



Accelerazione dei cicli di vita dello sviluppo del software con l'IA generativa
AWS

AWS Linee guida prescrittive



AWS Linee guida prescrittive: Accelerazione dei cicli di vita dello sviluppo del software con l'IA generativa AWS

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

I marchi e l'immagine commerciale di Amazon non possono essere utilizzati in relazione a prodotti o servizi che non siano di Amazon, in una qualsiasi modalità che possa causare confusione tra i clienti o in una qualsiasi modalità che denigri o discrediti Amazon. Tutti gli altri marchi non di proprietà di Amazon sono di proprietà dei rispettivi proprietari, che possono o meno essere affiliati, collegati o sponsorizzati da Amazon.

Table of Contents

Introduzione	1
Obiettivi	1
Destinatari principali	2
Esperienza di sviluppo	3
Utilizzo dell'intelligenza artificiale generativa	4
Framework 5-I	5
Panoramica del framework	6
Integrazione con SDLC	8
Funzionalità fondamentali	9
Gestione progettuale	16
Gestione dei requisiti	19
Architettura e design	20
Collaborazione	21
DevSecOps	22
Funzionamento e manutenzione	30
Assistenti AI	33
Analisi dei dati e informazioni dettagliate	35
Gestione delle conoscenze	38
Estensibilità	39
Best practice	41
Toolchain integrata	41
DevSecOps oleodotto	42
Strumenti e pratiche collaborative	42
Automazione delle attività	43
Revisione e iterazione	43
Pratiche di gestione dei progetti	44
Gestione delle conoscenze	44
Estensibilità e personalizzazione	45
Ottimizzazione	45
Informazioni basate sui dati	45
Approccio basato sulla piattaforma	46
Misurare il successo	47
Velocità di distribuzione	48
Qualità del codice	48

Efficienza operativa	49
Produttività e soddisfazione del team	49
Impatto aziendale	50
Conclusioni	51
Risorse	51
Cronologia dei documenti	53
.....	liv

Accelerazione dei cicli di vita dello sviluppo del software con l'IA generativa AWS

Chetan Makvana, Amazon Web Services

Aprile 2025 (cronologia del documento)

La crescente domanda di software di alta qualità spinge le organizzazioni a cercare costantemente modi per accelerare il ciclo di vita dello sviluppo del software (SDLC). Poiché le organizzazioni si sforzano di rimanere competitive, è fondamentale ridurre i tempi di commercializzazione mantenendo o migliorando al contempo la qualità dei prodotti. Per affrontare queste sfide, l'esperienza di sviluppo del software deve evolversi e utilizzare tecnologie, metodologie e pratiche all'avanguardia che semplifichino i processi e consentano ai team di sviluppo software di essere più produttivi e creativi. L'emergere dell'esperienza di sviluppo di nuova generazione segna un cambiamento significativo nel modo in cui il software viene concepito, creato, testato e distribuito. Integra una varietà di funzionalità, tra cui lo sviluppo nativo del cloud, l'automazione basata sull'intelligenza artificiale, la gestione avanzata dei progetti, gli strumenti collaborativi e, collettivamente, migliorano l'efficienza e l'efficacia dell'SDLC. DevSecOps

In prima linea in questa trasformazione c'è l'ascesa dell'intelligenza artificiale generativa nell'ingegneria del software. Secondo [Gartner](#), il 40% dei team di progettazione della piattaforma utilizzerà l'intelligenza artificiale per potenziare ogni fase dell'SDLC entro il 2027, rispetto a solo il 5% nel 2023. Questo rapporto afferma inoltre che i leader dell'ingegneria del software devono ora prepararsi ad adottare l'IA generativa in una gamma più ampia di aree fondamentali per il processo di sviluppo. In un altro rapporto, una [McKinsey](#) ricerca mostra che le aziende con un indice di velocità di sviluppo più elevato aumentano i ricavi 4-5 volte più velocemente, ottengono un rendimento per gli azionisti superiore del 60% e sono più innovative del 55%. Adottando l'intelligenza artificiale generativa oltre alla semplice generazione di codice, le organizzazioni possono sbloccare nuovi livelli di efficienza, produttività e innovazione nei flussi di lavoro di sviluppo software. Ciò può ridurre lo sforzo manuale, far emergere le informazioni e aumentare le competenze umane.

Obiettivi

Questo documento strategico delinea un framework, funzionalità fondamentali, casi d'uso, best practice e metriche di successo che possono aiutarvi ad accelerare il vostro SDLC con l'intelligenza

artificiale generativa. Descrive come integrare efficacemente l'IA generativa in tutte le fasi di sviluppo al fine di migliorare la qualità e l'efficienza del prodotto.

Questo documento strategico può aiutare te e la tua organizzazione a raggiungere i seguenti obiettivi:

- Implementate un framework, funzionalità di base, casi d'uso, best practice e metriche di successo per accelerare il vostro SDLC con l'intelligenza artificiale generativa.
- Integra efficacemente l'intelligenza artificiale generativa in tutte le fasi di sviluppo per migliorare la qualità del prodotto, la velocità di rilascio e l'efficienza dello sviluppo.
- Adattati alla nuova generazione di sviluppo software incorporando tecnologie, metodologie e pratiche di intelligenza artificiale all'avanguardia che semplificano i processi e potenziano i team di sviluppo.

Destinatari principali

Questo documento strategico è destinato ai leader IT, ai responsabili tecnici, ai Chief Technology Officer e ai team di sviluppo software che desiderano accelerare il ciclo di vita dello sviluppo del software applicando l'intelligenza artificiale generativa alle loro pratiche di sviluppo.

Comprendere l'esperienza di sviluppo del software

L'esperienza di sviluppo software comprende l'ambiente, gli strumenti e i processi utilizzati dai team di sviluppo durante l'intero ciclo di vita dello sviluppo del software (SDLC). Include l'ambiente di sviluppo integrato (IDE), le piattaforme di collaborazione, i framework di test, i sistemi di gestione della conoscenza, le pipeline di implementazione e altro ancora.

Un'esperienza di sviluppo ben progettata semplifica i flussi di lavoro, riduce lo sforzo manuale e consente ai team di concentrarsi su attività di alto valore, il che, in ultima analisi, accelera il vostro SDLC. Ad esempio, integrando perfettamente l'IDE, il sistema di controllo delle versioni e gli strumenti di distribuzione, consentite agli sviluppatori di scrivere, testare e distribuire codice con maggiore velocità ed efficienza rispetto a una toolchain frammentata che richiede passaggi manuali e cambio di contesto. Allo stesso modo, l'integrazione di un solido framework di gestione della conoscenza aiuta i team ad accedere e condividere facilmente le conoscenze, le migliori pratiche e la documentazione istituzionali. Ciò migliora la loro produttività complessiva e le capacità di risoluzione dei problemi.

L'esperienza di sviluppo software ha un impatto diretto sulle prestazioni complessive e sul successo di un team di sviluppo software. Un'esperienza non ottimale può portare a quanto segue:

- **Produttività ridotta:** strumenti inefficienti, flussi di lavoro complessi e mancanza di automazione ostacolano la produttività del team, rallentando la fornitura di funzionalità e aggiornamenti.
- **Aumento del debito tecnico:** strumenti e processi ad hoc scarsamente integrati possono causare debiti tecnici, il che rende più difficile la manutenzione e la scalabilità dei sistemi software nel tempo.
- **Innovazione ridotta:** quando è bloccato da attività manuali e ripetitive, la capacità del team di esplorare nuove tecnologie e promuovere l'innovazione è limitata.
- **Qualità compromessa:** i processi di test e implementazione frammentati aumentano il rischio di difetti e vulnerabilità del software. Ciò può influire negativamente sulla qualità complessiva del software fornito.

Investendo in un'esperienza di sviluppo software ben progettata, è possibile ottenere vantaggi significativi, come un time-to-market più rapido, una migliore qualità del software, una maggiore soddisfazione del team di sviluppo software e una maggiore agilità aziendale.

Potenziare l'esperienza di sviluppo software con l'intelligenza artificiale generativa

L'integrazione dell'IA generativa nel ciclo di vita dello sviluppo del software (SDLC) rappresenta un cambio di paradigma nel modo in cui interi team di sviluppo software concepiscono, progettano, implementano e gestiscono le soluzioni software. L'intelligenza artificiale generativa ha il potenziale per rivoluzionare ogni fase dell'SDLC, compresa la gestione dei progetti, la raccolta dei requisiti, la progettazione, la codifica, il test, l'implementazione e la manutenzione.

Fondamentalmente, un'esperienza di AI-powered sviluppo generativo funge da collaboratore intelligente per l'intero team di sviluppo software, inclusi responsabili di prodotto, designer, architetti di soluzioni, sviluppatori, tester e personale operativo. Fornisce assistenza in base al contesto, genera artefatti (come storie utente, prototipi di progettazione, frammenti di codice e casi di test), offre suggerimenti quasi in tempo reale e prevede persino potenziali problemi prima che si verifichino. Questo AI-augmented approccio riduce significativamente il carico cognitivo sui membri del team. Ciò consente loro di concentrarsi su decisioni strategiche di alto livello e sulla risoluzione di problemi complessi, mentre l'intelligenza artificiale generativa gestisce le attività più banali e ripetitive.

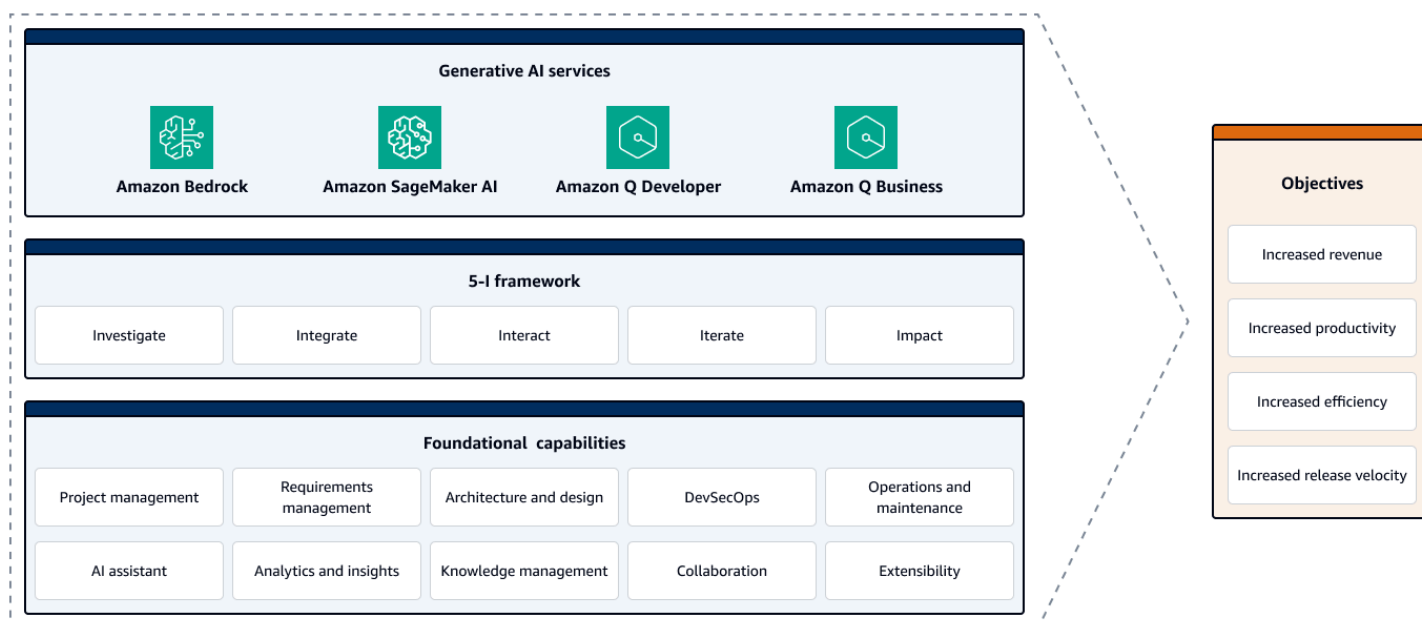
L'intelligenza artificiale generativa funge anche da amplificatore di conoscenza. Aiuta i membri del team ad accedere rapidamente alle informazioni, alle migliori pratiche e ai modelli pertinenti da vasti archivi di dati. Ciò può democratizzare efficacemente le competenze in tutta l'organizzazione. Integrando perfettamente le funzionalità di intelligenza artificiale generativa in tutta la toolchain di sviluppo, puoi creare un ambiente più intuitivo, efficiente e produttivo per tutti i tuoi team di sviluppo software. Questa esperienza di sviluppo avanzata accelera l'SDLC e migliora la qualità generale. Riduce inoltre gli errori e favorisce l'innovazione perché i membri del team possono esplorare nuove idee e approcci più rapidamente.

