



Norme di progetto

6Coders

`6Coders.unipd@gmail.com`

19 Novembre 2023



Registro delle Modifiche - Changelog

Ver	Data	Autore	Verificatore	Descrizione
1.1	22/11/2023	Chelhaoui Osama		Aggiunta Sezione “Automazione”
1.0	19/11/2023		Lovato Yuri	Approvazione
0.5	14/11/2023	Marchiorato Pietro Lovato Yuri Florian Edoardo		Completata sezione “Documentazione” relativa ai Processi di supporto
0.4	13/11/2023	Marchiorato Pietro Lovato Yuri Vogli Vullnet	Florian Edoardo	Completata sezione “Processi Organizzativi” e “Gestione della configurazione”. Completata sezione “Fornitura” dei “Processi primari”
0.3	12/11/2023	Bilinski Eleonora Chelhaoui Osama		Continuata sezione “Documentazione” e inserita sezione Gestione della Qualità
0.2	09/11/2023	Marchiorato Pietro Lovato Yuri	Osama lhaoui	Che- Definizione delle sezioni del documento
0.1	20/10/2023	Marchiorato Pietro	Osama lhaoui	Che- Creazione, introduzione e struttura del documento

Indice

1	Introduzione	4
1.1	Scopo del Documento	4
1.2	Scopo del Prodotto	4
1.3	Glossario	4
1.4	Note organizzative	4
2	Processi Primari	4
2.1	Fornitura	4
2.1.1	Descrizione	4
2.1.2	Scopo e aspettative	5
2.1.3	Piano di progetto	5
2.1.4	Piano di qualifica	6
2.1.5	Collaudo e rilascio	6
2.2	Sviluppo	7
2.2.1	Descrizione	7
2.2.2	Scopo e aspettative	7
2.2.3	Analisi dei requisiti	7
2.2.4	Progettazione	7
2.2.5	Codifica	7
3	Processi di Supporto	8
3.1	Documentazione	8
3.1.1	Descrizione	8
3.1.2	Scopo e aspettative	8
3.1.3	Ciclo di vita di un documento	8
3.1.4	Documenti prodotti	9
3.1.5	Struttura del documento	10
3.1.6	Prima pagina	10
3.1.7	Registro delle modifiche	10
3.1.8	Indice	10
3.1.9	Verbali	11
3.1.10	Nome dei file	11
3.1.11	Stile del testo	12
3.1.12	Norme tipografiche	12
3.1.13	Glossario	12
3.1.14	Sigle	12
3.1.15	Tabelle	12
3.1.16	Immagini	13
3.1.17	Preventivo	13
3.1.18	<i>Consuntivo_G</i>	13
3.2	Gestione della configurazione	13
3.2.1	Descrizione	13
3.2.2	Scopo e aspettative	14

3.2.3	Versionamento	14
3.2.4	Sistemi software utilizzati	14
3.2.5	Struttura repository	14
3.3	Gestione della qualità	14
3.3.1	Scopo	14
3.3.2	Piano di qualifica	15
3.3.3	Testing	15
3.3.4	Metriche	15
3.3.5	Aspettative	15
3.4	Automazione	15
3.4.1	Descrizione	15
3.4.2	Scopo	15
3.4.3	Script	16
3.5	Verifica	16
3.5.1	Descrizione	16
3.5.2	Scopo	16
3.5.3	Indicazioni	16
3.5.4	Verifica della documentazione	16
3.5.5	Verifica del prodotto	16
3.6	Validazione	17
3.6.1	Descrizione	17
4	Processi Organizzativi	17
4.1	Gestione di Processo	17
4.1.1	Coordinamento e Comunicazioni	17
4.1.2	Riunioni	17
4.2	Pianificazione	18
4.2.1	Ruoli	18
4.3	Infrastruttura	18
4.3.1	Strumenti	18
4.4	Miglioramenti	19
4.5	Formazione	19

1 Introduzione

1.1 Scopo del Documento

Con il seguente documento vengono definite tutte le norme ed i meccanismi che regolano il Way of Working. Tali direttive, pensate e sottoscritte da ogni membro del gruppo, verranno rispettate sino alla creazione del prodotto finale, al fine di lavorare in modo efficiente ed efficace.

