



<CodeHex16>

unipd.codehex16@gmail.com

Norme di Progetto

| | |
|-------------|------------|
| Data | 12/11/2024 |
|-------------|------------|

| | |
|-----------------|-------|
| Versione | 0.4.0 |
|-----------------|-------|

Sommario

Norme di progetto

Ruoli

Matteo Bazzan

Luca Ribon

Francesco Fragonas

Gabriele Magnelli

Filippo Sabbadin

Luca Rossi

Yi Hao Zhuo

Redattore, Verificatore

Verificatore

Registro delle Versioni

| Versione | Data | Autore | Cambiamenti | Verificatore |
|----------|------------|--------------------|---|-------------------|
| 0.4.0 | 08/01/2025 | Gabriele Magnelli | Stesura nuove sezioni, migliorie e correzioni varie | Yi Hao Zhuo |
| 0.3.0 | 12/12/2024 | Gabriele Magnelli | Redazione sezioni Processi di supporto e Processi organizzativi | Yi Hao Zhuo |
| 0.2.1 | 11/12/2024 | Yi Hao Zhuo | Aggiunta link glossario | Matteo Bazzan |
| 0.2.0 | 30/11/2024 | Francesco Fragonas | Redazione Processi di accordo | Filippo Sabbadin |
| 0.1.1 | 30/11/2024 | Francesco Fragonas | Revisione Introduzione | Filippo Sabbadin |
| 0.1.0 | 12/11/2024 | Filippo Sabbadin | Prima stesura | Gabriele Magnelli |

Indice

| | |
|-------------------------------------|----------|
| 1. Introduzione | 1 |
| 1.1. Scopo del documento | 1 |
| 1.2. Scopo del prodotto | 1 |
| 1.3. Glossario | 1 |
| 1.4. Riferimenti | 2 |
| 2. Processi di accordo | 3 |
| 2.1. Processo di fornitura | 3 |
| 2.1.1. Scopo e descrizione | 3 |
| 2.1.2. Rapporti con il proponente | 3 |
| 2.1.3. Documentazione prodotta | 4 |
| 2.1.3.1. Valutazione dei capitolati | 4 |
| 2.1.3.2. Preventivo dei costi | 4 |
| 2.1.4. Analisi dei Rischi | 4 |
| 2.1.4.1. Analisi dei requisiti | 4 |
| 2.1.4.2. Piano di progetto | 4 |
| 2.1.4.3. Piano di qualifica | 5 |
| 2.1.5. Strumenti utilizzati | 5 |
| 2.2. Sviluppo | 5 |
| 3. Processi di supporto | 6 |
| 3.1. Documentazione | 6 |
| 3.1.1. Scopo e descrizione | 6 |
| 3.1.2. Composizione tipografica | 6 |
| 3.2. Struttura dei documenti | 6 |
| 3.2.1. Intestazione | 6 |
| 3.2.2. Registro delle modifiche | 7 |
| 3.2.3. Indice | 7 |
| 3.2.4. Corpo del documento | 7 |
| 3.2.5. Documenti del progetto | 7 |
| 3.2.6. Elenchi puntati | 8 |
| 3.2.7. Formato delle date | 8 |
| 3.2.8. Strumenti | 8 |
| 3.3. Gestione della configurazione | 8 |
| 3.3.1. Scopo e descrizione | 8 |
| 3.3.2. Versionamento | 8 |

| | |
|--|----|
| 3.3.3. Repository | 9 |
| 3.3.3.1. Struttura della repository documentazione | 9 |
| 3.3.4. Sincronizzazione | 9 |
| 3.3.4.1. Branch | 9 |
| 3.3.4.2. Pull request | 10 |
| 3.3.5. Strumenti usati | 10 |
| 3.3.6. Verifica | 11 |
| 3.3.6.1. Analisi statica | 11 |
| 3.3.6.2. Analisi dinamica | 11 |
| 3.3.7. Validazione | 11 |
| 3.3.8. Gestione qualità | 11 |
| 3.4. Processi organizzativi | 11 |
| 3.4.1. Gestione dei processi | 11 |
| 3.4.1.1. Scopo e descrizione | 11 |
| 3.4.1.2. Ruoli | 11 |
| 3.4.1.3. Ticketing (Issue Tracking System) | 12 |
| 3.4.2. Coordinamento | 13 |
| 3.4.2.1. Comunicazioni e Riunioni | 13 |
| 3.4.2.2. Verbali | 14 |
| 3.4.3. Miglioramento | 15 |
| 3.4.3.1. Scopo e descrizione | 15 |
| 3.5. Misure di qualità per il prodotto | 15 |
| 3.5.1. Funzionalità | 15 |
| 3.5.2. Efficienza | 15 |
| 3.5.3. Usabilità | 15 |
| 3.5.4. Affidabilità | 15 |
| 3.5.5. Manutenibilità | 15 |

