



Acelerando os ciclos de vida de desenvolvimento de software com IA generativa AWS

AWS Recomendações



AWS Recomendações: Acelerando os ciclos de vida de desenvolvimento de software com IA generativa AWS

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

As marcas comerciais e imagens de marcas 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

Introdução	1
Objetivos	1
Público-alvo	2
Experiência de desenvolvimento	3
Usando IA generativa	5
Estrutura 5-I	6
Visão geral da estrutura	7
Integração com o SDLC	9
Capacidades fundamentais	10
Gerenciamento de projetos	17
Gerenciamento de requisitos	20
Arquitetura e design	21
Colaboração	22
DevSecOps	24
Operação e manutenção	32
Assistentes de IA	34
Análise e insights	36
Gerenciar conhecimento	39
Extensibilidade	40
Práticas recomendadas	42
Conjunto de ferramentas integrado	42
DevSecOps tubulação	43
Ferramentas e práticas colaborativas	43
Automação de tarefas	44
Revisão e iteração	44
Práticas de gerenciamento de projetos	45
Gerenciar conhecimento	45
Extensibilidade e personalização	46
Otimização	46
Insights baseados em dados	46
Abordagem baseada em plataforma	47
Medindo o sucesso	48
Velocidade de implantação	49
Qualidade do código	49

Eficiência operacional	50
Produtividade e satisfação da equipe	50
Impacto nos negócios	51
Conclusão	52
Recursos	52
Histórico do documento	54
.....	lv

Acelerando os ciclos de vida de desenvolvimento de software com IA generativa AWS

Chetan Makvana, Amazon Web Services

Abril de 2025 ([histórico do documento](#))

A crescente demanda por software de alta qualidade está levando as organizações a buscarem constantemente maneiras de acelerar seu ciclo de vida de desenvolvimento de software (SDLC). À medida que as organizações se esforçam para permanecer competitivas, é fundamental reduzir o tempo de lançamento no mercado e, ao mesmo tempo, manter ou melhorar a qualidade do produto. Para enfrentar esses desafios, a experiência de desenvolvimento de software deve evoluir e usar tecnologias, metodologias e práticas de ponta que simplifiquem os processos e capacitem as equipes de desenvolvimento de software a serem mais produtivas e criativas. O surgimento da experiência de desenvolvimento da próxima geração marca uma mudança significativa na forma como o software é concebido, construído, testado e implantado. Ele integra uma variedade de recursos, incluindo desenvolvimento nativo em nuvem, automação orientada por IA, gerenciamento avançado de projetos, ferramentas colaborativas e, coletivamente, aprimoram a eficiência e DevSecOps a eficácia do SDLC.

Na vanguarda dessa transformação está o surgimento da IA generativa na engenharia de software. De acordo com o [Gartner](#), 40% das equipes de engenharia de plataforma usarão a IA para aumentar todas as fases do SDLC até 2027, em comparação com apenas 5% em 2023. Este relatório também afirma que os líderes de engenharia de software agora devem se preparar para adotar a IA generativa em uma ampla gama de áreas que são críticas para o processo de desenvolvimento. Em outro relatório, [McKinsey](#) pesquisas mostram que empresas com maior índice de velocidade do desenvolvedor aumentam a receita de 4 a 5 vezes mais rápido, têm retornos 60% maiores para os acionistas e são 55% mais inovadoras. Ao adotar a IA generativa além da geração de código, as organizações podem desbloquear um novo nível de eficiência, produtividade e inovação em seus fluxos de trabalho de desenvolvimento de software. Isso pode reduzir o esforço manual, revelar insights e aumentar a experiência humana.

Objetivos

Este documento de estratégia descreve uma estrutura, recursos fundamentais, casos de uso, melhores práticas e métricas de sucesso que podem ajudá-lo a acelerar seu SDLC com IA

generativa. Ele descreve como integrar efetivamente a IA generativa em todos os estágios de desenvolvimento para melhorar a qualidade e a eficiência do produto.

Este documento de estratégia pode ajudar você e sua organização a alcançar os seguintes objetivos:

- Implemente uma estrutura, recursos fundamentais, casos de uso, melhores práticas e métricas de sucesso para acelerar seu SDLC com IA generativa.
- Integre efetivamente a IA generativa em todos os estágios de desenvolvimento para melhorar a qualidade do produto, a velocidade de lançamento e a eficiência do desenvolvimento.
- Adapte-se à próxima geração de desenvolvimento de software incorporando tecnologias, metodologias e práticas de IA de ponta que simplificam os processos e capacitam as equipes de desenvolvimento.

Público-alvo

Este documento de estratégia é para líderes de TI, gerentes de engenharia, diretores de tecnologia e equipes de desenvolvimento de software que desejam acelerar seu ciclo de vida de desenvolvimento de software aplicando IA generativa às suas práticas de desenvolvimento.

Entendendo a experiência de desenvolvimento de software

A experiência de desenvolvimento de software abrange o ambiente, as ferramentas e os processos que suas equipes de desenvolvimento usam em todo o ciclo de vida de desenvolvimento de software (SDLC). Ele inclui o ambiente de desenvolvimento integrado (IDE), plataformas de colaboração, estruturas de teste, sistemas de gerenciamento de conhecimento, pipelines de implantação e muito mais.

Uma experiência de desenvolvimento bem projetada simplifica os fluxos de trabalho, reduz o esforço manual e capacita suas equipes a se concentrarem em tarefas de alto valor, o que, em última análise, acelera seu SDLC. Por exemplo, ao integrar perfeitamente seu IDE, sistema de controle de versão e ferramentas de implantação, você permite que os desenvolvedores escrevam, testem e implantem código com maior velocidade e eficiência em comparação com uma cadeia de ferramentas fragmentada que requer transferências manuais e troca de contexto. Da mesma forma, a integração de uma estrutura robusta de gerenciamento de conhecimento ajuda as equipes a acessar e compartilhar facilmente o conhecimento institucional, as melhores práticas e a documentação. Isso aumenta sua produtividade geral e suas capacidades de resolução de problemas.

A experiência de desenvolvimento de software tem um impacto direto no desempenho geral e no sucesso de uma equipe de desenvolvimento de software. Uma experiência abaixo do ideal pode levar ao seguinte:

- **Produtividade reduzida** — Ferramentas ineficientes, fluxos de trabalho complexos e falta de automação prejudicam a produtividade da equipe, o que retarda a entrega de recursos e atualizações.
- **Aumento da dívida técnica** — ferramentas e processos ad-hoc mal integrados podem resultar em débito técnico, o que torna mais difícil manter e escalar seus sistemas de software ao longo do tempo.
- **Inovação reduzida** — Quando atolada em tarefas manuais e repetitivas, a capacidade da sua equipe de explorar novas tecnologias e impulsionar a inovação é limitada.
- **Qualidade comprometida** — processos fragmentados de teste e implantação aumentam o risco de defeitos e vulnerabilidades de software. Isso pode afetar negativamente a qualidade geral do software fornecido.

Ao investir em uma experiência de desenvolvimento de software bem projetada, você pode obter benefícios significativos, como menor tempo de lançamento no mercado, melhor qualidade do software, maior satisfação da equipe de desenvolvimento de software e maior agilidade nos negócios.

Potencializando a experiência de desenvolvimento de software com IA generativa

A integração da IA generativa no ciclo de vida de desenvolvimento de software (SDLC) representa uma mudança de paradigma na forma como equipes inteiras de desenvolvimento de software concebem, projetam, implementam e mantêm soluções de software. A IA generativa tem o potencial de revolucionar todas as fases do SDLC, incluindo gerenciamento de projetos, coleta de requisitos, design, codificação, teste, implantação e manutenção.

Em essência, uma experiência generativa de AI-powered desenvolvimento atua como um colaborador inteligente para toda a sua equipe de desenvolvimento de software, incluindo gerentes de produto, designers, arquitetos de soluções, desenvolvedores, testadores e equipe de operações. Ele fornece assistência contextual, gera artefatos (como histórias de usuários, modelos de design, trechos de código e casos de teste), oferece sugestões quase em tempo real e até mesmo prevê possíveis problemas antes que eles surjam. Essa AI-augmented abordagem reduz significativamente a carga cognitiva dos membros da equipe. Isso permite que eles se concentrem em decisões estratégicas de alto nível e na resolução de problemas complexos, enquanto a IA generativa lida com as tarefas mais mundanas e repetitivas.

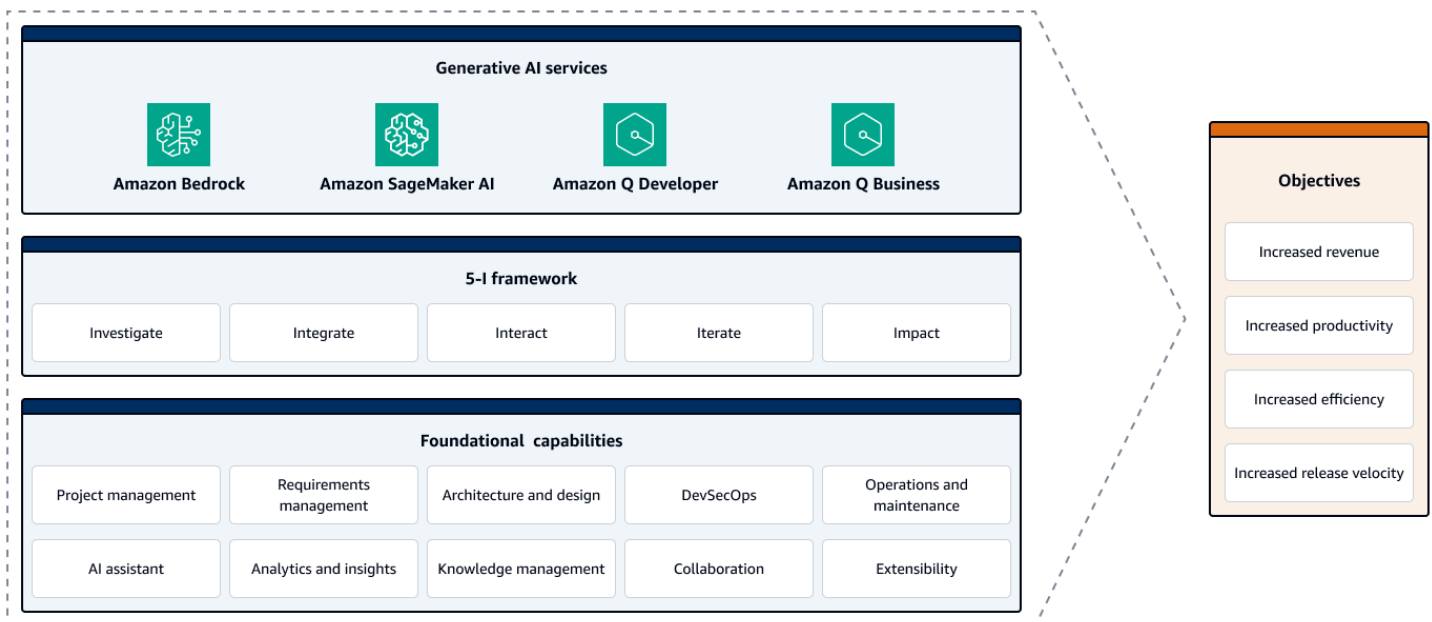
A IA generativa também serve como um amplificador de conhecimento. Ele ajuda os membros da equipe a acessar rapidamente informações relevantes, melhores práticas e padrões de vastos repositórios de dados. Isso pode democratizar efetivamente a experiência em toda a organização. Ao integrar perfeitamente os recursos generativos de IA em toda a cadeia de ferramentas de desenvolvimento, você pode criar um ambiente mais intuitivo, eficiente e produtivo para todas as suas equipes de desenvolvimento de software. Essa experiência aprimorada de desenvolvimento acelera o SDLC e melhora a qualidade geral. Também reduz os erros e promove a inovação, pois os membros da equipe podem explorar novas ideias e abordagens mais rapidamente.

