



**Università degli Studi di Padova**  
Laurea: Informatica  
Corso: Ingegneria del Software  
Anno Accademico: 2025/2026

**Gruppo: NullPointers Group**  
Email: nullpointersg@gmail.com

# Glossario

Stato	In approvazione
Versione	0.7.0
Data ultima modifica	23/01/2026
Redattori	M. Mazzaretto T. Ceron L. Pieripolli L. Casagrande M. Brunello L. Marcuzzo
Verificatori	M. Brunello L. Marcuzzo L. Casagrande M. Mazzaretto L. Pieripolli
Destinatari	NullPointers Group

## Registro delle modifiche

Vers	Data	Autore	Verificatore	Descrizione
0.7.0	23-01	L. Marcuzzo	M. Brunello	Aggiunta nuove voci
0.6.0	20-12	T. Ceron	L. Pieripolli	Aggiunta nuove voci
0.5.2	07-12	M. Brunello	L. Casagrande	Correzioni minori
0.5.1	05-12	M. Brunello	L. Casagrande	Correzioni minori
0.5.0	05-12	M. Brunello	L. Casagrande	Aggiunta nuove voci
0.4.0	01-12	T. Ceron	L. Casagrande	Fix e aggiunta nuove voci
0.3.1	29-11	L. Casagrande	M. Mazzaretto	Creazione macro per virgolette
0.3.0	27-11	M. Mazzaretto	L. Casagrande	Aggiunta di nuove voci
0.2.1	24-11	L. Pieripolli	L. Marcuzzo	Correzioni
0.2.0	21-11	L. Pieripolli	L. Marcuzzo	Aggiunta di nuove voci
0.1.1	10-11	M. Mazzaretto	M. Brunello	Correzioni estetiche
0.1.0	7-11	T. Ceron	M. Brunello	Creazione e stesura documento

## Indice

<b>Introduzione</b> . . . . .	<b>5</b>
<b>A</b> . . . . .	<b>6</b>
<b>B</b> . . . . .	<b>8</b>
<b>C</b> . . . . .	<b>10</b>
<b>D</b> . . . . .	<b>11</b>
<b>E</b> . . . . .	<b>13</b>
<b>F</b> . . . . .	<b>14</b>
<b>G</b> . . . . .	<b>15</b>
<b>H</b> . . . . .	<b>16</b>
<b>I</b> . . . . .	<b>17</b>
<b>J</b> . . . . .	<b>18</b>
<b>K</b> . . . . .	<b>19</b>
<b>L</b> . . . . .	<b>20</b>
<b>M</b> . . . . .	<b>21</b>
<b>N</b> . . . . .	<b>23</b>
<b>O</b> . . . . .	<b>24</b>
<b>P</b> . . . . .	<b>25</b>
<b>Q</b> . . . . .	<b>28</b>
<b>R</b> . . . . .	<b>29</b>
<b>S</b> . . . . .	<b>31</b>
<b>T</b> . . . . .	<b>32</b>
<b>U</b> . . . . .	<b>33</b>
<b>V</b> . . . . .	<b>34</b>
<b>W</b> . . . . .	<b>35</b>

<b>X</b>	.....	<b>36</b>
<b>Y</b>	.....	<b>37</b>
<b>Z</b>	.....	<b>38</b>

## Introduzione

Il documento definisce in modo univoco i termini adottati all'interno del progetto, così da evitare interpretazioni personali o variazioni di significato.

Le definizioni sono presentate in ordine alfabetico per rendere la consultazione diretta e stabile nel tempo.

Quando un termine compare nei documenti con un significato vincolato dal Glossario, viene segnalato con la notazione:

termine<sup>G</sup>

**A****AC**

Acronimo di *Actual Cost*, per la definizione vedere ‘Actual Cost’.

**Actual Cost**

È una metrica che rappresenta il costo in euro sostenuto fino al momento corrente per svolgere le ore produttive rendicontate.

**Agile**

In informatica, Agile è un approccio di lavoro basato su iterazioni brevi e frequenti feedback. Nel metodo Scrum, uno dei più diffusi, il lavoro viene suddiviso in Sprint di una o due settimane. Ogni Sprint inizia con una Sprint Planning, in cui si definiscono gli obiettivi, e si conclude con una Sprint Retrospective, in cui il team valuta cosa è andato bene e cosa migliorare. Durante lo Sprint si svolgono riunioni giornaliere (Daily Scrum) per monitorare l'avanzamento.

