Group 5

Author: Gong Jiali

Software Requirements

Huarong Road

Table of Contents

[System Objective 2](#_Toc38898524)

[Domain Analysis 2](#_Toc38898525)

[System Architecture 4](#_Toc38898526)

[Use Cases 5](#_Toc38898527)

[Software Requirements 5](#_Toc38898528)

[R1: UI 5](#_Toc38898529)

[R2: Player 6](#_Toc38898530)

[R3: GameDB 6](#_Toc38898531)

[R4: Chessman 6](#_Toc38898532)

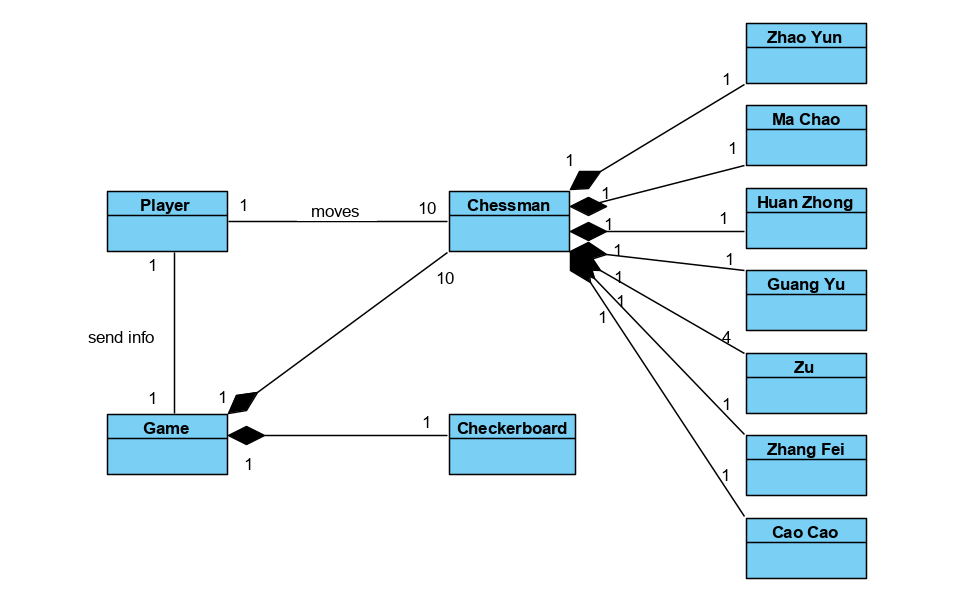
## System Objective

In this project, we are developing a software to implement a game called Huarong Road. This game is fun and can develop one’s intelligence and logic skills as it needs the player to focus on solving the problem for each step and look forward.

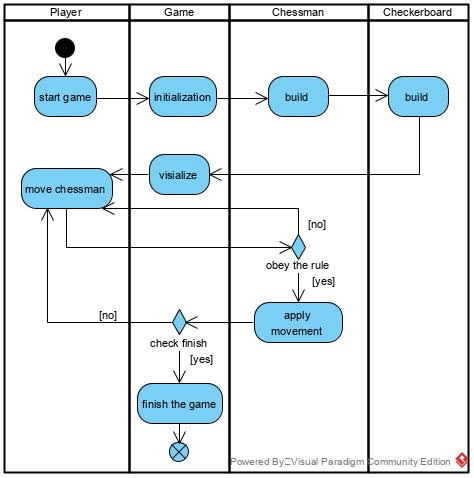
## Domain Analysis

In the game, all we need to consider are player, chessman and checkerboard.

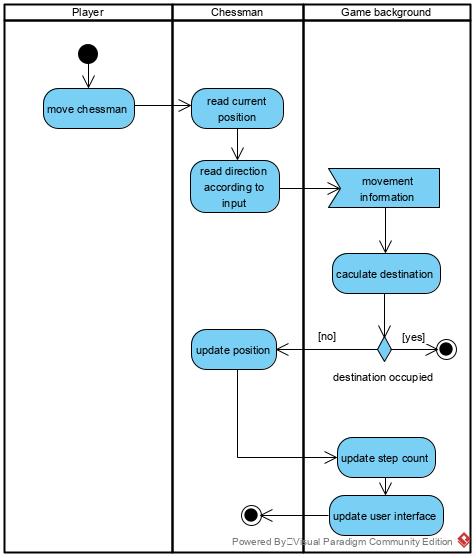
There are 7 kinds of chessmen in their own places of different size: 曹操(2\*2)、张飞(2\*1)、赵云(2\*1)、黄忠(2\*1)、关羽(1\*2)、马超(2\*1)、4 \* 卒(1\*1).



Here is the general logic to implement the whole game:

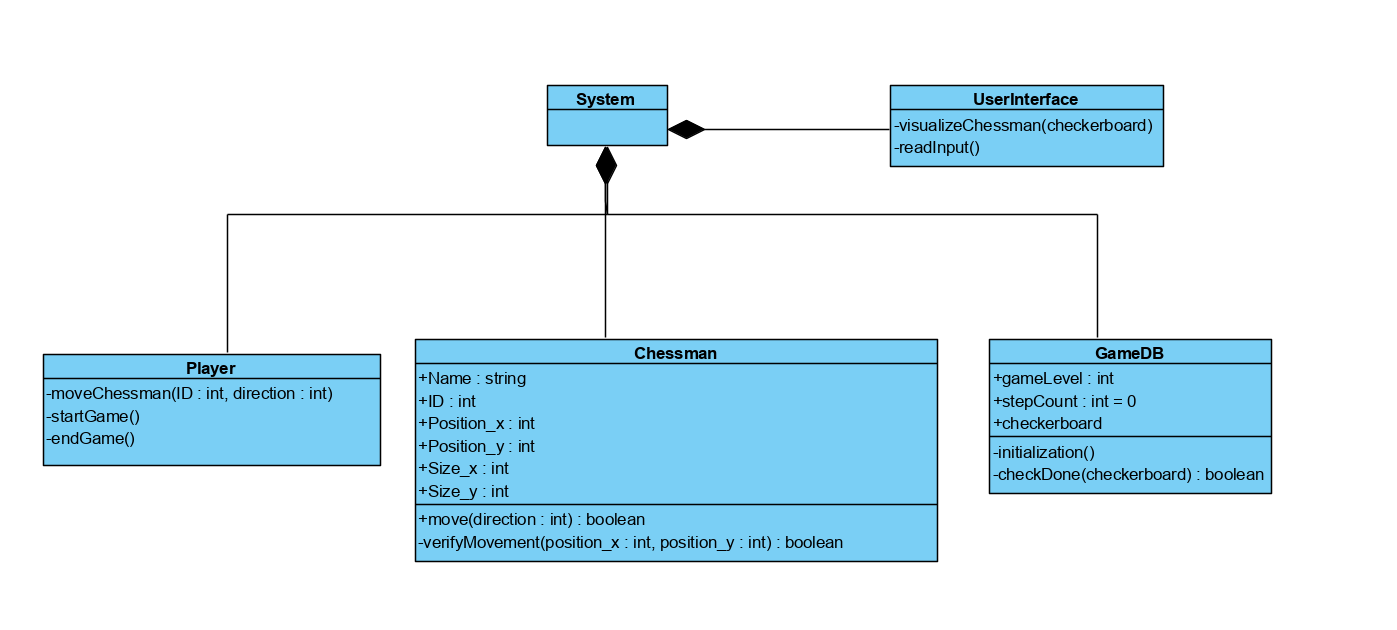


During the whole game, the most complicated step is to move a chessman and verify if it obeys the rule. So here is the detailed diagram for moving a chessman.



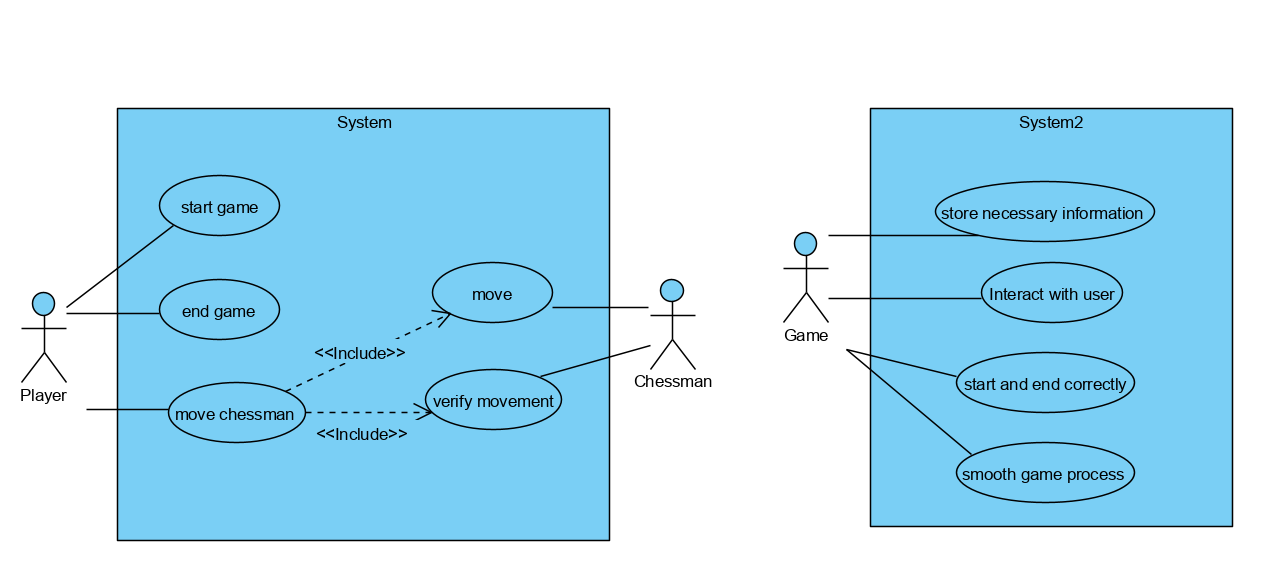
## System Architecture

From the information above, we will design a software system that allows the player to play the game according to game rules. There will be a UI window and background data support. The system architecture is shown below:



## Use Cases

The system can achieve the following use cases from each one’s perspective:



## Software Requirements

### R1: UI

* R1.1: Read input through mouse correctly and in time.
* R1.2: Output the checkerboard with chessmen.

### R2: Player

* R2.1
  + R2.1.1 Player can start a game at any time.
  + R2.1.2 Player can end a game at any time.
* R2.2 Player can move the chessmen If the rules are met.

### R3: GameDB

* R3.1 Correct and solvable initialization.
* R3.2 Correct data.
* R3.3 End when finish: Cao Cao needs to reach the exit which is at the bottom center of the checkerboard.

### R4: Chessman

* R4.1 Move
  + R4.1.1 All the parts of a chessman moves together.
  + R4.1.2 Each step crosses a checker vertically or horizontally, and one step at a time.
* R4.2 Verify movement
  + R4.2.1 User can move chessmen as long as their destination has not been occupied.
  + R4.2.2 If a player is going to move a chessman in violation of rules, do not implement that step.
* R4.3 Correct data.