



## BugBusters

Email: bugbusters.unipd@gmail.com

Gruppo: 4

Università degli Studi di Padova

Laurea in Informatica

CORSO: Ingegneria del Software

Anno Accademico: 2025/2026

## Glossario

Versione 1.0.1

<b>Stato</b>	In redazione
<b>Redattori</b>	Marco Favero, Alberto Pignat, Marco Piro, Linor Sade, Leonardo Salviato, Luca Slongo
<b>Verificatori</b>	Alberto Pignat
<b>Destinatari</b>	Prof. Tullio Vardanega, Prof. Riccardo Cardin, BugBusters, Eggon

## Registro delle modifiche

Versione	Data	Descrizione	Redatto	Verificato	Approvato
1.0.1	27/02/2026	Sono stati aggiunti nuovi termini	Alberto Autiero		-
1.0.0	30/01/2026	Approvazione del documento	-		Luca Slongo
0.1.0	30/01/2026	Verifica del documento	-	Alberto Pignat	-
0.0.11	29/01/2026	Sono stati aggiunti nuovi termini	Luca Slongo	-	-
0.0.10	06/01/2026	Sono stati aggiunti nuovi termini	Leonardo Salviato	-	-
0.0.9	23/12/2025	Sono stati aggiunti nuovi termini	Linor Sadé	-	-
0.0.8	14/12/2025	Sono stati aggiunti nuovi termini	Marco Piro	-	-
0.0.7	14/12/2025	Riorganizzazione documento	Marco Piro	-	-
0.0.6	12/12/2025	Sono stati aggiunti nuovi termini	Marco Piro	-	-
0.0.5	06/12/2025	Sono stati aggiunti nuovi termini	Alberto Pignat	-	-
0.0.4	30/11/2025	Sono stati aggiunti nuovi termini	Alberto Pignat	-	-
0.0.3	23/11/2025	Sono stati aggiunti nuovi termini	Marco Favero	-	-
0.0.2	17/11/2025	Sono stati aggiunti nuovi termini	Marco Favero	-	-
0.0.1	10/11/2025	Prima stesura del documento con i termini del capitolato aggiudicato (5) e diapositive dei docenti	Marco Favero	-	-

## Indice

A	4
Actual Cost (AC) . . . . .	4
Affidabilità (Reliability) . . . . .	4
Agile . . . . .	4
AI (Artificial Intelligence) . . . . .	4
AI Co-Pilot . . . . .	4
AI generativa . . . . .	4
Amministratore . . . . .	4
Analisi dei capitolati . . . . .	4
Analisi dei Requisiti . . . . .	5
Analista . . . . .	5
Angular . . . . .	5
API (Application Programming Interface) . . . . .	5
Approvazione . . . . .	5
Attore . . . . .	5
Attore secondario . . . . .	5
AWS (Amazon Web Services) . . . . .	5
B	6
Back-end . . . . .	6
Backlog . . . . .	6
Baseline . . . . .	6
Bedrock . . . . .	6
Best Practice . . . . .	6
Branch Coverage . . . . .	7
C	7
Candidatura . . . . .	7
Capitolato . . . . .	7
Casi d'uso . . . . .	7
Cedolini . . . . .	7
Cedolini Massivi . . . . .	7
Code Coverage . . . . .	7
Committente . . . . .	7
Complessità Ciclomatica (Cyclomatic Complexity) . . . . .	8
Consulente del Lavoro (CdL) . . . . .	8
Consuntivo . . . . .	8
Cost Performance Index (CPI) . . . . .	8
Cruscotto . . . . .	8
D	9

Dashboard . . . . .	9
Data Analyst . . . . .	9
Decisione esterna . . . . .	9
Decisione interna . . . . .	9
Discord . . . . .	9
Dispatch . . . . .	9
<b>E</b>	10
Earned Value (EV) . . . . .	10
Efficacia . . . . .	10
Efficienza . . . . .	10
Entity Resolution . . . . .	10
Error / Errore . . . . .	10
Estimate at Completion (EAC) . . . . .	11
Estimate to Complete (ETC) . . . . .	11
<b>F</b>	11
Front-end . . . . .	11
Funzionalità . . . . .	11
<b>G</b>	12
GDPR (General Data Protection Regulation) . . . . .	12
Gestione dei rischi . . . . .	12
GitHub . . . . .	12
Glossario . . . . .	12
Google Meet . . . . .	12
<b>H</b>	13
Human in the loop . . . . .	13
<b>I</b>	14
Implementazione . . . . .	14
Indice di Gulpease . . . . .	14
Interfaccia . . . . .	14
Issue . . . . .	14
<b>K</b>	15
KPI (Key Performance Indicator) . . . . .	15
<b>L</b>	16
Latex . . . . .	16
Lettera di candidatura . . . . .	16
LLM (Large Language Model) . . . . .	16
<b>M</b>	17
Manuale Utente . . . . .	17
Manutenibilità . . . . .	17

