



UNIVERSITÀ DEGLI STUDI DI PADOVA  
CORSO DI LAUREA IN INFORMATICA (L-31)

Corso di Ingegneria del Software  
Anno Accademico 2025/2026

# Glossario

Redattori: Giovanni Ponso, Davide Biasuzzi, Francesco Zanella

**Gruppo: NightPRO**

[swe.nightpro@gmail.com](mailto:swe.nightpro@gmail.com)

Data: 2026-01-30

Versione: 1.5

## Tabella delle Versioni

Versione	Data	Autore/i	Descrizione delle Modifiche	Verificatore
1.5	2026-01-30	Davide Biasuzzi	Aggiunti i termini Code Smell, ISO/IEC 25010, Linter, Microservizi, PB (Product Baseline), RAG (Retrieval-Augmented Generation), Refactoring, RTB (Requirements and Technology Baseline), Unit Test per supporto Piano di Qualifica v0.3	Francesco Zanella
1.4	2026-01-28	Davide Biasuzzi	Aggiunti i termini camelCase, Ciclo di Deming, Earned Value Management (EVM), Gulpease, PascalCase, PDCA e Penpot per supporto Norme di Progetto v1.3	Francesco Zanella
1.3	2026-01-04	Davide Biasuzzi	Aggiunti i termini Embedding, LLM, NER, RBAC, Speech-to-Text per supporto Analisi dei Requisiti v0.3	Leonardo Bilato
1.2	2025-12-04	Francesco Zanella	Aggiunti i termini API, Baseline, Build, Bug, Bug tracking system, CI (Continuous Integration), Consuntivo, Docker, ERP, Framework, Github Actions, Intelligenza Artificiale, Machine Learning, Merge, Mitigazione, Modello di sviluppo, MVP, NLP, Overleaf, Patch, PDF, Pipeline, Preventivo, Product Backlog, Scrum, Snake case, Sprint, Sprint Backlog, Sprint Planning, Sprint Retrospective, Sprint Review, Template, Versionamento	Giovanni Ponso
1.1	2025-11-16	Davide Biasuzzi	Inseriti termini Agile, Ciclo di Vita, Best practices, Issue, Milestone, Piano di Progetto, Piano di Qualifica, Branch, Pull Request, Architettura, Accoppiamento, Pattern Architeturali, Schema UML, Anlisi dei Requisiti, Requisiti, Specifiche Funzionali/Tecniche, Contesto applicativo, Azienda Proponente, Committente, Feedback, Proof of Concept (PoC)	Francesco Zanella
1.0	2025-11-05	Davide Biasuzzi	Inserimento definizioni iniziali mancanti	Francesco Zanella
0.1	2025-11-05	Giovanni Ponso	Creazione bozza glossario e inserimento definizioni	Francesco Zanella

## Indice

<b>Tabella delle Versioni</b>	<b>2</b>
<b>1 Informazioni Generali</b>	<b>6</b>
1.1 Componenti del Gruppo . . . . .	6
<b>2 Introduzione</b>	<b>6</b>
2.1 Obiettivo del documento . . . . .	6
2.2 Struttura del documento . . . . .	6
<b>A</b>	<b>7</b>
Accoppiamento . . . . .	7
Agile . . . . .	7
Analisi dei Requisiti . . . . .	7
API . . . . .	7
Approvazione . . . . .	7
Architettura . . . . .	7
Azienda Proponente . . . . .	7
<b>B</b>	<b>8</b>
Baseline . . . . .	8
Best practices . . . . .	8
Branch . . . . .	8
Bug . . . . .	8
Bug tracking system . . . . .	8
Build . . . . .	8
<b>C</b>	<b>8</b>
Capitolato . . . . .	8
camelCase . . . . .	8
CI (Continuous Integration) . . . . .	9
Ciclo di vita . . . . .	9
Ciclo di Deming . . . . .	9
Code Smell . . . . .	9
Commit . . . . .	9
Committente . . . . .	9
Consuntivo . . . . .	9
Contesto applicativo . . . . .	9
<b>D</b>	<b>10</b>
Diario della riunione . . . . .	10
Docker . . . . .	10
<b>E</b>	<b>10</b>
Embedding . . . . .	10
ERP . . . . .	10
Earned Value Management (EVM) . . . . .	10
<b>F</b>	<b>10</b>
Feedback . . . . .	10
Framework . . . . .	10