Para adotar uma experiência de AI-powered desenvolvimento generativo em sua organização, considere os seguintes elementos-chave:

- [Estrutura 5-I](#)— Composta por cinco dimensões, a estrutura 5-I fornece uma abordagem abrangente para navegar no processo de desenvolvimento de software moderno. Ele oferece uma metodologia estruturada que ajuda você a aplicar sistematicamente a IA generativa em todos os estágios do SDLC.

- **Capacidades fundamentais**— Para usar totalmente o poder da IA generativa em todas as dimensões do desenvolvimento de software moderno, você precisa estabelecer um conjunto robusto de recursos fundamentais. Esses recursos formam a espinha dorsal de uma experiência de AI-powered desenvolvimento. Esses recursos ajudam você a integrar e usar a IA generativa em todo o SDLC.

Juntos, a estrutura 5-I e os recursos fundamentais formam uma estratégia para reimaginar a experiência de desenvolvimento de software. As cinco dimensões fornecem uma estrutura estratégica para a aplicação da IA generativa, e os recursos fundamentais preparam sua organização para apoiar essa AI-driven abordagem. Serviços da AWS, como [Amazon Bedrock](#), [Amazon SageMaker AI](#), [Amazon Q Developer](#) e [Amazon Q Business](#), fornecem recursos e recursos de IA generativos que você pode integrar à sua experiência de desenvolvimento de software.



Estrutura 5-I para uma experiência de desenvolvimento AI-powered de software

A estrutura 5-I fornece uma abordagem estruturada para que as equipes de desenvolvimento de software integrem efetivamente a IA generativa em suas práticas de desenvolvimento. Ele ajuda você a estabelecer uma base sólida para usar a IA generativa em todo o SDLC. Também ajuda você a configurar as práticas de desenvolvimento, os fluxos de trabalho e as mentalidades corretas para aproveitar totalmente o potencial da IA generativa.

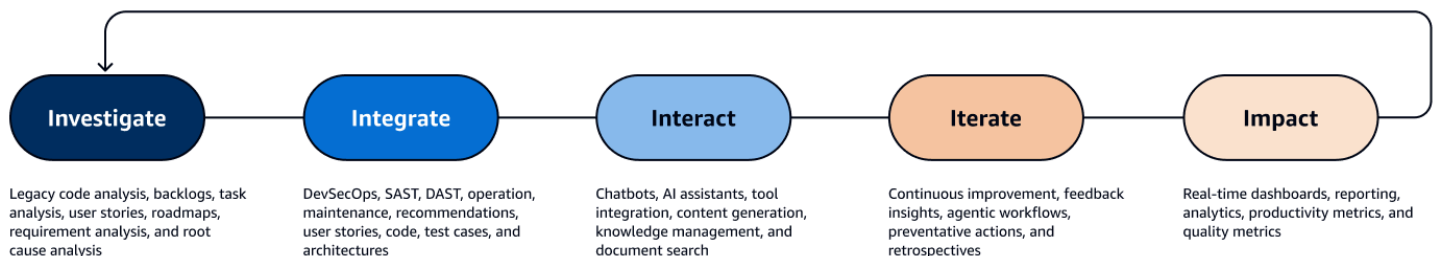
Esta seção contém os seguintes tópicos:

- [Visão geral da estrutura](#)
- [Integração com o ciclo de vida de desenvolvimento de software](#)

Visão geral da estrutura

A estrutura 5-I é construída em torno de cinco dimensões principais: Investigar, Integrar, Interagir, Iterar e Impactar. Cada dimensão representa uma área crítica em que a IA generativa aprimora significativamente o processo de desenvolvimento de software. Ao integrar estrategicamente a IA generativa em todas essas dimensões, a estrutura atende às crescentes necessidades do desenvolvimento de software moderno. Pode reduzir a carga cognitiva e ampliar o potencial criativo. Ela reconhece que a experiência de desenvolvimento ideal não envolve apenas ferramentas, mas sim criar um ambiente em que a IA aprimore perfeitamente as capacidades humanas em todas as etapas.

O diagrama a seguir mostra as cinco dimensões do desenvolvimento de AI-powered software. Para cada dimensão, ele mostra onde você pode integrar a IA generativa para impulsionar a eficiência e a inovação.



A seguir estão as cinco dimensões da estrutura:

- **Investigue** — aprimore cada tarefa analítica em seu processo de desenvolvimento de software com IA generativa. Use a IA generativa para entender os requisitos, processar grandes quantidades de dados, reconhecer padrões e gerar insights que podem estar além da capacidade humana ou levar muito mais tempo para serem produzidos. Esses insights ajudam você a tomar decisões mais informadas, identificar rapidamente oportunidades de melhoria e fornecer software de alta qualidade com mais eficiência. A IA generativa pode ser uma parceira inteligente para os processos analíticos em todo o SDLC. Ao aproveitar a IA generativa, você aplica uma análise aprofundada a áreas críticas, como coleta de requisitos, exame de bases de código antigas e otimização do backlog de produtos. Por exemplo, os proprietários de produtos podem usar a

IA generativa para analisar as jornadas ou os requisitos do usuário antes de criar histórias de usuários. As equipes de desenvolvimento podem descobrir ineficiências e identificar oportunidades de otimização nas bases de código existentes. DevOps os engenheiros podem aplicar a análise de causa raiz para diagnosticar rapidamente problemas de desempenho ou vulnerabilidades de segurança, o que pode melhorar a confiabilidade.

- **Integrar** — Integre a IA generativa para automatizar uma ampla variedade de tarefas e processos em todo o SDLC. Isso inclui a geração automática de trechos de código, casos de teste, projetos arquitetônicos, histórias de usuários e pipelines de implantação. Ao automatizar essas tarefas tipicamente manuais, as equipes podem se concentrar em um trabalho mais estratégico e inovador, o que aumenta o tempo de lançamento no mercado e aplicativos de alta qualidade. A dimensão Integrate representa uma mudança de paradigma no desenvolvimento de software, em que a IA se torna parte integrante do processo de desenvolvimento. Ele trabalha junto com sua equipe de desenvolvimento de software para aumentar a produtividade, melhorar a qualidade e impulsionar a inovação. Isso resulta em um tempo de lançamento mais rápido no mercado. Isso desafia suas equipes de desenvolvimento de software a avaliar regularmente seus processos e fluxos de trabalho perguntando em cada etapa: “Isso pode ser automatizado?”
- **Interaja** — Use AI-powered assistentes generativos para fornecer à sua equipe suporte instantâneo e contextual em uma variedade de tarefas e consultas. Esses assistentes inteligentes atuam como colaboradores experientes que se baseiam em um vasto repositório de informações. Eles podem responder perguntas sobre codificação, oferecer sugestões de design, explicar procedimentos operacionais padrão e ajudar a solucionar problemas complexos. A integração desses assistentes de IA ao fluxo de trabalho de desenvolvimento aumenta a produtividade e promove um ambiente mais colaborativo e de resolução de problemas.
- **Iterar** — use IA generativa para permitir ajustes rápidos e orientados por dados em todo o SDLC. Você pode analisar continuamente dados de fontes como feedback de clientes, padrões de uso, tendências de mercado e métricas de desempenho da equipe para tomar decisões informadas rapidamente. Essa adaptabilidade refina seu desenvolvimento de software de um processo estático e predefinido para uma abordagem fluida e responsiva. Ela se manifesta de várias maneiras, incluindo priorização dinâmica de atrasos, alocação flexível de recursos, estratégias de teste adaptáveis, documentação em evolução e processos de implantação responsivos. Por exemplo, os gerentes de produto podem usar AI-generated insights para reordenar seus backlogs, integrando os novos requisitos dos clientes e as tendências do mercado quase em tempo real. DevOps os engenheiros podem adaptar os planos de implantação e as configurações de infraestrutura com base na análise de desempenho, garantindo que os aplicativos permaneçam resilientes e otimizados. As equipes de desenvolvimento podem traduzir o feedback das

- retrospectivas do sprint em melhorias acionáveis para a próxima iteração, impulsionando uma cultura de aprimoramento contínuo do processo.
- **Impacto** — aplique IA generativa para avaliar a eficácia e o desempenho do seu processo de desenvolvimento de software. Ao usar AI-powered análises e métricas, você obtém insights mais profundos sobre a eficiência do desenvolvimento, a qualidade do código, o envolvimento do usuário e o desempenho geral do aplicativo. Essa abordagem baseada em dados ajuda você a tomar decisões informadas, otimizar seus fluxos de trabalho de desenvolvimento e melhorar continuamente a qualidade e a experiência do usuário de seus aplicativos. Ao avaliar a produtividade da equipe de software, a IA generativa analisa vários pontos de dados, como frequência de confirmação de código, tempos de resolução de problemas, velocidade de lançamento, taxas de entrega de recursos e muito mais. Ele também pode avaliar a qualidade das revisões de código, a eficácia das ferramentas de colaboração e o impacto de diferentes práticas de desenvolvimento na produção geral da equipe. Ao correlacionar essas métricas com os resultados do projeto, a IA identifica padrões e tendências que os analistas humanos podem ignorar e pode fornecer insights acionáveis que aumentam a produtividade da equipe. Além disso, a IA generativa pode ajudá-lo a comparar o desempenho da equipe com os padrões do setor ou dados históricos, oferecendo recomendações personalizadas para melhoria. Ele também pode prever possíveis gargalos ou riscos no processo de desenvolvimento para que você possa tomar medidas proativas.

Integração com o ciclo de vida de desenvolvimento de software

O SDLC consiste em várias fases, que podem diferir de organização para organização. Normalmente, essas fases incluem o seguinte: requisitos e planejamento, design e arquitetura, implementação, teste, implantação e operação e manutenção.

