자, 테스터, 설계자 등) 을 현대화 작업에 전담 시키십시오.

다음 단계

이 가이드는 애플리케이션을 현대화하기 위한 단계별 및 점진적 접근 방식을 제공했습니다. 단기, 중기 및 장기적 차원에서 기술 및 제공 범위를 파악하여 워크스트림 범위를 정하는 것이 좋습니다. 팀 전체에 더 효과적인 전달 및 엔지니어링 관행을 촉진하기 위해 툴, 프레임워크 및 관행에 더 많은 투자를 할 것으로 예상할 수 있습니다. 현대화 프로세스의 목표를 설정하고, 현대화하려는 각 애플리케이션을 이해하고, 각 애플리케이션에 맞는 최적의 현대화 접근 방식을 선택하십시오. 진행 상황을 모니터링하고 최적화하십시오.

AWS은(는) 조직의 비즈니스 개발 팀과 협력하여 기술 현대화를 추진하고 애플리케이션 아키텍처, 설계 및 개발, 모바일 지원, 테스트, 협업 솔루션 등 종합적인 요구 사항을 관리합니다. 이러한 서비스는 검증된 프로세스, 지능형 자동화, 데이터 및 패턴 사용, 개방형 표준, 레거시 애플리케이션을 현대화하는 데 도움이 되는 적절한 인력을 결합합니다. 주요 목표는 다음과 같습니다.

- 레거시 기술을 현대화하기 위한 완전 자동화된 툴 기반 접근 방식을 제공하고 이를 현대화에 대한 총체적 지식과 결합하는 플랫폼을 개발하도록 지원합니다.
- 고객 성과 및 규모에 초점을 맞춘 현대화 엔지니어링 기능을 갖춘 풀 스택 팀과 함께 MVP를 구축하십시오.

AWS 전문 서비스 및 AWS 파트너는 선임 기술 및 비즈니스 고객 이해 관계자와 직접 협력하여 기업 현대화를 추진하는 동시에 최종 고객 가치를 반복적으로 제공합니다.

리소스

관련 가이드

- [AWS 클라우드에서 애플리케이션을 현대화하기 위한 전략](#)
- [AWS 클라우드의 애플리케이션에 대한 현대화 준비 상태 평가](#)
- [AWS 클라우드에서 작업 현대화](#)
- [마이크로서비스로 모놀리식 유형 분해](#)
- [AWS 서버리스 서비스를 사용하여 마이크로서비스 통합](#)
- [AWS Change Acceleration 6-Point Framework 및 Organizational Change Management Toolkit](#)

AWS 리소스

- [AWS 설명서](#)
- [AWS 일반 참조](#)
- [AWS 용어집](#)

문서 기록

아래 표에 이 가이드의 주요 변경 사항이 설명되어 있습니다. 향후 업데이트에 대한 알림을 받으려면 [RSS 피드](#)를 구독하십시오.

변경 사항	설명	날짜
확장된 지침	—	2023년 5월 25일
최초 게시	—	2020년 12월 18일

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.