



Università
Ca' Foscari
Venezia

Piano di Progetto



Gruppo: The Alliance

Componenti:

- Guastavino Antonio
- Tarquini Vladimyr
- Toffolo Matteo
- Zennaro Andrea

Link pagina del progetto

http://blogs.unive.it/groups/software_engineering_2012/wiki/92625/The_Alliance.html

Indice

1. Introduzione

- 1.1 Overview del Progetto
- 1.2 Deliverables del Progetto
- 1.3 Evoluzione del Progetto
- 1.4 Materiale di riferimento
- 1.5 Definizioni e Abbreviazioni

2. Organizzazione del Progetto

- 2.1 Modello del Processo
- 2.2 Struttura Organizzativa
- 2.3 Interfacce Organizzative
- 2.4 Responsabilità di Progetto

3. Descrizione dei Processi Gestionali

- 3.1 Obiettivi e Priorità
- 3.2 Assunzioni, Dipendenze, Vincoli
- 3.3 Gestione dei rischi
- 3.4 Meccanismi di monitoraggio e di controllo
- 3.5 Pianificazione dello staff

4. Descrizione dei Processi Tecnici

- 4.1 Metodi, Strumenti e Tecniche
- 4.2 Documentazione del Software
- 4.3 Funzionalità di supporto al progetto

5. Pianificazione del lavoro, delle risorse umane e del budget

- 5.1 WBS (Work breakdown structure)
- 5.2 Dipendenze
- 5.3 Risorse Necessarie
- 5.4 Allocazione del Budget e delle Risorse
- 5.5 Pianificazione

1.1 Overview del Progetto

La nostra proposta di progetto consiste nello sviluppo di un'app che sia utile a tutti gli iscritti all'università della Ca' Foscari.

L'idea è di fornire agli studenti, principalmente a quelli che abitano fuori Venezia, informazioni sulle "calli/callette" che sono sommerse durante l'acqua alta e contemporaneamente fornire percorsi alternativi per arrivare alla sede Ca' Foscari desiderata.

I dati delle posizioni delle sedi e quelle sulle maree sono liberamente consultabili in rete, invece per trovare i dati riguardanti le planimetrie di Venezia abbiamo dovuto contattare un'agenzia Venice Project Center (VPC).

1.2 Deliverables del Progetto

| OGGETTO | DESCRIZIONE | DATA CONSEGNA |
|--|--|---------------|
| Piano di Progetto versione 1.0 | Consegna della prima versione del piano di progetto | 21/10/2016 |
| Documento di analisi e specifica dei requisiti | Consegna della prima versione del documento dei requisiti | 2/11/2016 |
| Piano di testing | Consegna della prima versione del piano di testing | 15/11/2016 |
| Documento di Progettazione | Consegna della prima versione del documento di progettazione | 10/12/2016 |
| Realizzazione e messa in linea | Pubblicazione di una versione stabile dell'app | 28/02/2017 |

1.3 Evoluzione del Progetto

L'evoluzione del nostro progetto prevede tre fasi:

- Progettazione dell'app;
- Sviluppo del codice:
 - layout grafico;
 - implementazione dell'algoritmo di instradamento;
 - elaborazione open data;
 - utilizzo di API di Google Maps.
- Stesura documentazioni richieste;

Le varie macro aree verranno sviluppate in modo incrementale, revisionando man mano il piano di progetto.

1.4 Materiale di riferimento

- Slide del corso di ingegneria del software a.a. 2016/2017
http://blogs.unive.it/groups/software_engineering_2012/wiki/e0305/Lucidi_Lezioni_20162017.html
- Guida sito ufficiale Android Studio
<https://developer.android.com/index.html>
- API google maps
<https://enterprise.google.com/intl/it/maps/products/mapsapi.html>
- Forum vari

1.5 Definizioni e Abbreviazioni

| | |
|------------------|---|
| VPC | Venice Projects Center |
| App | Si intende l'app descritta nel piano di progetto. |
| Android | Sistema operativo per dispositivi mobili sviluppato da Google. |
| Open data | Sono dati liberamente accessibili a tutti, raccolti da altre persone. |
| Utente | Persona fisica che utilizza l'app. |

