# “研题”软件工程项目报告

目录

[“研题”软件工程项目报告 1](#_Toc155522410)

[1. 项目内容 1](#_Toc155522411)

[2. 需求分析 1](#_Toc155522412)

[3. 软件结构 3](#_Toc155522413)

[4. 系统用例图和类图 3](#_Toc155522414)

[5. 软件设计流程 5](#_Toc155522415)

[5.1整体框架设计 5](#_Toc155522416)

[5.2前端框架设计 6](#_Toc155522417)

[5.3后端框架设计 12](#_Toc155522418)

[5.4爬虫实现 21](#_Toc155522419)

[5.5其他 23](#_Toc155522420)

[6. 软件运行结果 24](#_Toc155522421)

[6.1注册账户 24](#_Toc155522422)

[6.2登录界面 25](#_Toc155522423)

[6.3首页 25](#_Toc155522424)

[6.4练习 26](#_Toc155522425)

[6.5错题 29](#_Toc155522426)

[6.6我的 31](#_Toc155522427)

[7. 成员感想 35](#_Toc155522428)

[8. 项目总结 42](#_Toc155522429)

[9. 其他文件引导 43](#_Toc155522430)

## 项目内容

本项目软件提供了一个便捷美观且交互简易的刷题平台“研题”，使考研学子可以有选择地进行各类题目的练习，本软件主要针对计算机408考研用户。通过科目分类和难度区分，用户可以针对自己的学习需求有针对性地进行练习，提高解题能力和应试水平。

同时，软件还关注错题管理和个人学习进度追踪，帮助用户更好地管理自己的学习过程。通过清晰地列出错题并提供解析，帮助用户针对性地改进弱点，提升学习效率。个人学习进度的追踪则能够让用户清晰地了解自己的学习状态，有助于制定更合理的学习计划。

1. **需求分析**

**2.1背景与目标**

该软件旨在帮助考研学生高效地准备考试。它将提供一个功能完备的刷题平台，使用户能够有选择地练习各科目的题目，并且便捷地管理错题，同时提供个人界面来跟踪自己的学习进度。

**2.2.用户特征**

考研学生：有意愿通过刷题来提高自己的考试成绩的学生。

对计算机技术熟悉程度：基础操作熟练，不要求专业程度的计算机技能。

**2.3主要功能**

**2.3.1 刷题页面**

题目类型选择：用户能够按照科目分类，如数学、英语、数据结构等，快速选择所需练习的题目类型。

难度选择：提供不同难度的题目供用户选择。

题目展示：以清晰易读的方式展示题目及其选项，确保题目完整性。

答题交互：用户可选择答案，并能即时查看答案的正确与否，同时给予解析。

**2.3.2 错题页面**

错题展示：清晰列出用户做错的题目，提示题目重点考察内容，高亮题目主干，并提供详细解析。

错题分类：按科目、难度等因素对错题进行分类整理。

**2.3.3 用户个人界面**

个人信息管理：允许用户更改个人信息，如用户名、密码等。

学习进度追踪：提供学习统计数据，如考研初试倒计时、题目完成数量、正确率等，帮助用户了解自己的学习状态。

**2.4数据存储与管理**

MySQL数据库：用于存储题目信息、用户个人信息及学习数据。

数据备份与恢复：提供备份功能，确保用户数据的安全性。

**2.5性能与可靠性**

稳定性：保证软件运行稳定，避免崩溃或数据丢失。

响应速度：在用户操作时保持良好的响应速度，提高用户体验。

**2.6用户界面设计**

简洁易用：界面设计简单明了，易于操作。

美观性：考虑界面美观性，提升用户体验。

**2.7其他要求**

安全性：用户数据安全性是首要考虑的因素，确保用户信息不被非法获取。

软件更新：定期提供软件更新，增加新功能或修复已知问题。

## 软件结构

该软件主要分为四个主题，如图1：

1. 登录：提供用户注册登录功能使用户能够使用本软件。
2. 刷题页面：提供用户7个考研科目的试题练习，提高用户解题能力和应试水平。
3. 错题页面：对用户刷题过程中的错题进行收集，用户能够重复刷错题巩固知识。
4. 个人主页：提供个人信息、设置、切换账号功能。

