



Laurea Magistrale in informatica-Università di Salerno  
Corso di Gestione dei Progetti Software- Prof.ssa F. Ferrucci



# Software Project Management Plan Comun-ity

Riferimento	
Versione	1.5
Data	07/01/2023
Destinatario	Top Management
Presentato da	Testa Elio, Iannucci Michele
Approvato da	



## Revision History

Data	Versione	Descrizione	Autori
07/12/2022	0.1	Prima stesura	Testa Elio, Iannucci Michele
30/12/2022	1.0	Stesura dei Capitoli Iniziali	Testa Elio, Iannucci Michele
04/01/2023	1.1	Stesura Capitoli 3,4,5	Testa Elio
05/01/2023	1.2	Stesura Capitoli 6,7	Iannucci Michele
06/01/2023	1.3	Stesura Capitoli Finali	Testa Elio, Iannucci Michele
07/01/2023	1.4	Revisione	Iannucci Michele
24/01/2023	1.5	Revisione finale	Testa Elio, Iannucci Michele



## Indice

---

1. Project Overview	<b>5</b>
1.1. Project Review	5
1.1.1. Purpose, Scope & Objectives	5
1.1.2. Assumption & Constraints	5
1.1.3. Project Deliverables	6
1.1.4. Schedule & Budget Summary	7
2. Referencies	<b>9</b>
3. Definitions	<b>9</b>
4. Project Context	<b>10</b>
4.1. Process Model	10
4.2. Project Improvement Plan	10
4.3. Infrastructure Plan	10
4.4. Methods, Tools & Techniques	11
4.5. Product Acceptance Plan	12
4.6. Project Organization	12
5. Project Planning	<b>13</b>
5.1. Project Initiation	13
5.1.1. Estimation Plan	13
5.1.2. Staff Plan	13
5.1.3. Resource Acquisition Plan	13
5.1.4. Project Staff Training Plan	14



5.2. Estimation Plan	14
5.2.1. Work Activities	14
5.2.2. Schedule Allocation	15
6. Project Assessment & Control	<b>15</b>
6.1. Requirements Management Plan	15
6.2. Scope Change Control Plan	15
6.3. Schedule Control Plan	16
6.4. Budget Control Plan	16
6.5. Quality Assurance Plan	16
6.6. Project Closeout Plan	17
7. Product Delivery	<b>17</b>
8. Supporting Process Plan	<b>18</b>
8.1. Project Supervising & Work Environment	18
8.2. Decision Management	18
8.3. Risk Management	18
8.4. Configuration Management	19
8.5. Quality Assurance	20
8.6. Measurement	20
9. Additional Plans	<b>21</b>



# 1. Project Overview

---

## 1.1. Project Review

---

### 1.1.1. Purpose, Scope & Objectives

---

I Comuni intendono fornire sia ai propri cittadini che a futuri residenti un servizio di “Smart Community” che permetta loro di risolvere piccoli problemi di vita quotidiana in maniera semplice ed efficace, soddisfacendo le loro esigenze ed andando a sfruttare e rafforzare il senso di comunità digitale. Le feature principali previste per il sistema sono le seguenti:

- Gli utenti devono poter inserire informazioni sulla loro problematica in maniera più esaustiva e facile possibile;
- Gli utenti devono poter cercare fra le varie problematiche e poter decidere se accettare un problema di un altro utente o meno;
- Gli utenti devono poter lasciare feedback sull’operato di altri utenti che hanno accettato la propria problematica;
- Gli utenti che hanno accettato una richiesta devono poterla rifiutare nel caso in cui siano impossibilitati nel portarla a compimento;
- Gli utenti avranno la possibilità di interagire con un “portafoglio” con cui pagare e ricevere denaro;
- Altri requisiti suggeriti dagli utenti per migliorare l’esperienza sul software e/o il suo valore di business.

