



UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA

Informatica

**Ingegneria del Software**

Anno Accademico: 2023/2024

---

# JACKPOT CODING

Gruppo: Jackpot Coding  
Email: [jackpotcoding@gmail.com](mailto:jackpotcoding@gmail.com)

## GLOSSARIO

DESTINATARI: Prof. T. Vardanega, Prof. R. Cardin, Zucchetti S.p.A.

USO: ESTERNO  
VERSIONE: 1.0.4

## Registro delle modifiche

Versione	Data	Autore	Verificatore	Modifica
v1.0.4	07/05/2024	G. Moretto	M. Gobbo	Correzione code coverage e rimozione tempo di apprendimento e risposta
v1.0.3	25/04/2024	R. Simionato	G. Moretto	Aggiunte definizioni dal documento "Specifica Tecnica"
v1.0.2	25/04/2024	R. Simionato	G. Moretto	Aggiunte definizioni metriche dal documento "Piano di Qualifica"
v1.0.1	04/04/2024	G. Moretto	R. Simionato	Rimozione Redattori e Verificatori come da feedback RTB
v1.0.0	18/03/2024	-	E. Gallo	Verifica e approvazione documento
v0.1.4	18/03/2024	G. Moretto	E. Gallo	Aggiunte definizioni dal documento "Piano di Qualifica"
v0.1.3	09/03/2024	M. Gobbo	G. Moretto	Aggiunte definizioni dal documento "Piano di Progetto"
v0.1.2	04/03/2024	E. Gallo	G. Moretto	Aggiunte definizioni dal documento "Norme di Progetto"
v0.1.1	02/03/2024	E. Gallo	G. Moretto	Migliorata struttura documento e leggibilità del registro delle modifiche
v0.1.0	02/03/2024	E. Gallo	G. Moretto	Inserita sezione Introduzione e struttura. Verifica del documento
v0.0.3	16/02/2024	G. Moretto	E. Gallo	Inserito definizioni per i termini nelle sezioni B-U
v0.0.2	13/02/2024	G. Moretto	E. Gallo	Aggiunto termini, inserito definizioni per i termini nella sezione A
v0.0.1	15/11/2023	E. Gallo	G. Moretto	Creata struttura del documento

# Indice

<b>Introduzione e struttura</b>	<b>6</b>
<b>A</b>	<b>7</b>
Actual Cost . . . . .	7
AI/IA . . . . .	7
Agile . . . . .	7
Amministratore . . . . .	7
Analisi dei Requisiti . . . . .	7
API . . . . .	7
Attore . . . . .	7
<b>B</b>	<b>8</b>
Backlog . . . . .	8
Branch . . . . .	8
Brainstorming . . . . .	8
Browser . . . . .	8
Budget Variance . . . . .	8
<b>C</b>	<b>9</b>
Capitolato . . . . .	9
Caso d'uso . . . . .	9
ChatGPT . . . . .	9
Code Coverage . . . . .	9
Committente . . . . .	9
Comunicazione sincrona . . . . .	9
Comunicazione asincrona . . . . .	9
Copertura dei Requisiti . . . . .	9
Cost Variance . . . . .	9
Credenziali . . . . .	9
CSS . . . . .	9
<b>D</b>	<b>10</b>
Database . . . . .	10
Dashboard . . . . .	10
Design pattern . . . . .	10
Discord . . . . .	10
<b>E</b>	<b>11</b>
Earned Value . . . . .	11
Efficienza . . . . .	11
Estimated At Completion . . . . .	11
Estimated To Completion . . . . .	11
<b>F</b>	<b>12</b>
Failure Density . . . . .	12
File . . . . .	12
Form . . . . .	12
Formato . . . . .	12
Framework . . . . .	12
<b>G</b>	<b>13</b>
Git . . . . .	13
GitHub . . . . .	13

