



Università
Ca' Foscari
Venezia



PIANO DI PROGETTO v1.0

Università Ca' Foscari di Venezia
Dipartimento di Scienze Ambientali, Informatica e
Statistica
Corso di Ingegneria del Software A.A. 2017-2018
Docente: Prof. Agostino Cortesi

 **Gruppo:**

Tusar Dhali 858901

Abdelmoughit Faris 853732

Gianfilippo Bellin 853418

Badr Wahmane 989121

Sommario

1.Introduzione	3
1.1 Overview del Progetto.....	3
1.2 Deliverables del Progetto.....	3
1.3 Evoluzione del Progetto.....	3
1.4 Materiale di riferimento.....	4
1.5 Definizioni e Abbreviazioni.....	4
2.Organizzazione del progetto.....	5
2.1 Modello del Processo.....	5
2.2 Struttura Organizzativa.....	6
2.3 Interfacce Organizzative.....	6
2.4 Responsabilità di Progetto.....	6
3. Processi gestionali.....	7
3.1 Obiettivi e Priorità.....	7
3.2 Assunzioni, Dipendenze, Vincoli.....	7
3.3 Gestione dei rischi.....	8
3.4 Meccanismi di monitoraggio e di controllo.....	9
3.5 Pianificazione dello staff.....	10
4. Processi tecnici.....	10
4.1 Strumenti e tecniche, metodi.....	10
4.2 Documentazione del software.....	10
4.3 Funzionalità di supporto al progetto.....	10
5. Pianificazione lavoro, budget e risorse umane.....	11
5.1 Work Breakdown Strutture.....	11
5.2 Dipendenze.....	12
5.3. Risorse necessarie.....	13
5.4. Allocazione Budget e Risorse.....	13
5.5. Pianificazione.....	14



1.Introduzione

1.1 Overview del progetto

Il progetto che svolgeremo consiste in un'applicazione Android volta a sfruttare gli open data. La realizzazione è prevista nel corso di Ingegneria del Software, tenuto dal Professore Agostino Cortesi nell'a.a. 2017/2018 presso l'università Ca'foscari. Abbiamo intenzione di usare dati aperti relativi alla posizione dei parcheggi sul territorio nazionale, in modo da sviluppare un'applicazione che faciliti il compito di trovare un parcheggio, visualizzando su una carta geografica i parcheggi più vicini alla propria posizione tramite GPS e guidare l'utente a destinazione. Abbiamo scelto tale App in quanto è un problema comune di tutti i giorni. Il nostro scopo è quello di facilitare la vita dei guidatori.

1.2 Deliverables del progetto

Gli items con data di consegna che saranno consegnati sono i seguenti:

1. Piano di Progetto (23/10/2017)
2. Documento di analisi e specifica (9/11/2017)
3. Piano di testing (23/11/2017)
4. Documento di Progettazione (15/12/2017)
5. Realizzazione e messa in linea (31/01/2018)

Saranno consegnati via mail al professore Agostino Cortesi.

1.3 Evoluzione del progetto

Il progetto si trova in una fase iniziale. Le funzionalità dell'applicazione sono già state definite e discusse. In futuro implementeremo delle migliorie e nuove funzioni. Alcune di quelle ipotizzate sono: poter aggiungere ai preferiti le proprie destinazioni e dare un voto ai parcheggi.

1.4 Materiali di riferimento

Lista dei documenti cui ci si riferisce nel Piano di Progetto:

- Libro di Ingegneria del Software: ROGER PRESSMAN, Principi di Ingegneria del software, McGraw-Hill, quinta edizione, 2008
- Slides del corso di Ingegneria del Software tenuto dal professore Agostino Cortesi
- Libro: Android 5 Programming Kyle Mew
- Libro: Android Programming For Beginners John Horton
- Documentazione Datadroid fornita dal tutor del corso di ingegneria Alvisè Spanò

