



# Code7Crusaders

Software Development Team

Glossario

## **Membri del Team:**

Enrico Cotti Cottini, Gabriele Di Pietro, Tommaso Diviesti  
Francesco Lapenna, Matthew Pan, Eddy Pinarello, Filippo Rizzolo

Versioni

| Ver | Data       | Redattore            | Verificatore         | Descrizione                 |
|-----|------------|----------------------|----------------------|-----------------------------|
| 1.1 | 01/03/2025 | Eddy Pinarello       | Gabriele Di Pietro   | Aggiunta nuovi termini      |
| 1.0 | 04/02/2025 | Eddy Pinarello       | Enrico Cotti Cottini | Approvazione documento      |
| 0.3 | 21/01/2025 | Eddy Pinarello       | Gabriele Di Pietro   | Aggiunta di nuovi termini   |
| 0.2 | 18/11/2024 | Enrico Cotti Cottini | Gabriele Di Pietro   | Aggiunta di nuovi termini   |
| 0.1 | 05/11/2024 | Gabriele Di Pietro   | Filippo Rizzolo      | Prima stesura del documento |

# Indice

|  |           |
|--|-----------|
| <b>A</b>   | <b>4</b>  |
| Actual Cost . . . . .  | 4         |
| Analisi dei Requisiti . . . . .  | 4         |
| API REST (Representational State Transfer) . . . . .                     | 4         |
| Audit . . . . .  | 4         |
| <b>B</b>   | <b>5</b>  |
| Backend . . . . .  | 5         |
| Backlog . . . . .  | 5         |
| Benchmark . . . . .  | 5         |
| BERT (Bidirectional Encoder Representations from Transformers) . . . . . | 5         |
| <b>C</b>   | <b>6</b>  |
| Cost Variance . . . . .  | 6         |
| <b>D</b>   | <b>7</b>  |
| Database Relazionale . . . . .   | 7         |
| Database Vettoriale . . . . .  | 7         |
| Diagrammi UML . . . . .  | 7         |
| Documentazione Formale e Informale . . . . .                             | 7         |
| <b>E</b>   | <b>8</b>  |
| Earned Value . . . . .   | 8         |
| Embedding . . . . .  | 8         |
| Estimate at Completion . . . . .   | 8         |
| Estimate to Complete . . . . .   | 8         |
| <b>F</b>   | <b>9</b>  |
| FAISS . . . . .  | 9         |
| Frontend . . . . .   | 9         |
| <b>G</b>   | <b>10</b> |
| GitHub Actions . . . . .   | 10        |
| Glossario . . . . .  | 10        |
| <b>L</b>   | <b>11</b> |
| LangChain . . . . .  | 11        |
| LLM (Large Language Model) . . . . .                                     | 11        |
| <b>M</b>   | <b>12</b> |
| Machine Learning . . . . .   | 12        |
| Milestone . . . . .  | 12        |
| Mockup . . . . .   | 12        |
| <b>N</b>   | <b>13</b> |
| Natural Language Processing (NLP) . . . . .                              | 13        |
| Nearest Neighbor Search (NNS) . . . . .                                  | 13        |
| Norme di Progetto . . . . .  | 13        |
| <b>P</b>   | <b>14</b> |

|          |  |           |
|----------|--|-----------|
|          | PB (Product Baseline) . . . . .                      | 14        |
|          | Piano di Progetto . . . . .                          | 14        |
|          | Piano di Qualifica . . . . .                         | 14        |
|          | Pipeline RAG . . . . .                               | 14        |
|          | Planned Value . . . . .                              | 14        |
|          | PoC (Proof of Concept) . . . . .                     | 14        |
|          | PostgreSQL . . . . .                                 | 14        |
|          | Prompt . . . . .                                     | 14        |
|          | Props . . . . .                                      | 15        |
| <b>Q</b> |  | <b>16</b> |
|          | QMS . . . . .  | 16        |
|          | Query Semantica . . . . .                            | 16        |
| <b>R</b> |  | <b>17</b> |
|          | Requirements Stability Index . . . . .               | 17        |
|          | Roadmap di GitHub . . . . .                          | 17        |
|          | RTB (Requirements and Technology Baseline) . . . . . | 17        |
|          | Run . . . . .  | 17        |
| <b>S</b> |  | <b>18</b> |
|          | Scalabilità . . . . .                                | 18        |
|          | Scheduled Variance . . . . .                         | 18        |
|          | Stakeholder . . . . .                                | 18        |
| <b>T</b> |  | <b>19</b> |
|          | Template . . . . .                                   | 19        |
|          | Template di LaTeX . . . . .                          | 19        |
|          | Ticketing . . . . .                                  | 19        |
|          | Token . . . . .                                      | 19        |
| <b>V</b> |  | <b>20</b> |
|          | Versionamento . . . . .                              | 20        |

## A

### **Actual Cost**

L'Actual Cost (Costo Effettivo) rappresenta il costo effettivamente sostenuto per completare il lavoro fino al periodo in analisi.

