



ByteOps.swe@gmail.com

Glossario

Informazioni documento

Redattori	A. Barutta R.Smanio E.Hysa L. Skenderi F.Pozza
Verificatori	E. Hysa A.Barutta N.Preto D.Diotto L.Skenderi
Destinatari	ByteOps T. Vardanega R. Cardin

Registro delle modifiche

Versione	Data	Autore	Verificatore	Dettaglio
0.0.5	27/12/2023	E. Hysa	L. Skenderi	Aggiunta termini per presentazione RTB
.0.4	08/12/2023	D. Diotto	R. Smanio	Aggiunta termini metriche
0.0.3	28/11/2023	L. Skenderi	D. Diotto	Aggiunti i termini riguardanti le tecnologie utilizzate
0.0.2	22/11/2023	A. Barutta	E. Hysa	Aggiunti i termini riguardanti Github
0.0.1	20/11/2023	A. Barutta	E. Hysa	Aggiunti i termini del capitolato

Indice

ByteOps

Contents

A.	4
B.	6
C.	7
D.	9
E.	11
F.	12
G.	13
I.	14
J.	15
L.	16
M.	17
N.	18
O.	19
P.	20
R.	22
S.	23
T.	25
U.	26
V.	27
W.	28

Sommario

Il presente documento fornisce descrizioni dettagliate ai termini tecnici e/o ambigui presenti nella documentazione. La sua utilità primaria risiede nella chiarezza e nella standardizzazione del linguaggio adoperato all'interno di documenti, testi o comunicazioni. Un glossario consente di facilitare la comprensione e la comunicazione tra diversi partecipanti o *stakeholder_G* coinvolti nel medesimo progetto.

A

Actual cost: (Descrizione più approfondita rispetto a quella presente in *Norme di Progetto* nella sezione Metriche di qualità). Una metrica che valuta i costi reali sostenuti sin dall'inizio del progetto fino al presente.

Analisi dei requisiti: L'*analisi dei requisiti_G* è una fase fondamentale nello sviluppo del *software_G* che coinvolge la raccolta, l'analisi, la documentazione e la comprensione approfondita delle esigenze e delle specifiche di un *sistema_G* o di un'applicazione *software_G* che si intende sviluppare (oltre ad essere un documento all'interno del nostro progetto).

Apache kafka: Apache Kafka è una *piattaforma_G* di streaming distribuito *open-source_G* utilizzata per la gestione e l'elaborazione di feed di dati in tempo reale. Progettato per affrontare problemi di ingestione, archiviazione e trasmissione di grandi volumi di dati in modo scalabile, Apache Kafka è ampiamente utilizzato nell'ambito del *data streaming_G* e dell'elaborazione degli eventi.

API: Un'*API_G*, acronimo di Application Programming Interface, è un insieme di regole, protocolli e strumenti che consente a diverse applicazioni *software_G* di comunicare tra loro. In sostanza, fornisce un modo standardizzato per diverse parti del *software_G* di interagire e scambiare informazioni.

Architettura: L'*architettura_G* rappresenta il tessuto connettivo, le relazioni e le restrizioni che coordinano e organizzano i diversi componenti di un prodotto *software_G*. Essenzialmente, definisce la struttura generale, il modo in cui i vari elementi interagiscono tra loro e le modalità di comunicazione all'interno del *sistema_G*.

Arg: L'istruzione ARG viene utilizzata per definire variabili di build che possono essere utilizzate nel *Dockerfile_G* durante la creazione dell'immagine.

Artefatto: Gli Artefatti sono elementi generati o prodotti durante il processo di sviluppo del *software_G*. Gli Artefatti rappresentano risultati intermedi o finali di un processo, che possono essere file, documenti, codice compilato, *librerie_G*, o qualsiasi altro elemento creato durante lo sviluppo o il processo di build di un'applicazione *software_G*.

Attività: Un'Attività rappresenta un'unità di lavoro specifica che contribuisce al raggiungimento di un obiettivo nel ciclo di vita del *software_G*. Le Attività sono azioni o compiti distinti che devono essere eseguiti per portare avanti il *processo_G* di sviluppo.

B

Best practice: Il termine Best Practice si riferisce a un metodo o a un *processo_G* riconosciuto come una delle migliori e più efficaci modalità di esecuzione di un'*attività_G* o di un'operazione in un determinato contesto. Le Best Practice rappresentano un insieme di tecniche, metodi o procedure che sono state identificate come le più efficienti, efficaci o appropriate per raggiungere un obiettivo specifico o ottenere risultati di alta qualità.

Big Data: Il termine Big Data si riferisce a enormi insiemi di dati caratterizzati da volume massivo, velocità elevata di generazione e varietà di tipi e formati. Questi dati richiedono tecniche avanzate per l'elaborazione e l'analisi. L'obiettivo è estrarre informazioni significative per prendere decisioni informate e identificare modelli.

Bit: Bit è l'unità fondamentale di informazione in informatica, può assumere i valori 0 o 1.

