Norwegian University of Science and Technology

# Requirement and Architecture Phase
TDT4240

GRUPPE A13:
Stefan Bui
Jim Frode Hoff
Christian Tverås
Jonas Foyn Therkelsen
Julie Johnsen Kirkhus
Erik Reimer

February 20, 2014

# Innhold

# 1. Introduction

In this project we are supposed to choose between making a functioning multiplayer game for either Android, Iphone or XNA, or make a robot simulator.

The documentation phase is intended for planning of the chosen architecture design for the development of the game for this project.

## 1.1. Project description

Our project is a curling game project for Android. The game shall follow simple curling rules and will be a turn-based multiplayer game.

The game will have a simple menu screen with the options to choose between tutorial, play game and exit. The game itself will utilize the touch function on android devices, where you will be able to send the ball with a given speed and manipulate the speed with swiping near the ball, either to decrease the friction for it to go longer or to add or subtract spin. to give the player a better feeling of how well he/she is doing the game when played will have a minimap to give a better overall view of the game.

The tutorial will guide the player through the rules of curling and how the physics inn the game works.

# 2. Requirements

## 2.1. Functional Requirements

| ID | Case |
|---|---|
| FR1 | Start a new multiplayer game. |
| FR2 | Send the curlingball down the lane. |
| FR3 | Make the curlingball curl. |
| FR4 | Use the broom to decrease friction, or change the spin/curl of the ball. |
| FR5 | The curlingballs that collide shall obey the laws of physics. |
| FR6 | Have a minimap that shows the overview of the game. |
| FR7 | Scores shall be updated according to curling rules. |
| FR8 | The game shall follow the rules of curling. |
| FR9 | The game shall be pausable. |
| FR10 | The game shall have a startmenu. |
| FR11 | The game shall have a "how to play" screen. |
| FR12 | The winner of the game shall be the one with the highest score. |

## 2.2. Quality Requirements

### 2.2.1. Modifiability

| ID | A1 |
|---|---|
| Source | End User |
| Stimulus | Wishes to change the number of rounds in the game |
| Environment | Runtime |
| Artifact | Configurations. |
| Response | The amount of rounds has been changed |
| Response Measure | Time, effort Should be less than 5 seconds |

| ID | A2 |
|---|---|
| Source | Developer |
| Stimulus | Reduce coupling |
| Environment | Design time |
| Artifact | Code |
| Response | Deploy modification |
| Response Measure | Time, Should be no more than 3 hours |

### 2.2.2. Usability

| **ID** | **A3** |
|---|---|
| Source | End user |
| Stimulus | Learn to play the game |
| Environment | Runtime |
| Artifact | System |
| Response | The game is intuitive |
| Response Measure | User satisfaction and feedback.<br>Task time, in 10 min you should be familiar with the game and its functionalities |

### 2.3. COTS Components and Technical Constraints

The only constraints are that we program this game for android and its intended for mobile devices. If we were to publish the game, we would also get some constraints from Google's distribution mechanics.

### 2.4. Issues

### 2.5. Changes

## References

[1] Len Bass, Paul Clements, Rick Kazman. *Software Architecture in Practice*– Addison-Wesley. 3rd edition, 2012