



# TECH MINDS

## Glossario

[techminds.unipd@gmail.com](mailto:techminds.unipd@gmail.com)

### Sommario

Documento dei termini tecnici relativi al progetto.

## Changelog

Versione	Data	Descrizione	Autore	Verificatore
1.0.0	11/02/2025	Piccoli fix e aggiunta parole al glossario	Bressan Alessandro	Lazzarin Tommaso
0.5.0	08/02/2025	Aggiunta definizione dei test	Corradin Samuele	Lazzarin Tommaso
0.4.0	22/01/2025	Aggiunta definizione di Docker	Lazzarin Tommaso	Vallotto Caterina
0.3.0	19/01/2025	Aggiunti nuovi termini	Lazzarin Tommaso	Tutino Giuseppe
0.2.0	12/01/2024	Inserimento nuovi termini	Vallotto Caterina	Squarzoni Matteo
0.1.1	20/12/2024	Fix conformità indice di Gulpease	Vallotto Caterina	Tutino Giuseppe
0.1.0	11/11/2024	Prima versione	Salviato Leonardo	Vallotto Caterina

## Indice

Introduzione .....	5
A .....	5
AC .....	5
Accoppiamento .....	5
Agente .....	5
Analisi dei requisiti .....	5
Analisi dei rischi .....	5
API .....	5
Arco .....	5
Attore .....	5
AWS .....	5
B .....	6
BAC .....	6
Backlog .....	6
Baseline .....	6
Bedrock .....	6
Blocco .....	6
Branch .....	6
Build Automation .....	6
C .....	6
Capitolato .....	6
Caso d'uso .....	6
Changelog .....	6
Commit .....	6
CPI .....	7
CV .....	7
D .....	7
Daily stand-up meeting .....	7
Diagramma di Gantt .....	7
Diagramma UML .....	7
Discord .....	7
Docker .....	8

E	8
EAC	8
Economicità	8
Efficacia	8
Efficienza	8
ESLint	8
ETC	8
EV	8
F	9
G	9
Git	9
GitHub	9
GitHub Actions	9
GitHub Flow	9
GitHub Pages	9
Glossario	9
H	9
I	9
Indice Gulpease	9
Integrazione continua	9
Intelligenza Artificiale	9
Issue	9
ITS	10
J	10
K	10
L	10
Label	10
LaTeX	10
LLM	10
M	10
Major version	10
Milestone	10
Minor version	10
MongoDB	10
MVP	10
N	11
Node.js	11
Norme di progetto	11
O	11
P	11
Pastebin	11
Patch	11
PB	11
PDCA	11
Piano di progetto	11
Piano di qualifica	11
PoC	11
PodMan	11
Processo	12

Processo di ciclo di vita .....	12
Progetto .....	12
Pull request .....	12
Push .....	12
PV .....	12
Python .....	12
Q .....	12
Qualità .....	12
R .....	12
React .....	12
Refactoring .....	12
Registro delle modifiche .....	13
Repository .....	13
Requisito .....	13
Responsabile .....	13
Retrospettiva .....	13
RSI .....	13
RTB .....	13
S .....	13
Scrum .....	13
Slack .....	13
SPI .....	13
Sprint .....	14
Sprint planning .....	14
Stakeholder .....	14
SV .....	14
Swagger .....	14
T .....	14
Test di accettazione .....	14
Test di integrazione .....	14
Test di regressione .....	15
Test di sistema .....	15
Test di unità .....	15
Tracciamento .....	15
TypeScript .....	15
Typst .....	15
U .....	15
UML .....	15
User Story .....	15
V .....	16
Validazione .....	16
Verbale .....	16
Verifica .....	16
W .....	16
Way of working .....	16
Workflow .....	16
X .....	16
Y .....	16
Z .....	16

## Introduzione

Questo documento definisce tutti i termini specifici usati nell'ambito del progetto di Ingegneria del Software, con lo scopo di fornire una terminologia comune a tutti i membri del gruppo.

## A

### AC

Acronimo di *Actual Cost*. È una metrica che misura i costi effettivamente sostenuti per svolgere le attività completate fino a un dato momento.

