

# PROJECT PLAN

Versione 1: 18/10/2024

I principali componenti di un piano di progetto (**project plan**) sono:

## 1. **Introduzione:**

BGTransport è nato per facilitare l'utilizzo dei mezzi pubblici nella provincia di Bergamo. Questo progetto permette di visualizzare in modo chiaro, semplice e veloce, all'interno di una pagina web, tutte le linee di pullman, percorsi ferroviari, la funicolare e le linee tranviarie di Bergamo.

## 2. **Modello di processo:**

SCRUM: è stato scelto SCRUM perché ci permette di suddividere il lavoro e di darci degli obiettivi da soddisfare settimanalmente. Abbiamo utilizzato la sezione Projects all'interno di GitHub.

## 3. **Organizzazione del progetto:**

Development team: Maffeis Riccardo, Zanotti Matteo --> si occupano di portare a termine i compiti assegnati durante lo Sprint

## 4. **Standard, linee guida, procedure** I progetti software sono grandi progetti. Di solito, molte persone sono coinvolte. Occorre quindi una forte disciplina di lavoro, in cui ogni persona coinvolta segua gli standard, le linee guida e le procedure concordate. Oltre ad essere dichiarati sulla carta, molti di questi possono essere supportati o applicati da strumenti. Di estrema importanza sono gli accordi chiari sulla documentazione: quando consegnare la documentazione, come valutare la qualità della documentazione, come garantire l'aggiornamento della documentazione? In larga misura, questi standard e procedure saranno descritti in documenti separati, come il Piano di controllo della configurazione o il Piano di garanzia della qualità.

Per esempio, un progetto potrebbe dichiarare di seguire le convenzioni per la programmazione Java come definite da Oracle<sup>1</sup>.

Tutto il team deve seguire le seguenti linee guida:

- Tutti i bug/implementazioni devono essere segnati nella sezione Projects di GitHub e associati ad un Issue
- Quando vengono fatte delle modifiche a documenti deve essere creato un Issue per far sì che ogni componente del gruppo possa vedere le modifiche e approvare i cambiamenti
- Il software deve essere sempre testato dopo qualsiasi modifica/implementazione per accertarsi che le modifiche/implementazioni siano corrette
- Per qualsiasi idea mirata al miglioramento del software bisogna discuterne insieme a tutto il team
- Per qualsiasi problema bisogna discuterne insieme a tutto il team per risolvere le problematiche nel minor tempo possibile

5. **Attività di gestione** sono guidate da obiettivi e priorità fissati per il progetto. Ad esempio, la direzione dovrà presentare relazioni periodiche sullo stato e sullo stato di avanzamento del progetto. Dovrà anche seguire alcune priorità nel bilanciamento di requisiti, tempi e costi.
6. **Rischi** I potenziali rischi devono essere identificati il prima possibile. Ci saranno sempre dei rischi: l'hardware potrebbe non essere consegnato in tempo, il personale qualificato potrebbe non essere disponibile quando necessario, le informazioni critiche potrebbero mancare quando è necessario e così via. È piuttosto ingenuo supporre che un progetto di sviluppo software funzioni sempre senza intoppi. Anche in campi consolidati, come quello delle costruzioni, c'è sempre qualcosa che non va. Si dovrebbe diagnosticare precocemente i rischi di un progetto software e fornire misure per affrontarli; vedere anche [il capitolo 8](#) . Più incerti sono i vari aspetti del progetto, maggiori sono i rischi.

7. **Personale:**

Quantità di personale: 2

Maffeis Riccardo

Zanotti Matteo

Inizio del progetto: 18/10/2024

Fine del progetto:

Competenze: buone capacità di programmazione in java

8. **Metodi e tecniche** In questa sezione vengono forniti i metodi e le tecniche da utilizzare durante l'ingegneria dei requisiti, la progettazione, l'implementazione e le prove. In genere, qui viene descritto anche il modo in cui viene gestito il controllo della versione e della configurazione per i componenti software. Gran parte della documentazione tecnica sarà prodotta durante queste fasi. Si deve quindi precisare come sarà curata questa documentazione.
  9. **Garanzia di qualità** Quale organizzazione e procedure verranno utilizzate per garantire che il software in fase di sviluppo soddisfi i requisiti di qualità dichiarati? I molti aspetti di un Piano di Assicurazione della Qualità possono anche essere trattati in un documento separato. Il tema della garanzia della qualità è discusso nel [Capitolo 6](#) .
-

10. **Pacchetti di lavoro (workpackages)** I progetti più grandi devono essere suddivisi in attività, parti gestibili che possono essere allocate ai singoli membri del team. Ciascuna di queste attività deve essere identificata nel piano di progetto. La scomposizione gerarchica del progetto è rappresentata in una struttura di scomposizione del lavoro (vedi anche **Sezione 8.4** ).

11. **Risorse** Durante il progetto sono necessarie molte risorse. L'hardware, i cicli della CPU e gli strumenti necessari per supportare il progetto sono elencati in questa voce. Occorre inoltre indicare il personale necessario per le varie fasi del processo.

• **Software:**

Ambiente		di		sviluppo:		
IDE	per	lo	sviluppo	in	Java	(versione 2024-09)
Java		Development		Kit		(JDK):
Kit	di	sviluppo	Java	(versione	JDK	23)

• **Hardware**

Computer personali che dispongono delle caratteristiche sufficienti per lavorare con Eclipse, Papyrus, GitHub, ... (quantità: 2)

12. **Budget** e programma Il budget totale per il progetto deve essere assegnato alle varie attività come indicato nella struttura di ripartizione del lavoro. Anche le attività devono essere programmate in tempo, ad esempio utilizzando un grafico PERT (vedi **Sezione 8.4** ). In questa rubrica è indicato anche il modo in cui vengono tracciate le risorse e le altre spese. Il tema della stima dei costi e dei tempi è trattato ampiamente nel **Capitolo 7** .

13. **Cambiamenti** È stato affermato in precedenza che i cambiamenti sono inevitabili. Bisogna garantire che questi cambiamenti siano gestiti in modo ordinato. Sono quindi necessarie procedure chiare su come verranno gestite le modifiche proposte. Se il processo è agile, ogni iterazione comporta modifiche e queste vengono gestite in modo leggero. In realtà, non sono visti come cambiamenti. Nei processi più pesanti, ogni modifica proposta deve essere registrata e rivista. Quando una richiesta di modifica è stata approvata, è necessario stimarne l'impatto (costo). Infine, la modifica deve essere incorporata nel progetto. Le modifiche che vengono immesse tramite la porta sul retro portano a codice strutturato male, documentazione inadeguata e superamento di costi e tempi. Poiché le modifiche portano a versioni diverse sia della documentazione che del codice, le procedure da seguire per gestire tali modifiche vengono spesso gestite nel contesto di un piano di controllo della configurazione.

14. **Consegna** Devono essere indicate le procedure da seguire per la consegna dell'impianto al cliente.

La consegna prevista del progetto è stimata per la prima settimana di gennaio del 2025 e prevede la completa raggiunta degli obiettivi concordati ed un'eventuale risoluzione dei problemi affrontati in corso d'opera. Si stima che il progetto permetta a tutti gli utenti un utilizzo facilitato dell'interfaccia per semplificare la comprensione e l'utilizzo dei mezzi di spostamento pubblico nella provincia.