A tabela a seguir mapeia as dimensões da estrutura 5-I para as fases do SDLC e fornece o nível de integração para cada dimensão.

Dimensão da estrutura	Requisitos e planejamento	Design e arquitetura	Implementação	Teste	Implantação	Operação e manutenção
Investigue	Alto	Baixo	Baixo	Baixo	Baixo	Médio

Dimensão da estrutura	Requisitos e planejamento	Design e arquitetura	Implementação	Teste	Implantação	Operação e manutenção
Integrar	Médio	Médio	Alto	Médio	Alto	Alto
Interaja	Alto	Alto	Alto	Médio	Médio	Alto
Iterar	Médio	Baixo	Baixo	Baixo	Baixo	Médio
Impacto	Alto	Médio	Alto	Baixo	Alto	Alto

Os níveis de integração variam de alto a baixo. O mapeamento revela as principais áreas de foco para cada dimensão. Por exemplo, Investigate mostra alta intensidade na fase de requisitos e planejamento. O Integrate demonstra alta intensidade nas fases de implementação, implantação e operação e manutenção.

Ao usar esse mapeamento, você pode priorizar seus esforços de forma eficaz. Recomendamos que você se concentre no alto, depois no médio e depois no baixo. Certifique-se de adotar uma abordagem equilibrada e impactante que aprimore a experiência de desenvolvimento de software com IA generativa.

Capacidades fundamentais para uma experiência de desenvolvimento AI-powered de software

Para implementar com sucesso uma experiência generativa de desenvolvimento de AI-powered software, você precisa estabelecer um conjunto de recursos fundamentais que abranjam várias pessoas em sua organização. Esses recursos representam sua capacidade de implantar recursos, implementar processos e alcançar os resultados desejados com eficiência no contexto do desenvolvimento de AI-powered software. Ao cultivar esses recursos, você cria uma base robusta que ajuda a integrar perfeitamente a IA generativa em todos os estágios do SDLC.

AWS fornece serviços essenciais para ajudá-lo a implementar esses recursos. Por exemplo, o [Amazon Q Developer](#) ajuda a acelerar o desenvolvimento de software atuando como AI-powered assistente. [O Amazon Q Business](#) ajuda você a obter respostas rápidas e relevantes para perguntas urgentes, resolver problemas e gerar conteúdo. Ele também pode agir em seu nome integrando

ferramentas relacionadas ao desenvolvimento de software. [O Amazon Bedrock](#) fornece acesso a modelos básicos e a um amplo conjunto de recursos para personalizar fluxos de trabalho e requisitos específicos de desenvolvimento.

Ao cultivar esses recursos Serviços da AWS, você cria uma base robusta que ajuda a integrar perfeitamente a IA generativa em todos os estágios do SDLC.

A seguir estão os recursos básicos nos quais você deve se concentrar:

- [Gerenciamento de projetos](#)
- [Gerenciamento de requisitos](#)
- [Arquitetura e design](#)
- [Colaboração](#)
- [DevSecOps](#)
- [Operação e manutenção](#)
- [Assistentes de IA](#)
- [Análise e insights](#)
- [Gerenciar conhecimento](#)
- [Extensibilidade](#)

Cada capacidade fundamental se integra às dimensões da estrutura e aos diferentes estágios do SDLC. Essa integração ajuda você a usar os recursos de IA de forma eficaz em todo o processo de desenvolvimento de software. Ele aumenta a eficiência, a qualidade e a inovação em cada etapa. A sinergia entre esses recursos fundamentais, a estrutura e os estágios do SDLC cria um ecossistema abrangente para o desenvolvimento de software. AI-powered Isso ajuda você a aproveitar todo o potencial da IA generativa, impulsionar a melhoria contínua, acelerar os ciclos de desenvolvimento e fornecer produtos de software de qualidade.

A tabela a seguir mostra como os recursos e subcapacidades fundamentais são mapeados para as dimensões da estrutura e as fases do SDLC.

Capacidade: subcapacidade	Investigue	Integrar	Interagir	Iterar	Impacto
Gerenciamento de projetos: gerenciamento de problemas	Requisitos e planejamento	Nenhum	Nenhum	Nenhum	Nenhum
Gerenciamento de projetos: gerenciamento de tarefas e sprint	Requisitos e planejamento	Requisitos e planejamento	Nenhum	Nenhum	Nenhum
Gerenciamento de projetos: gerenciamento da lista de pendências do produto	Requisitos e planejamento	Nenhum	Nenhum	Requisitos e planejamento	Nenhum
Gerenciamento de projetos: mapeamento de histórias de usuários	Requisitos e planejamento	Nenhum	Nenhum	Nenhum	Nenhum
Gerenciamento de projetos:	Requisitos e planejamento	Nenhum	Nenhum	Nenhum	Requisitos e planejamento

Capacidade: subcapacidade	Investigue	Integrar	Interagir	Iterar	Impacto
relatórios e análises					
Gerenciamento de projetos: gerenciamento de roteiros de produtos	Requisitos e planejamento	Nenhum	Requisitos e planejamento	Nenhum	Nenhum
Gerenciamento de projetos: ciclos de feedback	Nenhum	Nenhum	Nenhum	Requisitos e planejamento	Nenhum
Gerenciamento de projetos: retrospectivas	Nenhum	Nenhum	Nenhum	Requisitos e planejamento	Nenhum
Gerenciamento de requisitos	Requisitos e planejamento	Requisitos e planejamento	Nenhum	Nenhum	Nenhum
Arquitetura e design: design da solução	Design e arquitetura	Design e arquitetura	Nenhum	Nenhum	Nenhum

Capacidade: subcapacidade	Investigue	Integrar	Interagir	Iterar	Impacto
Colaboração: gerenciamento de documentação	Todas as fases do SDLC	Nenhum	Todas as fases do SDLC	Nenhum	Nenhum
Colaboração: compartilhamento de conhecimento	Todas as fases do SDLC	Nenhum	Todas as fases do SDLC	Nenhum	Nenhum
Colaboração: gerenciamento de ativos do projeto	Nenhum	Todas as fases do SDLC	Todas as fases do SDLC	Nenhum	Nenhum
DevSecOps: CI/CD	Teste, implantação	Implementação, teste e implantação	Implantação	Nenhum	Nenhum
DevSecOps : DevOps segurança	Implementação	Implementação, teste, operação e manutenção	Nenhum	Implementação, teste, operação e manutenção	Nenhum
DevSecOps : Monitoramento do desempenho de aplicativos	Nenhum	Operação e manutenção	Nenhum	Nenhum	Nenhum

Capacidade: subcapacidade	Investigue	Integrar	Interagir	Iterar	Impacto
DevSecOps : Agregação e análise de registros	Operação e manutenção	Operação e manutenção	Nenhum	Nenhum	Nenhum
DevSecOps: AIOps	Operação e manutenção	Nenhum	Nenhum	Operação e manutenção	Nenhum
DevSecOps : Melhoria contínua	Nenhum	Nenhum	Nenhum	Operação e manutenção	Nenhum
DevSecOps : Monitoramento do painel	Nenhum	Operação e manutenção	Nenhum	Nenhum	Nenhum
DevSecOps : Insights de desempenho	Operação e manutenção	Nenhum	Nenhum	Operação e manutenção	Nenhum
Operação e manutenção: gerenciamento de incidentes	Nenhum	Nenhum	Nenhum	Operação e manutenção	Nenhum
Operação e manutenção: atualizações de código	Nenhum	Operação e manutenção	Nenhum	Nenhum	Nenhum

Capacidade: subcapacidade	Investigue	Integrar	Interagir	Iterar	Impacto
Operação e manutenção: otimização de código	Operação e manutenção	Operação e manutenção	Nenhum	Nenhum	Nenhum
Operação e manutenção: Gestão técnica da dívida	Nenhum	Operação e manutenção	Operação e manutenção	Nenhum	Nenhum
Operação e manutenção: gerenciamento de mudanças	Nenhum	Implementação, implantação	Nenhum	Nenhum	Nenhum
Operação e manutenção: engenharia reversa	Operação e manutenção	Nenhum	Nenhum	Nenhum	Nenhum
Operação e manutenção: modernização do código	Nenhum	Implementação	Nenhum	Nenhum	Nenhum
Operação e manutenção: otimização do desempenho	Nenhum	Operação e manutenção	Nenhum	Operação e manutenção	Nenhum

Capacidade: subcapacidade	Investigue	Integrar	Interagir	Iterar	Impacto
Análises e insights	Nenhum	Requisitos e planejamento	Nenhum	Nenhum	Todas as fases do SDLC
Assistente de IA	Nenhum	Nenhum	Todas as fases do SDLC	Nenhum	Nenhum
Gestão do conhecimento	Nenhum	Nenhum	Todas as fases do SDLC	Nenhum	Nenhum
Extensibilidade	Nenhum	Implantação	Nenhum	Nenhum	Nenhum

Casos de uso generativos de IA para gerenciamento de projetos

O gerenciamento eficaz de projetos está no centro do desenvolvimento de software bem-sucedido. No contexto da IA generativa, o gerenciamento de projetos assume novas dimensões. Ele pode se tornar mais preditivo, adaptável e orientado por dados. AI-powered as ferramentas de gerenciamento de projetos analisam dados históricos do projeto para gerar estimativas mais precisas de tempo e recursos. Eles podem priorizar tarefas automaticamente com base nos objetivos de negócios e na capacidade da equipe, e podem até mesmo prever possíveis obstáculos antes que eles ocorram. Por exemplo, um gerente de projeto pode usar a IA generativa para criar um plano de projeto preliminar com base nos requisitos do projeto e nos dados históricos de projetos semelhantes. A IA poderia então sugerir composições de equipe ideais que levassem em conta as habilidades, as cargas de trabalho e as necessidades do projeto. Durante todo o projeto, os AI-driven painéis fornecem informações quase em tempo real sobre o status do projeto, gerando relatórios automaticamente e destacando áreas que exigem atenção.

Essa AI-augmented abordagem ao gerenciamento de projetos pode aumentar a eficiência. Isso ajuda os gerentes de projeto a se concentrarem na tomada de decisões estratégicas e na liderança da equipe, em vez de ficarem atolados em tarefas administrativas rotineiras.

A tabela a seguir mostra os casos de uso de gerenciamento de projetos que você pode aprimorar com a IA generativa e a pessoa responsável por esses casos de uso.