### Accoppiamento

Nell'ambito della programmazione ad oggetti, è una misura del grado di dipendenza fra due classi. Più le classi sono accoppiate più è probabile che la modifica di una implica la modifica delle classi che ne dipendono.

### Agente

Nel campo dell'intelligenza artificiale, l'agente (detto anche Agente Intelligente) è un'entità in grado di percepire l'ambiente esterno ed effettuare delle azioni in modo autonomo, migliorando le performance nel tempo attraverso la conoscenza passata.

### Analisi dei requisiti

Attività preliminare allo sviluppo (o alla modifica) di un sistema software, il cui scopo è quello di definire le funzionalità che il nuovo prodotto (o il prodotto modificato) deve offrire. Tali funzionalità sono rappresentate dai requisiti che devono essere soddisfatti dal software sviluppato.

### Analisi dei rischi

L'analisi dei rischi è un processo che identifica, valuta e pianifica i rischi che potrebbero influenzare il progetto. Permette di prendere misure per mitigarli e gestirli durante lo sviluppo del software.

### API

Acronimo di *Application Programming Interface*, è un insieme di procedure atte a consentire la comunicazione tra diversi software o tra diversi componenti di software.

### Arco

L'arco è un elemento specifico richiesto nel capitolato. Rappresenta il collegamento tra due blocchi. Sopra di esso viene inserita una descrizione in linguaggio naturale che indica l'automazione che l'utente desidera sia compiuta.

### Attore

Nel contesto dell'analisi dei requisiti rappresenta un'entità che interagisce con il sistema svolgendo delle attività. Può essere un utente oppure un sistema esterno.

### AWS

Acronimo di *Amazon Web Services*, è una sussidiaria di Amazon che fornisce servizi di cloud computing.

## B

### **BAC**

Acronimo di *Budget at Completion*. È il budget totale previsto per il completamento del progetto.

### **Backlog**

Insieme di compiti/attività da completare per un certo obiettivo. Nel framework Scrum esistono due tipologie di backlog:

- il *product backlog* ovvero l'insieme di tutti i compiti/attività del progetto;
- lo *spring backlog*, che indica i compiti/attività da completare durante uno sprint.

### **Baseline**

Versione di un prodotto corrispondente ad obiettivi di raggiungimento, spesso fissati da una milestone. Una baseline viene approvata ed è modificabile solamente tramite un processo definito, formale e tracciato.

### **Bedrock**

Bedrock è un servizio di AWS che rende disponibili modelli di intelligenza artificiale tramite un sistema unificato.

### **Blocco**

Un blocco è un elemento specifico richiesto nel capitolato. Rappresenta un elemento che può essere trascinato e posizionato all'interno di un'area di lavoro (interfaccia drag and drop). Ogni blocco è collegato a un servizio esterno.

### **Branch**

Un branch è una ramificazione del codice sorgente. Permette di lavorare su una copia separata del progetto senza influenzare il ramo principale. I rami sono utilizzati per sviluppare nuove funzionalità, correggere bug, o sperimentare senza compromettere il resto del codice.

### **Build Automation**

Processo che automatizza la build dei sorgenti.

## C

### **Capitolato**

Documento del proponente in cui viene esposto un problema che si vuole risolvere attraverso un applicativo software, specificando norme, tecnologie e vincoli.

### **Caso d'uso**

Descrizione dettagliata di come un attore interagisce con il sistema nel compiere un'attività specifica. È utilizzato nel contesto dello sviluppo software per individuare i requisiti funzionali del prodotto.

### **Changelog**

Si veda [Registro delle modifiche](#).

### **Commit**

Operazione che registra le modifiche apportate al codice in un repository. Ogni commit rappresenta uno snapshot del progetto in un determinato momento.

## CPI

Acronimo di *Cost Performance Index*. È un indice che offre una misura relativa dell'efficienza dei costi di un progetto fino a un dato momento. Indica quanto valore è stato ottenuto per ogni unità di costo spesa.