Branch: Un Branch, anche chiamato ramo, costituisce un percorso separato di sviluppo. È una rappresentazione isolata utilizzata per modificare e registrare le modifiche di una specifica funzionalità, sviluppata in modo autonomo ma derivante dalla radice da cui è stato creato il Branch.

Broker: Un Broker agisce come un intermediario tra componenti *software_G*, *sistemi_G* o *servizi_G*, facilitando la loro interazione e comunicazione. Ci sono diverse tipologie di Broker, e il loro ruolo specifico può variare a seconda del contesto.

Budget at Completion (BAC): Budget preventivato ad inizio progetto.

Budget variance (BV): (Descrizione più approfondita rispetto a quella presente in *Norme di Progetto* nella sezione Metriche di qualità). La Budget Variance (BV) è una misura utilizzata nel project management per valutare la differenza tra la spesa effettiva e la spesa prevista o pianificata in un determinato periodo di tempo o per un'*attività_G* specifica all'interno di un progetto.

C

CA: Customer Acceptance (CA) è un acronimo che identifica una revisione facoltativa nel corso dello sviluppo di un progetto didattico. Durante questa fase, il prodotto *software_G*, considerato completo, viene presentato al *committente_G* in una sessione pubblica per ottenere la sua accettazione finale.

Calendario: Il Calendario rappresenta l'insieme del quantitativo di ore disponibili di lavoro utile di ogni membro del gruppo.

Change Management: Il Change Management è il *processo_G* di gestione e controllo dei cambiamenti apportati all'ambiente *IT_G*, inclusi *software_G*, infrastrutture, *processi_G* o procedure. Questo processo è progettato per gestire in modo controllato e documentato le modifiche proposte, garantendo che siano valutate, autorizzate, pianificate, implementate e monitorate in modo coerente e sicuro. L'obiettivo è minimizzare il rischio di interruzioni indesiderate o impatti negativi sul *sistema_G* a causa di modifiche non pianificate o non gestite correttamente.

Clickhouse: ClickHouse è un *database_G open-source_G* di analisi e archiviazione progettato per l'elaborazione rapida di grandi quantità di dati in modo scalabile ed efficiente. Questo *sistema_G* di gestione dei dati utilizza un modello di *architettura_G* di tipo column-oriented, in cui i dati vengono organizzati e memorizzati per colonne anziché per righe. Tale struttura ottimizza le prestazioni delle query analitiche e l'efficienza della compressione dei dati, consentendo operazioni di lettura ad alta velocità su grandi volumi di informazioni.

Committente: Un Committente è una parte esterna responsabile di assegnare l'esecuzione di un progetto a un esecutore, che stabilisce i termini contrattuali relativi all'esecuzione stessa.

Complessità ciclomatica per metodo (MCCM): (Descrizione più approfondita rispetto a quella presente in *Norme di Progetto* nella sezione Metriche di qualità). La Complessità Ciclomantica per Metodo (*MCCM_G*) è una metrica *software_G* utilizzata per misurare la complessità di un singolo metodo o funzione all'interno di un *programma_G*. È derivata dalla complessità ciclomatica di McCabe, che valuta la complessità di un intero *programma_G* attraverso la quantità di flussi di controllo. La Complessità Ciclomantica per Metodo concentra l'attenzione sulla complessità all'interno delle singole funzioni o metodi, valutando la quantità di cammini di esecuzione indipendenti all'interno di una specifica funzione. Questo aiuta a determinare la complessità delle logiche condizionali e dei percorsi di esecuzione all'interno del metodo stesso.

Configuration item (CI): Un Configuration Item (CI) è un elemento che viene identificato, controllato e gestito all'interno di un *sistema_G* o di un progetto. Rappresenta un'entità gestita e tracciabile all'interno di un ambiente di sviluppo o di produzione.

Configuration management: Configuration Management si riferisce al processo di gestione e controllo delle configurazioni *software_G* e hardware all'interno di un *sistema_G* o di un ambiente *IT_G*. Questo include il monitoraggio, la gestione e la registrazione dettagliata di tutte le componenti, le versioni e le modifiche apportate a *software_G*, hardware, documentazione e altri elementi del *sistema_G*. L'obiettivo è garantire la coerenza, la tracciabilità e la disponibilità delle configurazioni in modo che gli elementi possano essere facilmente riprodotti o ripristinati in un determinato stato, evitando confusione o errori dovuti a versioni obsolete o non autorizzate.

Container: Un Container, contestualizzato all'utilizzo di *Docker_G*, è un'unità di *software_G* che contiene un'applicazione e tutte le sue dipendenze, compresi i file di *sistema_G*, le *librerie_G* e le configurazioni necessarie per eseguire l'applicazione in un ambiente isolato e autonomo.

