

# 实验报告

---

## 功能实现

---

### 基本功能实现（全部完成）

1. 游戏界面：至少包含棋盘和棋子。
2. 游戏流程控制：至少包含包括开始、结束、暂停、重新开始。
3. 游戏玩法：至少实现本地双人对战，即两个人轮流在同一个界面上点击落子。
4. 游戏规则：至少实现猜先、落子、胜利条件。
5. 游戏结算：至少实现判断游戏是否结束，以及结算游戏的胜利者是哪一位玩家。

### 提高功能

1. 实现人机对战，使玩家能够自娱自乐。
2. 实现简易的聊天功能，更加贴近现实。
3. 输出了对局落子的记录
4. 实现了简单的登录界面，可以把输入的ID显示到实际游戏里。

## 游戏操作方式

---

在运行程序后可以在初始界面选择PVP（人人对战）和PVE（人机对战）两种模式

选择模式后进入登录界面，输入ID后就可以在游戏中显示ID

### PVP

选择**PVP**人人对战模式后

点击**NEW GAME**后会自动选边，弹出提示框显示选边结果，而且还会在玩家头像、昵称下方显示黑白色条来明确选边结果。点击确认后直接开始游戏。

点击**GIVE UP**可以认输，如果游戏还没开始，弹出错误提示显示认输失败，如果游戏正在进行且轮到某一方执子，则弹出该方认输的弹窗，并且结束比赛。

点击**UNDO**可以悔棋，如果游戏还没开始，弹出错误提示显示悔棋失败，如果游戏正在进行且轮到某一方执子，则可以直接悔棋。

点击**TIE**可以求和，如果游戏还没开始，弹出错误提示显示求和失败，如果游戏正在进行且轮到某一方执子，则弹出该方求和的弹窗，并且结束比赛。

点击**PAUSE**可以暂停，弹出暂停的弹窗，不结束比赛。

点击**LOGOUT**可以登出账户，界面跳转到登录前的界面。

点击**EXIT**可以退出游戏，界面关闭。

聊天功能：可以在第二个输入框中输入文字，点击**SEND**发送，还可以用右边的快捷信息发送。

复盘功能：玩家每走一步都会输出玩家：x, y 格式的记录

## PVE

选择PVE人机对战模式后

点击**NEW GAME**后默认玩家为黑棋先手，此时头像变为玩家和机器人，在玩家头像、昵称下方显示黑条，可以直接开始游戏。

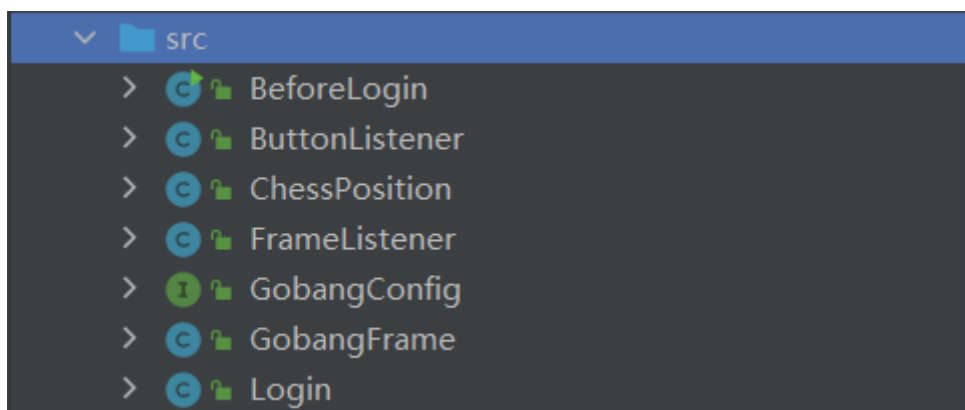
点击**GIVE UP**可以认输，如果游戏还没开始，弹出错误提示显示认输失败，如果游戏正在进行，则弹出玩家认输的弹窗，并且结束比赛。

点击**UNDO**可以悔棋，如果游戏还没开始，弹出错误提示显示悔棋失败，如果游戏正在进行则可以直接悔棋。

点击**TIE**可以求和，如果游戏还没开始，弹出错误提示显示求和失败，如果游戏正在进行，则弹出玩家求和的弹窗，并且结束比赛。

其他同上。

## 架构与设计



源代码里有6个类，1个接口

### BeforeLogin 类

`initUI()` 和 `paint()` 方法主要是实现一个界面。

`main()` 可运行。

可以通过按钮的方式选择PVP和PVE模式，点击按钮后起作用的是 `ButtonListener` 类，里面实现了各类 `ButtonListener` 方法，可以提供响应。PVP/PVE的响应是关闭 `BeforeLogin` 界面，启动 `Login` 界面。

### Login 类

`initUI()` 和 `paint()` 方法主要是实现一个界面。

`menubar()` 实现了一个不实用的菜单。

可以输入用户ID，进入游戏。如果选择的是PVP需要输入两个ID，如果选择的是PVE需要输入一个ID，第二个输入框为不可编辑状态。

## GobangFrame 类

`initUI()` 和 `paint()` 方法主要是实现一个界面。

`charArea()` 方法实现了一个简单的聊天功能。

`PopUp()` 方法实现了一个弹出窗口的功能，主要用于提示玩家。

## ChessPosition 类

`ChessPosition()` 构造函数创建一个位置。

主要配合 `GobangFrame` 类以及其他类中的 `ArrayList<ChessPosition>` 使用。

## ButtonListener 类

`ButtonListener()` 实现多类型构造函数，让所有的按钮能和它关联。

`actionPerformed(ActionEvent e)` 通过判别按钮上的文字进行相应操作。

## FrameListener 类

该类主要实现对下棋操作的响应。

`mouseClicked()` 检测鼠标点击格子下棋，可以判断该处是否已有棋子，如果有棋子了就发出错误提示，如果没有棋子就把该棋子记录到 `ArrayList<ChessPosition>` 并且检测是否五连子

## GobangConfig 接口

这个接口作为一个所有类的公共接口，提供了窗口界面，组件参数等统一的参数，相比于在各个类中自行设置参数，这种使用数据接口的方法可以统一设置参数，方便统一修改和减少重复代码量。

其他说明性内容，不计入文档分，但描述不清楚可能会影响功能点的给分

列举实现的功能和加分项，使助教可以发现并测试你实现的每一个功能

对于较复杂的功能或亮点，描述你的设计与实现，以及对应实现在源代码中的位置

其他任何你想让助教了解的内容