1.5 Definizioni ed abbreviazioni

- **Android:** è un sistema operativo principalmente per dispositivi mobili sviluppato da Google Inc. e basato sul kernel Linux. Non è da considerarsi, tuttavia, propriamente un sistema unix-like o una distribuzione GNU/Linux, dal momento che la quasi totalità delle utilità GNU è sostituita da software in Java.
- **Derivables:** prodotti e servizi da rilasciare per rispondere agli obiettivi definiti nella pianificazione del progetto.
- **GPS:** (Global Positioning System) sistema di posizionamento e navigazione satellitare civile che, attraverso una rete dedicata di satelliti artificiali in orbita, fornisce ad un terminale mobile o ricevitore GPS informazioni sulle sue coordinate geografiche ed orario, in ogni condizione meteorologica, ovunque sulla Terra o nelle sue immediate vicinanze ove vi sia un contatto privo di ostacoli con almeno quattro satelliti del sistema. La localizzazione avviene tramite la trasmissione di un segnale radio da parte di ciascun satellite e l'elaborazione dei segnali ricevuti da parte del ricevitore.
- **Modello a cascata:** si tratta del più tradizionale modello usato per lo sviluppo del Software, che prevede una sequenza di fasi distinte, ciascuna delle quali produce un ben preciso output che viene utilizzato come input per la fase successiva.
- **Ingegneria del software** : (software engineering in inglese) si intende quella disciplina che si occupa dei processi produttivi e delle metodologie di sviluppo finalizzate alla realizzazione di sistemi software.
- **Java:** Java è un linguaggio di programmazione ad alto livello, orientato agli oggetti e a tipizzazione statica, specificatamente progettato per essere il più possibile indipendente dalla piattaforma di esecuzione.