Subcapacidade: caso de uso	Pessoa
Gerenciamento de problemas: criar e atribuir problemas	Gerente de projetos
Gerenciamento de problemas: detecte problemas durante o teste e registre-os	Engenheiro de testes
Gerenciamento de problemas: priorize os problemas com base na gravidade e atribua-os aos desenvolvedores	Gerente de projetos
Gerenciamento de problemas: identifique e mescle problemas duplicados	Gerente de projetos
Gerenciamento de problemas: acompanhe e gere relatórios sobre os principais problemas, métricas e integridade geral do projeto	Gerente de projetos
Gerenciamento de sprint e tarefas: estime o esforço para tarefas e atribua pontos de história com base na capacidade da equipe	Mestre do Scrum
Sprint e gerenciamento de tarefas: distribua tarefas entre os membros da equipe para uniformizar a carga de trabalho em todo o sprint	Mestre do Scrum
Gerenciamento de sprint e tarefas: facilite as sessões de planejamento de sprint que alinhem os esforços da equipe às metas do sprint	Mestre do Scrum
Gerenciamento da lista de pendências do produto: reordene os itens da lista de	Proprietário do produto

Subcapacidade: caso de uso	Pessoa
pendências com base no valor comercial, na urgência e no feedback do usuário	
Gerenciamento da lista de pendências do produto: integre novos comentários de clientes e insights de mercado à lista de pendências do produto para priorização quase em tempo real	Proprietário do produto
Gerenciamento da lista de pendências do produto: identifique e gerencie dependências entre os itens da lista de pendências para agilizar o desenvolvimento	Gerente de produto
Mapeamento de histórias de usuários: crie mapas das jornadas do usuário para identificar todos os recursos necessários e suas histórias de usuário correspondentes	Proprietário do produto
Mapeamento de histórias de usuários: identifique lacunas ou etapas ausentes no fluxo de usuários	Analista de negócios
Mapeamento de histórias de usuários: priorize histórias de usuários com base em seu impacto no valor comercial	Gerente de produto
Relatórios e análises: gere painéis quase em tempo real que visualizam as principais métricas do projeto, como velocidade de sprint e taxas de resolução de problemas	Gerente de projetos
Relatórios e análises: analise dados históricos e preveja resultados futuros do projeto, como possíveis atrasos ou gargalos	Gerente de projetos

Subcapacidade: caso de uso	Pessoa
Relatórios e análises: crie relatórios personalizados, como relatórios de desempenho da equipe ou de status do projeto, personalizados para diferentes partes interessadas	Gerente de projetos
Gerenciamento do roteiro do produto: crie e mantenha um roteiro do produto que descreva os principais marcos e datas de lançamento	Gerente de projetos
Gerenciamento do roteiro do produto: atualize o roteiro com base nas mudanças nas prioridades ou nos cronogramas do projeto	Gerente de produto
Gerenciamento do roteiro do produto: compartilhe o roteiro com as partes interessadas para fornecer visibilidade sobre a direção do produto	Gerente de produto
Ciclos de feedback: colete feedback da equipe após cada sprint e identifique áreas de melhoria	Mestre do Scrum
Retrospectivas: traduza feedback em itens acionáveis para o próximo sprint, impulsionando a melhoria contínua	Mestre do Scrum
Retrospectivas: acompanhe o impacto das mudanças implementadas em retrospectivas anteriores para medir sua eficácia	Mestre do Scrum

Casos de uso generativos de IA para gerenciamento de requisitos

O gerenciamento de requisitos é um processo crítico que está intimamente ligado ao gerenciamento de projetos. Imagine um proprietário de produto usando uma ferramenta de IA para analisar o feedback do cliente, as tendências do mercado e as contribuições das partes interessadas. A ferramenta de IA pode gerar um conjunto abrangente de histórias e requisitos de usuários,

categorizá-los automaticamente, detectar possíveis conflitos ou lacunas e até mesmo sugerir priorização com base no valor comercial e na complexidade da implementação. À medida que o projeto progride e os requisitos evoluem, a IA pode atualizar e refinar continuamente os requisitos para garantir que eles permaneçam alinhados às mudanças nas necessidades comerciais e às restrições técnicas. Essa AI-driven abordagem dinâmica do gerenciamento de requisitos ajuda a garantir que os esforços de desenvolvimento permaneçam fortemente alinhados às necessidades do usuário e às metas de negócios durante todo o ciclo de vida do projeto.

A tabela a seguir mostra os casos de uso do gerenciamento de requisitos que você pode aprimorar com a IA generativa e a pessoa responsável por esses casos de uso.

Caso de uso	Pessoa
Crie requisitos de negócios	Analista de negócios
Crie épicos a partir de recursos	Proprietário do produto
Acompanhe o progresso de um épico monitorando a conclusão das histórias de usuários associadas	Gerente de produto
Crie histórias de usuários	Proprietário do produto
Estime o esforço necessário para cada história de uso e atribua pontos da história	Mestre do Scrum
Defina critérios de aceitação para cada história de usuário	Proprietário do produto

Casos de uso generativos de IA para arquitetura e design

Com uma base sólida de gerenciamento de projetos e requisitos bem definidos, a próxima capacidade crítica é a arquitetura e o design. Aqui, a IA generativa está abrindo novas possibilidades para criar arquiteturas de software robustas, escaláveis e eficientes. AI-powered as ferramentas de projeto podem analisar requisitos e restrições para sugerir padrões arquitetônicos e abordagens de design ideais. Eles geram várias alternativas de design e cada uma é otimizada para diferentes prioridades, como desempenho, escalabilidade ou capacidade de manutenção. Por exemplo, um arquiteto de soluções pode usar um assistente de IA para gerar rapidamente vários projetos

arquitetônicos de alto nível com base nos requisitos do projeto. Essa AI-augmented abordagem acelera o processo de design e ajuda os arquitetos a tomar decisões mais informadas. Isso leva a designs de software mais robustos e preparados para o futuro.

A tabela a seguir mostra os casos de uso de arquitetura e design que você pode aprimorar com a IA generativa e a pessoa responsável por esses casos de uso.

Caso de uso	Pessoa
Crie um documento de arquitetura	Arquiteto da solução
Crie um documento de design detalhado	Líder técnico
Entenda uma arquitetura e padrões de design existentes	Arquiteto da solução
Desenvolva modelos e protótipos detalhados de uma interface de usuário	UX/UI designer

Casos de uso generativos de IA para colaboração

O desenvolvimento de software é inerentemente um esforço colaborativo. Você pode usar a IA generativa para aprimorar a colaboração em sua equipe de desenvolvimento de software. AI-poweredas ferramentas de colaboração vão além da simples troca de mensagens e compartilhamento de arquivos. Eles facilitam uma comunicação mais eficaz resumindo longos tópicos de discussão, destacando as principais decisões e até sugerindo horários ideais para reuniões com base nos horários e padrões de produtividade dos membros da equipe. A IA pode ajudar nas revisões de código identificando automaticamente possíveis problemas, sugerindo melhorias e até mesmo explicando mudanças complexas aos revisores. Durante as sessões de brainstorming, a IA pode atuar como facilitadora, gerar ideias, ajudar a organizar pensamentos e até mediar discussões para garantir que todas as vozes sejam ouvidas. Para equipes distribuídas, a IA pode ajudar a superar barreiras culturais e linguísticas. Ele pode fornecer tradução de idiomas quase em tempo real em bate-papo e videochamadas e oferecer contexto cultural para ajudar a evitar mal-entendidos. Ao aumentar a colaboração humana com a IA, esse recurso ajuda as equipes a trabalhar com mais eficiência e eficácia, o que promove a inovação e melhora os resultados gerais do projeto.

A tabela a seguir mostra como você pode usar a IA generativa para aprimorar os casos de uso de colaboração.

Subcapacidade: caso de uso	Pessoa
Gerenciamento de documentos: crie e mantenha um repositório centralizado de documentação	Escritor técnico
Gerenciamento de documentos: permita que vários membros da equipe colaborem na documentação em tempo real	Desenvolvimento de equipe
Compartilhamento de conhecimento: use fóruns de discussão como uma plataforma para os desenvolvedores fazerem perguntas, compartilharem conhecimento e solucionarem problemas de forma colaborativa	Desenvolvimento de equipe
Compartilhamento de conhecimento: use fóruns de discussão para documentar e acompanhar as decisões tomadas durante as discussões do projeto, garantindo que a lógica por trás das principais decisões seja capturada e acessível para referência futura	Gerente de produto
Gerenciamento de ativos do projeto: Facilite o compartilhamento de recursos relacionados ao projeto	Desenvolvimento de equipe
Gerenciamento de ativos do projeto: implemente o controle de versão para conteúdo compartilhado para que os membros da equipe possam acompanhar as alterações, reverter para versões anteriores e colaborar nas atualizações de conteúdo	Desenvolvimento de equipe

Casos de uso generativos de IA para DevSecOps

AI-powered DevSecOps as ferramentas automatizam muitos aspectos do pipeline de entrega de software. Por exemplo, eles podem realizar análises inteligentes de código, detectar possíveis bugs, detectar vulnerabilidades de segurança e identificar problemas de desempenho quase em tempo real à medida que os desenvolvedores escrevem código. A IA gera e executa suítes de testes abrangentes e as atualiza automaticamente à medida que a base de código evolui. Essa AI-augmented abordagem DevSecOps acelera o pipeline de entrega e aumenta significativamente a segurança e a confiabilidade do software que está sendo entregue.

A tabela a seguir mostra os casos de DevSecOps uso que você pode aprimorar com a IA generativa e a pessoa responsável por esses casos de uso.

Subcapacidade: caso de uso	Pessoa
DevOps e entrega contínua: pipelines de implantação completos automatizados	DevOps engenheiro
DevOps e entrega contínua: receba feedback quase em tempo real sobre a qualidade do código e possíveis problemas	Desenvolvedor de software
DevOps e entrega contínua: receba recomendações de remediação e problemas de segurança quase em tempo real	Desenvolvedor de software
DevOps e entrega contínua: receba códigos e sugestões de melhores práticas quase em tempo real	Desenvolvedor de software
DevOps e entrega contínua: automatize tarefas repetitivas e integre comandos em scripts	DevOps engenheiro
DevOps e entrega contínua: crie código e gere artefatos automaticamente após cada confirmação de código	Desenvolvedor de software

Subcapacidade: caso de uso	Pessoa
DevOps e entrega contínua: crie código de acordo com os padrões e a estrutura da organização	Desenvolvedor de software
DevOps e entrega contínua: execute testes unitários automaticamente em cada commit para detectar erros no início do processo de desenvolvimento	Desenvolvedor de software
DevOps e entrega contínua: analise a cobertura dos testes unitários para garantir que todos os caminhos críticos do código sejam testados	Desenvolvedor de software
DevOps e entrega contínua: gerencie filiais e mescle alterações	Desenvolvedor de software
DevOps e entrega contínua: gerencie o controle de versões de códigos e artefatos	Desenvolvedor de software
DevOps e entrega contínua: armazene e gerencie dependências e artefatos de construção	DevOps engenheiro
DevOps e entrega contínua: resolva e busque dependências durante o processo de criação	Desenvolvedor de software
DevOps e entrega contínua: gere e execute testes de integração para garantir que os componentes funcionem juntos conforme o esperado	Engenheiro de testes
DevOps e entrega contínua: use serviços simulados durante os testes de integração para simular interações com sistemas externos	Engenheiro de testes

