



Software Project Management Plan

for

Archaeology Robot Group 13

Yufeng Bai 1600095	
Dawei Geng 1219181	
Jun Chen 1206265	
Quang Khoi Nguyen 1187070	
Shikai Li 1214223	
Yunyao Yao 1203525	
Yatong Zhou 1204471	

Version 0.1
September 1, 2012

Contents

1	Introduction	5
1.1	Purpose and Scope	5
1.2	Assumptions and constraints	5
1.3	Project deliverables	6
1.4	Evolution of the plan	6
2	Reference	7
3	Definitions	8
4	Project Organization	9
4.1	Roles and responsibilities	9
4.2	Risk management plan	11
4.2.1	Purpose	11
4.2.2	Risk Assessment	11
5	Process model	17
5.1	Waterfall Model	17
5.1.1	Advantages	18
5.1.2	Disadvantages	18
5.1.3	Minimise the Negative Impact	18
6	Work plan	19
6.1	Work activities	19
6.1.1	Poster	19
6.1.2	Robot Physical Design	19
6.1.3	Robot Testing	19
6.1.4	GUI	19
6.1.5	Manual Control	20
6.1.6	Wall Detection And Avoidance	20
6.1.7	Hidden Wall Detection	20
6.1.8	No-go Zone Detection And Avoidance	20
6.1.9	Speed Control	20
6.1.10	Path Finding	21
6.1.11	SRS	21
6.1.12	SPMP	21
6.2	Milestones	21
6.2.1	Week1	21
6.2.2	Week2	21
6.2.3	Week3	21
6.2.4	Week4	21
6.2.5	Week5	22

6.2.6	Week6	22
6.2.7	Week7	22
6.2.8	Week8	22
6.2.9	Week9	22
6.3	Schedule allocation	22
6.4	Resource allocation	22
7	Supporting plans	23
7.1	Configuration management plan	23
7.1.1	SVN Control Respository	23
7.1.2	File Folder	24
7.2	Documentation plan	24
7.2.1	Poster	25
7.2.2	Software Requirements Specification	25
7.2.3	Software Project Management Plan	25
7.2.4	Software Design Document	25
7.2.5	User Manual	25
7.2.6	Group Milestones	25
7.2.7	Minutes	25
7.2.8	Agendas	25
7.2.9	Testing report	26
7.3	Quality assurance plan	26
7.3.1	Verification and Validation process	26
7.3.2	Standard	26
A		28

Revision History

Name	Date	Reason For Changes	Version
Dawei Geng	21 Aug 2012	basic framework of the SPMP and Chapter 1	0.1
Dawei Geng	23 Aug 2012	Chapter 3 & Section 4.1	0.19
Dawei Geng	24 Aug 2012	Section 4.2.1 & 4.2.2	0.2
Dawei Geng	25 Aug 2012	Chapter 5	0.3
Jun Chen	25 Aug 2012	Spelling and grammar check , error fix	0.31
Yufeng Bai	27 Aug 2012	Chapter 7	0.32
Yaong Zhou	29 Aug 2012	Error Fix and Grammer Check	0.33

Chapter 1

Introduction

1.1 Purpose and Scope

This document aims to provide information that are required to manage the overall design, implementation and maintenance of the project of archaeology robot.

This document will be divided in to seven parts, including the Introduction, Definitions, Project organization, Risk management plan, Process model, Work plan, and the Supporting plans.

1. **Project organization** describes the responsibilities for each member of the team and the reason behind this arrangement.
2. **Risk management** outlines the risks may be found during the development of the project, also includes risk analysis and the processes and strategies for the team members to control or overcome these risks.
3. **Process model** describes the process model will be used throughout the whole development. Including the critical paths and major stages taken in the development. Rational behind this choice will be provided, along with the advantages and disadvantages of this process model.
4. **Work plan** includes the tasks will be done to deliver the product, and the milestones that the team will be meeting during the development. Work plan will also includes the duration of the tasks and the allocation of the resources to these tasks.
5. **Supporting plan** outlines the configuration management plan, the documentation plan, and quality assurance plan.

