

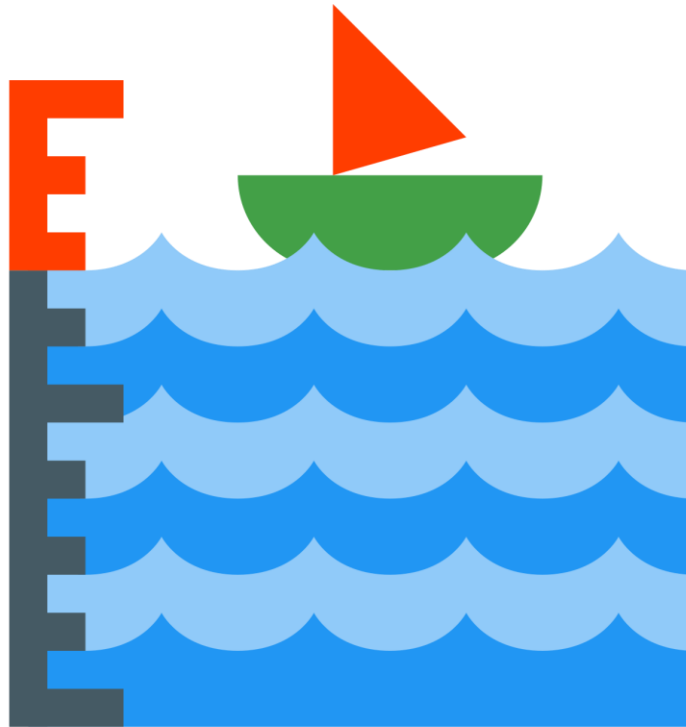
Marco Luisotto **859193**

23/10/2017

Lorenzo Pocchi **860840**

Radu Andrei Novac **857630**

Roberto Milan **857180**



“Acqua Alta”

Piano di progetto

Versione: 1.0

INDICE

1. Introduzione	<i>pag. 3</i>
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	<i>pag. 5</i>
2.1 Modello del Processo	
2.2 Struttura Organizzativa	
2.3 Interfacce Organizzative	
2.4 Responsabilità di Progetto	
3. Descrizione dei Processi Gestionali	<i>pag. 6</i>
3.1 Obiettivi e Priorità	
3.2 Assunzioni, Dipendenze, Vincoli	
3.3 Gestione dei rischi	
3.4 Meccanismi di monitoraggio e di controllo	
4. Descrizione dei Processi Tecnici	<i>pag. 9</i>
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	<i>pag. 10</i>
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. Introduzione

1.1 - Overview del Progetto

Il progetto consisterà nella creazione di un'applicazione Android pensata per i cittadini e i pendolari della laguna di Venezia.

L'applicazione sfrutterà gli Open Data forniti dalla Pubblica Amministrazione con lo scopo di monitorare e prevedere il livello della marea per notificare all'utente lo stato della laguna.

Il target a cui fa riferimento l'applicazione è il semplice cittadino o studente che vuole informarsi riguardo le condizioni dei livelli dell'acqua rispetto alla superficie per evitare eventuali allagamenti.

1.2 - Deliverables del Progetto

Il progetto porterà all'analisi e alla revisione di diversi oggetti.

Verrà consegnata una prima versione del Piano di progetto (questo stesso documento) entro il 23/10/2017.

Prima versione del Documento di analisi e specifica. Data consegna: 09/11/2017.

Prima versione del Documento di progettazione. Data consegna: 15/12/2017.

Prima versione ufficiale del software e messa in linea. Data consegna: 31/01/2018

Ogni singolo oggetto sarà revisionato ed eventualmente modificato, se necessario.

1.3 - Evoluzione del Progetto

Il progetto non è definitivo e alcuni dettagli potrebbero cambiare nel tempo, ci riserviamo eventuali modifiche in quanto è stato sviluppato in poco tempo prima della stesura di questo documento.