Subcapacidade: caso de uso	Pessoa
DevOps e entrega contínua: compare o desempenho do aplicativo sob diferentes cargas	Engenheiro de desempenho
DevOps e entrega contínua: simule cenários de alto tráfego para testar a escalabilidade e os tempos de resposta do aplicativo	Engenheiro de desempenho
DevOps e entrega contínua: teste a capacidade e do sistema de se recuperar de falhas, como falhas no servidor ou interrupções na rede	Engenheiro de confiabilidade do site
DevOps e entrega contínua: realize engenharia de caos	Engenheiro de confiabilidade do site
DevOps e entrega contínua: execute testes para verificar se o aplicativo atende aos requisitos de negócios	Engenheiro de QA
DevOps e entrega contínua: realizar testes de aceitação do usuário	Proprietário do produto
DevOps e entrega contínua: verifique as dependências em busca de vulnerabilidades e problemas de conformidade de licenças	Engenheiro de segurança
DevOps e entrega contínua: monitore e gerencie dependências de código aberto para garantir que elas estejam atualizadas e seguras	Engenheiro de segurança
DevOps e entrega contínua: gere e mantenha uma lista de materiais de software (SBOM) para rastrear todos os componentes e dependências	Engenheiro de segurança

Subcapacidade: caso de uso	Pessoa
DevOps e entrega contínua: use o SBOM para realizar auditorias de conformidade regulatória	Diretor de conformidade
DevOps e entrega contínua: crie notas de lançamento	Gerente de lançamento
DevOps e entrega contínua: planeje e coordene lançamentos	Gerente de lançamento
DevOps e entrega contínua: implemente procedimentos operacionais padrão para gerenciamento de reversão e liberação	Gerente de lançamento
DevOps e entrega contínua: use sinalizadores de recursos para ativar ou desativar recursos na produção sem implantar novo código	Gerente de produto
DevOps e entrega contínua: execute A/B testes usando sinalizadores de recursos para medir o impacto de diferentes recursos no comportamento do usuário	Gerente de produto
DevOps e entrega contínua: analise e monitore falhas na tubulação	DevOps engenheiro
DevOps e entrega contínua: crie e gerencie recursos de infraestrutura	DevOps engenheiro
DevOps e segurança: escaneie repositórios de código em busca de segredos codificados	DevOps engenheiro
DevOps e segurança: implemente a detecção quase em tempo real para alertar os desenvolvedores imediatamente se os segredos forem confirmados no repositório	DevOps engenheiro

Subcapacidade: caso de uso	Pessoa
DevOps e segurança: imponha o monitoramento contínuo da qualidade do código	Desenvolvedor de software
DevOps e segurança: Detecte e sinalize indicadores de possíveis vulnerabilidades de segurança no código	Desenvolvedor de software
DevOps e segurança: implemente testes automatizados para os 10 principais riscos de segurança do Open Worldwide Application Security Project (OWASP) para garantir que o aplicativo siga as práticas de segurança padrão do setor	Engenheiro de segurança
DevOps e segurança: atualize e eduque regularmente os desenvolvedores sobre os riscos do OWASP integrando verificações ao processo de desenvolvimento	Engenheiro de segurança
DevOps e segurança: escaneie bibliotecas e dependências de terceiros em busca de vulnerabilidades de segurança conhecidas	DevOps engenheiro
DevOps e segurança: escaneie o código e a infraestrutura do aplicativo para detectar vulnerabilidades	DevOps engenheiro
DevOps e segurança: analise o código em busca de vulnerabilidades antes da implantação	Engenheiro de segurança
DevOps e segurança: aplique políticas de segurança evitando que códigos com vulnerabilidades críticas sejam mesclados	Engenheiro de segurança

Subcapacidade: caso de uso	Pessoa
DevOps e segurança: implemente o controle de acesso baseado em funções (RBAC) para restringir o acesso a sistemas e dados confidenciais e garantir que somente pessoal autorizado possa acessar recursos essenciais	Engenheiro de segurança
DevOps e segurança: ajuste os controles de acesso com base nas funções e responsabilidades, adaptando-se às mudanças na estrutura da equipe	DevOps engenheiro
DevOps e segurança: teste aplicativos em execução para detectar vulnerabilidades de segurança quase em tempo real, simulando ataques ao ambiente de produção	Engenheiro de segurança
DevOps e segurança: monitore continuamente os aplicativos implantados em busca de vulnerabilidades de segurança	DevOps engenheiro
DevOps e segurança: agende verificações regulares de vulnerabilidades em todos os ambientes para identificar e resolver os pontos fracos da segurança	Engenheiro de segurança
DevOps e segurança: aplique patches e atualizações com base nos resultados da verificação de vulnerabilidades para ajudar a manter os sistemas seguros	DevOps engenheiro
Monitoramento do desempenho do aplicativo: monitore continuamente o desempenho do aplicativo quase em tempo real para detectar e diagnosticar problemas de desempenho antes que eles afetem os usuários	Engenheiro de confiabilidade do site

Subcapacidade: caso de uso	Pessoa
Monitoramento do desempenho do aplicativo: detecte anomalias de desempenho, como picos repentinos nos tempos de resposta ou maiores taxas de erro, e inicie alertas	DevOps engenheiro
Monitoramento do desempenho do aplicativo: rastreie as solicitações à medida que elas se propagam por um sistema distribuído para identificar gargalos de desempenho e problemas de latência	DevOps engenheiro
Monitoramento do desempenho do aplicativo: use o rastreamento distribuído para identificar o serviço ou componente exato responsável por falhas ou degradação do desempenho	DevOps engenheiro
Agregação e análise de registros: agregue registros de várias fontes em um sistema centralizado para facilitar a pesquisa e a análise, a fim de identificar tendências e problemas	Engenheiro de confiabilidade do site
Agregação e análise de registros: implemente a análise automatizada de registros para extrair informações relevantes e detectar padrões ou anomalias que possam indicar problemas	DevOps engenheiro
Agregação e análise de registros: colete e visualize as principais métricas de desempenho	Engenheiro de confiabilidade do site
Agregação e análise de registros: monitore métricas em relação a contratos de nível de serviço (SLAs) predefinidos	Gerente de produto

Subcapacidade: caso de uso	Pessoa
Operações de IA: detecte incidentes, analise as causas-raiz e inicie ações corretivas sem intervenção humana	DevOps engenheiro
Operações de IA: preveja futuras demandas de recursos e otimize o planejamento de capacidade para evitar interrupções	Engenheiro de confiabilidade do site
Melhoria contínua: monitore as interações reais do usuário com o aplicativo para coletar informações sobre o desempenho e identificar áreas de melhoria	Designer UX
Melhoria contínua: acompanhe o desempenho do aplicativo em diferentes regiões geográficas para garantir uma experiência de usuário consistente em todo o mundo	Gerente de produto
Monitoramento do painel: crie painéis personalizáveis para visualizar métricas, registros e rastreamentos críticos quase em tempo real, a fim de fornecer uma visão abrangente da integridade do sistema	Engenheiro de confiabilidade do site
Monitoramento de painéis: crie painéis para diferentes equipes (como equipes de desenvolvimento, operações e produtos) para fornecer informações relevantes com base em suas áreas de foco	DevOps engenheiro
Insights de desempenho: conduza uma análise detalhada do desempenho do aplicativo para identificar ineficiências e otimizar o código ou a infraestrutura	Desenvolvedor de software

Subcapacidade: caso de uso	Pessoa
Insights de desempenho: use insights de desempenho para melhorar iterativamente o desempenho do aplicativo e otimizar a experiência do usuário ao longo do tempo	Gerente de produto

Casos de uso generativos de IA para operação e manutenção

Depois que o software é implantado, o foco muda para operação e manutenção. A IA generativa pode aprimorar as abordagens tradicionais fornecendo um gerenciamento de sistema mais proativo e eficiente. AI-powered as ferramentas operacionais monitoram continuamente o desempenho do sistema e prevêm possíveis problemas antes que eles afetem os usuários. Eles realizam uma análise automatizada da causa raiz quando ocorrem problemas, o que reduz significativamente o tempo médio de resolução. A IA também otimiza o desempenho do sistema quase em tempo real. Ele ajusta automaticamente as configurações com base na alteração dos padrões de carga e nos comportamentos do usuário. Por exemplo, uma equipe de operações pode usar um assistente de IA para gerar cronogramas de manutenção preditiva, identificar automaticamente os componentes que provavelmente falharão e sugerir ações preventivas. A IA também pode ajudar no planejamento da capacidade analisando as tendências de uso e prevendo as necessidades futuras de recursos com alta precisão.

A tabela a seguir mostra os casos de uso de operação e manutenção que você pode aprimorar com a IA generativa e a pessoa responsável por esses casos de uso.

Subcapacidade: caso de uso	Pessoa
Gerenciamento de incidentes: gerencie incidentes quase em tempo real integrando ferramentas de monitoramento com plataformas de bate-papo para que as equipes possam detectar, discutir e resolver problemas diretamente no ambiente de bate-papo	Engenheiro de confiabilidade do site
Gerenciamento de incidentes: permita que as equipes iniciem implantações, executem scripts	DevOps engenheiro

Subcapacidade: caso de uso	Pessoa
e executem comandos diretamente da interface de bate-papo, o que simplifica as operações	
Atualizações de código: atualize as dependências e bibliotecas de código para reduzir o esforço manual e garantir que a base de código permaneça atualizada com as versões mais recentes	Desenvolvedor de software
Otimização de código: revise o código para oportunidades de otimização	Desenvolvedor de software
Otimização de código: identifique gargalos no código e refatore ou otimize o código para melhorar o desempenho	Desenvolvedor de software
Gestão da dívida técnica: registre a dívida técnica como parte do processo de desenvolvimento	Gerente de produto
Gerenciamento técnico da dívida: priorize e resolva a dívida técnica com base no impacto, risco e custo e integre-a ao processo regular de planejamento do sprint	Desenvolvedor de software
Gerenciamento da dívida técnica: reduza a dívida técnica no código de aplicativo existente	Desenvolvedor de software
Gerenciamento de mudanças: implemente um processo de aprovação de mudanças que garanta que todas as alterações no código sejam revisadas, testadas e aprovadas pelas partes interessadas necessárias antes da implantação	Gerente de mudanças

Subcapacidade: caso de uso	Pessoa
Gerenciamento de mudanças: realizar análise de impacto das mudanças propostas	DevOps engenheiro
Engenharia reversa: analise e compreenda a estrutura e o comportamento do código legado	Arquiteto da solução
Engenharia reversa: explique o código existente e gere documentação	Desenvolvedor de software
Modernização do código: Traduza o código de uma linguagem de programação para outra	Desenvolvedor de software
Modernização do código: modernize o código legado para a linguagem de programação mais recente	Desenvolvedor de software
Otimização do desempenho: monitore e ajuste continuamente o desempenho do sistema otimizando a alocação de recursos, o balanceamento de carga e a reconfiguração do aplicativo	Engenheiro de confiabilidade do site
Otimização de desempenho: identifique e refatore o código que está causando a degradação do desempenho para melhorar a velocidade e a capacidade de resposta do sistema	Desenvolvedor de software

Casos de uso para assistentes generativos de IA no desenvolvimento de software

A capacidade do assistente de IA está no centro da experiência de AI-powered desenvolvimento generativo. Esse sistema inteligente e sensível ao contexto serve como colaborador virtual para todos os membros da equipe em todo o SDLC. Imagine um desenvolvedor trabalhando em um código complexo. Eles podem simplesmente pedir ajuda ao assistente de IA, que pode fornecer

trechos de código relevantes, explicar algoritmos complexos ou até mesmo sugerir otimizações com base no contexto atual e nas melhores práticas. O assistente de IA pode ajudar um gerente de ITOps a entender um procedimento operacional padrão com base em documentos internos. Ao fornecer suporte instantâneo e contextual, os assistentes de IA reduzem significativamente a carga cognitiva dos membros da equipe. Isso os ajuda a se concentrar em tarefas criativas e de resolução de problemas de alto nível. Esse recurso atua como um multiplicador de forças que aumenta a produtividade e a qualidade em todos os estágios do desenvolvimento de software.