Microsoft Teams . . . . .	17
Milestone . . . . .	17
Minimum Viable Product (MVP) . . . . .	17
Mitigazione . . . . .	17
Modularità . . . . .	17
Modulo . . . . .	18
<b>N</b>	19
Norme di progetto . . . . .	19
<b>O</b>	20
OCR (Optical Character Recognition) . . . . .	20
<b>P</b>	21
PB (Product Baseline) . . . . .	21
Pericolosità . . . . .	21
Piano di Progetto . . . . .	21
Piano di Qualifica . . . . .	21
Planned Value (PV) . . . . .	21
POC (Proof of Concept) . . . . .	21
Postcondizioni . . . . .	21
PostgreSQL . . . . .	22
Precondizioni . . . . .	22
Preventivo . . . . .	22
Priorità (Obbligatorio/Desiderabile/Opzionale) . . . . .	22
Probabilità . . . . .	22
Prodotto (Product) . . . . .	22
Progettista . . . . .	22
Progetto . . . . .	22
Programmatore . . . . .	23
Prompt . . . . .	23
Proponente . . . . .	23
PWA (Progressive Web App) . . . . .	23
<b>Q</b>	24
Qualità . . . . .	24
<b>R</b>	25
Rating System . . . . .	25
RBAC (Role-Based Access Control) . . . . .	25
Redattore . . . . .	25
Repository . . . . .	25
Requirements Stability Index (RSI) . . . . .	25
Requisiti di qualità . . . . .	25

Requisiti di vincolo . . . . .	25
Requisiti funzionali . . . . .	25
Requisiti non funzionali . . . . .	26
Requisiti prestazionali . . . . .	26
Requisito . . . . .	26
Responsabile . . . . .	26
Retrospettiva . . . . .	26
Rischio . . . . .	26
Rischio Globale (RG) . . . . .	26
Rischio Individuale (RI) . . . . .	26
Rischio Tecnologico (RT) . . . . .	27
RTB (Requirements and Technology Baseline) . . . . .	27
Ruby . . . . .	27
Ruby on Rails . . . . .	27
 S	28
Sandbox . . . . .	28
Scalabilità . . . . .	28
Scenario . . . . .	28
Scenario principale . . . . .	28
Scenario secondario . . . . .	28
Schedule Performance Index (SPI) . . . . .	28
Scope . . . . .	28
SCRUM . . . . .	28
Sicurezza . . . . .	28
Specifica Tecnica . . . . .	29
Sprint . . . . .	29
Stakeholder . . . . .	29
Stand-alone . . . . .	29
Stima dei costi . . . . .	29
 T	30
Technology Baseline . . . . .	30
Template . . . . .	30
Test . . . . .	30
Tracciabilità . . . . .	30
 V	31
Validazione . . . . .	31
Verbale . . . . .	31
Verbale esterno . . . . .	31
Verbale interno . . . . .	31

Verifica . . . . .	31
Verificatore . . . . .	31
Versionamento/versioning . . . . .	31
<b>W</b>	<b>32</b>
Way of Working . . . . .	32
Workflow . . . . .	32

## A

### Actual Cost (AC)

Costo effettivamente sostenuto per completare il lavoro svolto in un determinato periodo. Include tutte le spese reali (risorse umane, strumenti) associate alle attività già eseguite e viene utilizzato per valutare l'efficienza economica del progetto.

### Affidabilità (Reliability)

Capacità di un sistema software di mantenere il proprio livello di prestazioni in condizioni specificate per un determinato periodo di tempo.

### Agile

Metodologia di sviluppo del software iterativa e incrementale che si basa su principi come la collaborazione continua con il cliente, la consegna frequente di software funzionante e la capacità di rispondere ai cambiamenti dei requisiti.

### AI (Artificial Intelligence)

Campo dell'informatica che si occupa di sviluppare sistemi in grado di svolgere compiti che normalmente richiedono l'intelligenza umana, come il riconoscimento vocale, la visione artificiale, l'elaborazione del linguaggio naturale e il processo decisionale.

### AI Co-Pilot

Modulo AI di NEXUM dedicato agli studi dei Consulenti del Lavoro per l'automazione della gestione documentale. Utilizza OCR, classificazione automatica, entity resolution e split di documenti massivi per riconoscere tipologia, destinatari e distribuire automaticamente cedolini, CU e comunicazioni ai dipendenti attraverso canali multipli (NEXUM App, email).

### AI generativa

Tipo di intelligenza artificiale in grado di creare autonomamente nuovi contenuti (testo, immagini, audio, codice) basandosi su pattern appresi dai dati di addestramento.

### Amministratore

Figura responsabile della gestione dell'infrastruttura, degli strumenti di sviluppo e dei processi del progetto. Si occupa di configurare e mantenere gli ambienti di lavoro e garantire che il team abbia a disposizione gli strumenti necessari.

### Analisi dei capitolati

Processo di studio e valutazione dei diversi capitolati d'appalto proposti dalle aziende proponenti, finalizzato a selezionare il progetto più adatto alle competenze, agli interessi e alle risorse del gruppo. Include la comprensione dei requisiti, delle tecnologie richieste e dei vincoli di progetto.

## Analisi dei Requisiti

Processo sistematico volto a identificare, documentare, classificare e validare i bisogni e i vincoli del progetto. Trasforma i requisiti utente (espressi nel capitolato, dal punto di vista del committente) in requisiti software (specificati tecnicamente, dal punto di vista dello sviluppatore), definendo "cosa" il sistema deve fare senza specificare "come". Produce il documento di Analisi dei Requisiti (AdR) e costituisce la base per la progettazione architettonale. Include attività di studio del dominio, definizione di casi d'uso, classificazione dei requisiti (obbligatori, desiderabili, opzionali; funzionali e non-funzionali) e tracciamento.

## Analista

Figura professionale specializzata nell'analisi dei requisiti e nella specifica delle funzionalità del sistema. Collabora con gli stakeholder per comprendere le esigenze e tradurle in specifiche tecniche.

## Angular

Framework JavaScript/TypeScript open-source sviluppato da Google per la costruzione di Single Page Applications (SPA) e applicazioni web dinamiche.