1.2 Assumptions and constraints

This project will be developed under such assumptions and constraints:

1. Every group member has a laptop/desktop installed LeJos Driver.
2. Every member's computer needs to install JRE and SDK.
3. Every member's comuter needs to have access to Internet and group's SVN repository.
4. A Bluetooth connection can be created between the robot and every team member's PC.
5. A Lego Mindstorm NXT robot will be used in this project.
6. This project has a duration of 11 weeks.

7. Every member have the knowledge or skills to carry out the tasks which are assigned to them.
8. This project requires at least 10 hours per week workload for every team member.

1.3 Project deliverables

The following items will be delivered to the client throughout the development of this project.

1. Team Poster
2. Software Requirement Specification (SRS)
3. Software Project Management Plan (SPMP)
4. Software Design Document (SDD)
5. Testing Report
6. User Manual
7. Project Milestone Demonstration
8. Final Project Demonstration

1.4 Evolution of the plan

This plan will be made by the team during the requirements elicitation and system design. The plan shall be made by gathering the feedback from the clients, and negotiation with clients about the milestones. In the software developing stage, the plan shall be changed under such circumstances:

1. Requirements changed by the clients
2. A defect of the plan was found
3. Other unexpected situations

At the planning stage, clients shall be able to change or raise requirements at anytime. However once the plan have been made and the team enter the developing stage, if a change is needed, the clients shall negotiate with the project manager along with team members to achieve the changes.

If a defect of the plan was found. It will be raised in the group meetings and the whole team shall discuss and respond to the possible changes.

Other unexpected situations such as team member missing and damaged hard drives, will lead to the whole team meeting with the client to discuss how to overcome such situations.

Chapter 2

Reference

Chapter 3

Definitions

Acronym	Definition
Gantt Chart	A type of bar chart, which illustrates a project schedule.
GUI	Graphic User Interface
L ^A T _E X	Document markup language for T _E X typesetting program
QA	Quality assurance
SEP	Software engineering project
SPMP	Software Project Management Plan
SRS	Software Requirements Specification
SVN	Subversion repository
T _E X	A system for computer typesetting. Used by L ^A T _E X
XML	Extensible Markup language

Chapter 4

Project Organization

4.1 Roles and responsibilities

The team for this project is made up of seven members, which will be divided into project manager, documentation manager, QA manager, developer, and QA engineer. Each team member will have one or two specific role, however, a team member will not have to be limited to his role and be able to help others. In this chapter, each role's main responsibilities will be explained and the team members who assigned to the role will be listed.

Role: Project Manager

Member: Dawei Geng

Responsibilities:

1. Ensure the goal of this project to be achieved and client's satisfaction.
2. Develop project milestones and overall project plan.
3. Organizing team members and their roles to fulfill the project needs.
4. Ensure the project runs according to the milestones and project plan.

Rational:

Being a project manager requires leadership, team spirit, problem solving skills, and much more. Dawei Geng have worked in groups and stood out when a plan is needed for the project. In case of leadership, team working, and project planning, Dawei Geng did a very nice job.

Role: Documentation Manager

Member: Yufeng Bai

Responsibilities:

1. Organizing the project's documents and keeping documents up to date.
2. Help project manager organizing meeting agendas and minutes.
3. Keep tracking and updating the SVN repository.
4. Spelling and grammar checking.

Rational:

As a documentation manager Yufeng Bai must take full responsible for all the document changes and delivery, this job requires responsibility and professional English skills to manage the documents. Yufeng Bai meets certain requirements.

Role: QA Manager

Member: Nugyen Khoi

Responsibilities:

1. Planning, organizing all the test-related tasks
2. Setting up the test strategies
3. Reviewing the code and test plans with QA engineers.

Rational:

Being a QA manager, besides of being a good QA engineer, it requires additional skills such as communication, creativity, and certain skills of team management. Nugyen Khoi can be a qualified QA manager.

Role: Main Developer

Member: Nugyen Khoi, Yatong Zhou, and Yaoyun Yao

Responsibilities:

1. Develop the software of the robot and the host machine based on the requirements.
2. Follow the reset milestones and project plan.
3. Deliver high quality code and software.