### 1.1.2. Assumption & Constraints

---

Il progetto ha tre vincoli che il team dovrà rigorosamente rispettare, rappresentati dalle scadenze della prima consegna intermedia datata il 12 dicembre 2022, della consegna finale del progetto datata 25 gennaio 2023 e dalle ore di lavoro imposte dal Top Manager. Per ogni membro del team, il Top Manager ha imposto un budget di 50 ore che tutti i team member dovranno garantire. Per quanto riguarda le scadenze, fare riferimento al paragrafo 1.1.3. Per quanto riguarda i vincoli si suddividono in 2 tipologie:

- Collaborativi e comunicativi:



- Utilizzo di sistema di versioning: nello specifico GitHub;
- Uso di tool per la gestione di task e attività: nello specifico Trello;
- Uso di tool di comunicazione tracciabile: nello specifico Slack;
- Tecnici: che si diversificano per i vari documenti:
  - RAD:
    - Specifica di minimo 2 massimo 4 scenari per ogni membro del team;
    - Specifica di minimo 2 massimo 4 requisiti funzionali per ogni membro del team;
    - Esattamente 1 Use Case per ogni membro del team;
    - Esattamente 3 Sequence Diagram;
    - 1 Activity Diagram Sistema Corrente e Proposto;
    - 1 Class Diagram senza Object Diagram;
  - SDD:
    - Specifica di minimo 2 massimo 4 design goal per ogni membro del team;
    - Definizione di un diagramma di decomposizione dei sottosistemi con descrizione e motivazione all'uso;
    - Definizione di un deployment diagram con descrizione e motivazione all'uso;
  - ODD:
    - Uso di minimo uno e massimo due design pattern per team;
    - Uso di UML;
  - Testing:
    - Test d'unità: Esattamente 1 funzionalità per ogni membro del team tramite category partition;
    - Test di sistema: Esattamente 1 funzionalità per ogni membro del team tramite category partition;

### 1.1.3. Project Deliverables

Codice	Documento	Responsabile	Data di completamento pianificata
D_0	RAD	Gabriele Santoro	27/11/2022



D_1	SDD	Alessandro Falcone	02/12/2022
D_2	TCS	Biagio Andreucci	09/12/2022
D_3	TP	Biagio Andreucci, Elio Testa, Michele Iannucci	07/12/2022
M_1	Consegna Intermedia	Tutti	12/12/2022
D_4	ODD	Andrea Aceto	23/12/2022
D_5	Implementazione	Michele Rabesco	16/01/2023
D_6	Test d'unità	Team Members	16/01/2023
D_7	Test d'integrazione	Team Members	16/01/2023
D_8	Test di Sistema	Team Members	16/01/2023
D_9	TIR	Michele Rabesco	16/01/2023
M_2	Consegna Finale	Tutti	25/01/2023

#### 1.1.4. Schedule & Budget Summary

Lo schedule del progetto sarà creato attraverso il software Microsoft Project, verranno stilate tutte le task inizialmente senza assegnazione di costi e risorse, e ogni task rappresenta un Work Package all'interno del WBS Dictionary. Verranno suddivise le task in gruppi basandosi sui vari documenti da creare attraverso un macro package con i nomi dei documenti e infine i macro package racchiusi in un unico progetto con il nome Comun-ity.

Ad ogni Work Package saranno poi assegnate le varie dipendenze e le varie risorse, poi si passerà all'analisi dei costi introducendo inizialmente la baseline dei costi stimati e in un secondo momento i costi effettivi.

Le risorse saranno assegnate di settimana in settimana, i costi effettivi saranno aggiornati quando la task sarà effettivamente completata e saranno riportate le seguenti informazioni:

- numero di risorse allocate;
- costo singola risorsa;
- numero di ore impiegate per lo svolgimento della task.