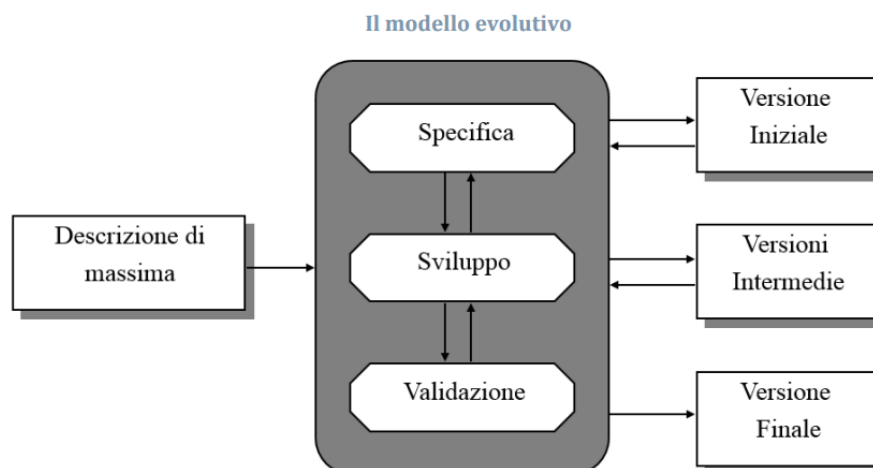
2.1 Modello del Processo

Premesse:

- L'app da sviluppare è di complessità medio / alta;
- Il team non ha mai lavorato su Android Studio;
- Il team non ha mai realizzato un'app mobile.

Sulla base di tali presupposti a causa dell'inesperienza generale del gruppo è stato preferito l'adozione del modello di tipo evolutivo.

Il team ha scelto questo modello perché si basa sulla prototipizzazione, facilitando così l'ottenimento di feedback sullo sviluppo in corso e permettendo di apportare modifiche al progetto, fino alla realizzazione dell'app finale, senza aumentare eccessivamente i costi, valutati in tempo di lavoro.



2.2 Struttura Organizzativa

Il team ha deciso che la sua organizzazione sarà di tipo democratico decentralizzato.

Questa organizzazione è sconsigliata in caso di un gran numero di componenti (problema di scalabilità), ma siccome siamo un piccolo gruppo di sole 4 persone (probabilmente per tutta la durata del progetto) pensiamo sia la scelta migliore per migliorare la comunicazione tra di noi.

Il ruolo del leader è ricoperto dall'intero team, che per ogni decisione da prendere si affiderà all'esito di una votazione tra tutti i membri.

Nel gruppo non ci saranno dei ruoli specifici, in base alla necessità ogni membro potrà essere assegnato a compiti diversi.

Nel caso in cui un membro del gruppo risultasse più competente a svolgere determinate attività, il resto del team può decidere di affidargli un potere decisionale maggiore solo in quella determinata area di competenza.

2.3 Interfacce Organizzative

Durante la realizzazione del progetto potrà risultare necessaria la comunicazione con entità esterne al team, per motivi quali richieste di informazioni o feedback.

Le entità con cui si prevedono possibili relazioni sono:

- Il forum di supporto degli sviluppatori di Android;
- Prof. A. Cortesi per feedback;
- Contatto VPC
- Campione di possibili utenti (tecnici) che fungano da tester per la versione beta dell'app (prima della pubblicazione nello store)
- Campione di possibili utenti (non tecnici) che fungano da tester per la versione beta dell'app (prima della pubblicazione nello store)

2.4 Responsabilità di Progetto

Avendo adottato un'organizzazione di tipo democratico decentralizzato, la responsabilità sarà suddivisa in maniera equa tra tutti i membri del team.

Ogni membro avrà comunque dei ruoli e dei lavori da rispettare, ma saranno comunque controllati da tutti gli altri membri.

Nello specifico Andrea Zennaro e Antonio Guastavino si sono occupati principalmente del codice, mentre Matteo Toffolo e Vladimyr Tarquini si sono occupati della documentazione e della ricerca dei dati.

