



HEPTA CODE

Glossario

Versione 2

Contenuto del Documento

Termini ambigui e le loro definizioni

email: heptacode7@gmail.com

sito: heptacode-unipd.github.io

Registro delle modifiche

Versione	Data	Autore	Verificatore	Descrizione
2.3.0	2026/03/12	Angela Favaro	Laura Venturini	Aggiunta terminologie per Piano di Progetto
2.2.0	2026/03/12	Angela Favaro	Laura Venturini	Aggiunta terminologie per Specifica Tecnica
2.1.0	2026/03/07	Angela Favaro	Riccardo Baldin	Aggiunta terminologie per Specifica Tecnica
2.0.0	2026/02/16	Angela Canazza	Angela Favaro	Rilascio versione 2.0.0 per RTB
1.2.0	2025/12/21	Alberto Reginato	Angela Canazza	Aggiunta terminologie per AdR
1.1.0	2025/12/20	Angela Favaro	Amerigo Vegliante	Aggiunta terminologie per RTB
1.0.0	2025/10/30	Angela Favaro	Angela Canazza	Revisione documento con nuove terminologie
0.1.0	2025/10/27	Nicola Simionato	Alberto Reginato	Creazione documento e primi termini

Indice dei contenuti

1. Scopo del Documento	1
2. A	2
3. B	3
4. C	4
5. D	5
6. F	6
7. G	7
8. H	8
9. I	9
10. L	10
11. M	11
12. N	12
13. O	13
14. P	14
15. R	16
16. S	17
17. T	19
18. U	20
19. V	21
20. W	22
21. Z	23

1. Scopo del Documento

Il fine di questo documento è tenere traccia di tutti i termini, ambigui o meno, che possono apparire all'interno della documentazione o nei meeting e darne una definizione quanto più precisa, in modo che le risorse e i discorsi fatti possano essere compresi facilmente.

2. A

2.1. Agente

Sistema software capace di percepire l'ambiente circostante, ragionare su ciò che accade e compiere azioni autonome per raggiungere obiettivi specifici, «agisce» per completare un compito.

2.2. Agile

Approccio filosofico e culturale alla gestione dei progetti. Si basa sullo sviluppo iterativo, sulla consegna continua di valore e sulla capacità di rispondere rapidamente ai cambiamenti, invece di seguire un piano rigido e lineare.

2.3. Amazon Web Services (AWS)

Piattaforma di Cloud Computing più completa e utilizzata al mondo, offerta da Amazon. Invece di acquistare, possedere e gestire server fisici e centri dati (Data Center) di proprietà, un'azienda può noleggiare potenza di calcolo, spazio di archiviazione, database e altre tecnologie direttamente dai server di Amazon, pagando solo per ciò che consuma.

2.4. Artificial Intelligence (AI)

Branca dell'informatica che si occupa di creare sistemi hardware e software capaci di simulare processi cognitivi umani. In termini semplici, è la capacità di una macchina di mostrare abilità umane come il ragionamento, l'apprendimento, la pianificazione e la creatività.

2.5. Asset Software

Qualsiasi risorsa digitale che ha valore per l'organizzazione e il progetto. Include non solo il codice sorgente, ma anche la documentazione, i dati, le immagini, le licenze e i file di configurazione necessari al funzionamento del software.

2.6. Attore

Ruolo svolto da un'entità esterna che interagisce con il sistema. Non è la persona fisica in sé, ma la funzione che essa ricopre mentre usa il software.

2.7. Audit

Revisione sistematica e indipendente del codice, dei processi o della sicurezza del sistema. Serve a verificare la conformità a standard specifici, normative o best practice e a identificare eventuali vulnerabilità o inefficienze.

3. B

3.1. Backend

Tutto ciò che avviene «dietro le quinte» di un'applicazione. È la parte invisibile all'utente, ma è il vero «cervello» del software: gestisce la logica, il salvataggio dei dati, la sicurezza e la comunicazione con altri sistemi.

3.2. Best Practice

Insieme di metodi, tecniche o procedure che si sono dimostrati, attraverso l'esperienza e la ricerca, più efficaci di altri per raggiungere un determinato risultato. Nello sviluppo software, indicano le convenzioni raccomandate per scrivere codice pulito e manutenibile.