Per altre informazioni relative allo schedule fare riferimento ai documenti:



- Schedule Management Plan
- WBS Dictionary
- WBS GANTT

Una prima stima dei costi del progetto si aggira intorno a 3000€ al primo anno. Questa stima è basata su diversi fattori:

- I 2 Project Manager lavorano per un totale di 25 ore mentre i 5 Team Member lavorano per un totale di 50 ore. 1 Project Manager guadagna 14€ all'ora mentre un Team Member 9€ all'ora;

Le spese di manutenzione sono stimate a 2300 € all'anno per andare ad effettuare una pulizia del database, risolvere bug ed effettuare nuovi rilasci. Questa stima si basa su diversi fattori:

- Sono impiegati nella manutenzione 1 Project Manager e 3 Team Member che lavoreranno in totale 25 ore il primo e 50 ore i secondi distribuite nell'arco dell'anno. 1 Project Manager guadagna 14€ all'ora mentre un Team Member 9€ all'ora;
- Spesa annuale di 600€ per la manutenzione e rinnovo dei costi di servizi tra cui, server, database e dominio;

Il 90% dei costi sono coperti annualmente dai fondi PNRR almeno per i 3 anni di vita utile del sistema.

La parte restante dei costi è coperta da 3 possibili fattori:

- Commissioni sui pagamenti: è stimato che dal secondo anno si avranno mediamente circa 400 transazioni con una media di 20€ a transazione e un commissione del 1,2% su ognuna di esse.
- Abbonamenti: è stimato che dal secondo anno si avranno mediamente circa 75 abbonamenti al prezzo di 10€ l'uno.
- Investimenti Privati: possibili investimenti da parte di altri enti e/o da cittadini interessati all'innovazione e proseguo del sistema stimate a circa 100€ l'anno.

Quindi il totale dei benefici stimati per ogni anno è di circa 4000€.

Sommariamente considerando il totale dei costi e dei benefici, il Net Present Value, il Return On Investment e considerando che è una stima preliminare dei costi possiamo dedurre che è abbastanza incoraggiante. Il rimborso dei costi è previsto già alla fine del primo anno di vita del sistema, il NPV è di 5000€ circa e il ROI basato sulla vita utile del sistema di 3 anni è del 60%.

Il budget finale dato dalla somma dei costi previsti e le riserve si attesta intorno ai 10771€.

Per altre informazioni relative allo schedule fare riferimento ai documenti:





- Business Case;
- Business Case Financial;

## 2. Referencies

---

- Statement of Work;
- Project Charter;
- Business Case;
- Business Case Financial;
- WBS Dictionary;
- WBS GANTT;
- Quality Plan;
- RAD;
- Schedule Management Plan;
- Risk Management Plan;
- ODD;
- Test Plan;
- Test Incident Report;
- Test Summary Report;
- Test Case Specification;
- SDD;

## 3. Definitions

---

- **Deliverables:** con tale termine si intende il prodotto di una o più fasi del progetto.
- **Milestone:** con tale termine si intende il raggiungimento di un importante traguardo nello svolgimento del progetto.
- **Work Breakdown Structure (WBS):** rappresentazione di un progetto tramite le attività che lo compongono, attraverso una rappresentazione gerarchica.
- **Diagramma di Gantt:** Rappresentazione della WBS attraverso il diagramma Gantt, utile per visionare le attività che compongono la WBS nel lasso temporale definito.



## 4. Project Context

---

### 4.1. Process Model

---

Il progetto viene sviluppato utilizzando il modello a cascata con verifica e validazione, con l'obiettivo di verificare e assicurarsi che il prodotto, gli artefatti e la relativa documentazione soddisfino i criteri di qualità e il cliente. Il processo a cascata viene applicato nelle prime fasi del progetto, favorendo l'interazione con il cliente e con l'obiettivo di individuare errori o difetti del sistema. Ogni fase sarà accompagnata da feedback. Per quanto riguarda lo sviluppo e l'implementazione del sistema, si è fatto utilizzo di un approccio Scrum-like, facendo uso degli Sprint di durata settimanale per la gestione dei task e con la programmazione dei daily standups meeting. Infine, lo sviluppo dell'intero sistema sarà eseguito tramite un processo di Continuous Integration e Continuous Deployment, utile per automatizzare e monitorare continuamente tutto il ciclo di vita dell'applicazione. L'obiettivo dell'utilizzo di una pipeline è quello di individuare errori e ridurre il tasso nel codice, attraverso il monitoraggio di piccole modifiche incrementali. Per altre informazioni fare riferimento al documento di Software Configuration Management Plan.