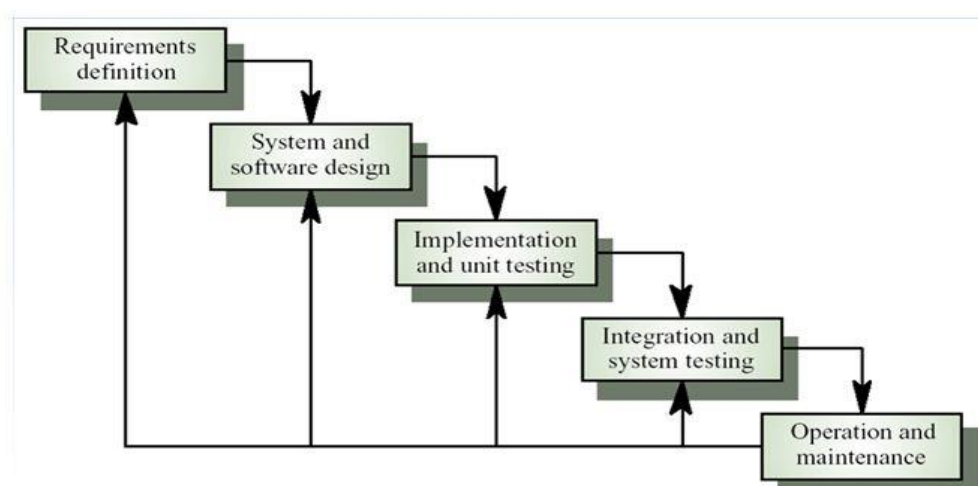
2. Organizzazione del progetto

2.1 Modello del processo

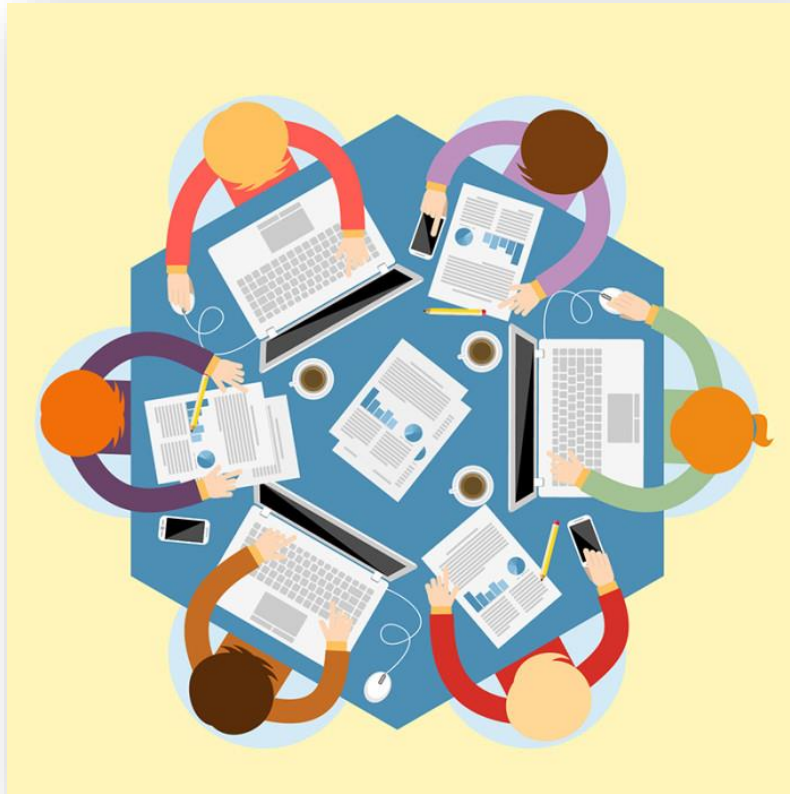
il modello del processo che andremo a seguire sarà il waterfall model ossia modello a cascata in quanto offre molti vantaggi, il modello è facile da capire e facile da gestire. A causa della rigidità del modello ogni fase presenta specifici risultati e un processo di revisione. In questo modello le fasi vengono elaborate e completate una alla volta. Le fasi non si sovrappongono. e inoltre Il modello Waterfall funziona bene per progetti più piccoli dove i requisiti sono ben compresi. ogni fase deve essere completata prima che la fase successiva possa iniziare. Questo tipo di modello di sviluppo software è fondamentalmente utilizzato per i progetti piccoli in cui non ci sono requisiti incerti. Al termine di ciascuna fase, viene effettuata una revisione per determinare se il progetto è sulla strada giusta e se continuare o scartare il progetto. In questo modello il test del software inizia solo dopo che lo sviluppo è completo.

Waterfall model

Il modello a cascata o ciclo di vita a cascata è un modello di ciclo di vita del software secondo cui la realizzazione di un prodotto software consta di una sequenza di fasi strutturata.



2.2 Struttura organizzativa



La struttura organizzativa che andremo a seguire sarà quella funzionale in cui tutte le risorse lavorano nella stessa area in quanto pensiamo che sia il modo migliore per lavorare e confrontarci a vicenda. E inoltre Funziona bene per piccole squadre e piccoli progetti e si il pieno controllo sui membri del team e sulle altre risorse necessarie. È possibile accedere facilmente agli esperti necessari perché sono nella stessa area. È più veloce ed efficiente il lavoro se ci troviamo tutti insieme per risolvere i problemi legati al progetto.

2.3 Interfacce organizzative

Le altre persone cui faremo riferimento per ottenere magari consigli e informazioni, giudizi riguardo l'applicazione che andremo a sviluppare saranno il professore del corso Agostino cortesi e il tutor Alvisè Spanò. Potremo riferirci anche ad altri conoscenti che magari sono più esperti in ambito di programmazione Android.

2.4 Responsabilità di progetto

Le responsabilità sono quelle di sviluppare un'applicazione efficiente user friendly e priva di bug e inoltre rispettare le scadenze di consegna descritte nei deliverable del progetto in caso di ritardi di consegna avviseremo in tempo il professor Agostino.



3. Processi gestionali

3.1 Obiettivi e priorità

L'obiettivo consiste nella creazione di un'applicazione funzionante per dispositivi Android che tramite la visualizzazione da mappa permetta di trovare il parcheggio più vicino all'utente e guidarlo fino ad esso.

Per quanto riguarda le priorità, abbiamo intenzione di:

- rispettare le scadenze stabilite, portando a termine in modo regolare ed efficiente i compiti assegnati
- raggiungere e mantenere coesione, collaborazione nel gruppo, per tutta la durata del progetto
- garantire un prodotto finale ben strutturato ed affidabile

3.2 Assunzioni, Dipendenze, Vincoli

Assumiamo che ciascun membro del gruppo svolga il proprio lavoro con responsabilità, contribuendo con il proprio bagaglio di conoscenze alla realizzazione dell'obiettivo finale.