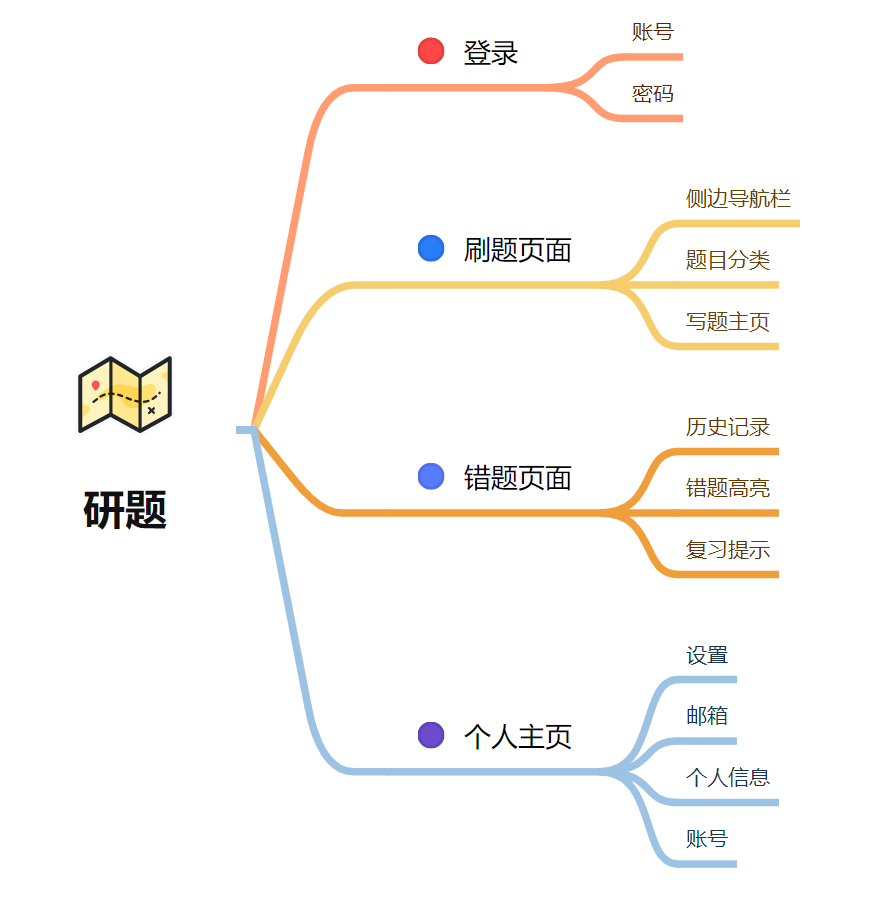


图1-软件结构图

## 系统用例图和类图

系统用例图，如图2：

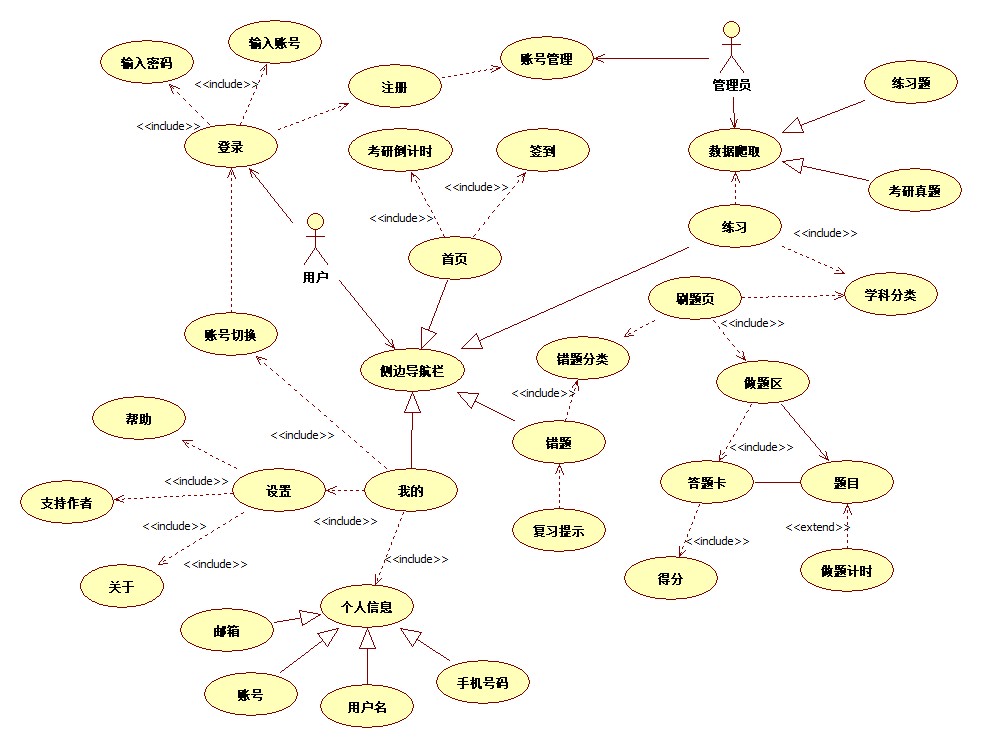


图2-系统用例图

系统类图，如图3：

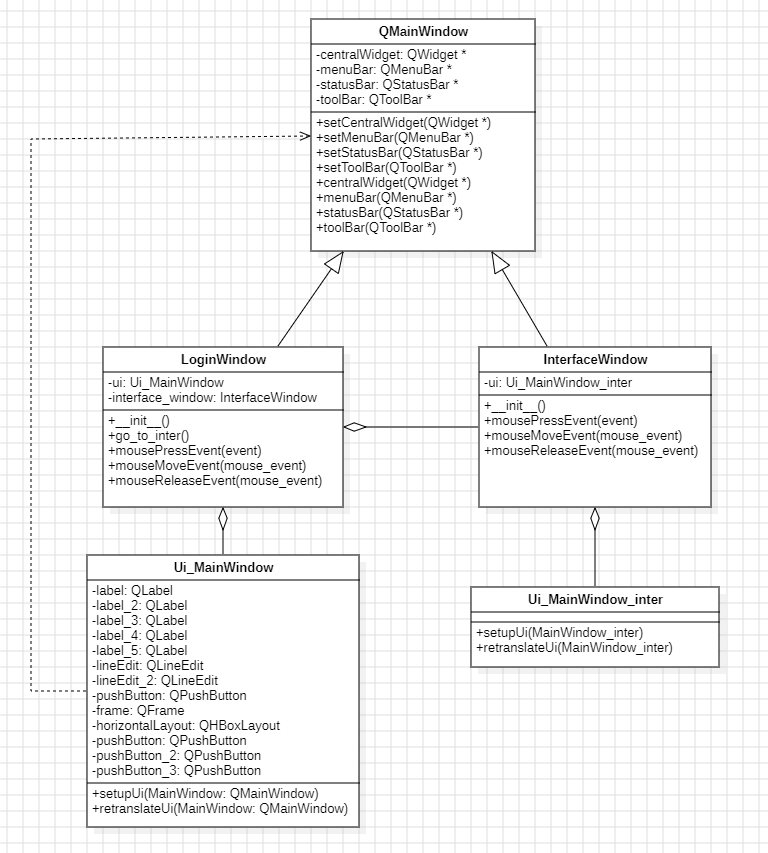


图3-系统类图

## 软件设计流程

### 5.1整体框架设计

本项目主要采用pyQt5进行前端设计。其所需的图片背景等，主要囊括在“UI前端”文件夹中。项目代码主要采用python进行编写，其中前后端互联部分主要囊括在“UI前端”文件夹中。后端主要包括服务器和数据库，也是由python编写，其实现内容主要囊括在“后端”文件夹中。数据库采用MySQL存取数据，题目数据爬取主要采用python爬虫，主要代码实现在“爬虫”文件夹中，题目分别存在“images”和“row question”文件夹中，“images”文件夹主要存放带有图片的题目、“row question”文件夹则存放一般题目。前期的需求分析内容则“需求分析”文件夹中。 “python\_language\_rules.rst” 是编码规范文件，其余则是“数据库设计”文档，“版本管理”文档，每次开会的“会议记录”，题目清洗和整合文档“oj.doc”，“题库修改整合版.doc”和项目的“README”文档以及部分项目运行截图“.png”，如图4：

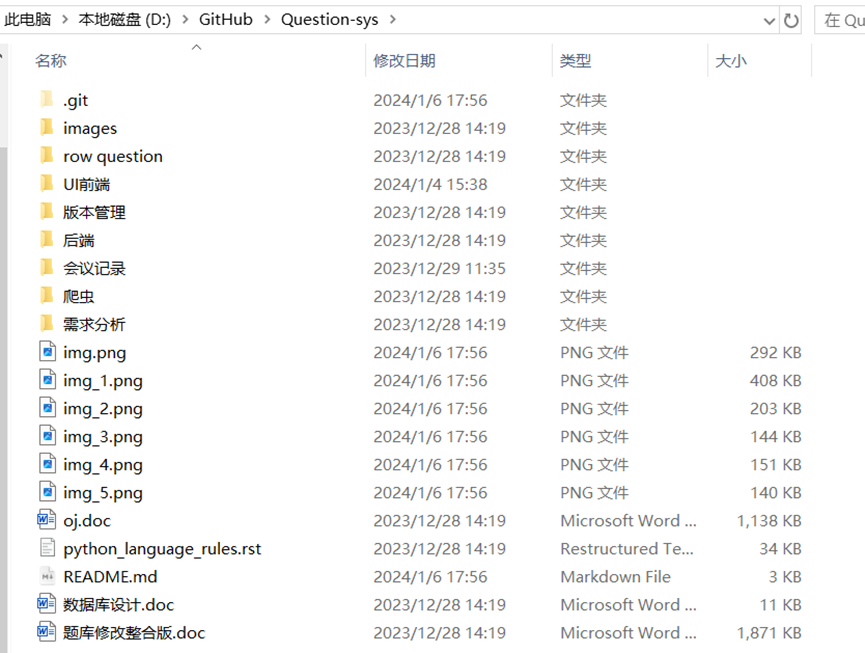


图4-项目整体框架设计

### 5.2前端框架设计

前端设计代码文件均保存在“UI前端”文件夹中，其中，同名有序文件夹存放项目的文件更新迭代不同版本的内容，可以通过查看之前的设计文件了解我们项目整体的版本更新过程，具体内容如图5：



图5-UI前端

**5.2.1 前端界面UI设计**

我们的最终版本前端页面实现python代码文件存放在图5中的0.0.12文件夹中（在github上为“0.0.12/item”文件夹），该python文件整合了所有设计本软件页面的前端代码。实现页面的文件主要有两个，分别为实现登录、注册页面的logindemo.py文件和实现软件内部主要页面Yanti\_v4\_new.py文件。这两个页面实现文件的代码主要分为两个函数。首先，第一个函数setupUi负责生成页面及UI，定义页面、UI的样式如位置、颜色等从而美化页面，第二个函数的功能是重新加载UI，将UI文本标签插入对应UI中并重载，实现前端页面，如图6、7：

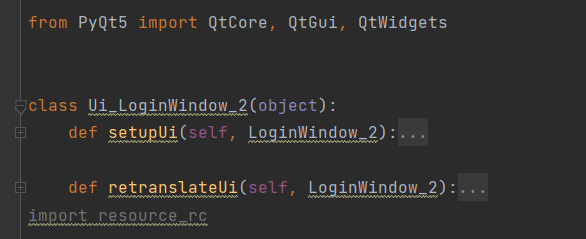


图6-logindemo.py实现登录、注册页（代码量：约313行）

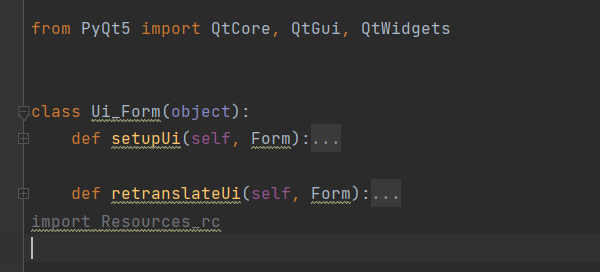


图7-Yanti\_v4\_new.py实现软件内部主要页面（代码量：约3702行）

**5.2.2 前端界面功能设计**

对于反馈用户在前端页面操作功能的文件则是main0.0.12.py文件。该文件设计了两个类，第一个类是LoginWindow，用于定义实现用户登录、注册，页面拖拽等功能，并连接到对应UI文件上，该类中的部分函数定义如图8、9：

