

**Open Bikes**



**Open Bikes**

*Piano di progetto – Corso di Ingegneria del software*

*Andrea Ottolino - 858165*

*Nicolas Cecco - 859116*

*Giacomo Arrigo - 860022*

*Marco Carfizzi – 860149*

## **1. Introduzione pag. 03**

- 1.1. Overview del Progetto
- 1.2. Deliverables del Progetto
- 1.3. Evoluzione del Progetto
- 1.4. Definizioni e Abbreviazioni

## **2. Organizzazione del Progetto pag.04**

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

## **3. Descrizione dei Processi Gestionali pag.05**

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

## **4. Descrizione dei Processi Tecnici pag.07**

- 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 pag.08**

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

# 1. Introduzione

## 1.1 Overview del progetto

Il progetto consiste nella creazione di una app android pensata per facilitare l'accesso e l'individuazione dei luoghi con servizi per biciclette. L'idea di fondo è facilitare l'utilizzo delle biciclette messe a disposizione dallo stato per qualsiasi cittadino che ne abbia necessità.

Essa si presenterà come una mappa in cui verranno segnalati tramite "marker" tutti i punti dove sarà possibile usufruire del servizio. Tramite la pressione dei suddetti "marker" si potrà accedere ad informazioni avanzate inerenti al punto (esempio se è un punto di noleggio o un parcheggio bici).

## 1.2 Deliverables del progetto

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

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

Piano di testing. Data di consegna entro: 23/11/2017

Documento di progettazione. Data di consegna entro: 15/12/2017

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

Ognuno di questi potrà essere soggetto a modifica nelle date precedenti a quelle specificate per la consegna.

## 1.3 Evoluzione del progetto

Uno dei punti fondamentali della nostra applicazione sarà essere accessibile e user friendly. Per questo motivo potrebbe essere necessario offrire un servizio di ticket per gestire il feedback degli utenti in modo da migliorare tanto l'interfaccia quanto la struttura stessa dell'applicazione. Una volta sviluppata il corpo base, potrebbe essere interessante sviluppare un sistema di navigazione verso un punto di interesse.

In futuro potrebbe presentare punti di utilizzo anche per altri mezzi di trasporto pubblico, i.e. fermate degli autobus o autostazioni. Nella prospettiva della creazione della Carta di identità digitale potrebbe presentare delle integrazioni con le quali sarà più semplice gestire e visualizzare i propri titoli elettronici (abbonamenti di autobus, treni, metro etc. oppure biglietti).

## 1.4 Definizioni e abbreviazioni

Di seguito elenchiamo i termini più comuni che verranno utilizzati in questo e nei futuri documenti:

- Applicazione Software: Programma installato su un computer con lo scopo di rendere possibile funzionalità, servizi o strumenti utili su richiesta dell'utente tramite una interfaccia.
- App: abbreviazione per Applicazione Mobile, ovvero una applicazione software per dispositivi mobili quali smartphone o tablet.
- Interfaccia utente: ciò che si frappa fra macchina e utente permettendone l'interazione reciproca.
- Bike sharing: è uno strumento di mobilità sostenibile a disposizione delle amministrazioni pubbliche che intendono aumentare l'utilizzo dei mezzi di trasporto pubblici e integrandoli dall'utilizzo delle biciclette condivise.
- Marker: un'icona che identifica una località su una mappa.
- User friendly: di facile utilizzo per l'utenza meno esperta.

## **2. Organizzazione del progetto**

### **2.1 Modello del processo**

Il modello che abbiamo deciso di utilizzare sarà di tipo evolutivo. Durante la realizzazione del processo aggiorneremo man mano sia il codice che la documentazione, in base alle necessità e ai problemi che si presenteranno. Come abbiamo detto precedentemente, per la nostra app vorremmo tenere in alta considerazione il feedback dell'utenza in modo da avere un risultato finale appetibile e user friendly. L'obiettivo finale sarà presentare un software che rispecchi la documentazione presentata e le nostre aspettative.

### **2.2 Struttura organizzativa**

Avendo già lavorato insieme in progetti precedenti, gestiremo il progetto nella maniera più democratica possibile. Le decisioni verranno prese di comune accordo e non ci sarà un vero e proprio leader. In questo modo contiamo di correggere errori e risolvere problemi velocemente ed in maniera efficiente.

### **2.3 Interfacce organizzative**

L'applicazione si avvale degli open data forniti da [openstreetmap.org](http://openstreetmap.org) , un progetto collaborativo finalizzato a creare mappe a contenuto libero del mondo.

### **2.4 Responsabilità di progetto**

Di seguito una lista delle principali funzioni per la realizzazione del software con il singolo responsabile. Ogni membro ovviamente darà il suo contributo per ogni aspetto della creazione dell'app, indipendentemente dall'incarico che gli è stato assegnato direttamente

Stesura documentazione: Andrea Ottolino

Interfaccia utente: Marco Carfizzi

Creazione app: Nicolas Cecco

Test app e correzione errori: Giacomo Arrigo

### 3. Processi gestionali

#### 3.1 Obiettivi e Priorità

- Completare l'applicazione entro la data di scadenza
- Minimizzare il numero di bug e presentare un'app funzionale
- Presentare un'interfaccia user friendly

#### 3.2 Assunzioni, Dipendenze, Vincoli

Il vincolo più grande è la distanza fra i membri del gruppo (due abitano in provincia di Treviso e due in provincia di Venezia).