<b>H</b>		<b>14</b>
HTML . . . . .		14
<b>I</b>		<b>15</b>
Implementazione . . . . .		15
Incrementi . . . . .		15
Indice Gulpease . . . . .		15
<b>Input</b>		<b>15</b>
Issue . . . . .		15
<b>J</b>		<b>16</b>
Jira . . . . .		16
<b>K</b>		<b>17</b>
<b>L</b>		<b>18</b>
LaTeX . . . . .		18
Libreria . . . . .		18
Linguaggio naturale . . . . .		18
LLM . . . . .		18
Login . . . . .		18
Logout . . . . .		18
<b>M</b>		<b>19</b>
Main branch . . . . .		19
Merging . . . . .		19
Milestone . . . . .		19
Modello . . . . .		19
MPC . . . . .		19
MPD . . . . .		19
<b>N</b>		<b>20</b>
Non-Calculated Risks . . . . .		20
Norme di Progetto . . . . .		20
<b>O</b>		<b>21</b>
<b>P</b>		<b>22</b>
Passed Test Percentage . . . . .		22
Password . . . . .		22
PoC . . . . .		22
Piano di Progetto . . . . .		22
Piano di Qualifica . . . . .		22
Planned Value . . . . .		22
Product Baseline . . . . .		22
Prompt . . . . .		22
Prompting . . . . .		22
Proponente . . . . .		22
Python . . . . .		22
<b>Q</b>		<b>23</b>
Quality Metric Satisfied . . . . .		23
Query SQL . . . . .		23

<b>R</b>	<b>24</b>
Raggiunta dell'Obiettivo . . . . .	24
Repository . . . . .	24
Requirements Stability Index . . . . .	24
Requirements & Technology Baseline . . . . .	24
Requisito . . . . .	24
<b>S</b>	<b>25</b>
Satisfied Obligatory Requirements . . . . .	25
Schedule Variance . . . . .	25
Sinonimi . . . . .	25
Software . . . . .	25
Software House . . . . .	25
Sprint . . . . .	25
SQL . . . . .	25
Strategy . . . . .	25
Struttura database . . . . .	25
<b>T</b>	<b>26</b>
Tabella . . . . .	26
Task . . . . .	26
Telegram . . . . .	26
Ticketing . . . . .	26
Tutorial . . . . .	26
<b>U</b>	<b>27</b>
Utente . . . . .	27
UML . . . . .	27
Unità architettureali . . . . .	27
<b>V</b>	<b>28</b>
Versionamento . . . . .	28
<b>W</b>	<b>29</b>
Way of Working . . . . .	29
<b>X</b>	<b>30</b>
<b>Y</b>	<b>31</b>
<b>Z</b>	<b>32</b>

## Introduzione e struttura

Il presente documento ha lo scopo di individuare le terminologie di progetto e facilitarne la comprensione sia ai membri del gruppo sia soprattutto alle persone esterne o eventuali interessati. È suddiviso in sezioni, inserite in ordine alfabetico per facilitarne la ricerca e la lettura.

Ciascun documento inoltre, riporta al proprio interno una sezione dedicata al glossario, con annesso collegamento. Ogni termine presente in questo documento, è indicato con una <sup>G</sup> al termine della parola.

## A

**Actual Cost:** Metrica che misura il budget speso fino al momento del calcolo.

**AI/IA:** Sigla per *Artificial Intelligence* (o Intelligenza Artificiale), identifica una disciplina che studia come simulare il pensiero umano con l'utilizzo di sistemi informatici

**Agile:** Approccio alla gestione di un progetto software. Prevede: la consegna frequente di software funzionante, la collaborazione e la comunicazione continua, l'adattamento continuo ai cambiamenti e l'auto-organizzazione dei team. L'approccio agile si basa su una serie di *framework* e metodologie specifiche, come Scrum.

**Amministratore:** Utente autenticato con la possibilità di poter effettuare tutte le operazioni eseguibili nell'applicazione .

**Analisi dei Requisiti:** Attività nella quale vengono raccolte le necessità di un proponente per quanto riguarda un programma software. Queste necessità vengono poi trasformate in requisiti descritti in un documento dedicato.

**API:** Sigla per *Application Program Interface*, descrive un insieme di procedure utilizzate per lo scambio di informazioni tra diversi sistemi, computer o software.

**Attore:** Viene identificato come attore chi interagisce con il sistema dall'esterno. Questo può essere una tipologia di utente o un sistema esterno.

## B

**Backlog:** Insieme di task ancora da completare. Per quanto riguarda il *framework* Scrum si ha: *product backlog* (lista di funzionalità da implementare nel prodotto) e *sprint backlog* (selezioni di elementi del *product backlog* che verranno trattati durante uno sprint).