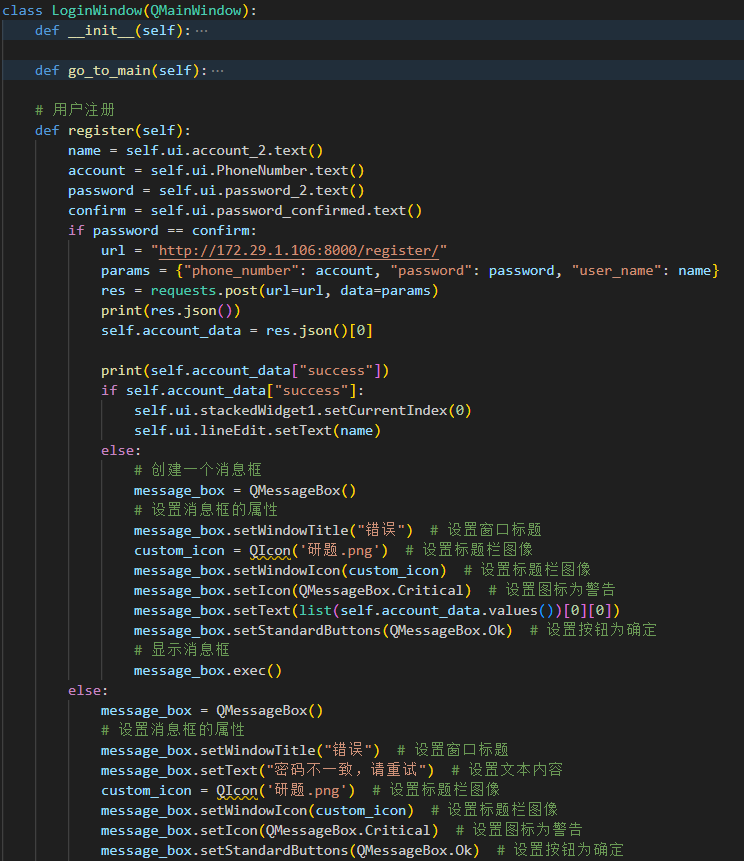


图8-用户注册功能实现

文本

描述已自动生成

图9-页面拖拽功能实现

第二个类是MainWindow，用于实现软件内页面功能，如倒计时实现、页面跳转、题目映射、答案判断等前后端交互功能，如图10：

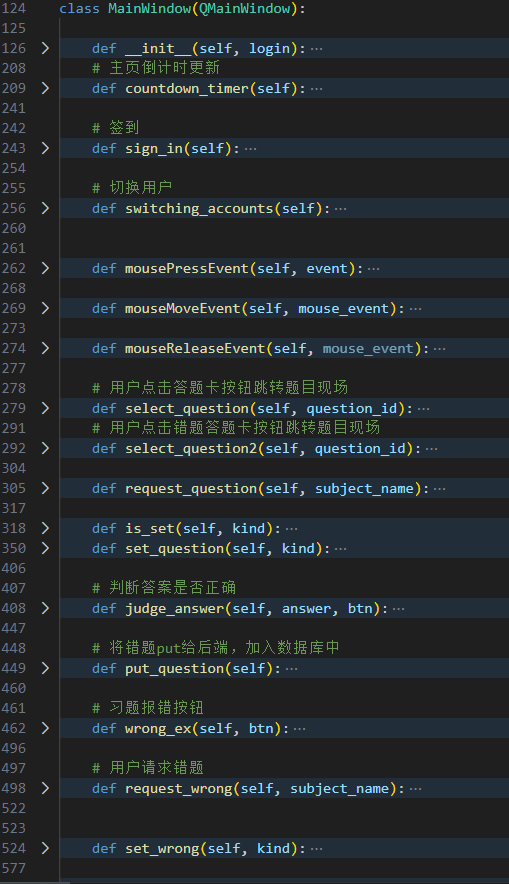


图10-MainWindow中的部分功能函数定义

其中实现了该软件主要功能的做题功能函数如图11、12、13：



图11-练习、错题界面题目跳转功能实现



图12-题目映射、判定对错、错题添加

文本

描述已自动生成

图13-题目映射、判定对错、错题添加（续）

### 5.3后端框架设计

本项目的后端实现主要由python实现，具体分为两大部分，分别是后端服务器与数据库处理，如图14：

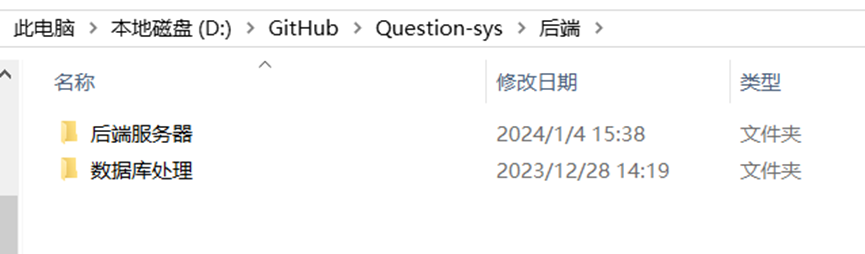


图14-后端结构

**5.3.1 后端服务器实现**

本项目的后端服务器代码文件在SoftwareProjectServer文件夹中，不同版本的文件夹代表了在项目过程中我们服务器不同阶段的更新迭代版本，如图15：

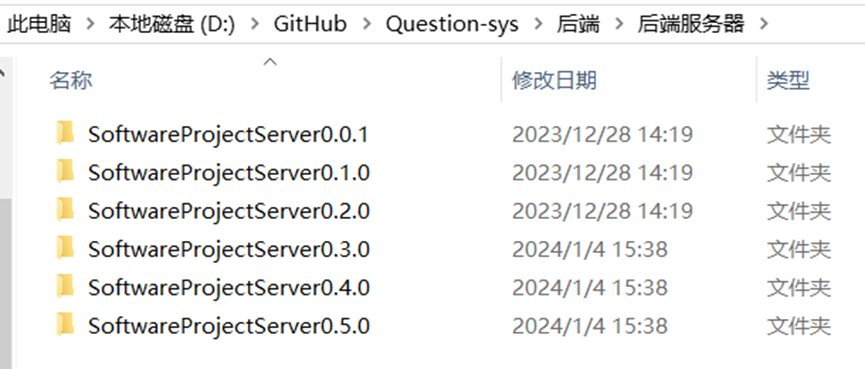


图15-不同版本的后端服务器

本项目最终版本的后端服务器项目代码文件（在最新的github项目提交上面，我们的项目最新后端服务器版本在“SoftwareProjectServer0.6.0/SoftwareProjectServer”文件夹中），如图16：

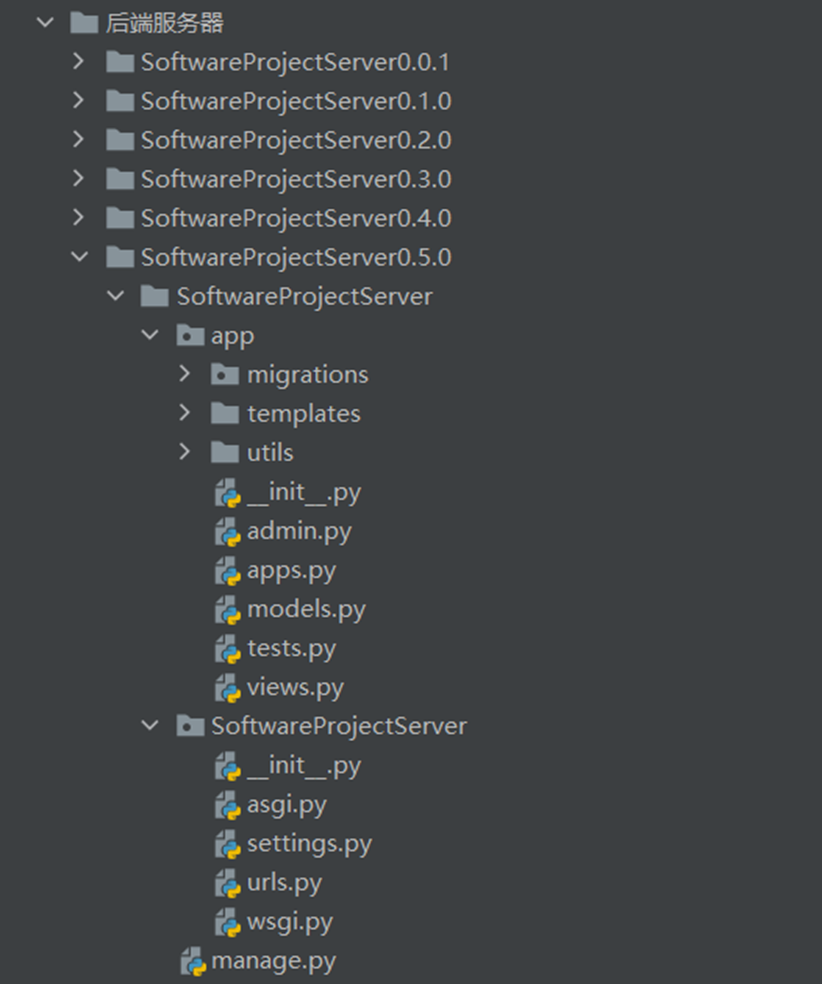


图16-后端服务器代码设计

models.py文件主要定义了用户、练习、错题、报错类，确定软件主要页面的各类接口调用，如图17：

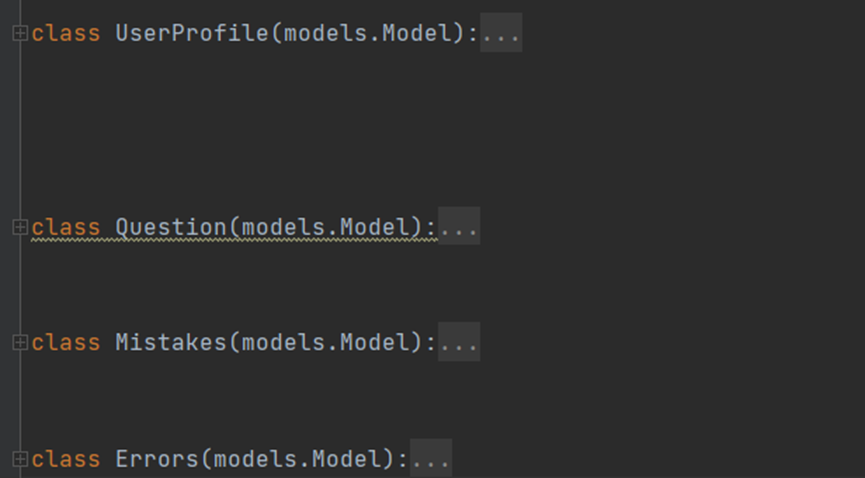


图17-用户、练习、错题、报错类

views.py文件主要是确定登录、注册页面的各类接口调用，以及反馈页面点击操作，实现各类逻辑功能，如图18、19：

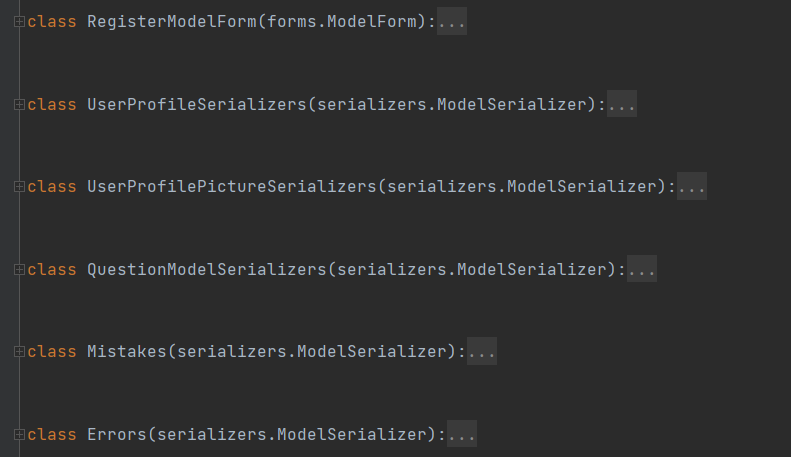


图18-登录、注册页面各类接口调用定义类

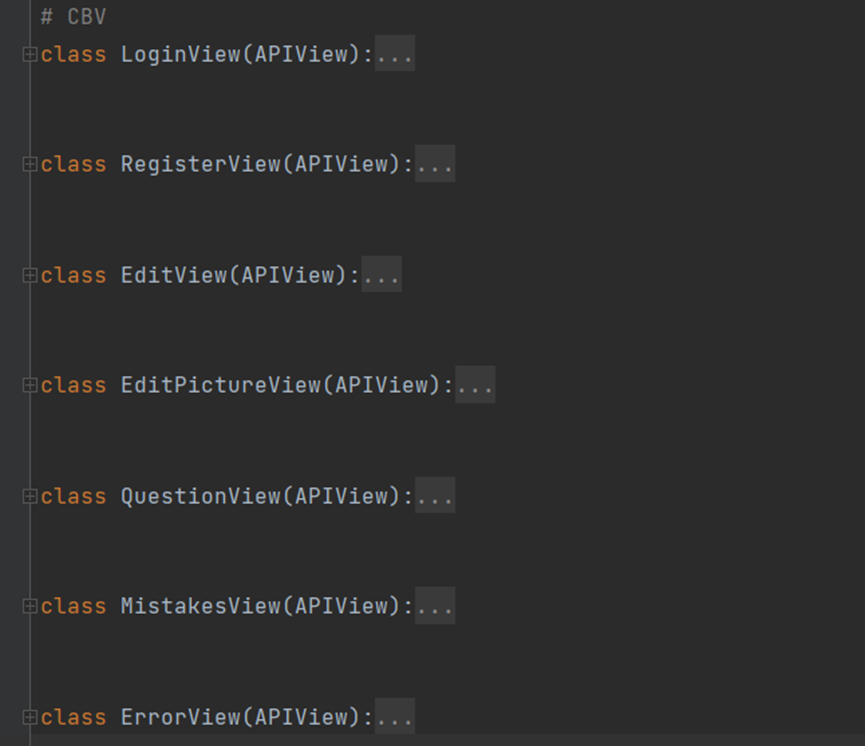


图19-反馈页面点击操作，实现各类逻辑功能类

**5.3.2 数据库实现**

本项目的数据库代码文件在数据库处理文件夹中，不同版本的文件夹代表了在项目过程中我们数据库不同阶段的更新迭代版本（在最新的github项目提交上面，我们的项目最新后数据库版本已更新到0.0.3），如图20：

