哈尔滨工业大学

**计算机科学与技术学院/国家示范性软件学院**

**2020年秋季学期**

**《软件过程与工具》课程**

**实验报告**

**Lab 1：增量模型应用+结对编程实践**

|  |  |  |
| --- | --- | --- |
| **姓名** | **学号** | **联系方式** |
| 许倩 | 1183710104 | 15537237025 |
| 李婧辰 | 1183710225 | 13505274428 |

目录

[1 项目概述 3](#_Toc54071163)

[1.1 总体需求分析 3](#_Toc54071164)

[1.2 软件系统总体设计 5](#_Toc54071165)

[1.3 开发任务计划 6](#_Toc54071166)

[1.4 开发环境与工具 7](#_Toc54071167)

[2 第1轮迭代：基本系统 7](#_Toc54071168)

[2.2 迭代目标 7](#_Toc54071169)

[2.3 需求分析 7](#_Toc54071170)

[2.4 设计方案 8](#_Toc54071171)

[2.5 结对编程过程记录 9](#_Toc54071172)

[2.6 单元测试 11](#_Toc54071173)

[2.7 集成测试 11](#_Toc54071174)

[3 第2轮迭代：增量1 12](#_Toc54071175)

[3.2 迭代目标 12](#_Toc54071176)

[3.3 需求分析 12](#_Toc54071177)

[3.4 设计方案 12](#_Toc54071178)

[3.5 结对编程过程记录 14](#_Toc54071179)

[3.6 单元测试 15](#_Toc54071180)

[3.7 集成测试 17](#_Toc54071181)

[4 第3轮迭代：增量2 17](#_Toc54071182)

[4.2 迭代目标 17](#_Toc54071183)

[4.3 需求分析 17](#_Toc54071184)

[4.4 设计方案 17](#_Toc54071185)

[4.5 结对编程过程记录 19](#_Toc54071186)

[4.6 单元测试 20](#_Toc54071187)

[4.7 集成测试 21](#_Toc54071188)

[5 项目总结 22](#_Toc54071189)

[5.2 增量模型应用总结 22](#_Toc54071190)

[5.3 结对编程总结 22](#_Toc54071191)

# 项目概述

（1）项目名称：加减法练习软件——未来星

（2）项目简要介绍：未来星加减法练习软件是面向小学一年级学生设计的练习加减法的软件，软件根据学生选择题目难度出题，批改学生提交答案，对每道题目提供答案，并记录答对和答错题数。软件通过积分奖励机制，激发学生学习动力；并通过卡通图案界面，增加学生学习兴趣。

1. 开发团队成员：许倩、李婧辰。

## 1.1 总体需求分析

1. 项目要开发的软件的目的和意义（回答为什么要开发该项目）：

随着互联网的发展，许多孩子很小就已经对网上的软件充满兴趣。并且许多家长因为工作繁忙，没有时间为孩子辅导功课，以至于孩子不能自觉进行学习，或者孩子在做完练习后无法得知正确结果和自己的作答情况。

本软件以孩子兴趣为引导，使孩子能够使用软件进行加减法练习。并且在孩子做完每道题后能够给出给出结果正确性判断和正确结果显示，并统计做题总数和正确、错误题目数目，使孩子能够养成自学的好习惯，提高孩子的学习主动性和独立性，降低对家长的依赖。

通过积分制和奖惩制，能够激发孩子的学习兴趣，提高孩子学习动力，使孩子在快乐中学习，在学习中快乐。

（2）软件系统的涉众分析（列出该系统跟哪些相关人员有关，其关心的核心内容）：

|  |  |  |
| --- | --- | --- |
| 涉众名称 | 涉众说明 | 人员期望 |
| 用户1：学生 | 外部用户是指使用该软件系统进行100以内整数的加法练习的用户，即小学生学生。 | 通过该软件系统，能够联系100以内加减法，使自己的计算能力得到提升 |
| 用户2：家长 | 家长通过该软件让孩子进行加减法练习 | 能够减轻家长的压力，并使孩子能力得到提升 |
| 用户3：教育机构 | 在线教育机构或网站 | 该软件能够吸引更多学生加入，带来经济收益 |
| 业务运行管理人员 | 负责管理软件系统的对外业务 | 保障软件顺利交付给用户 |
| 产品管理人员 | 负责维护软件系统整体正常运行，管理用户信息和公告的用户 | 软件系统的基本功能能够正确运行，软件可以定期更新 |
| 市场销售人员 | 负责向潜在的内部用户推广该软件系统 | 软件有较大的市场需求 |
| 软件工程师 | 开发软件的人员和团队 | 能够以较少的代码，较高的效率实现软件功能 |
| 支持和维护工程师 | 维护软件的人员和团队 | 软件的错误较少，并且自身有容错机制。 |

1. 软件系统的功能需求：

Story1：小学生自主学习能力较差，在掌握加减法基本规则后，需要大量的题目去进行训练，并且需要在做完题目后得到及时的反馈（即答案是否正确和标准答案）。

Story2：相比较传统做练习题过程，小学一年级学生更容易被游戏类的训练模式所吸引，希望做练习题能够像游戏闯关一样，每答对对一道题就能得到相应奖励，并且题目难度越大，得到的奖励也越大啊，在“玩”中快乐学习。