**Branch:** Nel contesto di Git, un "ramo" di sviluppo indipendente in cui sviluppare una singola funzione.

**Brainstorming:** Tecnica di generazione di idee in cui i partecipanti propongono liberamente soluzioni a un problema senza giudizi preliminari.

**Browser:** Applicazione utilizzata per la navigazione di risorse nel *web*.

**Budget Variance:** Metrica che misura la differenza tra budget preventivato e budget utilizzato in fase di completamento del prodotto.



## C

**Capitolato:** Documento che specifica i bisogni del proponente. In questo documento sono espresse caratteristiche e requisiti che il prodotto consegnato deve o può avere.

**Caso d'uso:** Uno scenario che descrive il comportamento che uno o più attori hanno con il sistema rendendo esplicite le interazioni tra loro.

**ChatGTP:** Un software creato dall'azienda *Open AI* basato su intelligenza artificiale. Il suo obiettivo è quello di poter effettuare conversazioni con un utente umano e rispondere a richieste espresse in linguaggio naturale.

**Code Coverage:** Metrica che misura la percentuale di codice che viene eseguito durante i test.

**Committente:** Colui che ordina un lavoro o una prestazione. In questo caso lo sviluppo di un software.

**Comunicazione sincrona:** Tipo di comunicazione in cui gli interlocutori comunicano in tempo reale, come le chiamate telefoniche o le videoconferenze.

**Comunicazione asincrona:** Tipo di comunicazione in cui gli interlocutori comunicano senza essere contemporaneamente presenti, come le email o i messaggi di testo.

**Copertura dei Requisiti:** Metrica che misura la percentuale di requisiti soddisfatti .

**Cost Variance:** Metrica che misura la differenza tra budget utilizzabile e budget utilizzato.

**Credenziali:** Combinazione di dati che permettono ad un utente di identificarsi. Di norma si tratta di nome utente e password.

**CSS:** Sigla per *Cascading Style Sheets* (fogli di stile a cascata), si tratta di un linguaggio utilizzato per applicare formattazione a pagine HTML utilizzate in siti internet e applicazioni web.

## D

**Database:** Detta anche base di dati, si tratta di una collezione di dati organizzata e accessibile in maniera elettronica.

***Dashboard:*** Detto anche cruscotto, si tratta di una rappresentazione grafica dei valori importanti riguardo l'andamento di un progetto. Questa permette una visione rapida dei valori scelti.

***Design pattern:*** Soluzione progettuale generale e riutilizzabile a un problema comune nell'ambito dello sviluppo del *software*.

***Discord:*** Piattaforma di comunicazione vocale e testuale orientata alla comunità, spesso utilizzata da giocatori e gruppi di discussione.

## E

***Earned Value***: Metrica che misura il valore prodotto e ottenuto fino a quel momento.

***Efficienza***: Misura della capacità di un sistema di ottenere risultati desiderati con il minimo spreco di risorse.

***Estimated At Completion***: Metrica che misura il costo monetario che si prevede di raggiungere al completamento del progetto.

***Estimated To Completion***: Metrica che misura il costo monetario necessario per raggiungere il completamento del progetto.

## F

**Failure Density:** Metrica che misura la percentuale di test falliti dal prodotto software.

**File:** Traducibile come archivio, indica un contenitore di dati che risiede in un supporto di archiviazione digitale.

**Form:** Traducibile come modulo, indica un interfaccia utente che permette l'inserimento e l'invio di dati ad un sistema tramite elementi che compongono l'interfaccia stessa (es. scelta multipla, campo di testo, pulsante, ecc...).

**Formato:** Si tratta di una convenzione adottata per la lettura e scrittura dei contenuti di un *file*.

**Framework:** Architettura logica di supporto, che solitamente implementa un particolare design pattern, e funge da struttura per la progettazione e realizzazione di un software.

## G

**Git:** *Software* creato per la gestione del codice sorgente ed il tracciamento delle modifiche apportate ad esso.

**GitHub:** Si tratta di un servizio creato per poter ospitare progetti che utilizzano git.

# H

**HTML:** Sigla per *HyperText Markup Language* (Linguaggio di marcatura d'ipertesto). Si tratta di un linguaggio utilizzato per definire la struttura di un documento ipertestuale visualizzato da un browser.