Cost performance index (CPI): (Descrizione più approfondita rispetto a quella presente in *Norme di Progetto* nella sezione Metriche di qualità). Il Cost Performance Index (*CPI_G*) è un'importante misura utilizzata nel project management per valutare l'efficienza dei costi in un progetto. È un indicatore di prestazione finanziaria che confronta il valore del lavoro effettuato (Earned Value, EV) con i costi effettivi (Actual Cost, AC) sostenuti fino a un certo punto nel tempo. In termini più semplici, il *CPI_G* mostra quanto valore del lavoro pianificato è stato effettivamente ottenuto per ogni unità di costo spesa.

D

Dashboard: Una Dashboard è un'interfaccia utente grafica che fornisce una panoramica visiva delle informazioni più importanti, dei dati o delle metriche pertinenti a un utente o a un processo specifico. Le Dashboard sono progettate per semplificare la visualizzazione dei dati complessi e offrire una rapida panoramica delle prestazioni, delle tendenze o delle metriche chiave. Le Dashboard possono essere personalizzate per soddisfare le esigenze specifiche di un utente o di un'organizzazione e spesso includono grafici, tabelle, grafici e altri elementi visivi che rappresentano i dati in modo chiaro e comprensibile.

Database: Un Database è un *sistema_G* organizzato per raccogliere, memorizzare e gestire dati in modo strutturato, al fine di consentire l'accesso, la gestione e l'aggiornamento efficiente di quei dati.

Database noSQL: Un Database NoSQL, o "non relazionale", è un tipo di *sistema_G* di gestione del *database_G* (DBMS) che si discosta dai tradizionali *database_G* relazionali. Questi *database_G* sono progettati per gestire volumi elevati.

Database SQL: Un Database SQL (Structured Query Language) è un tipo di *sistema_G* di gestione del *database_G* (DBMS) basato su un modello relazionale, in cui i dati sono organizzati in tabelle con righe e colonne. Il linguaggio *SQL_G* è utilizzato per definire, manipolare e interrogare i dati all'interno di questo tipo di *database_G*.

Data Gathering: Data gathering si riferisce alla raccolta di dati da diverse fonti. Può coinvolgere l'acquisizione, l'archiviazione e l'organizzazione di informazioni per un'analisi successiva.

Data Streaming Processing: Il Data Streaming Processing si riferisce al trattamento e all'analisi di dati in tempo reale mentre vengono trasmessi o generati. A differenza del modello di batch processing, in cui i dati vengono raccolti, salvati e successivamente processati, il *data streaming processing_G* opera su flussi continui di dati in tempo reale.

Data visualization: Il termine Data Visualization si riferisce al processo di rappresentare visivamente dati complessi e astratti in modo chiaro e comprensibile. L'obiettivo principale della Data Visualization è trasformare informazioni numeriche o qualitative in grafici, diagrammi, mappe e altre rappresentazioni visive che facilitino la comprensione, l'analisi e la comunicazione dei dati.

Deep dive: Il termine Deep dive è un'espressione inglese che viene spesso utilizzata per indicare un'indagine, un'analisi approfondita o un'immersione dettagliata in un argomento specifico.

Demo: Una Demo di prodotto o *servizio_G* è una presentazione pratica che mira a mostrare le funzionalità, le caratteristiche e i vantaggi di un determinato prodotto o *servizio_G*. Può coinvolgere l'uso diretto del prodotto o una spiegazione dettagliata delle sue capacità.

DevOps: DevOps, una combinazione di sviluppo (Dev) e operazioni (Ops), è l'unione di persone, *processi_G* e tecnologia per offrire continuamente valore ai clienti. DevOps permette a ruoli in precedenza isolati, tra cui sviluppo, operazioni *IT_G*, controllo della qualità e sicurezza, di coordinarsi e collaborare per fornire prodotti migliori e più affidabili.

Disaccoppiare: Il termine Disaccoppiare significa separare o isolare componenti o *processi_G* in modo che non siano fortemente dipendenti l'uno dall'altro.

Discord: *Discord_G* fornisce canali di chat testuale e vocali, consentendo agli utenti di comunicare tramite messaggi di testo o chiamate vocali, sia uno a uno che in gruppo.

Dispositivi IoT: I Dispositivi IoT sono dispositivi fisici che incorporano tecnologia, *sensori_G*, *software_G* e connettività di *rete_G* per consentire la raccolta, la condivisione e l'elaborazione di dati. Questi *dispositivi_G* sono progettati per interagire con l'ambiente circostante e/o con altri *dispositivi_G*, attraverso la connessione a Internet o attraverso reti locali.

Docker: *Docker_G* è una *piattaforma_G* che facilita la creazione, la distribuzione e l'esecuzione di applicazioni all'interno di contenitori *software_G*. I contenitori sono ambienti isolati e leggeri che includono l'applicazione e tutte le sue dipendenze (*librerie_G*, strumenti, codice, configurazioni), consentendo alle applicazioni di essere eseguite in modo coerente e affidabile su diversi ambienti, come lo sviluppo locale, i *test_G* e i server di produzione.

Dockerfile: Un *Dockerfile_G* è un file di testo che contiene una serie di istruzioni e comandi che definiscono i passaggi necessari per creare un'immagine *Docker_G*.