1. 软件系统的非功能需求：
2. 软件系统在较短的时间内给出习题，并在较短时间内对用户答案进行正误判定；
3. 对一百以内的整数的加减法计算做出正确计算；

界面要求：

1. 色彩明艳；
2. 采用卡通动漫风格设计，迎合小学生喜好；
3. 对用户答题进行积分，并显示个人段位,并在段位提升时弹出对话框提示；

技术指标：

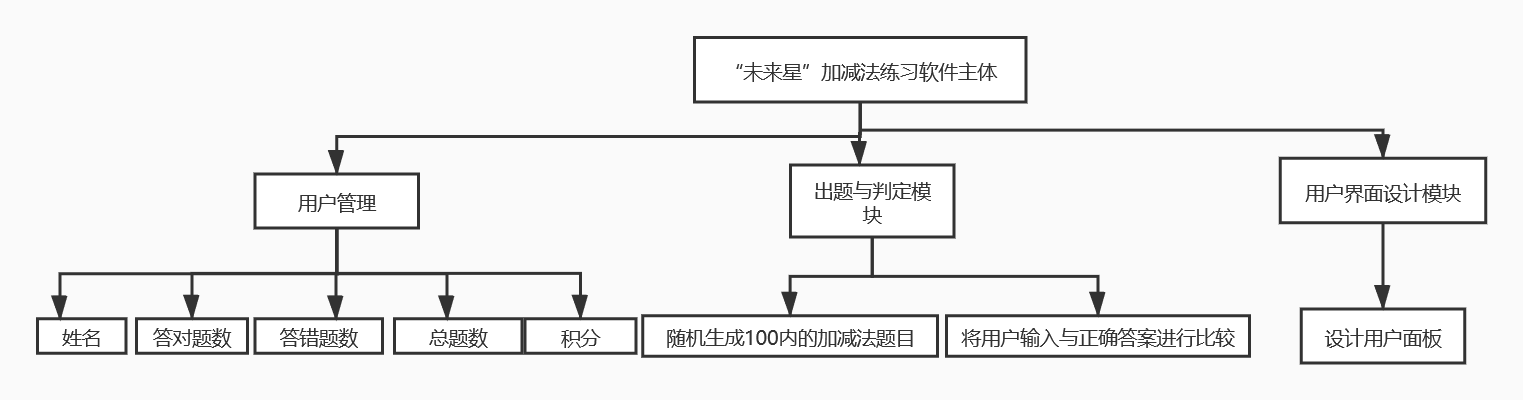
出题时间：200ms内

判断题目正确性时间：200ms内

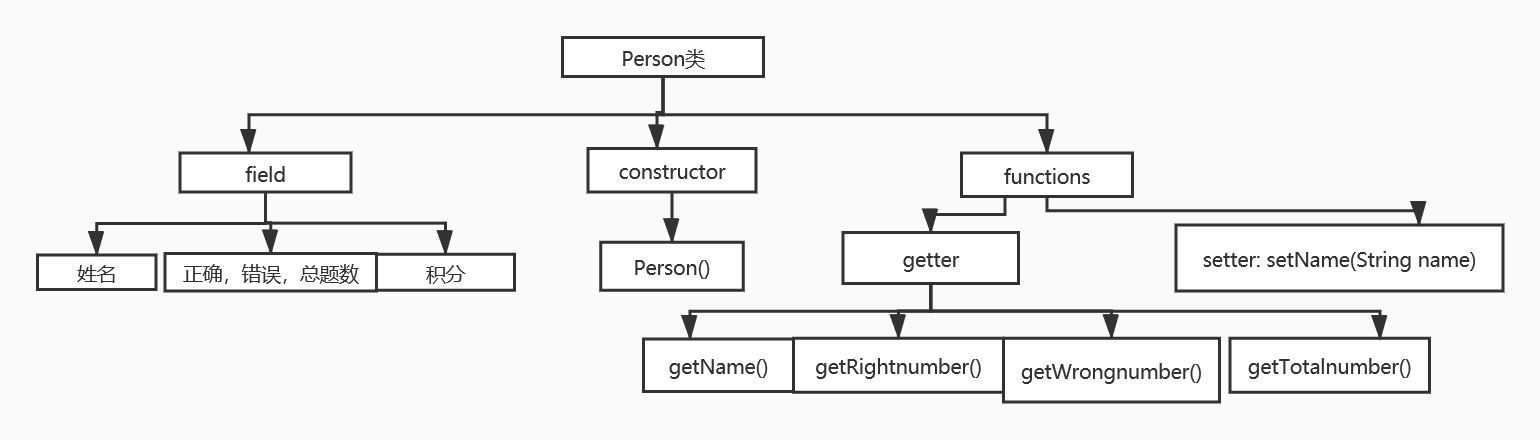
给出标准答案时间：200ms内

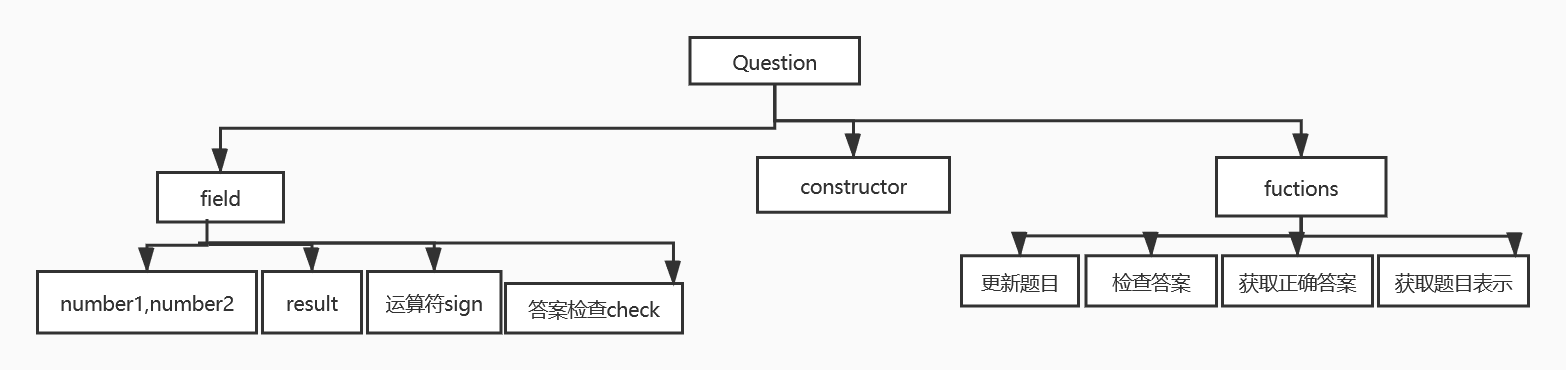
对用户答案进行正确性判断的准确率：100%

## 软件系统总体设计

（1）软件系统功能结构设计方案：给出软件功能模块及模块间的关联关系。

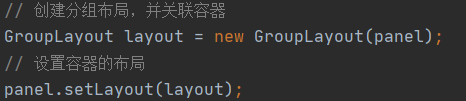
（2）软件系统相关的数据结构设计：给出必要的数据结构设计，包括配置参数、缓冲数据结构等。

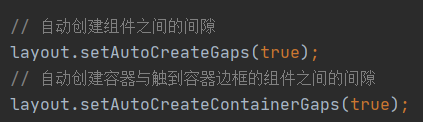


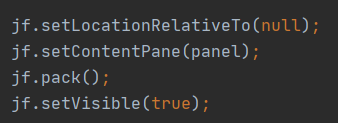


GUI采用分组布局结构，配置如下：









## 开发任务计划

1. 第1轮迭代的任务：完成加减法练习软件基本功能。

|  |  |  |
| --- | --- | --- |
| 时间 | 人员 | 具体任务安排 |
| 2020.10.16 | 许倩，李婧辰 | 完成需求分析和部分项目文档 |
| 2020.10.17 | 许倩，李婧辰 | 结对编程，开发基础类类，完成内部逻辑，初步设计GUI。完成姓名输入，作答情况显示，题目显示，正确答案显示，题目提交等基本功能 |
| 2020.10.17 | 许倩，李婧辰 | 对GUI进行设计修改，优化 |