1.2 Scopo del Prodotto

Con la crescente diffusione dell'intelligenza artificiale, si è raggiunto un punto in cui si parla di “prompt engineering” per definire la pratica di formulare richieste in modo ottimale al fine di ottenere migliori prestazioni dai sistemi di intelligenza artificiale. L'obiettivo di questo prodotto è generare un *prompt_G* in grado di guidare il sistema di intelligenza artificiale nella creazione di una *query_G SQL_G* a partire da una frase scritta in linguaggio naturale. L'algoritmo di *machinelearning_G* implementato nel prodotto, che genera il promptG, può essere sottoposto a un *fine – tuning_G* sulla struttura del database inserito dall'utente, consentendo così la creazione di *prompt_G* efficaci.

1.3 Glossario

Il documento Glossario è parte della documentazione di progetto dove è possibile reperire definizioni chiare e precise dei vocaboli non comuni utilizzati nei documenti prodotti. La presenza di un determinato vocabolo in Glossario viene segnata con la lettera G al pedice (es. *Vocabolo_G*).

1.4 Note organizzative

Il presente documento non può essere considerato versione definitiva al momento della lettura, in quanto redatto in maniera incrementale a seconda di cambiamenti operativi o presenza di esigenze differenti.

2 Processi Primari

2.1 Fornitura

2.1.1 Descrizione

Nella seguente sezione sono riportate tutte le regole, adottate dal gruppo “6Coders”, al fine di mantenere attivi e produttivi i rapporti con il *proponente_G* Zucchetti S.P.A ed evitare incomprensioni durante la realizzazione del progetto

2.1.2 Scopo e aspettative

L'obiettivo della procedura di fornitura è chiarire ogni dubbio riguardante l'intenzione del proponente, perseguendo quindi la completa comprensione delle richieste al fine di soddisfare le richieste completamente. Durante l'intero sviluppo del progetto, il gruppo 6Coders, si impegna a mantenere una comunicazione frequente con l'azienda Zucchetti S.P.A al fine di poter:

- Definire in modo univoco i bisogni ($needs_G$) del proponente;
- Ottenere dei $feedback_G$ del lavoro svolto da parte dell'azienda;
- Chiarire gli eventuali dubbi e incomprensioni che incontreremo nell'avanzamento dello sviluppo del progetto;
- Ottenere formazione sulle tecnologie proposte dall'azienda;
- Stabilire tempistiche e vincoli per la consegna del prodotto finale.

2.1.3 Piano di progetto

Il $projectplan_G$, o piano di progetto, rappresenta una tappa chiave nella fase iniziale della pianificazione, costituendo il momento cruciale in cui il progetto si definisce. Essenzialmente, si tratta di un documento ufficiale soggetto a revisioni e approvazioni, nel quale vengono delineati gli obiettivi del progetto e gli elementi indispensabili per il loro conseguimento. Un piano di progetto è caratterizzato da:

- **Piano di gestione dei rischi:** fattori che potrebbero far deragliare il progetto e un piano per come i problemi saranno identificati, affrontati e controllati, i rischi possono essere classificati in due tipi:
 - Rischi personali e organizzativi;
 - Rischi tecnologici.
- **Modello di sviluppo:** modello per lo sviluppo del progetto adottato dal gruppo;
- **Pianificazione:** viene selezionato una lista di $task_G$ da svolgere per il completamento delle attività, definendone una tempistica di terminazione;
- **Preventivo:** sommario schematizzato dei costi complessivi e del piano temporale richiesto per il progetto, con ulteriore dettaglio suddiviso per ciascuna fase;
- **Consuntivo:** registro dettagliato dell'evoluzione rispetto alla stima iniziale dei costi.

2.1.4 Piano di qualifica

Il presente documento costituisce una chiara delineazione delle responsabilità, compiti e attività assegnate al Verificatore nel contesto del progetto. Tali compiti sono essenziali per assicurare la qualità del prodotto finale, e si articolano attraverso l'impiego di metodi e procedure di verifica e validazione. In sostanza, il Verificatore svolge un ruolo cruciale nell'assicurare la conformità del lavoro svolto alle specifiche del progetto, implementando rigorose tecniche di controllo e validazione al fine di garantire l'ottenimento di un risultato finale di elevata qualità. Più in specifico un piano di qualifica è composto da:

- **Obiettivi di Qualità:** definizione chiara degli obiettivi di qualità che il progetto o il prodotto dovrebbero raggiungere;
- **Processi di Garanzia della Qualità:** procedure e metodologie che verranno implementate per garantire la qualità durante tutte le fasi del progetto;
- **Criteri di Accettazione:** standard e criteri che saranno utilizzati per valutare se il prodotto soddisfa i requisiti di qualità;
- **Revisione e Aggiornamento:** procedure per la revisione e l'aggiornamento del piano di qualifica in risposta a modifiche nel progetto o nelle condizioni.