Per adottare un'esperienza di AI-powered sviluppo generativo nella tua organizzazione, considera i seguenti elementi chiave:

- [Framework 5-I](#)— Composto da cinque dimensioni, il framework 5-I offre un approccio completo per orientarsi nel processo di sviluppo del software moderno. Offre una metodologia strutturata che consente di applicare sistematicamente l'IA generativa in tutte le fasi dell'SDLC.
- [Funzionalità fondamentali](#)— Per sfruttare appieno la potenza dell'intelligenza artificiale generativa in tutte le dimensioni dello sviluppo software moderno, è necessario stabilire un solido set di funzionalità di base. Queste funzionalità costituiscono la spina dorsale di un' AI-powered

esperienza di sviluppo. Queste funzionalità consentono di integrare e utilizzare l'IA generativa in tutto l'SDLC.

Insieme, il framework 5-I e le funzionalità di base costituiscono una strategia per reinventare l'esperienza di sviluppo del software. Le cinque dimensioni forniscono un quadro strategico per l'applicazione dell'IA generativa e le funzionalità di base preparano l'organizzazione a supportare questo approccio. AI-driven Servizi AWS, come [Amazon Bedrock](#), [Amazon SageMaker AI](#), [Amazon Q Developer](#) e [Amazon Q Business](#), offrono funzionalità e caratteristiche di intelligenza artificiale generativa che puoi integrare nella tua esperienza di sviluppo software.



Framework 5-I per un' AI-powered esperienza di sviluppo software

Il framework 5-I offre un approccio strutturato ai team di sviluppo software per integrare efficacemente l'IA generativa nelle loro pratiche di sviluppo. Ti aiuta a stabilire una solida base per l'utilizzo dell'IA generativa in tutto l'SDLC. Inoltre, consente di impostare le pratiche di sviluppo, i flussi di lavoro e la mentalità corretti per sfruttare appieno il potenziale dell'IA generativa.

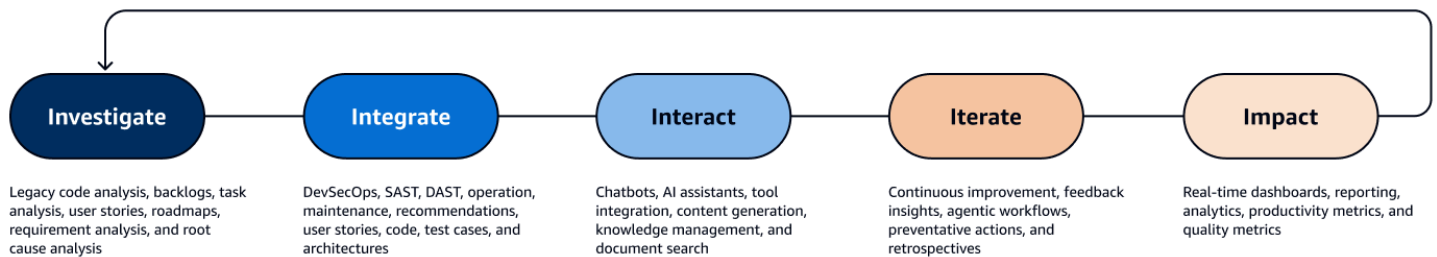
Questa sezione contiene i seguenti argomenti:

- [Panoramica del framework](#)
- [Integrazione con il ciclo di vita dello sviluppo del software](#)

Panoramica del framework

Il framework 5-I è costruito attorno a cinque dimensioni chiave: Investigate, Integrate, Interact, Iterate e Impact. Ogni dimensione rappresenta un'area critica in cui l'intelligenza artificiale generativa migliora in modo significativo il processo di sviluppo del software. Integrando strategicamente l'IA generativa in queste dimensioni, il framework risponde alle esigenze in evoluzione dello sviluppo di software moderno. Può ridurre il carico cognitivo e amplificare il potenziale creativo. Riconosce che l'esperienza di sviluppo ideale non riguarda solo gli strumenti, ma anche la creazione di un ambiente in cui l'intelligenza artificiale migliori senza soluzione di continuità le capacità umane in ogni fase.

Il diagramma seguente mostra le cinque dimensioni dello sviluppo del software. AI-powered Per ogni dimensione, mostra dove è possibile integrare l'IA generativa per promuovere l'efficienza e l'innovazione.



Le cinque dimensioni del framework sono le seguenti:

- **Indagate:** migliorate ogni attività analitica del processo di sviluppo del software con l'intelligenza artificiale generativa. Usa l'intelligenza artificiale generativa per comprendere i requisiti, elaborare grandi quantità di dati, riconoscere modelli e generare informazioni che potrebbero andare oltre le capacità umane o richiedere molto più tempo per essere prodotte. Queste informazioni ti aiutano a prendere decisioni più informate, identificare rapidamente opportunità di miglioramento e fornire in modo più efficiente software di alta qualità. L'intelligenza artificiale generativa può essere un partner intelligente per i processi analitici in tutto l'SDLC. Sfruttando l'intelligenza artificiale generativa, è possibile applicare un'analisi approfondita alle aree critiche, come la raccolta dei requisiti, l'esame della base di codice esistente e l'ottimizzazione del portafoglio di prodotti. Ad esempio, i proprietari dei prodotti possono utilizzare l'intelligenza artificiale generativa per analizzare i percorsi o i requisiti degli utenti prima di creare storie utente. I team di sviluppo possono scoprire le inefficienze e identificare opportunità di ottimizzazione nelle basi di codice esistenti. DevOps gli ingegneri possono applicare l'analisi delle cause principali per diagnosticare rapidamente problemi di prestazioni o vulnerabilità di sicurezza, il che può migliorare l'affidabilità.

- **Integrazione:** integra l'intelligenza artificiale generativa per automatizzare un'ampia gamma di attività e processi nell'intero SDLC. Ciò include la generazione automatica di frammenti di codice, casi di test, progetti architettonici, storie utente e pipeline di implementazione. Automatizzando queste attività tipicamente manuali, i team possono concentrarsi su attività più strategiche e innovative, il che consente un time-to-market più rapido e applicazioni di alta qualità. La dimensione Integrate rappresenta un cambio di paradigma nello sviluppo del software, in cui l'IA diventa parte integrante del processo di sviluppo. Collabora con il team di sviluppo software per migliorare la produttività, migliorare la qualità e promuovere l'innovazione. Ciò si traduce in un time-to-market più rapido. Sfida i team di sviluppo software a valutare regolarmente i propri processi e flussi di lavoro chiedendo in ogni fase: «Può essere automatizzato?»
- **Interagisci:** utilizza AI-powered gli assistenti generativi per fornire al tuo team un supporto istantaneo e contestuale per una serie di attività e domande. Questi assistenti intelligenti agiscono come collaboratori esperti che attingono da un vasto archivio di informazioni. Possono rispondere a domande di programmazione, offrire suggerimenti di progettazione, spiegare le procedure operative standard e aiutare a risolvere problemi complessi. L'integrazione di questi assistenti di intelligenza artificiale nel flusso di lavoro di sviluppo aumenta la produttività e favorisce un ambiente più collaborativo e orientato alla risoluzione dei problemi.
- **Iterazione:** utilizza l'intelligenza artificiale generativa per consentire regolazioni rapide basate sui dati in tutto l'SDLC. È possibile analizzare continuamente i dati provenienti da fonti quali il feedback dei clienti, i modelli di utilizzo, le tendenze di mercato e le metriche delle prestazioni dei team per prendere rapidamente decisioni informate. Questa adattabilità perfeziona lo sviluppo del software da un processo statico e predefinito a un approccio fluido e reattivo. Si manifesta in vari modi, tra cui la prioritizzazione dinamica degli arretrati, l'allocazione flessibile delle risorse, strategie di test adattive, documentazione in evoluzione e processi di implementazione reattivi. Ad esempio, i product manager possono utilizzare le AI-generated informazioni per riordinare gli arretrati, integrare le nuove esigenze dei clienti e le tendenze del mercato quasi in tempo reale. DevOps gli ingegneri possono adattare i piani di implementazione e le configurazioni dell'infrastruttura in base all'analisi delle prestazioni, assicurandosi che le applicazioni rimangano resilienti e ottimizzate. I team di sviluppo possono tradurre il feedback delle retrospettive Sprint in miglioramenti attuabili per la prossima iterazione, promuovendo una cultura di miglioramento continuo dei processi.
- **Impatto:** applica l'intelligenza artificiale generativa per valutare l'efficacia e le prestazioni del processo di sviluppo del software. Utilizzando AI-powered analisi e metriche, ottieni informazioni più approfondite sull'efficienza dello sviluppo, sulla qualità del codice, sul coinvolgimento degli utenti e sulle prestazioni complessive delle applicazioni. Questo approccio basato sui dati consente di prendere decisioni informate, ottimizzare i flussi di lavoro di sviluppo e migliorare continuamente

la qualità e l'esperienza utente delle applicazioni. Nel valutare la produttività del team addetto al software, l'intelligenza artificiale generativa analizza vari dati, come la frequenza di invio del codice, i tempi di risoluzione dei problemi, la velocità di rilascio, i tassi di distribuzione delle funzionalità e altro ancora. Può anche valutare la qualità delle revisioni del codice, l'efficacia degli strumenti di collaborazione e l'impatto delle diverse pratiche di sviluppo sull'output complessivo del team. Correlando queste metriche con i risultati del progetto, l'intelligenza artificiale identifica modelli e tendenze che gli analisti umani potrebbero non riconoscere, e può fornire informazioni utili che aumentano la produttività del team. Inoltre, l'intelligenza artificiale generativa può aiutarti a confrontare le prestazioni del team rispetto agli standard di settore o ai dati storici, offrendo consigli personalizzati per il miglioramento. Può anche prevedere potenziali strozzature o rischi nel processo di sviluppo in modo da poter adottare misure proattive.

Integrazione con il ciclo di vita dello sviluppo del software

L'SDLC è costituito da più fasi, che possono differire da un'organizzazione all'altra. In genere, queste fasi includono quanto segue: requisiti e pianificazione, progettazione e architettura, implementazione, test, implementazione, funzionamento e manutenzione.

La tabella seguente mappa le dimensioni del framework 5-I alle fasi SDLC e fornisce il livello di integrazione per ciascuna dimensione.