1. 第2轮迭代的任务：完成附加功能1——进行题目难度的选择。

|  |  |  |
| --- | --- | --- |
| 时间 | 人员 | 具体任务安排 |
| 2020.10.18 | 许倩，李婧辰 | 设计题目难度选择的算法 |
| 2020.10.19 | 许倩，李婧辰 | 完成下拉栏选择题目难度，并根据难度出题的界面功能 |

1. 第3轮迭代的任务：完成附加功能2——设置积分制和奖惩制。

|  |  |  |
| --- | --- | --- |
| 时间 | 人员 | 具体任务安排 |
| 2020.10.19 | 许倩，李婧辰 | 在用户类中记录，控制积分 |
| 2020.10.19 | 许倩，李婧辰 | 完成根据选择难度增加积分，积分显示功能。美化界面。 |

## 开发环境与工具

开发语言：Java

开发环境与工具：Win10，IDEA

# 第1轮迭代：基本系统

## 迭代目标

（1）任务1：通过加减法基本运算算法对输入结果进行判定。

（2）任务2：加减法题目显示栏，结果输入框。

（3）任务3：显示作答题目总数，显示答对题数，答错题数。

（4）任务4：标准答案显示。

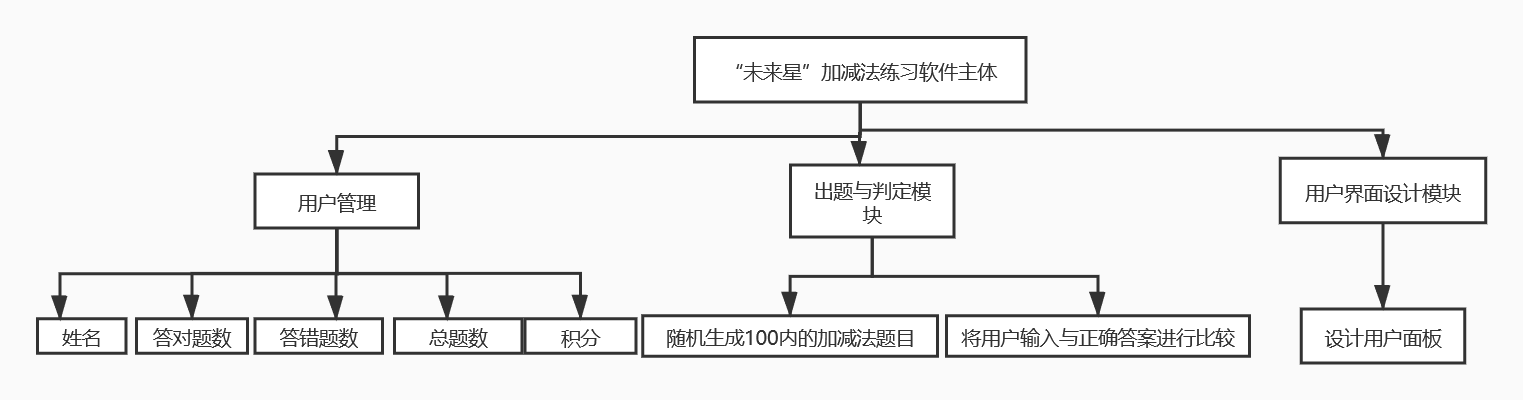
## 需求分析

1. 用户能够看到题目，并且能够输入并提交答案
2. 在答完题后，用户能够看到自己的答案是否正确，并且能够知道正确答案
3. 用户需要知道到目前为止，自己的作答题目总数，答对题数，答错题数