A tabela a seguir mostra casos de uso que você pode aprimorar com assistentes de IA e a persona beneficiada.

Caso de uso	Pessoa
Forneça assistência instantânea à equipe de desenvolvimento respondendo a perguntas , como sobre requisitos, arquiteturas e procedimentos operacionais padrão	Equipe de desenvolvimento de software
Pesquise ou recupere trechos de documentação extensa ou gere resumos usando consultas em linguagem natural	Equipe de desenvolvimento de software
Resuma documentos técnicos longos, como documentos de requisitos, documentações de projeto de arquitetura e processos internos	Equipe de desenvolvimento de software
Mantenha uma biblioteca de avisos que a equipe possa usar para tarefas comuns	Equipe de desenvolvimento de software
Integre perfeitamente a IA generativa às ferramentas e sistemas existentes	Equipe de desenvolvimento de software
Automatize tarefas em várias plataformas, ferramentas e sistemas internos	Equipe de desenvolvimento de software
Crie um repositório centralizado de conhecimento, incluindo melhores práticas, informações específicas do projeto e conhecimento da	Equipe de desenvolvimento de software

Caso de uso	Pessoa
equipe, acessível a todos os membros da equipe	
Recupere o conhecimento relevante do repositório com base no contexto da tarefa	Equipe de desenvolvimento de software
Realize análises automatizadas de código, análises de causa raiz, sugira melhorias, detecte possíveis bugs e solucione problemas	Desenvolvedor de software, DevOps engenheiro e engenheiro de confiabilidade do site
Analise os dados de desempenho para identificar tendências e padrões que possam informar as decisões sobre a otimização do desempenho	Engenheiro de confiabilidade do site
Forneça recomendações para melhorar a eficiência, reduzir a complexidade e aprimorar a segurança	Desenvolvedor de software
Sugira otimizações para o uso dos recursos da nuvem, como recomendações de escalabilidade ou estratégias de redução de custos	Desenvolvedor de software, DevOps engenheiro, engenheiro de confiabilidade do site e arquiteto de soluções
Gere novos conteúdos, como documentação baseada em código, guias do usuário ou lançamentos de recursos do produto	Equipe de desenvolvimento de software

Casos de uso generativos de IA para análises e insights

A capacidade de análise e insights ajuda a converter grandes quantidades de dados em insights acionáveis que impulsionam a tomada de decisões e a melhoria contínua. Ao usar IA generativa, esse recurso processa dados de várias fontes, incluindo repositórios de código, ferramentas de gerenciamento de projetos e plataformas de colaboração em equipe, para fornecer uma visão holística do processo de desenvolvimento e da produtividade da equipe. A IA generativa vai além das métricas tradicionais para oferecer análises preditivas e prescritivas. Ele pode prever possíveis problemas e sugerir melhorias específicas. Por exemplo, ele pode analisar padrões em confirmações

de código, taxas de resolução de bugs e velocidade de entrega de recursos para identificar equipes de alto desempenho, identificar gargalos e sugerir otimizações de processos. Além disso, pode fornecer informações sobre a dinâmica da equipe e o desempenho individual. Esses insights ajudam os líderes a tomar decisões baseadas em dados sobre distribuição da carga de trabalho, necessidades de treinamento e composição da equipe. Ao apresentar esses insights por meio de painéis interativos, o recurso capacita as partes interessadas em todos os níveis a tomar decisões informadas, otimizar processos e melhorar continuamente a produtividade da equipe, o que leva a uma entrega mais rápida de software de alta qualidade.

A tabela a seguir mostra os casos de uso de análise que você pode aprimorar com a IA generativa e a pessoa responsável por esses casos de uso.

Caso de uso	Pessoa
Monitore a produtividade individual e da equipe	Gerente de desenvolvimento
Analise as tendências de produtividade para detectar o potencial de esgotamento, para que você possa tomar medidas proativas para manter o bem-estar e a produtividade da equipe	Gerente de desenvolvimento
Acompanhe a frequência com que as alterações de código são implantadas na produção para avaliar a velocidade e a agilidade do processo de desenvolvimento	Gerente de produto
Analise os dados de frequência de implantação para identificar períodos de baixa atividade de implantação que possam indicar ineficiências do processo ou restrições de recursos	Gerente de produto
Meça o tempo entre a confirmação do código e a implantação para identificar oportunidades de simplificar os processos de desenvolvimento e implantação	Gerente de desenvolvimento

Caso de uso	Pessoa
Rastreie a porcentagem de implantações que resultam em falhas que exigem remediação imediata para avaliar a confiabilidade do processo de lançamento	Engenheiro de confiabilidade do site
Use métricas de taxa de falha de alteração para identificar áreas do código que frequentemente causam problemas, a fim de orientar os esforços direcionados de refatoração e teste	Desenvolvedor de software
Monitore quanto tempo é necessário para restaurar o serviço após uma interrupção ou incidente para que você possa reduzir o tempo de inatividade e melhorar a resiliência geral do sistema	Engenheiro de confiabilidade do site
Analise as tendências nos tempos de restauração para aprimorar os processos de resposta a incidentes e impulsionar uma recuperação mais rápida das falhas do sistema	DevOps engenheiro
Crie um painel personalizado que agregue as principais métricas, como frequência de implantação, prazo de entrega e taxa de falha de mudança, a fim de fornecer uma visão abrangente do desenvolvimento e da integridade operacional	Gerente de produto
Crie painéis adaptados às necessidades de diferentes equipes para fornecer informações focadas em suas áreas específicas de responsabilidade, como desenvolvimento, operações ou negócios	Gerente de produto

Caso de uso	Pessoa
Acompanhe os principais indicadores de desempenho (KPIs) de negócios, como impacto na receita, satisfação do cliente e participação no mercado, a fim de alinhar os esforços de desenvolvimento aos objetivos comerciais mais amplos	Gerente de produto
Análise o impacto de novos recursos nos KPIs de negócios para avaliar seu sucesso e orientar o desenvolvimento futuro de produtos	Analista de negócios
Monitore métricas de qualidade do código, como complexidade do código, cobertura de testes e densidade de bugs, para garantir que a base de código permaneça sustentável e segura	Desenvolvedor de software
Identifique áreas da base de código que requerem refatoração para impulsionar a sustentabilidade a longo prazo e reduzir a dívida técnica	Arquiteto da solução

Casos de uso generativos de IA para gerenciamento do conhecimento

Em qualquer organização de desenvolvimento de software, o conhecimento é um ativo essencial. A capacidade de gerenciamento do conhecimento, impulsionada pela IA generativa, aprimora a forma como esse ativo é capturado, organizado e usado. Os sistemas tradicionais de gerenciamento do conhecimento geralmente contêm muitas informações, contêm conteúdo desatualizado ou são difíceis de pesquisar para encontrar rapidamente informações relevantes.

A IA generativa aborda esses desafios de frente. Ele gera e atualiza automaticamente a documentação com base em alterações de código, conversas e artefatos do projeto. Isso garante que as bases de conhecimento permaneçam atualizadas sem exigir esforço manual dos membros da equipe. Mais importante ainda, a IA torna esse conhecimento acessível de forma intuitiva. Os membros da equipe podem fazer perguntas em linguagem natural e a IA pode fornecer respostas

relevantes. A IA pode se basear em uma variedade de fontes, como documentação oficial, comentários de código, tópicos de discussão e até mesmo recursos externos. Por exemplo, um novo membro da equipe tentando entender um componente específico pode perguntar à IA: “Como funciona o módulo de autenticação?” A IA então forneceria uma explicação concisa e links para seções de código relevantes, diagramas de arquitetura e mudanças recentes. Pode até mesmo adaptar essas informações com base na função e no nível de especialização do membro da equipe.

Esse recurso acelera a integração, reduz perguntas repetitivas e promove o compartilhamento de conhecimento em toda a organização. Isso ajuda a preservar o conhecimento institucional, facilitando que as equipes mantenham e desenvolvam sistemas complexos ao longo do tempo.

A tabela a seguir mostra os casos de uso do gerenciamento do conhecimento que você pode aprimorar com a IA generativa e a pessoa responsável por esses casos de uso.

Caso de uso	Pessoa
Crie uma plataforma unificada que facilita o acesso a todo o conhecimento relacionado ao projeto	Equipe de desenvolvimento de software
Capture conhecimento de várias atividades de desenvolvimento	Equipe de desenvolvimento de software
Forneça funcionalidade de pesquisa avançada para encontrar rapidamente conhecimentos relevantes em um repositório	Equipe de desenvolvimento de software
Personalize módulos e percursos de aprendizagem para a equipe	Equipe de desenvolvimento de software

Casos de uso generativos de IA para extensibilidade

A extensibilidade permite uma integração perfeita com ferramentas e fluxos de trabalho existentes, permitindo que as organizações adaptem o sistema de IA às suas necessidades específicas. Esse recurso fornece APIs, SDKs e interfaces personalizáveis robustos que facilitam a integração das funcionalidades de IA em ferramentas populares de desenvolvimento e gerenciamento de projetos. Por exemplo, as organizações podem aprimorar o Jira com AI-powered recursos para priorização

automática de tíquetes, estimativa de esforços e planejamento de sprint. Você pode ampliar os pipelines do Jenkins com IA para otimização inteligente de construção e seleção de testes preditivos.