Il progetto verrà suddiviso in 3 fasi:

- La prima fase sarà di ricerca delle informazioni e del confronto delle idee tra i membri del gruppo.
- La seconda fase sarà la realizzazione delle interfacce e l'implementazione dei sensori altimetrici nell'applicazione.
- La terza ed ultima fase sarà la ridefinizione dell'applicazione e correzione di eventuali bug attraverso il testing con l'implementazione finale di alcune features extra se il tempo lo permette.

1.4 - Materiale di riferimento

- Wikipedia (<http://it.wikipedia.org>) per le definizioni formali
- Guida sul sito ufficiale di Android Studio (<https://developer.android.com>)
- Forum come Stack Overflow per informazioni tecniche
- Dispense universitarie

1.5 - Definizioni e Abbreviazioni

Di seguito verranno illustrate le più comuni definizioni e abbreviazioni che si incontreranno, o che si hanno già incontrato, in questo documento:

- **Software:** l'insieme delle procedure e delle istruzioni in un sistema di elaborazione dati, si identifica con un insieme di programmi (in contrapposizione a hardware).
- **User Interface:** in italiano interfaccia utente e ciò che si frappone tra una macchina e un utente consentendo l'iterazione tra i due.
- **App:** con il neologismo *App* si intende una variante delle applicazioni informatiche dedicate ai dispositivi mobili, quali smartphone e tablet.
- **Open Data:** sono dati liberamente accessibili da parte di tutti le cui eventuali restrizioni sono l'obbligo di citare la fonte o di mantenere la banca dati sempre aperta, di solito sono rese disponibili dalle pubbliche amministrazioni.
- **Livello altimetrico:** è la differenza espressa in millimetri dell'attuale livello del mare in confronto ad una misura standard, che può essere il normale livello del mare o, come nel caso di Venezia, il livello mareografico di Punta della Salute (ZMPS).

2. Organizzazione del progetto

2.1 - Modello di Progetto

Il modello scelto sarà di tipo evolutivo.

Abbiamo scelto questo tipo di modello in modo da aggiornare nel tempo la nostra applicazione mentre proseguiremo nella scrittura del software.

In questo modo, dopo aver definito le specifiche principali saremo liberi di aggiungere e modificare eventuali funzioni.

2.2 - Struttura Organizzativa

Nel nostro gruppo non esiste una gerarchia di ruoli, ogni membro quindi avrà lo stesso potere decisionale, non esisterà un leader e le scelte critiche verranno prese democraticamente, in modo da non rallentare le decisioni in fase di scrittura codice.

2.3 - Interfacce organizzative

L'applicazione sfrutterà gli Open Data messi a disposizione dal Comune di Venezia, in particolare il livello altimetrico raccolto da diversi sensori sparsi nel territorio Veneziano.

2.4 - Responsabilità di Progetto

In data 23/10/2017 non abbiamo ancora deciso i ruoli della realizzazione del progetto.

Questi ruoli verranno assegnati al momento della stesura del codice avendo acquisito una visione generale sullo sviluppo dell'applicazione.

3. Processi gestionali

3.1 - Obiettivi e priorità

- Ultimare lo sviluppo dell'applicazione entro 31 gennaio 2018
- Costruire un'applicazione user-friendly
- Fornire all'utente una visione immediata e facilmente comprensibile dei dati

3.2 - Assunzioni, dipendenze e vincoli

Assunzioni

Assumerò che l'utente sia consapevole dei fenomeni di marea nella laguna di Venezia e che si sappia destreggiare tra i dati altimetrici.

Dipendenze

Lo sviluppo dell'applicazione dipenderà principalmente dalle possibilità che il linguaggio Java potrà offrire nell'ambiente di sviluppo di Android Studio, dai dati disponibili e dall'esperienza del team di sviluppo.

Vincoli

Consegna dei documenti e dell'applicazione entro i limiti prefissati.

