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

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. Architectural Description

2.1. Architectural Drivers/Architectural Significant Requirements (ASRs)

The main drivers (functional, quality and business requirements) that affect architecture and the system mostly.

ASRs consists of all the requirements that affects the architectural design in any way. Both functional and nonfunctional requirements can be ASRs as long as it has some impact on the design.

Functional

- The game will have a start menu.

- The game will have a pause function.

- The game will have a tutorial function.

- The device have to support touch functions.

Nonfunctional

- The game have to run on android.

- Logic will be written in java.

2.1.1. Quality Attributes

Modifiability

- Focus on modifiability is required.

- Use abstract classes where it compliments the code.

- Use mvc design as long as it will improve the modifiability.

Usability

- Use the lowest possible api so that the game can be played on as many devices as possible.

- Focus of usability for the end-user is required.

Others

- The game has to be a multiplayer game.

2.2. Stakeholders and concerns

The stakeholders in this project are identified as developers and end user

2.2.1. End user

The end users will be the players of the game. Their concerns will be to understand the game and to enjoy playing it.

2.2.2. Developers

The developers will be group A13. Their concerns will be to finish the game, make it work as planned. Test the code regularly. The code should be easy to understand and easy to modify.

2.3. Selection of Architectural Views (Viewpoint)
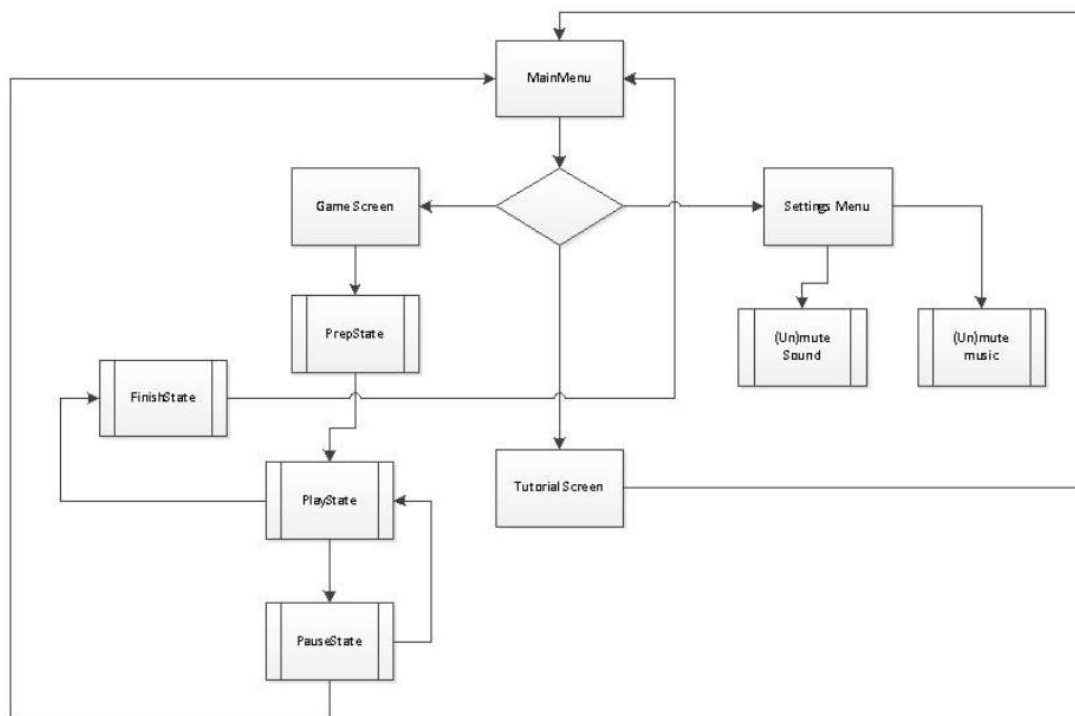
2.4. Architectural Tactics

One of our quality requirements goes directly at reducing coupling so, the tactics we plan on useing are the following; encapsulate, restrict dependencies, refactor and use abstract

common services. These tactics work to increase the modifiability which is one of our
QA.

To ensure that the usability does not fall behind we plan on giving the player the
option to cancel a game and give the player the option to pause.

2.5. Architectural and Design Patterns

When it comes to design patterns we plan on using mvc to increase the modifiability of
our game.

2.6. Views

Document architecture using your selected views (at least logical, process and develop-
ment) through diagrams and supporting/explaining text.
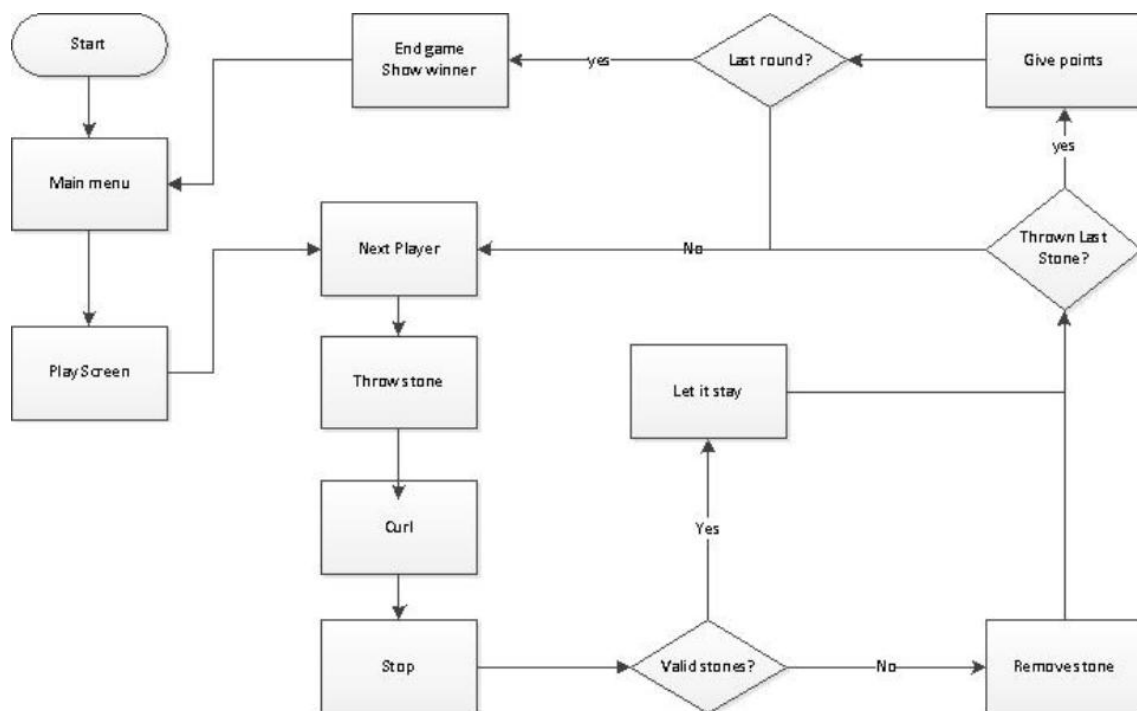
2.7. Consistency Among Views

Describe inconsistencies among the views in your architecture.

2.8. Architectural Rationale

Here you will explain why you have chosen the architecture you have chosen. This part should motivate for why the architecture will work and fulfill the quality, business and the functional requirements.

2.9. Issues

2.10. Changes

References

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

[2] **Article**
Phillipe B. Kruchten. "The 4+1 View Model of architecture", IEEE Software Magazine, Volume 12, Issue 6, 1995.