Application Requirements Specification

For

MyChessJ, a Java-based Playable Xiangqi Game

Prepared By:

Runlin Liu

INFO 5100, Section 8

# Purpose and Scope Statement

The purpose of MyChessJ is to create an complete playable application based on Java. The basic function of the game includes playing against an opponent (2 players, or control the chess board alternately), recording the progress and the result of each game.

# Requirements Narrative

1. Start a single-player game

In single player game mode, the user will be allowed to randomly generate a random distribution of chesses and play until the game ends, or read a certain game from a formatted document.

1. Start a dual-player game

The game will allow the user to match with an another player, play the game following the standard rule. The progress and the result will be recorded after the game.

1. Inquire game history

Users will be able to get their historical record of games.

1. In-game controls

During the game, users are able to do what is allowed in standard Xiangqi game, like moving or conquering.

1. Save current game

Users are allowed to save their game before they quit the application.

1. Retrieve a game and continue

Users are allowed to save a game in the database and continue it whenever they want.

# Objectives

1. Start a standard dual-player game

A gamer (Gamer 1) starts the application, select the “Dual Player Game”. The software will put the Gamer 1 in the waiting queue until another game(Gamer 2）enter the game. When both players are prepared, the application will generate an initial layout of chesses. Players make their moves alternately until the game ends. The progress will be displayed on the screen, and for each important move taken, a notification will appear on the screed too.

When the game ends, players are sent back to the initial page and are able to start another game.

1. Inquiry of game history

When no game is operating, users are allowed to check their game history in the database.

1. Retrieve a game and continue

When players feel like they are about to quit the game, they are offered with an option to save their current game. When no game is operating, users can select a game from the database and continue it as they want.

1. Generate a random game in single-player mode

When a player starts a single-player game, the application will generate an initial layout of chessboard. When no move is conducted, player can generate a random layout of chessboard or random challenge from a set of famous layouts. The player will need to try their best to win the game.

# Functional Specification

The panorama view of all Java classes used in the application(MVC aspect of UML graph, see Appendix 1)

The relationship between Java classes(Logic UML graph, see Appendix 2)