3.1 Obiettivi e Priorità

Obiettivi

- Sviluppare un'app funzionante per Android;
- Rispettare i tempi della consegna del prodotto entro la data prefissata;

Priorità

- Ottenere dati giornalieri aggiornati;
- Imparare tecniche di programmazione per l'utilizzo su piattaforme mobile;
- Pubblicare un software di qualità accettabile;

3.2 Assunzioni, Dipendenze, Vincoli

Assunzioni:

- Gli utenti sono studenti di Ca' Foscari;
- Tutti gli utenti usano dispositivi con Android;
- Il dispositivo mobile abbia accesso ad internet;
- Gli utenti finali scaricheranno e installeranno l'app da Play Store;
- L'utilizzo dell'app sarà progettata per utenti non portatori di handicap motori e visivi

Dipendenze:

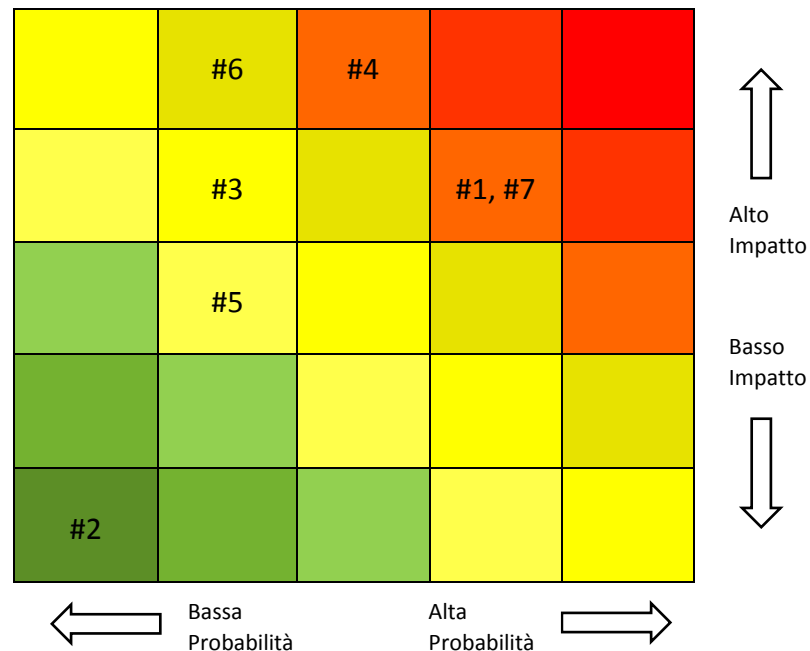
- Utilizzo di smartphone/tablet con sistema Android;
- La composizione del team non subirà cambiamenti (se non minimi e temporanei).

Vincoli:

- Preparazione degli esami di altri corsi;
- L'app utilizzerà dati prodotti dal Centro Maree e VPC;
Il corretto funzionamento dell'app è pertanto vincolato dalla disponibilità e correttezza di tali dati;
- Conoscenza di altri linguaggi di programmazione;
- Pubblicazione app 28 febbraio.

3.3 Gestione dei rischi

| | Identificazione del rischio: | Probabilità | Impatto | Azione |
|----|--|-------------|-------------|---|
| #1 | Cattiva stima della taglia del progetto | 75% | Alta | Aggiornamento settimanale della suddivisione del lavoro |
| #2 | Insoddisfazione dei membri del team a prodotto finito | 10% | Molto Basso | Chiarimenti tramite riunioni |
| #3 | Insoddisfazione del cliente | 20% | Alta | Migliorare il progetto tramite aggiornamenti e rework |
| #4 | Scorretto funzionamento delle risorse (open data) | 60% | Molto Alto | Ricerca di nuove risorse |
| #5 | Eccessiva complessità dell'ambiente di sviluppo | 40% | Media | Maggior approfondimento tramite manuali e test vari |
| #6 | Perdita dei dati dovuti a malfunzionamenti hardware dei pc su cui progettiamo. | 25% | Molto Alta | Backup dei dati |
| #7 | Cattiva stima della complessità degli algoritmi di instradamento da utilizzare | 70% | Alta | Chiedere supporto su tale implementazione |



3.4 Meccanismi di monitoraggio e di controllo

I controlli sulla qualità del lavoro svolto dal team saranno di due tipi:

- *Controlli pianificati*: almeno un controllo generale ogni settimana; tale numero può aumentare a seconda dei problemi rilevati durante la settimana precedente;
- *Controlli a discrezione del singolo membro*: ogni componente del team potrà a propria discrezione effettuare controlli di vario genere sul progetto.

Ogni membro del team, al termine di ogni modifica o implementazione di nuove features, è tenuto a commentare e/o segnalare eventuali TODO il proprio codice, in modo tale da rendere più comprensibile anche agli altri membri ogni modifica.

Il nostro canale di comunicazione sarà Telegram.

Per evitare la perdita di dati verranno utilizzati strumenti di archiviazione online, i quali saranno Google Drive e Github.

3.5 Pianificazione dello staff

È gradita ma non strettamente necessaria una buona familiarità con:

- Pacchetto Office;
- Conoscenza buona dell'IDE di sviluppo Android studio .