3.3. Branch

Si tratta di ambienti di sviluppo diversi gli uni dagli altri, tra i quali si può lavorare in modo parallelo e separando logicamente il lavoro.

3.4. Browser

Software applicativo progettato per consentire agli utenti di accedere, recuperare e visualizzare i contenuti presenti sul World Wide Web (Internet).

3.5. Business Owner

Nel contesto aziendale e dello sviluppo prodotto (specialmente in ambiti Agile e Scrum), è una figura strategica che detiene la responsabilità ultima del valore di business generato da un progetto o da un prodotto.

4. C

4.1. Caso d'uso

Descrizione di una sequenza di azioni che un sistema compie per fornire un risultato di valore a un determinato utente, chiamato Attore.

4.2. Chiavi API

Codice univoco (spesso una stringa alfanumerica) utilizzato per identificare e autenticare un'applicazione o un utente che richiede l'accesso ad un'API. Funziona come una password per i programmi.

4.3. Cloud

Modello di fornitura di servizi informatici (come server, archiviazione, database, rete, software) tramite Internet. Permette l'accesso on-demand alle risorse senza la necessità di gestire fisicamente l'infrastruttura.

4.4. Code Coverage

Metrica utilizzata nei test del software che indica la percentuale di codice sorgente che viene eseguita (coperta) durante l'esecuzione dei test automatici. Una copertura alta riduce la probabilità di bug non rilevati.

4.5. Code Smell

Caratteristica del codice sorgente che, pur non essendo tecnicamente un errore (il codice funziona), indica una debolezza nella progettazione o nello stile che potrebbe causare problemi futuri o difficoltà nella manutenzione.

4.6. Commit

Operazione che registra in modo permanente le modifiche apportate a uno o più file all'interno di un VCS. Ogni commit, solitamente accompagnato da una descrizione, crea un nuovo punto nella cronologia del progetto, permettendo di ripristinare una versione precedente.

4.7. Continuous Delivery (CD)

Pratica di sviluppo software in cui le modifiche al codice vengono preparate automaticamente per il rilascio in produzione. È l'estensione della Continuous Integration e garantisce che il software sia sempre in uno stato rilasciabile.

4.8. Continuous Integration (CI)

Pratica che consiste nell'unire frequentemente le copie di lavoro di tutti gli sviluppatori in una linea principale condivisa (mainline), diverse volte al giorno. Ogni integrazione è verificata da una build automatica e da test per rilevare errori il più rapidamente possibile.

5. D

5.1. Dashboard

Interfaccia grafica che fornisce una vista d'insieme dei dati e di altre metriche di sistema, permettendo un monitoraggio rapido e intuitivo.

5.2. Deployment

Processo attraverso cui un'applicazione software viene resa operativa e disponibile per l'utilizzo nel suo ambiente di destinazione — che sia un server, un cloud, un dispositivo mobile o un browser. Tutto ciò che accade dal momento in cui il codice è pronto a quando l'utente può effettivamente usarlo.

5.3. Design Pattern

Soluzione generale, riutilizzabile e collaudata a un problema ricorrente che si incontra nella progettazione del software. Non è codice pronto all'uso, ma un modello concettuale — una descrizione di come strutturare classi, oggetti e le loro interazioni per risolvere un determinato tipo di problema in un determinato contesto.

5.4. Design Thinking

Metodologia orientata alla risoluzione di problemi complessi, che parte dalla raccolta di idee generali e continua con l'affinamento delle stesse in un progetto più strutturato.

5.5. Developer

Professionista che progetta, scrive, testa e mantiene il codice sorgente di programmi informatici e applicazioni. È la figura che traduce i requisiti logici (il «cosa» deve fare il programma) in linguaggio macchina (il «come» farlo tecnicamente).

5.6. Diagrammi di PERTT

Strumento di project management utilizzato per pianificare, organizzare e coordinare i compiti all'interno di un progetto. Rappresentazione grafica reticolare che si concentra sulle relazioni di precedenza tra le attività.

5.7. Disco Rigido (HDD)

Dispositivo di archiviazione principale di un computer, utilizzato per conservare i dati in modo permanente.

