TDT4240

Software Architecture

Spring 2012

Group A17
Android SDK

Steinar Haram
Emil Andreas Mork
Stian Sørebø
Ole Jørgen Rishoff

Implementation Document v1.0

Primary focus:
Maintainability

Secondary focus:
Usability

Contents

List of Tables

List of Figures

1 Introduction

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

This document contains implementation details for our developed version of the classical *Nine Men's Morris* game. The game is developed as a native Android application. The applications primary attribute is modifiability, while its secondary attribute is usability. The second chapter will highlight the design and implementation details. The following chapter contains a user manual, while chapter four contains a brief description of the testing of functional and quality requirements. The relation between the implementation and the planned architecture will be reviewed in chapter five. IChapter six highlights encountered problems and gained experience.

2 Design and 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 User manual

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

3.1 Functional requirements

- Android OS v2.2

3.2 Running the application

The application is avalible at URL.
The Eclipse-project is avalible at ANOTHER URL.

3.2.1 Emulator

To run the application in the emulator, the user needs to open the project in Eclipse. File -> Open project -> Existing source code -> path to downloaded project

3.2.2 On Android device

The user has to connect the Android device and load the application file.
DETAILED (STEP BY STEP) DESCRIPTION NEEDED!

3.3 Game rules

The game is implemented with the same set of rules as the classic board game *Nine Men's Morris* [**?**]. The goal of the game is to either block any opponent moves, or to reduce your opponent's piece number to less than three. If you get three pieces in a row, you enter a morris state, and are allowed to remove one of your opponent's pieces. Pieces that are in a morris state, i.e. forms three in a row either horizontally or vertically, are not removable.

3.4 Creating Skiller account

3.5 How to play

3.5.1 Choosing game mode

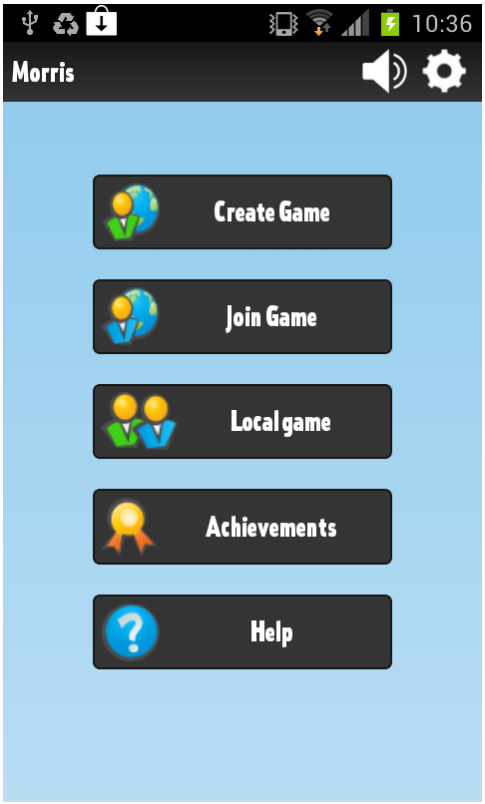
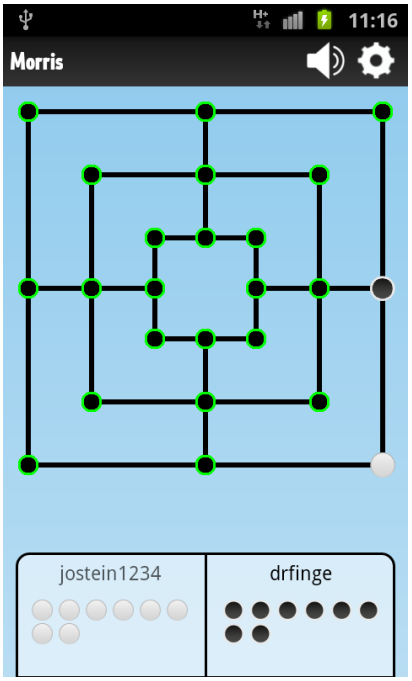


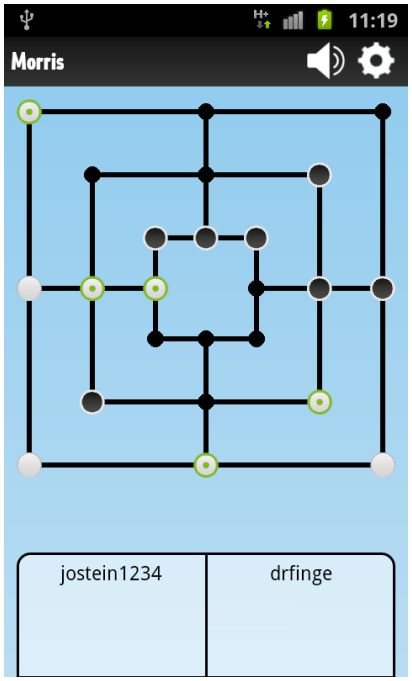
Figure 1: Available game modes

3.5.2 Placing, selecting, moving, and removing pieces

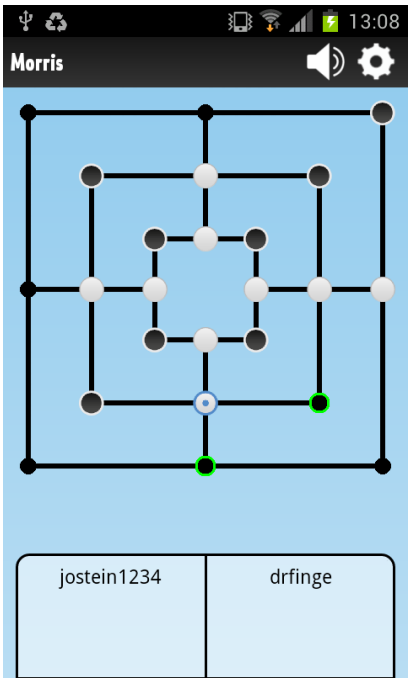
When it is your turn to move, either the board or your pieces will be highlighted. In addition, the name of the current player will be blinking as the game progresses.