E

Earned Value (EV): (Descrizione più approfondita rispetto a quella presente in *Norme di Progetto* nella sezione Metriche di qualità). L'Earned Value (EV) è un concetto chiave nel project management che rappresenta il valore del lavoro effettivamente completato in un dato momento nel corso di un progetto. È utilizzato per valutare le prestazioni del progetto in base al valore del lavoro svolto rispetto alla pianificazione.

Element: Element è un client di messaggistica istantanea libero e *open-source*_G basato sul protocollo Matrix e distribuito con licenza Apache 2.0.

Estimated at completion (EAC): (Descrizione più approfondita rispetto a quella presente in *Norme di Progetto* nella sezione Metriche di qualità). Estimated at Completion (*EAC*_G) è un termine utilizzato nella gestione dei progetti e dell'analisi dei costi. Rappresenta la stima del costo totale che sarà richiesto per completare un progetto quando è a un certo punto di avanzamento. In sostanza, è una previsione dei costi totali previsti per portare a termine un progetto sulla base delle prestazioni passate e attuali del progetto stesso.

Estimated to complete (ETC): (Descrizione più approfondita rispetto a quella presente in *Norme di Progetto* nella sezione Metriche di qualità). L'Estimate to Complete (*ETC*_G) è una previsione dei costi aggiuntivi necessari per portare a termine un progetto o una specifica *attività*_G entro il suo budget assegnato.

F

Faker: *Faker_G* è una libreria di Python progettata per generare dati falsi in modo casuale. Gli sviluppatori possono utilizzare questa libreria per creare insiemi di dati che rispecchiano la struttura e la variabilità dei dati reali, senza dover utilizzare informazioni sensibili o reali durante lo sviluppo o i *test_G* del *software_G*.

Feature brach: Il *Feature branch_G* è un approccio nel controllo delle versioni del *software_G* che coinvolge la creazione di rami separati (*branch_G*) nel *sistema_G* di gestione delle versioni, come Git, per sviluppare nuove funzionalità o modifiche senza influenzare il ramo principale del codice.

Fornitore: Un Fornitore è un'entità o un'organizzazione che offre beni o *servizi_G* relativi al *software_G*.

Framework: Un Framework nel contesto dell'informatica e dello sviluppo del *software_G*, si riferisce a un insieme di strumenti, *librerie_G*, linee guida e convenzioni di codifica predefinite che forniscono una struttura comune per lo sviluppo di *software_G*. In altre parole, un Framework è un'infrastruttura *software_G* che facilita lo sviluppo di applicazioni fornendo un ambiente predefinito e organizzato in cui i programmatori possono lavorare.

Front-end: Il termine Front-end si riferisce alla parte di un'applicazione o di un *sistema_G* che interagisce direttamente con gli utenti finali. È la parte visibile e interattiva di un'applicazione, responsabile della presentazione dell'interfaccia utente e dell'interazione con gli input dell'utente.

G

Gmail: Servizio di posta elettronica fornito da Google.

Git: Git è un *sistema_G* di controllo delle versioni distribuito (DVCS - Distributed Version Control System) progettato per gestire il tracciamento delle modifiche nel codice sorgente durante lo sviluppo del *software_G*. È uno strumento ampiamente utilizzato dai team di sviluppo *software_G* per tenere traccia delle revisioni del codice, facilitare la collaborazione e gestire le modifiche apportate al progetto.

Github: GitHub è una *piattaforma_G* di hosting per il controllo delle versioni basato su *Git_G*, utilizzato principalmente per la gestione dei *repository_G* di codice sorgente. Fornisce strumenti per lo sviluppo collaborativo del *software_G*, facilitando la gestione, l'hosting e la collaborazione su progetti *software_G*.

Github Actions: GitHub Actions è un *servizio_G* di automazione fornito da *Github_G* che consente di automatizzare diversi flussi di lavoro all'interno di un *repository_G* *Github_G*. Consente agli sviluppatori di creare, personalizzare e condividere *workflow_G* personalizzati che vengono attivati da specifici eventi all'interno del *repository_G*.

Google Meet: Google Meet è un *servizio_G* di videoconferenza sviluppato da Google. Consente agli utenti di organizzare e partecipare a riunioni online, conferenze video e chiamate virtuali. È stato progettato per scopi aziendali, educativi e personali e offre diverse funzionalità per facilitare la collaborazione a distanza.

Grafana: *Grafana_G* è una *piattaforma_G* *open-source_G* per la visualizzazione e l'analisi di dati metrici e *log_G*. Essa fornisce strumenti potenti per la creazione di *dashboard_G* interattive, che consentono agli utenti di monitorare e esplorare dati provenienti da una varietà di fonti. *Grafana_G* è ampiamente utilizzato nel monitoraggio di sistemi, nell'analisi dei dati e nella creazione di *dashboard_G* per presentare informazioni in modo chiaro e comprensibile.