5.8. Discord

Applicazione di messaggistica istantanea, VoIP e di distribuzione digitale. Gli utenti possono interagire privatamente o in comunità che possono contenere canali sia testuali che vocali.

5.9. Docs

Abbreviazione per «documentazione». Indica l'insieme dei documenti relativi al progetto, solitamente raccolti e gestiti in una cartella denominata «docs» all'interno della repository.

6. F

6.1. Fail Fast

Metodologia che mira ad identificare rapidamente un malfunzionamento all'interno di un progetto, in modo da poterlo correggere prima che questo porti a un accumulo di problemi di difficile gestione.

6.2. Framework

Struttura software preconfezionata che fornisce un'architettura di base, un insieme di strumenti e convenzioni su cui costruire un'applicazione, definendo lo scheletro entro cui lo sviluppatore inserisce il proprio codice. Ambiente strutturato che impone un modo preciso di organizzare e sviluppare il software

6.3. Frontend

Tutto ciò che un utente vede e con cui interagisce direttamente quando utilizza un sito web o un'applicazione. È la parte «visibile» del software, quella che trasforma il codice in una UI comprensibile.

7. G

7.1. Git

Software di versionamento distribuito dei file di una cartella, utilizzabile da linea di comando.

7.2. GitHub

Servizio web basato su git. Permette il versionamento dei file di un progetto online, aprendo quindi la collaborazione di più persone sui file. Offre inoltre diversi altri servizi come l'Issue Tracking System, la possibilità di gestire organizzazioni oppure hostare siti web.

7.3. GitHub Action (GHA)

Sistema di automazione offerto da GitHub e personalizzabile. Permette, al trigger di determinati eventi, di eseguire script ad esempio per la compilazione di file sorgenti oppure testare il software.

7.4. Google Meet

Servizio di videoconferenze che permette di organizzare e partecipare a riunioni online tramite video e audio in tempo reale.

8. H

8.1. Hardware

Parte fisica, tangibile e «pesante» di un sistema informatico. È l'insieme di tutti i componenti meccanici, elettronici, elettrici e magnetici che puoi toccare con mano.

9. I

9.1. Issue

Elemento atto a: pianificare, discutere e tracciare lavoro, bug, idee o nuove funzionalità all'interno di un repository di codice nella piattaforma di Github. Ad essa è possibile assegnare: un progetto, una Milestone, dei responsabili e delle etichette per rappresentarla all'interno di determinate categorie e facilitarne l'utilizzo e la ricerca.

10. L

10.1. Large Language Model (LLM)

Tipo di AI addestrato su enormi quantità di dati testuali per comprendere, generare e elaborare il linguaggio umano in modo simile a un essere umano, Esso è capace di compiti come la scrittura, la traduzione, il riassunto di testi e la conversazione, basandosi su architetture di deep learning, i quali gli permettono di cogliere contesti complessi.

10.2. Libreria

Raccolta di codice precompilato e riutilizzabile — funzioni, classi, costanti, routine — che un programma può importare e invocare per svolgere compiti comuni senza doverli implementare da zero.

- Non detta la struttura del programma. Il codice della libreria viene invocato dallo sviluppatore.
- È riutilizzabile, per essere condivisa tra progetti diversi.
- Può essere statica (incorporata direttamente nell'eseguibile a compile-time) o dinamica (caricata a runtime e può essere condivisa tra più programmi).

11. M

11.1. Machine Learning

Sottoinsieme dell'Intelligenza Artificiale che si concentra sulla creazione di sistemi che apprendono o migliorano le proprie performance in base ai dati che utilizzano, senza essere stati esplicitamente programmati per ogni singola regola.

11.2. Mail

Messaggio digitale che viene inviato e ricevuto tramite internet. Strumento di comunicazione scritto che permette di scambiare informazioni rapidamente tra persone o gruppi.

11.3. Milestone

Punto di controllo significativo che segna il completamento di una fase importante o il raggiungimento di un obiettivo chiave all'interno di un progetto. Rappresenta un traguardo raggiunto.

11.4. Minimum Viable Product

Anche chiamato **MVP**.

