

CptS 484: Software Requirements

Software Project Management Plan

Project Plan

Team Members

Christopher Young

Cong Trinh

Taryn Burns

Anne Lin

Sean Luchessa

Introduction	2
Project overview	2
Project deliverables	2
Evolution of this document	2
Revision History	3
References	3
Definitions, acronyms, and abbreviations	4
Project organization	5
Process model	5
Organizational structure	5
Team Members	5
Organizational boundaries and interfaces	6
Project responsibilities	6
Managerial process	8
Management objectives and priorities	8
Assumptions, dependencies, and constraints (1)	8
Risk management	8
Monitoring and controlling mechanisms	9
Technical process	11
Methods, Tools, and Techniques	11
Software documentation	11
Project support functions	12
Work elements, schedule, and budget	13

1. Introduction

1.1. Project overview

The project being presented in this project plan is an application that will be used to aid the blind with navigating buildings. This project plan will assess the needs of a blind person to lead the implementation of a helpful tool. Hardware that will be utilized includes voice commands, GPS for directions to navigate hallways, and make use of various sensors to detect obstacles.

The system will make use of features from a modern day smartphone while keeping the interface simple with speakers reading aloud the options a user can choose from. GPS will need to track the navigation progress, also relaying direction information out of the speaker. Graphical setup stages will be included and expected to be configured by a caretaker. Voice recognition modules will be most important to allow real life usage in the blind community. Detecting obstacles with the camera will alert the user when something has obstructed their path. To deal with failures, an alert will be sent out to registered emergency contacts.

1.2. Project deliverables

- **Phase 1**
 - Preliminary Project Plan
 - Requirements Specification
 - Architecture Specification
 - Component/Object Specification
 - Slideshow Presentation
 - Prototype Mock-up
- **Phase 2**
 - Modifications to Project Plan & Requirements Specification
 - Final Prototype
 - Functional Demonstration

1.3. Evolution of this document

The project plan documentation will be revised periodically throughout its lifespan throughout the project deliverable.

The document will be updated based on the following changes throughout the project's development cycle:

- **Software Specifications**
- **Technical Structure**
- **Workflow Schedule/Timeline**
- **References**

1.3.1. Revision History

		Update Comments
		Initial draft created.

1.4. References

- 1.4.1. Will add more when needed.
- 1.4.2. Project Scope
<https://app.perusall.com/courses/software-requirements-1/cpt-s-484-project-specification-i-v1-0>
- 1.4.3. Project Tool
<https://dotnet.microsoft.com/apps/xamarin>
- 1.4.4. Waterfall Model
<https://www.softwaretestinghelp.com/what-is-sdlc-waterfall-model/>
- 1.4.5. Case Studies
<http://faculty.washington.edu/jtenenbg/publications/shinoharaObservingSara-assets2007.pdf>

1.5. Definitions, acronyms, and abbreviations

Unified Modeling Language (UML) - A diagram with the purpose of visually representing a system along with its main actors, roles, actions, artifacts or classes, in order to better understand, alter, maintain, or document information about the system.

Internet Operating System (iOS) - An operating system used for mobile devices manufactured by Apple Inc.

Xamarin - An IDE that allows for the development of cross-platform mobile applications using C#.

Android - An open-source operating system used for smartphones and tablet computers.

Gyro sensors - A device that senses the change in rotational angle per unit of time.

Waterfall model - A breakdown of project activities into linear sequential phases, where each phase depends on the deliverables of the previous one and corresponds to a specialisation of tasks.

C# - An object-oriented programming language made by Microsoft.

Model View Controller (MVC) - A software design pattern commonly used for developing user interfaces which divides the related program logic into three interconnected elements. They include the *model* (data), the *view* (user interface), and the *controller* (processes that handle input).

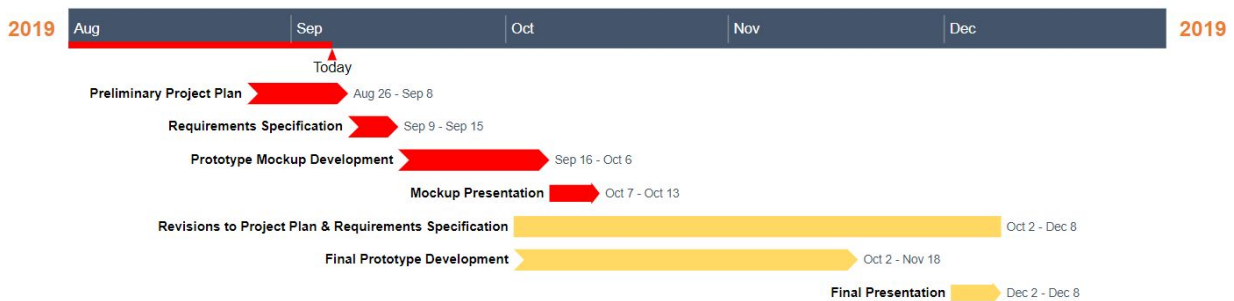
General risk - An activity or event that may compromise the success of a software development project.

