# 《英雄联盟杀 Killers of LOL》项目报告

完成人: 郑策文(32108036)

目录

[《英雄联盟杀 Killers of LOL》项目报告 1](#_Toc154838896)

[1 引言 1](#_Toc154838897)

[2 项目概述 1](#_Toc154838898)

[3 开发环境和工具 2](#_Toc154838899)

[4 游戏设计 2](#_Toc154838900)

[5 编程实现 3](#_Toc154838901)

[6 问题与挑战 8](#_Toc154838902)

[7 测试和调试 9](#_Toc154838903)

[8 结果与展望 16](#_Toc154838904)

## 引言

本报告旨在介绍一款名为“英雄联盟杀”的回合制卡牌游戏，该游戏是基于知名游戏《英雄联盟》而设计的。它将英雄联盟的角色、技能、装备等元素融入到卡牌游戏中，让玩家可以在卡牌游戏中体验英雄联盟的乐趣。通过本报告，我们将深入探讨游戏的规则、玩法、以及设计理念，为玩家提供更全面的了解和体验。

卡牌游戏是一种使用卡片作为主要元素的游戏形式，它可以分为实体卡牌游戏和数字卡牌游戏。卡牌游戏的魅力在于它可以结合不同的主题、规则、策略和随机性，创造出丰富多样的游戏体验。卡牌游戏也可以促进玩家之间的交流和互动，增强玩家的思维能力和创造力, 成为了一种社交娱乐形式，让玩家们可以在游戏中交流、竞技，共同享受游戏带来的乐趣。

## 项目概述

本项目旨在设计一款名为“英雄联盟杀”的回合制卡牌游戏，该游戏是基于知名游戏《英雄联盟》而设计的，它将英雄联盟的元素和卡牌游戏的机制相结合。玩家将扮演游戏中的英雄角色，通过收集、组建卡牌牌组，展开策略性对战，以击败对手并达到游戏胜利的目标，创造出一种新颖的游戏体验。

* 1. 主题和玩法：
* 游戏主题：以《英雄联盟》中的英雄和角色为主题，玩家可以在游戏中选择自己喜爱的英雄，并利用其独特的技能和能力进行对战。
* 玩法：游戏采用回合制卡牌对战模式，玩家通过抽取和使用卡牌来展开战斗，同时需要制定策略来应对对手的行动，消灭敌方玩家以取得胜利。
  1. 设计目标：
* 还原英雄联盟的风格和氛围，让玩家感受到英雄联盟的魅力和乐趣
* 提供多样的英雄选择和卡牌组合，让玩家体验到不同的战术和策略。
* 打造简单易懂的游戏规则和操作方式，让新手玩家能够快速上手，同时保持高深的战术深度，吸引有挑战性的玩家

通过本项目，我希望设计出一款富有创新性和趣味性的卡牌游戏，让玩家能够在游戏中感受到策略对战的乐趣，并为玩家带来全新的游戏体验。

## 开发环境和工具

* 1. 编程语言：

Java

* 1. 开发工具：

IDEA、cursor

* 1. 选择以上工具的原因：
* IDEA是一款集成开发环境，智能的代码编辑和提示，可以帮助你快速编写和修改代码，避免错误和拼写错误。强大的代码分析和重构，可以帮助你优化和改进代码的结构和质量，提高代码的可读性和可维护性。
* Cursor集成了gpt4.0功能，帮我梳理代码结构

## 游戏设计

* 1. 英雄信息

|  |  |  |
| --- | --- | --- |
| 姓名 | 生命值 | 技能 |
| 奥拉夫 | 3 | 狂暴(使用杀无限制次数) |
| 德玛 | 4 | 破釜(出牌阶段可以失去一个生命值摸两张牌) |
| 剑魔 | 4 | 绝境(失去最后一张手牌时可以再摸一张) |
| 卡牌 | 3 | 积累(可以跳过弃牌阶段) |
| 诺手 | 4 | 无情铁手(被伤害时，可以获得伤害你的那张牌) |
| 吸血鬼 | 4 | 置换(受到伤害时可以获得牌堆顶的两张牌) |

卡牌信息

|  |  |  |
| --- | --- | --- |
| 牌名 | 类型 | 详细信息 |
| 多兰剑 | 基础牌 | 对指定的一名敌人造成伤害 |
| 多兰盾 | 基础牌 | 抵挡敌方的伤害 |
| 血瓶 | 基础牌 | 回复一滴体力值 |
| 羊刀 | 武器牌 | 装备后，出牌阶段可以无限出多兰剑 |
| 水晶箭 | 锦囊牌 | 除你之外的所有人出一张多兰盾，否则受到伤害 |
| 怨夺 | 锦囊牌 | 充满怨恨，掠夺两张手牌 |
| 救赎 | 锦囊牌 | 对棋盘内所有角色使用，体力值加一 |