2.1.5 Collaudo e rilascio

La fase di collaudo e rilascio è l'ultima fase del ciclo di sviluppo di un progetto e coinvolge l'approvazione finale del prodotto o del sistema prima che venga messo a disposizione del committente. Questa fase è critica per garantire che il risultato del lavoro soddisfi le specifiche del progetto e le aspettative del committente. Se il collaudatore riscontra delle anomalie dovranno essere risolte, altrimenti potrà essere consegnato al committente il codice sorgente e la documentazione relativa al prodotto finale. Le principali attività coinvolte in questa fase sono:

- **Collaudo del prodotto:**
 - **Test funzionale:** verifica delle funzionalità del prodotto per assicurarsi che soddisfi i requisiti specificati;
 - **Test di presentazione:** valutazione delle prestazioni del sistema in condizioni diverse;
 - **Test di sicurezza:** verifica le misure di sicurezza.
- **Collaudo utente:** coinvolgimento degli utenti finali per valutare l'usabilità e la *userexperience_G* del prodotto;
- **Risoluzione di problemi e *bug_G*:** identificazione e risoluzione di eventuali problemi, *bug_G* o errori emersi durante il collaudo;
- **Approvazione e accettazione:** il committente verifica se il prodotto soddisfa i requisiti concordati e se è pronto per il rilascio;

- **Rilascio del prodotto:** una volta ottenuta l'approvazione del committente, il prodotto è ufficialmente rilasciato e reso disponibile per l'utilizzo.

2.2 Sviluppo

2.2.1 Descrizione

Lo sviluppo di un software include tutti i processi, dalla concezione del software alla sua manifestazione finale, in maniera strutturata e pianificata.

2.2.2 Scopo e aspettative

Lo scopo del processo di sviluppo è identificare le mansioni e le attività che il team deve svolgere per la creazione del software richiesto. Affinché il prodotto rispetti le aspettative richieste deve essere:

- In accordo con le richieste del proponente;
- Conforme agli obiettivi di sviluppo.

2.2.3 Analisi dei requisiti

L'obiettivo della fase di analisi dei requisiti è l'identificazione, la raccolta e la documentazione dei requisiti del software. Tali requisiti possono essere:

- **Funzionali:** descrivono le funzioni che il software deve essere in grado di eseguire;
- **Non funzionali:** descrivono le proprietà del software come la sicurezza, l'usabilità e le prestazioni.

Notare che la descrizione specifica cosa il sistema dovrà fare, non il come. L'insieme delle descrizioni vengono raccolti in documenti noti come documenti di specifica.

2.2.4 Progettazione

Successiva alla fase di analisi troviamo la progettazione che si occupa di definire come soddisfare i requisiti individuati. Viene definita l'architettura del software, i moduli che lo compongono e come questi interagiscono tra loro e le interfacce con l'utente. La progettazione può essere suddivisa in due fasi principali: la progettazione architetturale e la progettazione di dettaglio. Durante la progettazione architetturale si definisce la struttura generale del software; in quella di dettaglio si definiscono i dettagli implementativi dei singoli moduli.

2.2.5 Codifica

La codifica è il processo di implementazione del software. Ciò avviene tramite la scrittura di codice seguendo le linee guida di codifica e le specifiche di progetto.

3 Processi di Supporto

3.1 Documentazione

3.1.1 Descrizione

La presente sezione descrive le regole per la scrittura e verifica dei documenti prodotti per il progetto.

3.1.2 Scopo e aspettative

La presente sezione ha come scopo quello di fornire le norme e i vincoli che regoleranno tutta la documentazione prodotta durante il ciclo di vita del progetto, in modo tale da fornire una struttura comune e coerente per tutti i documenti. Le aspettative che il gruppo intende soddisfare sono:

- Fornire una struttura comune e chiara per i documenti in modo tale da facilitare sia la stesura che la lettura;
- Fornire delle norme precise per ogni aspetto della struttura dei documenti, in modo che ogni membro possa lavorare asincronamente ed autonomamente ad essi.