Versione del prodotto software che possiede appena le caratteristiche sufficienti per essere pubblicato e testato da utenti reali, con il minimo sforzo di sviluppo possibile.

11.5. Mockup

Rappresentazione statica e ad alta fedeltà dell'interfaccia di un'applicazione o sito web. Mostra come apparirà visivamente il prodotto finale — colori, typography, layout, icone, spaziature — ma senza funzionalità interattive.

11.6. MongoDB

Sistema di gestione di basi dati, basato su un modello a documenti, anziché sulle classiche tabelle a righe e colonne. È progettato per essere estremamente flessibile, scalabile e capace di gestire grandi quantità di dati non strutturati (come post sui social, cataloghi prodotti complessi o dati provenienti da sensori).

12. N

12.1. Node.js

Ambiente di runtime JavaScript open-source e multiplatforma che permette di eseguire codice JavaScript al di fuori di un browser web, tipicamente utilizzato per lo sviluppo lato server (Backend).

12.2. Normativa ISO

Standard tecnico volontario internazionale sviluppato dall'International Organization for Standardization (ISO) per definire requisiti, linee guida o specifiche che assicurano qualità, sicurezza, efficienza e interoperabilità di prodotti, processi e servizi, creando un linguaggio e un framework comuni riconosciuti a livello globale, che le organizzazioni possono adottare e certificare.

13. O

13.1. OWASP

Acronimo per Open Worldwide Application Security Project, una fondazione no-profit e una comunità globale aperta che si dedica a migliorare la sicurezza delle applicazioni software rendendo disponibili gratuitamente strumenti, documentazione e best practice per sviluppatori e organizzazioni, al fine di educare e guidare nella creazione di software più sicuri.

14. P

14.1. Progettazione & Analisi Baseline (PAB)

Milestone interna al gruppo che richiede la conclusione dell'attività di Analisi e di Progettazione per essere completata.

14.2. Personas utente (Personas)

Rappresentazioni fittizie ma realistiche dei gruppi di utenti target, basate su dati e ricerche. Aiutano il team di sviluppo a comprendere i bisogni, le esperienze, i comportamenti e gli obiettivi degli utenti reali per cui stanno progettando il prodotto.

14.3. Product Baseline

Anche chiamata **PB**.

Milestone che richiede la consegna del prodotto e l'accettazione da parte dei professori per essere superata.

14.4. Project Board

Strumento di visualizzazione dei compiti (task o issue) che permette di monitorare lo stato di avanzamento di un progetto in tempo reale.

14.5. Piano di Contingenza

Insieme di procedure da attivare dopo che un rischio si è effettivamente verificato e si è trasformato in un problema reale.

14.6. Piano di Mitigazione

Insieme di azioni strategiche e operative pianificate in anticipo per ridurre la probabilità che un rischio si verifichi o per limitare l'impatto negativo qualora l'evento dovesse accadere.

14.7. Product Backlog

Lista ordinata e dinamica di tutto ciò che potrebbe essere necessario per migliorare il prodotto. È l'unica fonte di requisiti per ogni modifica da apportare al software nel framework Scrum. Evolve man mano che il prodotto e l'ambiente in cui viene utilizzato cambiano.

14.8. Project Manager

Figura professionale responsabile della pianificazione, dell'esecuzione e della chiusura di un progetto. Il suo obiettivo principale è garantire che il lavoro venga completato entro i tempi, i costi e i requisiti di qualità stabiliti (il cosiddetto «triangolo dei vincoli»).

14.9. Provider

Indica un'entità o un fornitore che eroga un servizio (es. AWS per l'infrastruttura Cloud).

14.10. Pull

Comando di Git usato per scaricare la versione più recente di un file o di un progetto dall'ambiente condiviso con il resto dei collaboratori.

14.11. Pull Request (PR)

Meccanismo utilizzato nei tool di versionamento per notificare ai membri del team che una funzionalità o una modifica è stata completata su un branch secondario e si richiede la revisione del codice prima di unirla al ramo principale.

14.12. Push

Comando di Git usato per inviare i commit locali ad un repository remoto, condividendo così le proprie modifiche con il resto del team.

