**团队成员：**  
**201721144005 陈喜莲**

**201721123034 唐川（组长）**

**201721022095 江银彬**

**201721123038 梁才玉**

**201721123010 郭锐倩**

### 需求分析与设计

1. 软件需求分析报告要求

1） 参考《软件需求规格说明书》国标规范文本，撰写对应项目的软件需求规格说明书。

2） 除形式上满足规范文本要求外，整体内容必须围绕项目实质展开，对所要开发的项目确保尽力做到清晰完整准确。

3） 采用分层形式描述，随着“层”的深入，描述的内容细节越具体。

4） 使用一致的图形符号和文字描述内容。

5） 所有的缩写须事先定义。

6） 格式：需要有目录，word排版样式规范美观，图文并茂，通篇文档有一个统一的样式风格，注意字体和行距等。

7） 将自己置于读者的立场——如果对软件项目不熟悉的人员，通过阅读这份文档，能否完全读懂软件要做什么。

8） 访问软件项目的真实用户，确保软件真正体现用户的需求，为软件最终可用奠定基础。

二、参考资料

1. 用户调研方法参考《构建之法》第8章获取用户需求——用户调研 http://www.cnblogs.com/xinz/archive/2013/02/03/2890786.html

http://www.cnblogs.com/xinz/p/3308608.html

1. 分析和设计方法：http://www.cnblogs.com/xinz/p/4525232.html

在线作图工具ProcessOn：https://www.processon.com/

原型设计教程

**https://www.cnblogs.com/SivilTaram/p/9190373.html**

1. **原型工具参考**
   1. 移动应用原型与线框工具-墨刀
   2. 原型设计界的PS -Axure RP，Axure
   3. 网页和移动端的设计sketch
   4. 一款简洁高效的原型图设计工具mockplus
   5. 致力于高保真原型制作工具Justinmind
   6. 一款免费的带有手绘涂鸦风格的原型设计软件balsamiq mockups
   7. 更多选择，请参考：https://www.zhihu.com/question/19592829

三、系统设计

(1)在设计阶段，我们要清楚：软件是怎么解决这些需求的？

(2)一个好的分层式结构，可以使得开发人员的分工更加明确。一旦定义好各层次之间的接口，负责不同逻辑设计的开发人员就可以分散关注，齐头并进。

(3)如何才能最大限度地实现这些需求，这就是架构设计要解决的问题。请给出系统的架构设计

### 练习要求

1. **给出本团队项目的目标用户**

所有微信用户

**二、需求分析报告**

**1.引言**

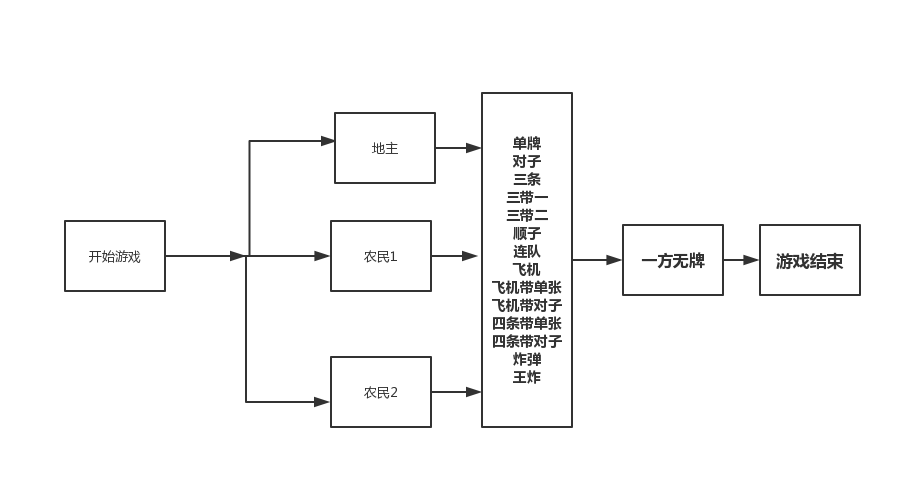
随着科技的进步，计算机已经普及每个家庭，而计算机游戏也随之越来越多，越来越丰富。网络游戏，单击游戏等休闲游戏让众多人喜欢，而斗地主也受广大群众的追捧。

