Implementation – robot/game project 1

1 Introduction

These are the instructions for the implementation phase of the Robot/Game project. The focus of this phase is to design, implement and test the robot/game application. The documentation to be delivered from this phase should focus on the test results when it comes to evaluating the robot/game application towards the requirements, and discussing the relation between the implemented the robot/game application and the architectural documentation. The implementation should be delivered both with source code and class files (where applicable), in addition to updates of the previous deliveries (Requirement, Architecture design, and ATAM*).

It is important that it should be easy to install and run the application.

2 Deliveries

The deliveries should be:

- Implementation document See below
- Application source code and class files (in a handy zip file).
- Architecture document (updated with changes listed)
- Requirements document (updated and with changes listed)
- ATAM document (If there are changes made)[1]

About the *change lists* of each document: They should not only list what has changed and how, but also why things have changed. If the "old" documents are not in the delivery, the original versions will be used for evaluation purposes.

3 Implementation delivery template

The implementation delivery should contain the following:

- **Front Page** that includes:
  - Group name
  - Group members
  - Chosen COTS (Robot, XNA, Andoid, iOS)
  - Primary chosen quality attribute
  - Secondary chosen quality attribute
- **Introduction** that includes:
  - Description of the project and the phase (implementation and testing).
  - If game project: Description of the game concept should be sufficiently described and explained here.
  - Structure of the document.
- **Design details / Implementation details** that includes:
  - More detailed description of how the game/robot controller was designed and implemented including complete class-diagram, description of the implemented classes etc.

[1] The ATAM delivery is a report about your opponent group' project which was evaluated and written by you.

- **User's Manual** that includes:
  - Description of how to install, compile and run the game/robot.
  - Screenshots from the game describing how to play the game.
- **Test report** that includes:
  - The report should contain test reports for both functional requirements and quality requirements (quality scenarios).
  - Test reports must include requirement ID (e.g. F1, F2, A2, A3..), description of requirement, who did the test, date for test, time used, evaluation (failure/success) and comment (discussion/comment about the result).
  - Quality requirement tests must in addition include stimuli, expected response measure, and observed response measure.
  - See figures below for example for reporting functional and quality requirement tests.
- **Relationship with the architecture** that must include:
  - This section should list the inconsistencies between your architecture and the implementation. Give the reasons for these inconsistencies. Discuss whether they could have been discovered at an earlier point, for instance during the ATAM evaluation.
- **Problems, Issues and Points Learned** that must include:
  - In addition to listing problems and issues with the document or with the implementation process, this is also a spot to reflect upon the project and discuss what you would have done differently if you were to start again from scratch.

F1: The robot should be able to pick up a ball 4 out of 5 times	
Executor:	Ola Normann
Date:	24.3.2013
Time used:	10 minutes
Evaluation:	Success
Comment:	The robot is able to pick up 9 balls out of 10 times

M1: Modify enemy movement	
Executor:	Ola Normann
Date:	05.10.2013
Environment:	Design time
Stimuli:	Change movement of enemy NPCs
Expected response measure:	20 minutes
Observed response measure:	50 minutes
Evaluation:	Failure
Comment:	Lack of coherence made changes more difficult and several modules had to be changed.