Si calcola come:

$$\text{CPI} = \frac{\text{EV}}{\text{AC}}$$

Si ha che:

- Se  $\text{CPI} > 1$  il progetto ha un'efficienza finanziaria positiva e sta guadagnando con il proprio lavoro più di quanto sta spendendo. Maggiore è il CPI, più efficiente è il progetto;
- Se  $\text{CPI} = 1$  il progetto sta spendendo esattamente quanto preventivato per ottenere il valore generato dal lavoro;
- Se  $\text{CPI} < 1$  il progetto ha un'efficienza finanziaria negativa e sta spendendo più di quanto sta guadagnando con il proprio lavoro. Minore è il CPI, meno efficiente è il progetto.

## CV

Acronimo di *Cost Variance*. Metrica che misura la differenza tra il valore e il costo del lavoro completato. Indica se il progetto è sotto o sopra il budget pianificato per il lavoro completato.

Si calcola come:

$$\text{CV} = \text{EV} - \text{AC}$$

Si ha quindi che:

- Se  $\text{CV} > 0$  il progetto sta spendendo meno di quanto ha guadagnato con il proprio lavoro ed è quindi sotto il budget previsto;
- Se  $\text{CV} = 0$  il progetto sta spendendo esattamente quanto ha guadagnato con il proprio lavoro ed è quindi perfettamente in linea con il budget previsto;
- Se  $\text{CV} < 0$  il progetto sta spendendo più di quanto ha guadagnato con il proprio lavoro ed è quindi sopra il budget previsto.

## D

### Daily stand-up meeting

Riunione giornaliera in cui i membri del team si tengono aggiornati riguardo il lavoro svolto e da svolgere, esponendo le difficoltà incontrate e proponendo possibili soluzioni. La continua comunicazione tra i membri del team permette di mantenere alta la produttività e di risolvere velocemente eventuali problemi.

### Diagramma di Gantt

Strumento a supporto della gestione dei progetti, costituito da barre orizzontali di lunghezza variabile che rappresentano le sequenze, la durata e l'arco temporale di ogni singola attività del progetto.

### Diagramma UML

Metodo per visualizzare sistemi e software utilizzando il linguaggio di modellazione unificato (UML).

### Discord

Piattaforma VoIP, messaggistica istantanea e distribuzione digitale.

## Docker

Piattaforma che permette di creare ambienti isolati di esecuzione chiamati container con un uso meno estensivo di risorse rispetto ad una macchina virtuale.

# E

## EAC

Acronimo di *Estimate at Completion*. È una metrica che stima il costo totale del progetto sulla base delle performance attuali.

Esistono diverse formule per il calcolo dell'EAC. La formula che più si addice alla natura di questo progetto è quella che prevede un miglioramento delle performance future rispetto a quelle attuali, ovvero:

$$EAC = \frac{BAC - EV}{CPI \cdot SPI} + AC.$$

## Economicità

Combinazione di efficienza ed efficacia.

## Efficacia

Misura della capacità di raggiungere l'obiettivo prefissato.

## Efficienza

Misura dell'abilità di raggiungere l'obiettivo impiegando le risorse minime indispensabili.

## ESLint

È uno strumento di analisi statica del codice JavaScript. Si occupa sia della qualità del codice che dello stile di codifica. È possibile sovrascrivere le regole di default tramite un file di configurazione contenuto all'interno del repository.

## ETC

Acronimo di *Estimate to Complete*. È una metrica che stima il costo rimanente necessario per completare il progetto.

La formula per calcolare l'ETC, basandosi sulle performance attuali, è:  $ETC = \frac{BAC - EV}{CPI}$ .

Questa formula assume che la performance di costo attuale non cambierà nel resto del progetto.

## EV

Acronimo di *Earned Value*. Misura il valore del lavoro effettivamente completato in un dato momento. È il valore che il progetto ha effettivamente generato fino a quel momento.

Si calcola come:  $EV = (\text{Percentuale di lavoro completato}) \cdot (BAC)$  ed è espresso in valore monetario.