**Amministratore**

L’Amministratore si occupa della configurazione e gestione dell’infrastruttura IT di supporto ad un progetto.

**Analisi dei Requisiti**

Documento contenente i Casi d’Uso identificati da un Team per i requisiti di un determinato progetto. Contiene anche i Requisiti funzionali, Requisiti di Qualità e Requisiti di Vincolo.

**Analista**

Si occupa di identificare e chiarire i requisiti, interpretando le esigenze degli utilizzatori finali per garantire una corretta definizione delle funzionalità.

**API**

Acronimo di *Application Programming Interface*, è un'interfaccia utile a permettere o facilitare la comunicazione tra diversi software o parti di un singolo software.

### **Attore**

Un'entità esterna e non controllabile dal progetto, ma che interagisce con esso, con attività o obiettivi specifici da soddisfare.

### **Automazione**

Esecuzione deterministica e ripetibile di processi (build, test, deploy, verifiche, provisioning) attivati da eventi del repository o da pianificazioni, definita tramite workflow dichiarativi in YAML, eseguita in ambienti isolati, senza intervento manuale.

### **Azione**

L'insieme di processi intrapresi per mettere in pratica una determinata Decisione.

**B****BAC**

Acronimo di *Budget at Completion*, per la definizione vedere ‘Budget at Completion’.

**Backend**

La parte di un progetto che si occupa del funzionamento a livello logico del sistema.

**Baseline**

Stato fissato di artefatti (requisiti, design, codice, documentazione) considerato stabile e sufficientemente maturo, dal quale retrocedere comporterebbe costo o rischio elevato.

**BERT**

Acronimo di *Bidirectional Encoder Representations from Transformers*, modello di linguaggio basato sull’architettura Transformer utilizzato per l’elaborazione del linguaggio naturale. BERT consente di comprendere il contesto delle parole in modo bidirezionale ed è impiegato per attività quali analisi semantica, classificazione del testo ed estrazione di informazioni.

**Branch**

Uno specifico ramo di lavoro all’interno di una repository, atta a mantenere una distinzione tra ramo “principale” e rami di lavoro in cui ciascun membro del gruppo può lavorare in maniera asincrona senza rischio di conflitto.

**Budget at Completion**

È il budget complessivo preventivato all’inizio di un progetto, che rappresenta il costo totale stimato per completare tutte le attività e consegnare il prodotto finale.

**Bug**

Difetto o errore nel codice di un programma che provoca comportamenti imprevisti o malfunzionamenti. Può derivare da errori di logica, sintassi, progettazione o interazioni inattese tra componenti software.

## Build

Processo con cui il codice sorgente, insieme alle sue dipendenze, viene trasformato in un artefatto eseguibile o distribuibile (ad esempio un binario, una libreria o un pacchetto).

## C

### **Capitolato**

Documento fornito dalla proponente, che espone le principali specifiche e condizioni di un progetto. Stabilisce delle regole e delle aspettative che il fornitore deve rispettare.

### **Caso d'uso**

Insieme di azioni all'interno di uno scenario, con in comune uno scopo finale per un Attore.

### **Code Smell**

Caratteristica negativa di un codice sorgente, sono di diverso tipo e possono indicare difetti di design o di implementazione, diminuendo la qualità del software e aumentandone il debito tecnico.

### **Committente**

Persona, ente o azienda che richiede e finanzia la realizzazione di un progetto software, di un sistema informatico o di un servizio IT. Il committente definisce gli obiettivi, approva i requisiti e verifica i risultati, ma non esegue direttamente lo sviluppo.

### **Consuntivo**

Rendiconto redatto dal gruppo al termine di uno sprint che riporta le ore effettivamente impiegate dai membri del gruppo, permettendo il calcolo del costo.

### **Continuous Integration**

Pratica di sviluppo del software che prevede che gli sviluppatori effettuino frequentemente commit verso una repository centrale, alla quale possono seguire test automatizzati.