Dimensione del framework	Requisiti e pianificazione	Design e architettura	Implementazione	Test in corso	Implementazione	Funzionamento e manutenzione
Indagare	Elevata	Bassa	Bassa	Bassa	Bassa	Media
Integrare	Media	Media	Elevata	Media	Elevata	Elevata
Interagisci	Elevata	Elevata	Elevata	Media	Media	Elevata
Iterare	Media	Bassa	Bassa	Bassa	Bassa	Media
Impatto	Elevata	Media	Elevata	Bassa	Elevata	Elevata

I livelli di integrazione variano da alti a bassi. La mappatura rivela le aree di interesse chiave per ogni dimensione. Ad esempio, Investigate mostra un'elevata intensità nei requisiti e nella fase di pianificazione. Integrate dimostra un'elevata intensità nelle fasi di implementazione, implementazione, funzionamento e manutenzione.

Utilizzando questa mappatura, puoi dare priorità ai tuoi sforzi in modo efficace. Ti consigliamo di concentrarti sull'alto, poi sul medio e poi sul basso. Assicurati di adottare un approccio equilibrato e di grande impatto che migliori l'esperienza di sviluppo del software con l'intelligenza artificiale generativa.

Funzionalità fondamentali per un'esperienza di sviluppo software AI-powered

Per implementare con successo un'esperienza di sviluppo AI-powered software generativo, è necessario stabilire una serie di funzionalità di base che coinvolgano più persone all'interno dell'organizzazione. Queste funzionalità rappresentano la capacità di distribuire in modo efficace le risorse, implementare i processi e ottenere i risultati desiderati nel contesto dello sviluppo del software. AI-powered Coltivando queste funzionalità, create una solida base che vi aiuta a integrare senza problemi l'IA generativa in tutte le fasi dell'SDLC.

AWS fornisce servizi chiave per aiutarvi a implementare queste funzionalità. Ad esempio, [Amazon Q Developer](#) aiuta ad accelerare lo sviluppo del software fungendo da AI-powered assistente. [Amazon Q Business](#) ti aiuta a ottenere risposte rapide e pertinenti a domande urgenti, risolvere problemi e generare contenuti. Può anche agire per tuo conto integrando strumenti relativi allo sviluppo del software. [Amazon Bedrock](#) fornisce l'accesso a modelli di base e un'ampia gamma di funzionalità per personalizzare requisiti e flussi di lavoro di sviluppo specifici.

Sviluppando queste funzionalità Servizi AWS, crei una solida base che ti aiuta a integrare senza problemi l'IA generativa in tutte le fasi dell'SDLC.

Le seguenti sono le funzionalità fondamentali su cui concentrarti:

- [Gestione progettuale](#)
- [Gestione dei requisiti](#)
- [Architettura e design](#)
- [Collaborazione](#)
- [DevSecOps](#)

- [Funzionamento e manutenzione](#)
- [Assistenti AI](#)
- [Analisi dei dati e informazioni dettagliate](#)
- [Gestione delle conoscenze](#)
- [Estensibilità](#)

Ogni funzionalità di base si integra con le dimensioni del framework e le diverse fasi dell'SDLC. Questa integrazione consente di utilizzare le funzionalità di intelligenza artificiale in modo efficace durante tutto il processo di sviluppo del software. Migliora l'efficienza, la qualità e l'innovazione in ogni fase. La sinergia tra queste funzionalità fondamentali, il framework e le fasi SDLC crea un ecosistema completo per lo sviluppo del software. AI-powered Ciò consente di sfruttare tutto il potenziale dell'intelligenza artificiale generativa, promuovere il miglioramento continuo, accelerare i cicli di sviluppo e fornire prodotti software di qualità.

La tabella seguente mostra come le funzionalità e le funzionalità secondarie di base si collegano alle dimensioni del framework e alle fasi SDLC.

Capacità: sottocapacità	Indagare	Integra	Interagisci	Iterare	Impatto
Gestione del progetto: gestione dei problemi	Requisiti e pianificazione	Nessuno	Nessuna	Nessuna	Nessuno
Gestione del progetto: Sprint e gestione delle attività	Requisiti e pianificazione	Requisiti e pianificazione	Nessuno	Nessuna	Nessuno
Gestione del progetto: gestione del backlog di prodotto	Requisiti e pianificazione	Nessuno	Nessuno	Requisiti e pianificazione	Nessuno

Capacità: sottocapacità	Indagare	Integra	Interagisci	Iterare	Impatto
Gestione del progetto: mappatura delle storie degli utenti	Requisiti e pianificazione	Nessuno	Nessuna	Nessuna	Nessuno
Gestione del progetto: reportistica e analisi	Requisiti e pianificazione	Nessuno	Nessuna	Nessuno	Requisiti e pianificazione
Gestione del progetto: gestione della roadmap del prodotto	Requisiti e pianificazione	Nessuno	Requisiti e pianificazione	Nessuno	Nessuno
Gestione del progetto: cicli di feedback	Nessuno	Nessuna	Nessuno	Requisiti e pianificazione	Nessuno
Gestione del progetto: Retrospective	Nessuno	Nessuna	Nessuno	Requisiti e pianificazione	Nessuno
Gestione dei requisiti	Requisiti e pianificazione	Requisiti e pianificazione	Nessuno	Nessuna	Nessuno
Architettura e design: progettazione della soluzione	Design e architettura	Design e architettura	Nessuno	Nessuna	Nessuno

Capacità: sottocapacità	Indagare	Integra	Interagisci	Iterare	Impatto
Collabora zione: gestione della documenta zione	Tutte le fasi di SDLC	Nessuno	Tutte le fasi SDLC	Nessuno	Nessuno
Collabora zione: condivisi one delle conoscenze	Tutte le fasi SDLC	Nessuno	Tutte le fasi SDLC	Nessuno	Nessuno
Collabora zione: gestione delle risorse del progetto	Nessuno	Tutte le fasi di SDLC	Tutte le fasi SDLC	Nessuno	Nessuno
DevSecOps: CI/CD	Test, implement azione	Implement azione, test, implement azione	Implement azione	Nessuno	Nessuno
DevSecOps : DevOps sicurezza	Implement azione	Implement azione, test, funzionam ento e manutenzi one	Nessuno	Implement azione, test, funzionam ento e manutenzi one	Nessuno

Capacità: sottocapacità	Indagare	Integra	Interagisci	Iterare	Impatto
DevSecOps : monitoraggio delle prestazioni delle applicazioni	Nessuno	Funzionamento e manutenzione	Nessuno	Nessuna	Nessuno
DevSecOps: aggregazione e analisi dei log	Funzionamento e manutenzione	Funzionamento e manutenzione	Nessuno	Nessuna	Nessuno
DevSecOps: AIOps	Funzionamento e manutenzione	Nessuno	Nessuno	Funzionamento e manutenzione	Nessuno
DevSecOps : Miglioramento continuo	Nessuno	Nessuna	Nessuno	Funzionamento e manutenzione	Nessuno
DevSecOps : Monitoraggio del dashboard	Nessuno	Funzionamento e manutenzione	Nessuno	Nessuna	Nessuno
DevSecOps : Informazioni sulle prestazioni	Funzionamento e manutenzione	Nessuno	Nessuno	Funzionamento e manutenzione	Nessuno

Capacità: sottocapacità	Indagare	Integra	Interagisci	Iterare	Impatto
Funzionamento e manutenzione: gestione degli incidenti	Nessuno	Nessuna	Nessuno	Funzionamento e manutenzione	Nessuno
Funzionamento e manutenzione: aggiornamenti del codice	Nessuno	Funzionamento e manutenzione	Nessuno	Nessuna	Nessuno
Funzionamento e manutenzione: ottimizzazione del codice	Funzionamento e manutenzione	Funzionamento e manutenzione	Nessuno	Nessuna	Nessuno
Funzionamento e manutenzione: gestione tecnica del debito	Nessuno	Funzionamento e manutenzione	Funzionamento e manutenzione	Nessuno	Nessuno
Funzionamento e manutenzione: gestione delle modifiche	Nessuno	Implementazione, distribuzione	Nessuno	Nessuna	Nessuno

Capacità: sottocapacità	Indagare	Integra	Interagisci	Iterare	Impatto
Funzionamento e manutenzione: ingegneria inversa	Funzionamento e manutenzione	Nessuno	Nessuna	Nessuna	Nessuno
Funzionamento e manutenzione: modernizzazione del codice	Nessuno	Implementazione	Nessuno	Nessuna	Nessuno
Funzionamento e manutenzione: ottimizzazione delle prestazioni	Nessuno	Funzionamento e manutenzione	Nessuno	Funzionamento e manutenzione	Nessuno
Analisi e approfondimenti	Nessuno	Requisiti e pianificazione	Nessuno	Nessuno	Tutte le fasi SDLC
Assistente AI	Nessuno	Nessuno	Tutte le fasi SDLC	Nessuno	Nessuno
Gestione della conoscenza	Nessuno	Nessuno	Tutte le fasi SDLC	Nessuno	Nessuno
Estensibilità	Nessuno	Implementazione	Nessuno	Nessuna	Nessuno

Casi d'uso dell'intelligenza artificiale generativa per la gestione dei progetti

Una gestione efficace dei progetti è alla base di uno sviluppo software di successo. Nel contesto dell'IA generativa, la gestione dei progetti assume nuove dimensioni. Può diventare più predittiva, adattiva e basata sui dati. AI-powered gli strumenti di gestione dei progetti analizzano i dati storici del progetto per generare stime più accurate di tempi e risorse. Possono assegnare automaticamente le priorità alle attività in base agli obiettivi aziendali e alla capacità del team e possono persino prevedere potenziali ostacoli prima che si verifichino. Ad esempio, un project manager potrebbe utilizzare l'intelligenza artificiale generativa per creare un piano di progetto preliminare basato sui requisiti del progetto e sui dati storici di progetti simili. L'intelligenza artificiale potrebbe quindi suggerire composizioni di team ottimali che tengano conto delle competenze, dei carichi di lavoro e delle esigenze del progetto. Durante tutto il progetto, AI-driven le dashboard forniscono informazioni quasi in tempo reale sullo stato del progetto generando automaticamente report ed evidenziando le aree che richiedono attenzione.

Questo AI-augmented approccio alla gestione del progetto può migliorare l'efficienza. Aiuta i project manager a concentrarsi sul processo decisionale strategico e sulla leadership del team, anziché impantanarsi nelle attività amministrative di routine.

La tabella seguente mostra i casi d'uso della gestione dei progetti che è possibile migliorare con l'intelligenza artificiale generativa e la persona responsabile di tali casi d'uso.