1. Introduzione

1.1. Scopo del documento

Questo documento ha lo scopo di delineare le principali fasi di sviluppo, i ruoli e le responsabilità dei membri del team [CodeHex16*](#). Al suo interno, viene fornita una guida completa per tutte le [Practice*](#) adottate dal gruppo e per il [Way of Working*](#), garantendo un approccio strutturato e organizzato alle attività collaborative.

Il documento non si limita a fornire una panoramica iniziale ma si propone come un riferimento dinamico, soggetto a revisioni e aggiornamenti continui. Tale approccio incrementale assicura che il contenuto resti sempre aggiornato rispetto alle esigenze del progetto e alle best practices emergenti, consentendo al gruppo di adattarsi rapidamente a nuovi requisiti o cambiamenti contestuali.

1.2. Scopo del prodotto

Il progetto prevede lo sviluppo di un [Chatbot*](#) avanzato, basato su modelli linguistici [LLM*](#) (Large Language Models), pensato per migliorare la comunicazione tra aziende fornitrici e i loro clienti. Questo [assistente virtuale*](#) permetterà agli utenti di ottenere rapidamente e in modo intuitivo informazioni dettagliate su prodotti o servizi offerti, eliminando la necessità di contattare direttamente l'azienda.

Il sistema includerà anche un'interfaccia dedicata per le aziende fornitrici, offrendo strumenti per gestire i clienti e i documenti di riferimento che contengono le informazioni necessarie. Questi documenti saranno utilizzati dal modello linguistico per generare risposte personalizzate e accurate, garantendo un'esperienza utente ottimale. L'intero sistema sarà accessibile tramite una [Webapp*](#), assicurando una gestione efficiente e una fruizione semplice per tutti gli utenti coinvolti.

1.3. Glossario

Per agevolare la comprensione del presente documento, è stato predisposto un glossario che spiega il significato dei termini specifici utilizzati nel contesto del progetto. Per facilitare la comprensione, questi termini avranno il seguente stile: [Esempio*](#)

Le definizioni sono disponibili nel documento Glossario.pdf e possono essere consultate anche tramite la seguente pagina web: [Glossario.pdf](#)

1.4. Riferimenti

- Capitolato C7 - Assistente Virtuale Ergon:

<https://www.math.unipd.it/~tullio/IS-1/2024/Progetto/C7.pdf>

- Sito del gruppo CodeHex16:

<https://codehex16.github.io/>

- Repository della documentazione del progetto:

<https://github.com/CodeHex16/documentazione>

- Valutazione capitolati:

<https://codehex16.github.io/docs/1%20-%20candidatura/Valutazione-Capitolati.pdf>

- Preventivo costi e impegni:

<https://codehex16.github.io/docs/1%20-%20candidatura/Preventivo-Costi-e-Impegni.pdf>

- [Analisi dei requisiti*](#):

<https://codehex16.github.io/docs/2%20-%20RTB/Analisi-dei-Requisiti.pdf>

- [Piano di progetto*](#):

<https://codehex16.github.io/docs/2%20-%20RTB/Piano-di-Progetto.pdf>

- Analisi dei rischi

...

-Piano di qualifica ...

- Glossario:

Documento: <https://github.com/CodeHex16/documentazione/tree/main/glossario/glossario.pdf> Pagina web: <https://codehex16.github.io/glossario.html>

2. Processi di accordo

2.1. Processo di fornitura

Il processo di fornitura è strutturato in conformità agli esiti previsti dalla clausola 6.1.2 dello [Standard*](#) ISO/IEC/IEEE 12207:2017. Tale processo include la definizione di requisiti concordati, l'analisi dei rischi associati, e la pianificazione di tempi e costi.

