Software Project Management Plan

SEP UG07

Software Engineering and Project
The University of Adelaide
21 October 2018

version 2.0

a1758818	Cameron Youngblood
a1693458	Jiming Li
a1687323	Cadeau Maniragaba
a1694970	Zaifeng Wang
a1696425	Chao Zhang
a1685651	Erik Praekelt
a1658197	Byeongjun An

Table of Contents

1. Introduction

1.1 Purpose and Scope

To create a prototype robot using the Lego Mindstorm EV3 kit that autonomously locates and digitally marks the positions of humans in a designated operating area so that rescue can be dispatched to their specific locations after a disaster.

1.2 Intended Audience

This document is aimed at the software development team managers and the University Natural Disaster Search and Rescue Service (UNDSRS) project managers.

2. Project Organisation

2.1 Member Responsibilities

All group members will be taking on all responsibilities. Group members will divvy up each task as necessary for all members to do equal amounts of work. Specifically for documentation reviews, the table below shows who talk on which type of review:

Cameron Youngblood	Project Management Plan Review
Jiming Li	Configuration Management Plan Review
Cadeau Maniragaba	Risk Management Plan Review
Zaifeng Wang	Software Requirements Specification Review
Chao Zhang	Software Design Specification Review
Erik Praekelt	Code Review Review
Byeongjun An	Testing Review

3. Process

3.1 Process Model-Agile Method

Due to our groups small size and the benefits of having face to face communication and close collaboration, we will be using the Agile Method for bring the UNDSRS project to fruition. We will be using this model due to the ever changing requirements and ideas that this project may bring about. We will create the base code of the robot and continuously build upon it until it satisfies all of our ideas and requirements. After the completion of the main requirements, the Agile method can be continuously used to add new features to the robot.

3.2 Weekly SCRUM

During the course of the project, we will have bi-weekly sprints. Once a week during each sprint, our team will hold a SCRUM meeting during our general group meetings. The meeting will lasting 15 minutes at the same time. During the meeting, each team member answer the following three questions:

1. What did you do since the last meeting?
2. What are you planning to for the rest of the week?
3. Are there any impediments in your way?

Event	Timeframe
General Group Meetings	Every week after client meetings, occasionally before client meetings if necessary
Requirement gathering	Once a week on Wednesday during client and general group meetings
Software design	8 August - 16 August
Code Management	16 August - End of the UNDSRS project
Testing	Testing each version at least once a week
Documentation	Written as necessary before the due date
Task Management	During the project as necessary

4. Risk Management Plan

A risk is a possibility or threat of damage, injury, liability, loss, or any other negative incident that is caused by external or internal vulnerabilities. A risk may be avoided through pre-emptive action. This section outlines the potential risks involved in the project as well as strategies that our team will be using in an event that a problem occurs. A wide-ranging risk analysis was performed on features of the project to identify every possible source of problems within the project. For each identified risk, the severity, likelihood, project impact and strategies used to minimise the risk were clearly defined. A higher severity shows a potential risk with big impacts on project workflow and product development. A higher likelihood shows a higher risk of the problem occurring.

4.1 Personnel Risks

4.1.1 Team Member Unavailable

Description: The project's progress could be affected if a group member is unavailable to do work for some reason.
Severity: Low
Likelihood: High

Impact: Work may need to be reassigned and in the worst-case, deadlines will need to be adjusted.
Strategies: If a team member is sick or unable to perfume there role the rest of the team shares that role. If 2 members are unavailable, the project manager will assign another half of the team to take over the responsibility of any necessary tasks.

4.1.2 Loss of Work

Description: During the project work may be lost due to technical failure or items going missing.
Severity: High
Likelihood: Low
Impact: Ideally work will backed up or it may need to be completely redone
Strategies: Due to the nature of the project, most of the work will be stored digitally and will be backed up regularly to either GitHub to minimise the impact of loss. Material items such as the EV3 will need to be stored securely in a locker and extra care will need to be taken during transportation of these items.

4.1.3 Failure to Meet Deadlines

Description: Due to insufficient planning or other unforeseen circumstances, deadlines may not be reached.
Severity: High
Likelihood: Low
Impact: Failure to meet a deadline will have a big impact on the project's performance and result. Unmet deadlines will need to be prioritized which may result
in other tasks being postponed and further progress of the project being impacted.
Strategies: Major deadlines are very well known by the rest of the team ,the project schedule and workflow can be adjustable to meet deadlines. Adjustments could include organising tasks, increasing workload and regulating short term goals.

4.1.4 Client Unable to Attend Meetings

Description: Client is unavailable for a meeting.
Severity: Medium
Likelihood: Medium
Impact: If the client is unable to attend a meeting, issues of discussion may be delayed for a week, and may have a significant impact on project progress that week. This may result in the delays to the project's development, with the likelihood that the deadlines will impacted and tasks needing to be reorganised.
Strategies: If the client is unavailable for a weekly meeting, small issues may be discussed with the client over email and larger issues may mean having to be organise someone other time that the client is available and if not possible the
issues will be worked around until it can be resolved.