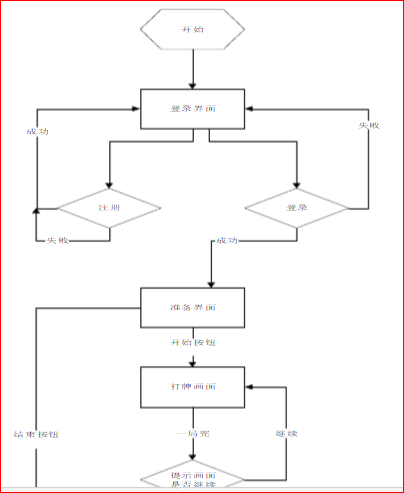
“斗地主”是流行于全国的一种扑克游戏，玩法简单，娱乐性强，老少皆宜。据传在万恶的旧社会，地主横行乡里，无恶不赦，人们为了发泄对地主的痛恨，常常在一天的劳作知乎，一家人关起门来“斗地主”。该游戏由三个人玩，用一副或两副牌，地主为一方，其余两家为一方，双方对战，先出完牌的一方获胜。

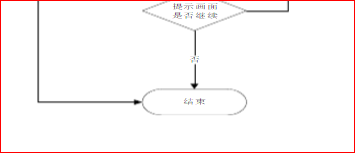
本设计以一副规则为准，本论文讲述了单击版斗地主的游戏算法设计，以java语言编译的一个应用程序。使用者可以简单方便地实现斗地主游戏。本程序界面友好，操作简单，比较实用。