2.1.1. Scopo e descrizione

Il processo di fornitura è finalizzato a garantire la realizzazione di un prodotto o servizio che soddisfi i requisiti concordati tra [Proponente*](#) e [Committente*](#). L'accordo tra le parti deve definire in modo chiaro i requisiti, le tempistiche e i costi da rispettare. Prima di stipulare tale accordo, il [Fornitore*](#) avrà condotto un'analisi dettagliata del progetto proposto, identificando i rischi correlati e stabilendo le linee guida necessarie per gestirli efficacemente.

2.1.2. Rapporti con il proponente

Il gruppo CodeHex16 manterrà un dialogo attivo e regolare con il Proponente per tutta la durata del progetto didattico, con l'obiettivo di raccogliere il maggior numero possibile di [Feedback*](#) sulla correttezza e qualità del lavoro svolto. La comunicazione si articolerà in due modalità principali:

1- Scritta ([asincrona*](#)) utilizzata per comunicazioni di breve durata, condivisione di verbali e materiale informativo e attività di coordinamento;

2- Incontri online ([sincrona*](#)) utilizzati per chiarimenti sul capitolato, approfondimenti relativi ai casi d'uso e requisiti e feedback sul lavoro svolto.

Il testo è chiaro, ma può essere migliorato leggermente per una maggiore fluidità e precisione:

I [Meeting*](#) saranno organizzati con cadenza variabile e fissati tramite e-mail in base alle necessità riscontrate durante lo sviluppo del progetto. Tutti i dettagli discussi durante questi incontri saranno documentati in verbali, con particolare attenzione alle decisioni prese. I verbali saranno disponibili al seguente link:

Verbali esterni - <https://github.com/CodeHex16/documentazione/tree/main/verbali/esterni>

2.1.3. Documentazione prodotta

In questa sezione viene illustrata la documentazione prodotta dal gruppo nel processo di fornitura, che sarà messa a disposizione del Proponente, Ergon Informatica, e dei Committenti, i professori Tullio Vardanega e Riccardo Cardin.

2.1.3.1. Valutazione dei capitoli

Nel documento *Valutazione Capitoli*, il gruppo ha analizzato tutte le proposte di capitolo, fornendo per ciascuna una breve descrizione, una panoramica dello stack tecnologico previsto e una valutazione finale. La scelta del capitolo è stata effettuata considerando diversi criteri, tra cui l'interesse dei membri del gruppo per il progetto, la sua rilevanza nel contesto lavorativo e la fattibilità complessiva.

2.1.3.2. Preventivo dei costi

Nel documento *Preventivo Costi e Impegni* è stata stabilita una data di consegna stimata del progetto, definita in accordo con tutti i membri del gruppo. La pianificazione tiene conto degli impegni personali di ciascun membro e prevede una stima delle ore settimanali da dedicare al progetto. Inoltre, dopo aver definito tutti i ruoli, è stata elaborata una tabella con la previsione delle ore che ogni membro deve svolgere per ciascun ruolo, garantendo una rotazione prestabilita per bilanciare equamente il carico di lavoro.

2.1.4. Analisi dei Rischi

Nel documento *Analisi dei Rischi* ...

2.1.4.1. Analisi dei requisiti

Nel documento *Analisi dei Requisiti*, il gruppo ha definito tutti gli [Use Case*](#) e i requisiti, frutto di un'attenta analisi del capitolo e della comprensione dell'utilizzo finale del progetto. Questa definizione è stata concordata con il referente dell'azienda proponente durante gli incontri svolti.

2.1.4.2. Piano di progetto

Nel documento *Piano di Progetto* è stato pianificato l'avanzamento del progetto suddiviso nei 3 periodi chiave (Candidatura, RTB e PB) con una particolare attenzione agli [Sprint*](#) settimanali effettuati. Per ognuno è stato descritto il lavoro svolto, il rendiconto delle ore e dei costi in base ai ruoli assegnati e le task future previste.

2.1.4.3. Piano di qualifica

Nel documento Piano di Qualifica ...

