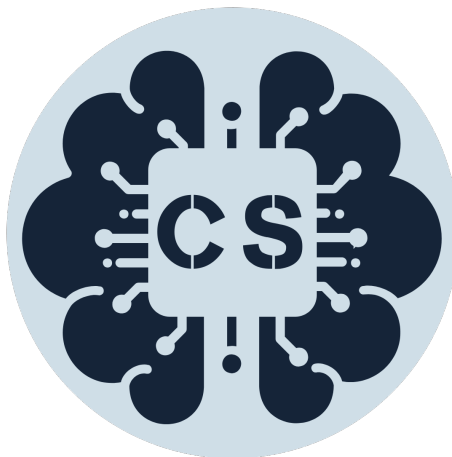


Glossario

CyberSorcerers Team



Membri del team:
Sabrina Caniato
Giulia Dentone
Nicola Lazzarin
Giovanni Moretti
Andrea Rezzi
Samuele Vignotto

Informazioni sul documento	
Destinatari:	Prf. Tullio Vardanega, Prf. Riccardo Cardin
Caratteristiche	Parole in ordine alfabetico
Indicate nei documenti con una "G" al pedice	

Indice

1 API	5
2 Attori	5
3 AWS	5
3.1 API Gateway	5
3.2 Bedrock	5
3.3 Cognito	6
3.4 Lambda-function	6
4 Baseline	6
5 Branch	6
6 Casi d'uso	6
7 Changelog	7
8 ChatGPT	7
9 Ciclo di vita	7
10Ciclo PDCA	8
11Cross Site Scripting	8
12Customer Acceptance	8
13Design	9
14Design Thinking	9
15Diagramma UML	9
16Epic stories	9
17Feature	10
18Firefox	10
19Framework Scrum	10
20Front-end	10
21Flow di Funzionamento	11
22GitHub	11

23 Google Chrome	11
24 IA	11
25 Injection	11
26 Issue	12
27 Issue tracking system	12
28 Javascript	12
29 Kotlin	13
30 Mappa dell'empatia	13
31 Microsoft Edge	13
32 Middleware	13
33 Modello Agile	14
34 Modello a V	14
35 Node.js	14
36 Open source	14
37 Performance	15
38 Plugin	15
39 Product Baseline	15
40 Project Manager	16
41 Proof of Concept	16
42 Protocollo SSL/TLS	16
43 Pull request	16
44 Repository	16
45 Requisiti di Business	17
46 Runtime system	17
47 Slack	17

48Stakeholder	17
49Swift	17
50Task	18
51Tag	18
52Technology Baseline	18
53Template	18
54Test	18
55Testing	19
56Trade Off	19
57Typescript	19
58User Stories	19
59Versione	19
60VSCode	20
61Way of working	20
62Web App	20
63XCode	21

1 API

API, acronimo di Application Programming Interface (Interfaccia di Programmazione delle Applicazioni), è un insieme di regole e protocolli che definiscono come i diversi componenti del software possono interagire tra loro. In sostanza, un'API determina quali operazioni e funzionalità un'applicazione rende disponibili per essere utilizzate da altre applicazioni o servizi.

Le API possono essere utilizzate per una vasta gamma di scopi, come l'accesso a dati o risorse da un server remoto, la comunicazione tra applicazioni diverse, l'integrazione di servizi di terze parti o la creazione di estensioni per applicazioni esistenti.

Le API possono essere di diversi tipi, tra cui API web basate su HTTP, API RESTful, API SOAP, API di libreria, e così via. Indipendentemente dal tipo, le API forniscono un modo standardizzato e strutturato per consentire la comunicazione e lo scambio di dati tra diverse applicazioni e servizi, favorendo l'integrazione e l'interoperabilità nel panorama software.

2 Attori

Nel contesto di un diagramma dei casi d'uso, un attore rappresenta un ruolo o una classe di utenti esterni al sistema che interagisce con il sistema stesso.

3 AWS

3.1 API Gateway

Amazon API Gateway è un servizio completamente gestito che semplifica per gli sviluppatori la creazione, la pubblicazione, la manutenzione, il monitoraggio e la protezione delle API su qualsiasi scala. Le API fungono da "porta di entrata" per consentire l'accesso delle applicazioni ai dati, alla logica aziendale o alle funzionalità dai servizi back-end. API Gateway consente di creare API RESTful e WebSocket che rendono possibili applicazioni di comunicazione bidirezionale in tempo reale. API Gateway supporta carichi di lavoro containerizzati e senza server, oltre che applicazioni Web.

3.2 Bedrock

Amazon Bedrock è un servizio completamente gestito che offre una scelta di modelli di fondazione (FM) ad alte prestazioni delle principali aziende di IA, come AI21 Labs, Anthropic, Cohere, Meta, Stability AI e Amazon, tramite un'unica API, insieme ad un'ampia gamma di funzionalità necessarie per creare applicazioni di IA generativa, utilizzando l'IA in modo sicuro, riservato e responsabile.