Rational:

Role: GUI Developer

Member: Yatong Zhou and Shikai Li

Responsibilities:

1. Developing the graphic user interface.
2. Provide test GUI to the main developers.

Rational:

Yatong Zhou and Shikai Li have great experiences of developing graphic user interface using Java and understand the concept of presenting the useful information to the user in a elegant, efficient way.

Role: QA Engineer

Member: Shikai Li and Jun Chen

Responsibilities:

1. Be responsible for all the test-related tasks.
2. Follow the testing plans.
3. Code reviewing.
4. Writing test report.

Rational:

4.2 Risk management plan

4.2.1 Purpose

Risk is defined as the events which occurred during the development of the project, and could have positive or negative impact to the development. A certain risk may be caused by one or more causes and may have one or more impacts. This plan document's processes, tools, and procedures will be taken to manage or control the events which could have negative effect to the project. It will include:

1. Foreseeable Risks Identification.
2. Risk's Impact Prediction.
3. Risk's Likelihood Prediction.
4. Risk Response Plan.

As summary, this plan will list each identified risk based on the priority: the likelihood of certain risk and the impact of it. A plan in which to reduce the risk from occurring will be suggested.

4.2.2 Risk Assessment

This section will list each identified risk and give it a ID initialed with letter "R". The probability and the impact of each risk will be represented as the category of "Low", "Medium", and "High".

R001: Team member unable to work **Probability:** High **Impact:** Medium

Description As a three months project, team members who get sick or have personal affairs which unable to perform enough in the project, someone else in the team will have to double the workload.

Indicator A team member does not attend to the lecture and the meeting with no work submitted for over one week or call for absence.

Mitigation All the work shall be evenly assigned. Every group member needs to pay attention to his/her health condition.

Response Plan Divide missing team member's workload into parts and assign them to other group members who are available to do it.

R002: Team member leave the project permanently **Probability:** Low **Impact:** High

Description Due to possible reasons, one team member may withdraw the course. He/she leaves the project permanently.

Indicator The team member notify the team with certain intention.

Mitigation The importance of each team member's work shall be roughly the same, means on one responsible for the overly important job.

Response Plan An emergency meeting will be held within the group immediately. The leaving member's original unfinished task will be reallocated to the remaining members.

R003: Conflict between group members **Probability:** Medium **Impact:** Low

Description Argument or conflict between team members about the developing or design.

Indicator Team members are arguing or refusing to complete their tasks.

Mitigation Every issue regarding the design or implementation of this project will be discussed in the internal meeting or client meeting. The solution of this issue shall satisfy every team member.

Response Plan An emergency meeting will be held within the group to discuss the controversial issue. A voting process may be held.

R004: Fail or late of delivery **Probability:** Medium **Impact:** Medium

Description Team member's deliverable is not on time.

Indicator Late commits or no commit over the preset deliverable deadline.

Mitigation Ensure team members have proper workload, understanding of the importance of deliver on time.

Response Plan Other team members shall help the team member who fail or late of delivery to make the project follow the project plan.

R005: Lack of contribution **Probability:** Medium **Impact:** Medium

Description Team member fail to contribute as much as others in the team.

Indicator Team member has no or lack of submission to the SVN repository.

Mitigation Project manager shall separate the workload properly.

Response Plan When this problem is found, team leader or project manager shall "rebalance" the workload per capital.

R006: Project file lost **Probability:** Low **Impact:** High

Description Data or files may lost caused by computer damage.

Indicator Team member's hardware failure.

Mitigation Every team member shall keep committing to the SVN repository constantly.

Response Plan Data rescue methods will be carried out in order to retrieve the missing data or files.

R007: SVN failure **Probability:** Low **Impact:** High

Description The team's SVN repository is "offline" or does not work correctly.

Indicator Grope member continuously fail to commit or update changes from the SVN repository. In addition, system failed or other error messages are received continuously.

Mitigation A backup repository shall be set up by the documentation manager to ensure the failure of the main repository won't affect the current workflow.

Response Plan Try to contact course coordinator and get the approximate date of the system recovery. In the meantime, try to retrieve files from the main repository as many as possible and backup repository.