本程序在详细设计时不断地修正和完善，经过测试阶段反复调试和验证，最终形成达到设计要求的可行系统。

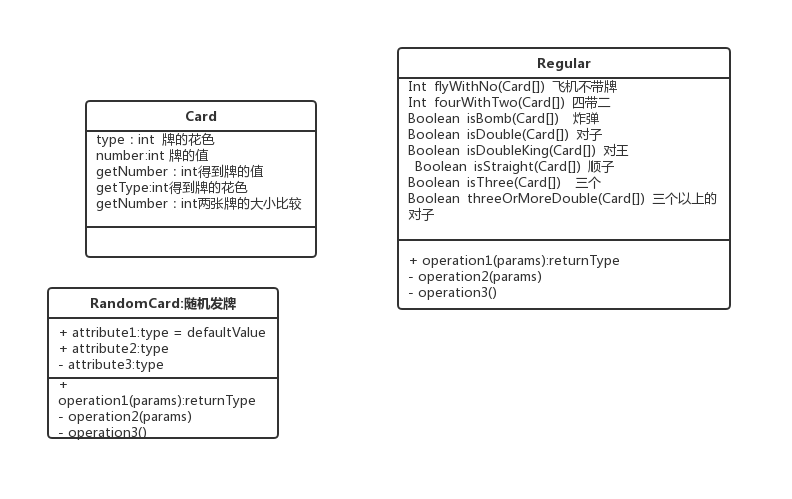
**2.用户场景示意图**

****





**3.类图**

****

1. **界面原型**

**主要界面**

**手机登录界面（包含单机模式、联网模式）**



**联网模式点击登录**



**1、点击匹配模式**



**点击任意一场游戏**



**联网匹配用户中**

**游戏中**



1. **点击好友对战**



**点击邀请好友后，显示在线好友**



**3、点击单机模式不用登录直接进入游戏界面**



**其他界面**

注册界面



获取验证码

l

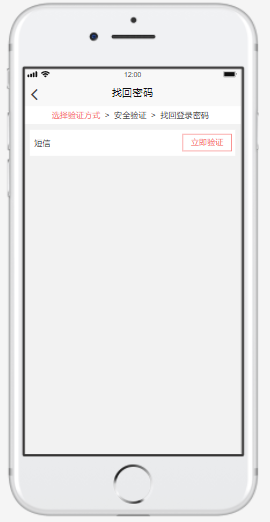


提交验证码



提交密码

忘记密码界面



立即验证



**5.主要功能描述**

1**.生成扑克牌  
2.分发牌给玩家  
3.为玩家从大到小排列牌的顺序**

**4.开始选择加倍机制，炸弹加倍机制和春天加倍机制**

**5.对战机制**

**6.下牌规则介绍**

1. **验收验证标准**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 测试功能 | 测试项 | 输入/操作 | 检验点 | 预期结果 | 验收 |
| 登陆功能 | 输入框 | 单击用户名输入框，输入用户名 | 用户名输入框允许的数据限制为4-10位字符 | 无法输入数字以外的任何字符。若输入框的数字已经10位，则无法再输入 |  |
| 单击密码框，输入密码 | 密码框密码保护 | 输入的字符在密码框短暂显示后或者输入第二个字符时隐藏为“\*” |  |
| 单击密码框，输入大小写字母、数字、下划线以外的字符 | 密码输入框只允许输入的数据为大小写字母、数字、下划线 | 密码框下面出现一行红色字的提示“密码只能含字母、数字、下划线”；若删除非法字符，则该提示消失 |  |
|  | 单击密码框，输入密码 | 密码包含6-16个字符 | 若字符数低于6个，则在输入框下提示“密码应包含6-16个字符” |  |
| 若字符数大于等于6个，则“密码应包含6-16个字符”这个提示消失 |  |
| 若字符数高于16个，则无法继续输入 |  |
| 登陆操作 | 输入不完整（包括空）的用户名和密码，点击登陆 | 用户名和密码检查 | 弹出提示“请输入正确的用户名和密码”，一段时间后消失 |  |
| 输入含大小写字母、数字、下划线以外的字符的密码，点击登陆 | 用户名和密码检查 | 弹出提示“请输入正确的用户名和密码”，一段时间后消失 |  |
| 输入正确的用户名和密码，点击登陆 | 登陆检查 | 若当前网络畅通，则弹出进度框，提示“正在登陆中”，一段时间后提示消失，并跳转到下一个界面 |  |
| 若登陆时网络断开或一分钟后没有登陆成功，则停止登陆，并提示“网络出错，请检查网络设置”，一段时候后消失 |  |
| 输入错误的用户名或密码，点击登陆 | 登陆检查 | 若当前网络畅通，则弹出进度框，提示“正在登陆中”，一段时间后提示消失，并提示“用户名或密码错误” |  |
| 若登陆时网络断开或一分钟后没有登陆成功，则停止登陆，并提示“网络出错，请检查网络设置”，一段时候后消失 |  |
| 忘记密码 | 点击“忘记密码？” | 忘记密码后的处理 | 弹出提示框，提示框正文内容为教学办电话，点击“确定”后提示框消失 |  |
| 游戏类型选择功能 | 单机模式 | 点击“单机模式” | 单机模式在登录界面，无需登录直接进行单机模式 | 跳转至人机游戏页面 |  |
| 匹配模式 | 点击“匹配模式” | 界面响应 | 跳转到等级选择页面 |  |
| 好友对战模式 | 点击“好友对战” | 界面响应 | 跳转到好友对战页面 |  |
| 等级选择页面 | 点击“简单模式” | 界面跳转 | 跳转至简单模式游戏页面 |  |
| 点击“中等模式” | 界面跳转 | 跳转至中等模式游戏页面 |  |
| 点击“困难模式” | 界面跳转 | 跳转至困难模式游戏页面 |  |
| 邀请好友 | 好友在线且未参与对战 | 点击“邀请”按钮 | 邀请处理 | 邀请右侧显示文字“已邀请” |  |
| 好友在线且在对战中 | 点击“预约”按钮 | 预约处理 | 右侧显示文字“已预约” |  |
| 好友离线 |  |  | 显示离线好友 |  |
| 查看并邀请微信好友 | 点击“微信好友”按钮进入微信好友列表，选择好友并点击“确定”按钮 | 选择好友数大于0 | 若选择好友数等于0，弹出提示框，提示框内容为“请选择好友”，点击确定关闭提示框。否则显示文字“成功邀请”，跳转到好友对战界面。 |  |
| 点击“微信好友”按钮进入微信好友列表，选择好友并点击“取消” |  | 返回邀请好友列表 |  |
| 游戏功能 | 随机选择场上一位参与者进行叫牌 | 叫牌 |  | 下一位参与者进行叫牌选择，选择分别为“叫牌”和“不叫” |  |
| 不叫 |  | 下一位参与者进行叫分选择，选择分别为“1分”，“2分”和“3分” |
| 叫分 | 1分 |  | 下一位参与者进行叫分选择，选择分别为“2分”和“3分” |  |
| 2分 |  | 下一位参与者进行叫分选择，选择为“3分” |  |
| 3分 |  | 选择“3分”的人自动成为地主，其他三人为农民。地主分33张牌，农民各分25张牌 |  |
| 出牌 | 点击选中要出的牌，点击“出牌” | 检查所选牌是否符合出牌规则 | 若不符合出牌规则，界面中央显示提示文字“不符合规则，请重新选择将要出的牌”，不做任何其他操作；若符合出牌规则，所选牌在自己牌中消失并显示在公共区域。 |  |
| 未选中要出的牌，点击出牌 | 所选牌符合出牌规则 | 界面中央显示提示文字“请选择要出的牌”，不做任何其他操作 |  |
| 不出 | 点击 “不出“ |  | 所有弹出的牌回归原位，并由下一位牌友进行操作 |  |
| 提示 | 点击一次“提示” | 提示操作所选的牌符合出牌规则 | 若无可匹配选择，直接进行“不出”操作，由下一位牌友进行游戏操作；否则显示后台选择出的提示牌面（弹出相应的牌） |  |
| 有可匹配选项后多次点击“提示” | 提示操作所选的牌符合出牌规则 | 根据符合条件的选择，从小到大依次显示（弹出相应的牌） |  |
| 快捷语言 | 点击快捷语言 | 正在游戏中 | 弹出下拉框并进行选择 |  |
| 退出游戏 | 点击退出按钮 | 正在游戏中 | 弹出选择框，其内容为“正在游戏中，退出游戏将会认定为输，扣除200积分，其他3人不会被扣分，并且游戏就此结束。是否退出游戏？”点击“确认”返回上一个界面，点击“取消”继续游戏 |  |
|  |
| 个人与好友信息 | 个人信息 | 点击主界面自己的头像 | 界面跳转 | 跳转至自己的个人信息页面 |  |
| 好友信息 | 邀请好友列表的好友头像 | 界面跳转 | 跳转到所点击好友的个人信息页面 |  |