### **Analisi dei Requisiti**

Processo di identificazione e definizione delle necessità e delle aspettative degli stakeholder per un progetto. Serve per ricavare i requisiti che andranno a essere soddisfatti tramite progettazione e sviluppo del prodotto.

### **API REST (Representational State Transfer)**

Un'architettura per la progettazione di servizi web che consente la comunicazione tra sistemi attraverso il protocollo HTTP. In un'API REST, le risorse (come dati o funzionalità) sono identificate da URL univoci e possono essere manipolate utilizzando operazioni standard come GET, POST, PUT e DELETE. L'architettura è stateless, il che significa che ogni richiesta contiene tutte le informazioni necessarie per essere elaborata senza fare affidamento su dati memorizzati sul server. È ampiamente utilizzata per creare applicazioni scalabili e interoperabili.

### **Audit**

Un audit è una revisione formale e indipendente di un progetto o di un processo, finalizzata a garantire che le attività siano conformi agli standard, alle normative e ai requisiti definiti. Si svolge attraverso ispezioni sistematiche e verifiche documentali per accertarsi che vengano seguite le procedure operative, i piani di qualità e le regolazioni previste. L'audit può anche essere utilizzato per identificare aree di miglioramento, valutare l'efficacia delle pratiche correnti e assicurarsi che i risultati siano raggiunti in modo efficace e trasparente.

## B

### Backend

La componente di un'applicazione software responsabile della logica di business, della gestione dei dati e dell'elaborazione delle richieste provenienti dal frontend. Si occupa di eseguire operazioni sul database, elaborare algoritmi complessi e garantire il corretto funzionamento delle funzionalità. Comunica con il frontend principalmente tramite API, fornendo i dati necessari per l'interfaccia utente.

### Backlog

Un elenco di attività, funzionalità o requisiti da completare all'interno di un progetto. Ogni voce del backlog rappresenta una parte del lavoro da svolgere, ordinata per priorità, che viene progressivamente affrontata durante lo sviluppo del progetto.

### Benchmark

Un benchmark è un punto di riferimento utilizzato per misurare le prestazioni di un sistema, di un'applicazione o di un dispositivo rispetto a un set di standard predefiniti o a sistemi comparabili. Nella tecnologia, un benchmark può riferirsi a test eseguiti per valutare l'efficienza, la velocità, la stabilità e altre caratteristiche importanti di hardware, software o processi aziendali. Questi test sono cruciali per identificare aree di miglioramento, confrontare soluzioni concorrenti e garantire che un sistema funzioni in modo ottimale rispetto alle aspettative del mercato o alle specifiche di progetto.

### BERT (Bidirectional Encoder Representations from Transformers)

BERT è un modello di linguaggio sviluppato da Google che utilizza trasformatori per comprendere il contesto di una parola considerando sia le parole precedenti che quelle successive. Questo approccio bidirezionale migliora la comprensione del testo in compiti come la classificazione, la risposta a domande e l'analisi del linguaggio naturale. BERT è pre-addestrato su grandi dataset e può essere facilmente adattato a diverse applicazioni specifiche.

## C

### **Cost Variance**

La Cost Variance (o Variazione dei Costi) rappresenta la differenza tra il valore del lavoro effettivamente completato e il costo effettivamente sostenuto per completarlo, calcolata in percentuale.

## D

### Database Relazionale

Un sistema di gestione dei dati che organizza le informazioni in tabelle strutturate composte da righe e colonne. Le tabelle sono correlate tra loro attraverso chiavi primarie e chiavi esterne, consentendo di mantenere l'integrità dei dati e facilitare l'accesso e la manipolazione. Utilizza il linguaggio SQL (Structured Query Language) per eseguire operazioni come la creazione, l'interrogazione, l'aggiornamento e l'eliminazione dei dati. I database relazionali sono ampiamente utilizzati in applicazioni aziendali e web per la loro efficienza e scalabilità.