## API (Application Programming Interface)

Insieme di definizioni, protocolli e strumenti per la costruzione e l'integrazione di software applicativo. Consente a diversi sistemi di comunicare tra loro.

## Approvazione

Processo formale attraverso il quale un documento o una funzionalità viene accettata e considerata completa dopo aver superato le verifiche e validazioni necessarie.

## Attore

Ruolo svolto da un utente, un sistema o un'entità esterna che interagisce con il sistema software scambiando informazioni. Un attore rappresenta una classe di utenti piuttosto che un individuo specifico.

## Attore secondario

Attore che non avvia il caso d'uso, ma che deve essere consultato o informato dal sistema affinché l'obiettivo dell'attore primario sia raggiunto. Fornisce servizi, dati o riceve notifiche dal sistema.

## AWS (Amazon Web Services)

Piattaforma di cloud computing che offre servizi di calcolo, storage, database e altre funzionalità per supportare lo sviluppo e il deployment di applicazioni.

## B

### Back-end

Parte di un'applicazione software che gestisce la logica di business, l'elaborazione dei dati e la comunicazione con il database. Opera sul server ed è inaccessibile direttamente all'utente finale.

### Backlog

Elenco prioritizzato di attività, funzionalità o requisiti da completare nel corso del progetto. Nel contesto Agile e Scrum, rappresenta l'insieme di lavoro pianificato per gli sprint futuri. Il backlog viene costantemente aggiornato e riordinato in base alle priorità e alle necessità emergenti del progetto.

### Baseline

Versione approvata e formalmente controllata di un documento o di un componente software che serve come riferimento per sviluppi successivi. Le modifiche successive richiedono procedure formali di controllo. Può riferirsi a diverse fasi (Requirements Baseline, Design Baseline, Product Baseline) e garantisce stabilità e tracciabilità.

### Bedrock

Servizio di Amazon Web Services (AWS) che fornisce accesso a modelli di intelligenza artificiale di varie aziende leader attraverso un'API unificata. Consente di integrare e utilizzare Large Language Models (LLM) per applicazioni di AI generativa, eliminando la necessità di gestire direttamente l'infrastruttura dei modelli.

### Best Practice

Insieme di tecniche, metodi e procedure che sono state riconosciute come le più efficaci ed efficienti per raggiungere un obiettivo specifico in un determinato contesto.

## Branch Coverage

Metrica di testing che misura la percentuale di rami decisionali (branch) del codice eseguiti durante i test. Ogni struttura condizionale (if-else, switch-case) crea branch multipli e questa metrica verifica che entrambi i percorsi (vero/falso) di ogni condizione siano stati testati almeno una volta. Un valore elevato di branch coverage indica una migliore copertura dei possibili flussi di esecuzione del programma, riducendo il rischio di bug nascosti in percorsi non testati.

## C

### Candidatura

Processo con cui un gruppo di studenti presenta la propria offerta per aggiudicarsi un capitolato d'appalto nel corso di Ingegneria del Software, proponendo una soluzione che soddisfi i requisiti del proponente.

### Capitolato

Documento contrattuale che specifica i requisiti, le caratteristiche tecniche e le condizioni di un progetto software proposto da un'azienda proponente per il corso di Ingegneria del Software.

### Casi d'uso

Tecnica di specifica dei requisiti che descrive le interazioni tra gli attori (utenti o sistemi esterni) e il sistema software per raggiungere un obiettivo specifico.

### Cedolini

Documenti retributivi che attestano la retribuzione corrisposta al dipendente per un determinato periodo di paga.

### Cedolini Massivi

File contenenti più documenti retributivi aggregati, da suddividere e assegnare ai singoli destinatari tramite riconoscimento automatico.

### Code Coverage

Metrica che misura la percentuale di codice sorgente effettivamente eseguito durante l'esecuzione dei test automatizzati. Rappresenta il rapporto tra le linee di codice testate e il totale delle linee di codice del programma, espresso in percentuale. Un elevato code coverage indica che la maggior parte del codice è stata verificata attraverso i test, riducendo la probabilità di bug non rilevati. Tuttavia, un'alta percentuale non garantisce automaticamente la qualità dei test, che devono anche verificare correttamente il comportamento atteso del software.

### Committente

Soggetto che commissiona il progetto, definendone gli obiettivi, i vincoli e i requisiti, e che ricepisce il prodotto finale. Nel contesto del corso di Ingegneria del Software, può essere un'azienda proponente o un docente.

## Complessità Ciclomatica (Cyclomatic Complexity)

Metrica software che misura la complessità strutturale di un programma o di un singolo metodo calcolando il numero di percorsi linearmente indipendenti attraverso il codice sorgente. Viene calcolata contando il numero di punti decisionali (if, while, for, case) più uno. Un valore elevato indica codice complesso, difficile da testare, mantenere e comprendere. Valori tipici: 1-10 (semplice), 11-20 (moderatamente complesso), 21-50 (complesso), oltre 50 (non testabile). Viene utilizzata per identificare metodi che necessitano di refactoring per migliorare la manutenibilità.

## Consulente del Lavoro (CdL)

Professionista che assiste aziende e lavoratori nella gestione degli adempimenti amministrativi, previdenziali, assistenziali e fiscali relativi al rapporto di lavoro.

## Consuntivo

Documento di contabilità che riporta i costi effettivamente sostenuti durante una fase del progetto, confrontandoli con le stime iniziali (preventivo). Viene utilizzato per monitorare l'andamento economico del progetto e per apportare eventuali correzioni nelle fasi successive.