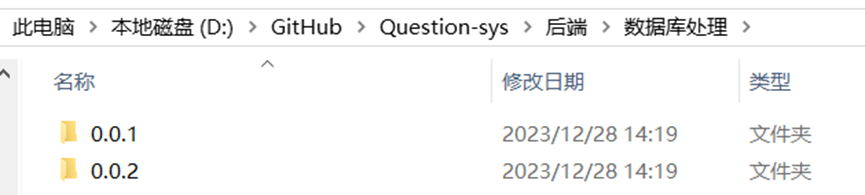


图20-不同版本的数据库文件

我们的数据库实现里面经过最终整合只有一个dataProcessing.py文件实现数据库。首先函数insert实现后端联接数据库，并将题目发送到后端服务器中，如图21：

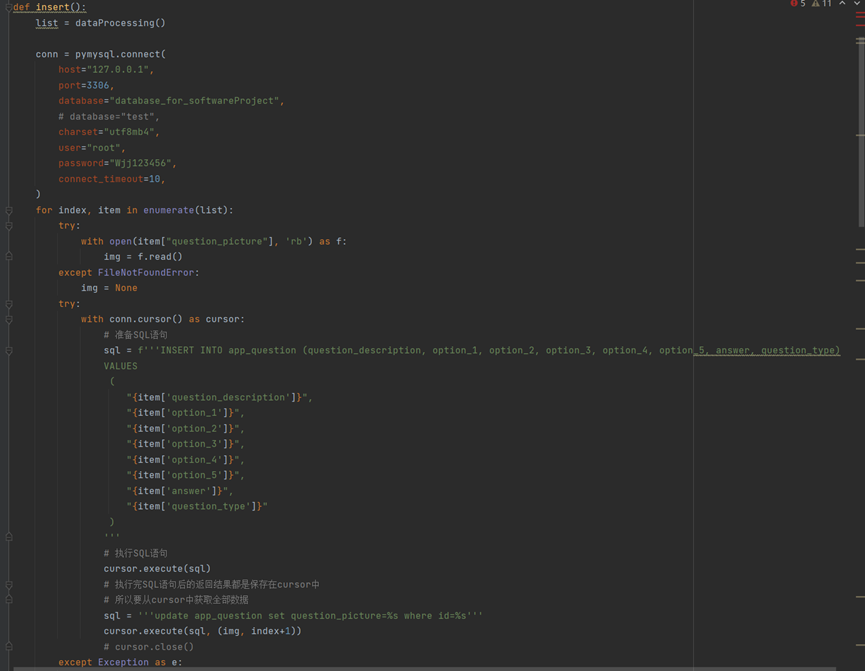


图21-数据库连接后端

接着，函数dataProcessing用于整理匹配题目数据，这里我们有已经经过数据清理后的题目，以.docx的格式进行存储。为了能够实现将文档中的题目自动插入到module.py模块中创建的问题表里，我们需要用到python中的Document库。这里我们生成Document对象，并把题目文件路径传进去，这样我们就可以通过这个对象实现对文档的读写操作。这里定义好了需要操作文档的对象，以及需要用到的正则表达式和一些变量。如图22：

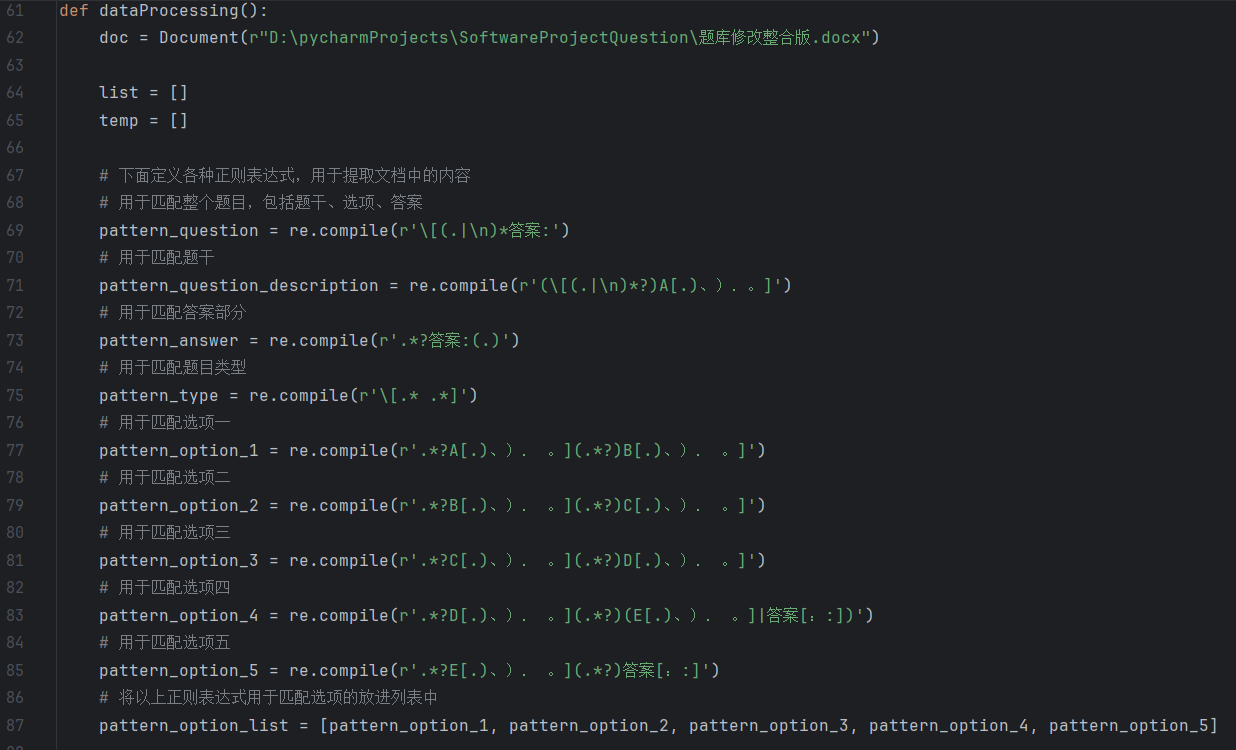


图22-函数dataProcessing（部分1）

再接下来我们开始读文档，通过遍历doc对象中的paragraphs属性可以获取到每个段落，这里需要注意，文档中的文本但凡出现空格或者换行，都会被doc对象视为一个段落。那么我们将段落插入到前面定义好的中间变量列表中，并将列中的字符串元素拼接成新的字符串。对这个字符串进行题目的正则表达式匹配，当匹配到完整的一道题目时（包括题干、选项、答案），进行更细一步的划分。在进一步细分中，先定义一个字典用来存储不同部分，如图23：



图23-函数dataProcessing（部分2）

定义好字典后，开始处理，这里的处理包括题目中的英文单双引号的转义（避免影响后续sql语法的运行）、匹配题干、匹配选项、匹配答案、匹配题型、匹配并拼接出相应题号的图片路径。划分好各个部分后，最后返回列表。遍历刚才返回的列表，拿出里面每个字典中对应的属性放入到表中对应的字段。这样就完成了将题目文档自动提取出各个部分的内容并存储到数据库的功能，如图24、25：



图24-函数dataProcessing（部分3）



图25-函数dataProcessing（部分4）

我们的数据库设计（1）用户表：id（主码）、电话号码、密码、邮箱、头像，表设计及其数据示例如图26、27：

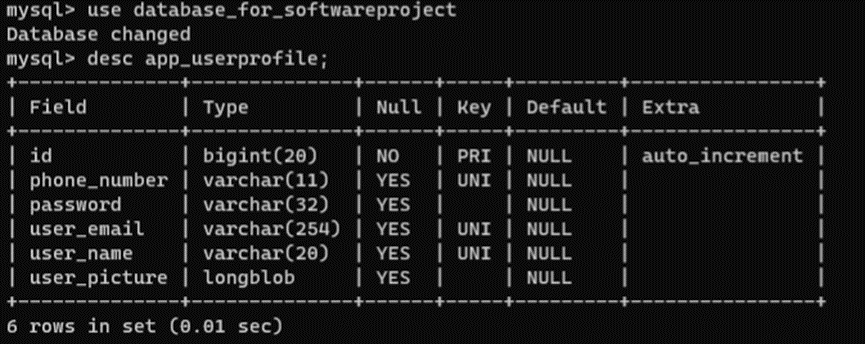


图26-数据库用户表

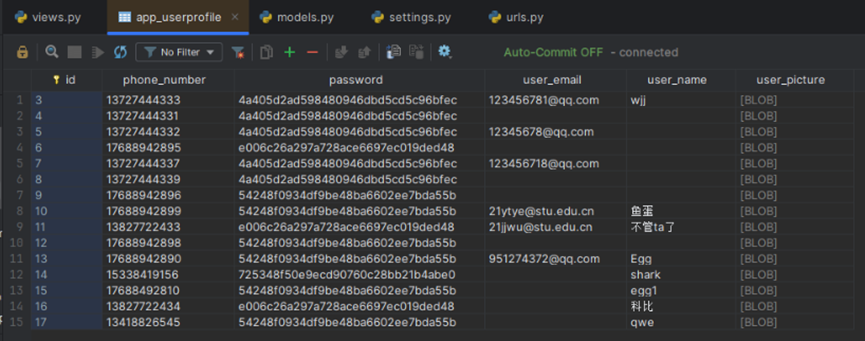


图27-用户表数据

（2）题目表（1张表）：id（主码）、题目内容、题目选项一、题目选项二、题目选项三、题目选项四、正确答案、题目解析（如果有）、题目分类，表设计及其数据示例如图28、29：