3.1.3 Ciclo di vita di un documento

Il ciclo di vita di un documento ne descrive le sue fasi per la realizzazione, nel dettaglio:

- **Pianificazione:** vengono analizzati i requisiti e vengono organizzate le varie parti del documento;
- **Impostazione:** viene creata la struttura del documento, in modo sincrono il responsabile crea una *issueG* per ogni sezione del documento, in modo da suddividere in modo logico il lavoro per il completamento del documento;
- **Realizzazione:** ad ogni *issueG* sono assegnati uno o più membri del *teamG*, il componente, una volta selezionata la *issueG* di suo interesse, comincia a sviluppare la sezione corrispondente.
- **Verifica:** la sezione, una volta terminata, viene posta a verifica da parte del verificatore, il quale ne controlla l'idoneità rispetto ai requisiti attesi;
- **Approvazione:** una volta verificato l'intero documento, il responsabile attualmente in carica, dopo un'attenta analisi, decide se approvare il documento con successiva aggiunta al *repositoryG* relativa allo sviluppo, oppure di rifiutare il documento il quale sarà posto ad ulteriore revisione.

3.1.4 Documenti prodotti

I documenti prodotti sono di due tipi:

- **Documenti Formali:** seguono delle regole specifiche per struttura, formato e linguaggio contenuto, vengono redatti secondo standard definiti. Sono soggetti a revisione e approvazione da parte del responsabile prima di essere considerati ufficiali e vincolanti. Regolano l'operato del gruppo e lo sviluppo delle attività durante il ciclo di vita del software. Le caratteristiche di questi documenti sono:
 - **Struttura standard:** i documenti formali seguono regole e standard definiti riguardo la loro struttura;
 - **Responsabilità e tracciabilità:** chiare responsabilità sono associate alla creazione, verifica, approvazione e archiviazione di questi documenti. Hanno una tracciabilità definita in modo da identificare chi ha effettuato quali modifiche e quando;
 - **Approvazione:** approvazione della versione da parte del responsabile.

Se un documento ha più di una versione, si considera sempre quella più recente approvata dal responsabile. I documenti formali possono essere di due tipi:

- **Interni:** utili al team di sviluppo in quanto riguardano le dinamiche interne al gruppo, ma poco utili al committente e proponente;
- **Esterni:** interessano direttamente il committente e proponente.

Sono documenti formali:

- **Norme di progetto:** stabilisce le linee guida, le procedure, le convenzioni e le regole specifiche che il tema 6Coders dovrà seguire durante lo svolgimento del progetto (Documento interno);
- **Glossario:** elenco di termini tecnici, acronimi, abbreviazioni e concetti specifici utilizzati nel contesto del progetto (Documento esterno);
- **Piano di progetto:** delinea la struttura, le attività, le risorse, le tempistiche e le strategie per la gestione e l'esecuzione del progetto (Documento esterno);
- **Piano di qualifica:** stabilisce le strategie, le metodologie, le attività e le risorse necessarie per garantire e controllare la qualità del progetto (Documento esterno);
- **Analisi dei requisiti:** vi è l'identificazione, la raccolta, e analisi dei bisogni, delle funzionalità e delle caratteristiche che il prodotto deve possedere per soddisfare le esigenze del committente (Documento esterno);
- **Manuale utente:** fornisce istruzioni, guide e informazioni dettagliate sull'utilizzo del prodotto (Documento esterno);
- **Specifica architetturale:** fornisce una descrizione dettagliata dell'architettura del prodotto. Descrive in modo approfondito la struttura, i componenti, i moduli, le interazioni e le relazioni tra le varie parti del sistema (Documento esterno).

- **Documenti Informali:** sono tutti quei documenti non ancora approvati dal responsabile e non soggetti a *versionamento_G* nella repository, sono meno strutturati e senza seguire uno standard preciso, vengono utilizzati per comunicazioni interne, appunti personali, *brainstorming_G* ed idee preliminari. Questi documenti sono contenuti nel *GoogleDrive_G* condiviso al team.