Frontespizio . . . . .	11
<b>G</b>	<b>11</b>
Git . . . . .	11
Github . . . . .	11
Github Actions . . . . .	11
Github Pages . . . . .	11
Github Projects . . . . .	11
Glossario . . . . .	11
Gulpease . . . . .	11
Google Meet . . . . .	12
<b>I</b>	<b>12</b>
Intelligenza Artificiale (IA) . . . . .	12
Issue . . . . .	12
ISO/IEC 25010 . . . . .	12
<b>L</b>	<b>12</b>
LaTeX . . . . .	12
LLM . . . . .	12
Linter . . . . .	12
<b>M</b>	<b>13</b>
Machine Learning . . . . .	13
Merge . . . . .	13
Milestone . . . . .	13
Mitigazione . . . . .	13
Microservizi . . . . .	13
Modello di sviluppo . . . . .	13
MVP . . . . .	13
<b>N</b>	<b>14</b>
NER . . . . .	14
NLP . . . . .	14
Norme di Progetto . . . . .	14
<b>O</b>	<b>14</b>
Ordine del giorno . . . . .	14
Overleaf . . . . .	14
<b>P</b>	<b>14</b>
Patch . . . . .	14
Penpot . . . . .	14
PascalCase . . . . .	15
Pattern architetturali . . . . .	15
PDF . . . . .	15
PDCA . . . . .	15
PB (Product Baseline) . . . . .	15
Piano di Progetto . . . . .	15
Piano di Qualifica . . . . .	15
Pipeline . . . . .	15
Preventivo . . . . .	16

Product Backlog . . . . .	16
Proof of Concept (PoC) . . . . .	16
Pubblicazione . . . . .	16
Pull Request . . . . .	16
<b>R</b>	<b>16</b>
RAG (Retrieval-Augmented Generation) . . . . .	16
Redazione . . . . .	16
Refactoring . . . . .	16
Repository . . . . .	17
Requisiti . . . . .	17
RBAC . . . . .	17
RTB (Requirements and Technology Baseline) . . . . .	17
<b>S</b>	<b>17</b>
Schema UML . . . . .	17
Scrum . . . . .	17
Snake case . . . . .	17
Specifiche funzionali . . . . .	17
Specifiche tecniche . . . . .	18
Sprint . . . . .	18
Sprint Backlog . . . . .	18
Sprint Planning . . . . .	18
Sprint Retrospective . . . . .	18
Sprint Review . . . . .	18
Speech-to-Text . . . . .	18
Stakeholder . . . . .	19
<b>T</b>	<b>19</b>
Task . . . . .	19
Telegram . . . . .	19
Template . . . . .	19
Topic (Telegram) . . . . .	19
<b>U</b>	<b>19</b>
Unit Test . . . . .	19
<b>V</b>	<b>19</b>
Verbali . . . . .	19
Verifica . . . . .	20
Versionamento . . . . .	20
<b>W</b>	<b>20</b>
Workflow . . . . .	20

## 1 Informazioni Generali

### 1.1 Componenti del Gruppo

Cognome	Nome	Matricola
Biasuzzi	Davide	2111000
Bilato	Leonardo	2071084
Zanella	Francesco	2116442
Romascu	Mihaela-Mariana	2079726
Ogniben	Michele	2042325
Perozzo	Samuele	2110989
Ponso	Giovanni	2000558

Tabella 1: Componenti del gruppo NightPRO.

## 2 Introduzione

### 2.1 Obiettivo del documento

Il presente glossario raccoglie i termini tecnici, specialistici o potenzialmente ambigui utilizzati nella documentazione del progetto. Il suo scopo è garantire una comprensione uniforme del linguaggio adottato dal gruppo, fornendo definizioni chiare e non equivoche.

### 2.2 Struttura del documento

I termini sono organizzati in ordine alfabetico. Alla loro prima apparizione nei documenti ufficiali, essi vengono contrassegnati dal pedice <sub>G</sub>, che ne indica la presenza nel glossario. Le occorrenze successive non riportano tale marcatura.

## A

### Accoppiamento

Il grado di interdipendenza tra i componenti di un sistema software. Un basso accoppiamento è desiderabile in quanto facilita la manutenibilità e la modifica indipendente dei componenti.

### Agile

Metodologia di sviluppo software che favorisce la collaborazione continua, la flessibilità e l'iterazione rapida. Permette di adattare il progetto ai cambiamenti dei requisiti attraverso cicli di sviluppo brevi e incrementali.