Acceptance testing - A level of software *testing* where a system is tested for acceptability. The purpose of this *test* is to evaluate the system's compliance with the business requirements and assess whether it is acceptable for delivery.

2. Project organization

2.1. Process model

A modified Waterfall model will be used to structure our design process. Due to the time constraints of the semester, the waterfall method will be beneficial and to cover all aspects of the project we will update our project plan at each stage of the deliverables. During the process we will heavily use UML diagrams to solidify understanding and to give developers references when implementing the planned system. UML structure will ensure a good system structure by mapping out the proposed module connections.



2.2. Organizational structure

2.2.1. Team Members

- Christopher Young
- Cong Trinh
- Taryn Burns
- Anne Lin
- Sean Luchessa

Week	Team Members	Deliverable Description
1	All	Organize group
2	All	Continue to work on Project Plan Document together. Each section split up and assigned to different team members
3	All	Work on and Turn-in Project Plan Document, Work on WRS Document
4	All	Work on project documents, begin implementing

		software structure for the app, Work on WRS document
5	All	Project Phase 1 check-in meeting, continuous to work on Phase 1
6	All	Continue Phase 1
7	All	Continue Phase 1, review our progress before Phase 1 presentation and completion
8	All	Phase 1 Completed, Presentation, and Mockup
9	All	Turn to building Phase 2
10	All	Continue on Phase 2 of project,
11	All	Phase 2 check-in meeting, continuously work on Phase 2
12	All	Continue on Phase 2
13	All	Continue on Phase 2
14	All	Continue on Phase 2, Shoot to finish Phase 2 by the end of this week
-	-	Thanksgiving Break
15	All	Phase 2 Completed, Project Submission

One thing that is not shown within the table above but as a team, team meetings are done every Tuesday. We additionally talk through Discord and plan a lot of stuff through there.

2.3. Organizational boundaries and interfaces

- **Client/Mentor** - Professor Bolang will be requesting certain conditions to be met for the final prototype of the project.
- **Liaison** - Chris will be responsible for coordinating team meetings, updates, communications, and team deliverables.

2.4. Project responsibilities

- Christopher Young is the team liaison for this project.
- The project plan is currently evenly distributed to each member in the team.

- The project WRT is also the same way, with all parts of the document equally distributed to each individual team member.

3. Managerial process

3.1. Management objectives and priorities

The objective of our application is to provide a means for visually impaired people to more easily find their way to an inputted destination. The main objective of the management position is to keep all team members on board at all times. Weekly meetings are had to ensure this.

3.2. Assumptions, dependencies, and constraints (1)

This system relies on the setup performed by a caretaker who will be able to interact with UI elements. A map will also need to be provided to allow the application to make the route to follow. We will need to depend on the sensors provided from the device, and any failure in these will cause critical failures in the application. Another assumption is that the user will properly hold the device with the camera facing forward. We will also need to consider the problems that will arise from our user base. The app being targeted to the blind will need to have minimal ui requirements.

3.3. Risk management

3.3.1. Financial Risk

- Our finances are budgeted through what is needed in order to finish the project.

3.3.2. Team Risks

- The effectiveness of the project is based around how the team is able to properly communicate and deliver what is needed for the project. If something falters, then it can potentially set back the entire project altogether.

3.3.3. Project Risks

- Upcoming deadlines for deliverables can show where the project is at. When something fails in the software, especially during coding, can make the project fail in some way, whether it be having to change architecture or a way to approach the project. Everything revolves around how the app runs and is controlled. Every bug can set the team back.

3.3.4. Technology Risk

- We are using software that will make our app be applicable on Androids and Iphones. However, there are always issues with having apps transfer and be used on both devices.

