Requirements and Analysis Document for 20

Version:
ver 1.0 – All participants
This version overrides all previous versions.

1. Introduction
This section gives a brief overview of the project.

1.1 Purpose of application
Purely for entertainment purposes.

1.2 General characteristics of application
Turn-based board game. Multiplayer over network with 1-8 players.
Possible to play offline on a single computer against persons in the room as well.

1.3 Scope of application
A complete digital representation of the original board game RoboRally™ by Avalon Hill Games. To be played in multiplayer over network.

1.4 Objectives and success criteria of the project
Playable with all necessary functions for adequate gameplay. Multiplayer offline is a must. Online if possible within time frame.

1.5 Definitions, acronyms and abbreviations
- GUI, the graphical interface i.e all the buttons and look of the game.
- Java, platform independent programming language.
- JRE, the Java Runtime Environment. Additional software needed to run a Java application.
- IntelliJ IDEA, Integrated Development Environment. An advanced text editor for writing and producing Java code with various additions.
- Host, a computer where the game will run.
- Card, a card which a specific robot movement is written on.
- Turn, the turn for each player. During a turn a player may choose cards to program his/her robot.
- Round, a round consists of

2. Requirements

In this section we specify all requirements

2.1 Functional requirements

1. Start new game
2. Choose how many players
3. Input name for player
4. Choose map
5. Player should have access to rulebook
6. Click ready
7. Start Round
8. Shuffle cards
9. Deal cards
10. Pick registercards
11. Choose whether to power down or not
12. Do five turns
13. Move robots (according to register each turn)
14. Robots take damage and die
15. Robots can fall off edge
16. Robots can spawn on last checkpoint
17. Robots can be moved on Conveyors
18. Players can use Robot-upgrades
19. Cards gets shuffled between each round
20. Players gets new cards according to how many damagetokens
21. Players registers can become locked if to much damage is taken
22. Players lose lifetokens and eventually dies
23. Player can win on last checkpoint or when everyone else is dead
24. Player can withdraw and end game
25. Player can restart game

2.2 Non-functional requirements

NA

2.2.1 Usability

Simple, clean and understandable user interface is our goal. Anyone who has played a boardgame or computergame should be able to play and understand the game with help from rulebook.

2.2.2 Reliability

NA

2.2.3 Performance
NA

2.2.4 Supportability
Desktop application. Our goal is to have Multiplayer support.

2.2.5 Implementation
The application will be runnable and adapted for Desktop/Laptop, OS X, Window osv..

2.2.6 Packaging and installation

2.2.7 Legal
We don’t own the rights to the original game so we cannot publish it.

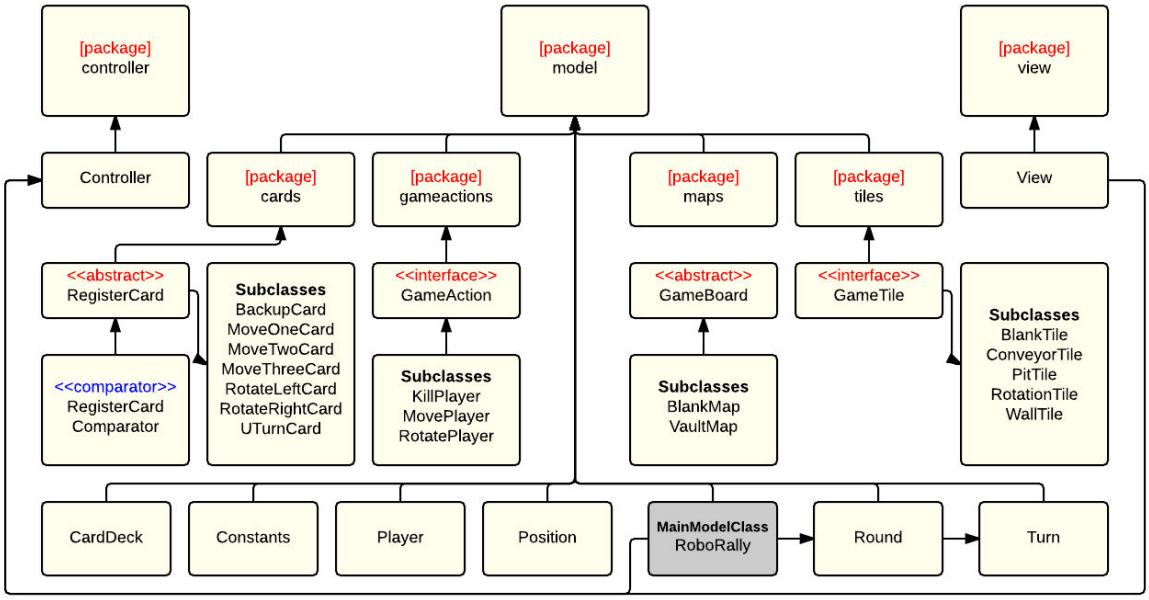
2.3 Application models

2.3.1 Use case model

2.3.2 Use cases priority

newGame	High
startRound	High
doTurn	High
usePerk	Low

2.3.3 Domain model



2.3.4 User interface

The GUI will be similar to the original analog board game. Should give the feeling that you're sitting by a table and playing the game.

2.4 References

APPENDIX

GUI

Domain model

Use case texts