R008: Misplacing locker key **Probability:** Low **Impact:** Medium

Description The key was lost by team member.

Indicator The misplacing of the key was reported.

Mitigation Once the team gets the locker key, a backup copy of the locker key shall be made and kept in safe place. The person in charge of the key will ensure key is kept safe at all times.

Response Plan Report to the course coordinator, and move the robot to a safe place using the copy of the key.

R009: Damage to the robot **Probability:** Low **Impact:** High

Description One component of the robot or the entire robot was damaged by the team member.

Indicator Missing or broken parts of the robot or the robot won't work correctly.

Mitigation Team member shall take care of the robot parts and the intelligent brick.

Response Plan Report to project manager and course coordinator for the broken of the robot.

R010: Milestone failure **Probability:** Low **Impact:** High

Description Team is unable to meet preset milestone or major milestone which agreed with clients.

Indicator Team fail to implement certain the functionality by due date of milestone.

Mitigation Project manager shall check the progress of the project every day, and ensure the milestone will be finished on schedule.

Response Plan Renegotiation with client to set new milestones. Re-schedule the tasks to team members. Team member will put more workload into this project to ensure the new milestone will be finished on time.

R011: Requirement changes **Probability:** Low **Impact:** Low

Description Requirements pulled from client or internal meetings.

Indicator Client ask for changes of requirements or team decides to change requirements.

Mitigation In the requirements elicitation and milestone phase, the team shall make sure all the requirements are recorded and both sides(the team and the client) are satisfied.

Response Plan Based on new requirements' importance, the project plan shall be re-arranged. QA team shall make sure the change won't have negative impact on the project's quality.

R012: Bluetooth connection error **Probability:** High **Impact:** Medium

Description The team member's PC will not connect to the robot's intelligent brick.

Indicator Team member's PC is not able to pair with robot through Bluetooth, or could not find the device over the NXJ Browse and NXJ Control.

Mitigation 2011 or later model Macintosh Computer(Mac) has known issue of connecting with the robot, team members who facing such issue shall try to working with the Bluetooth-enabled Windows PC. On the other hand, team members who own a Windows PC shall check their Bluetooth before the project starts.

Response Plan Team member who working with a Windows PC, if it has no Bluetooth component, shall purchase or borrow a Bluetooth connector in order to connect to the robot. Other team members who only have Macintosh computer can work on the school computer to get access to the Bluetooth connection.

R013: Undefined risks **Probability:** Medium **Impact:** Medium

Description Undefined issues which have negative impact to the project happened.

Indicator Situations defined as a negative impact happened which could not find in the risk management plan and have no response plan recorded.

Mitigation The initial risk management plan shall be as comprehensive as possible.

Response Plan The one who find this issue report to the project manager. A emergency meeting shall be held to discuss such issue and overcome the risk by implementing the required response.

R014: SVN commit conflict risk **Probability:** Medium **Impact:** Medium

Description Two or more group members are editing a same document. One of the members finishes his part and commit to the SVN respository. Others will not able to commit since their edition is not the newest version.

Indicator One group member find that he cannot commit a file to the SVN respository, meanwhile, error message comes out continuously.

Mitigation The group member should backup the parts he contributed immediately. The group member should inform other members the conflict exists, and check out the newest version, add his back up part and commit respository.

Response Plan Tasks allocation needs to be reviewed in the next group meeting. It is necessary to make sure that no more than one group members working on the same document in different working spaces.

Chapter 5

Process model

For this project, the Waterfall Model has been chosen. Waterfall Model is a well known software process model, it has strict deadline which to help the team focus on the software delivery on time. On the other hand, the feature of having stable requirements can be very helpful to the team's development.

