Project Management Plan

For

Blind Navigation In-Doors

Version 1.0 draft 1

Prepared by Charlie Wong, Alec Yeasting, Riley Hunter, Zach Gherman
Team CARZ - WSU CPT_S 484

9/17/2021

Revised: 9/18/2021

Table of Contents

Revision History

Name	Date	Reason for Changes	Version
Initial Creation	9/17/2021	initial draft	1.0 draft 0
Mod 1 Draft	9/18/2021	Modified Missing or Incorrect sections	1.0 draft 1

1. Overview

This project is intended to create a general navigation aid Android application for the use of blind people with limited assistance of seeing people. It will result in the delivery of three deliverables which will include presentation(s), code, and documentation by December 15th, 2021.

1.1. Project Purpose, Objectives, and Success Criteria

Purpose: Create a navigation aid for blind people.

Scope: Standalone app written in Java for Android devices.

Objectives: Handle object detection for collision avoidance using LIDAR, Handle indoor navigation using an IPS

Deliverables: A prototype app for Android that utilizes collision detection and or navigation intended for blind people.

Success Criteria: Aids a blind person in avoiding a collision or aids them in navigating a route indoors

Project Dependencies: We will use currently unselected libraries for LiDAR data processing and DepthNet code or structure for camera depth estimation

Shared Resources: Magnetometer, LiDAR device, WiFi, Bluetooth

1.2. Project Deliverables

Deliverable	Delivery Date	Delivery Method	Comments
D-1 Project Phase 1 Deliverables: ● Requirements Specification ● Mockup* ● Revised Project Plan ● Presentation*	10/17/2021	Online Submission	Mockup: UI description, User Manual Presentation: At least one slide out of the presentation on “creeping rates”.
D-2 Project Phase 2 Deliverables: ● TBD	12/12/2021	Online Submission	TBD. Required deliverables not specified at this time.
D-3 Prototype v1	12/15/2021	Online Submission	A working prototype to be delivered to Professor Bolong Zeng for grade.

1.3. Assumptions, Dependencies, and Constraints

Assumptions:

AS-1 The future user(s) will be blind individuals.

AS-2 The application CARZ is developing will be able to ***reliably*** perform collision detection and avoidance for blind individuals.

AS-3 The application will aid in navigation indoors.

Dependencies:

DE-1 LiDAR working on the test device

DE-2 WiFi working on the test device

DE-3 Bluetooth working on the test device

DE-4 Magnetometer working on the test device

Constraints:

CO-1 Possible test subject location is remote from team CARZ location; a good test may be difficult without monetary cost.

CO-2 Time/schedule developers can apply toward the project.

CO-3 Cost of hardware for testing

1.4. References

- Navigational BluetoothTechnology (IPS):
  https://www.bluetooth.com/press/bluetooth-enhances-support-for-location-services-with-new-direction-finding-feature/

- General IPS information:
  https://www.geospatialworld.net/blogs/indoor-positioning-indoors-gps-stops-working/

- Object Detection (DenseNets):
  https://towardsdatascience.com/depth-estimation-on-camera-images-using-densenets-ac454caa893

- Community Sources:
  Emeritus Professor Rod Moag; a member of the blind community.
  Spoke about difficulties with detecting objects/obstacles at torso and head height.

1.5. Definitions and Acronyms

Bluetooth: A standard for the short-range wireless interconnection of mobile phones, computers, and other electronic devices.

IPS: Indoor Positioning System.

LiDAR: Light Detection and Ranging, a remote sensing method using a special kind of camera to measure distances.

Likert Scale: A psychometric scale used to rate agreement with statements.

Magnetometer: An instrument used for measuring magnetic forces, especially the earth's magnetism.

SME: Subject Matter Expert.

SQA: Software Quality Assurance

Wifi: A facility allowing computers, smartphones, or other devices to connect to the internet or communicate with one another wirelessly within a particular area.

2. Project Organization

This section will detail the Process Model, Organizational Structure and Roles and Responsibilities of each member working on this project.

2.1. Process Model

For this project the team has chosen to adopt a Spiral Process Model (Annotations, Figure 1: Spiral Process Model). The Spiral Process Model has five main stages; Planning, Modeling, Construction, Deployment, and Communication.