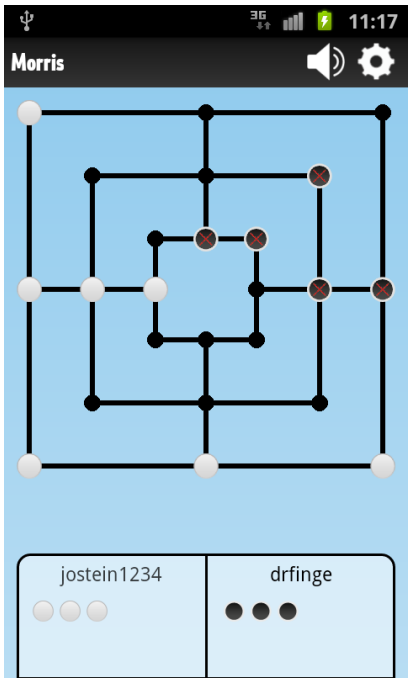
(a) Green indicator shows where you can place a piece



(b) Highlights selectable pieces



(c) Highlights selected piece, green indicator on possible moves



(d) Highlights removable pieces with a red cross

3.5.3 Hotseat mode

If you start a local game as described in section 3.5.1, you can control both players from the same device.

3.5.4 Online mode

If you start an online game as described in section 3.5.1, you are taken to the board screen, and need to wait for another player to join your game. The guest, i.e. the one who joins the game, will get the initial move. Your own pieces will always be white.

4 Test report

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

4.1 Functional requirements testing

FR1 - Placement of pieces	
Executor	Ole Jørgen Rishoff
Date	12.04.2012
Time used	5 minutes
Evaluation	The players successfully placed all nine pieces.

Table 1: Testing of FR1

FR2 - Moving pieces	
Executor	Ole Jørgen Rishoff
Date	12.04.2012
Time used	3 minutes
Evaluation	The players successfully moved their pieces one length at the time.

Table 2: Testing of FR2

FR3 - Morris state	
Executor	Ole Jørgen Rishoff
Date	12.04.2012
Time used	3 minutes
Evaluation	When placing three pieces in a row, the game successfully changed state, and a piece was removed from the opponent.

Table 3: Testing of FR3

FR4 - Flying pieces	
Executor	Ole Jørgen Rishoff
Date	12.04.2012
Time used	3 minutes
Evaluation	When the player had three pieces left, the game successfully changed state to Flying state, and the player was allowed to move to any vacant field.

Table 4: Testing of FR4

FR5 - Multiplayer	
Executor	Ole Jørgen Rishoff
Date	12.04.2012
Time used	10 minutes
Evaluation	Ole and Emil connected to each other via the Skiller framework, and successfully played a whole game.

Table 5: Testing of FR5

FR6 - Random start	
Executor	Ole Jørgen Rishoff
Date	12.04.2012
Time used	3 minutes
Evaluation	IKKE IMPLEMENTERT OMG.

Table 6: Testing of FR6

FR7 - Game board	
Executor	Ole Jørgen Rishoff
Date	12.04.2012
Time used	1 minute
Evaluation	The game has a board conforming with the layout of *Nine Men's Morris*.

Table 7: Testing of FR7

FR8 - Sound effects	
Executor	Ole Jørgen Rishoff
Date	12.04.2012
Time used	10 minutes
Evaluation	IKKE IMPLEMENTERT OMG.

Table 8: Testing of FR8

FR9 - Setting player name	
Executor	Ole Jørgen Rishoff
Date	12.04.2012
Time used	5 minutes
Evaluation	A player can set his own name when creating a Skiller account.

Table 9: Testing of FR9

FR10 -Highlighting possible moves	
Executor	Ole Jørgen Rishoff
Date	12.04.2012
Time used	10 minutes
Evaluation	Possible moves are highlighted in all states, as long as it is your turn.

Table 10: Testing of FR10

FR11 - Game over	
Executor	Ole Jørgen Rishoff
Date	12.04.2012
Time used	5 minutes
Evaluation	When a player has only two pieces left, or cannot move any of his or her pieces, the game successfully ends.

Table 11: Testing of FR11

4.2 Quality requirements testing

FR11 - Game over	
Executor	Ole Jørgen Rishoff
Date	23.04.2012
Stimuli	Addition of a new game variant
Expected response	The architecture should allow an easy extension to *Twelve Men's Morris*.
Observed response	The system is flexible and an extension can easily be added.
Evaluation	Successful

Table 12: Testing of M1

5 Relations to 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.

6 Issues

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.

6.1 Gained experieces