**Navigation App for Blind People**

**Software Project Management Plan**

**Team Members**

Carson Gustafson

Justin Crittenden

Michael Goodwyn

Orlando Trujillo-Ortiz

**Table of Contents**

1. Introduction
   1. Project Overview
   2. Project Deliverables
   3. Evolution of Document
   4. References
   5. Definitions, acronyms, and abbreviations
2. Project Organization
   1. Process Model
   2. Organizational Structure
   3. Organizational Boundaries and Interfaces
   4. Project Responsibilities
3. Managerial Process
   1. Management Objectives and Priorities
   2. Assumptions, Dependencies and Constraints
   3. Risk Management
   4. Monitoring and Controlling Mechanisms
4. Technical Process
   1. Methods, tools and techniques
   2. Software Documentation
5. Work elements, schedule, and budget
6. **Introduction**
   1. Project Overview

Our Navigation App for Blind People development will be regularly documented with this software project management plan. The objective of this project is to provide visually impaired people with an app that helps them navigate while inside the buildings of Cal Poly Humboldt.

This SPMP contains the project deliverables, how the project will be organized, which process the project will be using, and how the work will be split up among team members.

* 1. Project Deliverables

The project deliverables for this project are listed below:

| Phase | Deliverable |
| --- | --- |
| 1 | UML Diagram |
| 1 | SPMP |
| 1 | Interim Project I Presentation |
| 2 | Project Presentation and Demo |

* 1. Evolution of Document

This table will be used to track revisions of this document as our team moves forward with project development.

| Revision Number | Date of Revision | Revised By | Summary of update |
| --- | --- | --- | --- |
| 1.0 | 9/12/2023 | All Group Members | Created and outlined document |
| 1.1 |  |  |  |

* 1. References

Discord - Discord.com

Github - Github.com

References will be updated once conversing with the professor.

* 1. Definitions, Acronyms, and Abbreviations

UML - Stands for unified modeling language

RE - Stands for requirements engineering

SPMP - Software Project Management Plan Template

1. **Project Organization**
   1. Process Model

The development of the indoor navigation application will be through this group. During the development phases, the team will get together, and allocate the work needed to be done to complete the phase using an agile working methodology. By doing so will ensure a larger safety net when considering possible errors or flaws that come with the finished product, and provide proper validation for each section of our project.

* 1. Organizational Structure

This team will be organized as a single group, considering it is a team of “requirement engineers”. Through each phase of the project, there will be a linear shift in roles to ensure an even distribution of work and experience on the project.

The beginning of the first stage will start with 4 main developers, and as the phases move on there will be a shift of 1-2 developers to reviewers as needed (if a section gets completed, 1-2 people will review and test the code while the rest move on). The designated leader for the section will be in charge of coordinating group meetings and leading the group productively as needed.

* 1. Organizational Boundaries and Interfaces

The leading role will be rotating through each half phase of the project. Requirement, programming, formatting and reviewing roles will be dispersed as evenly as possible throughout the group.

* 1. Project Responsibilities

| S No. | Deliverables | Developers | Leading Developer |
| --- | --- | --- | --- |
| 1 | Software Project Management Plan | Orlando T, Justin C, Michael G. | Carson G. |
| 2 | Requirements Specification |  |  |
| 3 | Analysis Model |  |  |
| 4 | Architecture Specification |  |  |
| 5 | Component/Object Specification |  |  |
| 6 | Code |  |  |
| 7 | System Testing |  |  |
| 8 | User Documentation |  |  |

1. Managerial Process
   1. Management Objectives and Priorities

The focus and priorities of the members of this project are to meet every requirement necessary in order to achieve the common goal of establishing a software that is capable of navigating blind individuals inside indoor locations safely and accurately. Our project group will be meeting for a minimum of 1 hour a week to work on the project by discussing the goals of our project and the system functionality needed to achieve them, as well as collaborating on the implementation of the code for this project.

* 1. Assumptions, Dependencies and Constraints

Our assumptions, dependencies, and constraints going into this project are the following:

Assumptions:

* The users of this application will be blind to a certain degree, we do not expect users who have no use for this functionality to be users of our application

Dependencies:

* The application will depend on the data entered by the user in order to tailor its functionality to the user as best as possible
* The hardware being used by the user has to be to the standard of the application.

Constraints:

* The due date of the project
* The budget given to our project group to complete this project ($0)
* Conflicting responsibilities that inhibit the amount of time our project group is able to spend focusing on the project.
  1. Risk Management

The risks of the project are:

* Faults in the implementation (data loss)
* Conflicting schedules among group members
* Credibility and reputation among users of our software
  1. Monitoring and Controlling Mechanisms

1. Minimum meeting time of 1 hour per week
2. Shared GitHub repository, allowing version control of the project
3. Discord server to communicate with the team regarding aspects of the project
4. Shared Google Docs document to collaborate on team assignments/deliverables.
5. Technical Process
   1. Methods, tools and techniques

Version control will be achieved by use of an online GitHub repository. GitHub will allow for streamlined collaboration between members.

Discord will be the main method of communication between team members for the entirety of the project. Voice, video, and screen-sharing calls will be utilized to adequately collaborate remotely. An agenda will be set for every meeting to ensure proper goals are met during the session.

Google Docs will be the main word document collaboration tool for the project. It will allow functional synchronous and asynchronous text document collaboration that prevents data loss via cloud storage.

UML modeling will be created using the online LucidChart solution. LucidChart allows for online collaboration on documents.

The application will be written in the Java programming language. Android Studio will be used as the IDE for the project.

* 1. Software Documentation

As of writing, the following will be implemented:

* User manual
* Detailed comments in source code
* UML diagrams
* Source code documentation for contributed files in GitHub

1. Work Elements, Schedule, and Budget

| Phase | Sub-Phase | Developers | Start date | End date | Software resources |
| --- | --- | --- | --- | --- | --- |
| Software Project Management Plan | Software Project Management Plan | Orlando T, Justin C, Michael G, Carson G | 9/10/23 | 9/12/23 | Google Docs |
| Requirements Model | 1. Identifying the functional and non-functional requirements  2. Identifying and drawing use-case diagrams | Orlando T, Justin C, Michael G, Carson G |  |  | Google Docs  LucidChart |
| Analysis Model | 1. Identifying the major objects from the use-cases and mapping the objects from the use-cases to the sequence diagram  2. Identifying the classes and drawing the class diagrams | Orlando T, Justin C, Michael G, Carson G |  |  | Google Docs  LucidChart |
| Architecture Model | 1. Identifying the different feasible styles of architecture and discussing the trade offs of each  2. Component and package identification and drawing the package diagram | Orlando T, Justin C, Michael G, Carson G |  |  | Google Docs  LucidChart |
| Component/Object Specification | 1. Identifying the subsystems, their interfaces and their interactions with the other subsystems  2. Detailed design of each subsystem | Orlando T, Justin C, Michael G, Carson G |  |  | Google Docs  LucidChart  Android Studio |
| Code | 1. Mapping models to code and developing the system | Orlando T, Justin C, Michael G, Carson G |  |  | Android Studio |
| System Testing | Testing | Orlando T, Justin C, Michael G, Carson G |  |  | Android Studio |
| User Documentation | Composing user documentation for the system | Orlando T, Justin C, Michael G, Carson G |  |  | Google Docs |