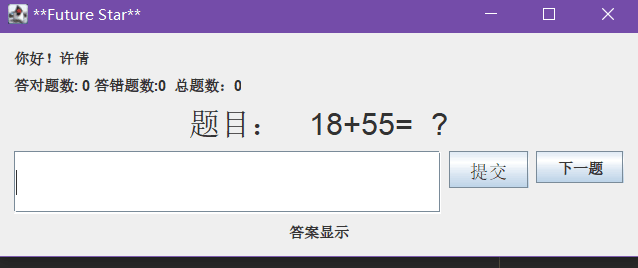
## 设计方案

1. 详细设计方案：模块功能结构设计，界面原型设计等。

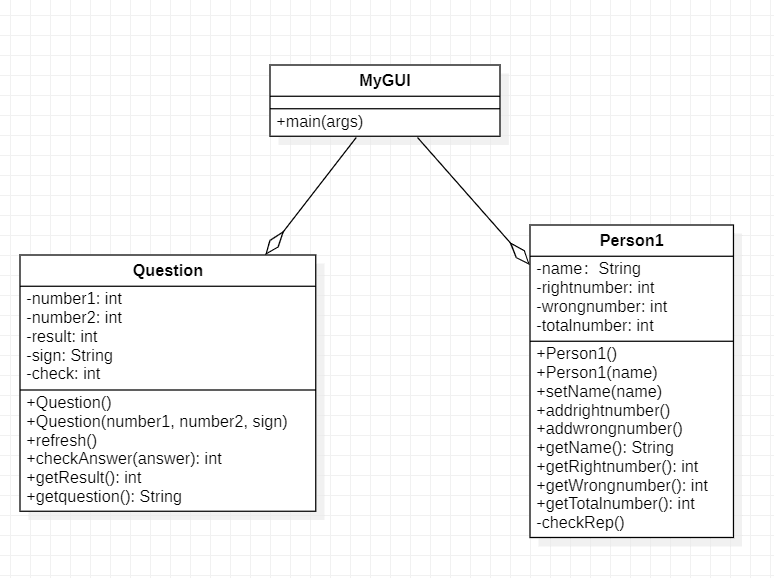
模块功能结构设计：



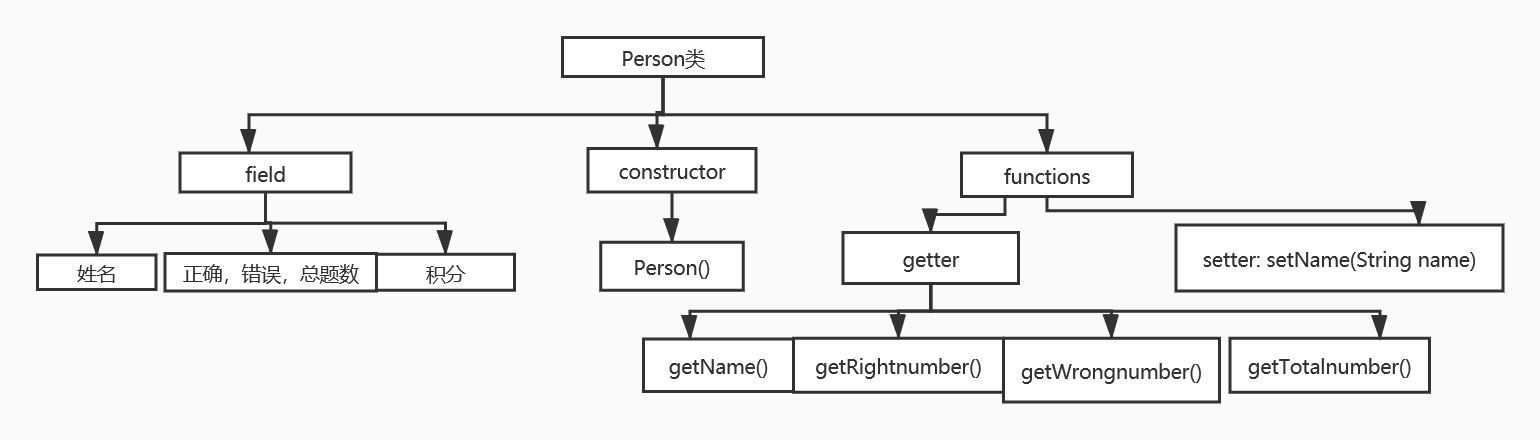
界面原型设计：



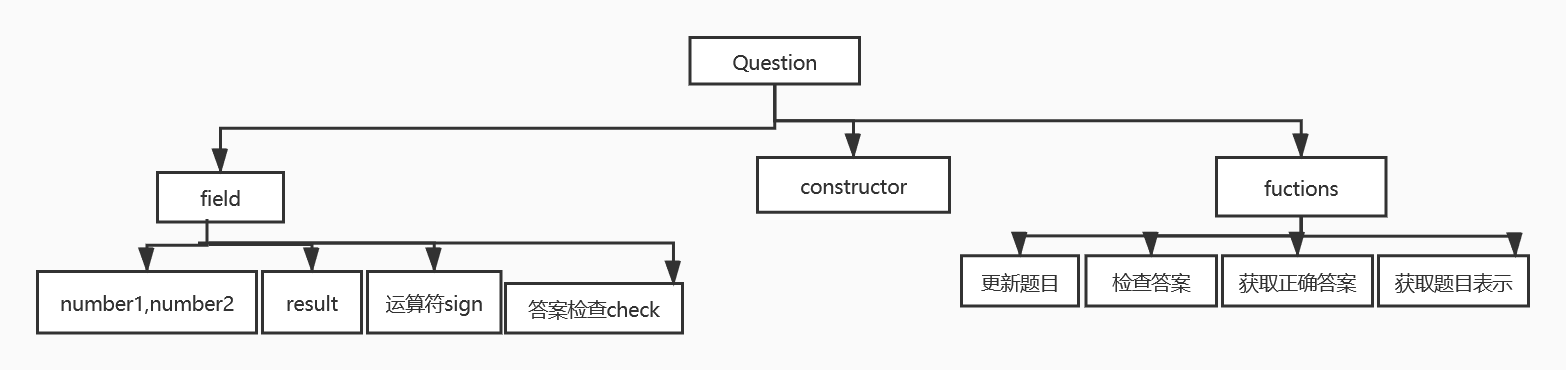
（2）模块相关的设计类（Design Class）：给出每个类的名字、属性、方法。



Person类：



Question类：



（3）所需的数据结构。

所用的是基础数据类型。

GUI界面采用分组布局

## 结对编程过程记录

**（1）角色切换与任务分工**

表1-1 第1轮迭代过程结对编程角色与任务分工

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 日期 | 时间(HH:MM - HH:MM) | 驾驶员角色 | 领航员角色 | 本段时间的任务 |
| 10.17 | 15:40-16:00 | 许倩 | 李婧辰 | 完成内部逻辑 |
| 10.17 | 16:00-16:40 | 李婧辰 | 许倩 | 完成测试，GUI原型 |
| 10.17 | 16:40-17：30 | 许倩 | 李婧辰 | 完善GUI，完成本增量所有功能 |

**（2）工作日志**

由领航员负责记录，记录结对编程期间的遇到的问题、两人如何通过交流合作解决每个问题的。

表1-2 第1轮迭代过程结对编程工作日志

|  |  |  |  |
| --- | --- | --- | --- |
| 日期/时间 | 问题描述 | 最终解决方法 | 交流过程 |
| 2020.10.17  16:10 | 如何编写正确代码完成GUI布局 | 使用GroupLayout分组布局 | 首先讨论确定需要的组件，然后再草稿纸上画出组件布局，然后通过验证不同的布局格式，经过讨论找到最合适的布局 |
| 2020.10.17  15:00 | 是否要增加接口和抽象类 | 不需 | 讨论增加抽象类、接口与单纯使用多个类的优点与区别。因为本系统逻辑相对简单，所以决定不适用抽象类 |

**（3）结对编程工作现场照片**

请其他同学帮助拍摄结对编程现场照片至少2张。



第1次迭代结对编程现场照片1



第1次迭代结对编程现场照片2

## 单元测试

对本轮迭代所开发的每个类进行单元测试。

表1-3 第1轮迭代过程的单元测试记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 日期/时间 | 类名：方法名 | 输入数据 | 输出数据 | 结论 |
| 2020.10.17  16:00 | testPerson: testnumber1() | 正确，错误，总题数分别为（1，1，2）的情况 | (1,1,2) | 计算正确，错误，总题数的函数正确 |
| 2020.10.17  16:00 | testPerson: testanumber2() | 正确，错误，总题数分别为（0，0，0）的情况 | (0,0,0) | 计算正确，错误，总题数的函数正确 |
| 2020.10.17  16:00 | testPerson:  testname() | 将用户名称修改为"name1"，再对用户名称进行获取 | "name1" | 能够正确的修改用户名称和获取用户名称 |
| 2020.10.17  16:00 | testQuestion:testResult() | 题目为1-1 | 0 | 说明Question中计算正确答案的部分正确 |
| 2020.10.17  16:00 | testQuestion:testResult() | 题目为1+1 | 2 | 说明Question中计算正确答案的部分正确 |
| 2020.10.17  16:00 | testQuestion:testCheck() | 判断4-1=3是否被正确判断 | 1 | 说明checkAnswer函数可以正确判断用户输入是否为正确答案 |