### **Cyclomatic Complexity**

Una metrica utilizzata per misurare la complessità di un programma; nello specifico misura il numero di cammini linearmente indipendenti di un metodo, una classe o un progetto intero.

## D

### Database

Un database (in italiano: base di dati) è un sistema per la conservazione e gestione dei dati. Comprende una componente fisica, che è il supporto di memorizzazione dei dati, e una componente software, detta DBMS (Database Management System), che gestisce l'accesso ai dati, ne garantisce la persistenza e fornisce un metodo sicuro ed efficiente per leggerli e modificarli.

### Debug

Processo di individuazione, analisi e correzione degli errori presenti nel codice di un programma.

### Decisione

Un accordo all'interno del Team, riguardante metodologie o convenzioni da adottare per soddisfare un determinato bisogno.

### Discord

Piattaforma di videochiamate e messaggistica asincrona. Caratterizzata dalla possibilità di avere spazi indipendenti: server; dove si possono creare vari canali testuali indipendenti, solitamente suddivisi per argomenti, così come canali vocali indipendenti, utilizzati per chiamate contemporanee tra i vari componenti del server.

### Distribuzione

Processo finale dello sviluppo software in cui il prodotto viene reso disponibile in una versione stabile per l'utilizzo da parte degli utenti.

### Docker

Software che permette di eseguire applicazioni in ambienti isolati chiamati container.

### Doxxygen

Software che permette la generazione automatica di documentazione a partire dal codice sorgente di un software.

### **Draw.io**

Software utilizzato per la creazione di diagrammi UML.

**E****EAC**

Acronimo di *Estimate at Completion*, per la definizione vedere ‘Estimate at Completion’.

**Earned Value**

È una metrica che rappresenta il valore del lavoro completato rispetto al budget totale previsto.

**EasyOCR**

Libreria python di tipo OCR per il riconoscimento di testo da immagini.

**Efficacia**

Indica il modo con cui un insieme di attività raggiunge gli obiettivi attesi.

**Efficienza**

Rapporto tra risultati ottenuti e risorse impiegate per ottenerli.

**ERP**

ERP (Enterprise Resource Planning) è un sistema software integrato che gestisce e coordina i processi aziendali fondamentali come contabilità, produzione, vendite, risorse umane e inventario, centralizzando i dati per migliorare efficienza e decisioni.

**Estimate at Completion**

È una metrica che rappresenta la proiezione del costo finale del progetto basata sulla performance attuale.

**EV**

Acronimo di *Earned Value*, per la definizione vedere ‘Earned Value’.

**F****Framework**

Insieme strutturato di componenti software, librerie e regole che forniscono una base riutilizzabile per sviluppare applicazioni. Un framework offre funzionalità già pronte e un'architettura predefinita, guidando lo sviluppatore nelle scelte progettuali e riducendo il lavoro necessario per creare nuovi programmi.

**Frontend**

La parte di un progetto che si occupa dell'interazione diretta con l'utilizzatore finale.

**Funzionalità**

Una caratteristica offerta da un Sistema, spesso con lo scopo di soddisfare un bisogno.

## G

### Git

Sistema di controllo di versione che traccia le modifiche al codice sorgente nel tempo. Permette a più sviluppatori di collaborare simultaneamente sullo stesso progetto, gestendo la cronologia delle versioni e supportando lo sviluppo parallelo.

### GitFlow

Modello che sfrutta i branch di Git definendo una struttura organizzata per la gestione dei rami di sviluppo; facilita il lavoro parallelo e consente la netta separazione della versione in produzione.

### GitHub

Servizio cloud che permette di gestire una o più repository Git, offre strumenti che semplificano la collaborazione e l'automazione di processi.

### GitHub Action

Strumento messo a disposizione da GitHub per effettuare automazioni all'interno di una repository.

### Google Meet

Piattaforma di videocomunicazione sviluppata da Google, utilizzata dal gruppo NullPointers Group per le riunioni online con l'azienda.

### Google Vision

Servizio di intelligenza artificiale sviluppato da Google che permette di analizzare e comprendere il contenuto delle immagini attraverso modelli di machine learning.