5.1 Waterfall Model

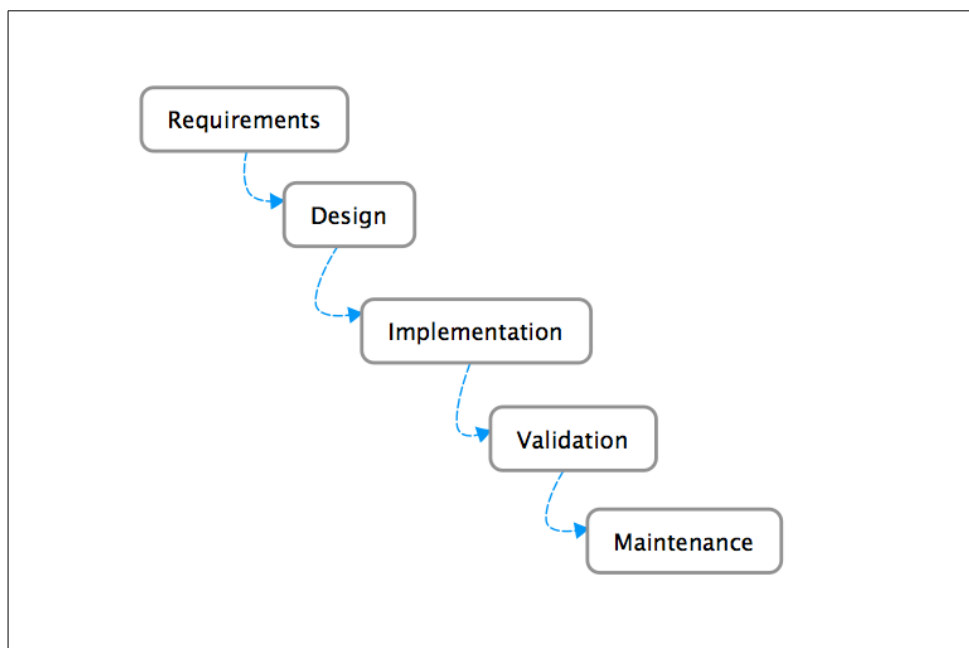


Figure 5.1: The Waterfall Model

The Waterfall Model is a well documented process model, which has more simple approach and disciplined. The waterfall model provides a structured development approach. It includes:

1. Requirements
2. Design
3. Implementation (Coding or construction of the project)
4. Validation (Testing or debugging)
5. Maintenance

Requirements At the Requirements specification phase, team will perform requirements elicitation based on the project description and the client meetings. All the selected requirements will be listed in the SRS document and implemented during the development phase.

Design: The Design phase is the phase that team members to design and establish a project plan. It is very important for the team to design all the aspects of the project.

Implementation: The main developers will start to implement the software of the project at this phase. Proper SRS documents and plans at earlier phases will be helpful.

Validation: After the implementation is done, the QA engineers will start the Validation phase, involving testing the overall system and ensuring all requirements have being achieved.

Maintenance: After the first release of the software, the team will need to maintain the system and fix possible bugs which would be found by users.

5.1.1 Advantages

1. Simple and disciplined linear software process flow.
2. Well written documentation will be produced earlier in the software life cycle.
3. Software release to the client on deadline.

5.1.2 Disadvantages

1. Not flexible to changing.
2. Late discovery of technical problems.
3. Can not have working software until the end of the development.
4. No testing stage in the early phases.

5.1.3 Minimise the Negative Impact

Chapter 6

Work plan

6.1 Work activities

6.1.1 Documents

Poster

Work ID:W000 Produce a poster that show the pictures and the important information(such as name,degree and speciality) of the all the group members.

Time Required: 2 Days

SRS

Work ID:W001 A Software Requirements Specification document (SRS) is produced. It define all the details as to the requirements of the robot software and the requirements of the host software.

Time Required: 1 Week

SPMP

Work ID:W002 Produce a Software Project Management Plan (SPMP), which provides all the information about how and who works on which aspect of the project and time arrangement.

Time Required: 1 Week

6.1.2 Robot Physical

Robot Physical DesignBuild

Work ID:W003 The robot must be built to complete all the tasks that are required. That is including detect all the elements, mark them on the map and auto-pilot.

Time Required: 3 Days

Robot Testing

Work ID:W004 Once the robot building is completed, it should be tested properly. These must include movement tests which tests that if the robot can moves by any directions correctly and smoothly on the map; accuracy tests which tests that if the robot can move accurately(for example if the program let the robot rotate 90 degrees, the robot should rotate exactly 90 degrees. Tests can be repeated to find small error); stability tests which aim to test that if the robot is likely to fall apart.