2.1.5. Strumenti utilizzati

Per lo svolgimento del progetto abbiamo utilizzato i seguenti strumenti:

- [Telegram*](#) per la comunicazione all'interno del gruppo;
- [Discord*](#) per i meet interni;
- **Zoom** per i meet esterni con il referente dell'azienda Ergon Informatica;
- **GitHub** per organizzare tutti i documenti e file sorgente del progetto;
- **GitHub Issue** per assegnare task ad ogni membro avendo un rendiconto preciso dei ruoli e delle ore svolte per ogni sprint, con l'assegnazione di label e milestone specifiche;
- **Google Fogli** per organizzare incontri con la compilazione di un calendario settimanale e per fissare le ore svolte avendo una visione generale dell'andamento del progetto;
- **Typst** per la stesura di tutti i documenti e verbali;
- **Canva** per la realizzazione delle presentazioni per i Diari di Bordo settimanali;
- [Notion*](#) per organizzare appunti e documenti non ufficiali;

2.2. Sviluppo

3. Processi di supporto

3.1. Documentazione

3.1.1. Scopo e descrizione

Lo scopo del processo di documentazione è quello di tracciare e, quindi, rendere immediatamente consultabile ogni attività e processo relativi al progetto. Inoltre, saranno riportate le decisioni prese e le norme scelte dal gruppo CodeHex16 che verranno rispettate da tutti i suoi membri al fine di procedere al meglio nello sviluppo del progetto.

3.1.2. Composizione tipografica

Per la composizione tipografica dei documenti si è deciso di usare [Typst*](#) per i seguenti motivi:

- Semplicità d'uso;
- Sintassi semplice;
- Compilazione immediata;

Grazie a Typst si riesce facilmente a creare e mantenere un documento non lasciando il lavoro di controllo grafico al gruppo, infatti si possono scrivere i vari tipi di documenti partendo dai template che si possono trovare nella repository documentazione nella cartella *template*.

3.2. Struttura dei documenti

Ci sono diversi tipi di documenti e generalmente sono organizzati nelle seguenti sezioni.

3.2.1. Intestazione

La prima pagina è l'intestazione del documento ed è composta generalmente dalle seguenti informazioni:

- **Titolo:** Titolo del documento che è anche il nome del documento;
- **E-mail:** E-mail del gruppo;
- **Logo:** Logo del gruppo;
- **Tipo:** Il verbale può essere di tipo interno o esterno;
- **Data:** La data in cui il gruppo si è incontrato e/o in cui il documento è stato redatto;

- **Ora:** L'orario in cui è avvenuto l'incontro di gruppo;
- **Versione:** La versione corrente del documento;
- **Ordine del giorno:** L'elenco degli argomenti principali discussi durante l'incontro;
- **Presenze/Ruoli:** Per ogni membro del gruppo presente durante l'incontro viene registrato il suo nome e cognome, il ruolo svolto e il tempo per cui è stato presente durante l'incontro;

3.2.2. Registro delle modifiche

La seconda pagina riguarda il registro modifiche il cui contenuto è organizzato mediante una tabella in cui vengono riportate le seguenti informazioni:

- **Versione:** Indica il numero della versione del documento;
- **Data:** Data della versione in cui è redatto il documento;
- **Autore:** Autore di quella versione del documento, cioè il membro del gruppo che ha apportato le modifiche al documento;
- **Cambiamenti:** I cambiamenti principali di quella versione del documento;
- **Verificatore:** Membro del gruppo che ha verificato il documento per quella versione;

3.2.3. Indice

Nella terza pagina, e se necessario le seguenti, è riservata all'indice del documento che elenca le sezioni di cui è composto il documento.

3.2.4. Corpo del documento

Il corpo del documento è suddiviso in capitoli ognuno dei quali può essere diviso in sottocapitoli che a loro volta possono essere suddivisi in altri sottocapitoli.

3.2.5. Documenti del progetto

Verranno prodotti i seguenti documenti:

- [Norme di Progetto*](#);
- **Piano di Progetto;**
- **Analisi dei Requisiti;**
- **Piano di Qualifica;**
- **Glossario;**
- **Verbali esterni;**
- **Verbali interni;**

3.2.6. Elenchi puntati

Ogni voce di un elenco puntato finisce con “;”.