3.3 Cognito

Amazon Cognito è un servizio di gestione dell'identità e degli accessi dei clienti (CIAM) incentrato sullo sviluppatore e conveniente. Fornisce un archivio di identità sicuro e opzioni di federazione che possono scalare fino a milioni di utenti. Amazon Cognito supporta l'accesso con provider di identità sociali e provider di identità basati su SAML o OIDC per esperienze utente piacevoli e offre funzionalità avanzate di sicurezza per proteggere i tuoi clienti e la tua attività. Supporta vari standard di conformità, opera su standard di identità aperti (OAuth2.0, SAML 2.0 e OpenID Connect) e si integra con un ecosistema esteso di risorse di sviluppo front-end e back-end e librerie SDK.

3.4 Lambda-function

AWS Lambda-funztion è un servizio di elaborazione che esegue il codice in risposta agli eventi e gestisce automaticamente le risorse di elaborazione, rendendolo il modo più veloce per trasformare un'idea in moderne applicazioni di produzione serverless.

4 Baseline

Rappresenta uno stato o una configurazione di un sistema o di un progetto in un determinato momento, che viene utilizzato come riferimento fondamentale per misurare il progresso, valutare le modifiche e gestire la configurazione. Una baseline può includere documenti, codice sorgente, specifiche di progetto, pianificazioni, e altri elementi chiave. Le baseline vengono stabilite in fasi cruciali di un progetto e servono come punti di riferimento stabili per la gestione del cambiamento. Le principali baseline in un progetto software possono includere la baseline dei requisiti, la baseline dell'architettura, la baseline del design e la baseline del codice. Ogni baseline rappresenta uno stato verificato e accettato del progetto in un dato momento, fornendo una base solida per valutare eventuali modifiche future e garantire la coerenza e l'integrità del sistema nel corso del tempo.

5 Branch

Si riferisce a una diramazione del codice sorgente che viene creata per lavorare su una specifica issue o per sviluppare una nuova funzionalità.

6 Casi d'uso

Un diagramma dei casi d'uso è uno strumento di modellazione utilizzato nell'ambito dell'ingegneria del software per rappresentare graficamente le interazioni tra un sistema e gli attori esterni che interagiscono con esso.

7 Changelog

Un "changelog" è un documento o un registro che traccia e descrive le modifiche apportate a un software o a un progetto nel corso del tempo. Questo registro fornisce una cronologia dettagliata delle versioni del software, elencando le nuove funzionalità aggiunte, le correzioni di bug, le miglioramenti e qualsiasi altra modifica rilevante. Il changelog è un componente essenziale nella gestione del software, poiché fornisce una visione chiara e strutturata delle modifiche effettuate. Queste informazioni sono utili per gli sviluppatori, gli utenti e altri stakeholder, in quanto consentono di comprendere come il software è evoluto nel tempo e forniscono indicazioni su cosa aspettarsi in ogni nuova versione.

8 ChatGPT

ChatGPT è un'intelligenza artificiale sviluppata da OpenAI, basata sull'architettura GPT (Generative Pre-trained Transformer), progettata per generare testo umanistico coerente e comprensibile. ChatGPT può essere utilizzato per una varietà di scopi, tra cui la generazione di risposte a domande, la creazione di contenuti creativi, l'assistenza virtuale e altro ancora. Con una vasta conoscenza di diverse tematiche e la capacità di apprendere da grandi quantità di testo, ChatGPT è in grado di interagire con gli utenti in modo naturale e fornire risposte informative e pertinenti.

9 Ciclo di vita

Il "ciclo di vita del software" è il periodo che inizia con la concezione di un'applicazione o di un sistema software e si estende attraverso le fasi di sviluppo, test, implementazione, manutenzione e, infine, il ritiro o l'aggiornamento del software. Questo concetto rappresenta il percorso completo che un software percorre dalla sua creazione fino alla sua conclusione o evoluzione successiva. Le fasi principali del ciclo di vita del software includono:

- Concezione: Identificazione delle esigenze e dei requisiti del software, definizione degli obiettivi e della portata del progetto;
- Sviluppo: Progettazione e implementazione del software in base ai requisiti definiti nella fase di concezione;
- Test: Validazione e verifica del software attraverso test funzionali, di sistema e di accettazione per garantire che soddisfi gli standard di qualità e risponda alle aspettative degli utenti;
- Implementazione: Distribuzione e installazione del software negli ambienti operativi previsti;
- Manutenzione: Fase in cui vengono apportate modifiche, correzioni di bug e miglioramenti al software in risposta a nuovi requisiti o a cambiamenti nell'ambiente operativo;

- Ritiro o Aggiornamento: Il software viene ritirato se non è più utile o supportato, oppure viene sottoposto a aggiornamenti o evoluzioni per adeguarlo a nuovi requisiti o tecnologie.