Time Required: 1 Day

6.1.3 Host Software

GUI

Work ID:W005 Produce a GUI that has map display, manual control button, battery capacity, Bluetooth status. It should be clear and easy to use. It allow user for full control over the robot.

Time Required: 5 Days

Manual Control

Work ID:W006 After the robot is built correctly and fully tested, the manual control software should be developed to achieve the basic movement: move forward, move backward, turn left, turn right. The robot can be controlled with PC keyboard or GUI.

Time Required: 5 Days

Speed Control

Work ID:W007 Add speed Control function into the GUI to allow users adjust speed while the robot is moving.

Time Required: 2 Days

6.1.4 Robot Software

Wall Detection And Avoidance

Work ID:W008 Robot can detect a wall in front and automatically find a path to avoid that wall. The robot should also mark that wall on the map. If the position of a wall is already known, the robot must avoid it and correct the position if there are errors.

Time Required: 2 Weeks

Hidden Wall Detection

Work ID:W009 Implement the function that let the robot can detect hidden walls by light sensor and mark them on map. The hidden wall is defined by different Grayscale.

Time Required: 1 Week

No-go Zone Detection And Avoidance

Work ID:W010 Be able to find the No-go zones and avoid them. The robot should also mark them on the map. If the position of a No-go zone is already known, the robot must avoid it and also correct the position or complete the border of that No-go zone.

Time Required: 1 Week

Path Finding

Work ID:W011 Add a path finding algorithm to the robot, let it automatically drive itself to the destination and avoid all the obstructer. The map is assumed to be correct, but when the robot finds an element that is not correctly marked on map it should correct it and recalculate path according to the new map.

Time Required: 1 Week

6.2 Milestones

6.2.1 Week1

Introduction All the group members will be introduced and will begin reading the requirements of project.

6.2.2 Week2

Poster A poster will be made to introduce all the team members to client.

6.2.3 Week3

Client Meeting The first meeting with client will be held this week. During the meeting everyone will be introduced to the client and may ask questions about the requirement. We should start to assemble our robot now.

6.2.4 Week4

Client Meeting The second client meeting will be held, and further requirements will be discussed.

6.2.5 Week5

SRS The first draft of SRS is completed this week as well as a basic version of GUI.

6.2.6 Week6

Robot Movement and Bluetooth Connection Some basic movement function should be now achieved, such as move forward, backward and 90 degree turn. The robot should also be able connected with computer via Bluetooth.

6.2.7 Week7

Manual Control Robot movement can be fully controlled by computer

6.2.8 Week8

Obstacles Location Robot will be able to locate Obstacles and avoid them, then mark them on the map.

6.2.9 Week9

Map Editor GUI should now be able to load map file and edit map. It can add or delete elements on the map.

6.2.10 Week10

Map Drawing and Auto-pilot Robot will fully pilot and discover the map by itself.

6.2.11 Week11

Error Correction and Improvement Tests will be run on the robot to find the error and correct them. Also if there are any improvement can be made, it should all be done at the end of this week.

6.3 Schedule allocation

ID: P

Name: Poster

Start: 31/07/12

End: 3/08/12

Duration: 4 days

Predecessors: None

Task: Design a poster for our group, with every group member's information. Print out the poster before the due date.

ID: SRS₁

Name: *SoftwareRequirementsSpecification1stDraft*

Start: 8/08/12

End: 20/08/12

Duration: 9days

Predecessors: None

Task: *DividetheSRSdocumentintopartsinthegroupmeeting. Everygroupmembercompletesthepartsheassignedto*

ID: SPMP₁

Name: *SoftwareProjectManagementPlan1stDraft*

Start: 20/08/12

End: 3/09/12

Duration: 10days

Predecessors: None

Task: *DividethefirstSPMPdraftdocumenttaskintopartsinthegroupmeeting. Everygroupmembercompletesthe*

ID: SDD₁

Name: *SoftwareProjectManagementPlan1stDraft*

Start: 27/08/12

End:14/09/12

Duration:15days

Predecessors:*None*

Task:*DividethefirstSDDdraftdocumenttaskintopartsinthegroupmeeting.Everygroupmembercompletetestthepie*

ID: TR

Name: Software Project Management Plan 1st Draft

Start: 10/09/12

End: 05/10/12

Duration: 20 days

Predecessors: None

Task: Test every implemented function during these four weeks, add the test result to the test report after every testing.