### GPT

Modello linguistico sviluppato da OpenAI e basato su reti neurali, specializzato nella comprensione e generazione del linguaggio naturale. È progettato per fungere da struttura base su cui costruire applicazioni software e sistemi di intelligenza artificiale.

## H

**I****IA**

Acronimo di *Intelligenza Artificiale*, per la definizione vedere ‘Intelligenza Artificiale’.

**IEEE**

È l’acronimo di *Institute of Electrical and Electronics Engineers*, un’associazione internazionale che promuove lo sviluppo delle tecnologie e dell’ingegneria in ambiti come l’elettronica, informatica e le telecomunicazioni.

**Intelligenza Artificiale**

Disciplina dell’informatica che studia e sviluppa sistemi in grado di eseguire compiti che richiedono tipicamente l’intelligenza umana, come il ragionamento, l’apprendimento e la comprensione del linguaggio naturale.

**ISO**

Ente di standardizzazione internazionale che definisce norme tecniche condive per garantire interoperabilità, qualità e uniformità tra prodotti, processi e sistemi.

**Issue**

All’interno dell’ecosistema GitHub, rappresenta un’attività che richiede modifiche al codice sorgente per essere svolta, può trattarsi di un bug da risolvere, una feature da aggiungere, di una correzione minore o anche di qualcosa definito dagli sviluppatori.

**J****JSON**

JSON (JavaScript Object Notation) è un formato leggero per la rappresentazione di dati strutturati, leggibile da esseri umani e facilmente parsabile, ovvero facilmente analizzato o interpretato da un parser.

## K

### KLOC

Acronimo di *Kilo Lines of Code*, le linee di codice totali che compongono un progetto, nell'ordine delle migliaia.

**L****LaTeX**

Linguaggio di markup utilizzato dal gruppo per produrre tutti i documenti.

**Licenza MIT**

Una licenza Open Source che permette di utilizzare, modificare e distribuire liberamente il software a patto che la licenza venga distribuita insieme al codice originale.

**LlaMa**

Famiglia di modelli linguistici sviluppata da Meta, progettata per essere efficiente e disponibile anche offline. Consente l'esecuzione locale, senza la necessità di appoggiarsi a servizi cloud.

**LLM**

Large Language Models. Indica i modelli addestrati su enormi quantità di testo come ChatGPT e DeepSeek.

**LOC**

Acronimo di *Lines of Code*, le linee di codice totali che compongono un progetto.

**Lua**

Un linguaggio di programmazione dinamico e procedurale, utilizzato come linguaggio di scripting per automatizzare vari processi.

## M

### Machine Learning

Una branca dell'intelligenza artificiale che permette ad un computer di svolgere un compito apprendendo dai dati forniti e migliorando le sue prestazioni nel tempo tramite più dati ed esperienze.

### Major

Una modifica atta ad approvare un documento in una sua versione “pronta al rilascio”. In un sistema di versionamento x.y.z, la Major è rappresentata dalla lettera x, e ogni modifica di tipo Major aggiorna un file dalla versione x.y.z alla versione (x+1).y.z.

### Manutenzione

Insieme di attività atte a risolvere i problemi e migliorare le funzionalità di un prodotto.

### Merge

Processo con il quale, in un sistema di versionamento come git, un ramo di lavoro viene fatto confluire in un altro, generalmente in un grado gerarchico più elevato.

### Metodo Agile

Metodo di sviluppo software per la gestione di progetti complessi basato sui cicli iterativi brevi chiamati *Sprint*, che favorisce la flessibilità del progetto e la collaborazione continua.

### Middleware

Strato software intermedio che si colloca tra applicazioni o componenti di un sistema, con lo scopo di facilitarne la comunicazione e l'integrazione. Il middleware fornisce servizi comuni quali la gestione delle comunicazioni, lo scambio di dati, la sicurezza, la gestione delle transazioni e l'astrazione delle risorse sottostanti.

### Milestone

Periodo di tempo entro il quale il team si pone il raggiungimento di determinati obiettivi.

### **Minimum Viable Product**

La prima versione di un prodotto che assolve le funzionalità minime per rispettare i requisiti obbligatori stabiliti dalla proponente.

### **Minor**

Una modifica atta ad aggiungere migliorie o funzionalità significative. In un sistema di versionamento x.y.z, la Minor è rappresentata dalla lettera y, e ogni modifica di tipo Minor aggiorna un file dalla versione x.y.z alla versione x.(y+1).0.