### Database Vettoriale

Un database vettoriale è un tipo di sistema di gestione dei dati progettato per memorizzare, indicizzare e cercare vettori, ossia rappresentazioni numeriche di oggetti come parole, immagini o concetti. I vettori sono utilizzati per catturare le caratteristiche semantiche di un dato e vengono comunemente usati nei modelli di machine learning e di linguaggio naturale. I database vettoriali permettono di effettuare ricerche rapide basate sulla similarità tra vettori, come nel caso della ricerca per vicinanza (nearest neighbor search), che è utile per applicazioni come il recupero delle informazioni e la raccomandazione di contenuti.

### Diagrammi UML

Diagrammi standardizzati utilizzati per modellare e visualizzare il design di sistemi software. Aiutano a descrivere la struttura, il comportamento e le interazioni tra componenti del sistema. I diagrammi UML includono classi, casi d'uso, attività e molti altri tipi, ciascuno con uno scopo specifico per rappresentare aspetti diversi del sistema.

### Documentazione Formale e Informale

La documentazione formale include materiali ufficiali come norme e piani di progetto, spesso richiesta per conformità e audit. La documentazione informale comprende appunti e bozze non ancora approvate, ed è più flessibile, utilizzata principalmente per comunicazioni interne. Entrambe supportano lo sviluppo del progetto.



## **E**

### **Earned Value**

L'Earned Value (Valore Guadagnato) rappresenta il valore del lavoro effettivamente completato fino al periodo in analisi.

### **Embedding**

Una tecnica utilizzata nell'intelligenza artificiale e nell'elaborazione del linguaggio naturale per rappresentare parole, frasi o interi documenti in uno spazio vettoriale. Ogni elemento viene trasformato in un vettore numerico, dove la posizione e la distanza tra i vettori riflettono il significato semantico. Gli embedding consentono di confrontare testi in modo efficiente, trovando similarità tra concetti e facilitando attività come la ricerca semantica, la classificazione dei testi e il clustering.

### **Estimate at Completion**

L'Estimate at Completion (o Stima da Completare) rappresenta una previsione aggiornata del costo totale del progetto basata sulle performance attuali, calcolata in base ai costi effettivamente sostenuti e ai costi stimati per completare il lavoro rimanente.

### **Estimate to Complete**

L'Estimate to Complete (o Stima al Completamento) rappresenta una previsione del costo necessario per completare le attività rimanenti del progetto basata sulle performance attuali.

## **F**

### **FAISS**

Una libreria open-source sviluppata da Facebook AI Research per l'indicizzazione, la ricerca e il recupero efficiente di vettori in spazi di grandi dimensioni. FAISS (Facebook AI Similarity Search) è progettata per gestire nearest neighbor search in dataset di grandi dimensioni, garantendo alta velocità ed efficienza. Viene utilizzata in applicazioni come la ricerca semantica, i sistemi di raccomandazione e l'elaborazione del linguaggio naturale.

### **Frontend**

La parte visibile di un'applicazione software che rappresenta l'interfaccia utente (UI) e consente agli utenti di interagire con il sistema. È responsabile della presentazione grafica, della navigazione e dell'acquisizione delle informazioni inserite dagli utenti. Il frontend comunica con il backend attraverso API per inviare e ricevere dati, garantendo un'esperienza utente fluida e interattiva.

## G

### GitHub Actions

Una piattaforma di automazione che permette di eseguire flussi di lavoro direttamente nei repository GitHub. Supporta azioni come compilazioni, test e deployment automatizzati, migliorando l'efficienza nello sviluppo software.

### Glossario

Elenco strutturato di termini tecnici o specializzati, ognuno corredato dalla propria definizione o spiegazione. Questo strumento aiuta a migliorare la comunicazione tra le varie parti coinvolte in un progetto, riducendo le ambiguità e garantendo una comprensione condivisa dei termini utilizzati in un determinato contesto.

## **L**

### **LangChain**

LangChain è una libreria open-source progettata per semplificare l'integrazione e l'orchestrazione di modelli di linguaggio avanzati (come GPT) con altre fonti di dati e sistemi. Viene utilizzata per costruire applicazioni complesse che combinano intelligenza artificiale e database, rendendo possibile la gestione di flussi di lavoro dinamici. LangChain permette di creare pipeline che automatizzano il processamento delle informazioni, la comunicazione con le API, l'elaborazione di documenti e l'interazione con vari strumenti esterni. Con un focus sull'accessibilità e la modularità, LangChain rende più semplice l'implementazione di soluzioni di AI che richiedono un'integrazione fluida tra diversi moduli e fonti di dati.