Va periodicamente confrontato con:

- **AC** per valutare se il progetto è in linea con il budget o se si sta spendendo più di quello che si sta guadagnando con il proprio lavoro. Ciò avviene tramite le metriche **CV** e **CPI**;
- **PV** per valutare se il progetto è in linea con la pianificazione dal punto di vista dell'andamento temporale dei costi. Ciò avviene tramite le metriche **SV** e **SPI**.



## F

## G

### **Git**

Software per il controllo di versione distribuito.

### **GitHub**

Servizio di hosting per progetti software, il nome deriva dal fatto che è una implementazione dello strumento di controllo di versione distribuito Git.

### **GitHub Actions**

Piattaforma di continuous integration e continuous delivery che permette di automatizzare la build, i test e il rilascio di un prodotto.

### **GitHub Flow**

Pratica usata nella gestione dei branch in un repository, il GitHub Flow utilizza un branch principale (solitamente chiamato main o master) da cui vengono creati dei branch di feature. All'interno dei branch di feature vengono effettuate le modifiche che poi verranno integrate nel branch main attraverso una Pull Request.

### **GitHub Pages**

Servizio di GitHub che fornisce hosting per siti web statici.

### **Glossario**

Elenco organizzato di termini tecnici, acronimi e definizioni utilizzati nel contesto del progetto.

## H

## I

### **Indice Gulpease**

Indice di leggibilità di un testo tarato sulla lingua italiana. Rispetto agli altri ha il vantaggio di utilizzare la lunghezza delle parole in lettere anziché in sillabe, semplificandone il calcolo automatico. La formula per il calcolo è  $89 + \frac{300(n_{\text{frasi}}) - 10(n_{\text{lettere}})}{n_{\text{parole}}}$ .

### **Integrazione continua**

Spesso indicata con CI (dall'inglese *Continuous Integration*), è una pratica di sviluppo in cui è presente un meccanismo di Build Automation e dove il team integra frequentemente le modifiche.

### **Intelligenza Artificiale**

Disciplina che studia come realizzare sistemi informatici in grado di simulare il ragionamento umano.

### **Issue**

Elemento che richiede attenzione o risoluzione all'interno di un progetto. Rappresenta un ticket dell'issue tracking system di GitHub.

## **ITS**

Acronimo di *Issues Tracking System*, è uno strumento che gestisce e mantiene una lista di Issues, spesso usato per progetti in team.

## **J**

## **K**

## **L**

### **Label**

In GitHub, una label è un'etichetta che può essere applicata a un issue o a una pull request per fornire informazioni aggiuntive, come lo stato, la priorità, o il tipo di problema. Le label sono utili per organizzare e categorizzare facilmente gli elementi nel repository.

### **LaTeX**

LaTeX è un linguaggio di markup utilizzato per la scrittura di documenti scientifici e tecnici.

### **LLM**

Acronimo di *Large Language Model*. Modello di deep learning progettato per eseguire task di Natural Language Processing, come la generazione di un testo. Un LLM apprende molti parametri in modo supervisionato, individuando relazioni statistiche tra una grande quantità di testi.

## **M**

### **Major version**

Nel contesto di gestione delle versioni, indica un cambiamento radicale o aggiornamento significativo (a volte non retrocompatibile) di un software o documento.

### **Milestone**

Data di calendario che determina un punto di avanzamento atteso del progetto. Le milestones sono uno strumento di pianificazione.

### **Minor version**

Nel contesto di gestione delle versioni, indica un aggiornamento che aggiunge contenuto/funzionalità senza cambiare il significato o il comportamento di un software o documento.

### **MongoDB**

MongoDB è un database NoSQL orientato ai documenti. Utilizza un formato di dati JSON-like per archiviare le informazioni e permette una gestione flessibile dei dati.

### **MVP**

Acronimo di *Minimum Viable Product*. Prodotto software ridotto nelle funzionalità che soddisfa i requisiti essenziali. È utilizzato a scopo esplorativo per ricevere feedback e valutare la bontà della visione di prodotto, in modo da prendere decisioni consapevoli sul completamento del software.