Assumiamo inoltre, che gli strumenti necessari per lo sviluppo dell'applicazione siano accessibili ad ogni componente. La realizzazione del prodotto dipende dalla propensione nell'apprendere nuove abilità come l'utilizzo di nuovi strumenti di progettazione, nell'interagire in modo funzionale tra i membri del gruppo e nello sviluppare effettivamente del codice. I vincoli sono di tipo temporale, in quanto bisogna rispettare le scadenze, ed organizzativo, dal momento che è necessario conciliare il progetto con altre attività.

3.3 Gestione dei rischi

Il progetto è soggetto a possibili rischi che, se non affrontati nella maniera adeguata, potrebbero alterare la buona riuscita della realizzazione del progetto, portando, così, ad un fallimento. I rischi che abbiamo identificato sono i seguenti:

- **Mancanza di collaborazione:** ciò può essere dovuto a difficoltà comunicative o alla carenza di interesse ed impegno da parte di un membro del gruppo. Per questa ragione, è necessario avere una strategia da adottare nel caso si ponga tale problema: si può, per esempio, richiamare il soggetto inadempiente.
- **Abbandono da parte di un componente del gruppo:** può succedere che qualcuno si ritiri dal progetto, sconvolgendo, così, quanto stabilito nella pianificazione. Ciò, tuttavia, si può risolvere attuando una ripartizione del carico di lavoro fra i membri rimasti.
- **Problemi di salute:** anche in questo caso si può reagire come nel punto precedente, ovvero provvedendo ad una ripartizione del carico di lavoro.
- **Malfunzionamento dei sistemi informatici:** è possibile che si presentino dei problemi relativi al funzionamento non corretto dei sistemi utilizzati nella realizzazione del progetto. Questa complicazione è risolvibile cambiando i mezzi informatici impiegati: è possibile, ad esempio, nel caso in cui un computer personale non funzioni, ricorrere alle macchine presenti nei laboratori dell'università.
- **Difficoltà di utilizzo degli strumenti:** può capitare che i componenti del gruppo riscontrino difficoltà nell'impiego dei mezzi disponibili. E' necessario, in tal caso, colmare queste lacune conoscitive mediante lo studio e l'approfondimento dei concetti mancanti.
- **Difficoltà nello sviluppo delle componenti di progetto:** è possibile che, a causa di un calcolo errato dei tempi e, in particolare, della fattibilità di certe componenti, si abbiano delle difficoltà nello sviluppo di alcune parti del progetto. E' necessario, in tale evenienza, ricercare le soluzioni e gli algoritmi appropriati.
- **Perdita dei dati:** non è da escludere la possibilità di perdita di materiale. Per questa ragione, è bene prevenire un simile imprevisto provvedendo a backup regolari dei dati prodotti nel corso della realizzazione del progetto.

Tabella dei rischi:

	Rischio	Probabilità	Impatto	Azione
1	Mancanza di collaborazione	Basso	Alto	Richiamo del componente
2	Abbandono da parte di un componente del gruppo	Molto basso	Medio	Ripartizione del carico di lavoro
3	Problemi di salute	Medio	Basso	Ripartizione del carico di lavoro
4	Malfunzionamento dei sistemi informatici	Medio	Basso	Sostituzione del mezzo
5	Difficoltà di utilizzo degli strumenti	Basso	Basso	Studio personale
6	Difficoltà nello sviluppo delle componenti di progetto	Medio	Medio	Ricerca di una soluzione
7	Perdita dei dati	Basso	Molto alto	Backup frequenti

Classificazione dei rischi:

Molto alto		7			
Alto		1			
Medio	2		6		
Basso		5	3-4		
Molto basso					
Impatto	Molto basso	Basso	Medio	Alto	Molto alto
Probabilità					

3.4 Meccanismi di monitoraggio e di controllo

Durante ogni fase del progetto verranno svolte delle attività di controllo in cui ogni membro del gruppo potrà esaminare la correttezza e la qualità del lavoro svolto dagli altri. Si tratterà inoltre di un momento per scambiare nuove idee, proposte di miglioramento, spiegare le difficoltà incontrate e proporre soluzioni ai problemi. Queste attività devono avvenire preferibilmente in maniera diretta con incontri periodici, o tramite strumenti di messaggistica istantanea. Le decisioni e le linee guida da seguire durante il progetto verranno prese in modo democratico.