## Cost Performance Index (CPI)

Indice di efficienza dei costi che misura il rapporto tra il valore del lavoro completato (EV) e il costo effettivamente sostenuto (AC). Calcolato come  $CPI = EV/AC$ , un valore superiore a 1 indica che il progetto sta spendendo meno del previsto, mentre un valore inferiore a 1 segnala un sovraccosto.

## Cruscotto

Strumento di monitoraggio che fornisce una visualizzazione in tempo reale degli indicatori chiave di prestazione (KPI) e delle metriche di qualità del progetto. Consente di tracciare lo stato delle attività di verifica e validazione.

## D

### Dashboard

Interfaccia utente che presenta in forma grafica e sintetica le metriche, gli indicatori di performance e lo stato corrente del progetto o dell'applicazione.

### Data Analyst

Figura professionale che analizza dati per estrarre informazioni utili, identificare pattern e supportare decisioni di business attraverso metriche, report e visualizzazioni.

### Decisione esterna

Scelta presa da entità esterne al team di progetto (come il committente o il proponente) che vincola le attività del progetto e che il team deve rispettare.

### Decisione interna

Scelta presa dal team di progetto riguardante aspetti tecnici, organizzativi o metodologici, documentata per garantire tracciabilità e coerenza nelle attività successive.

### Discord

Piattaforma di comunicazione tramite chat vocale, testuale e video, utilizzata dal gruppo per la comunicazione interna e la collaborazione quotidiana.

### Dispatch

Processo di distribuzione automatizzata dei documenti verso i destinatari finali attraverso diversi canali (es. app, portale, email, PEC).

## E

### **Earned Value (EV)**

Valore del lavoro effettivamente completato in un determinato momento del progetto, espresso in termini monetari. Rappresenta il budget autorizzato per le attività completate e costituisce una metrica fondamentale per valutare l'avanzamento reale del progetto rispetto alla pianificazione iniziale.

### **Efficacia**

Misura della capacità di raggiungere gli obiettivi prefissati e produrre i risultati attesi. Un processo o sistema è efficace quando soddisfa i requisiti e le aspettative degli stakeholder, indipendentemente dalle risorse impiegate. Metrica: grado di raggiungimento degli obiettivi (interni ed esterni).

### **Efficienza**

Misura dell'abilità di raggiungere gli obiettivi impiegando le risorse minime indispensabili. Un processo o sistema è efficiente quando ottimizza l'uso di tempo, persone, denaro e strumenti. Metrica: produttività, ovvero rapporto tra quantità di output prodotto e risorse consumate.

### **Entity Resolution**

Processo di identificazione e associazione di entità (es. persone o aziende) a partire da dati parziali o duplicati, tramite algoritmi di matching e disambiguazione.

### **Error / Errore**

Manifestazione di un fault all'interno del sistema durante la sua esecuzione. È uno stato interno del sistema che può portare a un failure se non gestito. Corrisponde a una discrepanza tra il valore calcolato, osservato o misurato e il valore vero, specificato o teoricamente corretto.

## **Estimate at Completion (EAC)**

Stima del costo totale finale previsto per il completamento dell'intero progetto, calcolata sulla base delle performance attuali. Viene aggiornata periodicamente considerando i costi già sostenuti e le proiezioni future basate sull'andamento del CPI, permettendo di prevedere eventuali scostamenti dal budget iniziale.

## **Estimate to Complete (ETC)**

Stima del budget residuo necessario per completare le attività rimanenti del progetto. Rappresenta la differenza tra il costo totale stimato a finire (EAC) e i costi già sostenuti (AC), fornendo una previsione delle risorse economiche ancora necessarie.

## **F**

### **Front-end**

Parte di un'applicazione software con cui l'utente interagisce direttamente, responsabile della presentazione dei dati e dell'acquisizione dell'input dell'utente.

### **Funzionalità**

Caratteristica o capacità specifica che un sistema software deve possedere per soddisfare i bisogni degli utenti e gli obiettivi del progetto.

## G

### **GDPR (General Data Protection Regulation)**

Regolamento generale sulla protezione dei dati dell'Unione Europea che stabilisce norme per la protezione e la libera circolazione dei dati personali.

### **Gestione dei rischi**

Processo sistematico di identificazione, analisi, pianificazione e controllo dei rischi di progetto, volto a minimizzare la probabilità di occorrenza e l'impatto degli eventi negativi.

### **GitHub**

Piattaforma di hosting per repository Git che offre strumenti per il version control, la collaborazione e la gestione del ciclo di vita del software.

### **Glossario**

Documento che raccoglie e definisce i termini specifici, gli acronimi e le parole ambigue utilizzati nella documentazione di progetto, con lo scopo di garantire una comprensione univoca della terminologia da parte di tutti i membri del team e dei destinatari.

### **Google Meet**

Piattaforma di videoconferenza sviluppata da Google che consente meeting virtuali, condivisione dello schermo e chat. Utilizzata dal gruppo per le riunioni di coordinamento e le presentazioni.

## H

### **Human in the loop**

Approccio collaborativo uomo-macchina in cui un operatore umano interviene per validare, correggere o disambiguare i risultati dell'AI quando la confidenza è sotto una soglia prestabilita o in presenza di ambiguità (es. omonimie, multi-match nei destinatari). Il sistema segnala i casi critici e l'operatore conferma o corregge tipo documento, destinatari e metadati, creando un ciclo di feedback per il miglioramento continuo del modello.

## I

### Implementazione

Processo di traduzione di un design in codice eseguibile, realizzando le specifiche funzionali e non funzionali.