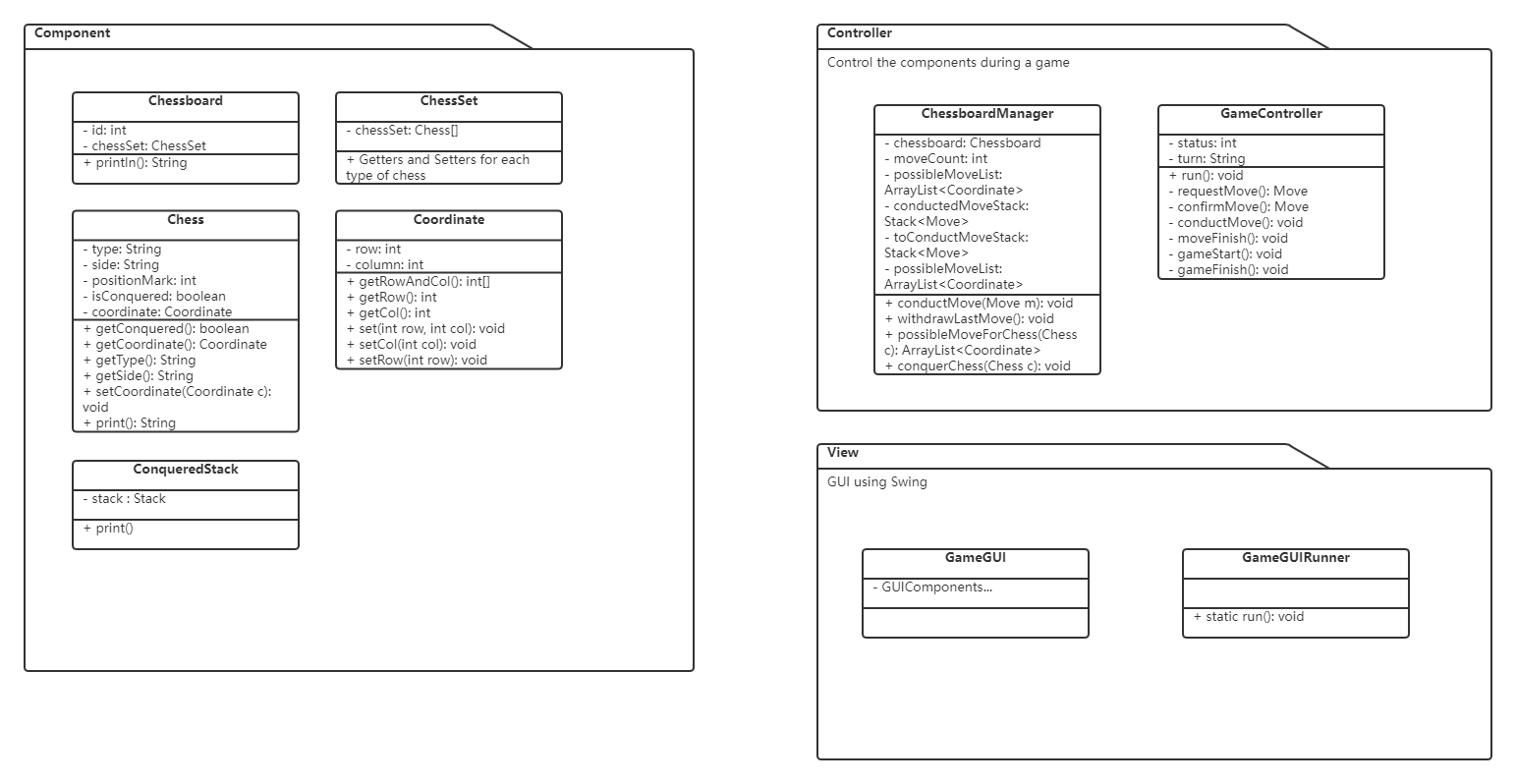
# Logic Specification

The General Flow of Application Operation(See Appendix 3)

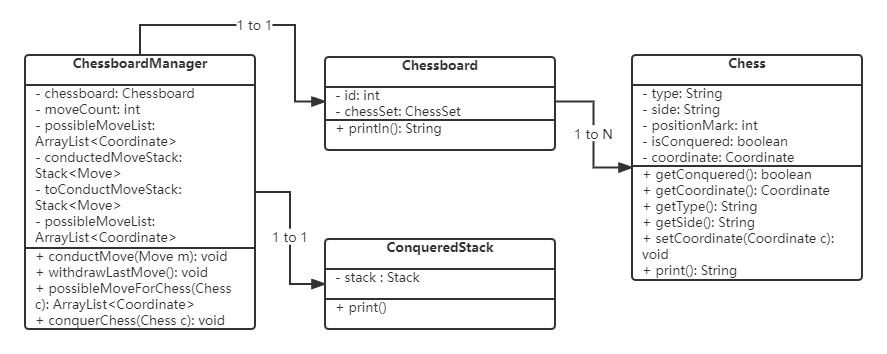
How a game conducted(See Appendix 4)

# Appendix

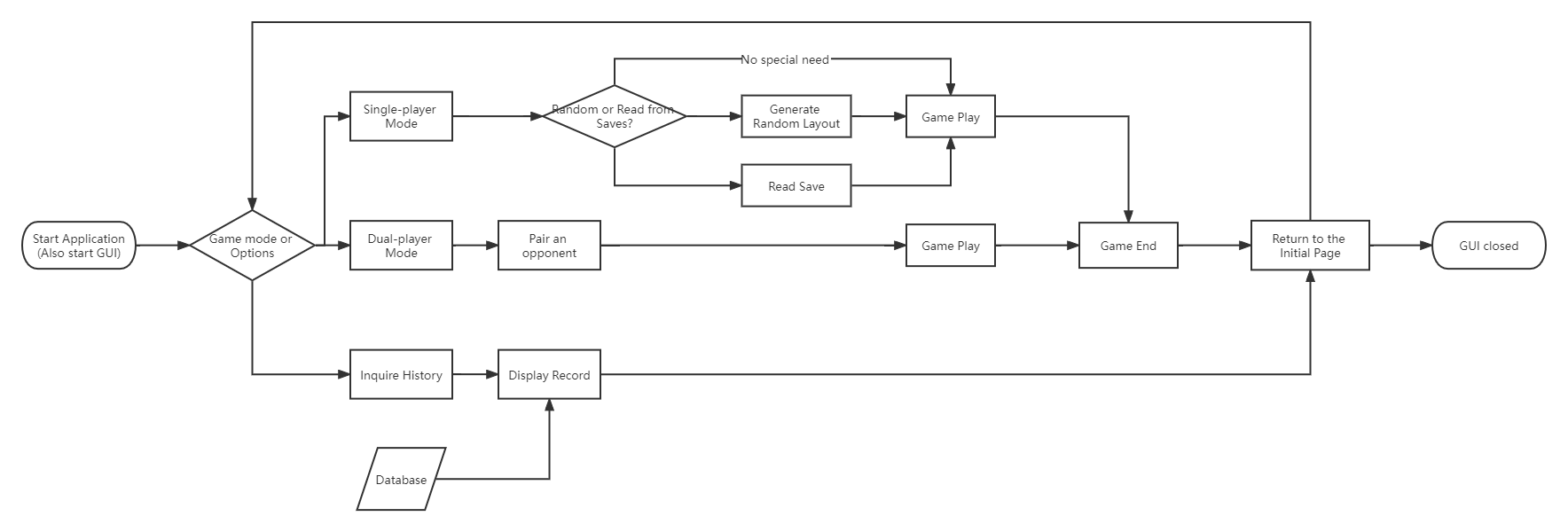
1. **All class used in the application**



1. **UML Design**



1. **Application Flow**



1. **Game Flow**