ID: UM

Name: User Manual

Start: 28/09/12

End: 19/10/12

Duration: 15 days

Predecessors: None

Task: User Manual is divided into different parts according to the robot's functions. Every group member is required to write the part of the function that he developed.

ID: MN₁

Name:*MilestoneNegotiation1*

Start:13/08/12

End:20/8/12

Duration:5days

Predecessors:*None*

Task:*None*

ID: MN₂

Name:*MilestoneNegotiation2*

Start:7/09/12

End:14/09/12

Duration:5days

Predecessors:*None*

Task:*None*

ID: GUI_F

Name:*MilestoneNegotiation2*

Start:7/09/12

End:14/09/12

Duration:5days

Predecessors:*None*

Task:*None*

6.4 Resource allocation

Chapter 7

Supporting plans

7.1 Configuration management plan

The configuration management plan demonstrates the basic plan about how we manage our project properly. It is important and necessary to make sure all items related to client's requirement are under control of the developers. The configuration management plan is also an approach to track the development and history of each developer's task. In our project, we mainly use the SVN control repository and File Folder for configuration management. The following is the specific explanation and structure of SVN and File Folder.

7.1.1 SVN Control Respository

Since every developer's documents and codes are saved in SVN control repository, it is convenient to check everyone's contribution to the project by client. SVN is also working as a database which is used by developers to look up the process and modification of each one. Then team members are able to understand the newest version of the documents and codes as soon as possible. Each group member has right to upload, download, check and modify(include delete) the files in SVN repository.

Here is the structure of SVN Repository:

1. SEP-Group13-Code This folder is used to save all code related to control the robot and PC in this project.

2. SPMP draft This folder is used to store all versions(except final version) of Software Project Management Plan report.

2.1 different versions of PDF draft

2.2 different versions of TEX draft

3. agendas This folder contains the agendas of the Client Meeting and Internal Meeting.

3.1 Team-13-InternalMeeting-"times of the meeting"-Agenda.pdf

3.2 Team-13-InternalMeeting-"times of the meeting"-Agenda.tex

3.3 Team-13-Meeting-"times of the meeting"-Agenda.pdf

3.4 Team-13-Meeting-"times of the meeting"-Agenda.tex

4. code samples This folder is used to collect all test codes for this project.

GUI

BTRobotServer

Robot Movement

test-codes

5. Documents This folder is used to store the final version of documents.

6. SRS draft This folder is used to save all versions(except final version) of Software Requirement Specification report

2.1 different versions of PDF draft

2.2 different versions of TEX draft

7. minutes This folder contains the minutes of the Client Meeting and Internal Meeting.

3.1 Team-13-InternalMeeting-"times of the meeting"-Minutes.pdf

3.2 Team-13-InternalMeeting-"times of the meeting"-Minutes.tex

3.3 Team-13-Meeting-"times of the meeting"-Minutes.pdf

3.4 Team-13-Meeting-"times of the meeting"-Minutes.tex

7.1.2 File Folder

File Folder is used to collect all materials which is printed out(including the minutes and the agendas). The feedback and the draft documents are also saved in the file folder. By using file folder, the client is easy to understand the progress of the project and is convenient to mention more requirements for us in every meeting.

The file folder is also used as a summary for every week in internal meeting. We use folder to record the complete part and incomplete part of the project. By this way, it is convenient for us to change the time schedule for every week.

7.2 Documentation plan

The project requires the group to produce professional quality documentation by using LATEX. All documents should use this format and it is necessary to store all documents in pdf and tex format in svn repository. The structure of each documents is required to base on the template which is post on the CS forum. For every document, we will use "divide and merge" method, which means every document is divided to different components, each group member takes responsibility to their own part. Then each group member gives a short presentation about what

they write and how they plan in every week's internal meeting.