3.3 - Gestione dei rischi

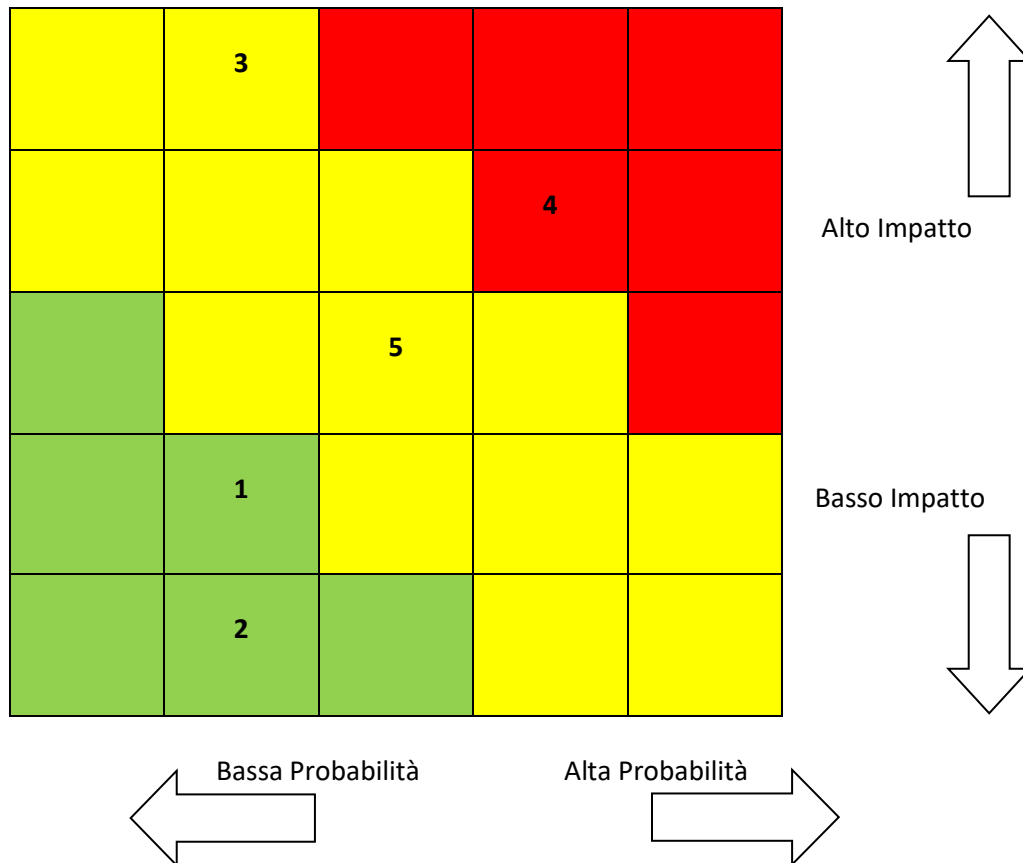
Analisi e quantificazione dei rischi

ID Rischio	Nome Rischio	Categoria	Descrizione del rischio	Probabilità	Impatto	Descrizione impatto
1	Malattia	Personale	Un componente del gruppo si ammala	Bassa	Basso	In caso di malattia di uno dei componenti si potrebbe perdere tempo
2	Guasto PC	Materiali	Il PC su cui si lavora si guasta	Bassa	Basso	La rottura di un componente del PC può causare perdite di dati non salvati
3	Perdita dati	Sviluppo	Alcuni file vengono persi o corrotti	Bassa	Medio/Alto	La perdita di dati porterà alla perdita di tempo per riscriverli
4	Problemi codice	Sviluppo	Rallentamenti nella stesura del codice dell'applicazione	Alta	Alta	Data la poca esperienza nella programmazione con Android Studio ci saranno rallentamenti importanti
5	Bug	Sviluppo	Rallentamenti dovuti alla scoperta e correzione di bug	Media	Medio	Ci potrebbero essere bug grafici di minore impatto e bug di natura numerica che possono avere un impatto pericoloso