### 4.2. Project Improvement Plan

---

La verifica e la validazione saranno ripetute più e più volte con l'obiettivo di migliorare il prodotto finale, utilizzando checklist fornite dai Project Manager dettate dall'esperienza pregressa. Infine, sono previste attività di training al fine di migliorare e velocizzare i processi di design e sviluppo, migliorando inoltre le soft e hard skills dei Team Members.

### 4.3. Infrastructure Plan

---

Per le prime fasi i Team Member lavoreranno quanto più possibile in presenza in modo da poter discutere eventuali problematiche fra di loro e migliorare lo spirito di squadra. Nella fase di Development potranno lavorare in remoto utilizzando Teams e lavorando ognuno sulla propria macchina.



## 4.4. Methods, Tools & Techniques

---

Per la comunicazione e la coordinazione del lavoro all'interno del Team, si fa utilizzo di:

- Trello, per la coordinazione dei task nelle prime fasi di progettazione;
- GitHub Projects, per la coordinazione dei task durante la fase di implementazione;
- Slack, per le comunicazioni formali;
- Teams, per eventuali meeting formali e comunicazioni in remoto fra i membri del team;

Inoltre, sono stati previsti i seguenti strumenti per aumentare l'affiatamento all'interno del team:

- Whatsapp, per le comunicazioni informali;
- Incontri in presenza;

Per lo sviluppo degli artefatti si fa utilizzo di:

- Microsoft Word e Excel online, per la stesura dei documenti e per i fogli di lavoro;
- draw-io, per la realizzazione dei diagrammi UML;
- Balsamiq, per la realizzazione rapida dei mockup;
- Eclipse, come IDE per lo sviluppo del codice sorgente;
- CheckStyle, per verificare costantemente la correttezza del codice Java;

Per quanto riguarda lo sviluppo e il testing, si farà utilizzo dei seguenti tool e tecnologie:

- HTML e CSS, per lo sviluppo del front-end all'interno delle JSP;
- JSP, per la costruzione di pagine web con codice Java;
- Java, per lo sviluppo del back-end;
- JavaScript e jQuery, per semplici operazioni di controllo sul front-end o per la costruzione di pagine dinamiche;
- MongoDB, come database NoSQL;
- JUnit, per il testing unitario delle componenti del sistema;
- Mockito, per semplificare il testing unitario, evitando la costruzione di stub/driver;
- Selenium, per il testing di sistema;



Inoltre fare riferimento al documento di Object Design Document per eventuali tecnologie aggiuntive adottate.

Infine, per la build e il deploy del sistema saranno utilizzati:

- Maven, per la gestione delle dipendenze, la build del progetto e l'esecuzione dei test in locale.
- GitHub per il versioning ed il pull-based development.
- GitHub Actions, per la pipeline su GitHub.
- Apache Tomcat, per il deploy in locale del progetto.