The Required documents includes:

7.2.1 Poster

The poster is used to demonstrate the personal information of each group member including the position and the main job of each one in this project. The client is able to use poster to understand each group member's strengths and weakness.

7.2.2 Software Requirements Specification

The purpose of this documents is to record client's requirement according to the Project Description and the requirements from each client meeting. Each group member is required to write the components of Software Requirements Specification.

7.2.3 Software Project Management Plan

The purpose of this documents is to describe the tasks what need to do to reach the client's requirements. In this documents, it is also necessary to demonstrate the allocation of tasks to each group member. The client also requires to analyse the potential risks and demonstrate how to avoid and solve all these risks.

7.2.4 Software Design Document

The purpose of this report is to provide an overall architecture model of the project. In addition, we are also required to give a brief introduction on individual component task. In this document, it is necessary to use some diagrams to demonstrate the design of the whole project.

7.2.5 User Manual

The purpose of this document is to provide a guidance and instruction sheet to the client to show them how to control this whole system. In this document, the developers need to consider that some users do not have professional knowledge related to this project.

7.2.6 Group Milestones

The milestone is used to demonstrate the progress of the project to client. For writing this document, the group members need to negotiate the working process in the internal meeting. The milestone is able to be renegotiated by discussion.

7.2.7 Minutes

The purpose of the minutes is to record the working progress and client's requirement in each meeting, The minutes are divide as client meeting minutes and internal meeting minutes. The client meeting minutes are used to record the new requirement from the client, the internal meeting minutes are used to record the content of group members' discussion.

7.2.8 Agendas

The agendas is used to schedule the process of each meeting(client meeting and internal meeting). The client is convenient to understand what hope to be demonstrated by developer in every meeting.

7.2.9 Testing report

The purpose of this document is to ensure that all students participate the testing process for the software development. Every group member is required to write a testing report to demonstrate some test case.

7.3 Quality assurance plan

7.3.1 Verification and Validation process

Each group member is required to review and check the documents and codes which are written by other group members. It is necessary to make a standard including how to write the documents and the codes. In addition, we offer the position as spell checker and code checker to check and fix every document and code to make sure the documents and codes are written as the standard. If the documents or codes is too hard to understand or do not follow the standard, the checks is able to ask the developers to rewrite the documents or codes. In internal meeting, the checkers have the responsibility to demonstrate the final version to the rest developers to do the double checking together.

7.3.2 Standard

Coding standard

1. When the developers build the Java Folder, the developers are required to use the function of this java program as the folder name. For example, GUI developers are required to name the folder as GUI, the robot developers are required to name the folder as Movement or Sensor detection. In this way, other developers are convenient to find the needed java program.
2. The developers are required to write the developer's name, writing date and the brief explanation of this file's function on the top of each java file. This standard is used to make the java file more clear to be read. Meanwhile, code checkers is more convenient to check the author and the fix date.
3. The developers are required to write the specific comments for each method, which is convenient for others to read and save the reading time.
4. The group members who modify others' program are required to write the name and the fix date on the bottom of the java file. The modify part is also needed to be labelled.

Documents standard

1. The file name of each documents(except minutes and agendas) is required to write in format like this: Team-13-document's name-version. The client minutes is required to write in format like this: Team-13-Meeting-"times of the meeting"-Minutes, the internal minutes is required to write in format like this: Team-13-InternalMeeting-"times of the meeting"-Minutes. The internal agenda is required to write in format like this: Team-13-InternalMeeting-"times of the meeting"-Agenda. The client agenda is required to write in format like this: Team-13-Meeting-"times of the meeting"-Agenda.
2. Each group member is required to understand how to use Latex to write the file. When developers submit the documents, it is necessary to upload both the tex and pdf file.

3. The documents must be summited in the correct folder. For example, the minutes have to summit in the minutes folder. It is not allowed to summit the files in the wrong folder. The developer has to make sure the documents summit in right place.

4. If one developer wants to delete documents, it is required to negotiate with the other group members.

Appendix A