Queste fasi possono variare a seconda del modello di sviluppo del software adottato, come il modello a cascata, il modello a spirale, o l'approccio agile. La gestione efficace del ciclo di vita del software è essenziale per garantire il successo del progetto, la qualità del prodotto e la soddisfazione degli utenti.

10 Ciclo PDCA

Il "ciclo PDCA" è un modello di gestione ciclico che rappresenta un approccio iterativo per il miglioramento continuo di processi e prodotti. L'acronimo PDCA sta per Plan-Do-Check-Act, che sono le quattro fasi chiave del ciclo:

- Plan (Pianificare): In questa fase, si pianificano e si stabiliscono gli obiettivi e le attività necessarie per raggiungerli. Si identificano i problemi, si definiscono le soluzioni, e si sviluppa un piano d'azione;
- Do (Eeguire): Durante questa fase, si mette in atto il piano d'azione sviluppato nella fase di pianificazione. Si implementano le soluzioni e si raccoglie dati e informazioni pertinenti;
- Check (Verificare): In questa fase, si valutano i risultati ottenuti mediante la raccolta e l'analisi di dati. Si confrontano i risultati con gli obiettivi prefissati, si identificano le eventuali deviazioni e si valutano le cause di eventuali problemi;
- Act (Agire): Basandosi sui risultati della fase di verifica, si prendono azioni correttive o preventive. Queste azioni possono includere miglioramenti ai processi, l'aggiornamento delle procedure o la revisione dei piani di azione.

11 Cross Site Scripting

Cross-Site Scripting (XSS), spesso abbreviato come XSS, è una vulnerabilità delle applicazioni web che consente a un attaccante di inserire script dannosi all'interno delle pagine web visualizzate da altri utenti.

12 Customer Acceptance

"Customer Acceptance", o accettazione del cliente, si riferisce al processo attraverso il quale un cliente verifica e conferma che un prodotto o servizio soddisfa i suoi requisiti, aspettative e specifiche concordate. È il momento in cui il cliente valuta il lavoro consegnato e decide se accettarlo o rifiutarlo in base a criteri prestabiliti. L'accettazione del cliente è un passaggio critico in molti progetti, in quanto conferma che il lavoro è stato completato con successo e che il cliente è soddisfatto del risultato finale.

13 Design

Il "Design" è il processo di creazione e pianificazione di soluzioni per risolvere problemi o soddisfare esigenze specifiche. Coinvolge la progettazione di prodotti, servizi, sistemi o esperienze che siano funzionali, esteticamente gradevoli e adatti all'uso previsto. Il design comprende la definizione di requisiti, la generazione di idee, la prototipazione, il testing e l'iterazione per raggiungere risultati ottimali. In sintesi, il design mira a trovare soluzioni efficaci e innovative per migliorare la vita delle persone e rispondere alle loro esigenze.

14 Design Thinking

Il "Design thinking" è un approccio innovativo e centrato sull'utente alla risoluzione dei problemi e alla generazione di soluzioni creative. Questa metodologia si basa sulla collaborazione interdisciplinare, integrando aspetti provenienti dal design, dall'analisi delle esigenze degli utenti e dalla prototipazione rapida. L'obiettivo principale del design thinking è comprendere profondamente le necessità e le prospettive degli utenti, stimolare la creatività del team e iterare rapidamente attraverso diverse soluzioni per raggiungere una risposta ottimale al problema in esame. Inoltre, il design thinking promuove l'empatia, il pensiero aperto e la sperimentazione, incoraggiando i professionisti a considerare molteplici punti di vista al fine di sviluppare soluzioni più innovative e orientate al contesto utente.

15 Diagramma UML

Un diagramma UML (Unified Modeling Language) è una rappresentazione visuale di un sistema software utilizzata per comprendere, progettare e comunicare la struttura, il comportamento e le relazioni all'interno del sistema stesso. Viene utilizzato per modellare diverse prospettive di un'applicazione software, inclusi gli aspetti strutturali come le classi e le relazioni tra di esse, nonché gli aspetti comportamentali come i casi d'uso e le sequenze di attività. I diagrammi UML forniscono uno standard convenzionale e una lingua comune per i professionisti del software per comunicare idee e concetti relativi al sistema che stanno progettando o analizzando.

16 Epic stories

Un' Epic story è una storia di dimensioni maggiori rispetto a una singola User Story. Le Epic story forniscono una visione ad alto livello dei requisiti di un progetto.

