**二手书处理系统的设计与思考**

——程设大作业报告

1. 简介

时间临近新一届学长学姐的毕业季，毕业生们是时候处理自己无需用到的书籍与教辅资料了。受此需求的启发，我们小组想到，是否可以创建一个二手书交易平台，实现资源的高效与多次利用，构建同学们之间合作共赢的平台？由此，历经所有成员的长期努力，我们初步建立了满足上述要求的平台，并希望这个平台可以发挥其作用。

1. 详细功能介绍

该平台功能大致可以分为以下若干部分：

1. 用户注册与登录。平台本身含有用户类，里面含有用户的UID，账号，密码等变量。用户只需在注册界面填写自己的账号，密码与联系方式即可成功完成注册，UID由系统自动分配。
2. 用户上传想要出售的书籍。平台本身含有书本类，分别为书籍匹配其id，出售状态，用户只需上传书的名称，破损程度，书籍分类与预期价格即可成功上传整套书籍信息，必要时我们配备了上传图片的功能，这使得买家可以对书籍有一个直观清晰的了解。
3. 用户购买自己心仪的书籍，这又可以分为由推荐渠道获得与由搜索渠道获得两部分。首先为推荐渠道。程序主界面会根据用户的购书记录为用户推荐其可能感兴趣的二手书，如果用户对此感到不满意，也可以手动进行“换一批”的操作。用户可以直接由系统推荐查看详细信息，再通过卖家提供的联系方式以进行购买。然后为搜索渠道。用户可经由系统的搜索页面按名称、书籍id，书籍tag等多种方式搜索自己心仪的书籍，在搜索结果中，双击该条结果即可查看书籍的详细信息。
4. 此外，我们连入了联网的deepseek以更好地支持用户来获取更佳的使用体验。用户可以通过程序接入的deepseek功能以询问deepseek有关书籍更详细的信息作为是否购买本书的依据。
5. 程序含有的模块及其实现

简单来说，程序显然可以分为前端与后端两个模块。以下详细介绍两个大模块下的子模块与其实现方法。

前端主要包含Windows.py,search\_ui.py,UI.py三个部分，均主要通过PyQt5来实现前端界面的构建与呈现。在Windows.py中，我们搭建了程序含有的多个主要界面，如登录界面控制用户登录与注册，主界面展示推荐书籍与搜索栏，买书和卖书的专题界面以实现用户主要需求，用户中心界面以方便用户查看个人信息。通过对于堆叠窗口(Stacked Widget)的使用，我们得以在不同的类（即不同的窗口）之间实现切换，而非打开一个窗口之后需要手动关闭上一个窗口。在UI.py中，我们详细补充了各页面的具体信息以进一步完善前端窗口。在search\_ui.py中，我们对搜索界面进行了处理和优化，并通过双击顺利连接了搜索结果与结果展示两个界面之间的切换。

后端主要包含Book.py,User.py,recommend.py,search.py四个程序。其中，Book,User分别为两个大类以存储用户与书籍信息。我们使用了对于python来说非常简洁有效的json库以方便地存储与读取信息，从而优化了后续进程的困难度。在recommend.py中，我们实现了对于书籍的推荐算法，推荐机制采用朴素而有效的cosine\_distance以计算两本书的相关度，具体实现为在该用户购书记录以外的所有书中找出与用户已购买书籍的相关度最高的前n本以进行展示。在search.py中，我们实现了对于多种要求进行匹配的功能，具体实现方法为通过defflib.SequenceMatcher来具体区分两个字符串的相似度以从高至低输出前n个匹配度最高的结果。

1. 小组分工情况

朱正同学几乎仅凭一己之力完成了所有前端代码的开发，主界面开发最需要的main.py, Windows.py, UI.py,均出自他手。他在程序后期还负责将各个板块搭建起来。此外，他还负责了接入deepseek这一本该由后端完成的任务，可谓团队的中流砥柱。在此致谢！

于子淇同学负责了部分后端开发，小部分前端开发以及录屏，数据测试等繁琐的工作。他独立完成了Book.py, search.py, ，录入book.json和image,对初版recommend进行测试修改等，对search\_ui等程序的贡献也居功至伟。感谢他不辞辛劳的付出！

陈泽羽同学主要负责部分后端开发，小部分前端开发，并负责撰写此篇报告文档、补充测试程序等工作。他独立完成了User类，并极具创造性地开发了recommend.py的所有算法，录入user.json,补充测试初版search\_ui感谢他的辛苦付出和卓越贡献！

注：由于后编写代码的需要，之前编写的代码时常被修改与覆盖，具体明晰哪一块归结至谁较为困难。因此，我们以上述程序的第一编写人划分了分工情况。

1. 项目总结与反思

总体而言，“二手书交易平台”项目基本实现了预期功能，此外还附加了一些团队成员的巧思，也不乏亮点。编者个人认为，总体上，这个项目已经达到了我们的预期要求。然而，项目本身确实存在一些不足值得思考。先不论UI设计过于朴素是否为缺陷，程序很多部分值得改进，比如常见的按下enter键代替确定键的功能应当加入，recommend算法计算相似度可以由“用户是否点击过”转变为更好的“用户点击次数”以提高推荐准确性，search算法也可以考虑实现更方便的模糊搜索与个性化搜索。一个程序生来注定是不完美的，而只有正视与反思这些不完美才能使我们进步。期待我们可以做出更好的作品！