### Indice di Gulpease

Metrica di leggibilità specifica per la lingua italiana che valuta quanto un testo sia comprensibile. Calcolato considerando la lunghezza delle parole e delle frasi, produce un valore da 0 a 100: valori superiori a 80 indicano testi molto facili, 60-80 testi di media difficoltà, 40-60 testi difficili, sotto 40 testi molto complessi. Utilizzato per garantire che la documentazione sia accessibile a tutti.

### Interfaccia

In programmazione orientata agli oggetti, un'interfaccia è un tipo astratto che contiene una serie di metodi dichiarati senza implementazione. Le classi che implementano un'interfaccia devono fornire un'implementazione per tutti i suoi metodi. Le interfacce definiscono un contratto che le classi devono seguire.

### Issue

Segnalazione di un problema, un bug o una richiesta di miglioramento nel sistema di tracking del progetto. Ogni issue viene tracciata, assegnata e gestita fino alla risoluzione.

## K

### **KPI (Key Performance Indicator)**

Indicatore chiave di performance che misura l'efficacia di un processo, un'attività o un'organizzazione nel raggiungere i propri obiettivi.

## L

### [Latex](#)

Sistema di composizione tipografica utilizzato per la produzione di documentazione tecnica e scientifica di alta qualità, particolarmente adatto per documenti complessi con formule matematiche.

### [Lettera di candidatura](#)

Documento formale con cui un gruppo di studenti esprime ufficialmente il proprio interesse a realizzare un capitolato d'appalto specifico, presentando le motivazioni, le competenze del gruppo e una proposta preliminare di approccio al progetto. Viene indirizzata ai docenti e al proponente.

### [LLM \(Large Language Model\)](#)

Modello di linguaggio di grandi dimensioni addestrato su vasti corpus di testo, in grado di generare, comprendere e elaborare linguaggio naturale in modo sofisticato.

## M

### Manuale Utente

Documento tecnico che descrive le funzionalità del prodotto software dal punto di vista dell'utente finale, fornendo istruzioni dettagliate sull'installazione, la configurazione e l'utilizzo del sistema. Ha lo scopo di guidare l'utente nell'uso corretto del prodotto, illustrando le operazioni disponibili, i flussi di lavoro e le soluzioni ai problemi più comuni.

### Manutenibilità

Capacità di un prodotto software di essere modificato per correggere errori, migliorare le prestazioni o adattarsi a un ambiente in cambiamento. La manutenibilità è uno degli attributi di qualità del software e influenza il costo del ciclo di vita del prodotto.

### Microsoft Teams

Piattaforma di collaborazione sviluppata da Microsoft che offre chat, videoconferenze, archiviazione file e integrazione con applicazioni. Utilizzata dal gruppo per le riunioni di coordinamento e la condivisione di documenti.

### Milestone

Data di calendario che fissa un punto di avanzamento atteso nel tempo di progetto, definendo obiettivi specifici e misurabili da raggiungere. Il raggiungimento di una milestone è dimostrato dalla realizzazione di una baseline approvata. Le milestone devono essere SMART: Specifiche (obiettivi chiari), Misurabili (risultati verificabili), Achievable (realisticamente raggiungibili), Rilevanti (significative per il progetto e gli stakeholder), Time-bound (con scadenza definita). Vengono identificate procedendo a ritroso dalla milestone finale verso l'inizio del progetto, determinando le attività e le risorse necessarie per ogni periodo.

### Minimum Viable Product (MVP)

Versione minimale ma funzionante di un prodotto software che include solo le funzionalità essenziali necessarie per soddisfare i requisiti fondamentali e consentire una prima valutazione da parte del committente o degli utenti. L'MVP permette di verificare le ipotesi di base del progetto, raccogliere feedback reali e ridurre il rischio di sviluppare funzionalità non necessarie, seguendo un approccio iterativo e incrementale.

### Mitigazione

Insieme di azioni e strategie pianificate per ridurre la probabilità di occorrenza di un rischio o per minimizzarne l'impatto negativo sul progetto qualora si verifichi. La mitigazione rappresenta una fase fondamentale della gestione dei rischi e viene definita durante la pianificazione del progetto.

### Modularità

Grado in cui un sistema è composto da componenti separati che possono essere sviluppati, testati e mantenuti indipendentemente.

## Modulo

Componente software autonomo e sostituibile che incapsula una specifica funzionalità e fornisce un'interfaccia ben definita. I moduli favoriscono la modularità, il riuso e la manutenibilità del sistema.

## N

### **Norme di progetto**

Documento che definisce le regole, le convenzioni, le procedure e gli standard adottati dal gruppo per la realizzazione del progetto. Include norme per la redazione dei documenti, la gestione del versionamento, le convenzioni di codifica, le procedure di verifica e validazione, e le modalità di comunicazione interna ed esterna.

## O

### **OCR (Optical Character Recognition)**

Tecnologia che converte immagini di testo scritto o stampato in testo digitale machine-readable.

## P

### PB (Product Baseline)

Baseline che definisce lo stato del prodotto software in un determinato momento, tipicamente al termine di una fase di sviluppo o di uno sprint. Include le specifiche del prodotto, la documentazione e il codice sorgente, e serve come riferimento per le fasi successive o per il rilascio.

### Pericolosità

Nel contesto della gestione dei rischi, indica la gravità dell'impatto che un rischio può avere sul progetto qualora si verifichi. La pericolosità viene valutata considerando le conseguenze negative sugli obiettivi, sui tempi, sui costi e sulla qualità del progetto. Insieme alla probabilità di occorrenza, la pericolosità determina la priorità con cui un rischio deve essere gestito.