17 Feature

Si riferisce a una funzionalità specifica o a un attributo distintivo di un software, di un'applicazione o di un sistema. Le features sono elementi distinti che contribuiscono alle capacità complessive di un prodotto e possono includere aspetti come la gestione dei dati, le interfacce utente, le prestazioni, la sicurezza, o qualsiasi altra funzione che fornisca un valore specifico agli utenti o ai gestori del sistema.

18 Firefox

Firefox è un browser web open source sviluppato da Mozilla Corporation. Conosciuto per la sua enfasi sulla privacy e la sicurezza, Firefox offre una navigazione rapida e affidabile. È caratterizzato da un'interfaccia pulita e personalizzabile, con una vasta gamma di estensioni disponibili per adattare l'esperienza di navigazione alle esigenze degli utenti. Firefox è noto anche per il suo impegno verso gli standard aperti e la trasparenza, offrendo agli utenti un'alternativa versatile e rispettosa della privacy rispetto ad altri browser.

19 Framework Scrum

Lo Scrum è un framework agile ampiamente utilizzato nel campo dello sviluppo software e della gestione dei progetti. Creato per favorire la flessibilità, la trasparenza e la collaborazione all'interno di un team, lo Scrum offre una struttura organizzativa che facilita la gestione di complessi processi di sviluppo. Il framework Scrum si basa su principi chiave, come la suddivisione del lavoro in brevi iterazioni chiamate "sprint", solitamente della durata di due settimane, e la definizione di ruoli chiave all'interno del team. Le pratiche fondamentali dello Scrum includono la pianificazione degli sprint, la revisione dello sprint e la retrospettiva dello sprint. La pianificazione del lavoro avviene attraverso la creazione di un backlog di prodotto, una lista prioritizzata di funzionalità o attività da svolgere. Durante lo sprint, il team si impegna a completare una porzione di questo backlog, producendo un incremento di prodotto al termine di ogni iterazione.

20 Front-end

Il front-end si riferisce alla parte di un'applicazione web o mobile con cui gli utenti interagiscono direttamente. È responsabile della presentazione e dell'interfaccia utente dell'applicazione, inclusi layout, colori, testo, immagini, pulsanti e tutti gli altri elementi visibili e interattivi.

21 Flow di Funzionamento

Il "flow di funzionamento" si riferisce al flusso di lavoro o alla sequenza di passaggi che vengono eseguiti per completare un compito o un processo specifico.

22 GitHub

GitHub è una piattaforma di hosting di codice sorgente basata su Git, che consente agli sviluppatori di collaborare, gestire e condividere il loro lavoro in modo efficiente. GitHub offre agli sviluppatori una vasta gamma di strumenti per gestire il controllo delle versioni del loro codice, facilitare la collaborazione tra team di sviluppo e coordinare il processo di sviluppo del software. Oltre al controllo delle versioni, GitHub offre funzionalità come problemi e richieste di pull per la gestione dei progetti, wiki per la documentazione, e una serie di strumenti per l'integrazione continua e il rilascio continuo (CI/CD). Con la sua ampia comunità di sviluppatori e il suo vasto ecosistema di progetti open source, GitHub è diventato uno degli strumenti più utilizzati nel mondo dello sviluppo software moderno.

23 Google Chrome

Google Chrome è un browser web sviluppato da Google, caratterizzato da velocità, stabilità e una vasta gamma di funzionalità. È noto per il suo motore di rendering avanzato basato su Chromium, che offre un'esperienza di navigazione fluida e reattiva. Chrome offre anche sincronizzazione tra dispositivi, gestione dei segnalibri intuitiva e un ampio supporto per gli standard web. È apprezzato per la sua sicurezza, grazie ai continui aggiornamenti e alle funzionalità di protezione dai siti dannosi. Con la sua interfaccia pulita e user-friendly, Google Chrome è diventato uno dei browser più popolari al mondo, sia su computer desktop che su dispositivi mobili.

24 IA

Acronimo per "Intelligenza Artificiale".

25 Injection

Un'injection si riferisce a una tecnica di attacco informatico in cui dati dannosi vengono inseriti in un'applicazione o un sistema al fine di comprometterne il funzionamento o ottenere un accesso non autorizzato.

26 Issue

Un'issue rappresenta un singolo problema, un'idea, una richiesta di funzionalità o un bug nel contesto di un progetto software ospitato su piattaforme di gestione del codice come GitHub. Ogni issue ha un numero univoco, un titolo descrittivo e un corpo testuale dettagliato che fornisce ulteriori informazioni sul problema in questione.

