Group 8

Author: HeZi’an

Software Requirements

HuaRongDao

Table of Contents

[System Objective 2](#_Toc44468429)

[Domain Analysis 2](#_Toc44468430)

[System Architecture 4](#_Toc44468431)

[Use Cases 4](#_Toc44468432)

[Software Requirements 4](#_Toc44468433)

[R1: GameUI 4](#_Toc44468434)

[R2: GameProcess 5](#_Toc44468435)

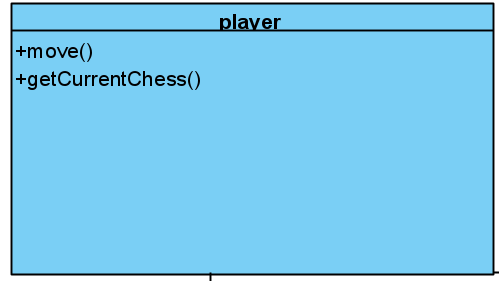
[R3: ChessDB 5](#_Toc44468436)

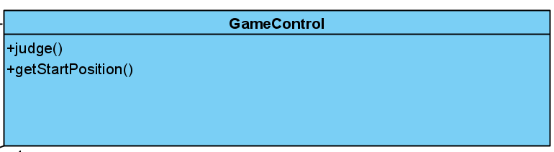
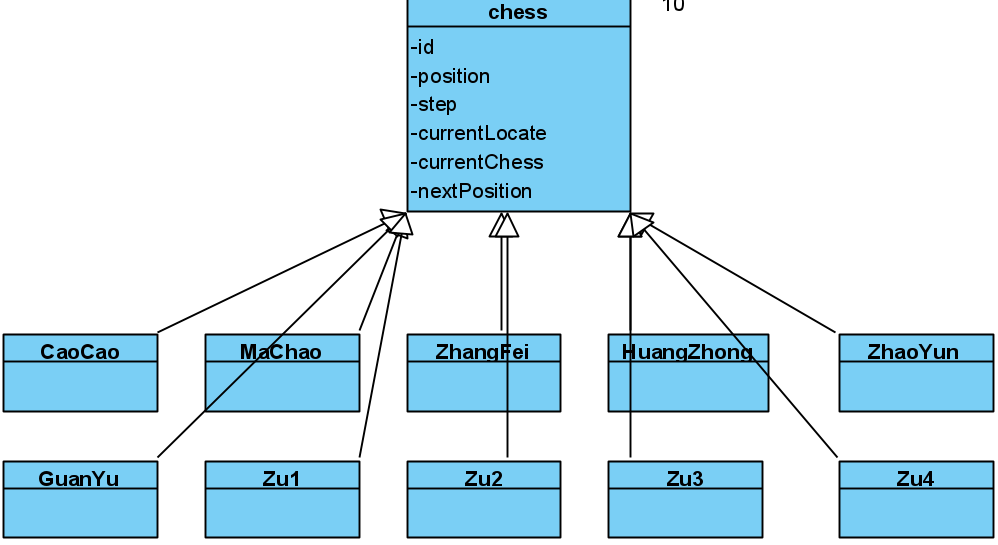
[R4: CheckSolution 5](#_Toc44468437)