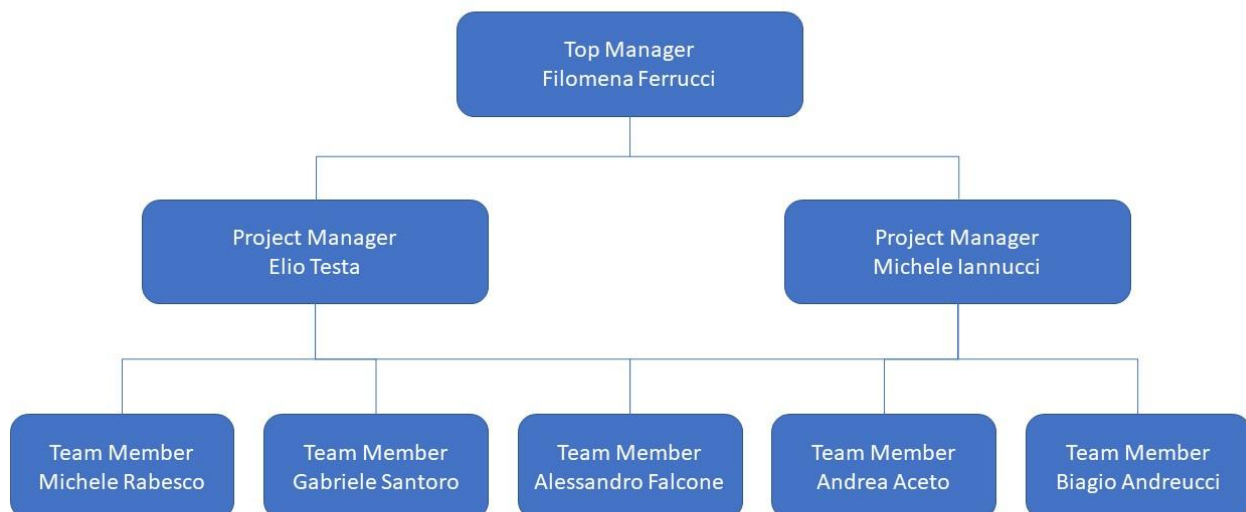
## 4.5. Product Acceptance Plan

---

- Il numero di warning sollevati da Checkstyle deve essere basso;
- Progetto completato entro i tempi stabiliti.

## 4.6. Project Organization

---





## 5. Project Planning

### 5.1. Project Initiation

#### 5.1.1. Estimation Plan

Il budget a disposizione per il progetto è rappresentato dalle ore di lavoro, pari a 250 ore. L'utilizzo di tale budget è previsto come segue:

- Requirements Elicitation & Analysis: 80h
- System Design: 25h
- Object Design: 27,5h
- Development: 77,5h
- Testing: 40h

#### 5.1.2. Staff Plan

Competenze/Caratteristiche	Livello Richiesto Basso-Medio-Alto	Importanza Basso-Medio-Alto
Lavoro di gruppo	Alto	Alto
Problem Solving	Medio	Medio
Programmazione Java	Medio	Alto
Basi di dati (SQL)	Basso	Basso
Basi di dati (NoSQL)	Medio	Alto
Sviluppo Web	Alto	Alto
Bootstrap	Medio	Alto

#### 5.1.3. Resource Acquisition Plan

Per quanto riguarda l'acquisizione delle risorse e la formazione dei team, si è tenuta una giornata in cui i vari Project Manager si sono presentati agli studenti del corso di Ingegneria del Software.



Successivamente, questi hanno espresso la loro preferenza circa il Project Manager scelto. Infine, i vari Project Manager si sono riuniti per discutere e valutare le varie preferenze espresse dagli studenti, in modo tale da comporre il team più adatto alle proprie necessità.

#### 5.1.4. Project Staff Training Plan

Durante lo sviluppo del progetto, saranno organizzate sedute di training sulle seguenti tematiche per colmare le carenze dei team members:

- Slack e Trello;
- Git, GitHub;
- Maven;
- Testing con JUnit, Mockito e Selenium se necessario;

Inoltre, verrà fornito del materiale di supporto come slides, documentazione e link a risorse per aiutare l'acquisizione di conoscenze oltre alle slide del corso di Ingegneria del Software.