# I

**Implementazione:** Fase del ciclo di vita del *software* in cui il codice viene sviluppato e integrato nel sistema.

**Incrementi:** Aggiunte progressive o miglioramenti a un prodotto o a un sistema.

**Indice *Gulpease*:** Metrica che misura la leggibilità di un testo.

**Input:** Termine inglese che ha come significato l'atto di immettere dati o informazioni, tramite una periferica adibita a tale scopo, verso un sistema che li elabori.

**Issue:** Problema o richiesta di miglioramento registrata in un sistema di tracciamento degli errori o di gestione dei problemi.

## **J**

**Jira:** Strumento software per la gestione di un progetto. Permette l'assegnazione di compiti, ed il tracciamento delle attività del progetto.



**K**

## L

***LaTeX***: Linguaggio di marcatura per la preparazione di testi. Utilizzato dal gruppo per la redazione dei documenti di progetto.

**Libreria**: Insieme di funzioni e/o strutture dati predefinite e predisposte all'utilizzo da altri programmi software.

**Linguaggio naturale**: Modalità di espressione utilizzata dagli essere umani.

**LLM**: Acronimo per *Large Language Model* (modello linguistico di grandi dimensioni), si tratta di un modello utilizzato per la comprensione e generazione del linguaggio naturale.

**Login**: In Italiano definito come accesso, o meglio il processo di identificazione di un utente nel momento in cui entra in un sistema informatico.

**Logout**: indentifica il processo di uscita di un utente da un sistema informatico.

# M

**Main branch:** Ramo principale di sviluppo. Tutti i rami figli dovranno essere uniti (operazione di *merge*) a questo una volta terminata l'attività sui branch figli.

**Merging:** Nel contesto di Git, operazione tramite la quale si riunisce un branch figlio a quello principale (o padre).

**Milestone:** Traducibile come "Pietra miliare". Un punto nel tempo che decreta il raggiungimento di obiettivi prefissati.

**Modello:** Un programma utilizzato nell'ambito dell'intelligenza artificiale, addestrato su un gruppo di dati, utilizzato per riconoscere alcuni pattern e effettuare decisioni senza intervento umano.

**MPC:** Model Predictive Control, ossia un modello basato su vincoli utilizzato per controllare i processi di un sistema.

**MPD:** Mean Percentage Difference, indica la media percentuale di errore tra i valori previsti e i valori effettivi dati dal programma.

## N

***Non-Calculated Risks:*** Metrica che misura gli eventi non programmati durante lo svolgimento del progetto che hanno portato ad un ritardo nello sviluppo.

**Norme di Progetto:** Documento che raccoglie le norme e le procedure che il gruppo di sviluppo deve utilizzare durante il ciclo di vita del *software*.

O

## P

***Passed Test Percentage:*** Metrica che misura la percentuale di test correttamente superati dal prodotto.

**Password:** Traducibile come "codice di accesso", consiste in una sequenza di caratteri alfanumerici e simboli, utilizzata da un utente per l'accesso esclusivo ad un risorsa informatica.

***PoC:*** Sigla per *Proof of Concept*, si tratta di un *software* che il gruppo di sviluppo utilizza per verificare l'applicazione delle tecnologie selezionate alle necessità espresse dal proponente.

**Piano di Progetto:** Documento nel quale vengono analizzati i rischi, la pianificazione e il preventivo dei costi, in modo da lavorare nel migliore dei modi.

**Piano di Qualifica:** Documento nel quale si specificano gli obiettivi qualitativi e quantitativi di prodotto e di processo, specificando anche quali e quanti test eseguire.

***Planned Value:*** Metrica che misura l'attività lavorativa impiegata fino a quel momento.

***Product Baseline:*** Conosciuta anche come PB. È la seconda revisione di avanzamento prevista dall'attività didattica. Il gruppo deve aver acquisito un'alta padronanza delle tecnologie utilizzate, avendo soddisfatto almeno tutti i requisiti obbligatori.

***Prompt:*** Un testo in linguaggio naturale interpretabile dal un modello di intelligenza artificiale.

***Prompting:*** Tecnica per la generazione di un testo in linguaggio naturale, interpretato poi da un'intelligenza artificiale.