* 1. 回合操作

玩家轮流进行回合操作，游戏开始时每人获得4张手牌，回合开始玩家先摸两张手牌，进入出牌阶段，有30s倒计时，也可主动提前结束出牌阶段；后面进行弃牌阶段，有15s倒计时，也可主动结束弃牌阶段。至此，该玩家的回合结束。

* 1. 胜利条件

玩家需要通过策略和战斗来降低对手的体力值，对手体力值为零，获得胜利。

## 编程实现

* 1. 代码结构



* 1. 卡牌的数据结构和存储方式

牌堆和弃牌堆卡牌的数据结构是一个ArrayList类型的cardPile变量，该变量存储了Card类的实例对象。

* 1. 游戏逻辑的实现

玩家回合的逻辑：

1. 如果选择的角色技能类型是突破限制，则发动技能。

2. 如果装备的武器牌的效果类型是解除限制，则发动武器牌的效果。

3. 如果当前状态为被击，则进行被击状态的处理，包括显示提示信息、判断是否发动多兰盾、更新图形等。

4. 如果当前玩家的手牌中没有多兰盾，则将所有非多兰盾牌的点击属性设置为不可点击，并且判断是否发动角色的受伤技能。

5. 在出牌阶段，如果手牌数量大于当前生命值，则进行丢牌处理，并显示相应的提示信息。

6. 设置当前玩家不是我的回合，并刷新当前玩家的UI。

7. 判断游戏是否结束，如果结束则跳出循环。

敌方回合逻辑：

在敌方回合中，首先判断敌方玩家是否可以使用技能"Push the Limits"，如果可以使用，则执行该技能的效果。然后，检查敌方玩家的装备牌是否可以使用，如果可以使用，则执行装备牌的效果。接下来，判断敌方玩家是否处于被伤害状态，如果是，则执行出多兰盾的环节；否则，正常回合则给敌方玩家发牌。发牌后，执行"computer\_dealTreatCard"方法，该方法是用于处理电脑角色的出牌逻辑。然后，判断电脑角色是否需要出多兰剑，如果需要，则执行多兰剑的逻辑。最后，判断电脑角色是否造成了伤害，如果造成了伤害，则敌方回合结束，我方玩家进入回合；否则，继续执行敌方玩家的下一步操作。如果我方玩家获胜，则跳出循环。

* 1. 用户界面的设计和交互
* InitGame() initView()，该类为一幅卡牌游戏的视图类。该方法用于初始化游戏的界面，包括创建组件和设置组件的位置等。其中，还创建了两个集合来存储玩家准备出的牌，并创建了一个倒计时器来倒计时10秒。最后，还创建了一些按钮和文本框，用于玩家进行出牌、结束、弃牌等操作，并对按钮添加了监听器。
* initJframe()对一个JFrame窗口进行初始化和布局。首先，设置窗口的标题、大小和关闭模式等属性。然后，设置窗口的一些布局属性，如设置窗口中不可调整大小。接下来，通过JLabel添加了一些背景图片和图标，并设置了它们的位置和大小。然后，通过JPanel创建了一个聊天窗口，并设置了它的背景颜色和大小，并将它添加到容器中。在聊天窗口中，添加了一个不可编辑的JTextArea用于显示聊天记录，一个JScrollPane组件用于滚动显示聊天记录，一个JTextField用于输入聊天内容，一个JButton用于发送聊天内容。最后，创建了一个记录窗口，用于显示对局记录。整体来说，这段代码主要是对窗口和其中组件的布局和设置，以及一些事件监听器的添加。
* selectRole()，用于选择英雄的角色。方法首先设置窗口的标题、大小、关闭模式等属性，然后在窗口中添加背景图片和标签等组件。同时，还添加了两个按钮：backspace和sure，并为它们添加了事件监听器。方法还使用循环在窗口中添加了多个角色的图片，并为每个图片添加了鼠标点击事件监听器。最后，根据选择的角色数量，动态地在窗口中添加或删除相应的组件。