Além disso, a extensibilidade permite uma integração profunda com ambientes de desenvolvimento integrados (IDEs), sistemas de controle de versão e plataformas de revisão de código. A IA pode ajudar a programar, automatizar revisões de código e gerar documentação contextual.

O recurso também oferece suporte ao treinamento e ao ajuste fino de modelos de IA em dados específicos da organização. Isso ajuda a IA a entender os padrões de codificação específicos da empresa, as preferências arquitetônicas e o conhecimento do domínio. Os resultados são uma assistência mais relevante e contextual em todas as ferramentas integradas. Ao fornecer esse nível de flexibilidade e integração, a extensibilidade garante que a experiência de AI-powered desenvolvimento evolua com a organização. Ele pode se adaptar às mudanças de tecnologias e necessidades de negócios e, ao mesmo tempo, aprimorar perfeitamente as cadeias de ferramentas e fluxos de trabalho existentes.

A tabela a seguir mostra os casos de uso de extensibilidade que você pode aprimorar com a IA generativa e a pessoa responsável por esses casos de uso.

Caso de uso	Pessoa
Integre ferramentas de terceiros ao ambiente de desenvolvimento	DevOps engenheiro
Crie fluxos de trabalho de automação personalizados, adaptados ao processo de desenvolvimento exclusivo da equipe	DevOps engenheiro
Conecte-se a várias APIs e serviços	DevOps engenheiro
Crie conectores para ferramentas multiplataforma	DevOps engenheiro

Melhores práticas para usar IA generativa no desenvolvimento de software

Esta seção descreve as melhores práticas para integrar a IA generativa no ciclo de vida de desenvolvimento de software (SDLC). Desde a implementação de cadeias de ferramentas e DevSecOps pipelines contínuos até a promoção da colaboração e a automação de tarefas repetitivas, essas diretrizes ajudam você a aproveitar o poder da IA para aprimorar seus processos e experiências de desenvolvimento. Ao seguir essas melhores práticas, as equipes de desenvolvimento de software podem desbloquear novos níveis de eficiência, inovação e qualidade em seu trabalho.

Esta seção discute as seguintes melhores práticas:

- [Implementando uma cadeia de ferramentas perfeita e end-to-end integrada](#)
- [Implementando um end-to-end CI/CD pipeline para DevSecOps](#)
- [Adotando ferramentas e práticas colaborativas](#)
- [Automatizando tarefas repetitivas](#)
- [Revisando e iterando regularmente a experiência de desenvolvimento](#)
- [Adotando práticas eficazes de gerenciamento de projetos](#)
- [Implementando a gestão do](#)
- [Fornecendo extensibilidade e personalização](#)
- [Otimização para operações](#)
- [Usando insights baseados em dados](#)
- [Adotando uma abordagem baseada em plataforma](#)

Implementando uma cadeia de ferramentas perfeita e end-to-end integrada

A implementação de uma cadeia de ferramentas perfeita e end-to-end integrada é uma prática recomendada fundamental para criar uma experiência de desenvolvimento generativa baseada em IA. A ideia principal é estabelecer um ecossistema coeso de ferramentas e plataformas que suas equipes de software possam usar em todo o SDLC. A equipe pode usar o conjunto de ferramentas para planejar, idealizar, codificar, criar, testar, implantar e gerenciar operações contínuas. Ao integrar

recursos generativos de IA a essa cadeia de ferramentas, você garante que a assistência de IA esteja disponível em todas as etapas. Essa integração reduz ou elimina as transferências manuais, reduz a troca de contexto e ajuda os dados e artefatos a fluírem sem problemas entre as diferentes fases de desenvolvimento. Por exemplo, trechos de código gerados por IA de seu ambiente de desenvolvimento integrado (IDE) podem fluir perfeitamente para seu sistema de controle de versão, e análises baseadas em IA de sua plataforma de implantação podem informar suas ferramentas de gerenciamento de projetos. Isso cria um ciclo de feedback contínuo que melhora seu processo de desenvolvimento.

Implementando um end-to-end CI/CD pipeline para DevSecOps

Para aproveitar essa cadeia de ferramentas integrada, implementar uma integração end-to-end contínua e uma implantação contínua (CI/CD) pipeline for DevSecOps. This AI-powered pipeline is a critical component that streamlines your software delivery processes. It helps you release new applications and updates more quickly and reliably. By embedding security practices throughout the entire SDLC, you can identify and address vulnerabilities much earlier, which reduces the overall cost and risk. The pipeline should incorporate AI at every stage, from continuous integration and testing to security checks and deployment. For instance, you can use AI to analyze code commits in near real time so that you can predict potential integration issues before they occur. In the CI/CD pipeline), você também pode usar a IA generativa para atualizar automaticamente as políticas de segurança com base na inteligência de ameaças mais recente.

Adotando ferramentas e práticas colaborativas

Ao aprimorar sua infraestrutura de desenvolvimento, não se esqueça do elemento humano. O desenvolvimento de software é inerentemente um esforço colaborativo. Envolve equipes multifuncionais compostas por desenvolvedores, designers, gerentes de produto, Scrum Masters, analistas de negócios e outras partes interessadas. Essas pessoas trabalham coletivamente para concretizar ideias. Ao usar ferramentas colaborativas modernas e promover uma cultura de comunicação aberta e compartilhamento de conhecimento, você pode aumentar significativamente a produtividade e a eficácia de suas equipes de desenvolvimento de software. Em sua experiência de desenvolvimento de software com inteligência artificial, essas ferramentas assumem novas dimensões. Você pode integrar a IA às plataformas de colaboração para facilitar uma comunicação mais eficaz e o compartilhamento de conhecimento entre os membros da equipe. Os assistentes de IA podem responder perguntas comuns, resumir discussões ou até mesmo mediar conflitos. A IA generativa pode aprimorar os processos de revisão de código sugerindo melhorias automaticamente

ou identificando possíveis problemas. Além disso, você pode usar a IA para criar documentação dinâmica e contextual que é atualizada quase em tempo real à medida que o projeto evolui, para que todos os membros da equipe tenham acesso às informações mais atuais e relevantes.

Automatizando tarefas repetitivas

Ao usar a IA generativa para lidar com atividades rotineiras e demoradas, você libera suas equipes de software para se concentrarem em trabalhos criativos e de alto valor que impulsionam a inovação e geram impacto nos negócios. Exemplos de tarefas repetitivas incluem gerar código padronizado, criar dados de teste, escrever documentação ou até mesmo elaborar planos iniciais do projeto. Ao transferir essas tarefas para a IA, os membros da equipe podem se concentrar em um trabalho mais criativo e estratégico. Por exemplo, ferramentas de preenchimento de código baseadas em IA podem acelerar significativamente o processo de codificação, sugerindo trechos de código relevantes com base no contexto e nos padrões de codificação. Da mesma forma, a IA generativa pode criar e atualizar automaticamente a documentação técnica à medida que o código muda. Isso mantém a documentação atualizada e reduz o esforço manual normalmente necessário para essa tarefa. Nos testes, a IA pode gerar casos de teste abrangentes com base em requisitos e análise de código, o que melhora a cobertura do teste e reduz a probabilidade de casos extremos negligenciados. Ao automatizar de forma inteligente essas tarefas repetitivas, a IA generativa acelera os cronogramas de desenvolvimento, melhora a consistência e reduz o erro humano. O resultado são saídas de software de maior qualidade.

Revisando e iterando regularmente a experiência de desenvolvimento

Sua experiência em desenvolvimento de software em si deve ser tratada como um produto que requer aperfeiçoamento contínuo. Isso envolve o estabelecimento de um processo sistemático para revisar e iterar regularmente todos os aspectos do ciclo de vida, das ferramentas e das práticas de desenvolvimento. Realize avaliações periódicas de toda a cadeia de ferramentas, fluxos de trabalho e processos. Obtenha feedback de todos os membros da equipe em várias funções, incluindo gerentes de produto, designers, arquitetos, desenvolvedores, testadores e equipe de operações. Peça-lhes que identifiquem pontos problemáticos, gargalos e oportunidades de aprimoramento. Por exemplo, as equipes podem realizar análises trimestrais do desempenho do CI/CD pipeline e analisar métricas como tempos de construção, frequência de implantação e taxas de erro para identificar áreas de otimização. Como os recursos de IA generativa continuam evoluindo rapidamente, é crucial avaliar consistentemente novas ferramentas e recursos baseados em IA que

possam simplificar ainda mais os fluxos de trabalho ou aumentar os recursos em todas as funções no SDLC.

Adotando práticas eficazes de gerenciamento de projetos

Para orquestrar seus esforços complexos de desenvolvimento de software de forma eficaz, adote práticas de gerenciamento de projetos com IA aumentada. Nesse contexto, o gerenciamento eficaz de projetos vai além das metodologias tradicionais. Ele adota abordagens aumentadas por IA que aprimoram o planejamento, a execução e o monitoramento em todo o SDLC. As estruturas ágeis promovem flexibilidade, colaboração e iteração rápida, e você pode usar a IA generativa para otimizar esses processos. Por exemplo, a IA generativa pode analisar dados históricos do projeto para obter estimativas mais precisas, gerar e priorizar automaticamente histórias de usuários com base nos objetivos comerciais e nos comentários dos clientes, além de fornecer informações inteligentes sobre o desempenho da equipe. As ferramentas de gerenciamento de projetos baseadas em IA podem prever possíveis obstáculos e sugerir atribuições de tarefas ideais com base nas habilidades e cargas de trabalho dos membros da equipe. Ao integrar recursos baseados em IA às práticas de gerenciamento de projetos, você pode obter maior visibilidade, tomar decisões baseadas em dados com mais rapidez e garantir que os membros da equipe estejam alinhados e trabalhando com eficiência em direção a metas comuns.

Implementando a gestão do

À medida que sua experiência de desenvolvimento de software com inteligência artificial amadurece, implemente um sistema robusto de gerenciamento de conhecimento. Um sistema robusto de gerenciamento de conhecimento ajuda você a capturar, organizar e conceder acesso a informações valiosas, melhores práticas e soluções. Todos os membros da equipe em todo o SDLC devem ter acesso fácil ao sistema. Use a IA generativa para criar bases de conhecimento dinâmicas e inteligentes que evoluem com sua organização. Por exemplo, a IA pode gerar e atualizar automaticamente a documentação com base em alterações de código, conversas e artefatos do projeto para que as informações permaneçam atualizadas sem intervenção manual. A IA generativa também pode potencializar recursos de pesquisa inteligente e ajudar os membros da equipe a encontrar rapidamente informações relevantes usando consultas em linguagem natural, mesmo que não saibam a terminologia exata. Além disso, a IA generativa pode apresentar proativamente informações relevantes aos membros da equipe com base em suas tarefas ou desafios atuais. Ele atua como um mentor virtual que aprimora a tomada de decisões e a resolução de problemas em todas as funções. Ao implementar um sistema de gerenciamento de conhecimento baseado em

IA, você pode eliminar silos, acelerar a integração, reduzir o trabalho redundante e promover uma cultura de aprendizado e inovação contínuos em toda a sua equipe de desenvolvimento de software.