Le skills mancati verranno colmate attraverso 3 metodi:

- Lettura di testi tecnici specifici per la progettazione e l'implementazione di app per la piattaforma Android;
- Ricerca online di corsi specifici sia per la programmazione di app per Android;
- Consultazione tutor laboratorio ingegneria del software;

4.1 Metodi, Strumenti e Tecniche

Sistemi di calcolo

Il team svilupperà il progetto utilizzando elaboratori su cui è installato il sistema operativo Windows 10.

In tali elaboratori dovrà essere installato l'ambiente di sviluppo necessario per la realizzazione delle app (Android studio).

- *Android Studio - Pro*: ambiente di sviluppo completo e pronto all'uso con software di virtualizzazione del dispositivo, costruzione dell'interfaccia grafica visuale e non. Rende possibile sviluppare app per ogni smartphone/tablet con sistema Android indipendentemente da marca e hardware. È disponibile gratuitamente per Windows, Linux, Mac
- *Android Studio - Contro*: Non è prevista la portabilità per altre piattaforme come iOS, Windows phone, ecc

L'app potrà essere testata sia su dispositivi fisici, quali Sony Z5 Premium che su un Huawei P8 lite, sia su dispositivi emulati direttamente da Android Studio.

Metodi di sviluppo

Utilizzando un modello di processo di tipo evolutivo, lo sviluppo dell'app sarà completato attraversando multiple fasi di prototipazione, ciascuna dei quali necessaria per aggiungere una particolare funzionalità. Ogni ciclo di prototipazione consiste in quattro fasi:

- Analisi della funzionalità da implementare;
- Progettazione della funzionalità;
- Implementazione della funzionalità
- Testing della funzionalità.

Quando saranno implementate tutte le funzionalità minime previste dal documento di specifica dell'app, essa sarà pronta per essere pubblicata.

4.2 Documentazione del Software

I documenti che saranno realizzati dal team possono essere raggruppati in due categorie:

- *Documenti di progetto*: il Piano di progetto, il Documento dei requisiti, il Documento di Progettazione ed eventualmente il codice sorgente dell'app realizzata;
- *Documentazione delle app*: un semplice manuale rivolto all'utente, che possa brevemente illustrarne l'utilizzo.

4.3 Funzionalità di supporto al progetto

Pianificazione della qualità

I punti di qualità che devono essere rispettati per assicurare una buona riuscita del progetto sono:

- Mantenere aggiornata la documentazione, perseguendo uno standard qualitativo di buon livello;
- Rispettare quanto definito come metodologia di sviluppo;
- Analizzare i problemi di tipo logico con metodo e rispettare gli standard di buona programmazione (indentazione);
- Rispettare i requisiti funzionali e non funzionali che verranno definiti in corso d'opera;
- Attuare una politica di testing meticoloso in modo da individuare per tempo i problemi e gli errori da correggere.

Pianificazione della gestione delle configurazioni

- Tenere traccia di tutte le versioni dei prototipi, in modo da poter risalire ai cambiamenti effettuati a fronte di determinati problemi e poter eseguire rollback di versione in caso di malfunzionamento successivo.
- Documentare dettagliatamente ogni cambiamento dei prototipi nella documentazione in modo chiaro e leggibile per poter supportare il lavoro di gruppo;

5.1 WBS (Work breakdown structure)

Diagramma di Pert

- A** Proposte idee per l'app;
- B** Definizione struttura del progetto;
- C** Definizione struttura del documento;
- D** Ricerca materiale statistico (open data);
- E** Apprendimento programmazione Android;
- F** Definizione schema UML dell'app;
- G** Definizione pattern;
- H** Definizione della grafica dell'app;
- I** Stesura codice dell'app;
- J** Immagini per il Play Store
- K** Stesura documento di analisi e specifica dei requisiti
- L** Progettare i test di usabilità;
- M** Svolgere test di usabilità;
- N** Raccolta statistiche usabilità;
- O** Consegnare piano di testing;
- P** Stesura documento finale di progettazione;
- Q** Pubblicazione versione beta sul Play Store;
- R** Pubblicazione versione stabile sul Play Store;
- S** Consegna del progetto al professore.
- T** Decisioni Ruoli interni