## 集成测试

对本轮迭代所开发功能模块进行测试。

表1-4 第1轮迭代过程的集成测试记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 日期/时间 | 功能名称 | 测试用例 | 测试结果描述 | 结论 |
| 2020.10.17  17:30 | 正确记录作答情况 | 4个正确答案，4个错误答案 | 显示正确题数：4 错误题数：4 总题数8 | 测试成功，无错误 |
| 2020.10.17  17:30 | 正确判断答案是否正确，并显示标准答案 | 4个正确答案，4个错误答案 | 每个正确答案，显示“恭喜你，答对了”，错误答案，会显示正确的标准答案 | 测试成功 |
| 2020.10.17  17:30 | 正确提交 | 4次提交 | 每次都提交正确 | 测试成功 |
| 2020.10.17  17:30 | 更新题目和标准答案 | 4个错误答案 | 会显示新的正确的标准答案 | 测试成功 |

# 第2轮迭代：增量1

## 迭代目标

（1）任务1：设计算法根据难度选择出题。

（2）任务2：在界面显示难度等级，并能够在界面勾选难度等级。

## 需求分析

本轮迭代所开发模块的详细需求分析。

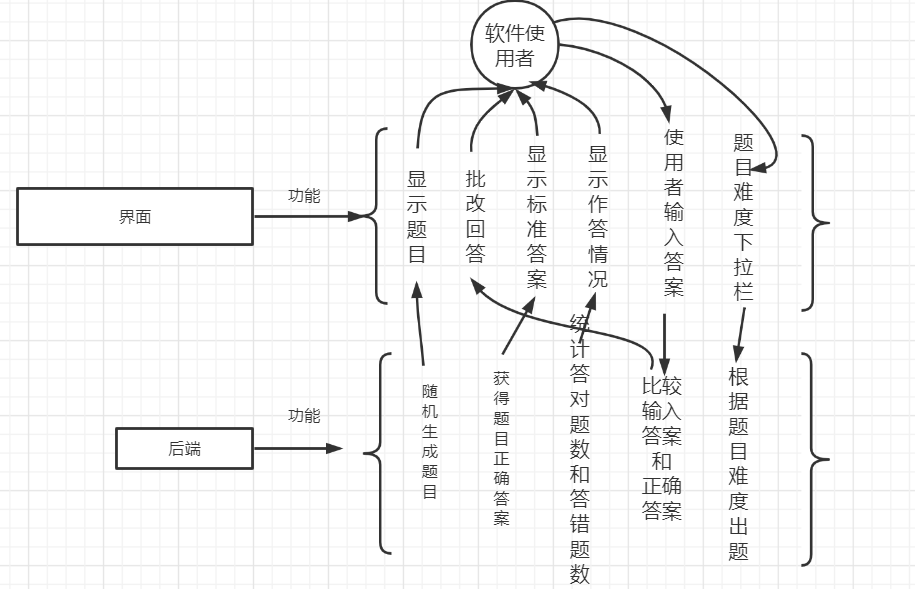
1. 用户可以在界面进行难度等级的勾选
2. 用户可以进行选择难度下的题目练习

## 设计方案

针对本轮迭代的开发内容给出设计方案：

1. 详细设计方案：模块功能结构设计，界面原型设计等。

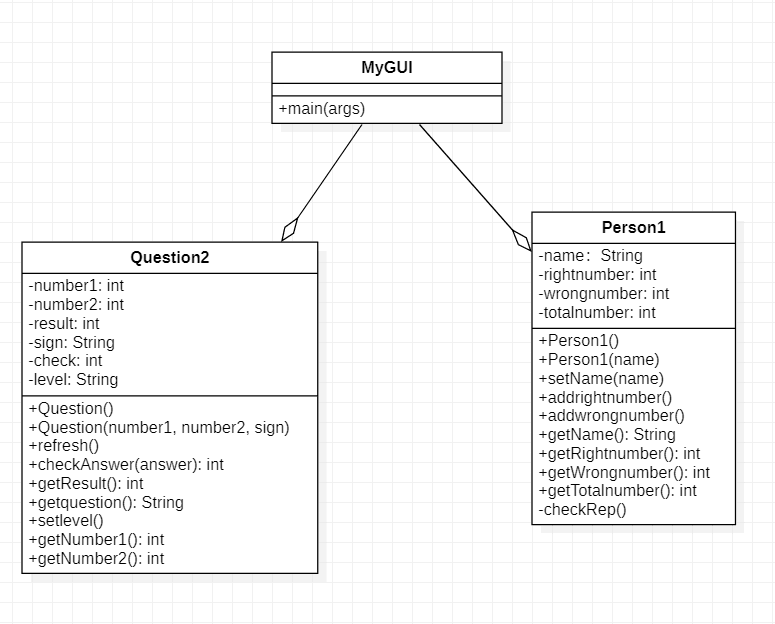
功能结构图：



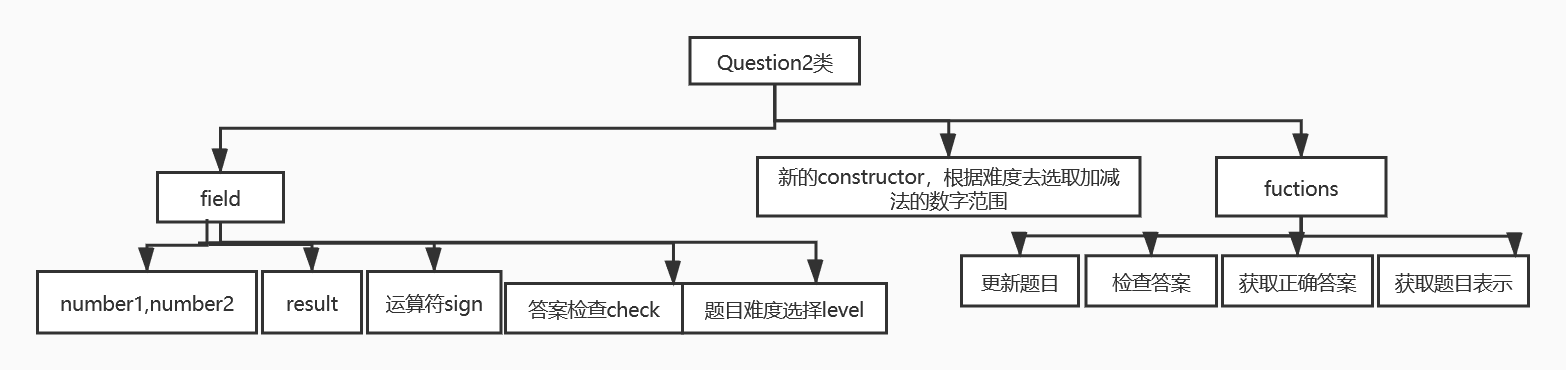
界面原型：



1. 模块相关的设计类（Design Class）：给出每个类的名字、属性、方法。



Question2类：（新增了根据用户选择的题目难度level去创建题目的方法）



（3）所需的数据结构。

同基本系统

## 结对编程过程记录

**（1）角色切换与任务分工**

表2-1 第2轮迭代过程结对编程角色与任务分工

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 日期 | 时间(HH:MM - HH:MM) | 驾驶员角色 | 领航员角色 | 本段时间的任务 |
| 2020.10.17 | 17：35-17:50 | 李婧辰 | 许倩 | 设计Qusetion2类，进行测试 |
| 2020.10.17 | 17:50-18:40 | 许倩 | 李婧辰 | 完成在界面下拉栏显示，选择难度，并根据难度出题的功能 |