**Proponente:** Figura che propone un lavoro o un progetto per la sua realizzazione.

***Python:*** Linguaggio di programmazione ad alto livello, orientato agli oggetti.

## Q

***Quality Metric Satisfied***: Metrica che misura la percentuale di metriche definite nel Piano di Qualifica soddisfatte.

***Query SQL***: Traducibile come "interrogazione". Si tratta di un comando inserito da un utente per la richiesta di dati o modifiche su dati presenti in un *database* (in questo caso di tipo SQL).

## R

**Raggiunta dell’Obiettivo:** Metrica che misura il numero di click che l’utente deve impiegare per ottenere la query SQL ritornata dal programma.

**Repository:** Si tratta di un ambiente dove vengono ”depositati” in modo digitale dati ed informazioni che compongono il progetto.

**Requirements Stability Index:** Metrica che misura la necessità di modificare i requisiti dopo averli definiti. Requisiti più stabili non hanno bisogno di essere modificati.

**Requirements & Technology Baseline:** Conosciuta anche come RTB. Prima revisione di avanzamento prevista dall’attività didattica. Il gruppo deve aver eseguito lo studio dei requisiti e aver studiato le tecnologie e dimostrato la loro possibilità di utilizzo tramite un *Proof of Concept*.

**Requisito:** Funzionalità e/o caratteristica di un sistema e/o le sue componenti. Questo viene descritto specificando il suo comportamento, i suoi ingressi, le sue uscite ed il suo obiettivo.



## S

***Satisfied Obligatory Requirements:*** Metrica che misura la percentuale di requisiti obbligatori soddisfatti dal prodotto.

***Schedule Variance:*** Metrica che misura l'anticipo o il ritardo rispetto a quanto pianificato.

***Sinonimi:*** Relazione tra due parole con lo stesso significato. Utilizzato in questo progetto per individuare il soggetto della richiesta da parte dell'utente.

***Software:*** Insieme di programmi, dati e istruzioni che consentono al *computer* di eseguire determinate funzioni o compiti.

***Software House:*** Azienda specializzata nello sviluppo di *software* e nella fornitura di servizi informatici.

***Sprint:*** Periodo di tempo prefissato entro il quale il lavoratore produce dei risultati documentati.

***SQL:*** Sigla per *Structured Query Language*, si tratta di un linguaggio standardizzato per interrogare e amministrare un *database* basato su modello relazionale.

***Strategy:*** Design pattern che data una famiglia di algoritmi che implementa una precisa funzionalità, permette all'applicazione di utilizzare dinamicamente l'algoritmo corretto.

***Struttura Database:*** Definizione della composizione di un *database* definendo le sue tabelle e la loro composizione.

## T

**Tabella:** Struttura utilizzata da un *database* basato su modello relazionale per la gestione dei dati. Questa è composta da dei campi che indicano il tipo di dato che esso contiene e altri dati utili all'utilizzo e l'amministrazione della tabella.

**Task:** Traducibile come "compito". Ciò che una persona deve fare in un determinato periodo di tempo.

**Telegram:** Piattaforma di messaggistica istantanea e di comunicazione basata su *cloud*.

**Ticketing:** Sistema di gestione delle richieste o dei problemi che assegna un numero di *ticket* univoco a ciascuna richiesta.

**Tutorial:** Guida passo-passo che fornisce istruzioni su come fare qualcosa o su come utilizzare un *software*.

## U

**Utente:** Colui che interagisce con un prodotto *software*.

**UML:** *Unified Modeling Language*, un linguaggio di modellazione generale che fornisce concetti e strumenti applicabili in tutti i contesti.

**Unità architetturali:** Componenti fondamentali di un sistema *software* che definiscono la sua struttura e organizzazione.

## V

**Versionamento:** Realizza il "controllo di versione". Traccia la storia delle azioni fatte e i cambiamenti avvenuti nel corso del tempo.

## W

***Way of Working***: Contenuto nel documento che noi chiamiamo "Norme di progetto". Si tratta dell'insieme di regole e tecnologie, in continua evoluzione, adottate da un gruppo di lavoro.

**X**

Y

***Z***