### Piano di Progetto

Documento di pianificazione che descrive come il progetto sarà condotto, gestito e controllato. Definisce gli obiettivi, le attività, le risorse, i tempi, i costi, i rischi e le modalità di monitoraggio e comunicazione del progetto.

### Piano di Qualifica

Documento che definisce le strategie, le risorse e le attività per garantire la qualità del processo di sviluppo e del prodotto software. Include la definizione degli standard di qualità, le metriche per la misurazione, i processi di verifica e validazione, e le responsabilità del team per il controllo della qualità.

### Planned Value (PV)

Valore pianificato del lavoro che dovrebbe essere stato completato in un dato momento secondo la pianificazione iniziale del progetto. Rappresenta il budget autorizzato allocato per le attività pianificate fino a una determinata data e serve come baseline per confrontare l'avanzamento effettivo.

### POC (Proof of Concept)

Realizzazione preliminare e limitata di un'idea o di un concetto, finalizzata a verificarne la fattibilità tecnica, l'adeguatezza rispetto ai requisiti o la validità di un'architettura proposta. Un POC viene solitamente sviluppato con risorse minime per valutare rapidamente se un approccio o una tecnologia siano promettenti prima di impegnare risorse significative in uno sviluppo completo.

### Postcondizioni

Condizioni che devono essere vere dopo l'esecuzione di un caso d'uso o di una funzionalità del sistema. Descrivono lo stato del sistema e dell'ambiente al termine del caso d'uso, inclusi gli effetti collaterali e i risultati prodotti.

## PostgreSQL

Sistema di gestione di database relazionale open-source e avanzato, noto per la sua affidabilità, robustezza e conformità agli standard SQL. È ampiamente utilizzato in applicazioni web e sistemi enterprise per la sua scalabilità e flessibilità.

## Precondizioni

Stato in cui devono trovarsi il sistema e l'ambiente circostante affinché un caso d'uso possa essere avviato con successo. Se le precondizioni non sono soddisfatte, il caso d'uso non viene eseguito.

## Preventivo

Documento di pianificazione che stima i costi e le risorse necessarie per lo svolgimento del progetto, basato sulle attività pianificate e sulle risorse disponibili.

## Priorità (Obbligatorio/Desiderabile/Opzionale)

Classificazione dei requisiti in base alla loro importanza per il progetto. I requisiti obbligatori devono essere soddisfatti affinché il sistema sia considerato accettabile, i desiderabili sono importanti ma non essenziali, e gli opzionali possono essere tralasciati senza compromettere le funzionalità fondamentali.

## Probabilità

Nel contesto della gestione dei rischi, rappresenta la possibilità che un evento rischioso si verifichi durante lo svolgimento del progetto. Viene generalmente espressa in termini qualitativi (bassa, media, alta) o quantitativi (percentuale). La valutazione della probabilità, insieme alla pericolosità, permette di classificare e prioritizzare i rischi da gestire.

## Prodotto (Product)

Nel contesto dei design pattern creazionali, il Product è l'oggetto complesso che viene costruito. In particolare, nel pattern Builder, il Product è l'oggetto finale che viene assemblato dal Director utilizzando il Builder.

## Progettista

Figura responsabile della progettazione dell'architettura software e delle soluzioni tecniche, garantendo che soddisfino i requisiti e siano realizzabili efficientemente.

## Progetto

Insieme di attività che devono raggiungere obiettivi specifici a partire da date specificate, con un inizio e una fine fissate in calendario, disponendo di risorse limitate (persone, tempo, denaro, strumenti) e consumando risorse nel loro svolgersi.

## Programmatore

Figura che implementa il codice sorgente secondo le specifiche tecniche, seguendo le best practice e gli standard di qualità definiti nel progetto.

## Prompt

Input (testo e/o altri dati) fornito a un modello di intelligenza artificiale per ottenere una risposta o un output specifico.

## Proponente

Azienda o organizzazione che propone un capitolato d'appalto per il progetto del corso di Ingegneria del Software, definendone requisiti e obiettivi.

## PWA (Progressive Web App)

Applicazione web che utilizza tecnologie web moderne per offrire un'esperienza simile a quella di un'app nativa, funzionando offline e potendo essere installata sul dispositivo.

## **Q**

### **Qualità**

Insieme delle caratteristiche di un prodotto software che gli conferiscono la capacità di soddisfare le esigenze espresse e implicite degli stakeholder. La qualità del software si misura in base a attributi interni (visibili agli sviluppatori) ed esterni (visibili agli utenti).

## R

### Rating System

Sistema di valutazione che assegna un punteggio o un giudizio a un'entità (es. prodotto, servizio, contenuto) in base a criteri prestabiliti.

### RBAC (Role-Based Access Control)

Modello di controllo degli accessi in cui i permessi sono assegnati a ruoli specifici, e gli utenti ottengono i permessi attraverso l'assegnazione a questi ruoli.

### Redattore

Membro del team responsabile della stesura e della produzione dei documenti di progetto, garantendo chiarezza, completezza e conformità alle norme stabilite.

### Repository

Archivio centrale in cui vengono memorizzati e versionati i file sorgente, la documentazione e le risorse del progetto utilizzando un sistema di controllo versione.

### Requirements Stability Index (RSI)

Indice che misura la stabilità dei requisiti durante il ciclo di vita del progetto, calcolato come rapporto tra i requisiti modificati/aggiunti/eliminati e il numero totale di requisiti. Un valore vicino a 1 indica alta stabilità, mentre valori bassi o negativi segnalano frequenti cambiamenti che possono impattare tempi e costi del progetto.

