



Università Ca' Foscari di Venezia

Dipartimento di Scienze Ambientali,
Informatica e Statistica

Corso di Ingegneria del Software
A.A. 2016-2017

Docente: Prof. Agostino Cortesi

PIANO DI PROGETTO

Data di consegna: 15/03/17

Versione 1.1



MASSIVE DYNAMICS

Paolo Vedorin
Via San Pelajo 139
31100 Treviso

345.8052880

849942@stud.unive.it

Mariagiovanna Czarnecki
Via per Cavriè 34
31030 Breda di Piave

346.0262298

854098@stud.unive.it

Dora Pavan
Strada delle Acquette 19
31100 Treviso

347.0358079

860980@stud.unive.it

Sommario

1 INTRODUZIONE	3
2 ORGANIZZAZIONE DEL PROGETTO	6
3. PROCESSI GESTIONALI	8
4. PROCESSI TECNICI.....	11
5. PIANIFICAZIONE LAVORO, BUDGET E RISORSE UMANE.....	12

1 INTRODUZIONE

1.1 OVERVIEW DEL PROGETTO

Il progetto che abbiamo intenzione di presentare, la cui realizzazione è prevista nel corso di Ingegneria del Software, svolto durante il Corso di Laurea in Informatica presso l'Università Ca' Foscari di Venezia e tenuto dal Professore Agostino Cortesi, nell'a.a. 2016/2017, consiste in un'applicazione Android volta a sfruttare gli open data europei, nazionali e regionali. Nel nostro specifico caso, prendiamo in considerazione i dati concernenti la Regione Veneto, in modo da sviluppare un'applicazione che faciliti la pianificazione di escursioni naturalistiche nella nostra regione, visualizzando, su una carta geografica del Veneto, le zone di interesse escursionistico, quali aree pic-nic, campeggi e aree naturali.

Tale scelta è dovuta alla mancanza di un'app che effettui la ricerca di questi elementi congiunti e dia, quindi, una visione complessiva dei luoghi in cui stare a contatto con la natura.

Lo scopo dell'applicazione è di aiutare l'utente nella pianificazione di un'escursione naturalistica, invogliandolo, così, ad una vicinanza più stretta con l'ambiente.

A tal fine, abbiamo pensato di creare una schermata in cui inserire la mappa del Veneto nella quale rendere selezionabili le varie province e poter visualizzare, in questo modo, i luoghi di interesse in maniera più ravvicinata. Premendo sui punti mostrati, inoltre, è possibile avere una visuale delle informazioni relative a ciascuno di essi e svolgere altre azioni.

1.2 DELIVERABLES DEL PROGETTO

Elenchiamo qui di seguito i vari items che verranno pubblicati man mano nella pagina del gruppo Massive Dynamics presente nel sito del corso di Software Engineering:

- Piano di Progetto (21/10/2016)
- Documento di analisi e specifica (11/11/2016)
- Documento di Progettazione (10/12/2016)
- Piano di testing (20/12/2016)
- Realizzazione e messa in linea (15/03/2017)

1.3 EVOLUZIONE DEL PROGETTO

Dal momento che ci troviamo ancora nella fase iniziale del progetto, probabilmente ulteriori funzionalità verranno stabilite in seguito ed alcune di quelle già previste potrebbero venire modificate o migliorate. Noi, comunque, ci impegneremo in modo da sviluppare i concetti proposti nella prima parte di questa introduzione ed approfondiremo successivamente la loro descrizione.

L'implementazione è ancora da definire, ma sarà oggetto di testing e verrà sicuramente trattata in future versioni del piano di progetto e nel resto della documentazione che sarà presentata.

1.4 MATERIALI DI RIFERIMENTO

In questa sezione elenchiamo i documenti a cui faremo riferimento nel piano di progetto:

- Slides del corso di Ingegneria del Software del Professor Cortesi
- Libro riguardante la programmazione di applicazioni Android: Android 5 Programming by Example, Kyle Mew, Packt Publishing Ltd.
- Libro di Ingegneria del Software: ROGER PRESSMAN, Principi di Ingegneria del software, McGraw-Hill, quinta edizione, 2008
- Sito web per sviluppatori Android: <https://developer.android.com/training/index.html>