14.13. Python

Linguaggio di programmazione ad alto livello, interpretato e versatile. È ampiamente utilizzato nel campo dell'Intelligenza Artificiale, del Machine Learning e dell'analisi dati grazie alla sua sintassi semplice ed alle svariate librerie disponibili.

15. R

15.1. React.js

Libreria JavaScript open-source utilizzata per creare interfacce utente (UI). Permette di creare componenti riutilizzabili che gestiscono il proprio stato.

15.2. Remediation

Processo di correzione delle vulnerabilità, dei difetti o delle non conformità identificate durante un Audit o un test di sicurezza atto a ripristinare la sicurezza e l'integrità del sistema.

15.3. Repository (Repo)

Ambiente digitale dove vengono salvati tutti i file riguardanti i progetti. Può esserne più di una in base alla suddivisione degli argomenti per cui vengono create.

15.4. Responsabile

Figura professionale incaricata di guidare un team tecnico nella realizzazione di un prodotto informatico, assicurando che vengano rispettati gli obiettivi di business, i tempi di consegna e i parametri di qualità.

15.5. Rischio

Evento o una condizione incerta che, se si verifica, ha un effetto (positivo o negativo) sugli obiettivi del progetto, come i tempi, i costi, l'ambito o la qualità.

15.6. Requirements & Technology Baseline (RTB)

Milestone che fissa i requisiti da soddisfare, in accordo con il proponente; nel documento Analisi dei Requisiti motiva le tecnologie, i framework, le librerie adottate, e ne dimostra adeguatezza e interoperabilità. Tramite Proof of Concept (demo eseguibile) rappresentativa delle richieste del capitolato. Il PoC viene posto in repo accessibile a committente e proponente.

16. S

16.1. Schema-less

Caratteristica di alcuni database NoSQL (come MongoDB) che non richiedono una struttura rigida e predefinita dei dati (schema) prima dell'inserimento. Ciò permette una maggiore flessibilità nel memorizzare dati eterogenei.

16.2. Scrum

Framework Agile più utilizzato al mondo per la gestione dello sviluppo di prodotti complessi, in particolare nel software. Non è una metodologia rigida, ma una struttura leggera che aiuta i team a generare valore attraverso soluzioni adattive a problemi complessi.

16.3. Secret Scanning

Processo automatizzato che analizza il codice sorgente, i log e altri file alla ricerca di credenziali esposte accidentalmente, come password, chiavi API o token privati, per prevenire accessi non autorizzati.

16.4. Single Page Application (SPA)

Applicazione web o sito che interagisce con l'utente riscrivendo dinamicamente la pagina corrente con nuovi dati dal server, anziché caricare intere nuove pagine. Questo rende l'esperienza utente più fluida e simile a un'applicazione desktop.

16.5. Slack

Piattaforma di comunicazione asincrona collaborativa basata sulla messaggistica istantanea, progettata specificamente per i team di lavoro e le aziende.

16.6. Server

Computer o un software che fornisce dati, servizi o risorse ad altri computer, chiamati client, attraverso una rete (locale o internet).

16.7. Specifica Tecnica (ST)

Documento formale che descrive in modo preciso e dettagliato come un sistema software deve essere progettato e realizzato per soddisfare i requisiti stabiliti. Ponte tra la fase di analisi e la fase di sviluppo, fornendo a tutto il team tecnico un riferimento univoco e condiviso.

16.8. Sprint

Intervallo di tempo predefinito e limitato (solitamente tra 1 e 4 settimane) durante il quale un team di lavoro deve completare una quantità specifica di lavoro.

16.9. Sprint Backlog

Insieme degli elementi del Product Backlog selezionati per lo Sprint corrente, con l'aggiunta di un piano dettagliato per consegnare l'incremento di prodotto e realizzare lo Sprint Goal. Una previsione a breve termine che appartiene interamente agli Sviluppatori.

16.10. Sprint Goal

Unico scopo prefissato per lo Sprint. Sebbene lo Sprint Backlog contenga l'elenco dei singoli compiti da svolgere, lo Sprint Goal rappresenta il «perché» il team sta investendo tempo ed energie in quel ciclo di lavoro. Viene definito durante lo Sprint Planning e serve come guida condivisa per il team, fornendo flessibilità riguardo alle funzionalità specifiche che vengono implementate.