27 Issue tracking system

Un "issue tracking system" (sistema di tracciamento delle problematiche) è uno strumento software utilizzato per registrare, monitorare e gestire le problematiche, i problemi o i task all'interno di un progetto o di un sistema. Le caratteristiche principali di un issue tracking system includono:

- Consentire agli utenti di registrare in modo strutturato le problematiche, fornendo dettagli come descrizione, priorità, assegnazione, e altri dati pertinenti;
- Assegnare le problematiche ai membri del team responsabili e monitorare lo stato di avanzamento nel tempo. Questo aiuta a garantire che ogni problema venga gestito in modo efficace e tempestivo;
- Fornire la possibilità di aggiungere commenti, note e allegati alle problematiche, consentendo una comunicazione efficace e il condividere di informazioni pertinenti;
- Permettere di classificare le problematiche in base a categorie specifiche, priorità o altri criteri rilevanti per l'organizzazione;
- Conservare un registro delle modifiche apportate alle problematiche nel tempo, consentendo la tracciabilità delle attività e delle decisioni;
- Generare report e dashboard per visualizzare statistiche, tendenze e metriche sulle problematiche, facilitando la valutazione delle prestazioni del progetto.

28 Javascript

JavaScript è un linguaggio di programmazione ad alto livello e interpretato, ampiamente utilizzato per lo sviluppo di applicazioni web interattive e dinamiche.

JavaScript è principalmente utilizzato per aggiungere interattività e dinamicità alle pagine web, consentendo agli sviluppatori di manipolare il contenuto HTML e CSS, rispondere agli eventi utente e comunicare con i server per ottenere o inviare dati senza dover ricaricare l'intera pagina.

JavaScript è un linguaggio di programmazione multiparadigma, supportando sia lo stile di programmazione orientato agli oggetti che funzionale.

29 Kotlin

Kotlin è un linguaggio di programmazione moderno e multiplatforma, progettato per essere interoperabile con Java e completamente supportato da Google per lo sviluppo di app Android.

Kotlin offre molte caratteristiche innovative, tra cui la null safety, che aiuta a prevenire errori NullPointerException durante l'esecuzione del programma, e le estensioni delle funzioni, che consentono di aggiungere nuove funzionalità alle classi esistenti senza doverle sottoporre a ereditarietà. Inoltre, il suo sistema di tipi statici e inferenza automatica dei tipi riduce al minimo il lavoro ripetitivo e aumenta la robustezza del codice.

30 Mappa dell'empatia

La "mappa dell'empatia" è uno strumento di visualizzazione utilizzato nel design thinking e nel processo di sviluppo di prodotti o servizi. Solitamente è divisa in sezioni che rappresentano differenti aspetti dell'esperienza dell'utente. La mappa dell'empatia è spesso realizzata durante sessioni di lavoro collaborative, coinvolgendo membri del team di progettazione, responsabili del prodotto e, se possibile, direttamente gli utenti.

31 Microsoft Edge

Microsoft Edge è un browser web sviluppato da Microsoft. È stato introdotto per la prima volta nel 2015 come successore di Internet Explorer, offrendo una navigazione più veloce, una migliore compatibilità con gli standard web e una serie di funzionalità innovative. Edge è basato sul motore di rendering Blink, lo stesso utilizzato da Google Chrome, garantendo un'esperienza di navigazione fluida e reattiva. È noto per le sue funzionalità di annotazione e di lettura avanzata, che consentono agli utenti di prendere appunti, evidenziare testi e organizzare le proprie ricerche in modo efficace. Edge è integrato anche con altre funzionalità di Microsoft, come Cortana e Microsoft 365, offrendo un'esperienza integrata e sincronizzata per gli utenti Windows.

32 Middleware

Il "middleware" è uno strato di software che agisce come intermediario tra il sistema operativo e le applicazioni, facilitando la comunicazione e lo scambio di dati tra diverse componenti di un sistema distribuito o di una rete. Essenzialmente, il middleware svolge un ruolo chiave nell'orchestrare e facilitare l'integrazione di diverse applicazioni e servizi. Le funzioni principali del middleware includono la gestione della comunicazione tra sistemi software, la gestione della distribuzione delle risorse, la sicurezza, la gestione degli errori e la fornitura di servizi di supporto per lo sviluppo di applicazioni distribuite.

33 Modello Agile

Il modello Agile nell'ingegneria del software è un approccio metodologico che promuove lo sviluppo incrementale e iterativo del software, focalizzandosi sulla collaborazione, la flessibilità e la risposta rapida ai cambiamenti nei requisiti del cliente. Questa metodologia si basa su un insieme di principi e valori definiti nel Manifesto Agile, che sottolinea l'importanza di individui e interazioni, software funzionante, collaborazione con il cliente e risposta positiva al cambiamento. Le caratteristiche chiave del modello Agile includono la pianificazione a breve termine attraverso iterazioni chiamate "sprint", la comunicazione continua tra team di sviluppo e stakeholder, il coinvolgimento attivo del cliente durante tutto il processo di sviluppo e la capacità di adattarsi rapidamente alle modifiche dei requisiti. Questo approccio mira a fornire prodotti software di alta qualità in modo efficiente, garantendo al contempo la soddisfazione del cliente e la consegna tempestiva di valore.