3.2.7. Formato delle date

Per le date viene utilizzato lo standard internazionale **ISO 8601** nella forma YYYY-MM-DD in cui:

- **YYYY**: Indica l'anno con 4 cifre;
- **MM**: Indica il mese con 2 cifre;
- **DD**: Indica il giorno con 2 cifre;

3.2.8. Strumenti

- **VS Code**: Editor di testo usato per scrivere i documenti;
- **Typst**: Linguaggio per la stesura dei documenti;
- **Github**: Servizio di hosting per il repository;

3.3. Gestione della configurazione

3.3.1. Scopo e descrizione

Il processo di gestione di configurazione identifica le norme adottate dal gruppo per garantire la tracciabilità della documentazione e del codice prodotto durante tutto l'arco di vita del progetto. Lo scopo principale è quello di organizzare la procedura di modifica della documentazione e del codice prodotto e di rendere immediatamente consultabili le varie modifiche apportate e i loro autori.

3.3.2. Versionamento

In generale una versione ha una sintassi del tipo X.Y.Z in cui:

- **X**: E' il numero di versione principale che viene incrementato ogni qual volta il documento sia terminato e pronto per una revisione;
- **Y**: E' il numero di versione secondaria che viene incrementato ogni qual volta il documento sia stato modificato in modo significativo;
- **Z**: E' il numero di versione di correzione, incrementato ogni qual volta il documento subisca correzioni minori;

In particolare, per i verbali vengono considerati solo i numeri Y e Z per il fatto di essere documenti molto brevi, in cui:

- **Y:** E' il numero di versione principale che viene incrementato ogni qual volta il documento sia terminato o venga modificato con correzioni importanti;
- **Z:** E' il numero di versione secondaria che viene incrementato ogni qual volta il documento sia stato modificato con piccole correzioni;

3.3.3. Repository

3.3.3.1. Struttura della repository documentazione

Il contenuto pronto e convalidato del repository è presente nel [branch*](#) main in cui sono presenti tutti i PDF dei documenti prodotti. Nel branch main si possono trovare diverse cartelle che servono per organizzare e dividere i vari tipi di documenti e codice, in particolare:

- In **1 - candidatura** si trovano i documenti relativi alla candidatura per la gara d'appalto di dei capitolati;
- In **2 - RTB** si trovano i documenti relativi alla fase di progetto **RTB(Requirement and Technology Baseline)**;
- In **3 - PB** si trovano i documenti relativi alla fase di progetto **PB(Product Baseline)**;
- In **diari-di-bordo** sono presenti tutti i diari di bordo prodotti;
- In **glossario** è presente il documento Glossario;
- In **verbali** sono presenti sia i verbali interni sia quelli esterni;

3.3.4. Sincronizzazione

La sincronizzazione avviene tramite repository condivise su [github](#) in cui ogni attività da svolgere è tracciata da una issue con il/i membro/i assegnato/i a tale issue così da sapere sempre chi la segue, o l'ha seguita.

3.3.4.1. Branch

Per gestire al meglio le varie issue e la documentazione si è deciso di creare dei branch appositi per ogni documento importante come **Norme di Progetto**, **Piano di Progetto**, **Glossario**, **Piano di Qualifica** e **Analisi dei Requisiti**. Inoltre, se più membri del gruppo lavorano allo stesso documento allora viene creato un branch per ogni membro con nome il nome del documento/attività e di chi lo sta svolgendo. Quindi in generale la sintassi del nome di un branch è la seguente: **[Titolo]-[Nome-Membro]-[Sprint]**. Una volta finito il lavoro da parte di tutti i membri che operano su quel documento, questo viene verificato (tramite [pull request*](#)) e viene eseguito il merge sul branch main, mentre i branch ausiliari vengono eliminati. Altri branch degni di nota fino ad ora sono:

- **diario-di-bordo**: Utilizzato per scrivere e verificare i diari di bordo;

3.3.4.2. Pull request

Per quanto riguarda le pull request si è deciso che per ogni documento redatto viene richiesta la verifica tramite pull request. A questo punto i verificatori incaricati del sprint corrente (vengono create delle issue per la verifica assegnandole a chi di dovere), tramite questa pull request:

1. Esegue il controllo locale

```
# Clonare il repository
git checkout <pr-branch-name>
```

2. Eseguono la verifica e aggiungere proprio nome nel Registro dell Versioni
3. Se è tutto corretto viene committato

```
git add .
git commit -m "Verifica <pr-branch-name>...."
git push origin <pr-branch-name>
```

4. Dopo di che viene fatto il merge delle modifiche apportate nel main sulla pagina Pull Request del Github
5. Altrimenti il verificatore può correggere direttamente il documento, oppure scrivere un commento con delle indicazioni per le correzioni da svolgere.

