

Communication Model for Cooperative Robotics Simulator

Implementation Plan

Version 1.0

1 Introduction

This section provides an overview of implementation plan for Communication Model for Cooperative Robotics Simulator project.

1.1 Purpose

The purpose of this section is to provide Work Breakdown Structure of implementation phase for “Communication Model for Cooperative Robotics Simulator” project.

2. Work Breakdown Structure

Deliverable	Task	Completion criteria	Time	Cost
Revised Architecture Design	1) Refine class diagram	Approved by committee members	June 23	1 day
	2) Refine sequence diagram	Approved by committee members	June 24	1 day
Revised Formal Requirement Specification	1) Refine Formal Requirement Specification	Approved by committee members	June 25	1 day
Revised Test Plan	1) Refine Test plan	Approved by committee members	June 28	1 day
Revised Formal Inspection Checklist	1) Refine Formal Inspection Checklist	Approved by committee members	June 29	1 day
Component Design	1) Complete class diagram	The internal design of each component is documented	June 29 - July 30	1.5 days
	2) Complete sequence diagram	All activities of software is defined and consistent with individual components	July 1 - July 2	2 days

	3) Complete component diagram	All activities of software is defined and consistent with individual	July 5 - July 7	3 days
Code	1) Code Message module.	Executable codes are produced	July 8	0.5 day
	2) Code PriorityQueue module	Executable codes are produced	July 9	1 day
	3) Code RobotParameter module.	Executable codes are produced	July 9	0.5 day
	4) Code RobotCommRecord module	Executable codes are produced	July 12	1 day
	5) Code CommunicationsSystem module	Executable codes are produced	July 13	1.5 day
Unit testing	1) Produce modules for unit testing.	Modules are produced to conform to unit testing in test plan	July 14 – July 16	3 days
	2) Perform unit testing	All unit testing are passed along with acceptance criteria defined in test plan	July 19 - July 20	2 days
Integration testing	1) Produce modules for integration testing.	Modules are produced based on test plan	July 21 - July 22	2 days
	2) Perform integration testing	All tests defined in test plan are passed	July 23 ,July 26	2 days
System testing	1) Perform system testing with other modules (robot environment)	All tests defined in test plan are passed	July 27 - July 28	2 days
User Manual	1) Produce user manual	Approved by committee members	July 29 - July 30	2 days
Assessment Evaluation	1) Create testing result document	Unit, Integration and System testing	August 2 - August 4	2 days

		are details including test description and failures.		
Project Evaluation	1) Evaluate the usefulness of the methodologies used	Approved by project advisor	August 5	0.5 day
	2) Evaluate the accuracy of the estimations	Approved by project advisor	August 5	0.5 day
	3) Evaluate the usefulness of the reviews	Approved by project advisor	August 6	0.5 day
	4) Evaluate whether it accomplishes the SRS	Approved by project advisor	August 6	0.5 day
References	1) Compile references from all documents	All references are gathered	August 9	1 day
Final Document	1) Compile all document	All documents are gathered	August 10 – August 11	2 days
Formal Technical Inspection Letters	1) Collect Formal Technical Inspection Letters	Two Inspection letters are collected	August 12	1 day
Total cost				37 days

Based on COCOMO I model, Boehm mentioned that there are 152 working hours per month or 7 working hours per day. Therefore, the implementation phase will need $37 \times 7 = 259$ hours to complete. According to COCOMO I model, the time to complete this project is 3.74 months or $3.74 \times 152 = 568.84$ hours. So far this project spent less than 250 hours; it seems to be that this project will spend time less than the estimation.

3. Implementation Plan

3.1 Updated document

After the second presentation, some documents of architecture phase will be updated based on the project committee members' suggestion.

3.2 User manual

The user manual of the project will be produced. It will include an overview of the system, the usage of the commands. The user manual also includes the description of how robot communicates with each other, how to set the parameters and description of error messages.

3.3 Component design

The internal design of each document will be documented. The class diagram and sequence diagram will be produced and consistent with the individual component.

3.4 Source code

The source code will be produced correspond directly to the architecture and component design. The JavaDoc standard will be used for source code documentation.

3.5 Assessment Evaluation

The assessment evaluation document will provide the detail of all test done in the project including test description and test results.

3.6 Project Evaluation

The project evaluation will be documented. The process will be reviewed, including the usefulness of the methodologies used, the accuracy of the estimations, and the usefulness of the reviews. In addition, the project will be reviewed and evaluated for whether it accomplishes the ideas presented in the initial overview.

3.7 References

The annotated bibliography will include cited references for all notations used in the portfolio.

3.8 Formal Technical Inspection Letters

The two letters from Kevin and Estaban will be collected. The letters will state that they successfully participated in the project as an inspector and the project had passed the architecture presentation.