System design document for ”Chalmers Risk”

Version: 2.00

Date: 20/5 - 2015

Author: Malin Thelin, Oskar Rutqvist, Björn Bergqvist, Robin Janssson

This version overrides all previous versions.

1 Introduction

1.1 Design goals

To support testing the software must support the standards of object oriented programming and it must be possible to test isolated part of the program.

1.2 Definitions, acronyms and abbreviations

- GUI,
- Java,
- JRE,
- MVC,
- Territory,
- Continent,
- Troop,
- Turn,

2 System design

The application will follow the MVC design pattern.

2.1 Overview

In this section we explain the overall design choices.

2.2 Software decomposition

2.2.1 General

Package diagram. For each package an UML class diagram in
appendix

2.2.X Rules
Three phases of a turn: deployment, attack and move. If playing with cards a draw card phase is going to happen prior to the deployment phase.

Deployment - Count the number of territories and continents to see how many troops a player gets. Then the player get to choose which territory he or she wants to place said troops in.

Attack - A player selects a territory and an amount of troops, not greater than one less than the current troops receding in said territory, to attack with and then moves them to a territory owned by another player. The

Move - A player selects a territory and an amount of troops, not greater than one less than the current troops receding in said territory, the player can then select a second territory he controls that are connected to the first one in order to move the selected troops there.

2.2.2 Decomposition into subsystems

2.2.3 Layering

2.2.4 Dependency analysis

2.3 Concurrency issues

2.4 Persistent data management

2.5 Access control and security

2.6 Boundary conditions

3 References

APPENDIX