**（2）工作日志**

由领航员负责记录，记录结对编程期间的遇到的问题、两人如何通过交流合作解决每个问题的。

表2-2 第2轮迭代过程结对编程工作日志

|  |  |  |  |
| --- | --- | --- | --- |
| 日期/时间 | 问题描述 | 最终解决方法 | 交流过程 |
| 2020.10.17  17:30 | 如何对题目难度进行设置，设置什么样的难度分层 | 根据用户选择的难度，创建不同难度的Question | 双方都讲述自己对难度的设计思路，最后采用个位数加减法，50内加减法，50到100的较大数的加减法 |
| 2020.10.17  17:50 | 如何让用户对题目难度进行选择 | 用户在交互界面选择题目难度 | 先考虑原始的用户输入题目难度的编号，然后考虑的是用户在下拉框选择题目难度 |

**（3）结对编程工作现场照片**

请其他同学帮助拍摄结对编程现场照片至少2张。



第2次迭代结对编程现场照片1



第2次迭代结对编程现场照片2

## 单元测试

对本轮迭代所开发的每个类进行单元测试。

表2-3 第2轮迭代过程的单元测试记录

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 日期/时间 | 类名：方法名 | 输入数据 | | 输出数据 | 结论 |
| 2020.10.17  17:50 | testPerson: testnumber1() | 正确，错误，总题数分别为（1，1，2）的情况 | | (1,1,2) | 计算正确，错误，总题数的函数正确 |
| 2020.10.17  17：50 | testPerson: testanumber2() | 正确，错误，总题数分别为（0，0，0）的情况 | (0,0,0) | | 计算正确，错误，总题数的函数正确 |
| 2020.10.17  17：50 | testPerson:  testname() | 将用户名称修改为"name1"，再对用户名称进行获取 | "name1" | | 能够正确的修改用户名称和获取用户名称 |
| 2020.10.17  17：50 | testQuestion:testResult() | 题目为1-1 | 0 | | 说明Question中计算正确答案的部分正确 |
| 2020.10.17  17：50 | testQuestion:testResult() | 题目为1+1 | 2 | | 说明Question中计算正确答案的部分正确 |
| 2020.10.17  17：50 | testQuestion:testCheck() | 判断4-1=3是否被正确判断 | 1 | | 说明checkAnswer函数可以正确判断用户输入是否为正确答案 |
| 2020.10.17  17：50 | testQuestion:testGetQuestion() | 判断题目"4-1=?"是否被正确显示 | "4-1=?" | | 说明getQuestion()可以正确显示题目 |
| 2020.10.17  17：50 | testQuestion:testlevel1() | 对设置和获取难度级别1进行检查 | 获取到正确的难度级别1，随机生成的两个数均为个位数 | | 正确生成难度级别为1的题目 |
| 2020.10.17  17：50 | testQuestion:testlevel2() | 对设置和获取难度级别2进行检查 | 获取到正确的难度级别2，随机生成的两个数均为50以内的数 | | 正确生成难度级别为2的题目 |
| 2020.10.17  17：50 | testQuestion:testlevel3() | 对设置和获取难度级别3进行检查 | 获取到正确的难度级别3，随机生成的两个数均为50到100区间内的数 | | 正确生成难度级别为3的题目 |

## 集成测试

对本轮迭代所开发功能模块进行测试。

表2-4 第2轮迭代过程的集成测试记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 日期/时间 | 功能名称 | 测试用例 | 测试结果描述 | 结论 |
| 2020.10.17  18:45 | 根据用户选择的题目难度（1，2，3分别进行测试）创建加减法题目 | 正确创建难度梯度为1，2，3所要求的题目 | 显示三个难度梯度的题数 | 测试成功，无错误 |

# 第3轮迭代：增量2

## 迭代目标

* + - 1. 任务1：对用户的答题情况进行积分。
      2. 任务2：将积分显示在界面上

## 需求分析

本轮迭代所开发模块的详细需求分析。

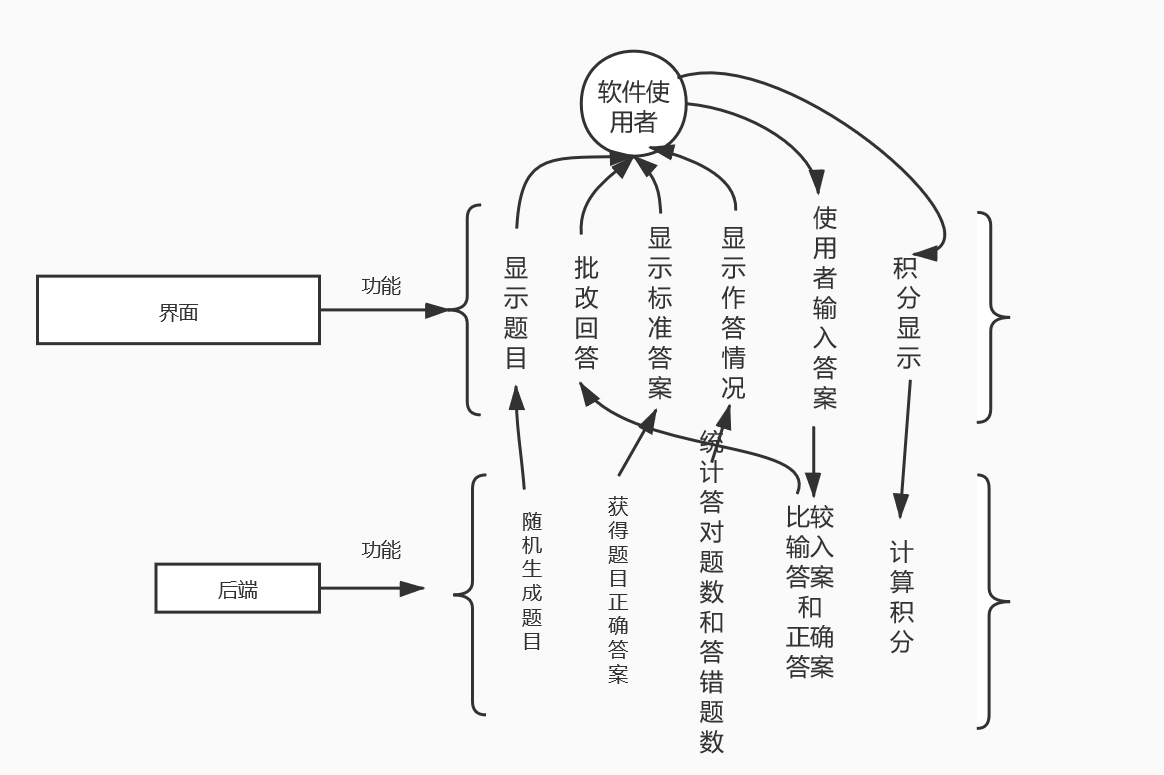
1. 用户可以在界面上实时看到自己答题的积分情况
2. 根据用户的答对题数进行积分