I

ID: ID_G è l'abbreviazione di "identificatore" o "identificativo". Un ID_G è un numero o un codice univoco assegnato a un oggetto, un record o un'entità per identificarlo in modo univoco in un $sistema_G$ o contesto specifico. Gli ID_G vengono spesso utilizzati per riferirsi a dati o risorse in modo univoco e per garantire che ciascun elemento all'interno di un $sistema_G$ possa essere distintamente identificato.

IEEE: L' $IEEE_G$ (Institute of Electrical and Electronics Engineers) è un'organizzazione internazionale che riunisce professionisti provenienti da una vasta gamma di discipline nell'ambito dell'ingegneria elettrica, dell'informatica, dell'elettronica e delle scienze informatiche. L' $IEEE_G$ svolge un ruolo cruciale nello sviluppo e nella promozione degli $standard_G$ tecnici in vari settori, pubblica una vasta gamma di contenuti tecnici, organizza eventi e gestisce numerose società.

Integrazione: In informatica, l'Integrazione si riferisce al processo di connessione e coordinamento di $sistemi_G$, applicazioni e dati diversi per farli lavorare insieme in modo efficace e sinergico. L'obiettivo principale dell' $integrazione_G$ è facilitare lo scambio di informazioni tra sistemi eterogenei per migliorare l'efficienza, l'automazione dei $processi_G$ e la coerenza dei dati.

Issue: Issue si riferisce a un problema, a una richiesta di funzionalità, a un bug o a un compito specifico che richiede attenzione all'interno di un $sistema_G$ di tracciamento dei problemi (ITS_G).

Issue Tracking System (ITS): Un ITS, acronimo di "Issue Tracking System", è un $software_G$ utilizzato per gestire e tenere traccia di problemi, bug, richieste di funzionalità e altre $attività_G$ correlate nello sviluppo del $software_G$ o in altri contesti lavorativi. Questi sistemi forniscono un'infrastruttura organizzativa per il $processo_G$ di gestione delle problematiche, consentendo agli sviluppatori, ai team di supporto o agli $stakeholder_G$ di registrare, monitorare, assegnare priorità e risolvere le problematiche che possono emergere durante lo sviluppo del $software_G$ o in altri progetti.

J

Jira: Jira è una *piattaform_G software_G* di gestione dei progetti sviluppata da Atlassian. È ampiamente utilizzata per la gestione delle *attività_G*, la tracciabilità dei progetti e la collaborazione tra team, in particolare nel contesto dello sviluppo *software_G* e della gestione dei progetti agili.

JSON: JSON (JavaScript Object Notation) è un formato di scrittura per lo scambio di dati. È basato su un sottoinsieme del linguaggio JavaScript, ma è un formato di testo indipendente dal linguaggio di programmazione, ampiamente utilizzato per la trasmissione di dati strutturati tra un server e un client web o tra diverse componenti *software_G*.

L

LaTeX: LaTeX è un linguaggio di markup e un *sistema_G* di composizione tipografica utilizzato principalmente per la produzione di documenti scientifici, accademici e tecnici di alta qualità.

Label: Una *label_G* in un ITS (Issue Tracking System) è un'etichetta o un'annotazione che viene associata a un'*attività_G*, un problema o una richiesta all'interno del *sistema_G* di tracciamento delle problematiche. Questa etichetta fornisce un'indicazione rapida e categorica sull'argomento, sulla tipologia o sulla priorità dell'elemento in questione.

Librerie: Il termine Libreria si riferisce a una raccolta di codice pre-sviluppato che fornisce funzionalità comuni e specifiche per facilitare lo sviluppo del *software_G*. Le Librerie sono composte da moduli o *routine di codice_G* che possono essere utilizzati da un *programma_G* per eseguire operazioni specifiche, senza dover scrivere tutto il codice da zero.

Log: Il termine Log si riferisce a un file o a un registro che registra eventi, *attività_G* o messaggi relativi al funzionamento di un *sistema_G*, di un'applicazione o di un componente *software_G*. I Log sono spesso utilizzati per monitorare il comportamento di un *sistema_G*, per diagnosticare problemi, e per tracciare le *attività_G* al fine di analizzarle in seguito.

M

Maschera: Il termine Maschera si riferisce generalmente a un'interfaccia grafica o a un elemento visuale che consente agli utenti di interagire con un'applicazione o un *sistema_G*. Le Maschere sono utilizzate per raccogliere o visualizzare informazioni, spesso in forma di moduli, finestre o schermate.

Milestone: Le Milestone sono punti specifici o traguardi significativi nel ciclo di sviluppo di un'applicazione o di un progetto. Questi sono eventi cruciali che segnalano il completamento di una fase importante o di un insieme di *attività_G* rilevanti verso il raggiungimento di un obiettivo.

Multi-stage builds: Le Multi-stage Builds in *Docker_G* sono un'approccio che consente di creare immagini *Docker_G* utilizzando più fasi o fasi di build separate all'interno di un singolo *Dockerfile_G*. Questa funzionalità è utile per ottimizzare le dimensioni delle immagini finali e per semplificare il *processo_G* di creazione delle immagini, eliminando la necessità di layer non necessari o file temporanei.