### **LLM (Large Language Model)**

Un modello di intelligenza artificiale addestrato su grandi quantità di dati testuali per comprendere, generare e contestualizzare il linguaggio naturale. È utilizzato in applicazioni come chatbot, traduttori e analisi del linguaggio.

# M

## Machine Learning

Una branca dell'intelligenza artificiale che utilizza algoritmi per apprendere dai dati e migliorare le prestazioni senza essere esplicitamente programmata. Trova applicazione in ambiti come la previsione, il riconoscimento e la personalizzazione.

## Milestone

Un obiettivo intermedio significativo nel ciclo di vita del progetto, utilizzato per monitorare i progressi e verificare il completamento di specifiche fasi o attività.

## Mockup

Un mockup è una rappresentazione visiva di un'interfaccia utente, utilizzata per mostrare il design di un prodotto. È una versione statica che mostra come appariranno gli elementi grafici senza interazione.

## N

### **Natural Language Processing (NLP)**

Tecnologia che combina linguistica e intelligenza artificiale per consentire ai computer di comprendere, analizzare e generare linguaggio naturale. Viene usata in chatbot, traduttori automatici e analisi testuale.

### **Nearest Neighbor Search (NNS)**

Un algoritmo utilizzato per trovare i punti più vicini in uno spazio vettoriale, spesso utilizzato in database vettoriali per recuperare embedding simili.

### **Norme di Progetto**

Regole e linee guida stabilite all'interno di un progetto per garantire coerenza e qualità nelle attività svolte. Definiscono standard e procedure, come documentazione, gestione delle versioni e criteri di codifica, per assicurare uniformità nell'approccio e nel risultato finale.

## P

### PB (Product Baseline)

Una revisione che documenta lo stato del progetto al termine dello sviluppo, descrivendo il prodotto finale in termini di specifiche tecniche, funzionalità e conformità ai requisiti iniziali.

### Piano di Progetto

Documento formale che delinea in dettaglio la pianificazione, l'esecuzione, il monitoraggio e il controllo di tutte le attività coinvolte nella realizzazione di un progetto. Questo documento fornisce una roadmap chiara e organizzata, comprensiva di obiettivi, risorse, scadenze e strategie di gestione dei rischi.

### Piano di Qualifica

Documento che stabilisce gli standard di qualità, i processi e le attività di testing che saranno implementati durante lo sviluppo di un progetto. Contiene una descrizione dettagliata delle strategie di testing, delle metriche di valutazione e dei criteri di accettazione del prodotto finale.

### Pipeline RAG

Retrieval-Augmented Generation (RAG) è un'architettura che combina il recupero di informazioni da un database con modelli di linguaggio per generare risposte accurate e contestualizzate. Il processo si articola in due fasi principali: il retrieval e la generazione. Nella fase di retrieval, il sistema cerca e recupera documenti rilevanti da un database o da altre fonti di conoscenza, utilizzando tecniche di ricerca avanzate per identificare le informazioni più pertinenti. Successivamente, nella fase di generazione, un modello di linguaggio utilizza le informazioni recuperate per produrre risposte coerenti e arricchite dal contenuto estratto. Questo approccio migliora significativamente la qualità delle risposte, rendendole più precise e informate rispetto ai modelli di linguaggio tradizionali che non integrano il retrieval.

### Planned Value

Il Planned Value (Valore Pianificato) rappresenta il valore del lavoro programmato per essere completato fino a un determinato momento. Si tratta del budget preventivato per lo sprint in corso.

### PoC (Proof of Concept)

Una dimostrazione pratica per validare la fattibilità di un'idea, tecnologia o progetto. Permette di identificare eventuali rischi o limitazioni prima di procedere con lo sviluppo completo.

### PostgreSQL

PostgreSQL è un sistema di gestione di database relazionali open-source, noto per la sua robustezza, scalabilità e supporto per operazioni avanzate come transazioni ACID e gestione di dati JSON. È ampiamente utilizzato per applicazioni che richiedono affidabilità e performance.

### Prompt

Input testuale o visivo fornito a un'intelligenza artificiale per guidarne la generazione di risposte, immagini o altri output. Nel contesto dei chatbot e dei modelli di IA, il prompt determina il comportamento e il contenuto della risposta generata.

## Props

Meccanismo in React che permette di passare dati e funzioni da un componente padre a un componente figlio. I props sono immutabili e consentono la personalizzazione e il riutilizzo dei componenti, mantenendo un flusso unidirezionale dei dati.



## Q

### QMS

Metriche di Qualità Soddisfatte indica il numero di metriche implementate e soddisfatte, tra quelle definite.