1.5 DEFINIZIONI ED ABBREVIAZIONI

- **Android:** è un sistema operativo 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.
- **App:** dicitura abbreviata per indicare un'applicazione software, sia ludica che di utilità, per dispositivi mobili quali smartphone, palmari e tablet.
- **Database:** collezione di dati, tra loro correlati, utilizzati per rappresentare una porzione del mondo reale. I dati sono strutturati in modo tale da consentire la loro gestione in termini di inserimento, aggiornamento, ricerca e cancellazione delle informazioni.
- **Derivables:** prodotti e servizi da rilasciare per rispondere agli obiettivi definiti nella pianificazione del progetto.
- **Diagramma di Gantt:** strumento utilizzato per modellizzare la pianificazione dei task necessari alla realizzazione di un progetto. Si tratta di uno strumento inventato nel 1911 da Henry L. GANTT. È strutturato a partire da un asse orizzontale che rappresenta l'arco temporale totale del progetto, suddiviso in fasi incrementali (ad esempio, giorni, settimane, mesi) e da un asse verticale che raffigura le mansioni o attività che costituiscono il progetto.
- **Diagramma di Pert:** (Program Evaluation and Review Technique) diagramma reticolare che evidenzia il cammino logico di un progetto, identificando i collegamenti esistenti tra le diverse attività, permettendo di monitorare i punti critici che possono condizionare i risultati 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.
- **Java:** linguaggio di programmazione orientato agli oggetti a tipizzazione statica, specificatamente progettato per essere il più possibile indipendente dalla piattaforma di esecuzione.
- **Milestone:** termine che significa pietra miliare e viene utilizzato soprattutto in discipline tecniche come ingegneria del software o project management. Indica importanti traguardi intermedi nello svolgimento del progetto.
- **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.
- **OpenStreetMap:** è un progetto collaborativo finalizzato a creare mappe a contenuto libero del mondo. Il progetto punta ad una raccolta mondiale di dati geografici, con scopo principale la creazione di mappe e cartografie.

2 ORGANIZZAZIONE DEL PROGETTO

2.1 MODELLO DEL PROCESSO

Il modello che andremo ad utilizzare per la realizzazione del nostro software è quello “a cascata” (“waterfall” in inglese), che prevede uno studio preciso sulla fattibilità della realizzazione del programma da noi ideato, determinando la strada da intraprendere nelle varie fasi di sviluppo.

A seguire, vi sarà l’analisi dei requisiti, attraverso la quale andremo a determinare cosa farà la nostra applicazione.

Determinato tutto questo, analizzeremo come il nostro software andrà sviluppandosi fase per fase.

Una volta affrontata questa fase, tratteremo lo sviluppo vero e proprio del programma, discutendo punto per punto le scelte di programmazione che hanno portato alla creazione del codice rispetto ad altre.

Ultimato il codice, seguiranno delle prove di verifica, dei test, che ci porteranno all’analisi della correttezza del nostro software.

Ognuna di queste fasi dovrà essere seguita in maniera precisa e completa, cercando di evitare incongruenze e analizzando qualsiasi tipo di problema che potrebbe presentarsi, permettendo di avere così un modello chiaro e attuabile.

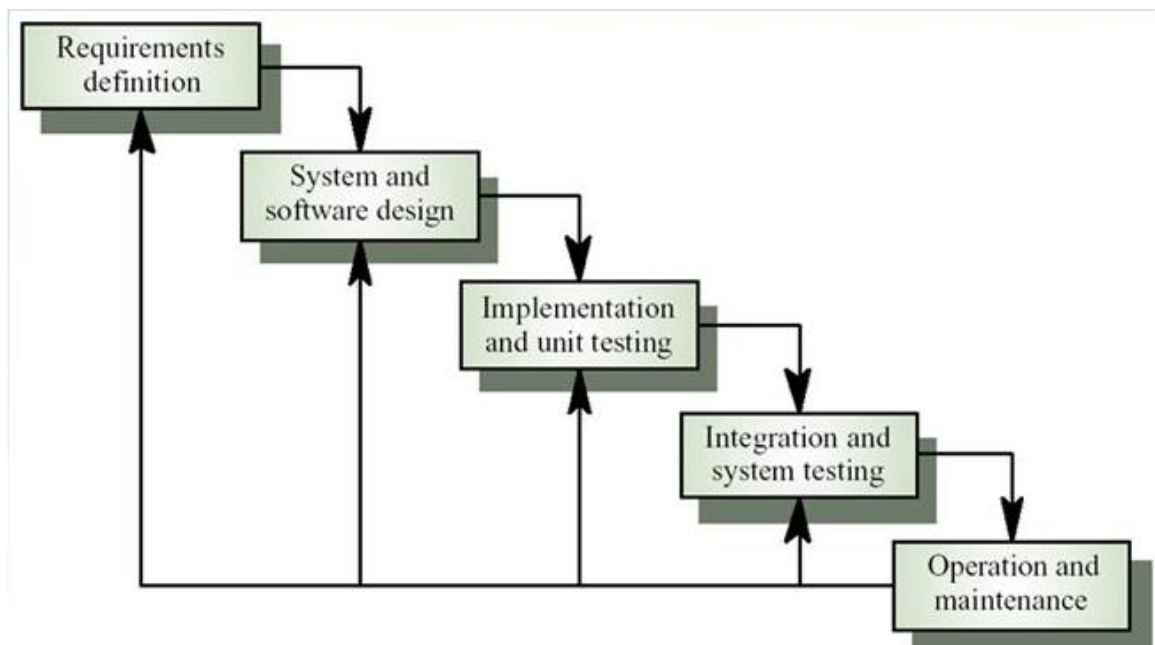


Figura 1. Immagine relativa al modello “a cascata”