MVP: Acronimo di Minimum Viable Product. Rappresenta una versione preliminare del prodotto *software_G* da realizzare, il cui scopo è raccogliere quante più informazioni possibili riguardo i needs del cliente con il minimo impiego di risorse.

N

Normalizzazione: La Normalizzazione dei dati è un *processo*_G utilizzato nell'ambito dell'analisi dei dati e del machine learning per standardizzare o regolarizzare le caratteristiche di un dataset. L'obiettivo della *normalizzazione*_G è rendere i dati confrontabili, eliminando eventuali differenze nelle unità di misura, nelle scale o nelle distribuzioni dei dati originali.

O

OLAP: OLAP è l'acronimo di "Online Analytical Processing", una tecnologia informatica che consente l'analisi interattiva e in tempo reale di grandi quantità di dati da varie prospettive. Questa metodologia organizza i dati in modo multidimensionale, facilita l'esplorazione interattiva, e supporta l'aggregazione dei dati a diversi livelli.

Olistico: Olistico deriva da "olismo", un concetto che si riferisce alla considerazione dell'insieme come maggiore della somma delle sue parti. Può indicare un approccio completo che considera tutte le componenti coinvolte in un *sistema_G* o un'idea, anziché focalizzarsi solo su alcuni aspetti isolati.

Open-source: Il termine Open-source si riferisce a un tipo di licenza *software_G* che permette agli utenti di accedere, modificare e distribuire liberamente il codice sorgente di un'applicazione. La filosofia principale dell'approccio Open-source promuove la trasparenza, la collaborazione e la condivisione nella creazione e nell'uso del *software_G*.

Overleaf: Overleaf è una *piattaforma_G* online per la scrittura collaborativa di documenti in *LaTeX_G*. Questa *piattaforma_G* offre un ambiente di scrittura *LaTeX_G* basato sul web, consentendo a più utenti di lavorare contemporaneamente sugli stessi documenti, facilitando la collaborazione e la revisione.

P

Paradigma: Il termine Paradigma si riferisce a un modello o a un approccio di programmazione fondamentale che guida la progettazione e l'implementazione del *software_G*. Ogni Paradigma ha principi di base che definiscono come vengono organizzati i programmi, come i dati sono manipolati e come il controllo del flusso è gestito.

Pattern: I Pattern forniscono un metodo riconosciuto e testato per affrontare e risolvere determinate sfide di progettazione e sviluppo, consentendo agli sviluppatori di condividere conoscenze e approcci comuni.

PDF: PDF è l'acronimo di Portable Document Format. Si tratta di un formato di file sviluppato da Adobe che consente di rappresentare documenti in modo indipendente dal *software_G*, dall'hardware e dal *sistema_G* operativo.

Piattaforma: Il termine Piattaforma si riferisce a un ambiente hardware e/o *software_G* che fornisce le risorse e i servizi necessari per lo sviluppo, l'esecuzione e la gestione di applicazioni *software_G*.

PB: Product Baseline (*PB_G*) è un acronimo che identifica una tappa fondamentale nello sviluppo di un progetto didattico. Durante questa revisione, viene dimostrata la solidità dell'*architettura_G* definita nella fase precedente (*RTB_G*) e il progresso nell'evoluzione del prodotto in sviluppo.

Planned value (PV): (Descrizione più approfondita rispetto a quella presente in *Norme di Progetto* nella sezione Metriche di qualità). Il Planned Value (*PV_G*) è il valore monetario del lavoro pianificato da completare fino a una determinata data all'interno di un progetto. Il *PV_G* è calcolato in base al budget assegnato per svolgere specifiche *attività_G* o raggiungere determinati *milestone_G* entro un periodo prestabilito. Rappresenta il valore totale pianificato che dovrebbe essere raggiunto in termini di costi o tempo alla data specificata.

POC: L'acronimo *POC_G* sta per "Proof of Concept". Un *POC_G* è un'implementazione pratica di un'idea o di un concetto, realizzata con l'obiettivo di dimostrare la fattibilità tecnica o la validità di un approccio particolare prima di impegnarsi in uno sviluppo a pieno regime.

Processi: Raggruppano e codificano le *attività_G* da svolgere per effettuare corrette transizioni di stato nel ciclo di vita di un prodotto *software_G*.

Programma: Un Programma è una sequenza di istruzioni scritte in un linguaggio di programmazione che definisce una serie di operazioni da eseguire da parte di un computer.

Proponente: In ambito di sviluppo *software_G*, il termine Proponente si riferisce generalmente a chi propone o suggerisce un progetto, un'idea o una soluzione.

Python: Python è un linguaggio di programmazione ad alto livello, versatile e orientato agli oggetti, utilizzato per lo sviluppo di *software_G*, scripting, automazione e altre applicazioni.