WBS

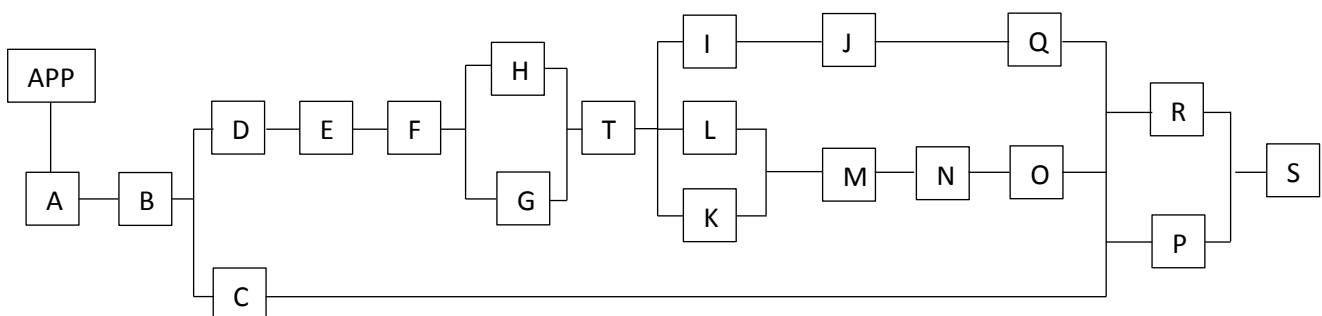
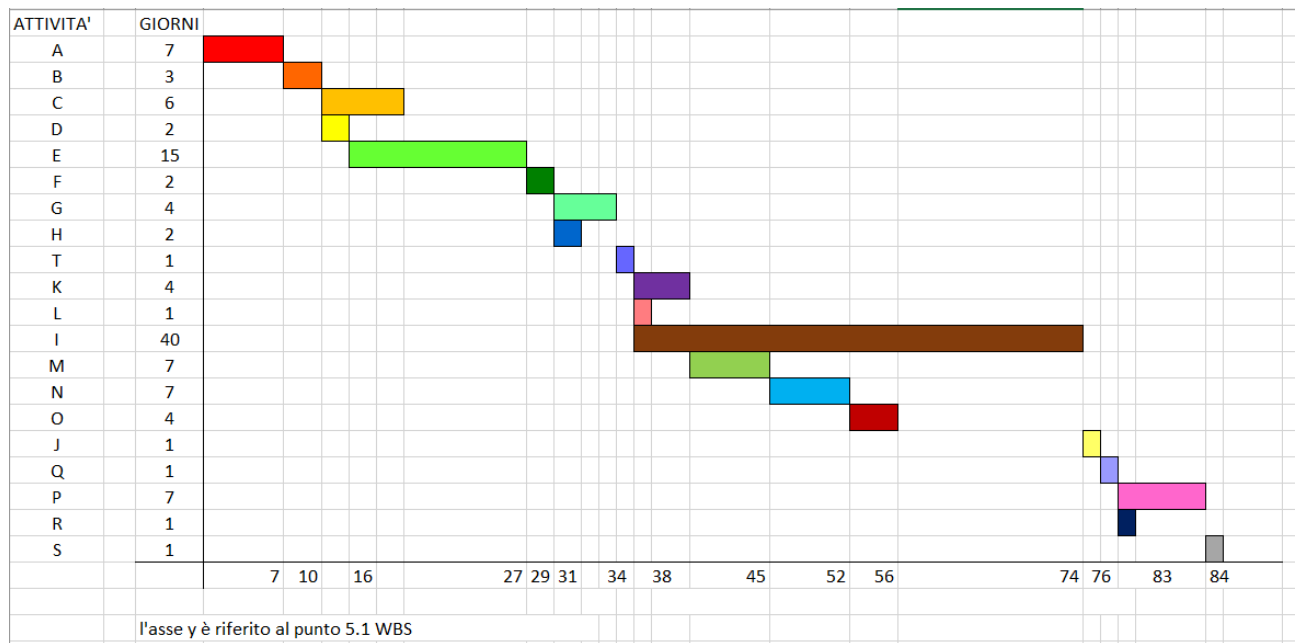
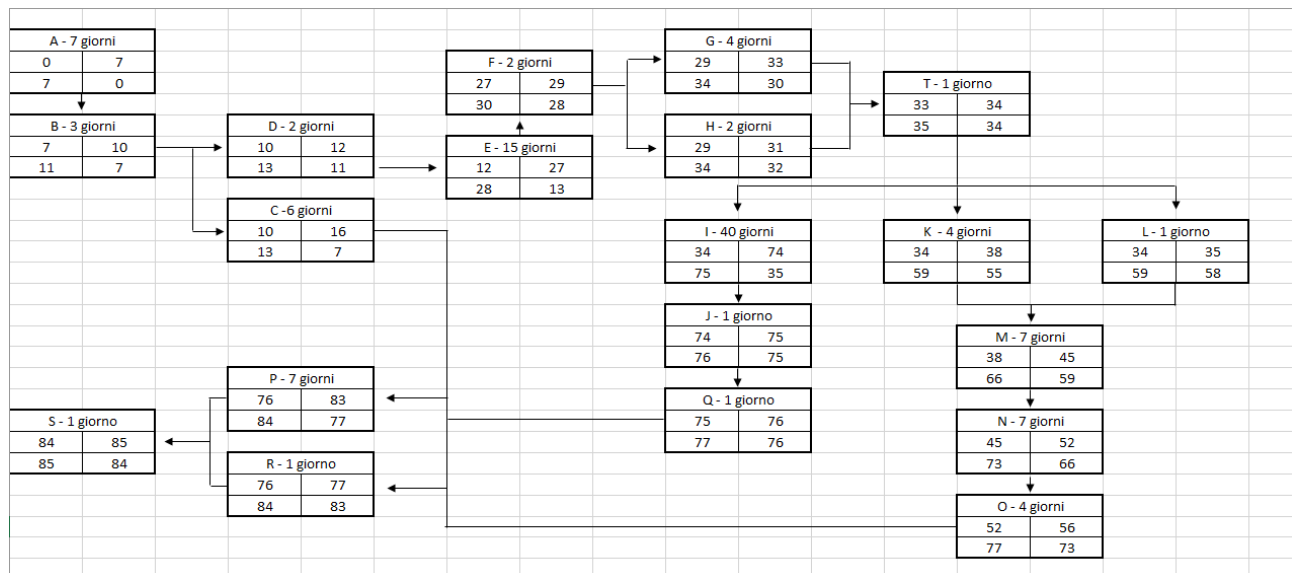