3.3.5. People Risk

- ~~Our application is specifically designed for people who have sight disabilities but everyone can use the app. However, we want to keep in the forefront that issues can go wrong, specifically if the person does not have a helper with them, which brings to the point that smartphones have a changing screen that does not have any sort of Braille-like feature.~~
- (revision-1): Potentially not as profitable as they would like due to the requirement of a smartphone.
- (revision-2):
- Due to its current system structure, having to manually coordinate new maps to load with the phone application can limit the use of the application in specific areas.
 - Can be frustrating for caretakers and maintainers to map out new area maps to load into the software.
- Since the map layout is currently planned to be hardcoded, this would limit the ability to change the map layout with ease as the current system would be coupled to the code itself.
 - Such inflexibility can cause more work for programmers and maintainers to update if an existing map's layout were to change.
- The application is developed with a specific demographic in mind, so it could potentially hinder the opportunity for marketers and developers alike to appeal the use to a greater audience.

3.3.6. Structure/Process Risk

- The biggest risk is how efficient the work will progress throughout the semester. Additionally how everything plays out in the overall effectiveness of the app to be a deliverable prototype.

3.4. Monitoring and controlling mechanisms

- Periodic Meetings

- Weekly meetings will be held every Tuesday/Thursday in order to:
 - Distribute workloads.
 - Check-up on current progress with each member.
 - Discuss project's direction.
- **Shared Document Repository**
 - Any changes to the project specification during its development will be reflected onto the revisions of the shared document.
 - A backup of each major iteration of the document should be saved on the document's cloud platform.
- **Stable Update Releases**
 - Updates to the shared repository for the project software itself will maintain a stable version of each iteration on the master branch.
 - Current branches that are a work in progress will be developed separately on a "staging" branch, pushed to by different team members' respective development branch.
 - A backup of each major milestone in the repo should be saved onto the branch.
- **Specification Control**
 - A version of each platform and tool used in the process of developing the targeted software will be recorded on the shared document or repository readme.
 - Each member is responsible for utilizing the correct version of software/tools during development to maintain development consistency.

4. Technical process

4.1. Methods, Tools, and Techniques

- **Language:**
 - C#, SQL
- **Platform:**
 - Android, iPhone
- **Tools:**
 - Software:
 - Visual Studio, XNA, Xamarin, Gyro sensors, Google Maps, Phone's hardware/software and camera, simple images for testing, databases.
- **Techniques:**
 - Clean formatting and precise variable names, MVC
- **Methods:**
 - Scheduled team meetings in order to review what has been accomplished, what needs to be done, scheduling out various milestones and meetings with Bolong Zeng, and various ideas and discussion for the project.
 - Recognizing strengths and weaknesses of each of the team members in order to determine which role and responsibilities is the best fit for them. The workload will more or less be equally shared by all members of the team.
 - Constant review of the documents and the work done by each team member in order to ensure the quality of each part of the project.

4.2. Software documentation

- Project Plan
- Project Specification
- UML Diagrams of Prototype Architecture
- Meeting Notes
- Test Documentation

4.3. Project support functions

- Technical Support
 - Maintaining and updating changes to existing maps loaded into the database, which includes adding/removing necessary paths if the building layout were to be altered.
 - Overall changes in the software will be recorded and committed to the repo, while major update releases reaching certain milestones and features will be documented, along with any obvious changes in design and architecture.
 - Test cases would rely heavily on actual user test results, since much of the technical features rely on smooth correlation and low runtime compilations.
 - Overall test cases should yield the minimum requirement for public use, taking factors into consideration such as ease of use and runtime performance.
- Non-Technical Support
 - Documentation on how to use the application will be necessary for caretakers or generic users, as well as documentation pertaining to the software project plan and requirements for current shareholders.
 - A braille manual would need to be considered for blind users, the main demographic of this application.
 - Technical support for users to contact and report any issues using the application in order to pinpoint and solve any issues/bugs in the code architecture.

5. Work elements, schedule, and budget

- 5.1. The budget will be given out by Bolong Zeng and the resources and equipment will include Androids, Xamarin, Gyro Sensors, etc.
- 5.2. The project lead will lead the team for each project phase. They will make sure to review and schedule each part out properly in order to complete each project task before the deadline.
- 5.3. The team will be following the Course Syllable schedule.
- 5.4. Good communication will always be emphasized as to reduce any risks/problems and increase our efficiency level.
- 5.5. Documents will be reviewed at the end of each project phase in order to see if any revision is necessary.