Pull request: Una Pull Request (PR) è una richiesta effettuata da un collaboratore di un *repository*_G *Git*_G o di una *piattaforma*_G di hosting come *Github*_G per incorporare le modifiche effettuate in un *branch*_G del *repository*_G principale.

R

Raw: Raw si riferisce ai dati grezzi o non elaborati provenienti direttamente dai *sensori_G*, senza ulteriori manipolazioni o trasformazioni.

Report: Il termine Report si riferisce a un documento scritto o a una presentazione che fornisce informazioni dettagliate su un determinato argomento, situazione o *attività_G*.

Repository: Un *repository_G* è uno spazio di archiviazione o deposito digitale in cui vengono conservati e gestiti file, dati, codice sorgente o altre informazioni pertinenti a un progetto, a un'applicazione o a un insieme di dati.

Requisiti contrattuali: Rappresentano l'accordo tra la parte *proponente_G* e il *fornitore_G* su quale deve essere l'oggetto e la forma del prodotto *software_G* da realizzare.

Requisiti soluzione: Requisiti che rispecchiano le funzionalità del prodotto, atti a soddisfare i *requisiti utente_G*.

Requisiti utente: Requisiti che rispecchiano il lato utente, vengono espressi sotto forma di bisogni.

Rete: Il termine Rete si riferisce a un insieme di dispositivi e *sistemi_G* interconnessi che comunicano tra loro per condividere risorse, dati e informazioni.

Roadmap: Una Roadmap è un piano strategico che visualizza le tappe chiave, le *attività_G* e le direzioni principali necessarie per raggiungere un obiettivo specifico nel tempo.

Rollback: Il Rollback nello sviluppo *software_G* si riferisce al *processo_G* di ripristino di una versione precedente, funzionante e stabile di un'applicazione o di un *sistema_G*.

Routine di codice: Il termine si riferisce a una sequenza di istruzioni o operazioni di programmazione che eseguono una specifica funzione o compito all'interno di un *programma_G software_G*.

RTB: Requirements and Technology Baseline (*RTB_G*) è un acronimo che identifica la revisione essenziale nel percorso di sviluppo di un progetto didattico.

S

Schedule Variance: (Descrizione più approfondita rispetto a quella presente in *Norme di Progetto* nella sezione Metriche di qualità). Lo Schedule Variance (SV) è un indicatore utilizzato nel project management per misurare il ritardo o l'avanzamento rispetto alla pianificazione temporale prevista per il lavoro o le *attività_G* in un progetto.

Scouting tecnologico: Lo *scouting tecnologico_G* è un *processo_G* strategico volto a identificare, valutare e monitorare le nuove tecnologie emergenti e le innovazioni nel settore tecnologico.

Script: Uno Script è un file contenente una serie di istruzioni o comandi scritti in un linguaggio di scripting, come *Python_G*, *Bash*, o *JavaScript*.

Sensore: In informatica e nell'ambito dell'Internet delle cose (IoT), un Sensore è un dispositivo in grado di rilevare o misurare un input fisico o ambientale e convertirlo in un segnale o dati.

Servizio: Un Servizio si riferisce a una funzionalità o a un'opzione offerta da un *sistema_G*.

Single Responsibility Principle (SRP): Il Single Responsibility Principle è uno dei cinque principi del *SOLID_G*, un insieme di linee guida e *best practice_G* per la progettazione del *software_G*.

Sistema: Il termine Sistema si riferisce a un insieme di componenti interconnessi che lavorano insieme per raggiungere uno scopo comune.

Smart city: Una Smart City (città intelligente) è una città che utilizza la tecnologia e l'innovazione digitale per migliorare la qualità della vita dei suoi cittadini e ottimizzare l'efficienza delle operazioni urbane.

Software: Un Software è una collezione di istruzioni, programmi, dati e documentazione che permette al computer di eseguire determinate funzioni o compiti.

SOLID: *SOLID_G* è un acronimo che rappresenta cinque principi di progettazione del *software_G* che mirano a creare codice più manutenibile, flessibile e facile da comprendere.

SQL: SQL (Structured Query Language) è un linguaggio di programmazione specializzato utilizzato per gestire e manipolare i dati all'interno di un *sistema_G* di gestione del *database_G* relazionale (RDBMS).

Stack tecnologico: Lo Stack Tecnologico si riferisce a un insieme di tecnologie *software_G*, strumenti e *framework_G* utilizzati per sviluppare e gestire un'applicazione o un *sistema_G software_G*.

Stakeholder: Lo Stakeholder è un termine utilizzato per identificare qualsiasi individuo, gruppo o entità che ha un interesse diretto o può essere influenzato dalle azioni.

Standard: Standard è un termine utilizzato per indicare un modello, una regola o una specifica che è comunemente accettata come riferimento.

Stand-up meeting: Gli Stand-up Meeting sono incontri brevi e regolari utilizzati comunemente in contesti di sviluppo *software_G* e in ambito lavorativo per favorire la comunicazione.