Within the Planning stage, the team works on developing (and further, redefining) and defining requirements (business, system, software, etc.). After the planning stage comes Modeling which covers the design and specification. This includes conceptual designs, logical designs, use-cases, etc. Construction covers the creation of a proof of concept, and prototypes. Deployment and Communication work hand-in-hand in this case.

Once the product has been deployed we communicate with the SME, stakeholders, and others involved in the project to discuss how the product can be expanded upon. Communication also deals with performing Risk Analysis upon the system after each deployment as well as reviewing test documentation, design specs and requirements. After each deployment, the process model's process starts over doing planning, modeling, construction, deployment and communication for the next iteration of the project.

2.2. Organizational Structure

The top of our organizational structure consists of the product manager Bolong Zeng who will oversee determining the success of the project on completion. For team communication with professor Zeng, Alec Yeasting will act as the team liaison communicating between the development team of Alec Yeasting, Riley Hunter, Zach Gherman, and Charlie Wong. The authority for making development decisions will be decentralized and solutions to development solutions will be arrived at through negotiation. For intra-team communication the software application Discord will be used to coordinate project work and discuss solution attempts. Zach Gherman, in addition to working on the software, will also provide QA for the software.

2.3. Roles and Responsibilities

- Project Manager - Alec Yeasting (Responsible for D-1 and D-2 communication organization)
- Product Manager - Bolong Zeng
- Hardware Lead - Riley Hunter
- Requirements Analyst - Charlie Wong, Alec Yeasting, Riley Hunter, Zach Gherman
- Software Engineer - Charlie Wong, Alec Yeating, Riley Hunter, Zach Gherman
- Hardware Engineer - Riley Hunter
- Test Engineer - Charlie Wong, Alec Yeasting, Riley Hunter, Zach Gherman
- Quality Assurance Engineer - Zach Gherman (Responsible for SQA on D-1, D-2, and D-3)
- Subject Matter Expert - Rod Moag (Blind Individual)

3. Managerial Process Plan

3.1. Management objectives and priorities

The main objectives in the management aspect of the team's work include:

- Good communication between all parties involved
- Project collaboration with **all** members of the team
- Extracting and optimizing the necessary inputs from the Client and SME
- Successful deployment of a working prototype of the product being built.

3.2. Assumptions, dependencies, and constraints

Assumptions: Each member working on the product has a basic understanding of software development. Each member working on the product is enthusiastic to work on it. Internal issues/conflicts will be resolved with haste.

Agreements: Members will work in a timely and efficient manner.

Constraints: Time, lack of hardware engineers, lack of expertise in IPS, lack of expertise in collision avoidance, lack expertise in LiDAR, lack of confidence.

3.3. Risk management

Plan for identifying risks

At every group meeting the project manager will go over any software or plan changes with the team and have an open discussion about the risks going ahead. If any risks are found the project manager will log the risk.

Plan for analyzing and controlling risks

Upon first identifying the risk a preliminary analysis will be performed by the team through team discussion about the significance of the risk and any mitigating steps that may be taken. If the team deems the significance of the risk high enough and the difficulty of mitigation low enough the appropriate measures will be put in place to mitigate the risk. Otherwise it will stay in the log. At the end of every team meeting the logged risks will be discussed to identify if there have been any changes to the significance or ease of mitigation.

Plan for prioritizing risks

When logging the risks the risks will be logged with a current significance level/impact and an ease of mitigation using 5 point likert scales. The combination of these two scores will be used as the risk priority.

Estimated percentage of project effort: 10%

The amount of time spent on risk mitigation is not estimated to be very high as any risks are unlikely to come out of nowhere and will likely be seen well in advance allowing us to mitigate them sooner.

4. Technical Process Plans

4.1. Methods, tools, and techniques

Methods: As stated in Section 2.1, the team will be following a Spiral Process Model. The team envisions following a Test-Driven Development methodology. This means that the team will be designing test cases for the Use-Case(s) created before any code concerning functionality is written. This will ensure validity of code and conformance to the Design and Requirements Specification Documents.