### Analisi dei Requisiti

Documento che descrive in dettaglio i servizi che il sistema deve fornire, specificando i requisiti funzionali e non funzionali raccolti durante la fase di analisi.

### API

Acronimo di *Application Programming Interface*. Insieme di definizioni, protocolli e strumenti che permettono a software diversi di comunicare tra loro, astruendo la complessità dell'implementazione sottostante.

### Approvazione

Fase formale del ciclo di vita di un documento, successiva alla verifica, in cui si certifica che il documento è completo, corretto e pronto per il rilascio ufficiale.

### Architettura

La struttura organizzativa di un sistema software che definisce i componenti principali, le loro relazioni e i principi di design che guidano la sua evoluzione.

### Azienda Proponente

L'organizzazione o società che propone il capitolato d'appalto e con cui il gruppo collabora per la realizzazione del progetto. Nel contesto del progetto NightPRO, l'azienda proponente è Ergon Informatica.

## B

### Baseline

Una versione specifica e stabile di un elemento di configurazione (documento o codice) che è stata formalmente approvata e che funge da base per le attività successive. Può essere modificata solo attraverso procedure formali di controllo delle modifiche.

### Best practices

Insieme di metodologie, tecniche e approcci consolidati e riconosciuti come i più efficaci per raggiungere un determinato obiettivo con qualità ed efficienza.

### Branch

Una linea di sviluppo indipendente nel sistema di versionamento Git. Permette di lavorare su modifiche isolate dal codice principale (main) fino al momento dell'integrazione.

### Bug

Un difetto o imperfezione nel software che causa un comportamento imprevisto, errato o diverso dalle specifiche funzionali definite.

### Bug tracking system

Applicazione software progettata per tracciare e gestire i bug segnalati nei progetti di sviluppo software, monitorandone lo stato dalla segnalazione alla risoluzione.

### Build

Il processo di conversione del codice sorgente in un artefatto software eseguibile o in un pacchetto distribuibile. Può includere compilazione, linking e packaging.

## C

### Capitolato

Documento fornito dal committente che specifica i requisiti, gli obiettivi e i vincoli del progetto da realizzare.

### camelCase

Convenzione di scrittura per identificatori in cui la prima parola inizia con lettera minuscola e le parole successive iniziano con lettera maiuscola, senza separatori (es. `nomeVariabile`, `calcolaTotale`). Utilizzata tipicamente per nomi di metodi, funzioni e variabili.



**CI (Continuous Integration)**

Pratica di sviluppo software che consiste nell'integrare frequentemente le modifiche al codice in un repository condiviso. Tale processo innesca automaticamente build e suite di test per validare immediatamente la qualità del nuovo codice integrato.

**Ciclo di vita**

L'insieme delle fasi attraverso cui passa un prodotto o un documento, dalla concezione alla dismissione, includendo sviluppo, verifica, approvazione e pubblicazione.

**Ciclo di Deming**

Modello iterativo di gestione della qualità, noto anche come PDCA (Plan-Do-Check-Act), che prevede quattro fasi cicliche: pianificazione degli obiettivi, esecuzione delle attività, verifica dei risultati e implementazione delle azioni correttive. Ideato da W. Edwards Deming, è uno strumento fondamentale per il miglioramento continuo dei processi.

**Code Smell**

Indicatore di potenziali problemi nel codice sorgente che, pur non costituendo un bug, suggerisce debolezze nel design che possono ostacolare lo sviluppo futuro o aumentare il rischio di introdurre difetti. Esempi comuni includono metodi troppo lunghi, classi con troppe responsabilità e codice duplicato.

**Commit**

Una singola operazione di salvataggio registrata nel sistema di versionamento (come Git). Rappresenta un insieme di modifiche apportate ai file del repository.

**Committente**

Il soggetto (il docente responsabile del corso) che richiede formalmente il progetto e a cui viene presentata l'offerta e la documentazione ufficiale.

**Consuntivo**

Documento o sezione del Piano di Progetto che riporta il resoconto delle ore e dei costi effettivamente sostenuti in un determinato periodo, confrontandoli con il preventivo per valutare l'andamento economico del progetto.

**Contesto applicativo**

L'ambiente operativo e il dominio di utilizzo del prodotto software, comprendente gli utenti target, i casi d'uso e le condizioni operative.