34 Modello a V

Il modello a V è un approccio di sviluppo software che organizza le fasi del ciclo di vita del software in una forma a "V". Le fasi di sviluppo procedono dal livello di specifica più astratto (in alto sulla "V") a quello più dettagliato (in basso sulla "V"), poi risalgono attraverso le fasi di testing e validazione. Le fasi superiori della "V" includono l'analisi dei requisiti, la progettazione architetturale e la progettazione dettagliata, mentre le fasi inferiori comprendono l'implementazione, l'integrazione e i vari livelli di testing. Questo modello promuove una forte correlazione tra ciascuna fase di sviluppo e la sua controparte di verifica e validazione, garantendo che il prodotto finale soddisfi i requisiti specificati e sia testato in modo completo.

35 Node.js

Node.js è un ambiente di runtime open source basato su JavaScript che consente agli sviluppatori di eseguire codice JavaScript lato server. Utilizzando il motore JavaScript V8 di Google Chrome, Node.js consente di eseguire codice JavaScript direttamente sul server anziché solo nel browser.

36 Open source

"Open source" si riferisce a software la cui licenza permette agli utenti di accedere, modificare e distribuire il codice sorgente liberamente. Questo modello favorisce la collaborazione, l'innovazione e la trasparenza, consentendo a sviluppatori di tutto il mondo di contribuire al miglioramento e all'evoluzione del software. L'approccio open source promuove una comunità di sviluppatori che lavorano insieme per creare soluzioni software accessibili e adattabili a una vasta gamma di esigenze.

37 Performance

In ambito software, il termine "performance" si riferisce all'efficienza e alle prestazioni generali di un'applicazione o di un sistema informatico. Gli aspetti chiave delle performance nel contesto software includono:

- Velocità di esecuzione: Misura quanto rapidamente un'applicazione può completare le sue operazioni, ad esempio, il tempo di risposta per eseguire una funzione o elaborare una richiesta;
- Utilizzo delle risorse: Valuta quanto efficientemente l'applicazione utilizza le risorse di sistema, come la memoria e l'archiviazione, evitando sprechi e garantendo un utilizzo ottimale delle risorse disponibili;
- Scalabilità: Indica la capacità di un'applicazione di gestire un aumento del carico di lavoro o delle richieste senza degradare significativamente le prestazioni;
- Stabilità: Si riferisce alla capacità di un'applicazione di mantenere un funzionamento affidabile e privo di errori sotto varie condizioni, senza crash o comportamenti imprevisti;
- Efficienza energetica: Misura il consumo energetico dell'applicazione o del sistema, essenziale soprattutto in ambienti in cui l'efficienza energetica è una priorità.

38 Plugin

Un "plugin" è un componente software aggiuntivo progettato per estendere le funzionalità di un'applicazione o di un programma principale. Si integra con il software esistente per fornire nuove caratteristiche, miglioramenti o capacità specializzate, senza richiedere modifiche sostanziali al nucleo dell'applicazione.

39 Product Baseline

La "Product Baseline" è un concetto utilizzato nella gestione dei progetti per identificare lo stato verificato e accettato di un prodotto o di un insieme di prodotti all'interno di un progetto. Indica un punto di riferimento stabile e consolidato, che rappresenta il risultato del lavoro svolto fino a quel momento, inclusi requisiti, specifiche, progettazione e implementazione del prodotto. La "Product Baseline" fornisce una base solida e chiara per valutare il progresso del progetto, coordinare le attività future e garantire che il prodotto soddisfi le aspettative e i requisiti del cliente.

40 Project Manager

Il Project Manager, o responsabile del progetto, è una figura chiave all'interno di un'organizzazione incaricata di pianificare, eseguire e concludere progetti. Il suo ruolo è multifunzionale e coinvolge diverse responsabilità per garantire che un progetto venga completato in modo efficace, efficiente e in linea con gli obiettivi prefissati.

41 Proof of Concept

Una proof of concept in ingegneria del software è una realizzazione pratica e limitata di un sistema o di una sua parte che mira a dimostrare la fattibilità tecnica o la validità di un concetto specifico. Questo concetto può riguardare l'architettura del software, l'integrazione di tecnologie, l'efficacia di un algoritmo, o qualsiasi altro aspetto chiave del progetto.

42 Protocollo SSL/TLS

Il Secure Sockets Layer (SSL) e il suo successore, il Transport Layer Security (TLS), sono protocolli crittografici utilizzati per garantire la sicurezza delle comunicazioni su Internet. Il protocollo SSL/TLS consente l'autenticazione delle parti coinvolte, la cifratura dei dati trasmessi e l'integrità dei messaggi scambiati tra client e server su una rete.