3.1.5 Struttura del documento

Ogni documento contiene le seguenti componenti:

- La prima pagina che specifica nome, data e versione del documento;
- Il registro delle modifiche per il versionamento e tenere traccia delle modifiche apportate;
- Il contenuto vero e proprio del documento la quale struttura può variare in base alla tipologia del documento stesso;

3.1.6 Prima pagina

La prima pagina del documento è formata da:

- Logo del gruppo;
- Nome del documento e data di ultima approvazione;
- Nome del gruppo e relativa email;
- Logo dell'università di Padova e relative informazioni sul corso.

3.1.7 Registro delle modifiche

Ogni documento contiene una tabella delle modifiche apportate al documento, nella quale viene tenuta traccia dei cambiamenti significativi del documento nel suo ciclo di vita, in particolare la tabella riporta:

- Versione del documento;
- Data di versionamento;
- Nominativo dell'autore e del verificatore;
- Descrizione della modifica apportata al documento.

3.1.8 Indice

I documenti presentano un indice dei contenuti per facilitare la navigazione, è posizionato appena dopo il registro delle modifiche. I verbali sia interni che esterni non presentano indice.

3.1.9 Verballi

Registro scritto che riporta in modo dettagliato le discussioni, le decisioni, gli accordi e le azioni intraprese durante una riunione, una conferenza o un incontro formale, un verbale è così organizzato:

- **Informazioni e partecipanti:**

- Tipo di riunione: può essere interna o esterna;
- Luogo;
- Data e ora;
- Responsabile;
- Verbalizzante;
- Verificatore;
- Partecipanti interni;
- Partecipanti esterni: se il verbale è interno viene lasciato vuoto questo campo.

- **Ordine del giorno:** riporta gli argomenti trattati e discussi durante la riunione;
- **Svolgimento della riunione**, solo per i verbali esterni: viene presentato lo svolgimento della riunione con il proponente;
- **Quesiti posti al proponente**, solo per i verbali esterni: tabella che riporta le domande e le relative risposte effettuate al proponente;
- **Todo / Ongoing:** tabella nella quale vengono descritti brevemente i taskG o le modifiche da apporre al progetto discusse durante la riunione, nel dettaglio:
 - Il campo *taskG* descrive l'azione o la modifica che verrà effettuata;
 - Il campo “assignees/assegnatari”G sono le persone del team al quale viene assegnato lo sviluppo dello taskG corrispondente.

3.1.10 Nome dei file

Il nome dei documenti è diviso nelle seguenti sezioni:

- Nome del gruppo
- Titolo effettivo che rappresenta il contenuto, scritto con lettera iniziale maiuscola. Se il titolo presenta più parole non vengono separate da spazi ma vengono scritte tutte attaccate e ogni parola inizia con la lettera maiuscola;
- Ultima Versione approvata del documento.

Ogni sezione è separata dal simbolo “_”. Il titolo dei verbali invece è formato da una “v” iniziale seguita da una sigla che può essere “INT” se il verbale è interno, oppure “EXT” se il verbale è esterno. Viene inserita la data nel formato **ggmmaa** ed è separata dal resto tramite il simbolo “_”. (e.g. vEXT_261023), inoltre non viene indicato il versionamento.

3.1.11 Stile del testo

Gli stili adottati dai nei documenti sono:

- Grassetto: per titoli, sottotitoli e per parole ritenute particolarmente importanti;
- Maiuscolo: per gli acronimi;
- Sottolineato: per i link presenti nei documenti.

3.1.12 Norme tipografiche

- Gli elenchi puntati o numerati iniziano sempre per la lettera maiuscola e terminano con punto e virgola, tranne l'ultimo punto che termina con punto fermo;
- Un elenco puntato utilizza il simbolo ●, un successivo annidamento il simbolo - e un altro ancora *;
- Gli annidamenti degli elenchi numerati sono ordinati con numeri arabi divisi da un punto fermo.

3.1.13 Glossario

Raccoglie e definisce in modo chiaro e conciso una serie di termini, parole o acronimi utilizzati nei documenti. Questo documento fornisce definizioni specifiche e significati precisi per ciascun termine, contribuendo a standardizzare e chiarire il linguaggio utilizzato. Ogni membro del team può aggiungere un termine, con relativa definizione, all'interno del glossario. Ciascun termine presente nel glossario verrà segnalato, almeno una volta, con la lettera G a pedice della parola interessata.