## D

### Diario della riunione

Sezione di un verbale che contiene il resoconto dettagliato della discussione, tipicamente suddiviso rispecchiando i punti dell'Ordine del Giorno.

### Docker

Piattaforma open source che permette di sviluppare, distribuire ed eseguire applicazioni all'interno di container, garantendo che il software funzioni uniformemente indipendentemente dall'ambiente in cui viene eseguito.

## E

### Embedding

Rappresentazione numerica (vettoriale) di dati testuali o audio che cattura le caratteristiche semantiche in uno spazio multidimensionale. Gli embedding permettono a modelli di Machine Learning di elaborare e confrontare informazioni non strutturate.

### ERP

Acronimo di *Enterprise Resource Planning*. Un software di gestione che integra tutti i processi di business rilevanti di un'azienda (vendite, acquisti, gestione magazzino, contabilità, ecc.) in un unico sistema.

### Earned Value Management (EVM)

Metodologia di gestione progetti che integra ambito, tempi e costi per misurare oggettivamente le prestazioni e l'avanzamento del progetto. Attraverso metriche come Earned Value, Actual Cost e Planned Value, permette di calcolare variazioni e indici di performance per prevedere l'andamento futuro del progetto.

## F

### Feedback

Informazioni di ritorno fornite durante la revisione del lavoro svolto, finalizzate a segnalare errori, suggerire miglioramenti e garantire la qualità del prodotto.

### Framework

Un ambiente software che fornisce funzionalità generiche e una struttura di supporto standardizzata per lo sviluppo di applicazioni, facilitando il lavoro dei programmatori attraverso il riutilizzo del codice.

## **Frontespizio**

La prima pagina di un documento ufficiale che contiene i metadati identificativi: titolo, autori, gruppo, versione, data, e contesto (es. università, corso).

## **G**

### **Git**

Sistema di controllo versione distribuito (DVCS) creato da Linus Torvalds. È lo strumento software che traccia la cronologia delle modifiche ai file di progetto (sorgenti, documenti) e permette la collaborazione.

### **Github**

Piattaforma web basata su Git per l'hosting di repository, la gestione del versionamento del codice e la collaborazione allo sviluppo software.

### **Github Actions**

Piattaforma di integrazione continua e distribuzione continua (CI/CD) integrata in GitHub, che permette di automatizzare la compilazione, il test e la distribuzione del software attraverso workflow definiti.

### **Github Pages**

Servizio di hosting per siti web statici fornito da GitHub. Utilizzato dal gruppo per pubblicare e rendere consultabile la documentazione di progetto.

### **Github Projects**

Strumento di gestione progettuale (project management) integrato in GitHub, utilizzato per pianificare, organizzare e tracciare lo stato di avanzamento delle attività (task).

## **Glossario**

Documento che raccoglie e definisce i termini tecnici, specialistici o potenzialmente ambigui utilizzati nel progetto, al fine di garantirne una comprensione uniforme.

## **Gulpease**

Indice di leggibilità specifico per la lingua italiana, sviluppato dal GULP (Gruppo Universitario Linguistico Pedagogico). Misura la facilità di lettura di un testo su una scala da 0 a 100, dove valori più alti indicano maggiore leggibilità. Considera la lunghezza delle parole e delle frasi rispetto al numero totale di lettere.

## Google Meet

Piattaforma di videoconferenza utilizzata dal gruppo per le riunioni sincrone e le attività collaborative che richiedono condivisione dello schermo.

## I

### Intelligenza Artificiale (IA)

L'IA è il ramo dell'informatica che progetta sistemi capaci di eseguire compiti che richiederebbero intelligenza se svolti da esseri umani, inclusi la percezione, il ragionamento, l'apprendimento e la risoluzione di problemi complessi.

### Issue

Un'attività, un problema o una richiesta tracciata nel sistema di gestione del progetto (GitHub Projects). Ogni issue rappresenta un elemento di lavoro assegnabile e monitorabile.

### ISO/IEC 25010

Standard internazionale che definisce un modello di qualità del prodotto software, identificando otto caratteristiche principali: funzionalità, efficienza prestazionale, compatibilità, usabilità, affidabilità, sicurezza, manutenibilità e portabilità. Fornisce un framework di riferimento per valutare e misurare la qualità del software.