In generale, le pull request vengono effettuate quando vi è una modifica interna al repository.

3.3.5. Strumenti usati

- **Git**: Software usato per il controllo della versione dei documenti e del codice;
- **Github**: Servizio di hosting per progetti software usato dal gruppo per coordinarsi sulle operazioni di versionamento e usato come **Issue Tracking System**;

3.3.6. Verifica

3.3.6.1. Analisi statica

3.3.6.2. Analisi dinamica

3.3.7. Validazione

3.3.8. Gestione qualità

3.4. Processi organizzativi

3.4.1. Gestione dei processi

3.4.1.1. Scopo e descrizione

Il processo di gestione ha lo scopo di identificare le attività e i compiti che ogni membro del gruppo dovrà eseguire per proseguire nel progetto.

3.4.1.2. Ruoli

I ruoli svolti, a rotazione, dai membri del gruppo sono:

1. **Responsabile:** Ha il compito primario di coordinare i membri del gruppo, inoltre deve:

- Determinare le attività da svolgere, assegnarle e verificarne l'avanzamento;
- Gestire i rapporti tra i membri del gruppo e i soggetti esterni;
- Redigere i verbali sia interni che esterni;

Il responsabile è una figura che sarà presente durante tutto l'arco del progetto.

2. **Amministratore:** Ha il compito primario di controllare e gestire l'ambiente di lavoro, inoltre deve:

- Stabilire gli strumenti necessari da usare durante il progetto;
- Gestire i processi e risolverne gli eventuali problemi;

L'amministratore è una figura che sarà presente durante tutto l'arco del progetto.

3. **Progettista:** Ha lo scopo principale di determinare le scelte realizzative del progetto, in particolare deve:

- Trovare l'architettura adeguata per gestire il progetto;

Il progettista è una figura che sarà, principalmente, presente durante la parte di sviluppo del progetto.

4. **Analista:** Ha il compito principale di trovare i requisiti che il progetto dovrà soddisfare e riportarli nel documento **Analisi dei Requisiti**, quindi deve:

- Studiare i bisogni dei committenti;
- Studiare i requisiti definendone la complessità;
- Scrivere il documento **Analisi dei Requisiti**

L'analista è una figura che sarà, principalmente, presente durante la prima parte del progetto in cui verrà analizzato e compreso appieno il capitolato. Solo in casi straordinari, cioè se il gruppo dovesse cambiare i requisiti del progetto, allora l'Analista potrebbe.

5. **Programmatore:** Ha il compito primario di svolgere l'attività di codifica e sviluppare l'architettura individuata dal Progettista, in particolare deve:

- Scrivere codice mantenibile che rispetti le **Norme di Progetto**;
- Creare test per la verifica e validazione del codice;

Il programmatore è una figura che sarà presente durante la parte di sviluppo del progetto.

6. **Verificatore:** Ha il compito principale di controllare e validare la documentazione e il codice prodotto, in particolare deve:

- Controllare ogni documento a lui assegnato, verificarlo e in caso correggerlo o notificare il redattore, specificare cosa non è corretto e richiederne la correzione;
- Controllare che tutto ciò che viene prodotto rispetti le **Norme di Progetto**;

Il Verificatore è una figura che sarà presente durante tutto il progetto. In particolare ogni membro, in un dato momento, può svolgere un solo ruolo alla volta, ma durante lo sprint può assumere più ruoli.

3.4.1.3. Ticketing (Issue Tracking System)

Il gruppo sfrutta l'ITS(Issue Tracking System) offerto da Github per gestire le attività da svolgere, cioè le **Issue**. Creare le issue è molto semplice e veloce e una volta create (ogni membro del gruppo può creare una issue), quindi quando viene individuata un'attività

specifica da svolgere, questa viene assegnata, in modo coerente, ad un membro del gruppo. Quando viene creata una issue, questa sarà composta da:

