PROJECT PLAN

# 

# INTRODUCTION

Il progetto è basato sulla creazione di un gioco per desktop/web app. Il gruppo di progetto è formato da Daidone, Drissi, Ianitchii, Marchesi.

L'obiettivo è stato quello di mettere in pratica le conoscenze acquisite durante il corso di Ingegneria del software e anche di altre già in nostro possesso.

# PROCESS MODEL

Per lo sviluppo di questo progetto abbiamo deciso di utilizzare l’approccio agile poiché riflette al meglio la modalità di lavoro che vorremmo adottare.

L’idea è quella di partire da un prototipo base che rappresenta un prodotto con l’implementazione delle funzionalità base e successivamente tramite miglioramenti incrementali portare modifiche per l’evoluzione del prototipo in un prodotto finito. A fine di ogni intervento vogliamo ottenere un prodotto sempre funzionante e quindi faremmo l'adozione di un'integrazione continua.

# ORGANIZATION OF THE PROJECT

Abbiamo definito un product team, nel quale ogni componente del gruppo è in grado di creare, modellare e ridefinire tutte le parti dello sviluppo del progetto. Questo ci permette di poter lavorare in parallelo essendo tutti in grado di intervenire su qualsiasi fase del progetto.

I punti di forza dei componenti del gruppo sono:

* Daidone: backend
* Drissi: testing
* Ianitchii: Software architect
* Marchesi: frontend

# STANDARDS,GUIDELINES, PROCEDURES

Come programmi software/tools abbiamo utilizzato:

* Github
* Vscode
* Vim
* Google docs
* IEE…
* Camel case per nomi classi
* Undescore per nomi file
* Flutter
* dart
* ....

# MANAGEMENT ACTIVITIES

Incontri eseguiti con frequenza periodica di periodo X. Abbiamo mantenuto un log di tutti gli incontri eseguiti, vedi file: IS PW - meetings

Abbiamo usato github per tenere traccia dei progressi eseguiti nel tempo.

# RISKS

Tra i possibili rischi di questo progetto quello più imponente è quello di non riuscire a rispettare la deadline fissata per la consegna del prodotto. Tra gli altri rischi possono essere individuati:

* Non riuscire a completare i compiti assegnati a ciascun individuo
* Non riuscire a eseguire le riunioni nei tempi definiti
* Non riuscire ad avere una visione unica sul prodotto
* Prodotto non funzionante a fine giornata

# STAFFING

Data la quantità limitata di personale abbiamo cercato di sfruttare al meglio le nostre abilità e abbiamo cercato di modellare le nostre conoscenze in base alla situazione ed alla richiesta

# METHODS AND TECHNIQUES

Durante il progetto abbiamo deciso di:

* Commit su github per avere la versione più aggiornata sempre disponibile
* Code reviews tramite github
* Creazione di pull request prima di fare modifiche sul main
* Creazione di branch per sezione critiche

# 

# QUALITY ASSURANCE

Per poter rispettare i nostri standard abbiamo utilizzato review dei codice da parte dei componenti del gruppo, testing periodici intensivi.

Per la qualità del codice sono stati utilizzati strumenti di linting e per il rispetto delle best practices.

Durante lo sviluppo abbiamo eseguito tests utilizzando persone esterne per assicurarci che il prodotto fosse comprensibile anche a elementi esterni al progetto. Da queste persone abbiamo poi raccolto dei feedback.

# 

# WORK PACKAGES

# 

# RESOURCES

Librerie esterne, browser/desktop per eseguire l’applicazione,

# 

# BUDGET AND SCHEDULE

In questo progetto non è stato presente un budget a livello monetario ma bensì un budget temporale. La risorsa limitata è stata il tempo e la conoscenza. Il tempo previsto per il completamento del progetto è entro il centinaio di ore per persona.

# CHANGES

Per gestire i cambiamenti abbiamo deciso di tenere una traccia precisa di tutte versione dei file soggetti a modifiche nel tempo, grazie al supporto diretto di github.

Dato il metodo di lavoro adottato i cambiamenti sono all’ordine del giorno. L’obbiettivo è quello di avere sempre un prodotto funzionante alla fine del giorno.

# DELIVERY

Build di un .exe oppure hosting della web app.