43 Pull request

La pull request, all'interno dell'issue tracking system, contiene le modifiche apportate al progetto, fornendo una panoramica delle modifiche e facilitando la revisione da parte di altri sviluppatori o responsabili del progetto.

44 Repository

Spazio di archiviazione digitale che contiene e gestisce un insieme organizzato di file, documenti o dati. Nei contesti di sviluppo software, il termine si riferisce comunemente a un "repository di codice sorgente", che è una struttura in cui viene conservato e versionato il codice di un progetto software. Un repository di codice sorgente consente agli sviluppatori di collaborare, tenere traccia delle modifiche nel tempo e gestire lo sviluppo del software in modo coordinato. Ogni modifica al codice viene registrata come una nuova "versione", consentendo agli sviluppatori di monitorare il progresso, risalire alle versioni precedenti e gestire eventuali conflitti.

45 Requisiti di Business

I requisiti di business sono le specifiche delle esigenze aziendali che guidano il processo di progettazione e sviluppo di un sistema, prodotto o servizio.

46 Runtime system

Un "Runtime system" è l'ambiente di esecuzione in cui un programma software viene eseguito durante l'esecuzione. Questo sistema fornisce le risorse e l'infrastruttura necessarie per l'esecuzione del programma, inclusi processi, thread, gestione della memoria, gestione delle risorse di sistema e altre funzionalità di supporto. In breve, il "Runtime system" è responsabile di tradurre e gestire l'esecuzione del codice sorgente del programma in istruzioni eseguibili dal computer, garantendo che il software funzioni correttamente e in modo efficiente durante l'esecuzione.

47 Slack

Slack è una piattaforma di comunicazione aziendale basata su cloud progettata per facilitare la collaborazione e la comunicazione tra team e individui all'interno di un'organizzazione. Offre funzionalità di messaggistica istantanea, chat di gruppo, chiamate vocali e video, condivisione di file e integrazioni con altre applicazioni e servizi. Slack è ampiamente utilizzato nelle aziende di tutto il mondo per migliorare la produttività, ridurre la dipendenza dalle e-mail e favorire una comunicazione più efficace e organizzata tra i membri del team.

48 Stakeholder

Un individuo, un gruppo o un'organizzazione che ha un interesse diretto o indiretto nel successo di un progetto software. Gli stakeholder includono, ma non sono limitati a, clienti, utenti finali, responsabili del progetto, sviluppatori, tester, dirigenti, responsabili delle risorse umane e qualsiasi altra parte coinvolta nel ciclo di vita del software.

49 Swift

Swift è un linguaggio di programmazione sviluppato da Apple nel 2014 per la creazione di app iOS, macOS, watchOS e tvOS. È progettato per essere moderno, sicuro, veloce e facile da imparare. Swift offre un'eccellente sintassi leggibile, potenti funzionalità di sicurezza dei tipi e prestazioni ottimizzate, rendendolo una scelta preferita per gli sviluppatori che lavorano nell'ecosistema Apple.

50 Task

In informatica, un "task" è un'unità di lavoro specifica o un compito assegnato che deve essere completato all'interno di un sistema o di un'applicazione software.

51 Tag

Un "tag" è un'etichetta associata a un elemento di dati, spesso utilizzata per organizzare e categorizzare informazioni. I tag sono utilizzati in diversi contesti, come la gestione di file, la catalogazione di contenuti web, eccetera. Possono essere parole chiave o identificatori brevi che forniscono informazioni specifiche sulla natura o il contenuto di un elemento. L'uso di tag facilita la ricerca, il filtraggio e l'organizzazione dei dati, migliorando l'efficienza nella gestione delle informazioni digitali.

52 Technology Baseline

La "Technology Baseline" è un termine utilizzato nella gestione dei progetti per indicare una base tecnologica consolidata e stabile su cui basare lo sviluppo futuro del progetto. Include una serie di tecnologie, strumenti e risorse che sono state valutate, selezionate e integrate nel progetto in base ai requisiti e agli obiettivi prefissati. La "Technology Baseline" fornisce una solida fondazione per la realizzazione del progetto, riducendo il rischio di cambiamenti improvvisi e fornendo una guida chiara per lo sviluppo e l'implementazione delle soluzioni tecnologiche necessarie.

53 Template

Modello o uno stampo predefinito che fornisce una struttura di base per la creazione di documenti, pagine web, codice o altri elementi digitali.