- **Titolo:** Il nome della issue che identifica l'attività da svolgere;
- **Assegnatario/i:** Il/I membro/i del gruppo a cui è affidata la issue;
- **Etichetta/e:** Identifica il tipo di issue;
- **Stato:** Avanzamento della issue;
- **Priorità:** Identifica l'importanza di svolgere la issue;
- **Peso:** Stima del carico di lavoro relativo alla issue;
- **Sprint:** Lo sprint in cui questa issue deve, idealmente, essere svolta;
- **Ruolo:** Identifica il ruolo associato allo svolgimento di tale issue;
- **Ore:** Il numero di ore impiegate per svolgere la issue;
- **Repository:** Repository a cui è associata la issue;
- **Milestone:** Identifica la milestone a cui è assegnata la issue;

Più in particolare quando viene individuato un compito da svolgere vengono eseguiti i seguenti passi:

1. Viene individuato il compito da svolgere e viene creata la issue relativa;
2. La issue viene assegnata ad uno o più membri del gruppo e vengono inserite le informazioni della issue scritte appena sopra.
3. La issue viene completata e chiusa, oppure se è richiesta la verifica del lavoro svolto allora:
 4. Il verificatore, che ha una issue di verifica parallela, controlla il lavoro svolto e;
 - 4.1. Se corretto vengono confermate le modifiche e la issue principale e quella del verificatore vengono chiuse;
 - 4.2. Altrimenti le issue rimangono aperte e il verificatore suggerisce dei cambiamenti e/o correzioni a carico dell'assegnatario, una volta apportate tali modifiche si torna al punto precedente (4.1);

3.4.2. Coordinamento

3.4.2.1. Comunicazioni e Riunioni

Le comunicazioni principali che avvengono durante lo svolgimento del progetto sono di due tipi:

- **Comunicazioni interne:** Il gruppo utilizza **Telegram** e **Discord** per le comunicazioni principali interne, in particolare Telegram viene usato per messaggi brevi, veloci e informali, mentre Discord viene usato per discussioni e riunioni a distanza.

Inoltre, in caso di problemi su Telegram, il gruppo può spostarsi su Discord per comunicare anche in modo non formale;

- **Comunicazioni esterne:** Per le comunicazioni esterne vengono usate la mail di gruppo **unipd.codehex16@gmail.com** e **Zoom** per chiarire dubbi e porre domande;

Anche le riunioni sono di due tipi:

- **Riunioni interne:** I membri del gruppo si riuniscono su **Discord** dove si fa il punto della situazione in quel momento e se necessario si introducono nuove attività da svolgere e/o si procede con quelle già indicate;
- **Riunioni esterne:** I membri del gruppo si riuniscono insieme al proponente, questa riunione di solito è richiesta dal gruppo per trattare problemi/dubbi di una certa importanza al fine di avere una migliore comprensione delle attività da svolgere e procedere con il progetto;

Sarà compito del **Responsabile** riassumere in un **verbale interno**, o un **verbale esterno** quello che si è discusso durante la riunione in questione;

3.4.2.2. Verbali

I verbali redatti sono di due tipi:

- **Verbali interni:** Questo tipo di verbale è la trascrizione dei punti salienti di una riunione interna e, generalmente, è scritto dal **Responsabile**, l'obiettivo principale è quello di discutere delle attività svolte e delle attività da svolgere, stabilendo quindi nuove issue, se necessario;
- **Verbali esterni:** Questo tipo di verbale è la trascrizione dei punti più importanti riscontrati durante una riunione esterna, il cui obiettivo principale è quello di risolvere dubbi/problemi riscontrati durante l'avanzamento del progetto. Anche questo verbale è, generalmente, redatto dal **Responsabile**;

Nel caso in cui il Responsabile non fosse presente durante le riunioni, il verbale corrispondente verrà redatto dall'**Amministratore**, se neanche l'amministratore era presente allora sarà uno dei membri presenti alla riunione che avrà l'incarico di redigere il verbale.

3.4.3. Miglioramento

3.4.3.1. Scopo e descrizione

Il miglioramento è un processo sempre attivo che durerà per tutto il progetto il cui obiettivo è quello di controllare e migliorare tutto quello che viene prodotto mantenendo un elevato grado di qualità.

3.5. Misure di qualità per il prodotto

3.5.1. Funzionalità

3.5.2. Efficienza

3.5.3. Usabilità

3.5.4. Affidabilità

3.5.5. Manutenibilità