Implementation – robot/game project

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. The documentation to be delivered from this phase should focus on the test results when it comes to evaluating the robot/game towards the requirements, and discussing the relation between the implemented robot or game 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).

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)

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:

3.1 Introduction

A tiny bit of project context like the other documents. What the document consists, what the purpose is. It should also contain the frequently requested quality attribute that you focus on.

3.2 Design details / Implementation details

Here you describe a more detailed view of the various parts of the architecture describing how the robot controller or game was designed.

3.3 User's Manual

How to run the robot. How to compile it, run it, etc.

3.4 Test report

The report should contain test reports for both functional requirements and quality requirements (quality scenarios).

The test reports for the functional requirements should look something like this:

F1: The robot should be able to jump along happily...	
Executor: Date: Time used: Evaluation:	Super Mario III 23.3.2005 5min Fail: White robots cannot jump!

For the quality requirements something like this may be useful:

A1: The robot should not get stuck	
Executor: Date: Stimuli: Expected response: Observed response: Evaluation:	Snurre Sprett 24.3.2005 The robot should be able to move around for 10 min Success in 8 of 10 executions Success in 3 of 10 executions Fail

For each requirement, there should be a discussion about what is observed, unless the result is a trivial success.

3.5 Relationship with the architecture

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.

3.6 Problems, Issues and Points Learned

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.