## N

### **Node.js**

Node.js è un ambiente di esecuzione JavaScript lato server. Permette di eseguire codice JavaScript al di fuori di un browser, ed è utilizzato per costruire applicazioni server-side come API, servizi web e applicazioni real-time.

### **Norme di progetto**

Documento gestionale interno redatto in forma esecutiva, ossia composto da regole e procedure, che descrive il way of working. È volto a standardizzare i processi in modo da assicurare la coerenza del lavoro svolto dai diversi membri e la qualità dei prodotti.

## O

## P

### **Pastebin**

Servizio web che permette di condividere frammenti di testo e codice sorgente.

### **Patch**

Piccolo aggiornamento, spesso utilizzato per indicare la risoluzione di un bug o piccole modifiche.

### **PB**

Acronimo di *Product Baseline*. Baseline il cui oggetto centrale è il MVP (Minimum Viable Product).

### **PDCA**

Acronimo di *Plan-Do-Check-Act*. È un metodo di gestione iterativo per il controllo e il miglioramento continuo dei processi e dei prodotti.

### **Piano di progetto**

Il piano di progetto è un documento che descrive gli obiettivi, le risorse, i tempi e le attività previste per un progetto. Include anche la gestione dei rischi, la pianificazione delle fasi e l'assegnazione delle responsabilità.

### **Piano di qualifica**

Documento che specifica gli obiettivi, indicati in modo quantitativo, di qualità dei prodotti e dei processi. Nel documento è presente anche il cruscotto di qualità attraverso il quale si tiene traccia dell'andamento del progetto e si valutano le performance.

### **PoC**

Acronimo di *Proof of Concept*. È una baseline tecnologica che usa le tecnologie proposte per lo sviluppo dei requisiti. Il suo scopo è acquisire conoscenza, valutare l'adeguatezza e la compatibilità di tali tecnologie. Deve richiedere un basso costo di sviluppo.

### **PodMan**

Podman è un tool per la gestione di container simile a Docker. Permette di creare ambienti isolati di esecuzione chiamati container con un uso meno estensivo di risorse rispetto ad una macchina virtuale.

## Processo

Insieme di attività correlate e coese che, consumando risorse, trasformano ingressi (bisogni) in uscite (prodotti). Ogni processo deve svolgersi secondo le indicazioni fornite dalle Norme di Progetto.

## Processo di ciclo di vita

Processi che definiscono quali attività svolgere per attuare, in modo efficiente ed efficace, corrette transizioni di stato nel ciclo di vita di un prodotto software.

## Progetto

Insieme di attività che devono raggiungere determinati obiettivi a partire da determinate specifiche, con una data di inizio e una data di fine fissate, con risorse limitate che si consumano nello svolgersi.

## Pull request

Nel contesto di un repository di versionamento distribuito, è un modo per notificare ai membri del team che è stata completata una funzionalità in un branch di feature e che si può effettuare il merge nel branch main, con previa verifica.

## Push

Push è l'operazione che consente di inviare le modifiche locali (commit) a un repository remoto su GitHub o un altro server Git. Permette di aggiornare il codice condiviso con gli altri membri del team.

## PV

Acronimo di *Planned Value*. È una metrica che misura il valore del lavoro che, secondo la pianificazione, dovrebbe essere completato in un determinato momento.

Si calcola come:  $PV = (\text{Percentuale di completamento pianificata}) \cdot (\text{BAC})$ .

## Python

Linguaggio ad alto livello orientato agli oggetti, apprezzato per la sua semplicità e flessibilità.

# Q

## Qualità

Insieme delle caratteristiche di un'entità, che ne determinano la capacità di soddisfare esigenze sia espresse che implicite.

Può avere diversi punti di vista:

- Intrinseca: conformità ai requisiti, idoneità all'uso;
- Relativa: soddisfazione del cliente;
- Quantitativa: misurazione oggettiva, per confronto.

# R

## React

React è una libreria JavaScript per la costruzione di interfacce utente. È utilizzata per creare applicazioni web in modo dichiarativo e reattivo.