## 设计方案

针对本轮迭代的开发内容给出设计方案：

（1）详细设计方案：模块功能结构设计，界面原型设计等。

模块功能结构设计：

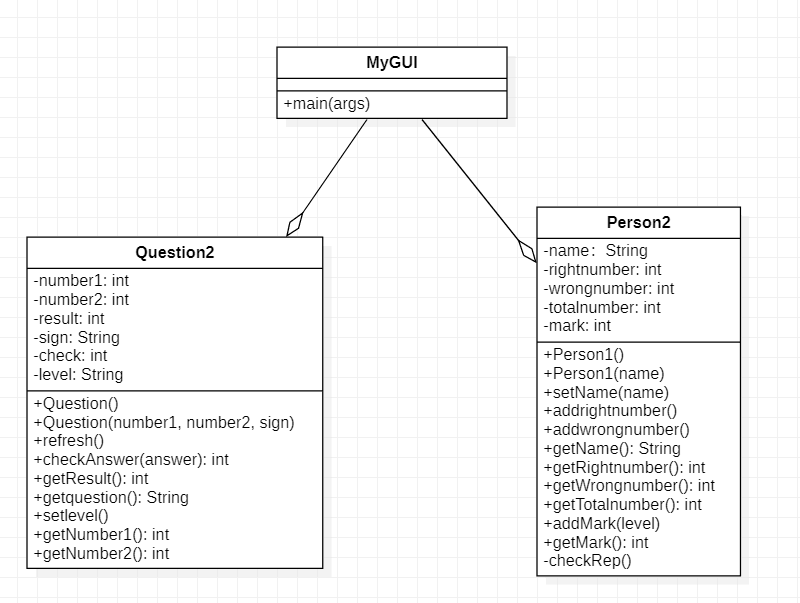


界面原型设计：

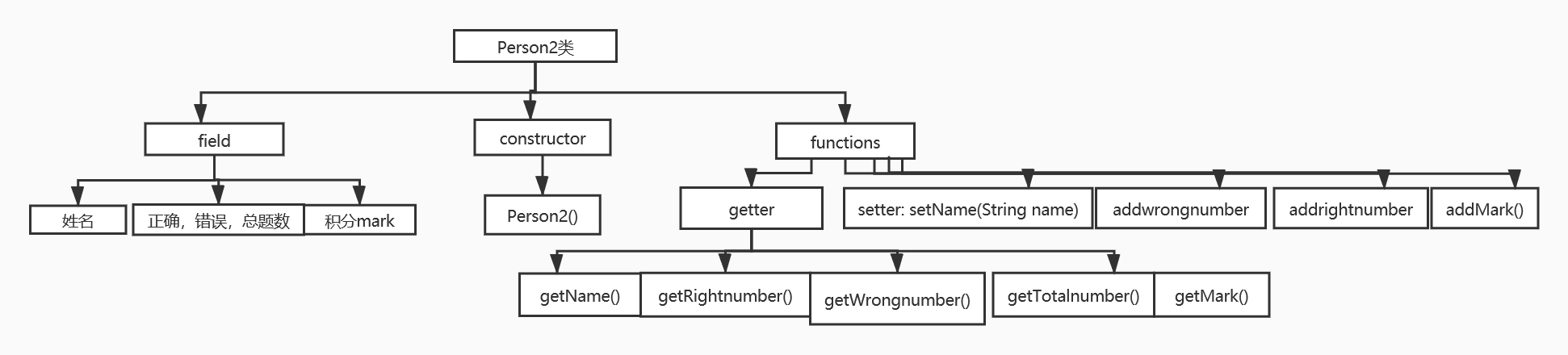


（2）模块相关的设计类（Design Class）：给出每个类的名字、属性、方法。

UML：



Person2类：（新增了积分属性及相关方法）



（3）所需的数据结构。

数据结构同基本系统

## 结对编程过程记录

**（1）角色切换与任务分工**

表2-1 第2轮迭代过程结对编程角色与任务分工

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 日期 | 时间(HH:MM - HH:MM) | 驾驶员角色 | 领航员角色 | 本段时间的任务 |
| 2020.10.19 | 18：30-18:50 | 李婧辰 | 许倩 | 构造Person2类，完成积分的算法，并完成测试 |
| 2020.10.19 | 20：10-21：10 | 许倩 | 李婧辰 | 在GUI页面显示积分，并根据难度增加积分 |

