

Communication Model for Cooperative Robotics Simulator

Software Quality Assurance Plan

Version 1.0

1. Purpose

The objective of this document is to define the Software Quality Assurance Plan of the “Communication Model for Cooperative Robotics Simulator” which will be used throughout the software life cycle.

2. Reference Documents

- IEEE Guide for Software Quality Assurance Planning Std 730.1-1995.
- IEEE Standard for Software Quality Assurance Plans Std 730-1998.

3. Management

3.1 Organization

The organization of this project consists of a major professor, two committees, a developer, which is MSE student and two formal technical inspectors.

Major Processor

Dr. William H. Hsu

Committee

Dr. Scott A. DeLoach

Dr. William J. Hankley

Developer

Acharaporn Pattaravanichanon

Formal Technical Inspector

Esteban Guillen

Kevin Sung

3.2 Tasks

All the tasks in three phases: inception, elaboration and construction phase, to be performed in this project will be delineated in project plan.

3.3 Responsibilities

Major professor and committees are responsible for participating in three presentations and demonstrations given by the developer, give suggestion and provide feedback. In addition, major Professor will supervise, evaluate and approve every task done by the developer.

Developer will responsible for the entire task in developing this project throughout the software life cycle including planning, designing, implementation and documentation. The developer will responsible for scheduling the time for reviewing with major professor and committees for all three phases. Furthermore, the developer will perform all tasks under the supervision of the major professor and suggestion of committees.

Formal Technical Inspectors will responsible for reviewing the architecture design artifact and submit a report, which will contain cover letter and a formal checklist that prepared by the developer.

4. Documentation

4.1 Purpose

The main purpose of the documentation is to ensure the development, verification and validation, use and maintenance of the software. The required documents are followed according to the MSE portfolio guideline for MSE student at CIS department of Kansas State University.

4.2 Minimum documentation requirements

4.2.1 Software Requirement Specification (SRS)

The SRS will describe the essential functionality of the software and external interfaces. The SRS can be modified in architecture phase to ensure that all the essential requirements are included.

4.2.2 Software Design Description (SDD)

SDD depicts how the software will be structured to satisfy the requirements in the SRS. It describes components and subcomponent of software design. Object diagram, class diagram and sequence diagram will be created using Rational Rose.

4.2.3 Project Plan

Project Plan will delineate time period, job and milestone. It can be adjusted in each phase to present the actual status of the project.

4.2.4 Software Test Plan

Software Test Plan will address the required test to show that the product satisfies the requirements.

4.2.5 Project Evaluation

The development process will be reviewed and evaluated including the accuracy of the estimation and the usefulness of the methodologies.

4.2.6 User Manual

User Manual will provide an overview, user commands, error messages and data formats.

All documents will be posted on the developer web page before presentation of each phase.

5. Standard, Practices, Conventions and Metrics

5.1 Purpose

This section describes the standards, practice, conventions and metrics used in Communication Model for Cooperative Robotics Simulator project.

5.2 Content

5.2.1 Documentation Standards

The IEEE standards will be used to develop some documents (where applicable). All documents will have brief explanation describing purpose of the document and version number, which will be increased by 0.1.

5.2.2 Logic Structure Standards

UML notation will be used for the analysis and design documents such as use case diagram, class diagram and sequence diagram.

5.2.3 Coding Standards

The software will use all standards coding practices; for examples the indent for each control structure, embedded standard comment.

5.2.4 Testing Standards

Unit testing, integration testing and acceptance testing will be conducted during testing process. All modules must pass unit testing before doing integration testing. Acceptance testing will be based on the evaluation criteria, which are outlined in test plan.

6 Reviews and Audits

6.1 Purpose

All documentation and software produced during the project will be subjected to regular reviews to ensure the highest level of correctness and quality. This section describes how the reviews and audits are accomplished.

6.2 Minimum Requirements

The three presentations will be conducted during the software development process. All documents for presentation must be submitted to major professor and committees at least one week before presentation. Moreover, there are two formal technical inspectors, which are MSE students. They will participate in reviewing architecture. The formal technical inspectors will provide a review report, which will be one of the required documents of the final presentation.

7 Test

The test plan will provide all the test activities, evaluation criteria, unit testing, system integration testing, and acceptance testing. All tests must meet the evaluation criteria. The test report will provide all test conducted and test result.

8 Problem Reporting and Corrective action

Problem that occurs during the software development will be logged for tracking, resolving and reporting. Only critical problem that impact the project progression will be reported to major professor the others will be resolved, recorded and informed to major professor.

9 Tools, techniques, and methodologies

UML notation is used for the analysis and design documents using Rational Rose 2002. Microsoft Project will be used for creating project plan.

10 Code control

The developer uses her own computer and maintains her software version using several subdirectories. The code and the binary files will be organized in different directories. The backup copies will be created every week on CD-ROM.

11 Training

The developer will responsible for doing research about the system. The major professor and committees will provide additional knowledge of the system. Attending in weekly meeting of the Cooperative Robots Simulator research group will help in deeply understand of the system.