## System Objective

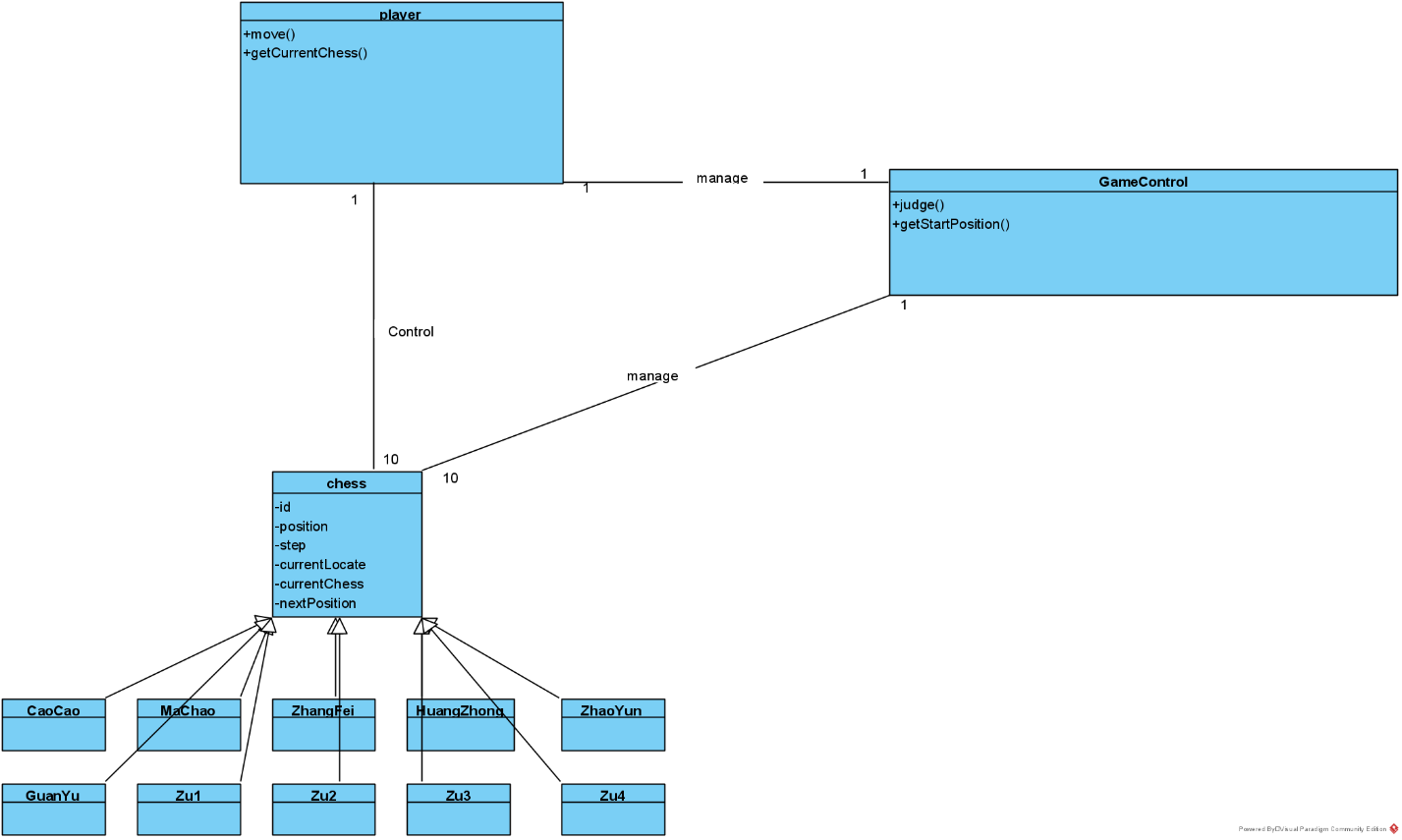
In this project, we are developing a game called Huarong Road. The main gameplay is that the player moves the position of each piece on the board to make the designated piece (Cao Cao) leave the board through the exit.