Sottocapacità: caso d'uso	Utente
Gestione dei problemi: creazione e assegnazione di problemi	Project manager
Gestione dei problemi: rileva i problemi durante i test e li registra	Tecnico di test
Gestione dei problemi: assegna priorità ai problemi in base alla gravità e assegnali agli sviluppatori	Project manager
Gestione dei problemi: identifica e unisci i problemi duplicati	Project manager

Sottocapacità: caso d'uso	Utente
Gestione dei problemi: monitora e genera report su problemi chiave, metriche e stato generale del progetto	Project manager
Sprint e gestione delle attività: stima lo sforzo svolto per le attività e assegna gli story point in base alla capacità del team	Scrum Master
Sprint e gestione delle attività: distribuisce le attività tra i membri del team per un carico di lavoro uniforme durante lo sprint	Scrum Master
Gestione dello sprint e delle attività: facilita le sessioni di pianificazione dello sprint che allineano gli sforzi del team agli obiettivi dello sprint	Scrum Master
Gestione del backlog di prodotto: riordina gli elementi del backlog in base al valore aziendale, all'urgenza e al feedback degli utenti	Proprietario del prodotto
Gestione del backlog di prodotto: integra i nuovi feedback dei clienti e le informazioni di mercato nel backlog dei prodotti per una prioritizzazione quasi in tempo reale	Proprietario del prodotto
Gestione del backlog di prodotto: identifica e gestisci le dipendenze tra gli elementi del backlog per semplificare lo sviluppo	Responsabile del prodotto
Mappatura delle storie degli utenti: crea mappe dei percorsi degli utenti per identificare tutte le funzionalità necessarie e le storie utente corrispondenti	Proprietario del prodotto

Sottocapacità: caso d'uso	Utente
Mappatura delle storie degli utenti: identifica le lacune o le fasi mancanti nel flusso degli utenti	Business analyst
Mappatura delle storie degli utenti: dai priorità alle storie degli utenti in base al loro impatto sul valore aziendale	Responsabile del prodotto
Reportistica e analisi: genera dashboard quasi in tempo reale che visualizzano le metriche chiave del progetto, come la velocità di sprint e i tassi di risoluzione dei problemi	Project manager
Reportistica e analisi: analizza i dati storici e prevedi i risultati futuri del progetto, come potenziali ritardi o strozzature	Project manager
Reportistica e analisi: crea report personalizzati, ad esempio rapporti sulle prestazioni del team o sullo stato del progetto, personalizzati per le diverse parti interessate	Project manager
Gestione della roadmap del prodotto: crea e gestisci una roadmap di prodotto che delinei le tappe principali e le date di rilascio	Project manager
Gestione della roadmap del prodotto: aggiorna la roadmap in base ai cambiamenti nelle priorità o nelle tempistiche del progetto	Responsabile del prodotto
Gestione della roadmap del prodotto: condividi la roadmap con le parti interessate per fornire visibilità sulla direzione del prodotto	Responsabile del prodotto
Cicli di feedback: raccogli il feedback del team dopo ogni sprint e identifica le aree di miglioramento	Scrum Master

Sottocapacità: caso d'uso	Utente
Retrospective: Traduci il feedback in elementi utilizzabili per lo sprint successivo, favorendo il miglioramento continuo	Scrum Master
Retrospective: monitora l'impatto delle modifiche implementate dalle retrospective precedenti per misurarne l'efficacia	Scrum Master

Casi d'uso dell'intelligenza artificiale generativa per la gestione dei requisiti

La gestione dei requisiti è un processo critico strettamente legato alla gestione dei progetti. Immagina un proprietario di prodotto che utilizza uno strumento di intelligenza artificiale per analizzare il feedback dei clienti, le tendenze del mercato e gli input delle parti interessate. Lo strumento di intelligenza artificiale potrebbe generare una serie completa di storie e requisiti degli utenti, classificarli automaticamente, rilevare potenziali conflitti o lacune e persino suggerire l'assegnazione delle priorità in base al valore aziendale e alla complessità dell'implementazione. Man mano che il progetto avanza e i requisiti evolvono, l'IA può aggiornare e perfezionare continuamente i requisiti per assicurarsi che rimangano in linea con le mutevoli esigenze aziendali e i vincoli tecnici. Questo AI-driven approccio dinamico alla gestione dei requisiti aiuta a garantire che gli sforzi di sviluppo rimangano strettamente allineati con le esigenze degli utenti e gli obiettivi aziendali durante tutto il ciclo di vita del progetto.

La tabella seguente mostra i casi d'uso della gestione dei requisiti che è possibile migliorare con l'intelligenza artificiale generativa e la persona responsabile di tali casi d'uso.

Caso d'uso	Utente
Crea requisiti aziendali	Business analyst
Crea epopee partendo dalle funzionalità	Proprietario del prodotto
Tieni traccia dei progressi di un'epopea monitorando il completamento delle storie utente associate	Responsabile del prodotto

Caso d'uso	Utente
Crea storie di utenti	Proprietario del prodotto
Stima lo sforzo richiesto per ogni storia d'uso e assegna punti alla storia	Scrum Master
Definisci i criteri di accettazione per ogni storia utente	Proprietario del prodotto

Casi d'uso dell'intelligenza artificiale generativa per l'architettura e il design

Con una solida base di gestione dei progetti e requisiti ben definiti, la prossima funzionalità fondamentale è l'architettura e il design. Qui, l'intelligenza artificiale generativa sta aprendo nuove possibilità per creare architetture software solide, scalabili ed efficienti. AI-powered gli strumenti di progettazione possono analizzare requisiti e vincoli per suggerire modelli architettonici e approcci progettuali ottimali. Generano diverse alternative di progettazione e ciascuna è ottimizzata per priorità diverse, come prestazioni, scalabilità o manutenibilità. Ad esempio, un architetto di soluzioni potrebbe utilizzare un assistente AI per generare rapidamente diversi progetti architettonici di alto livello in base ai requisiti del progetto. Questo AI-augmented approccio accelera il processo di progettazione e aiuta gli architetti a prendere decisioni più informate. Ciò porta a progetti software più robusti e a prova di futuro.

La tabella seguente mostra i casi d'uso di architettura e progettazione che è possibile migliorare con l'intelligenza artificiale generativa e la persona responsabile di tali casi d'uso.

Caso d'uso	Utente
Crea un documento di architettura	Architetto di soluzioni
Crea un documento di progettazione dettagliato	Responsabile tecnico
Comprendi l'architettura e gli standard di progettazione esistenti	Architetto di soluzioni
Sviluppa modelli e prototipi dettagliati di un'interfaccia utente	UX/UI designer

Casi d'uso dell'intelligenza artificiale generativa per la collaborazione

Lo sviluppo del software è intrinsecamente un'attività collaborativa. Puoi utilizzare l'intelligenza artificiale generativa per migliorare la collaborazione nel tuo team di sviluppo software. AI-powered gli strumenti di collaborazione vanno oltre la semplice messaggistica e la condivisione di file. Facilitano una comunicazione più efficace riassumendo i lunghi thread di discussione, evidenziando le decisioni chiave e persino suggerendo orari ottimali per le riunioni in base agli orari e ai modelli di produttività dei membri del team. L'intelligenza artificiale può aiutare nella revisione del codice identificando automaticamente potenziali problemi, suggerendo miglioramenti e persino spiegando modifiche complesse ai revisori. Durante le sessioni di brainstorming, l'intelligenza artificiale può fungere da facilitatore, generare idee, aiutare a organizzare i pensieri e persino mediare le discussioni per assicurarsi che tutte le voci vengano ascoltate. Per i team distribuiti, l'intelligenza artificiale può aiutare a superare le barriere culturali e linguistiche. Può fornire una traduzione linguistica quasi in tempo reale nelle chat e nelle videochiamate e offrire un contesto culturale per aiutare a prevenire malintesi. Aumentando la collaborazione umana con l'intelligenza artificiale, questa funzionalità aiuta i team a lavorare in modo più efficiente ed efficace, il che favorisce l'innovazione e migliora i risultati complessivi del progetto.

La tabella seguente mostra come utilizzare l'intelligenza artificiale generativa per migliorare i casi d'uso della collaborazione.

Sottocapacità: caso d'uso	Utente
Gestione dei documenti: creazione e gestione di un archivio di documentazione centralizzato	Scrittore tecnico
Gestione dei documenti: consenti a più membri del team di collaborare sulla documentazione in tempo reale	Team di sviluppo
Condivisione delle conoscenze: utilizza i forum di discussione come piattaforma per gli sviluppatori per porre domande, condividere conoscenze e risolvere problemi in modo collaborativo	Team di sviluppo
Condivisione delle conoscenze: utilizza i forum di discussione per documentare e	Responsabile del prodotto

Sottocapacità: caso d'uso	Utente
tenere traccia delle decisioni prese durante le discussioni sul progetto, assicurandoti che la logica alla base delle decisioni chiave sia acquisita e accessibile per riferimenti futuri	
Gestione delle risorse di progetto: facilita la condivisione delle risorse relative al progetto	Team di sviluppo
Gestione delle risorse di progetto: implementa il controllo delle versioni per i contenuti condivisi in modo che i membri del team possano tenere traccia delle modifiche, tornare alle versioni precedenti e collaborare agli aggiornamenti dei contenuti	Team di sviluppo

Casi d'uso dell'intelligenza artificiale generativa per DevSecOps

AI-powered DevSecOps gli strumenti automatizzano molti aspetti della pipeline di distribuzione del software. Ad esempio, possono eseguire revisioni intelligenti del codice, rilevare potenziali bug, rilevare vulnerabilità di sicurezza e identificare problemi di prestazioni quasi in tempo reale mentre gli sviluppatori scrivono il codice. L'intelligenza artificiale genera ed esegue suite di test complete e le aggiorna automaticamente man mano che la codebase si evolve. Questo AI-augmented approccio DevSecOps accelera la pipeline di distribuzione e migliora significativamente la sicurezza e l'affidabilità del software distribuito.

La tabella seguente mostra i casi DevSecOps d'uso che è possibile migliorare con l'intelligenza artificiale generativa e il responsabile di tali casi d'uso.

Sottocapacità: caso d'uso	Utente
DevOps e distribuzione continua: intere pipeline di implementazione automatizzate	DevOps ingegnere

Sottocapacità: caso d'uso	Utente
DevOps e distribuzione continua: ricevi feedback quasi in tempo reale sulla qualità del codice e sui potenziali problemi	Software developer
DevOps e distribuzione continua: ricevi suggerimenti su problemi di sicurezza e soluzioni quasi in tempo reale	Software developer
DevOps e distribuzione continua: ricevi codice e suggerimenti sulle migliori pratiche quasi in tempo reale	Software developer
DevOps e distribuzione continua: automatizza le attività ripetitive e integra i comandi negli script	DevOps ingegnere
DevOps e distribuzione continua: crea codice e genera artefatti automaticamente dopo ogni commit del codice	Software developer
DevOps e distribuzione continua: crea codice in base agli standard e al framework dell'organizzazione	Software developer
DevOps e distribuzione continua: esegui automaticamente test unitari su ogni commit per catturare gli errori nelle prime fasi del processo di sviluppo	Software developer
DevOps e distribuzione continua: analizza la copertura degli unit test per assicurarti che tutti i percorsi di codice critici vengano testati	Software developer
DevOps e distribuzione continua: gestisci le filiali e unisci le modifiche	Software developer

Sottocapacità: caso d'uso	Utente
DevOps e distribuzione continua: gestisci il controllo delle versioni di codice e artefatti	Software developer
DevOps e distribuzione continua: archivia e gestisci gli artefatti e le dipendenze delle build	DevOps ingegnere
DevOps e distribuzione continua: risolvi e recupera le dipendenze durante il processo di creazione	Software developer
DevOps e distribuzione continua: genera ed esegui test di integrazione per assicurarti che i componenti funzionino insieme come previsto	Tecnico di test
DevOps e distribuzione continua: utilizza servizi fittizi durante i test di integrazione per simulare le interazioni con sistemi esterni	Ingegnere di test
DevOps e fornitura continua: confronta le prestazioni delle applicazioni con carichi diversi	Ingegnere delle prestazioni
DevOps e distribuzione continua: simula scenari ad alto traffico per testare la scalabilità e i tempi di risposta dell'applicazione	Ingegnere delle prestazioni
DevOps e distribuzione continua: verifica la capacità del sistema di ripristinarsi in caso di guasti, come arresti anomali del server o interruzioni della rete	Ingegnere dell'affidabilità del sito
DevOps e consegna continua: esegui l'ingegneria del caos	Ingegnere dell'affidabilità del sito
DevOps e distribuzione continua: esegui test per verificare che l'applicazione soddisfi i requisiti aziendali	Ingegnere QA