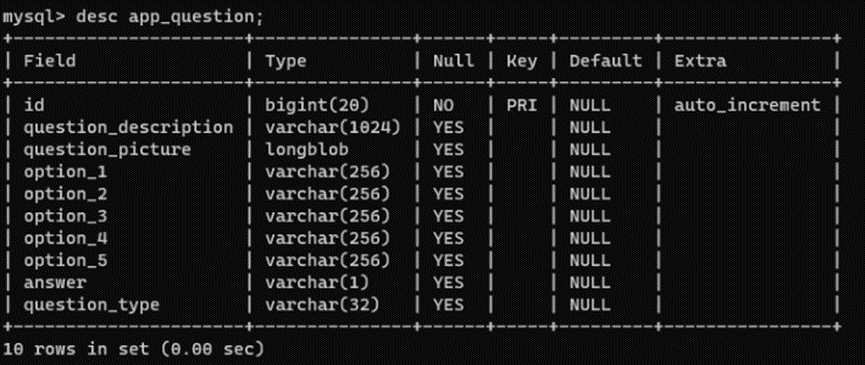


图28-数据库题目表

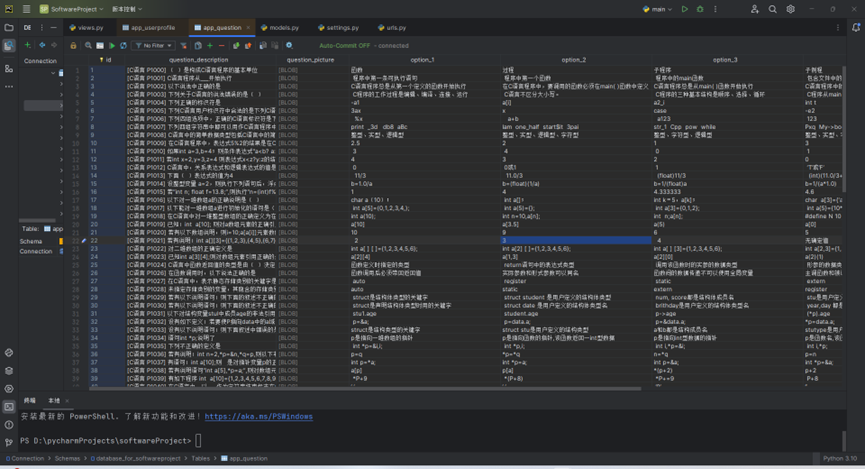


图29-题目表数据

（3）错题表：在题目表的情况下：id（主码）、用户id（外码）、题目id（外码），表设计及其数据示例如图30、31：



图30-数据库错题表

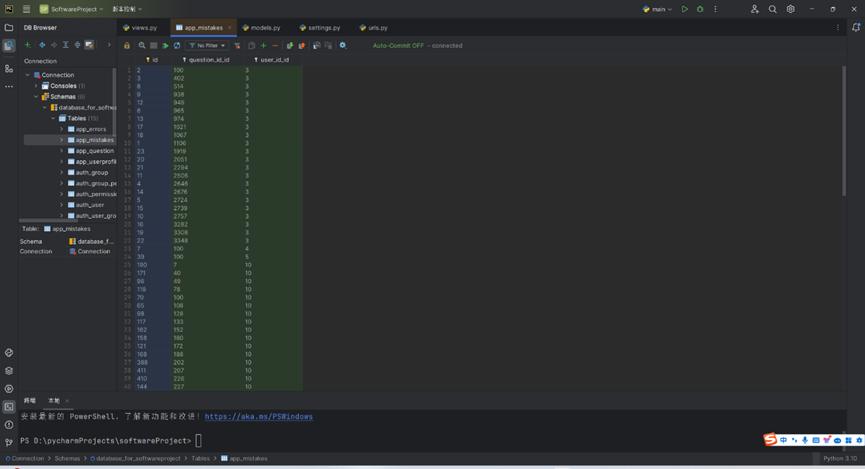


图31-错题表数据

（4）报错表：id（主码）、题目id（外码），表设计及其数据示例如图32、33：



图32-数据库报错表

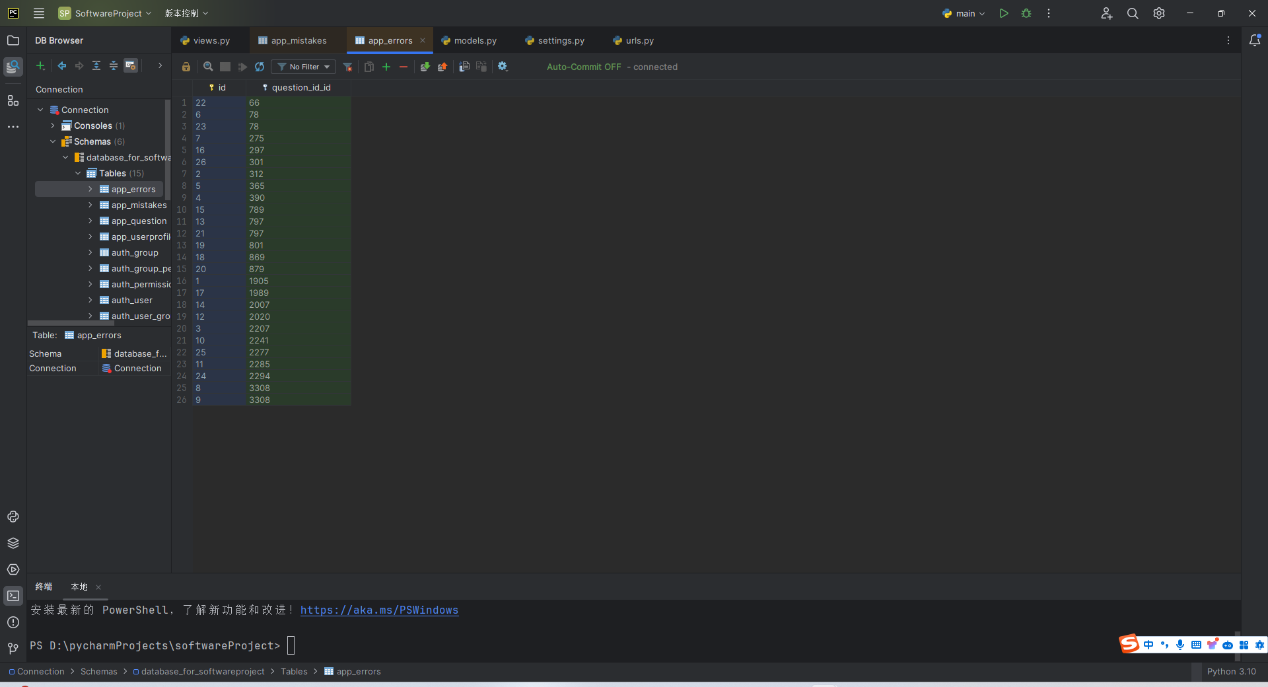


图33-数据库报错数据

### 5.4爬虫实现

爬虫的实现也是依靠python，主要功能是爬取各类题目，存放在数据库中，具体代码实现如图34、35：

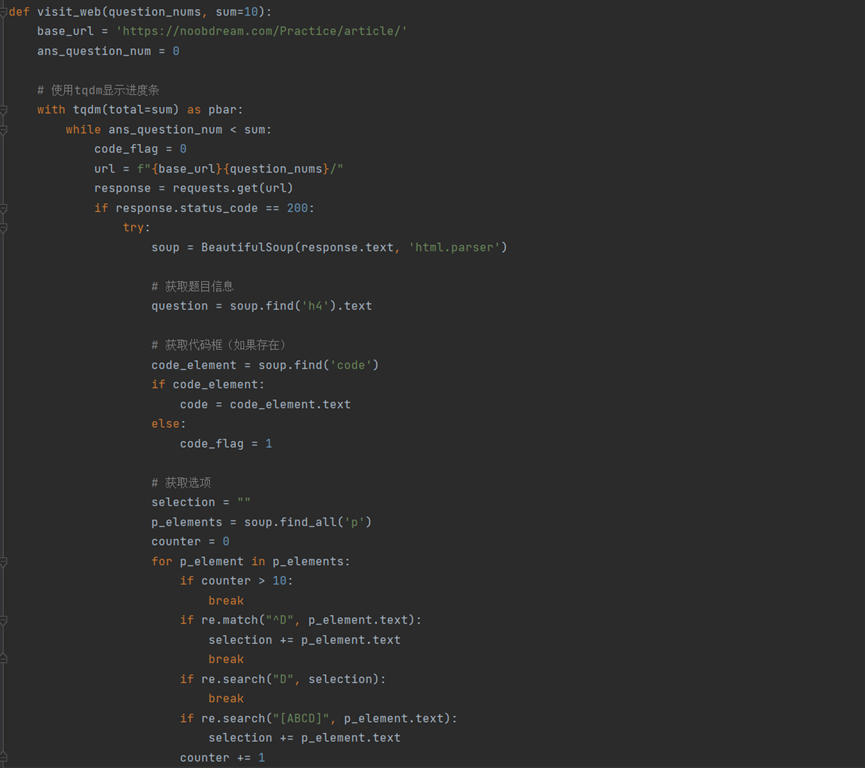


图34-爬虫代码（部分1）



图35-爬虫代码（部分2）

### 5.5其他

对于我们项目的题目，我们不仅设置了常规文本叙述题目，还有带有图片的题目，该功能我们也做了实现，其对应的题目图片展示如图36：

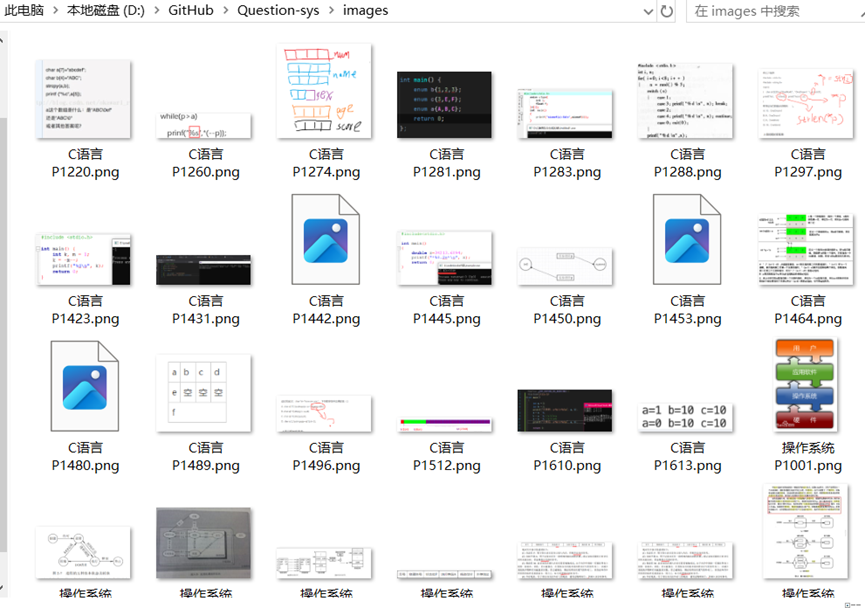


图36-有图题目的图片展示

这就是我们全部的软件设计流程。

## 软件运行结果

### 6.1注册账户

注册页面，用户可以通过手机号进行账号注册，如图37：



图37-注册页面

### 6.2登录界面

登录主页面，用户可通过已有账号进行登录，如图38：



图38-登录页面

### 6.3首页

用户登录成功后则会进入软件主页面，主页面有4个侧边栏，分别是：首页，练习，错题和我的。“首页”这个板块主要展示考研倒计时和签到欢迎，在首页，我们可以看到距离最近的一次考研时间的倒计时，用户也可以进行签到提醒自己刷题，如图39：



图39-首页

### 6.4练习

#### 6.4.1科目选择

“练习”板块分成7大科目（以计算机专业考研为例），每个科目对应不同的题目，用户点击对应的科目可以进入做题页面进行刷题，系统会随机生成25道题目给用户，如图40：



图40-科目选择页面

#### 6.4.2做题页面

在做题页面，用户可以根据不同的分类选择自己需要的题目进行练习。用户选择正确的答案会显示绿色填充，错误的答案会是红色填充，并将正确答案用绿色显示出来。用户做完一道题后点击“下一题”会跳转到下一道题。用户如果不想继续做题了，也可以点击题号左边的返回标志，退回到练习主页面，用户如果点击退出，系统会保存用户的做题进度，如图41、42：



图41-做题页面（1）



图42-做题页面（2）

#### 6.4.3答题卡

用户也可以点击“答题卡”查看自己当前做了哪些题目，正确与否，再点击“返回”会回到之前做题的界面，点击对应题号也可以直接跳转到对应的题目，如图43：



图43-答题卡页面

#### 6.4.4成绩单

用户做过25道题后就可以提交，然后看到自己的做题情况：正确题数，错误题数和未作答题数，然后可以通过点击“继续作答”按钮回到练习主页面进行继续刷题，如图44：



图44-完成页面

### 6.5错题

用户点击错题界面可以看到和练习界面一样的分类布局，用户可以选择需要重复练习的错题类型，直接在错题界面进行刷题。“错题”板块与“练习”板块的布局类似，会记录用户在练习时的错题，也可以重复做错题，如图45、46：