## L

### LaTeX

Linguaggio di markup e sistema di preparazione di documenti ampiamente usato in ambito accademico per la sua elevata qualità tipografica e la gestione di formule complesse. È il formato sorgente per la documentazione ufficiale del progetto.

### LLM

Acronimo di *Large Language Model* (Modello Linguistico di Grandi Dimensioni). Modello di intelligenza artificiale addestrato su enormi quantità di testo per comprendere e generare linguaggio naturale, utilizzato per attività come elaborazione semantica, traduzione e generazione di contenuti.

### Linter

Strumento di analisi statica del codice che identifica errori sintattici, problemi stilistici e potenziali bug senza eseguire il programma. Aiuta a mantenere uno stile di codifica consistente e a prevenire errori comuni prima della compilazione o dell'esecuzione.

## M

### Machine Learning

Sottocampo dell'Intelligenza Artificiale che si occupa di creare sistemi in grado di apprendere dai dati, identificando modelli e prendendo decisioni con il minimo intervento umano.

### Merge

Operazione di unione del contenuto di due branch distinti in un unico branch. In Git, è il processo fondamentale per integrare il lavoro svolto parallelamente.

### Milestone

Un punto di controllo significativo nel progetto che rappresenta il completamento di un insieme di attività o il raggiungimento di un obiettivo intermedio. Utilizzata per monitorare i progressi verso gli obiettivi principali.

### Mitigazione

Nell'ambito della gestione dei rischi, indica l'insieme delle azioni pianificate per ridurre la probabilità che un rischio si verifichi o per limitarne l'impatto negativo sul progetto qualora accadesse.

### Microservizi

Stile architetturale che struttura un'applicazione come una collezione di servizi piccoli, indipendenti e debolmente accoppiati. Ogni microservizio implementa una funzionalità specifica, può essere sviluppato e distribuito in modo autonomo, e comunica con gli altri tramite API ben definite.

### Modello di sviluppo

La struttura metodologica utilizzata per organizzare i processi di sviluppo del software (es. Modello a cascata, Modello Agile, Modello a V), definendo l'ordine e la natura delle fasi del ciclo di vita.

### MVP

Acronimo di *Minimum Viable Product* (Prodotto Minimo Funzionante). La versione di un nuovo prodotto che include solo le caratteristiche fondamentali necessarie per essere rilasciata ai primi utenti e raccogliere feedback per gli sviluppi futuri.

## N

### NER

Acronimo di *Named Entity Recognition* (Riconoscimento di Entità Nominate). Tecnica di NLP che identifica ed estrae automaticamente entità specifiche da testi non strutturati, come nomi di persone, luoghi, quantità, date e codici prodotto.

### NLP

Acronimo di *Natural Language Processing* (Elaborazione del Linguaggio Naturale). Campo dell'intelligenza artificiale che si occupa dell'interazione tra computer e linguaggio umano, permettendo alle macchine di leggere, decifrare e comprendere le lingue umane.

### Norme di Progetto

Il documento che definisce il metodo di lavoro, le regole redazionali, gli strumenti e i processi di gestione adottati dal gruppo per garantire coerenza, tracciabilità e qualità.

## O

### Ordine del giorno

Elenco degli argomenti (chiamato anche "Agenda") pianificati per la discussione durante una riunione. È una sezione obbligatoria dei verbali.

### Overleaf

Editor LaTeX collaborativo in cloud, utilizzato dal gruppo per la redazione simultanea, la compilazione e la gestione della documentazione di progetto in tempo reale.

## P

### Patch

Un aggiornamento software rilasciato per correggere bug, vulnerabilità di sicurezza o per migliorare l'usabilità e le prestazioni di un programma esistente.

### Penpot

Piattaforma open source di design e prototipazione per interfacce utente, che permette la creazione collaborativa di wireframe, mockup e prototipi interattivi. Alternative open source a strumenti proprietari come Figma, offre funzionalità avanzate per il design di interfacce web e mobile.

**PascalCase**

Convenzione di scrittura per identificatori in cui ogni parola inizia con lettera maiuscola, senza separatori (es. `NomeClasse`, `CalcolatorePrezzi`). Utilizzata tipicamente per nomi di classi e interfacce.

**Pattern architetturali**

Soluzioni progettuali riutilizzabili che descrivono l'organizzazione strutturale di sistemi software, fornendo template collaudati per risolvere problemi ricorrenti di design.