|  |  |
| --- | --- |
| 英雄选择界面 |  |
| 选好后可进入游戏（也可以撤销你的选择） |  |
| 游戏界面 |  |

* 1. 关键算法
* computer\_dealBeHurt()方法用于处理电脑角色受到伤害的逻辑，它首先找到电脑代理的对应玩家、手牌和生命值，然后检查电脑手牌中是否有多兰盾卡，如果没有则扣除电脑角色的生命值，并在界面上显示扣血效果，然后更新生命槽并记录扣血情况，最后将电脑角色的状态更改为无。如果电脑手牌中有多兰盾卡，则出多兰盾并显示相应信息。
* computerDealHurtCard()方法用于处理电脑角色遭受伤害卡的效果，它首先找到玩家对应的伤害卡，然后在界面上显示时间，并等待一段时间，接着根据伤害卡的类型和玩家是否可被攻击来执行相应的逻辑，并在界面上显示相应信息。
* computer\_dealTreatCard()方法用于处理电脑角色使用治疗卡的逻辑，它首先找到电脑角色可使用的治疗卡，然后根据需要治疗的血量和可用治疗卡的数量来确定使用治疗卡的数量，并依次执行治疗卡的效果，并在界面上显示相应信息。
* ShowCard()的作用是显示玩家的倒计时，并在倒计时结束后执行一些操作。首先，它会将玩家的回合时间组件设置为可见。然后，它会检查是否需要弃牌。如果需要弃牌，则会从玩家的手中移除卡牌并执行相应的操作。接下来，它会将下一个玩家的组件设置为可见，并将一些状态设置为true。最后，它会隐藏玩家的回合时间组件。
* timeWait()的作用是等待倒计时结束，并在计时结束后执行相应的动作。首先，它会根据玩家的回合来处理不同的情况。如果是自己的回合，则会根据倒计时的数值来更新回合时间组件，并在每次更新后暂停1秒。当倒计时为-1时，如果是自己的回合，则会执行弃牌阶段的操作，包括从玩家手中移除卡牌并执行相应的动画效果。最后，它会将玩家的回合时间组件设置为不可见，并重绘游戏界面。
* win()，其作用是判断是否有玩家在游戏中失败。方法通过遍历玩家数组，检查每个玩家的生命值是否为0。如果某个玩家的生命值为0，那么会根据玩家的身份展示不同的胜利消息，并弹出消息对话框。最后，如果没有玩家失败，则方法返回false。
* actionPerformed()，其作用是处理按钮的点击事件。方法通过检查点击的按钮来源来确定是哪个玩家点击了按钮。然后，在控制台上打印出相应玩家手牌的变化。
* deckUpdate方法，其作用是更新牌堆和手牌。方法首先对牌堆中的牌进行随机洗牌，并将洗牌后的牌移动到桌面。然后，将桌面的牌添加到牌堆中，清空手牌。
* initPlayer()方法生成两个角色，并为每个角色创建一个玩家对象，设置其属性和位置，并添加到容器中。
* initCard()方法将卡牌分发给玩家，并将玩家的手牌添加到玩家对象中。最后，将玩家对象添加到玩家列表中。
* actionPerformed()用于处理按钮点击事件。在按钮被点击时，会执行其中的代码。该方法首先判断点击的按钮是否为玩家按钮，如果是，则会根据点击的玩家和牌的类型执行相应的操作，比如出杀、装备牌、锦囊牌等。如果点击的按钮不是玩家按钮，则会执行相应的操作，比如取消、发动技能等。最后，将下一个玩家设置为可玩状态。

## 问题与挑战

* 1. 敌方电脑的操作逻辑问题

用三个方法分别处理电脑角色受到伤害、遭受伤害卡、使用治疗卡的逻辑。进入敌方电脑回合后，先判断是否能使用技能，然后检查装备牌是否可用，然后先执行使用血瓶的方法，（体力值不满且有血瓶卡就会使用血瓶），然后执行多兰剑的方法（有多兰剑卡就会使用），如果对我方造成了伤害回合就会结束；否则，继续执行敌方玩家的下一步操作。如果我方玩家获胜，则跳出循环。

* 1. 界面卡牌移动动画问题

Common类包含多个静态方法，用于处理卡牌的移动、重新摆放、流畅移动、手牌移动、隐藏牌、处理卡牌效果等。其中，move方法用于按照给定的起始点和目标点进行卡牌的直线移动；rePositon方法用于重新摆放卡牌到指定位置，并按照前后顺序重新排序；moveCard方法用于进行卡牌的流畅移动；dealBeHurt方法用于处理卡牌的伤害效果；sendCardTao方法用于发送卡牌到指定玩家，并进行移动和状态修改等操作。

## 测试和调试

* 1. 功能测试

|  |  |
| --- | --- |
| 英雄选择功能正常 |  |
|  |
| 多兰剑功能正常 |  |
| 多兰盾功能正常 |  |
| 弃牌功能正常 |  |
| 电脑出多兰剑功能正常 |  |
| 造成伤害功能正常 |  |
| 血瓶功能正常 |  |
| 回应多兰盾功能正常 |  |
| 触发技能(无情铁手）功能正常 |  |
| 水晶箭功能正常 |  |
| 救赎功能正常 |  |
| 怨夺功能正常 |  |
|  |
| 羊刀功能正常 |  |
|  |

## 结果与展望

较好的实现的大作业要求，完成了基本的卡牌游戏功能和较好的界面和交互，游戏比较丰富，卡牌类型有3种：基础牌（3种），武器牌（1种）和锦囊牌（3种）。英雄有6名，并分别都有各自的技能。

如果以后有机会可以再丰富一下武器牌和锦囊牌的种类与英雄的数量。还可以在对局中再添加1至2名玩家，不再局限于两名玩家的对战。