2.2 STRUTTURA ORGANIZZATIVA

Il nostro gruppo sarà composto da individui con pari potere decisionale. Ogni aggiunta o modifica verrà esaminata insieme, valutando la fattibilità, quindi i pro e i contro, delle stesse. Questo permetterà una maggiore unità di gruppo, evitando situazioni in cui un individuo possa abusare del potere conferitogli, garantendo quindi un fluire di idee e proposte continuo e più spontaneo.

La garanzia di potersi relazionare in maniera diretta e senza preoccupazioni di alcun tipo ci permetterà, inoltre, di affrontare eventuali problemi che si potrebbero presentare nella compilazione del codice e di porvi un rimedio. La scelta di un tipo di gerarchia democratica decentralizzata, faciliterà, oltretutto, la risoluzione di eventuali problemi insiti al gruppo: qualora uno dei componenti non lavorasse quanto dovrebbe, si procederà ad un richiamo nei suoi confronti, invitandolo a portare a termine i punti assegnatigli nei tempi stabiliti. Nel momento in cui questo non dovesse funzionare, si procederà con l'espulsione dell'individuo tramite una votazione democratica. In caso di pareggio si procederà comunque al suo allontanamento definitivo, ritenendolo non più meritevole di fiducia.

Nel caso in cui uno dei componenti dovesse registrare un ritardo nella consegna del proprio lavoro per motivazioni che vanno oltre le proprie facoltà decisionali, si procederà alla distribuzione equa del lavoro tra i componenti del gruppo fino a cessate necessità.

2.3 INTERFACCE ORGANIZZATIVE

Ogni individuo sarà libero di informare i suoi colleghi riguardo eventuali conoscenti che posseggano competenze nell'ambito dell'informatica e di design di interfacce, ai quali poter chiedere consigli e/o giudizi circa l'applicativo che andremo ad implementare, in particolar modo l'interfaccia, l'usabilità e la praticità.

La comunicazione fra i membri del gruppo avverrà tramite messaggistica istantanea ed interazione fisica. Oltretutto, faremo anche riferimento al docente del corso, Agostino Cortesi.

2.4 RESPONSABILITÀ DI PROGETTO (DA MODIFICARE IN ULTIMA)

Ognuno dei componenti seguirà un programma stabilito in sede di riunione in cui verranno discussi i punti da sviluppare definiti in maniera scrupolosa. Al momento, le scadenze sono le seguenti:

- Consegna del piano di progetto (21/10/2016)
- Documento di analisi e specifica (2/11/2016)
- Piano di testing (15/11/2016)
- Documento di progettazione (10/12/2016)
- Realizzazione e messa in linea (15/03/2017)

Qualora si dovesse sfiorare nelle consegne, ci si rivolgerà al professor Agostino Cortesi, avvisandolo debitamente in anticipo che il gruppo ha registrato un ritardo nell'ultimazione dei punti e fissando con lo stesso un nuovo termine della consegna.

3. PROCESSI GESTIONALI

3.1 OBIETTIVI E PRIORITÀ

L'obiettivo consiste nella creazione di un'applicazione funzionante per dispositivi Android che permetta la visualizzazione da mappa di zone pic-nic, campeggi e aree naturalistiche presenti nella regione Veneto. Per quanto riguarda le priorità, abbiamo intenzione di:

- rispettare in modo puntuale le scadenze stabilite, portando a termine in modo regolare ed efficiente i compiti assegnati
- raggiungere e mantenere nel gruppo, per tutta la durata del progetto, coesione interna e collaborazione
- 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 di ogni membro ad apprendere nuove abilità come l'utilizzo di nuovi strumenti di progettazione.

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 universitarie.

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.

	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

Figura 2. Tabella dei rischi

Molto alto		7				
Alto		1				
Medio	2		6			
Basso		5	3 - 4			
Molto basso						
Imp	Prob	Molto Basso	Basso	Medio	Alto	Molto alto

Figura 3. Classificazione dei rischi

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 sono necessarie conoscenze relative alla programmazione orientata agli oggetti (Java), conoscenza dell'ambiente di sviluppo Android, conoscenza delle basi di dati. Queste abilità sono consolidate dai componenti del gruppo o sono facilmente acquisibili.