Sottocapacità: caso d'uso	Utente
DevOps e consegna continua: esegui test di accettazione da parte degli utenti	Proprietario del prodotto
DevOps e distribuzione continua: scansiona le dipendenze per individuare vulnerabilità e problemi di conformità delle licenze	Ingegnere della sicurezza
DevOps e distribuzione continua: monitora e gestisci le dipendenze open source per assicurarti che siano aggiornate e sicure	Ingegnere della sicurezza
DevOps e fornitura continua: genera e gestisci una distinta base del software (SBOM) per tenere traccia di tutti i componenti e le dipendenze	Ingegnere della sicurezza
DevOps e fornitura continua: utilizza lo SBOM per condurre audit di conformità normativa	Responsabile della conformità
DevOps e distribuzione continua: crea note di rilascio	Gestore delle versioni
DevOps e distribuzione continua: pianifica e coordina i rilasci	Responsabile dei rilasci
DevOps e distribuzione continua: implementa procedure operative standard per il rollback e la gestione dei rilasci	Release manager
DevOps e distribuzione continua: utilizza i flag di funzionalità per abilitare o disabilitare le funzionalità in produzione senza implementare nuovo codice	Responsabile del prodotto

Sottocapacità: caso d'uso	Utente
DevOps e distribuzione continua: A/B esegui test utilizzando i flag di funzionalità per misurare l'impatto di diverse funzionalità sul comportamento degli utenti	Responsabile del prodotto
DevOps e distribuzione continua: analizza e monitora i guasti della pipeline	DevOps ingegnere
DevOps e distribuzione continua: crea e gestisci le risorse dell'infrastruttura	DevOps ingegnere
DevOps e sicurezza: scansiona gli archivi di codice alla ricerca di segreti codificati	DevOps ingegnere
DevOps e sicurezza: implementa il rilevamento quasi in tempo reale per avvisare immediatamente gli sviluppatori se nel repository vengono archiviati segreti	DevOps ingegnere
DevOps e sicurezza: applica il monitoraggio continuo della qualità del codice	Software developer
DevOps e sicurezza: rileva e contrassegna gli indicatori di potenziali vulnerabilità di sicurezza nel codice	Software developer
DevOps e sicurezza: implementate test automatici per identificare i 10 principali rischi di sicurezza dell'Open Worldwide Application Security Project (OWASP) per assicurarvi che l'applicazione aderisca alle pratiche di sicurezza standard del settore	Ingegnere della sicurezza
DevOps e sicurezza: aggiorna e istruisci regolarmente gli sviluppatori sui rischi OWASP integrando i controlli nel processo di sviluppo	Ingegnere della sicurezza

Sottocapacità: caso d'uso	Utente
DevOps e sicurezza: scansiona le librerie e le dipendenze di terze parti per individuare vulnerabilità di sicurezza note	DevOps ingegnere
DevOps e sicurezza: scansiona il codice e l'infrastruttura dell'applicazione per rilevare le vulnerabilità	DevOps ingegnere
DevOps e sicurezza: analizza il codice per individuare eventuali vulnerabilità prima della distribuzione	Ingegnere della sicurezza
DevOps e sicurezza: applica le politiche di sicurezza impedendo che il codice con vulnerabilità critiche venga unito	Ingegnere della sicurezza
DevOps e sicurezza: implementa il controllo degli accessi basato sui ruoli (RBAC) per limitare l'accesso a sistemi e dati sensibili e per garantire che solo il personale autorizzato possa accedere alle risorse critiche	Ingegnere della sicurezza
DevOps e sicurezza: regola i controlli di accesso in base a ruoli e responsabilità adattandoti ai cambiamenti nella struttura del team	DevOps ingegnere
DevOps e sicurezza: testa le applicazioni in esecuzione alla ricerca di vulnerabilità di sicurezza quasi in tempo reale simulando gli attacchi all'ambiente di produzione	Ingegnere della sicurezza
DevOps e sicurezza: monitora continuamente le applicazioni distribuite per individuare eventuali vulnerabilità di sicurezza	DevOps ingegnere

Sottocapacità: caso d'uso	Utente
DevOps e sicurezza: pianifica scansioni periodiche delle vulnerabilità in tutti gli ambienti per identificare e risolvere i punti deboli della sicurezza	Ingegnere della sicurezza
DevOps e sicurezza: applica patch e aggiornamenti in base ai risultati della scansione delle vulnerabilità per contribuire a mantenere i sistemi sicuri	DevOps ingegnere
Monitoraggio delle prestazioni delle applicazioni: monitora continuamente le prestazioni delle applicazioni quasi in tempo reale per rilevare e diagnosticare i problemi di prestazioni prima che influiscano sugli utenti	Ingegnere dell'affidabilità del sito
Monitoraggio delle prestazioni delle applicazioni: rileva le anomalie delle prestazioni, come picchi improvvisi nei tempi di risposta o un aumento dei tassi di errore, e avvia avvisi	DevOps ingegnere
Monitoraggio delle prestazioni delle applicazioni: traccia le richieste man mano che si propagano attraverso un sistema distribuito per identificare i rallentamenti delle prestazioni e i problemi di latenza	DevOps ingegnere
Monitoraggio delle prestazioni delle applicazioni: utilizza il tracciamento distribuito per individuare il servizio o il componente esatto responsabile dei guasti o del degrado delle prestazioni	DevOps ingegnere

Sottocapacità: caso d'uso	Utente
Aggregazione e analisi dei log: aggrega i log provenienti da più fonti in un sistema centralizzato per facilitare la ricerca e l'analisi al fine di identificare tendenze e problemi	Ingegnere dell'affidabilità del sito
Aggregazione e analisi dei log: implementa l'analisi automatica dei log per estrarre le informazioni pertinenti e rilevare modelli o anomalie che potrebbero indicare problemi	DevOps ingegnere
Aggregazione e analisi dei log: raccogli e visualizza le metriche chiave delle prestazioni	Ingegnere dell'affidabilità del sito
Aggregazione e analisi dei log: monitora le metriche rispetto agli accordi sui livelli di servizio (SLA) predefiniti	Responsabile del prodotto
Operazioni di intelligenza artificiale: rileva gli incidenti, analizza le cause profonde e avvia azioni correttive senza l'intervento umano	DevOps ingegnere
Operazioni AI: prevedi le future richieste di risorse e ottimizza la pianificazione della capacità per evitare interruzioni	Ingegnere dell'affidabilità del sito
Miglioramento continuo: monitora le interazioni degli utenti reali con l'applicazione per raccogliere informazioni sulle prestazioni e identificare le aree di miglioramento	Progettista UX
Miglioramento continuo: monitora le prestazioni delle applicazioni in diverse aree geografiche per garantire un'esperienza utente coerente a livello globale	Responsabile del prodotto

Sottocapacità: caso d'uso	Utente
Monitoraggio del dashboard: crea dashboard personalizzabili per visualizzare metriche, log e tracce critici quasi in tempo reale, al fine di fornire una visione completa dello stato del sistema	Ingegnere dell'affidabilità del sito
Monitoraggio delle dashboard: crea dashboard per diversi team (ad esempio team di sviluppo, operativi e di prodotto) per fornire informazioni pertinenti in base alle aree di interesse	DevOps ingegnere
Informazioni sulle prestazioni: conduci un'analisi dettagliata delle prestazioni delle applicazioni per identificare le inefficienze e ottimizzare il codice o l'infrastruttura	Software developer
Informazioni sulle prestazioni: utilizza le informazioni sulle prestazioni per migliorare in modo iterativo le prestazioni delle applicazioni e ottimizzare l'esperienza utente nel tempo	Responsabile del prodotto

Casi d'uso dell'intelligenza artificiale generativa per il funzionamento e la manutenzione

Dopo l'implementazione del software, l'attenzione si sposta sul funzionamento e sulla manutenzione. L'intelligenza artificiale generativa può migliorare gli approcci tradizionali fornendo una gestione del sistema più proattiva ed efficiente. AI-powered gli strumenti operativi monitorano continuamente le prestazioni del sistema e prevedono potenziali problemi prima che influiscano sugli utenti. Eseguono un'analisi automatizzata delle cause principali quando si verificano problemi, il che riduce significativamente il tempo medio di risoluzione. L'intelligenza artificiale ottimizza anche le prestazioni del sistema quasi in tempo reale. Regola automaticamente le configurazioni in base alla modifica dei modelli di carico e dei comportamenti degli utenti. Ad esempio, un team operativo potrebbe utilizzare un assistente AI per generare programmi di manutenzione predittiva, identificare automaticamente i componenti che potrebbero guastarsi e suggerire azioni preventive. L'intelligenza artificiale potrebbe

anche aiutare nella pianificazione della capacità analizzando le tendenze di utilizzo e prevedendo le future esigenze di risorse con elevata precisione.

La tabella seguente mostra i casi d'uso di funzionamento e manutenzione che è possibile migliorare con l'intelligenza artificiale generativa e la persona responsabile di tali casi d'uso.

Sottocapacità: caso d'uso	Utente
Gestione degli incidenti: gestisci gli incidenti quasi in tempo reale integrando gli strumenti di monitoraggio con le piattaforme di chat in modo che i team possano rilevare, discutere e risolvere i problemi direttamente all'interno dell'ambiente di chat	Ingegnere dell'affidabilità del sito
Gestione degli incidenti: consente ai team di avviare implementazioni, eseguire script ed eseguire comandi direttamente dall'interfaccia di chat, il che semplifica le operazioni	DevOps ingegnere
Aggiornamenti del codice: aggiorna le dipendenze e le librerie del codice per ridurre lo sforzo manuale e assicurarti che la codebase rimanga aggiornata con le versioni più recenti	Software developer
Ottimizzazione del codice: rivedi il codice per individuare opportunità di ottimizzazione	Software developer
Ottimizzazione del codice: identifica i punti deboli nel codice e rifattorizza o ottimizza il codice per migliorare le prestazioni	Software developer
Gestione tecnica del debito: registra il debito tecnico come parte del processo di sviluppo	Responsabile del prodotto
Gestione tecnica del debito: assegna priorità e gestisci il debito tecnico in base all'impatto,	Software developer

Sottocapacità: caso d'uso	Utente
al rischio e ai costi e integratelo nel normale processo di pianificazione degli sprint	
Gestione tecnica del debito: riduci il debito tecnico nel codice applicativo esistente	Software developer
Gestione delle modifiche: implementa un processo di approvazione delle modifiche che assicuri che tutte le modifiche al codice siano riviste, testate e approvate dalle parti interessate necessarie prima della distribuzione	Gestore delle modifiche
Gestione delle modifiche: esegue l'analisi dell'impatto delle modifiche proposte	DevOps ingegnere
Ingegneria inversa: analizza e comprendi la struttura e il comportamento del codice legacy	Architetto di soluzioni
Ingegneria inversa: spiega il codice esistente e genera documentazione	Software developer
Modernizzazione del codice: Traduci il codice da un linguaggio di programmazione a un altro	Software developer
Modernizzazione del codice: modernizza il codice legacy con il linguaggio di programmazione più recente	Software developer
Ottimizzazione delle prestazioni: monitora e ottimizza continuamente le prestazioni del sistema ottimizzando l'allocazione delle risorse, il bilanciamento del carico e la riconfigurazione dell'applicazione	Ingegnere dell'affidabilità del sito

Sottocapacità: caso d'uso	Utente
Ottimizzazione delle prestazioni: identifica e rifattorizza il codice che causa un peggioramento delle prestazioni per migliorare la velocità e la reattività del sistema	Software developer

Casi d'uso per assistenti generativi di intelligenza artificiale nello sviluppo di software

La funzionalità dell'assistente AI è al centro dell'esperienza di sviluppo AI-powered generativo. Questo sistema intelligente e sensibile al contesto funge da collaboratore virtuale per tutti i membri del team dell'intero SDLC. Immagina uno sviluppatore che lavora su un codice complesso. Possono semplicemente chiedere aiuto all'assistente AI, che può fornire frammenti di codice pertinenti, spiegare algoritmi complessi o persino suggerire ottimizzazioni basate sul contesto e sulle migliori pratiche correnti. L'assistente AI può aiutare un manager ITOps a comprendere una procedura operativa standard basata su documenti interni. Fornendo un supporto istantaneo e contestuale, gli assistenti AI riducono significativamente il carico cognitivo sui membri del team. Questo li aiuta a concentrarsi su attività creative e di risoluzione dei problemi di livello superiore. Questa funzionalità funge da moltiplicatore di forza che migliora la produttività e la qualità in tutte le fasi dello sviluppo del software.