3.5 Pianificazione dello staff

Per portare a termine il progetto è necessario avere conoscenza e padronanza della programmazione ad oggetti (Java) e dell'ambiente di sviluppo Android. Queste abilità sono consolidate dai componenti del gruppo grazie a precedenti corsi universitari frequentati o sono acquisibili attraverso la ricerca personale.

4. Processi tecnici

4.1 Strumenti e tecniche, metodi

Ogni membro del gruppo di lavoro sarà attivo in ogni fase del progetto e lavorerà simultaneamente su diversi aspetti, in modo da diminuire i tempi di completamento dell'applicazione. I documenti saranno condivisi tramite Dropbox, in modo da condividere senza problemi il proprio lavoro e di visualizzare quello degli altri componenti. Ogni membro del gruppo punta ad un lavoro corretto di comunicazione e gestione delle diverse prospettive del progetto. L'applicazione verrà realizzata con l'utilizzo della programmazione ad oggetti, tramite Java. Ogni membro lavorerà sul proprio computer, con un proprio Sistema Operativo (Windows, Linux). Gli strumenti che noi utilizzeremo sono i seguenti:

- ❖ Software per lo sviluppo di applicazioni in ambiente Android (**Android Studio**)
- ❖ Software per la gestione della documentazione

4.2 Documentazione del software

La documentazione verrà messa per iscritto in base alle scadenze indicate nella sezione Deliverables del Progetto presente al punto 1.2. dal momento che siamo solo all'inizio del nostro progetto. Il piano di progetto verrà consegnato tramite email.

4.3 Funzionalità di supporto al progetto

➤ Pianificazione della qualità

La qualità del progetto è direttamente dipendente dalle nostre idee, le quali puntano a un lavoro corretto, affidabile, riusabile e facile all'uso.

➤ Pianificazione della gestione delle configurazioni

L'applicazione dovrà essere in grado di consentire future revisioni ed eventuali modifiche. Il codice verrà inoltre commentato adeguatamente in modo da facilitare il lavoro di gruppo.



5. Pianificazione lavoro budget e risorse umane

5.1 Work Breakdown Strutture

1. Specifiche del progetto

1.1 Presentazione del progetto

- 1.1.1. Definizione dell'ambito di progetto
- 1.1.2. Definizione degli obiettivi generali
- 1.1.3. Analisi dei requisiti
- 1.1.4. Definizione delle conoscenze specifiche software richieste

1.2. Stesura documento di progettazione

- 1.2.1. Definizione delle fasi principali del progetto
- 1.2.2. Definizione del modello generale da seguire

1.3. Definizione team di progetto

- 1.3.1. Definizione dei ruoli
- 1.3.2. Definizione delle responsabilità individuali

1.4. Pianificazione dei tempi

1.5. Analisi dei rischi

- 1.5.1. Individuazione dei rischi
- 1.5.2. Definizione del piano di gestione dei rischi
- 1.5.3. Studio delle probabilità

1.6. Analisi dei costi

- 1.6.1. Definizione dei costi materiali
- 1.6.2. Definizione dei costi da lavoro

1.7. Pianificazione della qualità

- 1.7.1. Definire gli standard di qualità
- 1.7.2. Uso di procedure per garantire la qualità
- 1.7.3. Monitoraggio del feedback