4. PROCESSI TECNICI

4.1 METODI, STRUMENTI E TECNICHE

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 redatti tramite Google docs, in modo da condividere senza problemi il proprio lavoro e di visualizzare quello degli altri componenti. Ognuno di noi membri punta ad un lavoro corretto di comunicazione e gestione delle diverse prospettive del progetto.

L'applicazione verrà realizzata sul paradigma della programmazione ad oggetti, tramite Java. Ogni membro lavorerà sulla propria macchina, 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 redatta successivamente 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. Renderemo disponibili tali documenti nella pagina relativa al nostro gruppo, "Massive Dynamics", e reperibile tramite il seguente link:

http://blogs.unive.it/groups/software_engineering_2012/wiki/ee041/Massive_Dynamics.html

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 WBS (Work Breakdown Structure)

1. Specifiche del progetto

1.1 Presentazione del progetto

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

1.2. Stesura documento di progettazione

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

1.3. Definizione team di progetto

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

1.4. Pianificazione dei tempi

1.5. Analisi dei rischi

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

1.6. Analisi dei costi

- 1.6.1. Definizione dei costi riguardanti gli strumenti
- 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 metodi e procedure per garantire la qualità
- 1.7.3. Monitoraggio dei risultati

2. Architettura applicazione

2.1. Specifiche funzionali

2.1.1. Modello di 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. Scelta della versione di Android

3.1.2. Implementazione interfaccia grafica

3.1.3. Implementazione codice

3.1.3.1. Test di rilascio

3.1.3.2. Correzione bug

3.2. Progettazione hardware

4. Prova Applicazione

4.1. Configurazione dell'ambiente di prova

4.2. Prova software

4.3. Prova hardware

5. Rilascio Applicazione

5.2. Dipendenze



Figura 4. Diagramma di Gantt

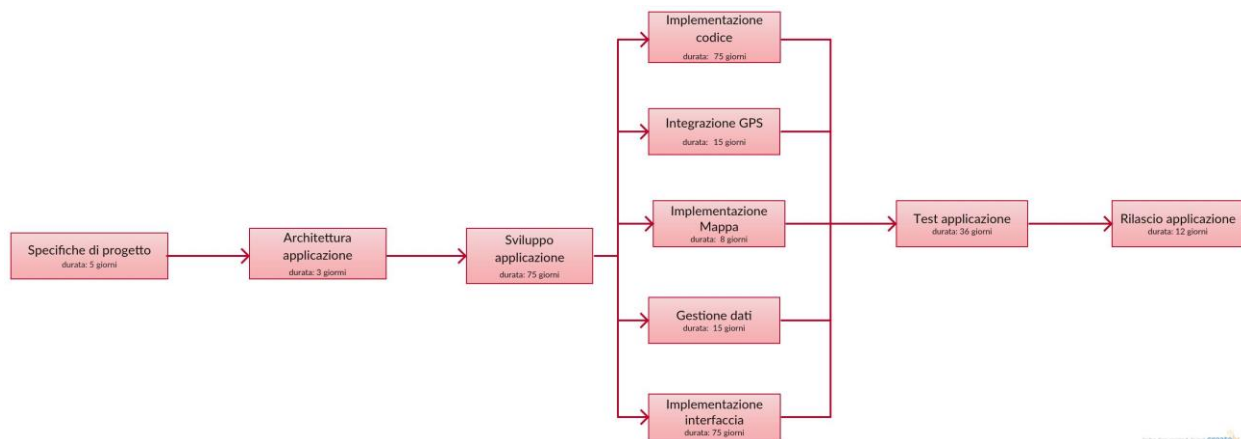


Figura 5. Diagramma di Pert

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, infatti, sviluppato da 3(tre) persone e le risorse tecniche necessarie si limitano all'uso di computer e di "Android Studio", un ambiente di sviluppo integrato per lo sviluppo 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 dalle 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 sarà 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 delle milestones, che rappresentano scadenze intermedie, per poter monitorare l’avanzamento del progetto. Queste comprendono i termini di consegna della documentazione:

- Piano di Progetto (21/10/2016)
- Documento di analisi e specifica (11/11/2016)
- Documento di Progettazione (10/12/2016)
- Piano di testing (20/12/2016)
- Realizzazione e messa in linea (15/03/2017)

Sono state, inoltre, previste ulteriori milestones:

- Specifiche di progetto (25/10/2016)
- Architettura Applicazione (28/10/2016)
- Sviluppo Applicazione (11/01/2017)
- Test Applicazione (16/02/2017)
- Rilascio Applicazione (15/03/2017)