### **ML**

Acronimo di *Machine Learning*, per la definizione vedere ‘Machine Learning’.

### **Modifica**

Qualsiasi intervento sul codice esistente, sulla documentazione o sul metodo di lavoro volto a correggere degli errori, aggiornare i contenuti o adattarli ai nuovi requisiti.

### **MVP**

Acronimo di *Minimum Viable Product*, per la definizione vedere ‘Minimum Viable Product’.

**N****Natural Language Processing**

È il ramo dell'informatica e dell'intelligenza artificiale che sviluppa metodi e algoritmi per far comprendere, generare e interagire i computer con il linguaggio umano.

**NLP**

Acronimo di *Natural Language Processing*, per la definizione vedere ‘Natural Language Processing’.

**Norme di Progetto**

Documento che descrive il Way of Working del gruppo. Contiene le regole che NullPointers Group rispetta per garantire un lavoro efficace ed efficiente.

# O

## OCR

Processo di riconoscimento ottico di caratteri che converte un'immagine o un documento scannerizzato in un formato leggibile dalla macchina.

## ODBC

Acronimo di *Open Database Connectivity*, standard che permette l'accesso a basi di dati tramite un'interfaccia uniforme, indipendente dal sistema di gestione utilizzato. Attraverso driver dedicati, ODBC consente la comunicazione tra applicazioni e database, favorendo la separazione dei componenti e la portabilità del software.

## OpenAI

Azienda di ricerca e sviluppo specializzata nell'intelligenza artificiale, nota per la creazione dei modelli GPT. Fornisce inoltre API e servizi per l'integrazione di queste tecnologie all'interno delle applicazioni software.

**P****Patch**

Una modifica atta a risolvere errori ortografici o di struttura. In un sistema di versionamento x.y.z, la patch è rappresentata dalla lettera z, e ogni modifica di tipo patch aggiorna un file dalla versione x.y.z alla versione x.y.(z+1).

**PB**

Acronimo di *Product Baseline*, per la definizione vedere ‘Product Baseline’.

**Pianificazione**

Processo di definizione degli obiettivi del progetto identificandone le risorse ed i tempi richiesti per la sua realizzazione.

**Piano di Progetto**

Documento che si occupa di definire l’organizzazione del gruppo, specificando l’assegnazione dei ruoli, il monte ore previsto per ogni risorsa, le tempistiche e il conto economico preventivo in funzione dell’avanzamento del progetto.

**Piano di Qualifica**

Documento che si occupa di garantire il rispetto dei requisiti del progetto definendo le strategie di verifica e validazione affinché vengano rispettate le attese della proponente.

**Planned Value**

È la metrica che rappresenta il valore del lavoro pianificato rispetto al budget preventivato previsto.

**PoC**

Acronimo di *Proof of Concept*, per la definizione vedere ‘Proof of Concept’.

**POC**

Codice stabilito dal Team, rappresenta un'attività con corrispettiva issue in GitHub all'interno della repository dedicata al *Proof of Concept*. Il codice è seguito da un numero incrementale che permette di distinguere univocamente una singola attività.

## **PostgreSQL**

Sistema di gestione di database relazionali open source, noto per la sua scalabilità e la conformità agli standard SQL.

## **PR**

Acronimo di *Pull Request*, per la definizione vedere ‘Pull Request’.

## **Processo**

Insieme di attività correlate e coese che trasformano ingressi (bisogni) in uscite (prodotti) secondo regole date, consumando risorse nel farlo.

## **Product Baseline**

Fase finale in un progetto, che prevede la realizzazione di un Minimum Viable Product, a seguito del superamento dei test concordati nella Requirements and Technology Baseline.

## **Produzione**

Processo di traduzione della progettazione in codice sorgente mediante la programmazione ed i test.

## **Progettazione**

Processo che definisce la struttura di un sistema, identificandone componenti, relazioni e livello di dettaglio necessari per la realizzazione. Serve a semplificarne lo sviluppo e la manutenzione futura.

## **Progettista**

Ruolo dedito alla progettazione dell'architettura di un progetto, al fine di assicurarsi codice più corretto e manutenibile.