**PDF**

Acronimo di *Portable Document Format*. Formato di file utilizzato per presentare e scambiare documenti in modo affidabile, indipendentemente dal software, dall'hardware o dal sistema operativo. È il formato finale di pubblicazione dei documenti del gruppo.

**PDCA**

Acronimo di *Plan-Do-Check-Act*. Modello iterativo a quattro fasi per il miglioramento continuo dei processi: Plan (pianificare obiettivi e processi), Do (eseguire il piano), Check (verificare i risultati rispetto agli obiettivi), Act (agire per migliorare). Noto anche come Ciclo di Deming.

**PB (Product Baseline)**

Seconda revisione di avanzamento del progetto didattico, successiva all'RTB. Rappresenta il punto in cui il team dimostra di aver sviluppato un prodotto software funzionante (MVP) che soddisfa i requisiti minimi concordati, corredato dalla documentazione di progettazione e test.

**Piano di Progetto**

Documento che descrive la pianificazione temporale del progetto, l'allocazione delle risorse, la suddivisione del lavoro e la gestione dei rischi.

**Piano di Qualifica**

Documento che definisce le strategie di verifica e validazione adottate dal gruppo, descrivendo le tecniche, le metriche e i test utilizzati per garantire la qualità del prodotto.

**Pipeline**

In ambito CI/CD, una serie di passaggi automatizzati (workflow) che il software attraversa, dalla scrittura del codice alla compilazione, ai test e infine al deployment.

**Preventivo**

Documento o sezione del Piano di Progetto che stima le risorse economiche e temporali necessarie per completare il progetto, calcolate in base alle ore produttive pianificate per ciascun ruolo.

**Product Backlog**

In Scrum, è la lista di tutto ciò che si sa essere necessario nel prodotto. È l'unica fonte dei requisiti per ogni modifica da apportare al prodotto.

**Proof of Concept (PoC)**

Dimostrazione realizzata per verificare la fattibilità tecnica di una soluzione proposta, utilizzata per validare le scelte tecnologiche prima dell'implementazione completa.

**Pubblicazione**

Fase finale del ciclo di vita di un documento che consiste nel renderlo accessibile agli stakeholder, ad esempio caricandolo nell'archivio PDF sul sito web del gruppo.

**Pull Request**

Richiesta di integrazione delle modifiche da un branch al branch principale del repository. Permette la revisione del codice o della documentazione prima del merge definitivo.

**R****RAG (Retrieval-Augmented Generation)**

Architettura di intelligenza artificiale che combina la generazione di testo di un LLM con il recupero di informazioni da una base di conoscenza esterna. Permette al modello di accedere a dati aggiornati e specifici del dominio non presenti nel suo addestramento originale, migliorando l'accuratezza e la pertinenza delle risposte.

**Redazione**

Fase iniziale del ciclo di vita di un documento, durante la quale uno o più autori (redattori) creano e aggiornano i contenuti.

**Refactoring**

Processo di ristrutturazione del codice esistente senza modificarne il comportamento esterno. L'obiettivo è migliorare la leggibilità, la manutenibilità e la struttura interna del codice, riducendo la complessità e il debito tecnico.



## Repository

Archivio centrale gestito da un sistema di versionamento (come Git) che contiene tutti i file del progetto (codice sorgente, documenti) e la cronologia completa delle loro modifiche.

## Requisiti

Le necessità e le aspettative che il prodotto software deve soddisfare, classificati in funzionali (cosa il sistema deve fare) e non funzionali (come il sistema deve comportarsi).

## RBAC

Acronimo di *Role-Based Access Control* (Controllo degli Accessi Basato sui Ruoli). Modello di gestione delle autorizzazioni che assegna permessi agli utenti in base al loro ruolo nel sistema, garantendo sicurezza e separazione dei privilegi.

## RTB (Requirements and Technology Baseline)

Prima revisione di avanzamento del progetto didattico. Rappresenta il punto in cui il team dimostra di aver compreso i requisiti del capitolato, definito l'analisi dei requisiti e validato le scelte tecnologiche attraverso un Proof of Concept (PoC).

## S

### Schema UML

Diagramma creato utilizzando il linguaggio di modellazione unificato (Unified Modeling Language) per rappresentare visivamente la struttura, il comportamento e le interazioni di un sistema software.

## Scrum

Un framework Agile per la gestione di progetti complessi. Si basa su iterazioni brevi (Sprint), ruoli definiti e cerimonie regolari per favorire la trasparenza, l'ispezione e l'adattamento.