**（2）工作日志**

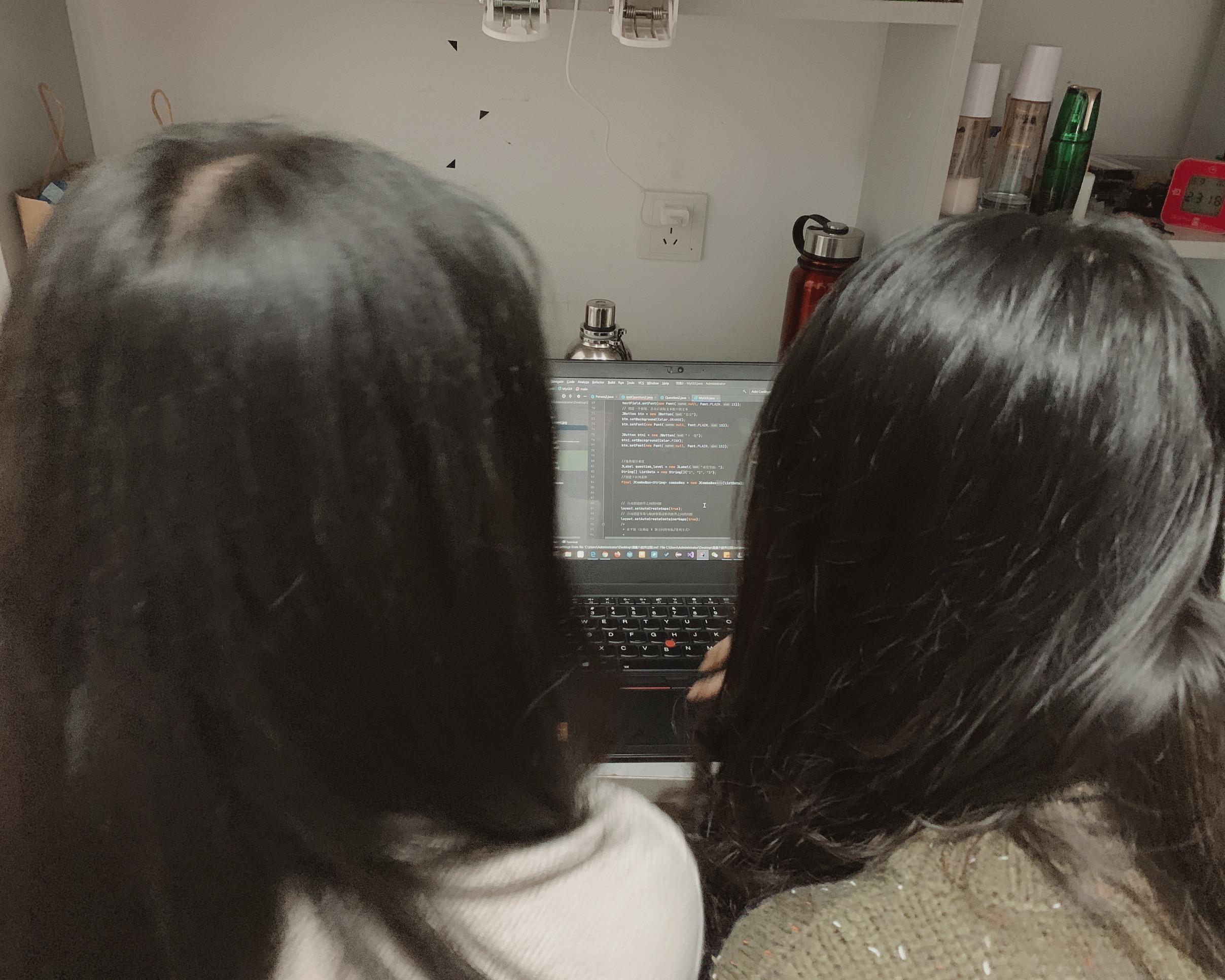
由领航员负责记录，记录结对编程期间的遇到的问题、两人如何通过交流合作解决每个问题的。

表2-2 第2轮迭代过程结对编程工作日志

|  |  |  |  |
| --- | --- | --- | --- |
| 日期/时间 | 问题描述 | 最终解决方法 | 交流过程 |
| 2020.10.19 | 用户积分的具体计算 | 通过用户正确题数，计算积分，错题不减积分 | 参考很多游戏积分类型软件设计，错误失败不减分，且用户积分与答对题目的难度有关，使用户更与成就感 |

**（3）结对编程工作现场照片**

请其他同学帮助拍摄结对编程现场照片至少2张。



第3次迭代照片1



第3次迭代照片2

## 单元测试

对本轮迭代所开发的每个类进行单元测试。

表2-3 第2轮迭代过程的单元测试记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 日期/时间 | 类名：方法名 | 输入数据 | 输出数据 | 结论 |
| 2020.10.19  18:30 | testPerson2：testname() | "name1" | "name1" | 能够正确的设置和获取用户名 |
| 2020.10.19  18:30 | testPerson2: testnumber1() | 正确，错误，总题数分别为（1，1，2）的情况 | (1,1,2) | 计算正确，错误，总题数的函数正确 |
| 2020.10.19  18:30 | testPerson2: testanumber2() | 正确，错误，总题数分别为（0，0，0）的情况 | (0,0,0) | 计算正确，错误，总题数的函数正确 |
| 2020.10.19  18:30 | testPerson2：testmark1() | 对用户积分计算进行测试，答对难度级别分别为1的题目1题，积分应该为1 | 1 | 积分计算正确 |
| 2020.10.19  18:30 | testPerson2：testmark2() | 对用户积分计算进行测试，答对题目难度为2的题目1题，积分应该为3 | 3 | 积分计算正确 |
| 2020.10.19  18:30 | testPerson2：testmark3() | 对用户积分计算进行测试，答对题目难度为3的题目1题，积分应该为5 | 5 | 积分计算正确 |

【注意】该表格可自行增加更多的行

## 集成测试

对本轮迭代所开发功能模块进行测试。

表2-4 第2轮迭代过程的集成测试记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 日期/时间 | 功能名称 | 测试用例 | 测试结果描述 | 结论 |
| 2020.10.19  21:10 | 计算并在界面显示用户积分 | 答对难度为1级和2级别的题目1题，答错一题 | 积分为4 | 积分计算正确 |

# 项目总结

## 增量模型应用总结

对增量模型应用过程进行总结，包括但不限于以下几方面的内容：

* 增量划分是根据工作量划分还是根据用户需求来划分？

用户需求

* 第1个增量你们团队是如何确定的？

根据用户story1，通过用户的核心需求，确定软件的基本功能

* 你认为增量模型的好处在哪里？有什么不好之处吗？它的适用场合是什么？

好处：

1. 在时间要求较高的情况下可以交付产品（完成几个增量）
2. 逐步增加产品功能
3. 最重要的基础模块首先完成，并首先被检验
4. 对小型软件，只需要一个开发团队时，较好

不好：

1. 如何在不改变原来软件已有架构和代码的的条件下，加入新的功能模块
2. 无法处理需求发生变化情况
3. 前一个增量交付后，才能进行下一个增量，对多个团队软件协作开发不友好

适用场合：

1. 要求较短时间得到产品
2. 团队数目较少
3. 需求变化较少

## 结对编程总结

对结对编程过程进行总结，包括但不限于以下几方面的内容：

* 与2人分别编程相比，是否体验到编程效率的提高、编程质量的提高？

答：虽然结对编程花费的时间更多，但是结对编程的代码质量高，编程效率提升，编程质量也得到提高。

* 你认为，你们2人本次合作是否真的提高了效率和质量？有什么成功的体验和需要反省的不足或教训？

答：是。两人合作，可以减少很多不必要的基础错误，同时在编程过程中持续交流，逐步完成项目目标。不足：两人结对编程，交流得到的思路，参考的资料重合度较高，不易于思维的发散。

* 你认为结对编程的优势在哪里、有什么不适应之处？它的适用场合是什么？

答：优势：相互监督，不容易开小差偷懒；相互学习编程思路和技巧；产生的bug比个人编程少；

不适应之处：两人需要协调共同工作时间；难度较高的项目，首先需要个人的深入研究和独立钻研；在进行运行测试等步骤时，两人同时等待，浪费时间；

适用场合：成员关系融洽，可以高效率的进行合作；成员拥有可以相互交流，协同工作的开放工作环境；