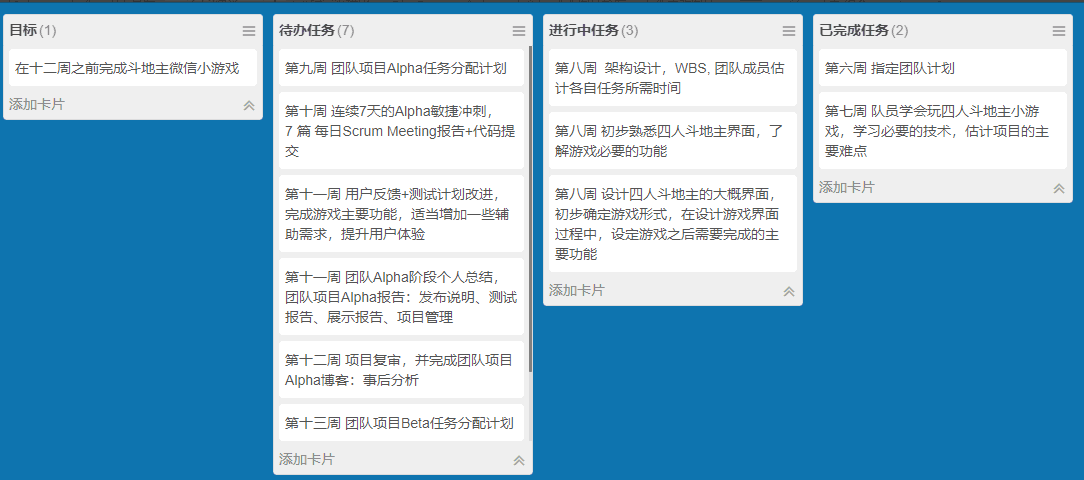
**三、将需求分析报告上传至团队的github上**

**四、功能分析和任务分解WBS**

**1. 功能分析和任务分解WBS(1) 参考《构建之法》P173图8-8，给出功能分析的四个象限定位**

第一象限（杀手功能，必要需求）：用户可以联网登录账号密码跟其他三个用户玩斗地主游戏，或者断网可以玩单机模式的斗地主。&#160;第二象限（外围功能，必要需求）：完善的登录界面，可以注册账号找回密码，整洁的游戏界面第三象限（外围功能，辅助需求）：可以邀请微信好友一起玩。第四象限（杀手功能，辅助需求）：游戏设低中高等级，用户可以自行选择合适的难度玩斗地主。