54 Test

Il "Test" è un processo di valutazione sistematica e controllata di un sistema, un componente o un prodotto per verificare che soddisfi i requisiti specificati e funzioni correttamente. Consiste nell'esecuzione di una serie di operazioni pianificate, chiamate testcase, al fine di rilevare difetti o anomalie nel software o nell'hardware. I test vengono utilizzati per garantire la qualità, l'affidabilità e la conformità di un prodotto rispetto agli standard stabiliti, consentendo agli sviluppatori di identificare e correggere eventuali errori prima del rilascio finale.

55 Testing

Si riferisce al processo sistematico di valutazione e verifica di un software o di un sistema informatico per identificare errori, problemi o discrepanze rispetto ai requisiti specificati. L'obiettivo del testing è garantire che il software sia robusto, affidabile e in grado di soddisfare le aspettative degli utenti e gli obiettivi del progetto.

56 Trade Off

Il termine "trade-off" si riferisce a una situazione in cui si verificano vantaggi e svantaggi o compromessi tra due o più opzioni o obiettivi. In altre parole, implicano la scelta di rinunciare a qualcosa in favore di qualcos'altro.

57 Typescript

TypeScript è un linguaggio di programmazione open source sviluppato da Microsoft. Si tratta di un superset di JavaScript, il che significa che tutte le funzionalità di JavaScript sono supportate in TypeScript, ma con l'aggiunta di tipi statici opzionali e altre caratteristiche avanzate.

I tipi statici consentono agli sviluppatori di definire il tipo di dati di variabili, parametri di funzione e strutture dati, migliorando la robustezza e la manutenibilità del codice. Questa caratteristica consente anche agli sviluppatori di rilevare errori nel codice durante la fase di sviluppo anziché a tempo di esecuzione, contribuendo così a prevenire bug e semplificare il processo di debugging.

Inoltre, TypeScript supporta molte delle caratteristiche avanzate dei linguaggi di programmazione orientati agli oggetti, come le classi, le interfacce e i moduli, che favoriscono una migliore organizzazione del codice e la sua riutilizzabilità.

58 User Stories

Una User Story rappresenta un requisito o una funzionalità dal punto di vista dell'utente. È una breve descrizione di una caratteristica desiderata, scritta in modo da essere comprensibile agli stakeholder e orientata al valore per l'utente finale.

59 Versione

Si riferisce a una specifica iterazione o release del programma che ha subito modifiche rispetto alla sua precedente.

60 VSCode

Visual Studio Code (VSCode) è un popolare editor di codice sorgente sviluppato da Microsoft. Si distingue per la sua interfaccia utente pulita e personalizzabile, che offre una vasta gamma di funzionalità per migliorare il flusso di lavoro degli sviluppatori. Dotato di una vasta gamma di estensioni disponibili tramite il suo sistema di marketplace integrato, VSCode supporta numerosi linguaggi di programmazione e framework, rendendolo estremamente versatile e adatto a una vasta gamma di progetti software. Grazie alla sua natura open source e alla sua comunità attiva, VSCode continua a essere uno strumento di riferimento per gli sviluppatori di tutto il mondo.

61 Way of working

In un contesto di ingegneria del software, il termine "way of working" si riferisce al metodo e alle pratiche che guidano l'approccio complessivo adottato per lo sviluppo del software all'interno di un progetto. Questo comprende le metodologie, i processi, gli strumenti e le interazioni che definiscono come il team di sviluppo organizza e gestisce il lavoro. La "way of working" in ingegneria del software può includere aspetti come:

- Specifica l'approccio metodologico adottato per pianificare, progettare, implementare e testare il software;
- Descrive le fasi e le attività coinvolte nello sviluppo del software, inclusi processi di codifica, testing, revisione del codice, gestione delle modifiche e rilascio del prodotto;
- Indica gli strumenti, le piattaforme e le tecnologie utilizzate per supportare il processo di sviluppo, inclusi ambienti di sviluppo, sistemi di gestione di versione (come Git), sistemi di build e strumenti di testing;
- Riguarda le modalità di comunicazione e collaborazione all'interno del team di sviluppo, inclusi incontri regolari, strumenti di comunicazione, e pratiche di condivisione delle informazioni;
- Include l'approccio alla pianificazione del progetto, la gestione delle risorse, il monitoraggio e il controllo delle attività, e la gestione dei rischi.

62 Web App

La web app è un'applicazione software che viene eseguita su un browser web e che interagisce con gli utenti attraverso l'interfaccia utente fornita dal browser stesso.

63 XCode

Xcode è un ambiente di sviluppo integrato (IDE) sviluppato da Apple per la creazione di software per i dispositivi del suo ecosistema.

Xcode integra un editor di codice avanzato con funzionalità di evidenziazione della sintassi, completamento automatico, debugging integrato e strumenti per la gestione del versionamento del codice attraverso Git.

Grazie alla sua integrazione con le tecnologie proprietarie di Apple, Xcode è lo strumento principale per lo sviluppo di software per l'ecosistema Apple.