## Snake case

Convenzione di scrittura per identificatori in cui le parole sono separate da un trattino basso (\_\_) e scritte in minuscolo (es. `nome_variabale`).

## Specifiche funzionali

Descrizioni dettagliate delle funzionalità che il sistema deve fornire, definendo il comportamento atteso in risposta a determinati input o condizioni.

## **Specifiche tecniche**

Descrizioni dettagliate degli aspetti tecnici del sistema, incluse le tecnologie, le architetture, i protocolli e le interfacce da utilizzare.

## **Sprint**

In Scrum, un periodo di tempo prefissato (time-box), solitamente da una a quattro settimane, durante il quale viene creato un incremento di prodotto "Done", utilizzabile e potenzialmente rilasciabile.

## **Sprint Backlog**

Insieme degli elementi selezionati dal Product Backlog per essere sviluppati durante lo Sprint corrente, corredato da un piano attuativo per la loro realizzazione. È uno strumento di pianificazione dinamico, gestito dagli Sviluppatori, che visualizza in tempo reale tutto il lavoro necessario per conseguire lo Sprint Goal.

## **Sprint Planning**

Evento collaborativo che inaugura lo Sprint, durante il quale l'intero Scrum Team definisce l'obiettivo da raggiungere (Sprint Goal) e pianifica le attività necessarie. In questa fase, il team seleziona gli elementi dal Product Backlog in base alla propria capacità produttiva e stabilisce il piano operativo per trasformarli in un Incremento di valore entro la fine dello Sprint.

## **Sprint Retrospective**

Riunione conclusiva dello Sprint focalizzata sul processo. Il team ispeziona sé stesso per identificare miglioramenti da applicare nello Sprint successivo.

## **Sprint Review**

Evento che si tiene al termine dello Sprint per ispezionare l'Incremento di prodotto realizzato e valutare i progressi rispetto all'Obiettivo di Prodotto. Coinvolge lo Scrum Team e gli stakeholder in una sessione di lavoro volta a raccogliere feedback e, se necessario, ad aggiornare il Product Backlog per ottimizzare il valore delle versioni future.

## **Speech-to-Text**

Tecnologia di intelligenza artificiale che converte il parlato umano in testo scritto. Utilizza modelli di riconoscimento vocale per trascrivere file audio in formato testuale, gestendo variabilità come accenti, dialetti e qualità di registrazione.

**Stakeholder**

Qualsiasi individuo o gruppo che ha un interesse nel progetto o è influenzato dal suo risultato (es. membri del team, committenti, docenti).

**T****Task**

Un'attività o un compito specifico e tracciabile che deve essere svolto. La gestione dei task avviene tramite GitHub Projects.

**Telegram**

Applicazione di messaggistica istantanea usata come canale di comunicazione principale dal gruppo.

**Template**

Modello predefinito di documento che contiene la struttura, lo stile e le impostazioni grafiche standard (come loghi e intestazioni), utilizzato per garantire uniformità in tutta la documentazione di progetto. s

**Topic (Telegram)**

Funzionalità dei gruppi Telegram che permette di suddividere le conversazioni in argomenti specifici, garantendo maggiore ordine e tracciabilità delle discussioni.

**U****Unit Test**

Test automatizzato che verifica il corretto funzionamento di una singola unità di codice (tipicamente un metodo o una funzione) in isolamento dalle altre componenti del sistema. Costituisce il primo livello della piramide dei test e permette di identificare rapidamente regressioni nel codice.

**V****Verbali**

Documenti ufficiali che riportano le discussioni, le decisioni prese e le attività assegnate durante gli incontri del gruppo, al fine di tracciare l'evoluzione del progetto.

**Verifica**

Fase del ciclo di vita di un documento in cui un membro del gruppo, diverso dal redattore, controlla la correttezza formale, la qualità e la coerenza dei contenuti prima dell'approvazione o pubblicazione.

**Versionamento**

La gestione delle diverse versioni di un documento o di un software. Permette di tracciare la storia delle modifiche, ripristinare stati precedenti e gestire lo sviluppo concorrente (vedi anche Git).

**W****Workflow**

Una sequenza configurabile di operazioni automatizzate. Nel contesto di GitHub Actions, definisce i passi per processi come la compilazione o la pubblicazione. Sinonimo di Pipeline.