#### 3.3 Gestione dei rischi

- Non riuscire a rispettare le scadenze
- Problemi di prestazione dei nostri hardware
- Creazione di un'interfaccia poco user friendly
- Implementazione di funzioni troppo complesse o pesanti
- Creazione di funzioni deprecabili
- Documentazione incompleta o non chiara
- Incomprensioni tra i membri del team
- Perdita di un membro
- Ritardi dovuti a problemi di salute o famigliari

	Rischi	Ambito	Probabilità	Rilevanza
1	Non riuscire a rispettare le scadenze	Prodotto	10%	5
2	Problemi di prestazione dei nostri hardware	Progetto	30%	2
3	Creazione di un'interfaccia poco user friendly	Prodotto	15%	3
4	Implementazione di funzioni troppo complesse o pesanti	Prodotto e Progetto	60%	2
5	Creazione di funzioni deprecabili	Prodotto e Progetto	30%	3
6	Documentazione incompleta o non chiara	Progetto	20%	4
7	Incomprensioni tra i membri del team	Progetto	10%	4
8	Perdita di un membro	Progetto	10%	5
9	Ritardi dovuti a problemi di salute o famigliari	Progetto	15%	3

Rilevanza	PROBABILITA'				
	0%-20%	21% - 40%	41% - 60%	61% - 80%	81% - 100%
5	1-8				
4	6-7				
3	3-8-9	5			
2		2	4		
1					

### 3.4 Meccanismi di monitoraggio e controllo

Contiamo di utilizzare la comunicazione interna al gruppo per prevedere la possibilità che un problema possa insorgere per un singolo membro. Per i problemi legati al progetto, ogni membro controllerà il lavoro dei propri compagni in modo da rilevare problemi ed errori facilmente risolvibili. Per quanto riguarda i possibili problemi di comunicazione, utilizzeremo molteplici sistemi per scambiarsi informazioni (Telegram e Discord).

### 3.5 Pianificazione dello staff

Tutti i membri del nostro gruppo hanno una solida conoscenza del linguaggio Java utilizzato da Android Studio per la creazione dell'applicazione. Per il reparto grafico un membro del gruppo ha esperienza come graphic designer mentre un altro membro ha già lavorato come sviluppatore per altre applicazioni, web e mobili.

## **4. Processi Tecnici**

### **4.1 Metodi, Strumenti e Tecniche**

- Strumenti Hardware:
  - 3 Computer portatili. Due con sistema operativo Linux Debian e uno con Windows 10. 3 Fissi, uno con Windows 7, uno con Windows 8.1 ed uno con Windows 10. Tutti questi dispositivi verranno utilizzati per la stesura di codice e documenti.
  - 3 Smartphone Android per il testing dell'applicazione
- Strumenti Software:
  - Sistemi operativi Linux Debian, Windows 7, 8.1 e 10
  - Android Studio
  - Telegram e Discord per la comunicazione tra i membri del team
  - Microsoft Office Word per la stesura della documentazione

### **4.2 Documentazione del software**

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

Documento di progettazione. Data di consegna entro: 15/12/2017

### **4.3 Funzionalità di supporto al progetto**

In quanto alla qualità, come abbiamo suggerito precedentemente l'idea di fondo della nostra applicazione è basarci sul feedback dell'utenza per fare in modo che sia il più user friendly possibile.

Per lavorare in maniera ottimale e tenere aggiornati tutti i membri del gruppo sulle varie modifiche che verranno apportate all'applicazione, oltre che a tenere uno storico delle modifiche, pensavamo di sfruttare l'archiviazione cloud fornita da Telegram in cui salveremo le singole versioni del codice e terremo conto delle modifiche apportate dai singoli.

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

### 5.1 WBS (Work breakdown structure)

1. Specifiche
  - Definizione Obiettivi
  - Definizione Vincoli
  - Definizione risorse
  - Definizione risorse software/hardware
  - Definizione risorse umane
  - Definizione modello del progetto
  - Definizione responsabilità
  - Definizione e gestione rischi
  - Stesura piano di progetto
2. Progettazione Concettuale
  - Modellazione del sistema
  - Analisi dei viewpoint/utenti
  - Definizione dei requisiti
  - Definizione dei requisiti funzionali
  - Definizione dei requisiti non funzionali
  - Stesura documento di analisi e specifica
3. Progettazione tecnica
  - Stesura documento di progettazione
4. Client
  - Creazione user interface
  - Integrazione UI-Dataset
  - Porting
5. Testing
  - Test app
  - Debugging
6. Rilascio Software



## 5.2 Documentazione del software

Analisi dei requisiti → //

Progettazione → //

Documento di analisi dei requisiti → Analisi dei requisiti + Progettazione

Documento di progettazione → documento di analisi dei requisiti

User-Interface → Documento di progettazione

Compilazione app → test + debugging

## 5.3 Risorse necessarie

Risorse hardware:

- Computer fissi(3) e portatili(3) per compilare il codice ed elaborare i documenti
- Smartphone Android per testare l'applicazione

Risorse software:

- Sistemi operativi Windows 10 e Linux Debian
- Android Studio
- Microsoft Office Word per la creazione dei documenti

## 5.4 Allocazione del Budget e delle Risorse

Di seguito sono definite le risorse in ore/attività

- Completamento Documento di analisi e specifica: 5h
- Completamento Piano di Testing: 6h
- Completamento Documento di Progettazione: 6h
- Completamento codice: 20h
- Completamento UI: 15h
- Completamento app test: 2h

Queste ore sono da ritenersi indicative in quanto potrebbero subire variazioni in corso d'opera.

## 5.5 Pianificazione

Piano di progetto (questo stesso testo): 23/10/2017.

Documento di analisi e specifica: 09/11/2017

Piano di testing: 23/11/2017

Documento di progettazione: 15/12/2017

Completamento e compilazione codice: 10/01/2018

Completamento UI: 20/01/2018

Completamento test app e debugging: 21/01/2018

Consegna codice: 31/01/2018