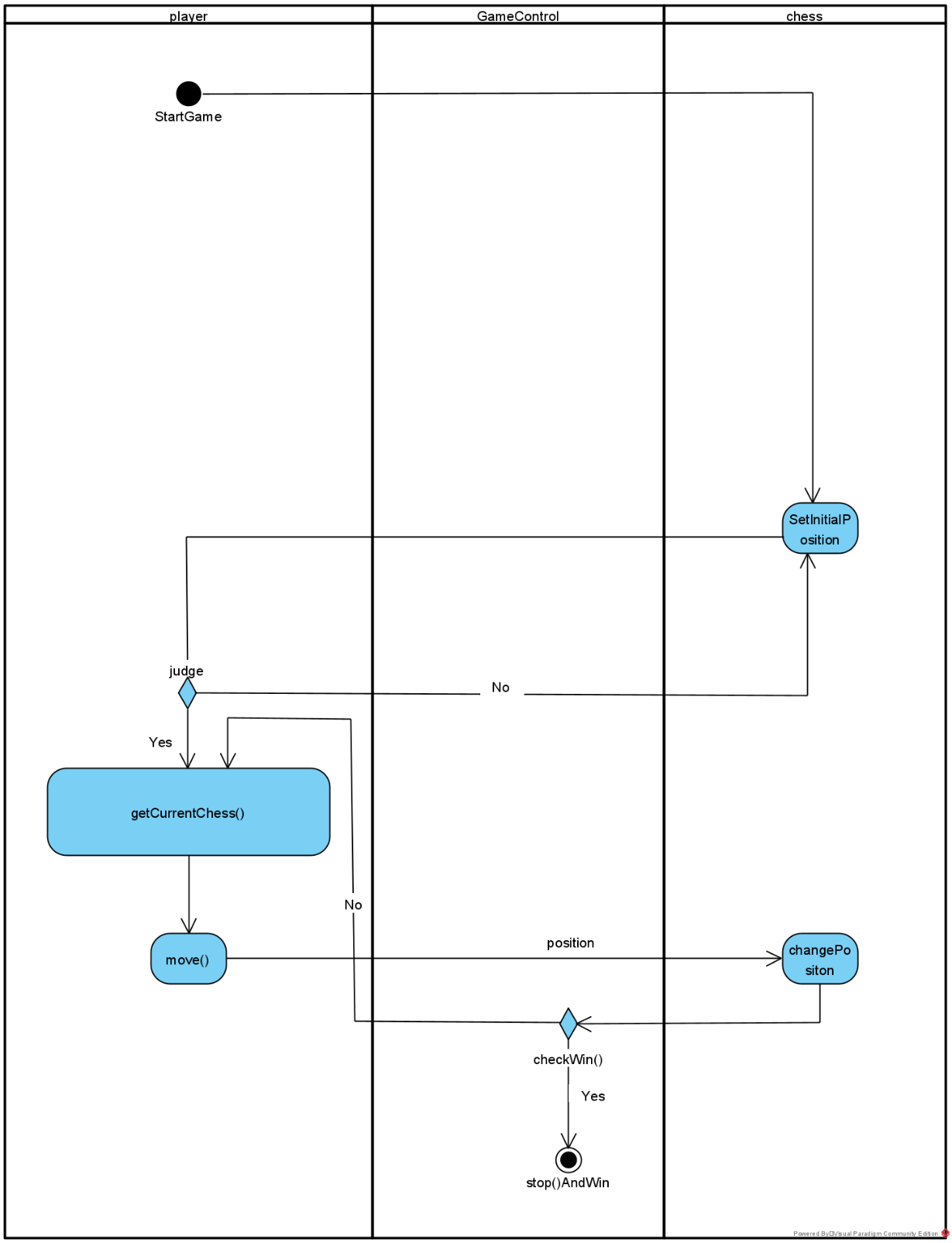
## Domain Analysis

 The components of this game can be categorized into Player, GameControl and Chess.

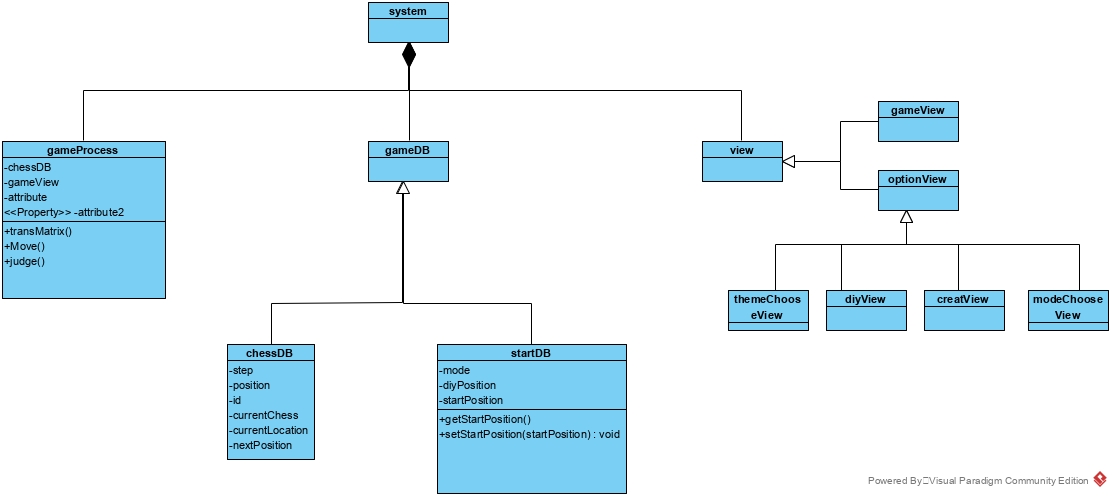


The relationships among different components are shown as follows:



Here is the sequence of the game:

## System Architecture

From the information above, we will design a software system shown as below:

## Use Cases

手机屏幕的截图

描述已自动生成

The system can achieve the following use cases from the player’s perspectives:

## Software Requirements

### R1: GameUI

* R1.1: The chess on board UI can be chosen by player by clicking.
* R1.2: The player can choose the direction to move. (By select button or through keyboard)
* R1.3: Resume
* R1.4: Change model
* R1.5: DIY Opening
* R1.6: Change Theme

### R2: GameProcess

* R2.1: Judge if the next step is illegal
* R2.2: Check if the game is over

### R3: ChessDB

* R3.1: Can solve the position of each chess on board.
* R3.2: The position of each chess on board can be read by system for displaying and processing.

### R4: CheckSolution

* R4.1: Check if there is a solution