### 5.2. Estimation Plan

#### 5.2.1. Work Activities

▲ <b>Comun-ity</b>	<b>63 g</b>	<b>ven 28/10/22</b>	<b>mar 17/01/23</b>
▷ <b>RAD</b>	<b>24 g</b>	<b>ven 28/10/22</b>	<b>dom 27/11/22</b>
▷ <b>SDD</b>	<b>5 g</b>	<b>lun 28/11/22</b>	<b>ven 02/12/22</b>
▷ <b>Test Plan</b>	<b>4 g</b>	<b>sab 03/12/22</b>	<b>mer 07/12/22</b>
▷ <b>Test Case Specification</b>	<b>2 g</b>	<b>gio 08/12/22</b>	<b>ven 09/12/22</b>
Scadenza Intermedia	0 g	lun 12/12/22	lun 12/12/22
▷ <b>ODD</b>	<b>11 g</b>	<b>sab 10/12/22</b>	<b>ven 23/12/22</b>
▷ <b>Development</b>	<b>19 g</b>	<b>lun 26/12/22</b>	<b>lun 16/01/23</b>
Consegna Progetto	0 g	mar 17/01/23	mar 17/01/23

Per ulteriori dettagli sulla WBS fare riferimento al documento di WBS Document e WBS GANTT'.



### 5.2.2. Schedule Allocation

---

L'allocazione delle risorse inizialmente sarà di tutte le risorse disponibili per ogni task, in quanto si reputa opportuna la conoscenza da parte di tutti i team member di ogni parte del progetto. Nel caso in cui subentri un eventuale ritardo nella schedulazione si andrà ad effettuare una parallelizzazione dividendo le risorse su task differenti. Per ulteriori informazioni consultare i documenti di WBS per l'effettiva allocazione e il documento di Schedule Management Plan.

## 6. Project Assessment & Control

---

### 6.1. Requirements Management Plan

---

Nelle prime fasi di progetto, saranno individuati i requisiti funzionali ed i requisiti non funzionali, con l'obiettivo di avere una visione chiara di ciò che dovrà essere e contenere il prodotto. Ai requisiti verrà assegnato un determinato livello di priorità, che potrà essere:

- **ALTA:** funzionalità che si ritengono fondamentali per il sistema.
- **MEDIA:** funzionalità che completano il sistema.
- **BASSA:** funzionalità di contorno che potranno essere implementate in versioni future.

Inoltre, si terrà traccia di tutti i requisiti attraverso una matrice di tracciabilità, con l'obiettivo di aumentare la manutenibilità del sistema.

### 6.2. Scope Change Control Plan

---

Durante lo sviluppo del progetto è possibile che sorgano delle Change Request da parte degli stakeholder. Le Change Request dovranno seguire il seguente ordine:

- **Proposta:** il proponente, può inviare la propria Change Request direttamente al Project Manager;
- **Valutazione:** il Project Manager analizzerà l'impatto del cambiamento, in termini di modifiche da effettuare e rischi associati, con relativo valore di business. Inoltre la Change Request non dovrà in alcun modo ritardare l'intero progetto o sforare il numero di ore progettuali;



- **Accettazione/Rifiuto:** l'accettazione o il rifiuto della Change Request è il risultato della valutazione precedente;
- **Implementazione:** in caso di accettazione, la Change Request viene assegnata e implementata.

### 6.3. Schedule Control Plan

---

L'andamento del progetto è verificato dai Project Manager attraverso:

- Meeting formali previsti con cadenza settimanale per tutta la durata del progetto
- Daily Scrums per la fase di Development
- Continua interazione con i team members su Slack nel caso di problematiche

Inoltre, si terrà informato il Top Management attraverso Status Report previsti con cadenza bisettimanale.

### 6.4. Budget Control Plan

---

Il controllo del budget orario è previsto tramite la consegna bisettimanale del foglio di lavoro al Top Manager. Inoltre, si farà utilizzo di strumenti atti al calcolo dell'Earned Value, monitorando costantemente l'indice di performance dei costi (CPI), l'indice di performance dello schedule (SPI) ed altre metriche associate. Inoltre sarà effettuata la stima dei costi sulla piattaforma Microsoft Project insieme al Diagramma di GANTT con annessa allocazione delle risorse.