## Refactoring

Attività di ristrutturazione del codice sorgente, senza modificarne il comportamento esterno, al fine di renderlo più leggibile, manutenibile e riusabile.

## Registro delle modifiche

Struttura che tiene traccia delle modifiche effettuate ad un documento, con indicazione della versione, della data, della descrizione della modifica, dell'autore e del verificatore.

## Repository

Luogo in cui vengono salvati i prodotti creati. Costituisce una fonte di condivisione del lavoro svolto da ciascun membro. È caratterizzato da persistenza e consistenza, garantite dalle procedure indicate nelle Norme di Progetto.

## Requisito

La capacità necessaria a un utente per raggiungere un obiettivo (lato bisogno/requisiti utente) o la capacità necessaria a un sistema per rispondere a una aspettativa (lato soluzione/requisiti software).

## Responsabile

Persona che ha il compito di coordinare le attività di gruppo, l'elaborazione di piani e scadenze, e approva il rilascio di prodotti parziali o finali (software/documentazione).

## Retrospettiva

Attività che si svolge al termine di uno sprint, in cui il team riflette sul lavoro svolto, individua i punti di forza e debolezza e propone azioni di miglioramento per lo sprint successivo.

## RSI

Acronimo di *Requirements Stability Index*. È un indice che misura la stabilità dei requisiti durante il ciclo di vita del progetto. Indica quindi quanto i requisiti cambiano nel tempo. Un valore alto indica che i requisiti cambiano frequentemente e costituisce un segnale d'allarme.

Si calcola come  $RSI = \frac{\text{Numero di requisiti modificati}}{\text{Numero totale di requisiti}}$ .

## RTB

Acronimo di *Requirements and Technology Baseline*. Baseline il cui oggetto principale sono i requisiti e l'adeguatezza delle tecnologie, valutata tramite il PoC. I requisiti sono concordati con il proponente e raccolti nel documento Analisi dei Requisiti. Una tecnologia è considerata adeguata se permette di raggiungere l'obiettivo per cui è stata scelta e se è integrabile con le altre.

# S

## Scrum

Framework di sviluppo agile che suddivide il lavoro in cicli iterativi chiamati sprint e permette l'adattamento rapido ai cambiamenti attraverso feedback regolari.

## Slack

Piattaforma di messaggistica istantanea utilizzata dal team per la maggior parte delle comunicazioni con l'azienda proponente.

## SPI

Acronimo di *Schedule Performance Index*. È un indice che offre una misura relativa dell'efficienza del progetto, mettendo in relazione il lavoro completato con quello pianificato.

Si calcola come:

$$SPI = \frac{EV}{PV}$$

Si ha che:

- Se  $SPI > 1$  il progetto sta avanzando più velocemente rispetto a quanto pianificato;
- Se  $SPI = 1$  il progetto sta avanzando esattamente come pianificato;
- Se  $SPI < 1$  il progetto sta avanzando più lentamente rispetto a quanto pianificato.

## **Sprint**

Periodo di tempo prefissato, di durata massima di un mese, in cui il team sviluppa un piccolo incremento di prodotto finale.

## **Sprint planning**

È una pratica agile in cui pianifichiamo le attività da completare durante uno sprint. In questa fase, vengono scelti i task da affrontare in base alla priorità e alla disponibilità di ogni membro del team.

## **Stakeholder**

Persona o gruppo di persone che hanno interesse nel progetto, e lo influenzano o vengono influenzati da esso, come il committente o il fornitore.

## **SV**

Acronimo di *Schedule Variance*. Metrica che misura quanto il valore del lavoro completato si discosta da quanto pianificato. Indica se il progetto è in anticipo, in ritardo o in linea rispetto alla pianificazione.

Si calcola come:

$$SV = \underline{EV} - \underline{PV}$$

Si ha quindi che:

- Se  $SV > 0$  il progetto è in anticipo rispetto alla pianificazione;
- Se  $SV = 0$  il progetto è in linea con la pianificazione;
- Se  $SV < 0$  il progetto è in ritardo rispetto alla pianificazione.