**Progetto**

Insieme ordinato di attività, istanziate dai processi di ciclo di vita che soddisfano gli obiettivi dati.

**Programmatore**

La persona incaricata di svolgere le attività di codifica, data l'architettura.

**Proof of Concept**

Prodotto dimostrativo atto a presentare la fattibilità dell'idea.

**Proponente**

La persona, il gruppo o l'organizzazione che propone un progetto, un'idea o un'iniziativa, e ne definisce obiettivi, requisiti e motivazioni. La proponente di NullPointers Group è Ergon s.r.l.

**Pull Request**

All'interno dell'ecosistema GitHub, uno strumento che permette di sospendere momentaneamente la pubblicazione di una modifica nel ramo principale della repository, in attesa di approvazione e/o workflow automatici (e.g. GitHub Action).

**PV**

Acronimo di *Planned Value*, per la definizione vedere ‘Planned Value’.

**Python**

Linguaggio di programmazione di alto livello, open source, noto per la sua sintassi chiara; è ampiamente utilizzato nel campo dell'intelligenza artificiale e del machine learning.

## Q

## R

### **React**

Framework open source sviluppato da Meta, basato su JavaScript, utilizzato per costruire interfacce utente dinamiche nelle applicazioni web front-end.

### **Redattore**

Colui che scrive il documento, viene identificato come “Autore” nel registro delle modifiche.

### **Repository**

Uno spazio condiviso tra tutti i membri del Team, in cui viene memorizzato e versionato tutto il codice sorgente prodotto.

### **Requisito di vincolo**

Tipo di requisito che descrive le restrizioni imposte dall’azienda proponente che il progetto deve rispettare.

### **Requisito funzionale**

Definiscono cosa il sistema deve fare, descrivendo le sue azioni e funzioni specifiche. Sono le capacità che l’utente si aspetta, come “aggiungere un prodotto al carrello” in un negozio online.

### **Requisito non funzionale**

Specificano come il sistema deve comportarsi mentre svolge quelle funzioni. Si concentrano su qualità come le prestazioni, la sicurezza o l’affidabilità, ad esempio “il sistema deve supportare 1000 utenti contemporanei senza rallentamenti”.

### **Responsabile**

Il Responsabile coordina le attività del gruppo e garantisce una pianificazione efficace.

## REST

REST (Representational State Transfer) è uno stile architetturale per lo sviluppo di servizi web che utilizza protocolli standard (tipicamente HTTP) per operazioni su risorse identificate tramite URL, basandosi su metodi come GET, POST, PUT, DELETE.

## RTB

Baseline iniziale e vincolante di requisiti e tecnologie approvate per il progetto. Comprende requisiti funzionali, non funzionali, di sicurezza e conformità, insieme a strumenti, linguaggi, piattaforme e infrastrutture scelte. Costituisce il riferimento stabile per verificare coerenza e avanzamento. Qualsiasi modifica richiede revisione e approvazione formale.

**S****SM**

Codice stabilito dal Team, rappresenta un'attività, con corrispettiva issue in GitHub all'interno della Repository ‘SmartOrder’, indica dunque un'attività da svolgere relativa al progetto e al *PoC*. Il codice è seguito da un numero incrementale che permette di distinguere univocamente una singola attività.

**SMD**

Codice stabilito dal Team, rappresenta un'attività, con corrispettiva issue in GitHub all'interno della Repository ‘Documentazione’, indica dunque un'attività da svolgere relativa alla documentazione di progetto. Il codice è seguito da un numero incrementale che permette di distinguere univocamente una singola attività.

**Sprint**

Ciclo di lavoro di durata fissa, tipico dello sviluppo Agile, durante il quale il team si impegna a completare un insieme prefissato di attività per produrre un incremento del software.

**SQL**

Acronimo di *Structured Query Language*, è un linguaggio di programmazione con lo scopo di creare, filtrare ed eliminare dati memorizzati all'intero di tabelle nei database relazionali.

**Stakeholder**

Qualunque persona, gruppo o organizzazione che ha interesse, influenza o è coinvolta nel progetto, includendo clienti, utenti, sviluppatori, dirigenti, fornitori e persino organizzazioni esterne, ed è fondamentale per definire requisiti.