3.1.14 Sigle

- Sigle per i nomi dei documenti:
 - Glossario: G
- Sigle per le revisioni:
 - Requirements and Technology Baseline: RTB
 - Product Baseline: PB
 - Customer Acceptance: CA

3.1.15 Tabelle

Le tabelle presenti all'interno dei documenti hanno il seguente formato:

- Didascalia: che ne descrive il contenuto della tabella, ha un allineamento centrale nella pagina e il font è in grassetto;
- Righe: hanno lo sfondo di colore alternato per semplificare la lettura della tabella;
- Colonne: la prima riga di ogni colonna ne descrive il contenuto ed il font è in corsivo.

3.1.16 Immagini

Le immagini sono centrate orizzontalmente e accompagnate da una didascalia. I diagrammi UML all'interno dei documenti sono inseriti tramite immagini.

3.1.17 Preventivo

Per ogni periodo pianificato viene indicato:

- I ruoli che opereranno;
- Preventivo orario:
 - Tabella con i ruoli per colonna che descrive per ogni membro il numero di ore produttive preventivate in un determinato ruolo;
 - Un grafico a torta che segna le percentuali di ore per ogni ruolo.
- Preventivo economico:
 - Tabella che descrive per ogni ruolo il numero di ore complessive e il costo complessivo preventivato;
 - Un grafico a torta che segna le percentuali di costo per ogni ruolo.

3.1.18 *Consuntivo_G*

Per ogni periodo pianificato viene indicato:

- Consuntivo orario: tabella con i ruoli per colonna che descrive per ogni membro il numero di ore produttive che ha effettivamente eseguito in un determinato ruolo;
- Consuntivo economico: tabella che descrive per ogni ruolo il numero di ore complessive eseguite e il costo complessivo;
- Conclusioni: sezione in cui viene fatta un'analisi fra le differenze tra preventivo e consuntivo a scopo di individuare eventuali problemi nel periodo appena concluso.

3.2 Gestione della configurazione

3.2.1 Descrizione

La presente sezione descrive come il gruppo intende gestire ed organizzare la produzione di risorse. Il processo di gestione della configurazione ha lo scopo di organizzare la produzione di documentazione e di codice in maniera sistematica, ordinata e standardizzata.

3.2.2 Scopo e aspettative

Le aspettative che il gruppo intende soddisfare sono:

- Registrazione delle modifiche apportate alle risorse, con la possibilità di effettuare *rollback_G*;
- Uniformazione degli strumenti utilizzati;
- Gestione semplice ed efficace degli errori riscontrati.

3.2.3 Versionamento

Il gruppo ha deciso di applicare il seguente sistema di versionamento, per poter ricostruire la storia di un determinato documento/risorsa.

[X].[Y]

[X] corrisponde alla Versione di Produzione (pubblicabile in repository pubblica), soggetta ad Approvazione da parte di un soggetto appositamente predisposto. [Y] corrisponde all'incremento di versione per modifiche ed integrazioni minori, sottoposte a verifica da parte dei Verificatori.

3.2.4 Sistemi software utilizzati

Per la creazione e la gestione della repository viene utilizzata la piattaforma *GitHub_G*, che a sua volta rende possibile l'integrazione con *GIT_G* per la gestione del sistema di versioning. I task vengono gestiti, assegnati e suddivisi utilizzando le Issues e le *PullRequests_G*.

3.2.5 Struttura repository

La repository pubblica dove sono presenti tutti i documenti approvati è accessibile al seguente indirizzo: 6coders.github.io. Per l'organizzazione della repository il gruppo si è basato sul sistema di GitHub Flow, organizzato a rami e di seguito illustrato:

- Ramo **main** (Principale) è presente la versione [X], approvata dei documenti;
- Rami secondari utilizzati per la redazione dei vari paragrafi della documentazione che, una volta completati, vengono uniti al ramo principale dal Verificatore attraverso l'operazione di **merge**.

3.3 Gestione della qualità

3.3.1 Scopo

Per gestione della qualità si intende l'insieme di processi e attività, effettuate durante lo sviluppo di un progetto, per garantire che la qualità dei DeliverableG sia in accordo con gli obiettivi e i requisiti prestabiliti. Gli obiettivi che il gruppo ha stabilito sono i seguenti:

- Analizzare i requisiti del richiedente, assicurandosi che siano ben definiti;

- Ideare un piano di progetto che stabilisce stime realistiche dei tempi e delle risorse;
- Realizzare il progetto seguendo il piano stabilito, in conformità con i tempi e le risorse.