### 6.5. Quality Assurance Plan

---

Il processo di controllo della qualità avviene ogni volta che un team member consegna il proprio task su Trello. Successivamente, i Project Manager provvederanno a revisionare il task e lasciare il proprio feedback durante il meeting settimanale. Inoltre, ogni settimana i team member riceveranno una valutazione sulla qualità degli artefatti prodotti e sulla produttività. Le valutazioni verranno consegnate ai team member ogni settimana, in modo da permettergli di migliorarsi nelle settimane successive di lavoro. Per altre informazioni fare riferimento al documento di Quality Plan.





## 6.6. Project Closeout Plan

---

Il sistema sarà consegnato al Top Manager e al cliente quando questo sarà terminato. La data prevista è il 17 gennaio 2023. La consegna avverrà sulla piattaforma e-learning mediante un file .zip contenente l'intera documentazione ed il link al repository del progetto. Inoltre, è prevista una presentazione finale del progetto sia dai Project manager sia dai Team Members per descrivere tutta la documentazione.

## 7. Product Delivery

---

La consegna del prodotto avverrà sulla piattaforma e-learning e conterrà i seguenti documenti suddivisi per categoria:

- Documenti di management:
  - Statement of Work
  - Project Charter
  - Team Contract
  - Business case
  - Financial Analysis
  - SPMP
  - Schedule Management Plan
  - Scope Management
  - Quality Plan
  - Risk Management Plan
  - WBS Dictionary
  - WBS GANTT
  - Software Configuration Management Plan
  - Agende
  - Foglio di lavoro
  - Status Reports
  - Progetto Microsoft Project
  - Minute
- Documenti di Sviluppo:



- RAD
- SDD
- ODD
- Test Plan
- Test Case Specification
- Test Incident Report
- Test Summary Report
- Test Execution Report
- Test Integration Plan
- Matrice di tracciabilità
- Codice sorgente in una cartella compressa

## 8. Supporting Process Plan

---

### 8.1. Project Supervising & Work Environment

---

Gli Ambienti di lavoro durante le lezioni saranno principalmente aule appartenenti al Dipartimento di Informatica dove i Team Member si riuniranno per lavorare sulla documentazione e per effettuare i meeting settimanali previsti. Durante le festività invece l'ambiente di lavoro sarà spostato su Teams dove effettueranno la fase di Development e i meeting settimanali previsti da remoto.

La supervisione sarà effettuata settimanalmente durante i meeting dove ci si confronterà per conoscere l'andamento e su Slack per eventuali problematiche. Ogni documento ha il proprio supervisore che attraverso una checklist informerà i Project Manager cosa è stato portato e termine e cosa è in corso in modo da ottenere ulteriore feedback sulla documentazione. Infine la documentazione sarà revisionata dai Project Manager e sottoposta al Top Manager.

### 8.2. Decision Management

---

Tutte le decisioni relative al management del progetto saranno prese solo ed unicamente dai Project Manager. Tutte le decisioni riguardanti lo sviluppo verranno prese di comune accordo tra i Project Manager e i Team Member durante i meeting, nei quali ognuno potrà proporre la propria soluzione.



## 8.3. Risk Management

---

Verrà utilizzato un approccio suddiviso in 4 fasi:

- **Identificazione dei rischi:** questa fase ha come obiettivo l'individuazione dei possibili rischi.
- **Analisi dei rischi:** questa fase ha come obiettivo l'analisi dei rischi individuati nella fase precedente assegnando ad ognuno di essi una priorità e una probabilità che accadano.
- **Risk Planning:** questa fase ha come obiettivo la pianificazione di una eventuale azione nel caso in cui il rischio avvenga.
- **Risk Monitoring:** in questa fase si monitorano quelli che sono gli indicatori che ci porteranno a comprendere quanto il rischio individuato sta per avverarsi.

Per ciò che concerne la categorizzazione dei rischi, essi possono essere classificati come rischi relativi alle persone del team o rischi di management (per errori di stima o di organizzazione).

La tecnica adoperata per la rilevazione è quella del brainstorming.