La tabella seguente mostra i casi d'uso che è possibile migliorare con gli assistenti di intelligenza artificiale e la persona beneficiaria.

Caso d'uso	Utente
Fornisci assistenza immediata al team di sviluppo rispondendo a domande, ad esempio su requisiti, architetture e procedure operative standard	Team di sviluppo software
Cerca o recupera estratti da un'ampia documentazione o genera riassunti utilizzando query in linguaggio naturale	Team di sviluppo software

Caso d'uso	Utente
Riassumi lunghi documenti tecnici, come documenti relativi ai requisiti, documentazioni sulla progettazione dell'architettura e processi interni	Team di sviluppo software
Gestisci una libreria di istruzioni che il team può utilizzare per attività comuni	Team di sviluppo software
Integra perfettamente l'intelligenza artificiale generativa negli strumenti e nei sistemi esistenti	Team di sviluppo software
Automatizza le attività su varie piattaforme, strumenti e sistemi interni	Team di sviluppo software
Crea un archivio centralizzato di conoscenze, che include best practice, informazioni specifiche sul progetto e conoscenze del team, accessibile a tutti i membri del team	Team di sviluppo software
Recupera le conoscenze pertinenti dal repository in base al contesto dell'attività	Team di sviluppo software
Esegui revisioni automatiche del codice, analisi delle cause principali, suggerisci miglioramenti, rileva potenziali bug ed esegui la risoluzione dei problemi	Sviluppatore di software, DevOps ingegnere e tecnico dell'affidabilità del sito
Analizza i dati sulle prestazioni per identificare tendenze e modelli in grado di orientare le decisioni sull'ottimizzazione delle prestazioni	Ingegnere dell'affidabilità del sito
Fornisci consigli per migliorare l'efficienza, ridurre la complessità e migliorare la sicurezza	Software developer

Caso d'uso	Utente
Suggerisci ottimizzazioni per l'utilizzo delle risorse cloud, come consigli di scalabilità o strategie di riduzione dei costi	Sviluppatore di software, DevOps ingegnere, ingegnere dell'affidabilità del sito e architetto di soluzioni
Genera nuovi contenuti, come documentazione basata su codice, guide per l'utente o versioni di funzionalità del prodotto	Team di sviluppo software

Casi d'uso dell'intelligenza artificiale generativa per analisi e approfondimenti

La funzionalità di analisi e approfondimenti aiuta a convertire grandi quantità di dati in informazioni fruibili che guidano il processo decisionale e il miglioramento continuo. Utilizzando l'intelligenza artificiale generativa, questa funzionalità elabora i dati provenienti da varie fonti, tra cui archivi di codice, strumenti di gestione dei progetti e piattaforme di collaborazione in team, per fornire una visione olistica del processo di sviluppo e della produttività del team. L'intelligenza artificiale generativa va oltre le metriche tradizionali per offrire analisi predittive e prescrittive. Può prevedere potenziali problemi e suggerire miglioramenti mirati. Ad esempio, può analizzare i modelli nei commit di codice, i tassi di risoluzione dei bug e la velocità di distribuzione delle funzionalità per identificare team ad alte prestazioni, individuare i punti deboli e suggerire ottimizzazioni dei processi. Inoltre, può fornire informazioni sulle dinamiche del team e sulle prestazioni individuali. Queste informazioni aiutano i leader a prendere decisioni basate sui dati sulla distribuzione del carico di lavoro, sulle esigenze di formazione e sulla composizione del team. Presentando queste informazioni tramite dashboard interattivi, la funzionalità consente alle parti interessate a tutti i livelli di prendere decisioni informate, ottimizzare i processi e migliorare continuamente la produttività del team, il che porta a una fornitura più rapida di software di alta qualità.

La tabella seguente mostra i casi d'uso dell'analisi che è possibile migliorare con l'intelligenza artificiale generativa e la persona responsabile di tali casi d'uso.

Caso d'uso	Utente
Monitora la produttività individuale e del team	Responsabile dello sviluppo

Caso d'uso	Utente
Analizza le tendenze della produttività per rilevare il potenziale esaurimento in modo da poter adottare misure proattive per mantenere il benessere e la produttività del team	Responsabile dello sviluppo
Tieni traccia della frequenza con cui le modifiche al codice vengono implementate nella produzione per misurare la velocità e l'agilità del processo di sviluppo	Responsabile del prodotto
Analizza i dati sulla frequenza di implementazione per identificare i periodi di scarsa attività di implementazione che potrebbero indicare inefficienze dei processi o vincoli di risorse	Responsabile del prodotto
Misura il tempo che intercorre tra l'invio del codice e l'implementazione per identificare le opportunità per semplificare i processi di sviluppo e implementazione	Responsabile dello sviluppo
Tieni traccia della percentuale di implementazioni che provocano errori che richiedono una correzione immediata per valutare l'affidabilità del processo di rilascio	Ingegnere dell'affidabilità del sito
Utilizzate le metriche relative al tasso di errore delle modifiche per identificare le aree di codice che causano frequentemente problemi, in modo da orientare le attività mirate di refactoring e test	Software developer
Monitora il tempo necessario per ripristinare il servizio dopo un'interruzione o un incidente in modo da ridurre i tempi di inattività e migliorare la resilienza complessiva del sistema	Ingegnere dell'affidabilità del sito

Caso d'uso	Utente
Analizza le tendenze dei tempi di ripristino per migliorare i processi di risposta agli incidenti e accelerare il ripristino in caso di guasti del sistema	DevOps ingegnere
Crea una dashboard personalizzata che aggrega le metriche chiave, come la frequenza di implementazione, il lead time e il tasso di errore delle modifiche, al fine di fornire una visione completa dello sviluppo e dello stato operativo	Responsabile del prodotto
Crea dashboard personalizzate in base alle esigenze dei diversi team per fornire informazioni mirate sulle rispettive aree di responsabilità specifiche, come lo sviluppo, le operazioni o il business	Responsabile del prodotto
Tieni traccia degli indicatori chiave di prestazione (KPI) aziendali, come l'impatto sui ricavi, la soddisfazione dei clienti e la quota di mercato, al fine di allineare gli sforzi di sviluppo con obiettivi aziendali più ampi	Responsabile del prodotto
Analizza l'impatto delle nuove funzionalità sui KPI aziendali per valutarne il successo e guidare lo sviluppo futuro dei prodotti	Business analyst
Monitora le metriche di qualità del codice, come la complessità del codice, la copertura dei test e la densità dei bug, per assicurarti che la codebase rimanga gestibile e sicura	Software developer
Identifica le aree della codebase che richiedono o il refactoring per promuovere la sostenibilità a lungo termine e ridurre il debito tecnico	Architetto di soluzioni

Casi d'uso dell'intelligenza artificiale generativa per la gestione della conoscenza

In qualsiasi organizzazione di sviluppo software, la conoscenza è una risorsa fondamentale. La capacità di gestione della conoscenza, basata sull'intelligenza artificiale generativa, migliora il modo in cui questa risorsa viene acquisita, organizzata e utilizzata. I sistemi tradizionali di gestione della conoscenza spesso contengono troppe informazioni, contengono contenuti obsoleti o sono difficili da cercare per trovare rapidamente le informazioni pertinenti.

L'intelligenza artificiale generativa affronta queste sfide a testa alta. Genera e aggiorna automaticamente la documentazione in base a modifiche al codice, conversazioni e artefatti del progetto. Ciò garantisce che le basi di conoscenza rimangano aggiornate senza richiedere interventi manuali da parte dei membri del team. Ancora più importante, l'intelligenza artificiale rende queste conoscenze accessibili in modi intuitivi. I membri del team possono porre domande in linguaggio naturale e l'intelligenza artificiale può fornire risposte pertinenti. L'intelligenza artificiale può attingere a una varietà di fonti, come documentazione ufficiale, commenti sul codice, thread di discussione e persino risorse esterne. Ad esempio, un nuovo membro del team che cerca di comprendere un componente specifico potrebbe chiedere all'IA: «Come funziona il modulo di autenticazione?» L'intelligenza artificiale fornirebbe quindi una spiegazione concisa e collegamenti alle sezioni di codice pertinenti, ai diagrammi architettonici e alle modifiche recenti. Potrebbe persino personalizzare queste informazioni in base al ruolo e al livello di esperienza del membro del team.

Questa funzionalità accelera l'onboarding, riduce le domande ripetitive e promuove la condivisione delle conoscenze all'interno dell'organizzazione. Aiuta a preservare le conoscenze istituzionali, rendendo più facile per i team mantenere ed evolvere sistemi complessi nel tempo.

La tabella seguente mostra i casi d'uso della gestione della conoscenza che è possibile migliorare con l'intelligenza artificiale generativa e la persona responsabile di tali casi d'uso.

Caso d'uso	Utente
Crea una piattaforma unificata che faciliti l'accesso a tutte le conoscenze relative ai progetti	Team di sviluppo software
Acquisisci le conoscenze derivanti da varie attività di sviluppo	Team di sviluppo software

Caso d'uso	Utente
Fornisci funzionalità di ricerca avanzate per trovare rapidamente le conoscenze pertinenti all'interno di un repository	Team di sviluppo software
Personalizza i moduli e i percorsi di apprendimento per il team	Team di sviluppo software

Casi d'uso dell'intelligenza artificiale generativa per l'estensibilità

L'estensibilità consente una perfetta integrazione con gli strumenti e i flussi di lavoro esistenti, consentendo al contempo alle organizzazioni di adattare il sistema di intelligenza artificiale alle proprie esigenze specifiche. Questa funzionalità fornisce API, SDK e interfacce personalizzabili robuste che facilitano l'integrazione delle funzionalità di intelligenza artificiale nei più diffusi strumenti di sviluppo e gestione dei progetti. Ad esempio, le organizzazioni possono migliorare Jira con AI-powered funzionalità per la prioritizzazione automatica dei ticket, la stima degli sforzi e la pianificazione degli sprint. Puoi potenziare le pipeline Jenkins con l'intelligenza artificiale per l'ottimizzazione intelligente delle build e la selezione predittiva dei test.

Inoltre, l'estensibilità consente una profonda integrazione con ambienti di sviluppo integrati (IDE), sistemi di controllo delle versioni e piattaforme di revisione del codice. L'intelligenza artificiale può aiutare a codificare, automatizzare le revisioni del codice e generare documentazione contestuale.