3.3.2 Piano di qualifica

Per garantire che la qualità sia in accordo con i requisiti del progetto si utilizza un Piano di Qualifica, un documento dove il gruppo si impegna a definire le metriche di valutazione del prodotto, la quantità e la qualità dei test e documentare l'esito ottenuto da tali test.

3.3.3 Testing

3.3.4 Metriche

Per definire le metriche il gruppo ha definito la seguente nomenclatura:

M[Tipologia][Acronimo]

Dove:

- **Tipologia** indica il tipo della metrica:
 - PC: processo;
 - PD: prodotto;
- **Acronimo** che indica il nome della metrica utilizzata.

3.3.5 Aspettative

Con una gestione della qualità adatta, il gruppo si aspetta di:

- Riuscire ad individuare gli errori e risolverli in maniera efficace e tempestiva;
- Mantenere aggiornata la documentazione;
- Migliorare la collaborazione all'interno del gruppo;
- Ottenere il prodotto finale richiesto.

3.4 Automazione

3.4.1 Descrizione

L'aspetto dell'automazione mira a ottimizzare i processi, migliorare l'efficienza di sviluppo e ridurre gli errori umani.

3.4.2 Scopo

Il principale utilizzo dell'automazione verterà sulle procedure di test e verifica del codice. I test automatizzati possono essere eseguiti in modo rapido e ripetibile, permettendo di identificare tempestivamente le eventuali problematiche o regressioni che si potrebbero riscontrare.

3.4.3 Script

Al fine di testare il codice prodotto, si realizzano unit test e successivamente una *pipeline*_G CI/CD con Github Actions. Github Actions ha tre livelli:

- **workflows** è un file che contiene il processo di automazione. Un workflow può gestire più **jobs**.
- i **jobs** sono un set di istruzioni che sono eseguite sullo stesso runner. Un runner è un sistema operativo dove il **job** dovrebbe essere eseguito.
- gli **steps** sono una sequenza di processi che devono essere eseguiti come parte del **job**.

Tale pipeline successivamente può essere eseguita effettuando un push su codice presente nella repository GitHub, che gestirà il resto del lavoro.

3.5 Verifica

3.5.1 Descrizione

Il processo di verifica prevede un'attività di controllo necessaria a stabilire se un determinato prodotto o documento è stato realizzato in maniera corretta, completa ed in linea con le richieste.

3.5.2 Scopo

La verifica prevede l'analisi di ogni processo e sottoprocesso e viene effettuata ogni qualvolta si presenta un cambiamento di stato.

3.5.3 Indicazioni

Il processo di verifica va attuato seguendo procedure chiare e definite, secondo criteri validi ed affidabili.

3.5.4 Verifica della documentazione

L'attività di verifica della documentazione segue le fasi seguenti:

- Controllo della correttezza dei contenuti del documento;
- Controllo della corretta applicazione delle regole sintattiche e ortografiche;
- Controllo stile e regole tipografiche.

3.5.5 Verifica del prodotto

Al fine di minimizzare gli errori il codice viene controllato mediante attività di analisi statica e dinamica. [DA COMPLETARE QUANDO SCEGLIAMO LE TECNOLOGIE UTILIZZATE].

3.6 Validazione

3.6.1 Descrizione

Per validazione si intende l'insieme degli incontri effettuati con la Proponente e successivamente il Committente per verificare che lo stato di avanzamento del progetto o il prodotto finale presentato sia conforme ai requisiti minimi individuati in fase di analisi e rispetti le aspettative in termini di tempo pianificato e costo preventivato.

4 Processi Organizzativi

4.1 Gestione di Processo

4.1.1 Coordinamento e Comunicazioni

I canali di comunicazione scelti dai membri del gruppo per le comunicazioni interne sono:

- *Discord_G* per le comunicazioni sincrone durante gli incontri;
- Telegram per le comunicazioni asincrone (comunicazioni urgenti, informali, sondaggi e organizzazione incontri).

Ogni membro del gruppo deve monitorare attentamente la Project Board su GitHub per seguire lo stato di avanzamento del progetto e verificare/aggiornare lo stato delle task assegnate. Le comunicazioni esterne con proponente e committente avvengono tramite posta elettronica in modalità asincrona e attraverso Zoom per gli incontri sincroni in tempo reale. Il responsabile di progetto in carica si occupa di organizzare e gestire, a nome del gruppo, le comunicazioni esterne.