**Sviluppo**

Attività pratica di scrittura, test e integrazione del codice necessaria a implementare le funzionalità definite durante la fase di progettazione.

**T****Task**

Un'attività ben definita, assegnata ad uno o più membri del gruppo, che contribuisce allo sviluppo del progetto. Una task può essere parte di un'attività più grande o un obiettivo specifico da raggiungere.

**TD**

Acronimo di *To Do*, indica un'attività futura decisa in un incontro interno/esterno, presenta uno o più assegnatari. Il codice è seguito da un numero incrementale che permette di distinguere univocamente una singola attività.

**Test**

Un processo dello sviluppo software che si occupa di verificare la funzionalità di una o più parti del prodotto.

**Throttling**

Tecnica di controllo del flusso che consiste nel limitare intenzionalmente il numero di richieste, operazioni o risorse utilizzabili in un determinato intervallo di tempo, al fine di garantire la stabilità del sistema, prevenire sovraccarichi e assicurare un utilizzo equo delle risorse disponibili.

**Token**

Unità fondamentale di testo (come una parola, un carattere o un frammento di parola) in cui viene suddiviso l'input per essere elaborato dai modelli di linguaggio.

**Tokenizzazione**

Processo di segmentazione del testo in unità elementari chiamate *Token*, passaggio necessario agli LLM per l'interpretazione del linguaggio naturale.

**U****UML**

Acronimo di *Unified Modeling Language*, si tratta di un linguaggio di modellazione standard. Utilizzato per la visualizzazione di sistemi complessi tramite diagrammi, per esempio: diagrammi di classi, casi d'uso, diagrammi delle attività, etc.

**V****Validazione**

Insieme di controlli per assicurarsi che un prodotto (o una parte di esso) soddisfi in modo congruo ciò per cui è stato creato.

**VE**

Acronimo di *Verbale Esterno*, viene utilizzato per tracciare nel tempo decisioni e azioni intraprese in un incontro esterno. Il codice associato ai VE segue la forma VE x.y, dove x identifica il numero progressivo del verbale e y identifica una specifica decisione o voce registrata all'interno del verbale.

**Verifica**

Insieme di processi atti a garantire che una funzione all'interno di un prodotto soddisfi determinati requisiti.

**Verificatore**

Il Verificatore si occupa di assicurare la qualità dei prodotti e dei processi adottati, effettuando revisioni e test.

**Versionamento**

Metodo per gestire e tenere traccia di documenti, codice o altri artefatti digitali, organizzandoli in versioni identificate da un codice univoco, in modo da poter monitorare modifiche, confrontare stati diversi e ripristinare versioni precedenti.

**VI**

Acronimo di *Verbale Interno*, viene utilizzato per tracciare nel tempo decisioni e azioni intraprese in un incontro interno. Il codice associato ai VI segue la forma VI x.y, dove x identifica il numero progressivo del verbale e y identifica una specifica decisione o voce registrata all'interno del verbale.

**ViT**

Acronimo di *Vision Transformer*, modello di visione artificiale basato sull'architettura Transformer che consente l'analisi del contenuto visivo delle immagini tramite rappresentazioni sequenziali. Nel contesto di sistemi multimodali, i ViT possono essere utilizzati per estrarre feature visive da integrare con informazioni testuali o audio.

**W****Walkthrough**

Revisione guidata di un documento, codice o progetto, in cui l'autore mostra il lavoro al gruppo per ricevere commenti, chiarire dubbi e individuare errori prima della consegna o della revisione formale.

**Way of Working**

Insieme di metodologie e strutture organizzative approvate dall'intero Team per svolgere in modo coerente i propri compiti.

**WhatsApp**

Applicazione di messaggistica istantanea adottata dal gruppo NullPointers Group per le comunicazioni interne e con l'azienda.

**Whisper**

Modello di riconoscimento vocale sviluppato da OpenAI specializzato nella trascrizione di audio in testo.

**X****XML**

XML (eXtensible Markup Language) è un formato di testo per rappresentare dati strutturati con tag personalizzabili, progettato per trasportare e archiviare informazioni in modo leggibile e interoperabile.

## Y

**Z**