图45-错题科目选择页面

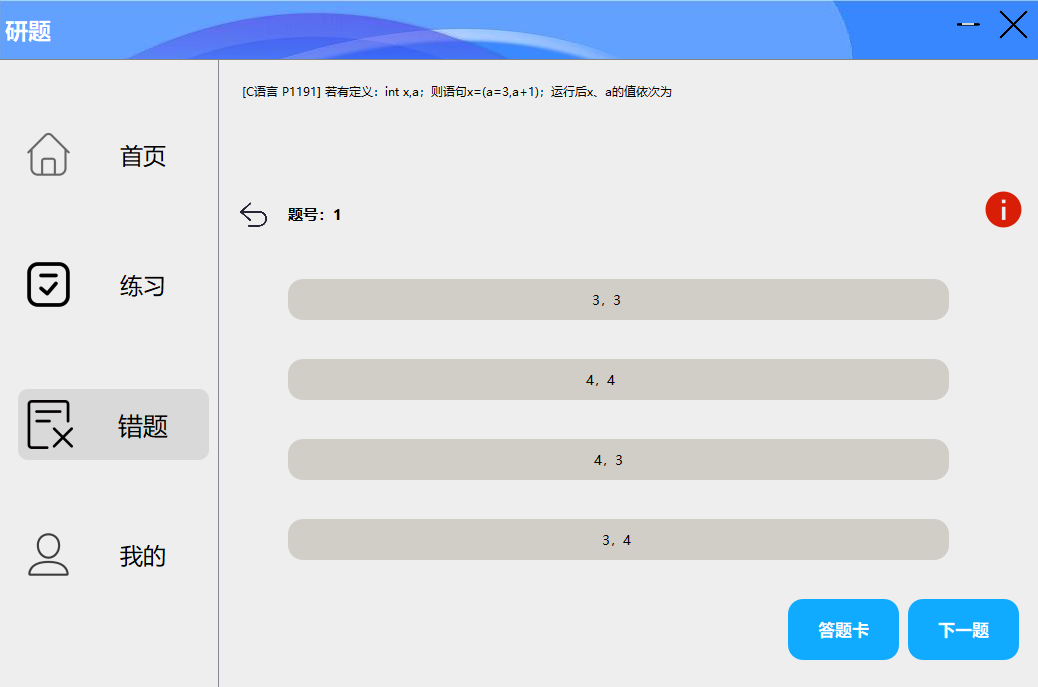


图46-错题做题页面

### 6.6我的

#### 6.6.1功能选择

我的页面主要包含三大板块：个人信息，设置和切换账户，如图47：



图47-我的页面

#### 6.6.2修改个人信息

用户点击“修改个人信息”可以修改用户名、手机号（用于登录）和邮箱，也可以点击头像更改头像，信息修改完毕后点击“保存”即可成功更改信息，如图48：

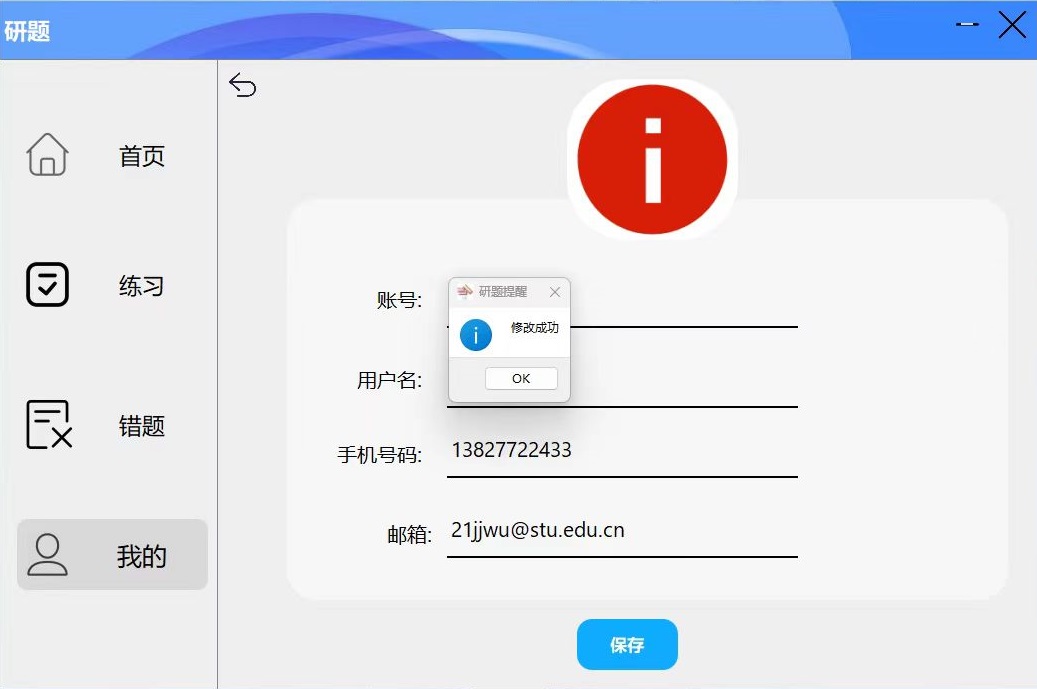


图48-个人信息页面

#### 6.6.3设置

设置部分包含：帮助（readme文档等）、关于（版本信息等）、支持作者（可以给作者打赏），如图49：

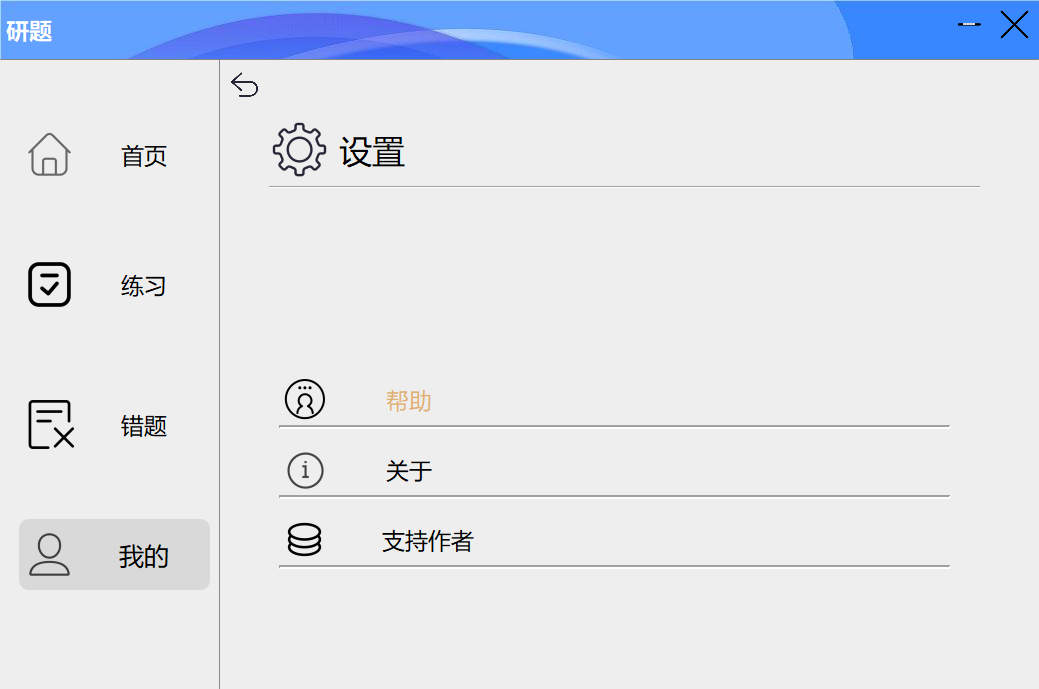


图49-设置页面

用户点击“帮助”会弹出弹框，给出作者联系邮箱，如同50：



图50-帮助功能

用户点击“关于”会弹出版本号提示，如图51：

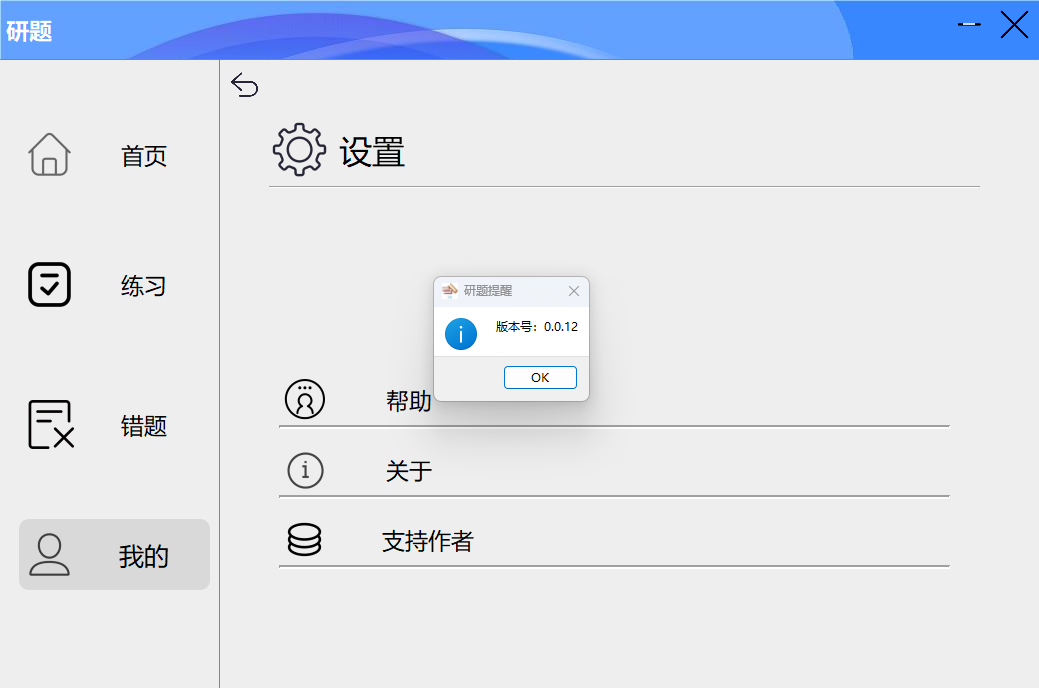


图51-版本号提示

用户点击“支持作者”会弹出收款二维码，可以对作者进行打赏，如图52：



图52-支持作者

#### 6.6.4切换账号

用户点击“切换账号”后，系统会注销当前账号并返回登录页面，如图53、54：



图53-点击切换账号



图54-返回登录页面

## 成员感想

|  |  |
| --- | --- |
| **成员** | 刘星宇 |
| **工作评价** | 作为项目组的组长，我积极组织并主持了项目会议，确保团队成员之间的有效沟通和合作。我具备较强的组织能力和团队管理技巧，在项目过程中始终保持团队的高效运作。我能够清晰记录会议内容并及时总结项目阶段性成果，以确保团队在正确的方向上前进。我在规范定义前后端API接口方面有着丰富的经验，能够有效地沟通和协调前后端开发工作。我深入了解了项目需求，对后端数据进行了有效的清洗和整理，确保数据的质量和准确性。此外，我也能胜任演讲报告的工作，能够清晰地传达项目进展和成果。在未来，我希望能够进一步提升自己的技术能力，特别是针对Python后端开发方面的深入学习和实践。同时，我也计划加强对前端开发的了解，以更好地协调前后端工作。我也会继续提升团队合作能力，以更好地领导和支持团队成员，共同实现项目的目标。 |
| **心得体会** | 在“研题”项目中，我深刻体会到了团队合作的重要性。团队的协作与沟通是项目成功的关键，每个团队成员都发挥了自己的专长和擅长领域，共同完成了一个功能完善、高质量的习题软件。在这个过程中，我学会了倾听团队成员的意见和建议，及时解决问题，以确保项目的顺利进行。同时，通过这个项目，我也不断提高了自己的技术能力和项目管理能力。在与前后端开发人员的合作中，我学到了很多关于API接口规范定义和数据清洗的实践经验。这些经验对我未来的职业发展有着重要的意义，我会继续努力学习，不断进步。“研题”项目让我更加深入地了解了软件工程的实践，也让我更加坚信团队合作和不断学习的重要性。这个项目不仅是技术上的挑战，更是团队合作和个人成长的宝贵经历。 |

|  |  |
| --- | --- |
| **成员** | 陈开枝 |