Strategie e possibili soluzioni

ID Rischio	Nome Rischio	Categoria	Descrizione del rischio	Strategia	Azione da pianificare
1	Malattia	Personale	Un componente del gruppo si ammala	Mitigarlo	Il componente malato verrà sostituito dei suoi compiti temporaneamente da un altro componente
2	Guasto PC	Materiali	Il PC su cui si lavora si guasta	Mitigarlo	In caso di guasto si può proseguire sui PC di laboratorio o di altri componenti
3	Perdita dati	Sviluppo	Alcuni file vengono persi o corrotti	Evitarlo	Eseguire spesso backup cercando di evitare il problema, se i file vengono persi si può ripartire dagli ultimi dati caricati su GitHub
4	Problemi codice	Sviluppo	Rallentamenti nella stesura del codice dell'applicazione	Mitigarlo	Imparare tecniche di programmazione più sofisticate
5	Bug	Sviluppo	Rallentamenti dovuti alla scoperta e correzione di bug	Mitigarlo	Cercare di scrivere codice più pulito possibile e a posteriori risolvere i bug in fretta

Tabella della classificazione di rischi



3.4 - Meccanismi di monitoraggio e di controllo

I controlli sull'applicazione verranno effettuati ogni due settimane per rilevare eventuali problemi.

Comunicheremo attraverso l'applicazione VoIP "Discord" e useremo lo spazio di archiviazione in cloud di "Dropbox" e GitHub per il passaggio di codice.

3.5 - Pianificazione dello staff

Le conoscenze richieste per questo progetto sono:

- Buone conoscenze del linguaggio Java
- Conoscenza base di Android Studio
- Abilità di lavorare in gruppo

Le conoscenze da acquisire avverranno per mezzi di:

- Approfondimento online da forum o video tutorial
- Consultazione del tutor del corso

4. Processi tecnici

4.1 - Metodi, strumenti e tecniche

Nel corso del progetto si utilizzeranno calcolatori con sistemi operativi Windows o Distribuzioni Linux, in più verrà utilizzato l'IDE Android Studio come principale strumento di stesura del codice e Microsoft Word per la scrittura della documentazione. Infine per testare l'applicazione verranno utilizzati uno o più smartphone Android di proprietà dei componenti del gruppo.

4.2 - Documentazione del software

I documenti redatti saranno:

- Piano di progetto
- Documento di specifica dei requisiti
- Documento di progettazione

4.3 - Funzionalità di supporto al progetto

Pianificazione della qualità

- Rispettare gli obiettivi e i requisiti del progetto
- Definire una buona metodologia di testing per poter individuare eventuali errori nel codice
- Mantenere una documentazione interna condivisa da tutto il gruppo per garantire coerenza tra le fasi di sviluppo