### Requisiti di qualità

Sottoinsieme dei requisiti non funzionali che specificano gli attributi di qualità del prodotto software, come manutenibilità, affidabilità, usabilità, efficienza, portabilità e sicurezza. Definiscono i criteri qualitativi che il sistema deve soddisfare per essere considerato accettabile dagli stakeholder.

### Requisiti di vincolo

Requisiti che impongono limitazioni o condizioni specifiche allo sviluppo del sistema, derivanti da fattori tecnologici, organizzativi, legali o di altro tipo. Esempi includono l'uso di specifiche tecnologie, conformità a standard, interfacce obbligate, vincoli temporali o di budget. A differenza dei requisiti funzionali e non funzionali, che definiscono cosa il sistema deve fare o come deve comportarsi, i requisiti di vincolo definiscono i limiti entro cui il sistema deve essere sviluppato.

### Requisiti funzionali

Requisiti che descrivono le funzionalità specifiche che il sistema deve essere in grado di eseguire. Definiscono il comportamento del sistema a fronte di specifici input, specificando cosa il sistema deve fare senza entrare nel merito di come viene implementato.

## Requisiti non funzionali

Requisiti che definiscono gli attributi di qualità, i vincoli e le proprietà sistemiche (come affidabilità, sicurezza, manutenibilità, efficienza). Non descrivono "cosa" fa il sistema, ma "come" deve comportarsi o quali standard deve rispettare.

## Requisiti prestazionali

Sottocategoria dei requisiti non funzionali focalizzata sulle metriche di efficienza del sistema. Definiscono vincoli quantitativi su tempi di risposta, throughput, latenza e utilizzo delle risorse hardware/software in condizioni di carico specificate.

## Requisito

Condizione o capacità che deve essere posseduta da un sistema o componente software per soddisfare un contratto, standard, specifica o altro documento formalmente imposto.

## Responsabile

Figura di riferimento del progetto con compiti di coordinamento, pianificazione, gestione delle risorse e comunicazione con docenti e proponenti.

## Retrospettiva

Cerimonia del framework Scrum che si tiene al termine di ogni sprint, durante la quale il team riflette sul processo di lavoro adottato, identifica ciò che ha funzionato bene e ciò che può essere migliorato, e definisce un piano di azione per incrementare l'efficacia e la qualità del lavoro negli sprint successivi.

## Rischio

Evento o condizione incerta che, qualora si verificasse, potrebbe avere un impatto positivo o negativo sugli obiettivi del progetto. Nella gestione di progetto, i rischi vengono identificati, analizzati, classificati e monitorati attraverso un processo sistematico di gestione dei rischi. Ogni rischio è caratterizzato da una probabilità di occorrenza e da una pericolosità (impatto potenziale).

### Rischio Globale (RG)

Categoria di rischi che riguardano l'intero team o il progetto nel suo complesso, piuttosto che singoli individui o aspetti tecnici specifici. Esempi includono disaccordi significativi tra membri del team, malcomprensioni dei requisiti del capitolato, sottostima o sovrastima delle attività pianificate. Questi rischi possono influenzare l'andamento generale del progetto e richiedono strategie di mitigazione a livello di gruppo.

### Rischio Individuale (RI)

Categoria di rischi che riguardano la disponibilità e le capacità dei singoli membri del team. Esempi includono impegni derivanti da altre attività universitarie (esami, progetti paralleli) o indisponibilità personali impreviste (problemi di salute, emergenze familiari). Questi rischi

vengono mitigati attraverso la comunicazione tempestiva e la redistribuzione del carico di lavoro tra i membri disponibili.

### Rischio Tecnologico (RT)

Categoria di rischi legati agli aspetti tecnici del progetto, come l'utilizzo di tecnologie, linguaggi o strumenti non completamente conosciuti dal team, oppure la presenza di errori e bug nel codice sviluppato. Questi rischi richiedono strategie di mitigazione che includono formazione, supporto tecnico interno al team e allocazione di tempo per la risoluzione di problemi tecnici.

### RTB (Requirements and Technology Baseline)

Baseline che combina i requisiti e le tecnologie approvate per il progetto. Definisce l'insieme dei requisiti concordati e formalmente approvati e lo stack tecnologico di riferimento, costituendo la base per lo sviluppo del progetto. Ogni modifica a questa baseline deve seguire un processo formale di controllo delle modifiche.

### Ruby

Linguaggio di programmazione dinamico, open source, orientato agli oggetti e interpretato, noto per la sua semplicità e produttività.

### Ruby on Rails

Framework per applicazioni web scritto in Ruby, che segue il pattern Model-View-Controller (MVC) e i principi di convenzione sopra configurazione (convention over configuration).

## S

### Sandbox

Ambiente isolato per testare e validare funzionalità senza influenzare i sistemi di produzione, utilizzato per sviluppare e verificare nuove feature in sicurezza.

### Scalabilità

Capacità di un sistema di gestire un aumento del carico di lavoro aggiungendo risorse, senza modifiche all'architettura.

### Scenario

Sequenza di interazioni tra attori e sistema che descrive un percorso specifico attraverso un caso d'uso. Può essere principale (percorso di successo) o alternativo (variazioni ed eccezioni).

### Scenario principale

Sequenza di interazioni tra l'attore principale e il sistema che descrive il percorso di successo del caso d'uso, dove l'obiettivo viene raggiunto senza intoppi o condizioni eccezionali. Rappresenta il flusso ideale e più frequente di esecuzione.