Fornecendo extensibilidade e personalização

Para maximizar os benefícios da IA generativa no desenvolvimento de software, certifique-se de que suas ferramentas e plataformas baseadas em IA sejam extensíveis e personalizáveis. Isso ajuda você a adaptar os recursos de IA às suas necessidades específicas, fluxos de trabalho e pilhas de tecnologia. Por exemplo, você pode ajustar os modelos de IA em suas próprias bases de código e documentação, criar ferramentas personalizadas com inteligência artificial para tarefas específicas ou integrar recursos de IA às ferramentas e processos existentes. Essa extensibilidade ajuda você a desenvolver a experiência de desenvolvimento baseada em IA para atender às necessidades em constante mudança da organização. Também ajuda a otimizar a experiência para domínios ou tipos de projetos específicos.

Otimização para operações

A IA generativa desempenha um papel crucial na otimização das operações e manutenção do software. Otimize as operações integrando os recursos de IA às suas ferramentas e processos operacionais. Por exemplo, use IA generativa para analisar dados de log quase em tempo real, prever possíveis falhas no sistema e automatizar tarefas de manutenção de rotina. A IA generativa também pode ajudar na análise da causa raiz, correlacionando eventos em sistemas distribuídos complexos. Isso melhora a confiabilidade do sistema, reduz o tempo de inatividade e libera suas equipes de operações para se concentrarem em iniciativas mais estratégicas.

Usando insights baseados em dados

Use insights baseados em dados em toda a sua jornada de desenvolvimento com inteligência artificial. Implemente sistemas para coletar, analisar e agir com base nos dados de todos os estágios do SDLC. Isso inclui métricas de código, resultados de testes, dados de implantação, feedback do usuário e desempenho operacional. Use a IA generativa para descobrir padrões e insights que podem não ser aparentes para observadores humanos. Em seguida, insira esses insights em seu processo de desenvolvimento para informar tudo, desde decisões de arquitetura até a priorização de recursos.

Adotando uma abordagem baseada em plataforma

Para aproveitar plenamente os benefícios da IA generativa no desenvolvimento de software, adote uma abordagem baseada em plataforma. Crie uma plataforma abrangente e integrada que incorpore recursos de IA em todos os aspectos do SDLC. A plataforma deve fornecer uma experiência de usuário consistente, gerenciamento e dados centralizados e integração perfeita entre diferentes ferramentas e processos. Isso torna os benefícios da IA disponíveis uniformemente em toda a organização, reduz a sobrecarga do gerenciamento de várias e diferentes ferramentas de IA e fornece uma base para a melhoria contínua e a expansão dos recursos de IA.

Medindo o sucesso da IA generativa no desenvolvimento de software

Para medir com eficácia o efeito da implementação de uma experiência generativa de desenvolvimento de software baseada em IA, você precisa estabelecer um conjunto abrangente de métricas que abranja várias dimensões do seu ciclo de vida de desenvolvimento de software (SDLC). Essas métricas devem capturar melhorias imediatas em eficiência e produtividade e também refletir ganhos de longo prazo na qualidade do software, na satisfação da equipe e no valor comercial.

Faça o seguinte para usar com eficácia as métricas recomendadas nesta seção:

1. Estabeleça linhas de base — Antes de começar a implementar sua experiência de desenvolvimento baseada em IA, reserve um tempo para coletar dados abrangentes sobre seu desempenho atual em todas essas métricas. Isso fornece um ponto de partida claro e ajuda você a fazer comparações significativas posteriormente.
2. Estabeleça metas realistas — Com suas linhas de base em mãos, defina metas de melhoria alcançáveis para cada métrica. Seja ambicioso, mas realista. Lembre-se de que o progresso sustentável geralmente é incremental.
3. Implemente monitoramento contínuo — Use ferramentas automatizadas para coletar e analisar constantemente dados dessas métricas em seu ambiente. O monitoramento quase em tempo real ajuda você a monitorar o progresso e identificar rapidamente quaisquer problemas ou oportunidades.
4. Conduza revisões regulares — Agende sessões de revisão trimestrais ou semestrais em que você e sua equipe avaliem minuciosamente seu progresso em relação às metas. Use essas sessões para identificar áreas que precisam ser melhoradas ainda mais e comemorar seus sucessos.
5. Repita e ajuste — Com base nos insights que você obteve, refine continuamente sua implementação generativa de IA e ajuste as metas conforme necessário.

Esta seção descreve as seguintes categorias de métricas:

- [Velocidade de implantação](#)
- [Qualidade do código](#)
- [Eficiência operacional](#)
- [Produtividade e satisfação da equipe](#)

- [Impacto nos negócios](#)

Velocidade de implantação

Considere medir as seguintes métricas de velocidade de implantação.

Métrica	Description
Hora de comercializar	Meça a redução no tempo desde a concepção da ideia até a implantação da produção
Velocidade de sprint	Acompanhe o aumento nos pontos de história concluídos por sprint por suas equipes
Frequência de confirmação de código	Monitore o aumento nas confirmações de código, o que indica a aceleração dos ciclos de desenvolvimento
Tempo de resolução do pull request	Avalie a diminuição do tempo necessário para revisar e mesclar as alterações de código em seus repositórios
Velocidade de liberação	Meça o aumento no número de lançamentos por trimestre ou ano

Qualidade do código

Considere medir as seguintes métricas de qualidade de código.

Métrica	Description
Densidade do defeito	Meça a redução de bugs de software
Cobertura de código	Acompanhe o aumento na porcentagem de cobertura de testes em toda a sua base de código

Métrica	Description
Dívida técnica	Monitore a diminuição da dívida técnica identificada ao longo do tempo
Pontuações de análise de código estático	Avalie as melhorias na qualidade do código com base em suas ferramentas de análise automatizada

Eficiência operacional

Considere medir as seguintes métricas de eficiência operacional.

Métrica	Description
Frequência de implantação	Meça o aumento no número de implantações bem-sucedidas
Tempo médio de recuperação (MTTR)	Acompanhe a redução no tempo necessário para se recuperar de falhas no sistema
Alterar a taxa de falha	Monitore a diminuição na porcentagem de alterações que resultam em falhas em suas implantações

Produtividade e satisfação da equipe

Considere medir as seguintes métricas de produtividade e satisfação da equipe.

Métrica	Description
Melhoria da produtividade	Monitore o aumento na porcentagem de produtividade de cada tarefa

Métrica	Description
Pontuação de satisfação	Realize pesquisas regulares para avaliar a melhoria no moral e na satisfação profissional de sua equipe
Eficiência no compartilhamento de conhecimento	Meça a redução no tempo que sua equipe gasta pesquisando informações ou fazendo perguntas repetitivas
Horário de embarque	Acompanhe a diminuição do tempo necessário para que novos membros da equipe se tornem produtivos

Impacto nos negócios

Considere medir as seguintes métricas de impacto nos negócios.

Métrica	Description
Taxa de adoção de recursos	Avalie o aumento no engajamento do usuário com os novos recursos que você lançou
Pontuação de satisfação do cliente	Acompanhe as melhorias nos comentários e avaliações de seus usuários
Impacto na receita (direto e indireto)	Avalie o aumento na receita atribuído ao aumento da velocidade de lançamento ou ao aumento da produtividade

Conclusão

Este documento de estratégia fornece uma visão geral de uma experiência generativa de desenvolvimento de software com inteligência artificial. Ele explora as cinco dimensões da [estrutura 5-I](#) — Investigar, Integrar, Interagir, Iterar e Impactar. Essas dimensões fornecem um roteiro estratégico para integrar a IA generativa em todo o ciclo de vida de desenvolvimento de software (SDLC). Também descreve os [recursos fundamentais](#) necessários para implementar com êxito essa estrutura. Os recursos abrangem áreas como gerenciamento de projetos DevSecOps, assistentes de IA, gerenciamento de conhecimento e muito mais. Ele fornece [as melhores práticas](#) a serem consideradas ao integrar a IA generativa e ajuda você a usar [métricas](#) para medir o impacto que a IA generativa tem na sua experiência de desenvolvimento de software.

A integração da IA generativa nos processos de desenvolvimento de software representa uma mudança de paradigma que tem o potencial de acelerar a inovação, melhorar a qualidade e aumentar a produtividade. No entanto, é importante reconhecer que essa não é uma implementação única. É uma evolução contínua que requer esforço sustentado e refinamento contínuo.

Ao embarcar nessa jornada, recomendamos que você comece com uma avaliação completa das capacidades e prontidão atuais de sua organização. A [Ferramenta de AWS Avaliação](#) é uma ferramenta de avaliação de desenvolvimento de software baseada em IA que pode ajudá-lo a identificar áreas prioritárias e criar um roteiro de implementação personalizado.

Recursos

Depois de identificar as principais áreas prioritárias, os seguintes recursos podem ajudá-lo a implementar seu roteiro:

AWS documentação

- [Automatize as operações de AWS infraestrutura usando o Amazon Bedrock \(orientação AWS prescritiva\)](#)
- [Melhores práticas com o Amazon Q Developer para geração de código em linha e assistente \(orientação AWS prescritiva\)](#)
- [Desenvolva um assistente baseado em bate-papo totalmente automatizado usando agentes e bases de conhecimento do Amazon Bedrock \(AWS orientação prescritiva\)](#)
- [Transformando modelos operacionais de desenvolvimento e manutenção de aplicativos AWS com IA generativa \(orientação AWS prescritiva\)](#)

- [Use o Amazon Q Developer como assistente de codificação para aumentar sua produtividade](#) (orientação AWS prescritiva)

AWS postagens de blog e tutoriais

- [Publicações no blog da Amazon Q](#)
- [Acelere seu ciclo de vida de desenvolvimento de software com o Amazon Q](#) (AWS publicação no blog)
- [Criando um agente de IA AWS para arquitetos de soluções: aproveitando o Amazon Bedrock para arquitetura e implantação automatizadas \(vídeo\)](#) AWS
- [Operações generativas de tecnologia alimentadas por IA](#) (AWS postagem no blog)
- [Modernize seu aplicativo Java com o Amazon Q Developer](#) (publicação AWS no blog)
- [Use o Amazon Bedrock para gerar, avaliar e entender o código em seu pipeline de desenvolvimento de software](#) (postagem no AWS blog)

Histórico do documento

A tabela a seguir descreve alterações significativas feitas neste guia. Se desejar receber notificações sobre futuras atualizações, inscreva-se em um [feed RSS](#).

Alteração	Descrição	Data
Publicação inicial	Não aplicável	18 de abril de 2025

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.