2. Architettura applicazione

2.1. Specifiche funzionali

- 2.1.1. Modello processo
- 2.1.2. Interfaccia utente

2.2. Specifiche tecniche

- 2.2.1. Definizione delle specifiche della piattaforma

3. Sviluppo applicazione

3.1. Progettazione Software

3.1.1. Implementazione interfaccia grafica

3.1.2. Implementazione del codice

3.1.2.1. Alpha testing

3.1.2.2. Correzione bug

3.2. Progettazione hardware

4. Test Applicazione

4.1. Configurazione dell'ambiente di prova

4.2. Test software

4.3. Test

5. Rilascio Applicazione

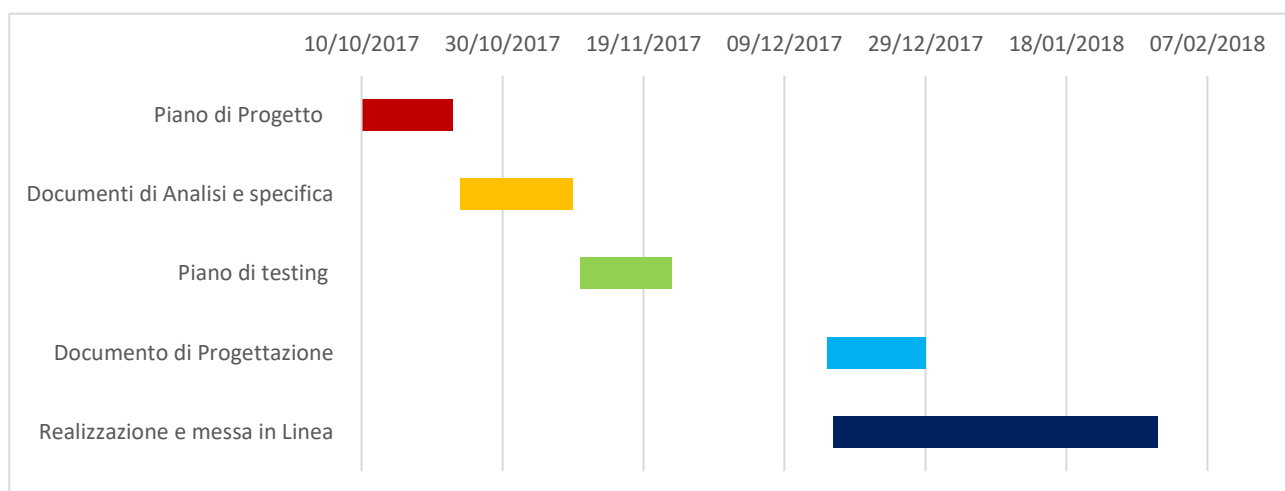
5.1. Rilascio versione Beta

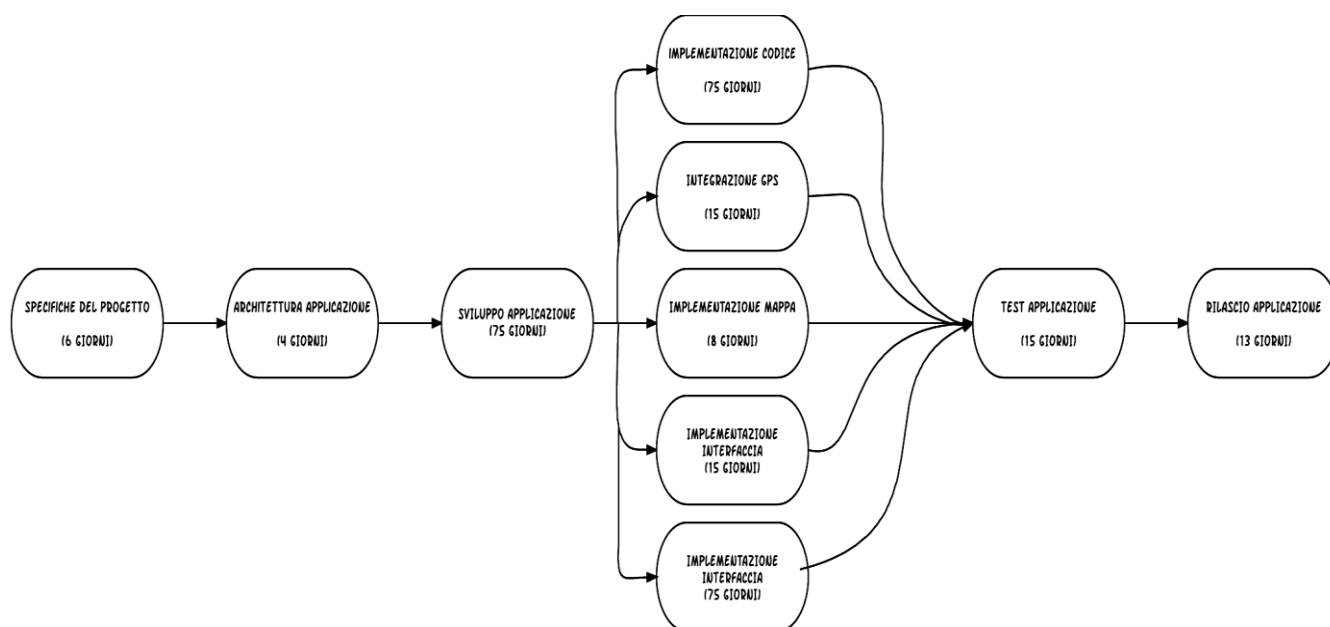
5.2. Correzione dei bug

5.3. Rilascio versione definitiva

5.2 Dipendenze:

Progetto in Ingegneria Gantt	Data di inizio	Data di fine
Piano di Progetto	10/10/2017	23/10/2017
Documenti di Analisi e specifica	24/10/2017	09/11/2017
Piano di testing	10/11/2017	23/11/2017
Documento di Progettazione	15/12/2017	24/12/2017
Realizzazione e messa in Linea	16/12/2017	31/01/2018





5.3. Risorse necessarie

Le risorse necessarie per lo sviluppo di un progetto consistono in persone, materiali, conoscenze e tempo. I gruppi di lavoro si trovano, spesso, a dover lavorare con un numero ristretto di risorse. Il nostro progetto sarà, sviluppato dai 4 elementi del gruppo e le risorse tecniche necessarie sono i Pc, "Android Studio" ed un ambiente di sviluppo integrato su piattaforma Android.

Ogni membro del gruppo, inoltre, lavorerà al progetto impiegando le proprie conoscenze riguardanti il linguaggio di programmazione Java e il sistema operativo per dispositivi mobili Android. Queste conoscenze saranno affiancate dalla capacità di strutturare ed elaborare la documentazione del progetto. Per quanto riguarda il tempo, come risorsa essenziale, il gruppo si impegna a seguire un piano ben definito con scadenze prefissate. È importante, infatti, fare una stima accurata del tempo necessario per una buona gestione del progetto.