Pianificazione della gestione delle configurazioni

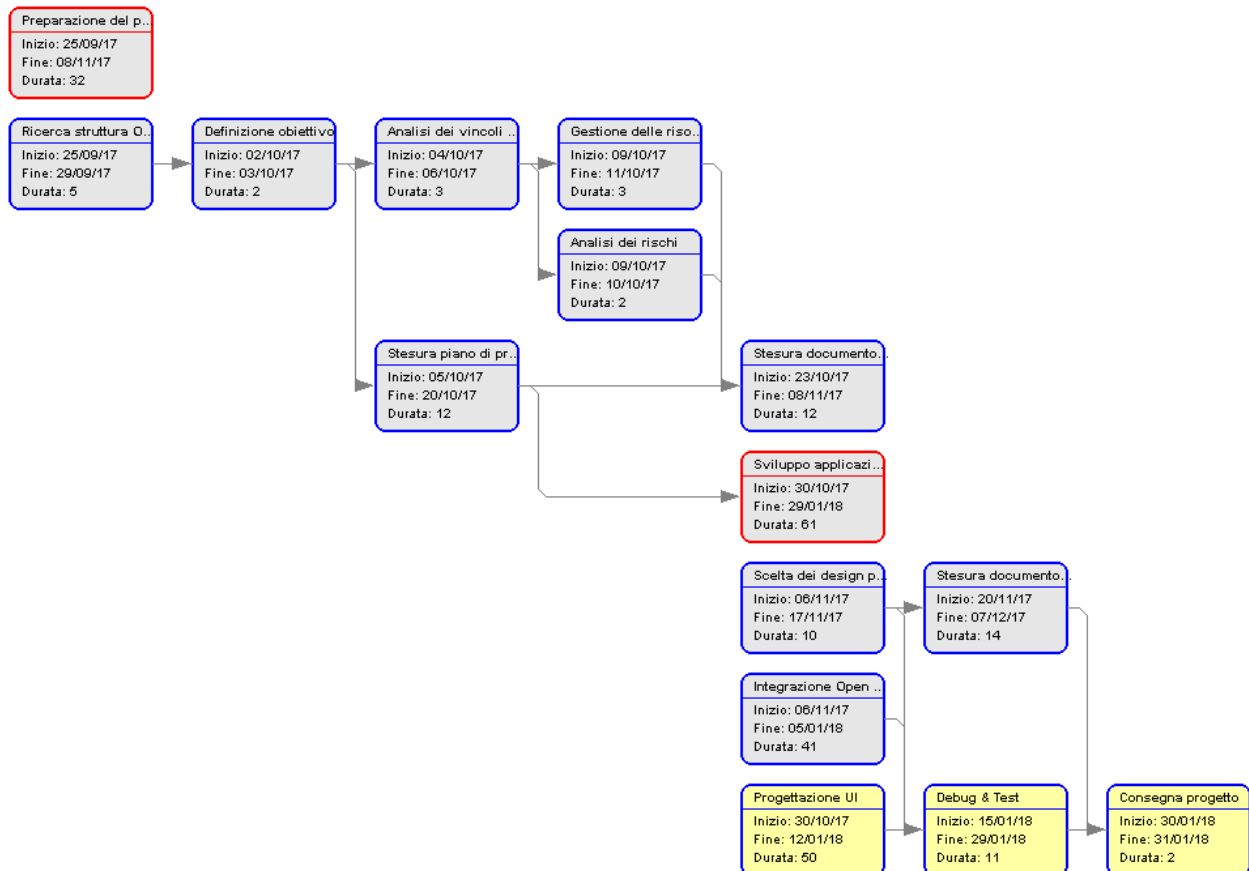
- Tenere traccia delle modifiche al codice attraverso servizi di controllo di versione quali GitHub
- Strutturare il codice attraverso pattern di design che rendano la struttura dell'applicazione scalabile

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

5.1 - WBS (Work Breakdown Structure)

1. Preparazione del progetto
 - 1.1. Ricerca struttura open data
 - 1.2. Definizione obiettivo
 - 1.3. Analisi dei vincoli e delle scadenze
 - 1.4. Gestione delle risorse
 - 1.5. Analisi dei rischi
 - 1.6. Piano di progetto
 - 1.7. Documento di analisi e specifica
2. Sviluppo applicazione
 - 2.1. Progettazione UI (*User Interface*)
 - 2.2. Integrazione Open Data
 - 2.3. Scelta dei design pattern
 - 2.4. Debug & Test
 - 2.5. Documento di progettazione
3. Consegna del progetto

5.2 – Dipendenze



5.3 - Risorse necessarie

Risorse Hardware:

- 4 computer portatili con sistema operativo Windows 10, distribuzioni Linux o MacOS
- Smartphone Android per test dell'applicazione

Risorse Software:

- Android Studio
- Telegram per comunicazione testuale
- Discord per comunicazione VoIP
- Dropbox per passaggio di materiale e Backup codice
- Microsoft Word per stesura documento

5.4 - Allocazione del budget e delle risorse

Le risorse saranno calcolate in ore/attività:

- Piano di progetto: 8 h
- Documento di analisi e specifica: 8 h
- Documento di progettazione: 8 h
- Progettazione UI: 25 h
- Integrazione Open Data: 20 h
- Debug & Test: 5 h

5.5 – Pianificazione