4.1.2 Riunioni

Le riunioni interne coinvolgono, per quanto possibile, tutti i membri del gruppo. Possono essere di due tipologie:

- Ordinaria: si tratta di riunioni programmate con una cadenza quindicinale;
- Straordinaria: si tratta di riunioni non pianificate che si svolgono per necessità imminenti non previste.

Le riunioni esterne possono coinvolgere i membri del gruppo, l'azienda proponente e i docenti (committenti). Per ottimizzare il tempo a disposizione, è necessario preparare una scaletta dei punti da discutere la quale potrà essere condivisa con i partecipanti esterni prima dell'incontro. In seguito, è necessario ottenere l'approvazione da parte dei partecipanti esterni coinvolti prima della pubblicazione dei verbali relativi. Il responsabile di progetto in carica ha il compito di organizzare e moderare le riunioni, garantendo un tempestivo aggiornamento ai membri del gruppo i quali, per cause di forza maggiore, sono stati impossibilitati a prenderne parte. Ad ogni riunione segue il relativo verbale redatto a turno dai componenti del gruppo.

4.2 Pianificazione

4.2.1 Ruoli

Ogni membro del gruppo, a rotazione, ricopre in modalità e tempistiche equivalenti uno tra i seguenti ruoli secondo quanto preventivato:

- **Responsabile:** ha conoscenza dello stato di avanzamento del progetto, si occupa di gestire e coordinare lo svolgimento di tutti i processi. Ha il compito di assegnare i task e gestire eventuali problematiche o ritardi incontrati;
- **Amministratore:** garantisce che lo stato di avanzamento progredisca in maniera efficiente ed efficace. Si occupa della gestione dell'infrastruttura e degli strumenti/piattaforme utilizzati;
- **Analista:** è un ruolo chiave che caratterizza le prime fasi del progetto. Ha il compito di comprendere e tradurre in requisiti le necessità e le richieste avanzate dalla proponente. È il principale contributore del documento “Analisi dei Requisiti”;
- **Progettista:** ha il compito di modellare il problema seguendo e rispettando i requisiti presenti nel documento “Analisi dei Requisiti”. Se necessario deve approfondire conoscenze tecniche e ricercare nuovi strumenti tecnologici utili nello specifico ambito di applicazione;
- **Programmatore:** ha il compito implementare quanto emerso nella fase di progettazione, in particolare possiede competenze tecniche, ma autonomia e responsabilità circoscritte;
- **Verificatore:** ha il compito di controllare che quanto prodotto rispetti i requisiti di qualità attesi.

4.3 Infrastruttura

4.3.1 Strumenti

- **Gmail:** servizio di posta elettronica utilizzato per l'invio e la ricezione di comunicazioni ufficiali da parte del gruppo;
- **Telegram:** piattaforma utilizzata per la comunicazione asincrona del gruppo;
- **Discord:** strumento utilizzato per la comunicazione sincrona e la condivisione di risorse utili;
- **GitHub:** sistema di versionamento e gestione della repository pubblica di sviluppo e documentale;
- **Zoom:** strumento utilizzato per la comunicazione sincrona con i partecipanti esterni durante le riunioni;
- **Suite Google (Drive, Docs, Spreadsheet):** strumenti utilizzati per la produzione dei documenti ufficiali rilasciati dal gruppo.

4.4 Miglioramenti

Non avendo esperienza in questo ambito i documenti prodotti saranno soggetti a continuo miglioramento, con l'obiettivo di correggere eventuali errori. Il documento Piano di Qualifica, in particolare nella sezione Valutazioni per il miglioramento, descrive i problemi principali riscontrati dal gruppo, con opportune descrizioni e soluzioni.

4.5 Formazione

Ogni componente del gruppo deve studiare e comprendere in maniera autonoma gli strumenti e le tecnologie utilizzati per lo svolgimento del progetto (di seguito riportati), in maniera tale da garantire un livello di avanzamento organizzato, e simultaneo.

- [Git](#);
- [GitFlow](#);
- Python (PyThorc);
- Flask;
- [LLM e Prompt-Engineering](#);
- Libreria “Transformers” di Hugging Face.