## **Swagger**

Insieme di strumenti che permette di documentare le API contestualmente alla loro definizione nel codice e testarle usando una interfaccia web.

# **T**

## **Test di accettazione**

Test che verifica che il sistema soddisfi i requisiti utente specificati nel capitolato e concordati con il proponente. Corrisponde all'ultima fase di test prima del possibile rilascio del prodotto.

## **Test di integrazione**

Test che verifica che le varie componenti o unità del sistema funzionino correttamente quando vengono integrate tra loro. Si svolge dopo i test di unità e prima dei test di sistema.

Esistono due strategie di integrazione:

- Bottom-up: si integrano prima le componenti/unità di livello più basso, più interne al sistema e con meno dipendenze;
- Top-down: si integrano prima le componenti/unità di livello più alto, più esterne al sistema e con più dipendenze.

## Test di regressione

Test che verifica che le modifiche apportate al software non abbiano introdotto difetti in parti del software che funzionavano correttamente in precedenza. Si svolge dopo ogni modifica al codice ed è una ripetizione di altri test già eseguiti precedentemente.

## Test di sistema

Test che verifica che il sistema soddisfi i requisiti software specificati nel documento di analisi dei requisiti. Si svolge dopo i test di integrazione e prima dei test di accettazione.

## Test di unità

Test che esamina una singola unità software. Si divide in funzionale e strutturale:

- Funzionale (black-box):
  - esaminano solo il legame tra input e output;
  - utilizzano dati in ingresso che corrispondono a specifici esiti: valori del dominio, valori di limite e valori illegali.
- Strutturale (white-box):
  - verificano la logica interna dell'unità software;
  - un singolo caso di prova attiva un singolo cammino di esecuzione;
  - l'obiettivo è quello di avere un insieme di casi di prova che riescono a far eseguire tutti i comandi e attraversare tutti i branch dell'unità in esame.

## Tracciamento

Attività che permette di mantenere la coerenza tra i requisiti e i componenti del sistema, in modo da poter tracciare l'evoluzione del sistema e verificare che i requisiti siano soddisfatti.

## TypeScript

TypeScript è un linguaggio di programmazione open source. Il linguaggio estende la sintassi di JavaScript in modo che qualunque programma scritto in JavaScript sia anche in grado di funzionare con TypeScript senza nessuna modifica. È quindi destinato a essere compilato in JavaScript per poter essere interpretato da qualunque web browser o app.

## Typst

Typst è un sistema di impaginazione basato su markup progettato per essere un'alternativa a LaTeX pur essendo più facile da apprendere ed utilizzare.

# U

## UML

Acronimo di *Unified Modeling Language*, è un linguaggio di modellazione utilizzato per descrivere, visualizzare e documentare i componenti di un sistema software.

## User Story

Breve descrizione di una funzionalità del software, scritta dal punto di vista dell'utente, che descrive chi è l'utente, cosa vuole e perché.

## V

### **Validazione**

Si occupa di accertare che il prodotto finale soddisfi i requisiti specificati e corrisponda alle attese (Did I build the right system?) quindi pone l'attenzione sul prodotto finale.

### **Verbale**

Il verbale è un documento che registra in dettaglio ciò che è stato discusso e deciso in una riunione. Viene utilizzato per fare riferimento a decisioni e le rispettive azioni da intraprendere in seguito. Il verbale è definito interno quando si svolge solamente tra i membri del team, esterno altrimenti. Viene redatto dal responsabile o dall'amministratore (se delegato).

### **Verifica**

Si occupa di accertare che lo svolgimento delle attività di sviluppo non introduca errori (Did I build the system right?) quindi pone l'attenzione sul way of working.

## W

### **Way of working**

Insieme di regole e pratiche che stabiliscono come devono avvenire i processi del progetto.

### **Workflow**

Il workflow è un elemento strettamente collegato con il capitolato. Serie di attività orchestrate e ripetibili, eseguite in sequenza o in parallelo, con lo scopo di raggiungere un obiettivo specifico.

## X

## Y

## Z