16.11. Sprint Planning

Evento che dà ufficialmente inizio a ogni Sprint. Si tratta di una riunione collaborativa in cui l'intero Scrum Team si riunisce per definire che cosa verrà fatto nel ciclo di lavoro appena iniziato e come verrà realizzato.

16.12. Sprint Retrospective

Una delle cerimonie fondamentali del framework Agile Scrum. Si tiene alla fine di ogni Sprint, subito dopo la Sprint Review, e ha lo scopo di analizzare come ha lavorato il team per migliorare i processi nel ciclo successivo.

16.13. Sprint Review

Evento fondamentale del framework Scrum che si tiene alla fine di ogni Sprint. Il suo scopo principale è ispezionare l'Incremento (il lavoro completato) e adattare il Product Backlog se necessario.

16.14. Stakeholder

Qualsiasi individuo, gruppo o organizzazione che possa essere direttamente o indirettamente coinvolto dalle attività, dalle decisioni o dai risultati di un'azienda.

16.15. Software

Insieme delle componenti immateriali (programmi, dati e istruzioni) che dicono a un computer o a un dispositivo elettronico cosa fare.

16.16. Sviluppatore

Vedi definizione di ["Developer"](#).

17. T

17.1. Telegram

Applicazione di messaggistica istantanea tra utenti e/o gruppi. Permette in particolar modo di creare sottocategorie all'interno di un gruppo, per isolare differenti tematiche in compartimenti separati. Consente inoltre l'invio di file di tipo testuale, visivo e audio.

17.2. Token

Oggetto digitale (spesso una stringa crittografata) che rappresenta il diritto di eseguire un'operazione. Nell'autenticazione, viene usato per provare l'identità di un utente o di un servizio senza dover reinserire le credenziali ad ogni richiesta.

17.3. Training

Processo attraverso il quale un modello di Intelligenza Artificiale o un sistema software autonomo impara a compiere azioni specifiche per raggiungere un obiettivo all'interno di un ambiente.

17.4. Typst

Sistema di preparazione documenti basato su un linguaggio di markup leggero, progettato per essere un'alternativa moderna e più semplice a LaTeX, ma più potente di Word o Google Docs.

18. U

18.1. Use Case (UC)

Vedi definizione di ["Caso d'uso"](#).

18.2. UCB - Use Case Baseline

Milestone interna al gruppo che richiede la conclusione della stesura di tutti gli Use Case del progetto per essere completata. Fondamentale per definire i requisiti funzionali del software.

18.3. User Interface (UI)

Insieme di tutti gli elementi visivi e interattivi attraverso i quali un essere umano comunica con un software, un'applicazione o un dispositivo Hardware.

19. V

19.1. Verificatore

Figura professionale responsabile di esaminare un prodotto software per identificare difetti, errori (bug) o mancanze rispetto ai requisiti iniziali.

19.2. Version Control System (VCS)

Un sistema di controllo di versione che permette di tenere traccia nel tempo di tutte le modifiche riguardanti un file, compresa la sua creazione ed eliminazione. Permette di registrare la cronologia delle revisioni, ripristinare versioni precedenti e facilitare la collaborazione simultanea di più persone.

19.3. Visual Studio Code (VScode)

Editor di testo e in particolare di codice sorgente. Offre una gamma di strumenti per facilitare lo sviluppo del codice attraverso funzionalità integrate, come strumenti di debugging, controllo versione e un vasto ecosistema di estensioni.

20. W

20.1. Way of Working (WoW)

Insieme di regole, metodologie, strumentazione e processi atti ad un avanzamento del progetto secondo modi concordati e accettati dai membri del gruppo.

20.2. Web App

Software applicativo che non risiede sul disco rigido del tuo dispositivo (PC, smartphone o tablet), ma viene eseguito su un server remoto e reso accessibile tramite un comune browser web (come Chrome, Safari o Firefox).

21. Z

21.1. Zoom

Applicazione software, per computer e dispositivi mobili, che permette la creazione e gestione di conferenze online, condivisibili anche tramite calendario tramite pianificazione.