### Scenario secondario

Variante di uno scenario che descrive un percorso alternativo o un'eccezione nel caso d'uso, che può verificarsi in condizioni specifiche e che porta a un risultato diverso (ad esempio, un errore o un'alternativa valida).

### Schedule Performance Index (SPI)

Indice di efficienza temporale che misura il rapporto tra il valore del lavoro completato (EV) e il valore pianificato (PV). Calcolato come  $SPI = EV/PV$ , un valore superiore a 1 indica che il progetto è in anticipo sulla pianificazione, mentre un valore inferiore a 1 segnala un ritardo.

### Scope

Ambito di un progetto, che definisce i confini, i deliverables, gli obiettivi e i compiti che sono inclusi nel progetto.

### SCRUM

Framework agile per la gestione dello sviluppo software che enfatizza lo sviluppo iterativo, l'adattamento ai cambiamenti e la consegna incrementale di valore.

### Sicurezza

Insieme di caratteristiche e meccanismi che proteggono il sistema da accessi non autorizzati, garantiscono la riservatezza, l'integrità e la disponibilità dei dati, e prevengono attacchi e vulnerabilità.

## Specifica Tecnica

Documento che descrive l'architettura del sistema software e le scelte progettuali adottate per soddisfare i requisiti definiti nell'Analisi dei Requisiti. Definisce la struttura dei componenti, le loro interazioni, i pattern architetturali utilizzati, le tecnologie impiegate e i criteri di qualità perseguiti. Costituisce il riferimento tecnico per le attività di codifica e verifica, e rappresenta il principale deliverable della fase di progettazione (Product Baseline) nel ciclo di vita del progetto.

## Sprint

Periodo di tempo fisso (tipicamente 2-4 settimane) in Scrum durante il quale il team sviluppa e consegna un incremento di prodotto potenzialmente rilasciabile.

## Stakeholder

Portatore di interesse, ovvero qualsiasi individuo, gruppo o organizzazione che può influenzare o essere influenzato da un progetto, un'azienda o un sistema.

## Stand-alone

Nel contesto del progetto sviluppato dal team BugBusters, si riferisce a un'applicazione o modulo software sviluppato in modo autonomo e indipendente dall'infrastruttura NEXUM esistente. I moduli AI Assistant Generativo e AI Co-Pilot per i CdL saranno realizzati come applicazioni stand-alone che simulano i comportamenti e le interfacce di NEXUM dove necessario, con l'obiettivo di poter essere integrati successivamente nella piattaforma principale. Questo approccio consente uno sviluppo più flessibile e isolato, pur mantenendo la compatibilità con l'architettura e le API di NEXUM.

## Stima dei costi

Processo di previsione dei costi associati alle attività di progetto, considerando risorse umane, strumenti, infrastrutture e altri fattori che influenzano il budget complessivo.

## T

### Technology Baseline

Insieme delle tecnologie, framework, librerie e strumenti di sviluppo selezionati e approvati per il progetto. Definisce lo stack tecnologico di riferimento e costituisce la base per le scelte implementative, garantendo coerenza e standardizzazione nell'architettura software.

### Template

Struttura o modello predefinito che definisce il layout e l'aspetto di un'interfaccia utente, separando la presentazione dalla logica di business.

### Test

Processo sistematico di verifica che il software soddisfi i requisiti specificati e identifichi difetti, attraverso l'esecuzione controllata di casi di test.

### Tracciabilità

Capacità di ricostruire la storia completa di un documento o processo attraverso registrazioni dettagliate (audit trail), dal caricamento all'invio alla lettura, garantendo conformità e accountability.

## V

### Validazione

Processo che accerta che il prodotto software sviluppato soddisfi le effettive esigenze del cliente e gli obiettivi di business per i quali è stato realizzato. Risponde alla domanda "Stiamo costruendo il prodotto giusto?" e viene tipicamente effettuata attraverso test di accettazione con il cliente.

### Verbale

Documento formale che registra quanto discusso, deciso e pianificato durante una riunione. Include informazioni come data, partecipanti, ordine del giorno, punti discussi, decisioni prese e azioni da intraprendere.

### Verbale esterno

Verbale redatto in seguito a una riunione con soggetti esterni al gruppo di progetto, come il proponente o i docenti. Oltre ai contenuti standard di un verbale, può includere specifiche decisioni o accordi presi con le parti esterne.

### Verbale interno

Verbale redatto in seguito a una riunione interna al gruppo di progetto. Documenta le discussioni e le decisioni prese dai membri del team, e viene utilizzato per tracciare lo stato di avanzamento, assegnare compiti e coordinare le attività.

### Verifica

Processo sistematico che determina se i prodotti di lavoro (documenti, codice, componenti) soddisfano i requisiti e le specifiche definite per loro. Risponde alla domanda "Stiamo costruendo il prodotto nel modo giusto?" e include attività come revisioni, ispezioni e test.

### Verificatore

Membro del team responsabile di controllare che documenti, codice e altri prodotti di lavoro rispettano gli standard di qualità definiti, le norme di progetto e siano privi di errori, incoerenze o ambiguità.

### Versionamento/versioning

Sistema per gestire le diverse versioni di file, codice sorgente o documenti, permettendo di tracciare le modifiche, collaborare in team e revertire a versioni precedenti.

## W

### Way of Working

Insieme di processi, metodologie, strumenti e pratiche adottati dal team per organizzare e svolgere le attività di progetto in modo coordinato ed efficiente. Definisce come il team collabora, comunica e gestisce il lavoro quotidiano.

### Workflow

Sequenza di attività che definiscono un processo di business, dove compiti, informazioni o documenti passano da un partecipante all'altro secondo regole prestabilite.