

A wooden chessboard with alternating light and dark squares is positioned on a highly reflective, dark surface. The board is partially covered by a transparent glass sheet. Dark chess pieces are arranged on the left side of the board, while light-colored pieces are on the right. The reflective surface creates clear, inverted reflections of the board and its pieces. The word "Chess" is overlaid in the center in a white, sans-serif font.

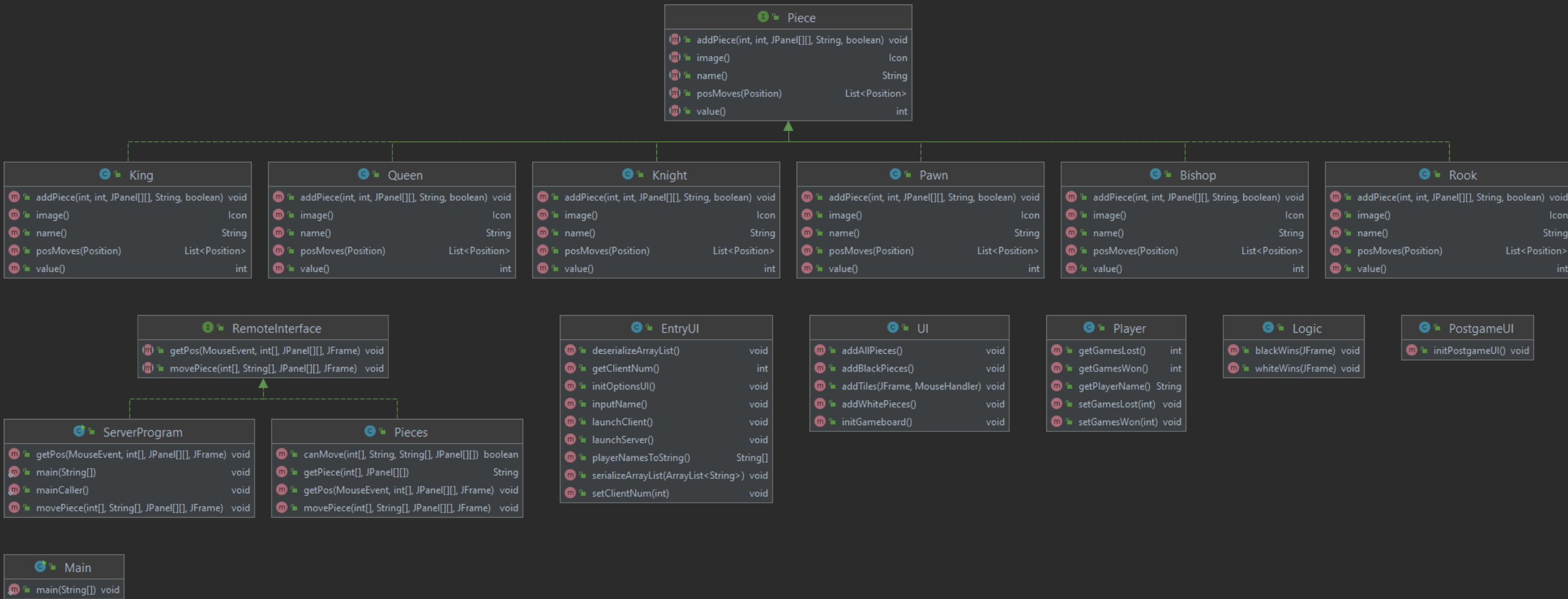
Chess

Lasse Tristan Feldermann-Welkner, Sven Specht

Inhaltsverzeichnis

1. Klassendiagramm
2. EntryUI & PostgameUI
3. UI/Gameboard
4. Piece Interface & Piece Klassen
5. Pieces Klasse
6. RMI Funktionalität

Klassendiagramm



EntryUI, PostgameUI

- EntryUI bietet Möglichkeiten zum starten des Servers und einzelner Clienten
- Beim Start eines Clienten öffnet sich ein Pop-Up Fenster welches die Namenseingabe verwaltet
- PostgameUI öffnet sich wenn ein Spiel beendet ist

UI/Gameboard

- UI Klasse stellt das Gameboard als 8x8 Array dar
- Einzelne Felder werden in der initGameboard()-Methode befüllt
- InitGameboard()-Methode setzt sich aus den Methoden addWhitePieces und addBlackPieces zusammen

Piece Interface und Piece Klassen

- Definiert die `addPiece()`-Methode vor welche von den einzelnen Klassen implementiert wird
- Unterschiede der Implementierung lediglich im Filepath der entsprechenden Bilder

Pieces-Class

- Managed das Bewegen der Figuren auf dem Spielfeld
- Besteht aus Vier Methoden
 1. `getPos()`
 2. `getPiece()`
 3. `canMove()`
 4. `movePiece()`

Pieces-Class (getPos()-/getPiece()-Methode)

GetPos()-Methode:

- Speichert die ersten und zweiten Koordinaten in ein Array
- Fragt ab welche Farbe am Zug ist
- Leitet an getPiece() und canMove() weiter

GetPiece()-Methode:

- Holt sich den geklickten Figurentyp

Pieces-Class (canMove()-Methode)

- Prüft im Switch/case ob der gewollte Zug für die Figur möglich ist
- Im fälle "Pawn" wird gecheckt ob dieser schon einmal bewegt wurde
- Geregelt durch mehrere if-Abfragen
- Gibt einen boolean "allowed" zurück

Pieces-Class (movePiece()-Methode)

- Checkt ob eine Figur geschlagen wird und entfernt diese
- Prüft außerdem ob der gegnerische König geschlagen wurde
- Leitet dafür an die Logic-Class weiter
- Bewegt die Figur an die gewünschte Position

RMI Funktionalität

- ServerProgramm & Pieces implementieren die RMI Funktionalität
- ServerProgramm startet per main-Methode den Server
- Server Start wird per mainCaller()-Methode aufgerufen
- Für jeden Clienten wird ein neues Pieces Objekt erstellt
- Pieces Objekt verbindet im Konstruktor zum Server