La funzionalità supporta anche la formazione e l'ottimizzazione dei modelli di intelligenza artificiale su dati specifici dell'organizzazione. Questo aiuta l'intelligenza artificiale a comprendere i modelli di codifica specifici dell'azienda, le preferenze architetturali e la conoscenza del dominio. Il risultato è un'assistenza più pertinente e sensibile al contesto in tutti gli strumenti integrati. Fornendo questo livello di flessibilità e integrazione, l'estensibilità garantisce che l'esperienza di AI-powered sviluppo si evolva con l'organizzazione. Può adattarsi alle tecnologie e alle esigenze aziendali in evoluzione, migliorando al contempo senza problemi le toolchain e i flussi di lavoro esistenti.

La tabella seguente mostra i casi d'uso dell'estensibilità che è possibile migliorare con l'intelligenza artificiale generativa e il responsabile di tali casi d'uso.

Caso d'uso	Utente
Integra strumenti di terze parti nell'ambiente di sviluppo	DevOps ingegnere
Crea flussi di lavoro di automazione personalizzati adattati al processo di sviluppo unico del team	DevOps ingegnere
Connect a varie API e servizi	DevOps ingegnere
Crea connettori per strumenti multiplatforma	DevOps ingegnere

Le migliori pratiche per l'utilizzo dell'IA generativa nello sviluppo di software

Questa sezione descrive le migliori pratiche per l'integrazione dell'IA generativa nel ciclo di vita dello sviluppo del software (SDLC). Dall'implementazione di toolchain e DevSecOps pipeline senza interruzioni alla promozione della collaborazione e all'automazione di attività ripetitive, queste linee guida vi aiutano a sfruttare la potenza dell'intelligenza artificiale per migliorare i processi e le esperienze di sviluppo. Seguendo queste best practice, i team di sviluppo software possono sbloccare nuovi livelli di efficienza, innovazione e qualità nel loro lavoro.

In questa sezione vengono illustrate le seguenti best practice:

- [Implementazione di una toolchain integrata e senza interruzioni end-to-end](#)
- [end-to-end CI/CD Implementazione di una pipeline per DevSecOps](#)
- [Adottare strumenti e pratiche collaborative](#)
- [Automatizzazione delle attività ripetitive](#)
- [Revisione e iterazione periodiche dell'esperienza di sviluppo](#)
- [Adottare pratiche efficaci di gestione dei progetti](#)
- [Attuazione della gestione della conoscenza](#)
- [Fornire estensibilità e personalizzazione](#)
- [Ottimizzazione per le operazioni](#)
- [Utilizzo di approfondimenti basati sui dati](#)
- [Adottare un approccio basato sulla piattaforma](#)

Implementazione di una toolchain integrata e senza interruzioni end-to-end

L'implementazione di una toolchain end-to-end integrata e senza interruzioni è una best practice fondamentale per creare un'esperienza di sviluppo generativa basata sull'intelligenza artificiale. L'idea principale è creare un ecosistema coeso di strumenti e piattaforme che i team di software possano utilizzare nell'intero SDLC. Il team può utilizzare la toolchain per pianificare, ideare, codificare, creare, testare, implementare e gestire le operazioni in corso. Integrando le funzionalità di intelligenza

artificiale generativa in questa toolchain, ti assicuri che l'assistenza basata sull'intelligenza artificiale sia disponibile in ogni fase. Questa integrazione riduce o elimina i passaggi manuali, riduce il cambio di contesto e aiuta i dati e gli artefatti a fluire senza intoppi tra le diverse fasi di sviluppo. Ad esempio, i frammenti di codice generati dall'intelligenza artificiale provenienti dall'ambiente di sviluppo integrato (IDE) possono confluire senza problemi nel sistema di controllo delle versioni, mentre l'analisi basata sull'intelligenza artificiale proveniente dalla piattaforma di implementazione può contribuire agli strumenti di gestione dei progetti. Questo crea un ciclo di feedback continuo che migliora il processo di sviluppo.

end-to-end CI/CD Implementazione di una pipeline per DevSecOps

Per sfruttare questa toolchain integrata e implementare un'integrazione end-to-end continua e una distribuzione continua (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), puoi anche utilizzare l'intelligenza artificiale generativa per aggiornare automaticamente le politiche di sicurezza in base alle più recenti informazioni sulle minacce.

Adottare strumenti e pratiche collaborative

Man mano che ottimizzi la tua infrastruttura di sviluppo, non dimenticare l'elemento umano. Lo sviluppo del software è intrinsecamente un'attività collaborativa. Coinvolge team interfunzionali composti da sviluppatori, designer, product manager, Scrum Master, analisti aziendali e altre parti interessate. Queste persone lavorano collettivamente per portare a compimento le idee. Utilizzando strumenti collaborativi moderni e promuovendo una cultura di comunicazione aperta e condivisione delle conoscenze, è possibile migliorare in modo significativo la produttività e l'efficacia dei team di sviluppo software. Nella tua esperienza di sviluppo software basata sull'intelligenza artificiale, questi strumenti assumono nuove dimensioni. Puoi integrare l'intelligenza artificiale nelle piattaforme di collaborazione per facilitare una comunicazione e una condivisione delle conoscenze più efficaci tra i membri del team. Gli assistenti di intelligenza artificiale possono rispondere a domande comuni, riassumere le discussioni o persino mediare i conflitti. L'intelligenza artificiale generativa può migliorare i processi di revisione del codice suggerendo automaticamente

miglioramenti o identificando potenziali problemi. Inoltre, puoi utilizzare l'intelligenza artificiale per creare documentazione dinamica e sensibile al contesto che si aggiorna quasi in tempo reale man mano che il progetto si evolve, in modo che tutti i membri del team abbiano accesso alle informazioni più aggiornate e pertinenti.

Automatizzazione delle attività ripetitive

Utilizzando l'intelligenza artificiale generativa per gestire attività di routine e dispendiose in termini di tempo, consentite ai team addetti al software di concentrarsi su attività creative di alto valore che promuovono l'innovazione e generano un impatto aziendale. Esempi di attività ripetitive includono la generazione di codice standard, la creazione di dati di test, la stesura di documentazione o persino la stesura di piani di progetto iniziali. Affidando queste attività all'intelligenza artificiale, i membri del team possono concentrarsi su un lavoro più creativo e strategico. Ad esempio, gli strumenti di completamento del codice basati sull'intelligenza artificiale possono accelerare notevolmente il processo di codifica suggerendo frammenti di codice pertinenti in base al contesto e ai modelli di codifica. Allo stesso modo, l'intelligenza artificiale generativa può creare e aggiornare automaticamente la documentazione tecnica man mano che il codice cambia. Ciò mantiene la documentazione aggiornata e riduce lo sforzo manuale tipicamente richiesto per questa attività. Nei test, l'intelligenza artificiale può generare casi di test completi basati sui requisiti e sull'analisi del codice, il che migliora la copertura dei test e riduce la probabilità di casi limite trascurati. Automatizzando in modo intelligente queste attività ripetitive, l'intelligenza artificiale generativa accelera le tempistiche di sviluppo, migliora la coerenza e riduce l'errore umano. Il risultato sono output software di qualità superiore.

Revisione e iterazione periodiche dell'esperienza di sviluppo

La tua stessa esperienza di sviluppo software dovrebbe essere trattata come un prodotto che richiede continui perfezionamenti. Ciò comporta l'istituzione di un processo sistematico per la revisione e l'iterazione periodiche di tutti gli aspetti del ciclo di vita, degli strumenti e delle pratiche di sviluppo. Esegui valutazioni periodiche dell'intera toolchain, dei flussi di lavoro e dei processi. Raccogli feedback da tutti i membri del team in vari ruoli, tra cui responsabili di prodotto, designer, architetti, sviluppatori, tester e personale operativo. Chiedi loro di identificare i punti deboli, gli ostacoli e le opportunità di miglioramento. Ad esempio, i team potrebbero condurre revisioni trimestrali delle prestazioni della CI/CD pipeline e analizzare metriche quali tempi di costruzione, frequenza di implementazione e tassi di errore per identificare le aree di ottimizzazione. Poiché le funzionalità di intelligenza artificiale generativa continuano a evolversi rapidamente, è fondamentale valutare

costantemente nuovi strumenti e funzionalità basati sull'intelligenza artificiale che potrebbero semplificare ulteriormente i flussi di lavoro o aumentare le capacità in tutti i ruoli dell'SDLC.

Adottare pratiche efficaci di gestione dei progetti

Per orchestrare in modo efficace le tue complesse attività di sviluppo software, adotta pratiche di gestione dei progetti basate sull'intelligenza artificiale. In questo contesto, una gestione efficace dei progetti va oltre le metodologie tradizionali. Adotta approcci basati sull'intelligenza artificiale che migliorano la pianificazione, l'esecuzione e il monitoraggio nell'intero SDLC. I framework agili promuovono flessibilità, collaborazione e iterazione rapida e puoi utilizzare l'intelligenza artificiale generativa per ottimizzare questi processi. Ad esempio, l'intelligenza artificiale generativa può analizzare i dati storici dei progetti per stime più accurate, generare automaticamente e assegnare priorità alle storie degli utenti in base agli obiettivi aziendali e al feedback dei clienti e fornire informazioni intelligenti sulle prestazioni del team. Gli strumenti di gestione dei progetti basati sull'intelligenza artificiale possono prevedere potenziali ostacoli e suggerire l'assegnazione ottimale delle attività in base alle competenze e ai carichi di lavoro dei membri del team. Integrando le funzionalità basate sull'intelligenza artificiale nelle pratiche di gestione dei progetti, puoi ottenere una maggiore visibilità, prendere decisioni basate sui dati più velocemente e assicurarti che i membri del team siano allineati e lavorino in modo efficiente verso obiettivi comuni.

Attuazione della gestione della conoscenza

Man mano che la tua esperienza di sviluppo software basata sull'intelligenza artificiale matura, implementa un solido sistema di gestione della conoscenza. Un solido sistema di gestione delle conoscenze ti aiuta ad acquisire, organizzare e concedere l'accesso a informazioni, best practice e soluzioni preziose. Tutti i membri del team dell'SDLC dovrebbero avere un facile accesso al sistema. Utilizzate l'intelligenza artificiale generativa per creare basi di conoscenza dinamiche e intelligenti che si evolvono con la vostra organizzazione. Ad esempio, l'intelligenza artificiale può generare e aggiornare automaticamente la documentazione in base a modifiche al codice, conversazioni e artefatti del progetto in modo che le informazioni rimangano aggiornate senza l'intervento manuale. L'intelligenza artificiale generativa può anche potenziare funzionalità di ricerca intelligenti e aiutare i membri del team a trovare rapidamente le informazioni pertinenti utilizzando query in linguaggio naturale, anche se non conoscono la terminologia esatta. Inoltre, l'intelligenza artificiale generativa può fornire in modo proattivo informazioni pertinenti ai membri del team in base alle loro attività o sfide attuali. Agisce come un mentore virtuale che migliora il processo decisionale e la risoluzione dei problemi in tutti i ruoli. Implementando un sistema di gestione della conoscenza

basato sull'intelligenza artificiale, è possibile abbattere i silos, accelerare l'onboarding, ridurre il lavoro ridondante e promuovere una cultura di apprendimento e innovazione continui in tutto il team di sviluppo software.