5.4. Allocazione Budget e Risorse

Saper gestire i costi di un progetto costituisce un impegno significativo per ogni membro del gruppo. Definire il budget di progetto richiede, in particolare, la capacità di analizzare in dettaglio il lavoro da svolgere prima che ogni attività venga iniziata, in modo da avere un quadro complessivo delle risorse necessarie per raggiungere gli obiettivi prefissati. Occorre, inoltre, considerare i rischi associati allo sviluppo del progetto per poter configurare un budget di "emergenza", che andrà a sommarsi a quello già definito per sostenere l'intero progetto. Poiché questo progetto è sviluppato da studenti universitari, non verrà definito un budget vero e proprio, ma saranno ben utilizzate le risorse a disposizione.

5.5. Pianificazione

Sono state prefissate scadenze intermedie, per poter monitorare l'avanzamento del progetto.

Queste comprendono i termini di consegna della documentazione:

- Piano di Progetto (23/10/2017)
- Documento di analisi e specifica (9/11/2017)
- Documento di Progettazione (15/12/2017)
- Piano di testing (23/11/2017)
- Realizzazione e messa in linea (31/01/2018)

Inoltre, sono stati pianificati dei test che riguardano le singole componenti dell'applicazione. Ogni volta che verrà realizzata una parte software si eseguirà il relativo test per verificarne la correttezza.

Alla fine della realizzazione dell'App verrà fatto un test globale finale per verificare il funzionamento di tutte le parti in collaborazione tra loro.