### Query Semantica

La query semantica è una tecnica di ricerca che si basa sull'analisi del significato delle parole nel linguaggio naturale, piuttosto che sulla semplice corrispondenza di termini. Permette di trovare informazioni rilevanti in un database tenendo conto del contesto e delle relazioni tra concetti, migliorando la qualità dei risultati. Viene utilizzata in motori di ricerca e assistenti virtuali per comprendere meglio le intenzioni dell'utente.

## R

### Requirements Stability Index

Il Requirements Stability Index indica la percentuale di requisiti che sono rimasti invariati rispetto al totale dei requisiti inizialmente definiti. Si tratta di una metrica utilizzata per misurare quanto i requisiti di un progetto rimangono stabili durante il ciclo di vita del progetto stesso, è particolarmente utile per comprendere l'impatto delle modifiche ai requisiti sul progetto.

### Roadmap di GitHub

Strumento visuale per pianificare e monitorare il progresso delle attività di un progetto all'interno di GitHub. Gestite tramite GitHub Projects, le roadmap aiutano a pianificare iterazioni, milestone e obiettivi di un progetto software, facilitando la gestione delle priorità, delle scadenze e dell'assegnazione dei compiti.

### RTB (Requirements and Technology Baseline)

Una revisione formale che stabilisce i requisiti funzionali e tecnici di un progetto, definendo il punto di riferimento per lo sviluppo e la verifica delle funzionalità richieste.

### Run

Una Run rappresenta una singola esecuzione o operazione all'interno di un chatbot basato su LangSmith. Ogni Run traccia un'unità di lavoro, come una chiamata al modello di linguaggio (LLM), l'esecuzione di una catena di operazioni o l'invocazione di uno strumento. LangSmith memorizza questi dati in un formato strutturato, permettendo di monitorare le prestazioni, analizzare errori e ottimizzare le risposte del chatbot. Ogni Run può includere informazioni come tempi di esecuzione, input e output, costi di elaborazione e stato dell'operazione.

## S

### Scalabilità

La scalabilità è la capacità di un sistema di gestire un aumento del carico di lavoro o delle risorse senza compromettere le prestazioni. Un sistema scalabile può adattarsi a una maggiore domanda aggiungendo risorse in modo efficace, sia aumentando la capacità di singole risorse (scalabilità verticale) che aggiungendo più risorse (scalabilità orizzontale). È essenziale per mantenere alte le performance in ambienti in crescita.

### Scheduled Variance

La Schedule Variance (o Variazione di Programma) rappresenta la differenza tra il valore del lavoro effettivamente completato e il valore del lavoro pianificato, calcolata in percentuale.

### Stakeholder

Insieme dei portatori di interessi, ossia le persone che, a vario titolo, sono coinvolte nel ciclo di vita del software, influenzandone il prodotto o il processo. Possono includere clienti, utenti finali, sviluppatori, finanziatori e altri soggetti che, direttamente o indirettamente, contribuiscono al successo del progetto, determinando requisiti, vincoli e obiettivi da raggiungere.

## **T**

### **Template**

Funzionalità predefinite di un chatbot che permettono di rispondere a domande frequenti o eseguire azioni specifiche senza dover fare una chiamata all'LLM ogni volta. I template sono domande registrate nel sistema, associate a risposte già esistenti, configurate dagli amministratori. Questo permette di rispondere rapidamente a richieste comuni, migliorando l'efficienza e riducendo il carico computazionale.

### **Template di LaTeX**

Un modello predefinito per la creazione di documenti in LaTeX. Consente di standardizzare la struttura e il formato dei documenti, semplificando la redazione e migliorando la coerenza stilistica.

### **Ticketing**

Sistema di gestione e tracciamento delle attività, problematiche o richieste in un progetto, che consente di assegnare, monitorare e risolvere ciascun elemento in modo ordinato e strutturato, migliorando l'efficienza del team e la visibilità sul progresso.

### **Token**

Un token è l'unità minima di testo elaborata da un modello di linguaggio. Può essere una parola intera, un simbolo, una parte di una parola o una sequenza di caratteri che il modello interpreta come un singolo elemento. I modelli di linguaggio suddividono il testo in token per analizzare e comprendere il contenuto, facilitando l'elaborazione del linguaggio naturale. La suddivisione in token è un passaggio cruciale nella comprensione e generazione del testo da parte dei modelli di intelligenza artificiale.

## V

### **Versionamento**

Processo di gestione delle modifiche ai documenti o al codice sorgente mediante strumenti che tracciano e memorizzano le varie versioni. Garantisce trasparenza e facilita la collaborazione tra membri del team.