Tools:

- Hardware/OS: **Android**
- Programming language: **Java**
- Development environment (IDE): **Android Studio**
- Version Control and documentation: **Github**
- Design modeling and requirement management: **UML (Use-case, ACD, class and component diagrams, made with draw.io)**

Techniques:

- Requirement collection
- Analysis & Design
- Testing

Coding Standards:

All pushed commits made to multi-user branches should:

- Use proper indentation (tab based) in code
- Have commit messages that are clear, concise, and accurately describe modifications
- All variable and function names should be intuitively descriptive
- Contain function header comments
- Contain function body comments for functions over 10 lines of code

Software Design Standard:

The components of the software should have high cohesion and low coupling

Software Quality Assurance Procedures:

To ensure a quality product, the team will perform regular Peer Review of code, and designs and requirements specifications. This will ensure that the product being created will have less bugs in both the software (code) and physical documents (designs and requirements specifications). To perform software testing, the team will use JUnit to ensure the integrity of the product being created.

4.2. Software documentation

Document	Template or Standard	Created By	Reviewed By	Target Date	Distribution
Doc-1 Software Project Management Plan	Template	CARZ	Team	9/19/2021	Canvas, Github
Doc-2 Requirements Specification	Template	CARZ	Team	10/17/2021	Canvas, Github
Doc-3 User Manual	Standard	CARZ	Team	10/17/2021	Canvas, Github
Doc-4 UI Description	Standard	CARZ	Team	10/17/2021	Canvas, Github
Doc-5 Revised Phase 1 Plan	Template	CARZ	Team	10/17/2021	Canvas, Github
Doc-6 Meeting Records	Standard	CARZ	Team	10/17/2021	Canvas, Github
TBD	TBD	CARZ	Team	12/12/2021	Canvas, Github

A. Annotations

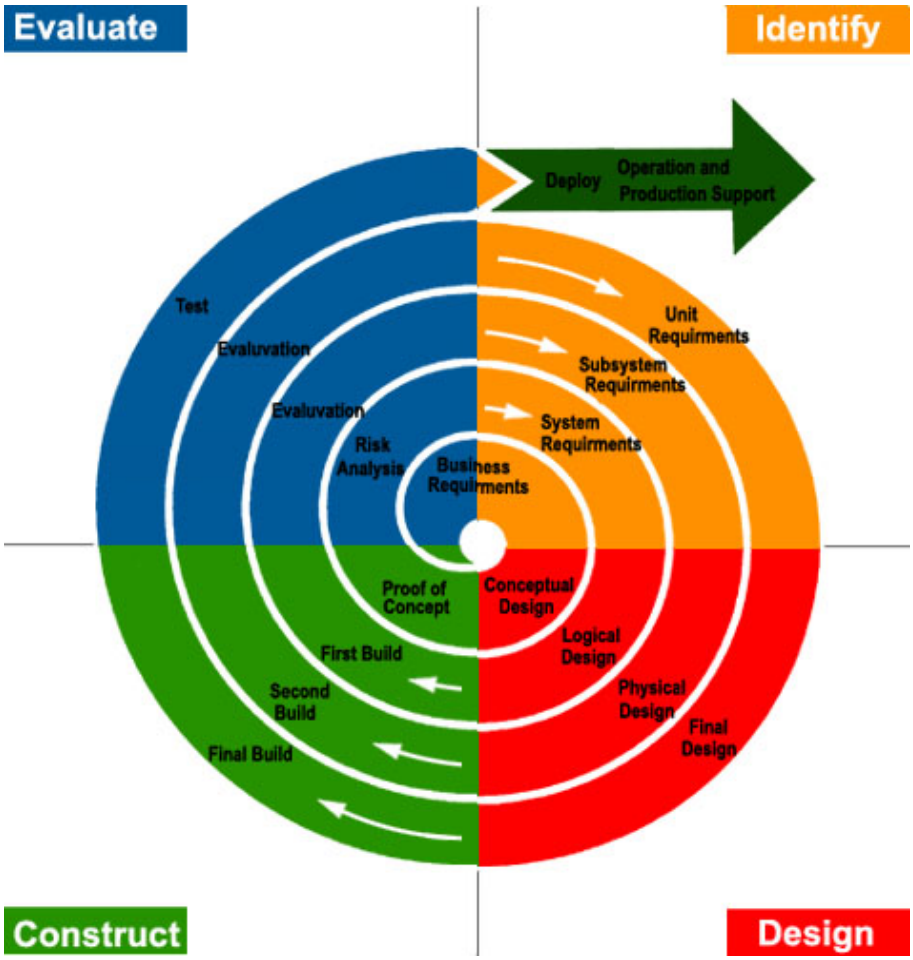

Figure 1: Spiral Process Model, https://eternalsunshineoftheismind.files.wordpress.com/2013/02/i-s-spiral.jpg