Fornire estensibilità e personalizzazione

Per massimizzare i vantaggi dell'IA generativa nello sviluppo di software, assicurati che gli strumenti e le piattaforme basati sull'intelligenza artificiale siano estensibili e personalizzabili. Questo ti aiuta a personalizzare le funzionalità di intelligenza artificiale in base alle tue esigenze, ai flussi di lavoro e agli stack tecnologici specifici. Ad esempio, puoi ottimizzare i modelli di intelligenza artificiale in base alle tue basi di codice e alla tua documentazione, creare strumenti personalizzati basati sull'intelligenza artificiale per attività specifiche o integrare le funzionalità di intelligenza artificiale negli strumenti e nei processi esistenti. Questa estensibilità consente di far evolvere l'esperienza di sviluppo basata sull'intelligenza artificiale per soddisfare le mutevoli esigenze dell'organizzazione. Inoltre, consente di ottimizzare l'esperienza per domini o tipi di progetto specifici.

Ottimizzazione per le operazioni

L'intelligenza artificiale generativa svolge un ruolo cruciale nell'ottimizzazione delle operazioni e della manutenzione del software. Ottimizza le operazioni integrando le funzionalità di intelligenza artificiale nei tuoi strumenti e processi operativi. Ad esempio, utilizza l'intelligenza artificiale generativa per analizzare i dati di registro quasi in tempo reale, prevedere potenziali guasti del sistema e automatizzare le attività di manutenzione ordinaria. L'intelligenza artificiale generativa può anche aiutare con l'analisi delle cause principali correlando gli eventi tra sistemi distribuiti complessi. Ciò migliora l'affidabilità del sistema, riduce i tempi di inattività e consente ai team operativi di concentrarsi su iniziative più strategiche.

Utilizzo di approfondimenti basati sui dati

Utilizza approfondimenti basati sui dati durante il tuo percorso di sviluppo basato sull'intelligenza artificiale. Implementa sistemi per raccogliere, analizzare e sfruttare i dati provenienti da tutte le fasi dell'SDLC. Ciò include metriche del codice, risultati dei test, dati di implementazione, feedback degli utenti e prestazioni operative. Usa l'intelligenza artificiale generativa per scoprire modelli e approfondimenti che potrebbero non essere evidenti agli osservatori umani. Quindi, inserisci queste informazioni nel tuo processo di sviluppo per informarti su tutto, dalle decisioni sull'architettura alla prioritizzazione delle funzionalità.

Adottare un approccio basato sulla piattaforma

Per sfruttare appieno i vantaggi dell'intelligenza artificiale generativa nello sviluppo del software, adotta un approccio basato sulla piattaforma. Crea una piattaforma completa e integrata che incorpori funzionalità di intelligenza artificiale in tutti gli aspetti dell'SDLC. La piattaforma dovrebbe fornire un'esperienza utente coerente, gestione e dati centralizzati e una perfetta integrazione tra diversi strumenti e processi. Ciò rende i vantaggi dell'IA disponibili in modo uniforme in tutta l'organizzazione, riduce il sovraccarico derivante dalla gestione di strumenti di intelligenza artificiale multipli e disparati e fornisce le basi per il miglioramento e l'espansione continui delle funzionalità di intelligenza artificiale.

Misurare il successo dell'IA generativa nello sviluppo di software

Per misurare efficacemente l'effetto dell'implementazione di un'esperienza di sviluppo software generativa basata sull'intelligenza artificiale, è necessario stabilire un set completo di metriche che coprano varie dimensioni del ciclo di vita dello sviluppo del software (SDLC). Queste metriche dovrebbero rilevare miglioramenti immediati in termini di efficienza e produttività e riflettere anche i miglioramenti a lungo termine in termini di qualità del software, soddisfazione del team e valore aziendale.

Effettua le seguenti operazioni per utilizzare in modo efficace le metriche consigliate in questa sezione:

1. Stabilisci delle linee di base: prima di immergerti nell'implementazione della tua esperienza di sviluppo basata sull'intelligenza artificiale, dedica del tempo alla raccolta di dati completi sulle tue prestazioni attuali in base a queste metriche. Ciò fornisce un chiaro punto di partenza e ti aiuta a fare confronti significativi in un secondo momento.
2. Stabilisci obiettivi realistici: con le tue linee di base in mano, stabilisci obiettivi di miglioramento raggiungibili per ogni metrica. Sii ambizioso ma realistico. Ricorda che il progresso sostenibile è spesso incrementale.
3. Implementa il monitoraggio continuo: utilizza strumenti automatizzati per raccogliere e analizzare costantemente i dati relativi a queste metriche nel tuo ambiente. Il monitoraggio quasi in tempo reale consente di monitorare i progressi e identificare rapidamente eventuali problemi o opportunità.
4. Conduci revisioni regolari: pianifica sessioni di revisione trimestrali o semestrali in cui tu e il tuo team valutate attentamente i progressi compiuti rispetto agli obiettivi. Utilizza queste sessioni per identificare le aree di ulteriore miglioramento e celebrare i tuoi successi.
5. Iterate e modificate: sulla base delle informazioni acquisite, perfezionate continuamente l'implementazione dell'intelligenza artificiale generativa e aggiustate gli obiettivi secondo necessità.

Questa sezione descrive le seguenti categorie di metriche:

- [Velocità di distribuzione](#)
- [Qualità del codice](#)
- [Efficienza operativa](#)

- [Produttività e soddisfazione del team](#)
- [Impatto aziendale](#)

Velocità di distribuzione

Prendi in considerazione la misurazione delle seguenti metriche di velocità di implementazione.

Metrica	Description
Tempo di commercializzazione	Misura la riduzione del tempo che intercorre tra la concezione dell'idea e l'implementazione in produzione
Velocità di sprint	Tieni traccia dell'aumento dei punti storia completati dai tuoi team in ogni sprint
Frequenza di invio del codice	Monitora l'aumento dei commit di codice, il che indica un'accelerazione dei cicli di sviluppo
Tempo di risoluzione delle pull request	Valuta la riduzione del tempo impiegato per rivedere e unire le modifiche al codice nei tuoi repository
Velocità di rilascio	Misura l'aumento del numero di rilasci per trimestre o anno

Qualità del codice

Valuta la possibilità di misurare le seguenti metriche di qualità del codice.

Metrica	Description
Densità dei difetti	Misura la riduzione dei bug del software
Copertura del codice	Tieni traccia dell'aumento della percentuale di copertura dei test in tutta la tua codebase

Metrica	Description
Debito tecnico	Monitora la diminuzione del debito tecnico identificato nel tempo
Punteggi di analisi statica del codice	Valuta i miglioramenti della qualità del codice in base ai tuoi strumenti di analisi automatizzati

Efficienza operativa

Prendi in considerazione la misurazione delle seguenti metriche di efficienza operativa.

Metrica	Description
Frequenza di implementazione	Misura l'aumento del numero di implementazioni riuscite
Tempo medio di ripristino (MTTR)	Tieni traccia della riduzione del tempo necessario per il ripristino in seguito a guasti del sistema
Modifica la percentuale di errori	Monitora la diminuzione della percentuale di modifiche che provocano errori nelle distribuzioni

Produttività e soddisfazione del team

Valuta la possibilità di misurare le seguenti metriche di produttività e soddisfazione del team.

Metrica	Description
Miglioramento della produttività	Monitora l'aumento della percentuale di produttività per ogni attività

Metrica	Description
Punteggio di soddisfazione	Conduci sondaggi regolari per valutare il miglioramento del morale e della soddisfazione sul lavoro del tuo team
Efficienza nella condivisione delle conoscenze	Misura la riduzione del tempo impiegato dal team per cercare informazioni o porre domande ripetitive
Tempo di imbarco	Tieni traccia della riduzione del tempo necessario affinché i nuovi membri del team diventino produttivi

Impatto aziendale

Valuta la possibilità di misurare le seguenti metriche di impatto aziendale.

Metrica	Description
Tasso di adozione delle funzionalità	Misura l'aumento del coinvolgimento degli utenti con le nuove funzionalità che hai rilasciato
Punteggio di soddisfazione del cliente	Tieni traccia dei miglioramenti apportati al feedback e alle valutazioni degli utenti
Impatto sui ricavi (diretto e indiretto)	Valuta l'aumento dei ricavi attribuito all'aumento della velocità di rilascio o all'aumento della produttività

Conclusioni

Questo documento strategico fornisce una panoramica di un'esperienza di sviluppo software generativa basata sull'intelligenza artificiale. Esplora le cinque dimensioni del [framework 5-I](#): Investigate, Integrate, Interact, Iterate e Impact. Queste dimensioni forniscono una tabella di marcia strategica per l'integrazione dell'IA generativa nell'intero ciclo di vita dello sviluppo del software (SDLC). Descrive inoltre le [funzionalità di base necessarie per implementare con successo](#) questo framework. Le funzionalità abbracciano aree come la gestione dei progetti, gli assistenti di intelligenza artificiale DevSecOps, la gestione della conoscenza e altro ancora. Fornisce [le migliori pratiche](#) da considerare quando si integra l'IA generativa e aiuta a utilizzare le [metriche](#) per misurare l'impatto dell'intelligenza artificiale generativa sulla propria esperienza di sviluppo software.

L'integrazione dell'IA generativa nei processi di sviluppo del software rappresenta un cambio di paradigma che ha il potenziale per accelerare l'innovazione, migliorare la qualità e aumentare la produttività. Tuttavia, è importante riconoscere che non si tratta di un'implementazione una tantum. È un'evoluzione continua che richiede uno sforzo costante e un perfezionamento continuo.

Quando intraprendete questo viaggio, vi consigliamo di iniziare con una valutazione approfondita delle capacità e della preparazione attuali della vostra organizzazione. L'[AWS Assessment Tool è uno strumento](#) di valutazione dello sviluppo software basato sull'intelligenza artificiale che può aiutarvi a identificare le aree prioritarie e creare una roadmap di implementazione personalizzata.

Risorse

Una volta identificate le aree prioritarie chiave, le seguenti risorse possono aiutarvi a implementare la tua tabella di marcia:

AWS documentazione

- [Automatizza le operazioni AWS dell'infrastruttura utilizzando Amazon Bedrock](#) (AWS Prescriptive Guidance)
- [Best practice con Amazon Q Developer per la generazione di codice in linea e assistita](#) (AWS Prescriptive Guidance)
- [Sviluppa un assistente basato su chat completamente automatizzato utilizzando gli agenti e le knowledge base di Amazon Bedrock](#) (AWS Prescriptive Guidance)
- [Trasforma i modelli operativi di sviluppo e manutenzione delle applicazioni AWS con l'IA generativa \(Prescriptive Guidance\)](#) AWS

- [Usa Amazon Q Developer come assistente di programmazione per aumentare la tua produttività](#) (AWS Prescriptive Guidance)

AWS post di blog e tutorial

- [Post del blog di Amazon Q](#)
- [Accelera il ciclo di vita dello sviluppo del software con Amazon Q](#) (AWS post sul blog)
- [Creazione di un agente AI di AWS Solutions Architect: utilizzo di Amazon Bedrock per l'architettura e la distribuzione automatizzate \(video\)](#) AWS
- Operazioni [tecnologiche generative basate sull'intelligenza artificiale \(post](#) sul blog) AWS
- [Modernizza la tua applicazione Java con Amazon Q Developer](#) (post AWS sul blog)
- [Usa Amazon Bedrock per generare, valutare e comprendere il codice nella tua pipeline di sviluppo software](#) (AWS post sul blog)

Cronologia dei documenti

La tabella seguente descrive le modifiche significative apportate a questa guida. Per ricevere notifiche sugli aggiornamenti futuri, puoi abbonarti a un [feed RSS](#).

Modifica	Descrizione	Data
Pubblicazione iniziale	Non applicabile	18 aprile 2025

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.