| **工作评价** | **对组长的评价：**  组长刘星宇积极主动地调动组员，主动组织指导会议，每次会议后都会总结会议内容并为组员解答疑惑。  **对队友的评价：**  跟我一起制作UI的李定坤同学认真负责，一起讨论UI设计并反馈UI存在的问题。前端负责人叶择锬在每次UI制作完之后都会给出反馈，提示UI还需要添加什么，并且对前端组件功能实现的工作也非常认真负责。此外，其他前端成员对待工作也非常仔细认真，后端的同学也一直紧跟着前端的更新内容不断更新软件版本，合作非常融洽。  **对自己的评价：**  本人负责软件内部的UI设计，过程中学到了许多Qt制作以及美化UI的方法和功能。软件制作过程中总体上换了两次UI风格，并且多次调整了背景以及组件颜色，使UI整体看起来更加官方正式。此外，UI设计还添加了许多组件的配图以及点击效果，增加了组件的灵动性。通过本次项目的实践，我不仅学到了许多Qt以及Python的相关知识，也培养了小组之间沟通合作、协同进度的能力。 |
| **心得体会** | 设计UI在具有挑战性的同时也充满了趣味以及设计完成后的成就感，整个项目完成之后我不仅学到了UI设计的新技能也提升了团体合作的能力。 |

|  |  |
| --- | --- |
| **成员** | 黄杰豪 |
| **工作评价** | **对组长的评价：**  组长刘星宇积极主动地调动组员，主动组织会议，每次会议都会进行会议记录，会后都会总结会议内容并为组员解答疑惑、分配内容。  **对队友的评价：**  我们整个项目的8位同学都很认真负责、按时完成自己的工作。前后端分工合理有序。此外，叶择锬主要统筹前端工作，对前端做出改进；吴家杰主要负责后端统筹，进行服务器部署运行。项目后期大家也一直紧跟着前端的更新内容不断更新软件版本，合作非常融洽。  **对自己的评价：**  本人主要负责前期项目构思、初步页面设计、题目数据清洗以及后期项目运行和撰写报告。在过程中我也学会了如何使用Qt以及如何在一个项目小组中合作完成一个软件项目。同时我在参与后期项目运行和报告撰写分析时，我更深刻地了解了整个软件项目的框架和代码结构，进一步丰富了我python写代码的功力。同时，我也积极参加小组项目会议，在会议中积极发表自己的看法。在整个项目开发过程中，我的工作态度也非常认真努力。 |
| **心得体会** | 总体来说通过本次项目的实践，我不仅学到了许多Qt以及Python的相关知识，也培养了小组之间沟通合作、协同进度的能力。 |

|  |  |
| --- | --- |
| **成员** | 郑泽佳 |
| **工作评价** | **对组长的评价：**  组长刘星宇对项目非常用心，积极召集组员开展会议，认真记录会议内容，项目期间合理跟进项目进度，使我们小组项目的进度持续稳定提高，是非常用心的组长。  **对队友的评价：**  我的队友们都非常用心负责，按时认真地完成组长分配的任务，在开展项目会议的过程中积极沟通，通力合作，能够在遇到问题的时候互相帮助解决，具有很强的团队凝聚力。  **对自己的评价：**  我在小组中主要负责的工作为绘制项目的用例图、题目数据的清洗、项目汇报ppt的制作、项目报告的撰写整合。由于本人主攻Web前端方向，对Qt开发不太熟悉，但还是有进行努力学习，对项目开发拥有一定的贡献。对于每次会议我都能积极参加，发表自己的看法，对组长安排的任务，我能够按时认真完成。总体上，我对自己的工作态度和工作完成度非常满意。 |
| **心得体会** | 我在本次项目的开发过程中学到了Qt开发的知识与技能，我在后期撰写报告时，也深入理解了整个项目的框架结构和代码实现，提高了我的python代码编写能力，在小组成员通力合作一起完成一个项目的过程中，我明白了小组合作的重要性，学会了如何团队协作去完成一个大项目。 |