**(2) 请给出团队项目的WBS**

  
**五、编码规范**

**5.1代码风格规范**

**5.1.1 缩进**

4个空格

**5.1.2 行宽**

100字符

**5.1.3 括号**

在必要或者复杂的条件表达中，需用括号清楚地表达逻辑优先级

**5.1.4 断行与{}**

统一花括号一行，一条语句一行，如：

If（A）

{

语句A；

}

else

{

语句B；

}

**5.1.5 命名**

采用骆驼命名法，第一个字母小写，然后每一个逻辑断点都有一个大写字母标记；

命名尽量有意义。

**5.1.6 下划线**

同一类的对象前面加一个统一标识，如：animala\_cat=new animal()。

**5.1.7 注释**

解释程序是做什么的，为什么这么做，以及某些需要注意的地方；函数的注释应该放在开头。

**5.2代码设计规范**

**5.2.1 函数**

一个函数只需实现一个功能。

**5.2.2 goto**

可以使用goto。

**5.2.3 错误处理**

在DeBug版本中，所有的参数都要验证其正确性。在正式版本中，从外部（用户或别的模块）传递过来的参数要验证其正确性。

**5.3 代码复审**

**5.3.1 复审**

在复审前必需成功编译，同时编译DeBug|Retail版本；必须测试过代码。

**5.3.2 复审的核查表**

**1．概要部分**

（1）代码能符合需求和规格说明么？

（2）代码设计是否有周全的考虑？

（3）代码可读性如何？

（4）代码容易维护么？

（5）代码的每一行都执行并检查过了吗？

**2．设计规范部分**

（1）设计是否遵从已知的设计模式或项目中常用的模式？

（2）有没有硬编码或字符串/数字等存在？

（3）代码有没有依赖于某一平台，是否会影响将来的移植（如Win32到Win64）？

（4）开发者新写的代码能否用已有的Library/SDK/Framework中的功能实现？在本项目中是否存在类似的功能可以调用而不用全部重新实现？

（5）有没有无用的代码可以清除？（很多人想保留尽可能多的代码，因为以后可能会用上，这样导致程序文件中有很多注释掉的代码，这些代码都可以删除，因为源代码控制已经保存了原来的老代码。）

**3．代码规范部分**

（1）修改的部分符合代码标准和风格么（详细条文略）?

**4．具体代码部分**

（1）有没有对错误进行处理？对于调用的外部函数，是否检查了返回值或处理了异常？

（2）参数传递有无错误，字符串的长度是字节的长度还是字符（可能是单/双字节）的长度，是以0开始计数还是以1开始计数？

（3）边界条件是如何处理的？Switch语句的Default是如何处理的？循环有没有可能出现死循环？

（4）有没有使用断言（Assert）来保证我们认为不变的条件真的满足？

（5）对资源的利用，是在哪里申请，在哪里释放的？有没有可能导致资源泄露（内存、文件、各种GUI资源、数据库访问的连接，等等）？有没有可能优化？

（6）数据结构中是否有无用的元素？

**5．效能**

（1）代码的效能（Performance）如何？最坏的情况是怎样的？

（2）代码中，特别是循环中是否有明显可优化的部分（C++中反复创建类，C#中 string 的操作是否能用StringBuilder 来优化）？

（3）对于系统和网络调用是否会超时？如何处理?

**6．可读性**

代码可读性如何？有没有足够的注释？

**7．可测试性**

代码是否需要更新或创建新的单元测试？

还可以有针对特定领域开发（如数据库、网页、多线程等）的核查表。

**六、系统设计**

**1、体系结构图**

## **2、数据库设计**

用户信息表

| **数据项** | **数据项含义** | **数据类型** | **允许空** |
| --- | --- | --- | --- |
| Uno(主键) | 用户标识 | char(20) | no |
| Uname | 用户昵称 | char(10) | no |
| Uscore | 用户积分 | int | no |
| Unum | 用户游戏总次数 | Int | no |
| Uvictory | 用户胜利次数 | Int | no |

好友信息表

| **数据项** | **数据项含义** | **数据类型** | **允许空** |
| --- | --- | --- | --- |
| Fno(主键) | 好友标识 | char(20) | no |
| Fname | 好友昵称 | char(10) | no |
| Fscore | 好友积分 | int | no |
| Fnum | 好友游戏总次数 | int | no |
| Fvictory | 用户胜利次数 | int | no |

排行榜表

| **数据项** | **数据项含义** | **数据类型** | **允许空** |
| --- | --- | --- | --- |
| Uno(主键) | 用户标识 | char(20) | no |
| Rank | 排名 | int | no |
| Fno(主键) | 好友标识 | char(20) | no |