Statement coverage (MSC): (Descrizione più approfondita rispetto a quella presente in *Norme di Progetto* nella sezione Metriche di qualità). Lo Statement Coverage (*MSC_G*) è una metrica utilizzata nel testing del *software_G* per valutare quanto del codice sorgente viene eseguito durante l'esecuzione dei *test_G*. Questa metrica indica la percentuale di istruzioni nel codice sorgente che sono state eseguite almeno una volta durante l'esecuzione dei *test_G* rispetto al totale delle istruzioni presenti nel *programma_G*.

Stato avanzamento lavori (SAL): Lo Stato di Avanzamento Lavori (*SAL_G*) è una misurazione o un resoconto che indica quanto un progetto o un lavoro è progredito rispetto alle *attività_G* pianificate o alle fasi di sviluppo.

Storage: Il termine Storage si riferisce alla memorizzazione permanente e alla gestione dei dati su dispositivi di archiviazione.

Stream: Il termine Stream in informatica si riferisce a un flusso di dati in movimento continuo da una sorgente a una destinazione.

Stream processing: Lo Stream Processing è un *paradigma_G* di elaborazione dei dati in cui le informazioni vengono analizzate e elaborate in tempo reale man mano che vengono generate o ricevute.

T

Technology baseline: Il termine Technology Baseline (Baseline Tecnologica) si riferisce a una linea di base o ad una configurazione di tecnologie, strumenti e risorse necessarie per lo sviluppo e la realizzazione di un progetto o di un *sistema_G* specifico. Questa linea di base comprende una serie di elementi essenziali, come hardware, *software_G*, *framework_G*, strumenti di sviluppo, metodologie, *standard_G* e altre risorse tecnologiche fondamentali necessarie per supportare e facilitare il processo di sviluppo e la realizzazione di un *sistema_G*, prodotto o *servizio_G*. La Technology Baseline viene stabilita durante le fasi iniziali di pianificazione di un progetto, definendo gli strumenti e le tecnologie che verranno utilizzati come punto di partenza per lo sviluppo.

Telegram: Applicazione di messaggistica istantanea.

Test: Il termine Test si riferisce a un *processo_G* o a un'*attività_G* che mira a verificare il funzionamento di un *software_G* o di un *sistema_G* informatico al fine di identificare eventuali difetti o problemi.

Test di accettazione: La definizione accurata di *test_G* di accettazione viene data nel paragrafo §3.2.3.8 del documento *Norme di Progetto*.

Test di integrazione: La definizione accurata di *test_G* di *integrazione_G* viene data nel paragrafo §3.2.3.5 del documento *Norme di Progetto*.

Test di regressione: La definizione accurata di *test_G* di regressione viene data nel paragrafo §3.2.3.7 del documento *Norme di Progetto*.

Test di sistema: La definizione accurata di *test_G* di *sistema_G* viene data nel paragrafo §3.2.3.6 del documento *Norme di Progetto*.

Test di unità: La definizione accurata di *test_G* di unità viene data nel paragrafo §3.2.3.4 del documento *Norme di Progetto*.

Tool: La parola Tool (strumento) si riferisce a un *software_G* o a un'applicazione che fornisce funzionalità specifiche o servizi per facilitare un compito o un'*attività_G*. Un Tool può essere progettato per svolgere una vasta gamma di compiti.

U

Use case: Nel contesto UML rappresenta una lista di azioni compiute da un attore sul *sistema_G software_G* al fine di raggiungere un determinato fine.

UML: UML (Unified Modeling Language) è un linguaggio di modellazione visuale utilizzato principalmente nell'ingegneria del *software_G* per descrivere, progettare e documentare *sistemi_G software_G* complessi.

Unità architettuale: Un'Unità Architettuale si riferisce a un modulo, a una componente o a un'entità di base che costituisce una parte significativa di un *sistema_G software_G* o hardware.

V

VSCode: Visual Studio Code (VSCode) è un editor di codice sorgente leggero, estensibile e *open-source*_G sviluppato da Microsoft. È ampiamente utilizzato dagli sviluppatori *software*_G per scrivere, modificare e debuggare codice in diversi linguaggi di programmazione.

W

Way of working: Insieme delle metodologie, dei *pattern_G* e degli strumenti utilizzati in maniera sistematica e disciplinata durante lo sviluppo di un progetto *software_G*.

Widget: Un Widget è un componente grafico o un oggetto interattivo che può essere incorporato in un'interfaccia utente di un'applicazione *software_G* o di una pagina web. I Widget sono progettati per fornire funzionalità specifiche e interazioni dirette con gli utenti, migliorando l'esperienza utente complessiva.

Workflow: Un Workflow rappresenta il flusso o la sequenza di *attività_G*, *processi_G* o azioni che vengono eseguite in un particolare ordine per completare un compito o raggiungere un obiettivo specifico. È una rappresentazione visuale o concettuale di come le *attività_G* sono strutturate e connesse tra loro all'interno di un processo più ampio.