Diagramma di Gantt



Pert



5.2 Dipendenze

| Mansioni | Durata(giorni) | Dipendenze |
|----------|----------------|------------|
| A | 5 | - |
| B | 3 | A |
| C | 6 | B |
| D | 2 | B |
| E | 15 | D |
| F | 2 | E |
| G | 4 | F |
| H | 2 | F |
| T | 1 | G, H |
| K | 4 | T |
| L | 1 | T |
| I | 40 | T |
| M | 7 | K, L |
| N | 7 | M |
| O | 4 | N |
| J | 1 | I |
| Q | 1 | J |
| P | 7 | O, C, Q |
| R | 1 | O, C, Q |
| S | 1 | P, R |

5.3 Risorse Necessarie

Le risorse necessarie per la realizzazione del progetto possono essere raggruppate nelle seguenti categorie:

- *Risorse umane*: gli sviluppatori (membri del team). Non è prevista l'introduzione di ulteriori componenti all'interno del team, ma una volta raggiunta la fase di test dell'app potrebbe essere necessario reclutare dei tester;
- *Risorse hardware*: ogni membro del team avrà bisogno di un computer connesso a Internet;
- *Risorse software*: su ogni computer utilizzato per lo sviluppo dell'app dovrà essere installato Android Studio.

5.4 Allocazione del Budget e delle Risorse

Lo sviluppo dell'applicazione non richiede nessun tipo di risorsa economica, in quanto i software impiegati (Android Studio, Telegram) sono tutti gratuiti.

I dati su cui si baserà l'applicazione sono anch'essi ad accesso libero.

Gli unici costi saranno quelli relativi all'iscrizione al PlayStore e all'utilizzo delle mappe google; inoltre la realizzazione dell'applicazione richiederà sicuramente un costo temporale indicato in Giorni/Uomo (vedi tabella 5.2).

5.5 Pianificazione

La pianificazione del progetto seguirà la stesura dei documenti da consegnare nelle scadenze stabilite con il professor Cortesi all'interno del corso di Ingegneria del Software:

- Piano di progetto versione 21/10/2016
- Documento di analisi e specifica dei requisiti 11/11/2016
- Documento di Progettazione 10/12/2016
- Piano di testing 20/12/2016
- Manuale utente 22/02/2017

Durante il periodo di sviluppo dell'app i documenti verranno corretti ed aggiornati