|  |  |
| --- | --- |
| **成员** | 吴家杰 |
| **工作评价** | **对组长的评价：**  组长在项目的整个流程中都表现得非常活跃，积极召开会议。在会议过程中，也充分给予组内成员发言权，并且将很大一部分权利都下放给了组员，例如在进行一些重要决策的时候，组员的话语权可以大于组长，能够很好地团结组内成员。并且每次会议的记录都细心总结，很快就给出会议记录，行动迅速，不拖沓。  **对队友的评价：**  在如此组长的带领下，我的组员能够清晰地知道自己的分工和责任。在会议上也踊跃发言、沟通、交流，使得整个会议流程进行得很顺利。我也能够很容易同步到各个负责部分的信息、进展。当然，落实到项目开发过程中，我得知，在其他组员负责的领域中遇到了重大的开发上的问题，在经过相当长的一段时间后，终于得到解决，可见组内成员的毅力以及积极程度。而最终的成果也没有辜负他们的一片赤诚之心，我们将各自负责的内容融合到一起的时候就免去了不少麻烦。  **对自己的评价：**  在高行动力的组长的领导下，以及积极的组员氛围中，我所分配到的是后端服务器以及数据库设计开发的工作。在初步学习了解过后，从初步搭建原型开始，到中期基本成型，再到后期成熟阶段，以及在与前端连通的过程中遇到了，不少的麻烦。但是在我与组员共同的努力下，都得到解决了。也算是没有辜负我的组员。 |
| **心得体会** | 在这个项目中，团队协作是不可或缺的，正是因为有前后端的高效协作，项目才能顺利进行。通过这次项目，让我认知到了我所学的内容的有限性，需要保持持续学习的心态面对未来的路。尽管我的工作是后端开发为主，不与用户直接沟通，但在开发过程中，前端开发的组员就是我的用户，我需要满足，前端开发对后端提出的需求，使我们的项目更加完善。在开发中，代码的质量和可维护性十分重要，良好的代码风格、书写规范以及注释能够提高代码的阅读性。 |

|  |  |
| --- | --- |
| **成员** | 李兵 |
| **工作评价** | **对组长的评价：**  组长刘星宇展现了积极的组织和指导能力。他在项目中积极调动团队，组织并引导会议，总结会议内容并为团队解答疑惑。他的领导方式有助于确保团队的高效协作和顺畅进行。  **对队友的评价：**  整个团队在UI制作、前后端开发等方面展现了高度的责任感和专业性。大家保持着良好的沟通合作，这使得软件版本得到及时更新和整合，工作效率得到了提升。  **对自己的评价：**  在项目中，我负责了在网址中爬取题目、前期需求分析中类图的绘制以及后期项目README文档的撰写工作。我在这些任务中展现了深入理解和专业的工作态度，提供了清晰、准确且易于理解的文档，为整个项目提供了必要的支持和指导。 |
| **心得体会** | 在项目的开展过程中，我们小组从前期的茫然过度到中后期的分工明确，拥有一个良好的团队是必不可少的，庆幸的是我是这个良好团队的一份子。在项目开展过程中，我们小组开了几次项目会议，通过大家的头脑风暴，我们很快地确定了项目的选题和个人的分工。作为负责题库的爬取人员，我深知自己能否获得优良的题库在一定程度上较大地影响到项目的推进，如何格式化、准确地获取题目，是我要思考的问题。为此我学习了爬虫、http协议、HTML、xPath定位、selenium库方面与爬虫相关的技术以解决我所遇到的难题。同时我也懂得了以后作为互联网相关领域的从业者，最需要获取的品质是持续学习的能力。互联网相关技术的迭代非常的快，如果不能够很快地上手新技术，你的个人竞争力也会越来越低，工作效率同理。 |

|  |  |
| --- | --- |
| **成员** | 李定坤 |
| **工作评价** | **对组长的评价：**  会议组织积极，有力保障了项目的顺利进行.会议氛围轻松，提供了集思广益的平台积极跟进项目的进程，确保了项目的落地和实现  **对队友的评价：**  首先，我要郑重表扬一下我们的前端同事陈开枝和叶择锬等。他们的设计和用户体验能力令人印象深刻。通过巧妙的界面设计和顺畅的用户操作流程，他们赋予了软件以独特的魅力。这种视觉和交互的优势，为整个项目增色不少。而我们的后端同事同样也是团队的支柱。他们在数据处理和系统架构方面的专业知识令人叹为观止。通过构建高效且稳定的后端系统，他们保障了软件的性能和可靠性。在面对各种技术挑战时，他们总是能够轻松应对，为项目的成功付出了良多努力。  **对自己的评价：**  界面设计和实现： 我在项目中承担了设计软件界面的部分责任，包括登录注册界面和主页面等。通过使用PyQt5和Python，我成功创建了精美且直观的界面，使用户能够轻松地与软件进行交互。我的界面堆叠布局得当，呈现出良好的视觉效果，增强了用户体验。  页面跳转和导航： 我不仅设计了界面，还实现了页面之间的无缝跳转和导航功能。这使得用户可以轻松地从登录注册界面切换到主页面，实现了用户友好的交互流程。通过编写相应的逻辑代码，我确保了页面之间的平稳过渡和数据传递。  技术应用： 我充分利用了PyQt5和Python的强大功能，包括窗体设计、信号与槽机制、事件处理等，以确保界面的高度可定制性和互动性。我对这些工具的熟练应用帮助我有效地实现了项目的目标。  团队合作和沟通： 在小组中，我积极与队友交流和合作。我愿意倾听队友的建议和反馈，不断改进和优化界面设计，以确保它们符合项目需求和用户期望。我能够与团队协作，确保整个项目能够顺利推进。  总的来说，我对自己在软件界面设计和实现方面的表现感到自豪。我成功地将我的创意转化为可用的界面，并通过技术工具的巧妙应用，提高了用户体验。在团队中，我展现了积极的合作精神，愿意与他人共同努力，以确保项目的成功。这次经验使我更加自信，我期待在未来的项目中继续发挥我的设计和开发技能。 |
| **心得体会** | 参与这个软件开发项目让我获得了许多宝贵的感受和经验，以下是我在项目中的心得体会：  学习和成长： 这个项目让我学到了很多新知识和技能。我不仅掌握了PyQt5和Python的使用，还深入了解了界面设计原则和用户体验。通过不断学习和实践，我的编程和设计能力得到了显著的提升。  创造力的发挥： 设计软件界面是一个创造性的过程，我喜欢思考如何将用户需求转化为具体的界面元素和交互流程。这个过程允许我发挥创造力，为用户提供更好的体验。  团队协作： 在小组中工作是一种宝贵的经验。我学会了与不同背景和技能的队友协作，有效地分工合作，共同追求项目的成功。团队合作不仅加速了项目的进展，还让我从他人那里学到了许多。  挑战和解决问题： 在项目中，我们经常面临各种挑战和问题，如界面设计的复杂性、技术难题和时间压力。这些挑战激发了我的解决问题能力，让我不断寻找创新的方法来克服困难。  满足感： 当看到项目逐渐完善，用户可以愉快地使用我们设计的界面时，我感到非常满足。这种成就感激励着我继续努力，追求更高的目标。  总的来说，参与这个软件开发项目是一次宝贵的经历。它不仅增强了我的技术和团队合作能力，还让我更深刻地理解了软件开发的挑战和乐趣。我期待着将这些感受和经验应用到未来的项目中，继续成长和发展。这个过程不仅是一次技术上的提升，也是一次个人成长的旅程。 |

|  |  |
| --- | --- |
| **成员** | 叶择锬 |
| **工作评价** | **对组长的评价：**  组长刘星宇认真负责，跟进项目进度。在每次阶段工作后，会积极组织进行项目会议，并记录会议内容，在会议结束之后会总结项目目前进度以及下阶段任务，调度组内组员，分派任务，帮助实现小组内各成员之间的协作。  **对队友的评价：**  其他组员们都很优秀，李兵同学在项目前期很积极主动学习了爬虫并爬取了多个网站的考研题目数据。黄杰豪、郑泽佳和刘星宇同学也在短短两天内就把数据清理完成并与后端吴家杰同学对接；吴家杰同学建立数据库并积极检查数据搭建了服务器，与前端对接，即时反馈。李定坤、陈开枝同学根据需求分析做出了很多版本的UI界面，并美化了界面。很快我们的软件就成型了，通过我们前端小组成员的不断测试并debug做出了最终版研题。这个过程比较顺利，与其他组员间的沟通协作通畅，各组员对自己负责的部分很认真并经常为自己揽任务，感觉处在一个很好的项目组内。  **对自己的评价：**  在本次项目中，我主要负责前端QT软件的开发编写响应函数，与进行前端与后端的顺畅对接，确保数据的有效交换和处理，参与了测试和debug过程，保证产品质量，修复潜在的问题和漏洞。通过与团队成员的交流和合作，我能够有效地理解他们的需求。在编写响应函数的过程中，注重代码的效率和可维护性。在测试和debug阶段，我对解决问题充满了热情，对问题进行了全面而细致的分析，寻找并解决了各种潜在的错误。提高了我不仅是技术方面的知识和经验，还有团队合作和沟通方面的能力。 |
| **心得体会** | 在这个项目中，我扮演了前端与后端的桥梁角色，这是一次全方位的技术挑战。从编写QT组件的响应函数到与后端团队紧密合作，我深刻体会到了前后端协作的重要性。这种综合性的工作经历让我更加全面地理解了软件开发的生态系统，也提升了我的沟通能力和问题解决能力。在这个过程中，度过了一次快节奏的开发环境，学到了如何平衡多个任务并且保持高效率以及如何在多个团队和技术领域中协调工作，这对于我的职业发展是一次极为宝贵的经验。很庆幸能在氛围与效率都很好的团队使我受益良多。 |

## 项目总结

本次我们的小组项目完成的非常好，不论是从前期确定题目、小组分工还是中后期的代码编写、功能实现，我们项目的进度总体上都处于领先，也非常感谢老师在中期汇报时对我们小组的肯定。我们的项目功能实现完整，软件界面美观，bug处理完全，做到了项目整体几乎没有bug。项目符合软件设计流程，较为完整地实现了需求。同时，项目小组成员都非常认真努力地完成工作，对工作内容认真负责，积极开展会议，合作商讨互助，解决在开发过程中遇到的问题。但是，我们小组的项目仍然有些许不足，并没能够实现管理系统，这是我们的一大不足。至此，我们的项目圆满完成，再次感谢老师的教导。

## 其他文件引导

在github上的文件相对路径：

编码规范文档：python\_language\_rules.rst

版本管理文档：版本管理/研题-版本管理.pdf

会议记录文档：会议记录

README文档：README.md

项目运行图合集：项目运行图

项目演示视频：演示视频.mp4