4.1.5 Team Member Inexperience

Description: Due to the fast pace nature of the project and the many new technologies being intro-produced, there may be a lack of knowledge and skills within the team.
Severity: Low
Likelihood: High

Impact: Mistakes may be made in the implementation of features, the project's progress may be impacted and a lack of experience may mean that errors are left unnoticed or difficult to correct.
Strategies: In order to overcome this risk, members are expected to pick up new skills and knowledge during the project, share any current or gained knowledge with their team member where relevant and discuss with the group any problems faced. during group meetings or informally over Discord.

4.2 Hardware and Technological Risks

4.2.1 Robot Damage, Hardware Failure or Loss

Description: The robot may be damaged due to misuse or hardware failure.
Severity: High
Likelihood: Low
Impact: Parts may need to be replaced and this could mean redesigning the robot and/or software.
Strategies: Extreme care will be taken when handling the robot. Members will avoid operating the robot in areas where it could be damaged. Random failures may be unavoidable, therefore the aim will be to structure the development process to minimize the impact of not having complete access to the robot.

4.2.2 New or Revised Requirement

Description: During the project new requirements may arise or existing requirements may be revised so the team may need to adapt accordingly
Severity: Medium
Likelihood: Medium
Impact: Depending on the type of requirement, a new requirement may completely change the project's course. Adjusting to new requirements may mean needing to prioritize work to meet deadlines.
Strategies: Sprints will be utilized to handle these changes and To avoid large impacts on the project, software will be implemented with a high degree of modularity.

5. Work Plan

5.1 Milestones

Milestones are used to have something significant accomplished by the set time in each milestone. The milestones are as follows:

- Milestone 1 negotiation on week 6 and delivery on week 8.
- Milestone 2 negotiation on week 8 and deliver on week 9.

Features that need to be accomplished for Milestone 1:

1. The robot moves in a straight line and moves around the edges of the map.
2. The robot should detect obstacles and survivors and display a message.

Features that need to be accomplished for Milestone 2:

1. Log humans when they are found and display them in the GUI on a map.
2. Create physical map for the robot to travel on.
3. The robot should be able to autonomously move through a map and stay within the map's boundaries.
4. The robot should avoid humans and obstacles.
5. The GUI should contain a way to manually control the robot.

5.2 Sprint Reports

Group Sprint reports are written to show what task was accomplished in each sprint (every two weeks) and and who was responsible for each task. Sprint reports will also include the group meeting minutes.

5.3 Schedule allocation

Team members will meet every week for a minimum of 2 hours to continue implementation.

6. Supporting Plan

6.1 Tools & Programs

The team will be using different tools and programs to complete the project. These include:

- Task Management & Logistics
    - Trello
    - Slack
- Programming
    - Eclipse IDE with Java
    - leJOS EV3 (Version 0.9.1-beta)
- Version Control & Code Management
    - Github (Hosted by the University of Adelaide-github.cs.adelaide.edu.au)
- Documentation
    - Google Docs

6.2 Code Management

The enterprise Github repository will store all code. The "master" branch will contain only complete and functional build-capable versions of the project. The "development" branch will contain the amalgamated source code of team members. Each team member will have their own branch to make changes. Changes are then merged into the "development" branch. Pull requests will be used to merge the "development" branch into the "master" branch.

6.2.1 Code Review

Each member is responsible for the code they contribute. Prior to merging pull requests into the "master" branch, the code reviewer will ensure code to be merged meet convention and does not deteriorate the condition of the "master" branch source code.

6.2.2 Code Conventions

The code for this project will be written in Java, since leJOS is based in Java. The source code is to adhere to the following conventions:

1. camelCase method, variable and class names (e.g. Robot.trackMovement)
2. Indentation will use 4 whitespace (equivalent to 1 tab space)
3. Not allowed to use public variables

6.3 Documentation Management

Documentation will be created via Google Docs and updated on the repository every Tuesday night in PDF format. Discussion about documentation will take place in the associated document's Trello card and Slack channel (Reference section *6.3. Task Management*). Comments within the Google Doc for minor matters are also permitted.

6.3.1 Document Conventions

For simplicity, most default Google Doc settings will be used, with the following exceptions and specifications:

1. Page margins will be 2cm on all sides.
2. N'th-level titles will use the default Heading N formatting. This allows Google Docs to automatically update the Outline indices.
   e.g. *6.2. Documentation Management* will use Heading 2 formatting.
3. N'th-level titles will be sequentially numbered, following by a full-stop (except last level digits). e.g. Note the formatting of the following: *6.2.1 Document Conventions*
4. Page numbers will be included in the footer (bottom-right corner) of each page, excluding the title page. Numbering commences on the page following the title page.
5. Table of Contents will be automatically generated with Google Docs. Using correct N'th-level headings makes this possible.

6.4 Task Management & Group Logistics

6.4.1 Trello

Trello will be used for task tracking and management. Within the UG07 team, two boards will be used:

- Admin & Docs - Tasks related to project documentation and administration
- Bot Development - Tasks related to the EV3 bot development and issues

6.4.2 Slack

Slack will be used for project discussions and group organization and logistics. A dedicated channel will exist for different types of logistics and organization, namely:

- #clientmeetings - Discussion relating to client meeting logistics

- #development - Discussion relating to the EV3 development
- #documentation - Discussion relating to documentation logistics
- #general - Discussion relating to general group conversations and logistics
- #groupmeeting - Discussion relating to group meeting logistics