Inoltre, ognuno di essi avrà:

- ID / rank;
- Nome;
- Probabilità che accada;
- Impatto sul progetto;

Per la probabilità di verifica si avranno le seguenti categorie:

- **Molto Bassa:** 0% - 20%;
- **Bassa:** 21% - 50%;
- **Media:** 51% - 80%;
- **Alta:** maggiore dell' 80%;

Per l'impatto sul progetto si avranno le seguenti categorie:

- **Insignificante:** non incide in alcun modo sul progetto;
- **Tollerabile:** rischio che impatta relativamente poco il progetto;
- **Grave:** rischio che potrebbe impattare in modo significativo il progetto;
- **Catastrofico:** rischio che impatta significativamente il progetto;

Per ogni rischio individuato si andrà ad effettuare un piano di prevenzione per ridurre la probabilità che avvenga ed un piano di contingenza da attuare se quel rischio si verifica.

Per altre informazioni fare riferimento al documento di Risk Management Plan.



## 8.4. Configuration Management

---

I Configuration Item rappresentano tutti i tipi di oggetti che sono coinvolti nel Configuration Control.

Gli elementi che fanno parte dei Configuration Item sono:

- Documentazione di progetto
- Software di terze parti con relativa documentazione
- Altre componenti, a discrezione del Project Manager

Ogni Configuration Item è caratterizzato da un identificativo univoco, definito dalla sigla del progetto, il nome del documento e la versione, secondo la seguente forma: “C17\_NomeDoc\_vX.Y”.

X.Y rappresenta la versione dell’Item e per tale caratteristica si fa utilizzo del versionamento semantico, dove:

- X rappresenta la versione MAJOR e viene aggiornata quando vi è il completamento dell’item o il raggiungimento di una milestone.
- Y rappresenta la versione MINOR e viene aggiornata con modifiche di minore entità.

Per informazioni più dettagliate fare riferimento al documento di Software Configuration Management Plan.

## 8.5. Quality Assurance

---

Per quanto riguarda la definizione di qualità del software si fa riferimento allo standard ISO/IEC 9126.

Questo si divide in quattro parti:

- Modello della qualità del software: classificato da sei caratteristiche generali e varie sotto-caratteristiche misurabili attraverso delle metriche;
- Metriche esterne per la qualità: misurano i comportamenti del software sulla base dei test, dell’operatività e dall’osservazione durante la sua esecuzione, in funzione degli obiettivi stabiliti;
- Metriche interne per la qualità: si applica al codice sorgente, durante la fase di codifica;
- Metriche per la qualità in uso: rappresenta il punto di vista dell’utente sul software e il livello di qualità in uso è raggiunto quando il livello di qualità esterna e interna sono raggiunti;

Per altre informazioni fare riferimento al documento di Quality Plan.



## 8.6. Measurement

---

I Team Members verranno valutati con cadenza settimanale secondo i seguenti criteri di valutazione:

- Proattività: verrà valutata la capacità di risoluzione dei Team Members alle problematiche;
- Qualità del lavoro: verrà valutata la qualità degli artefatti realizzati dai Team Members;
- Produttività: verrà valutata la produttività dei Team Members in base agli artefatti realizzati;
- Teamwork: verrà valutata la capacità dei Team Members di lavorare in gruppo per raggiungere obiettivi comuni di progetto, rispettando le scadenze;
- Comportamento ai meeting: verrà valutata la partecipazione e il comportamento ai meeting dei Team Members.

La votazione sarà espressa in una scala da 0 a 10. La valutazione finale verrà ottenuta attraverso la media delle votazioni ricevute durante l'arco di tutto il progetto.

Per quanto riguarda le metriche di dimensione e complessità del progetto, queste saranno:

- Lines of Code (LOC): utile per formalizzare la dimensione dell'applicativo tramite le linee di codice che lo compongono.
- Comment Lines of Code (CLOC): utile per ottenere il numero totale di linee di commenti scritti.

## 9. Additional Plans

---

In questo progetto non sono e non saranno necessari ulteriori piani oltre a quelli già specificati nel documento.