**迭代评估报告**

　　　　　　　　　　　　　　　　　　评估日期：2022.12.13

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 5 | 项目名称 | 白云社区 |
| 迭代名称 | 技术原型迭代 | 实际起止日期 | 2022.11.7-2022.12.13 |
| 任务达成情况：（完成的任务、实现的功能、进度、质量等）   1. 采用前后端分离式Web开发，确定了项目的物理架构、逻辑架构、技术架构并编写了软件架构文档。  * 物理架构：B/S架构 * Web browser * Web Server * Database Server * 逻辑架构：3-tiers * 客户层 * 用户界面层：首页、案例首页、问答首页、个人中心 * 应用层 * 服务层：API、Web * 业务逻辑层：内容管理、用户管理 * 数据层 * 数据访问层：用户信息、案例、问题、回答、标签 * 数据储存：数据库、文件 * 技术架构 * 前端：Element-Plus+Vue3+axios * 后端：Spring Boot：Apache Shiro、Spring Data JPA * 数据库：MySQL  1. 选定了Google编程规范并按照规范实现技术原型  * 搭建软件架构：配置Spring Boot、MySQL * 实现案例的增删改查： * 实现案例的发布和编辑，支持富文本和Markdown两种编辑方式。 * 实现案例的删除，以及案例的开放或关闭，案例关闭后除了发布者外其他用户无法浏览。 * 实现案例与用户所添加的标签的对应关系。 * 实现问答的增删改查： * 实现问答的发布和编辑，支持富文本和Markdown两种编辑方式。 * 实现问答的删除以及问题的开放或关闭，问答关闭后其他用户无法回答该问题。 * 实现问答与用户所添加的标签的对应关系。 * 实现回答与其评论的对应关系，并且实现了评论的发布和删除功能。 * 实现用户信息的增删改查： * 实现了用户的注册和登录，支持用户修改个人信息（头像、个性签名、身高体重、患病信息等）。 * 实现了用户查看发布的案例、问答的功能。 * 实现用户之间的关注、私信功能。  1. 设计了推荐、监测算法，并初步实现了模糊搜索  * 利用Spring Data JPA提供的接口实现对问题、案例和用户的模糊搜索 * 内容检测算法： * 图片检测神经网络结构已设计完毕 * 文字检测神经网络结构已设计完毕 * 推荐算法：HRL-Rec | | | |
| 评审/测试的结果：（执行了哪些评审和测试？评审和测试的结果如何？）   1. 对案例的增删改查、问答的增删改查进行了评审和测试，不断调整前后端代码及数据库的结构最终的达到不错的鲁棒性。 2. 对图片上传、用户信息上传、私信功能进行了评审和测试，发现图片上传的组件接口的上传方式与默认方式不一致后进行了调整，目前这些功能均可正常使用。 3. 对搜索的结果进行评审和测试，确定了搜索结果的正确性。 | | | |
| 问题、变更和返工：（遇到的问题、发生的变更、是否需要返工等）   1. 前期在静态界面设计时由于没有考虑后端而漏掉了一些前端本该有的细节和界面，于是需要更新许多前端的内容。 2. 初期在设计数据库时考虑不够周到，漏掉一些必要的字段，导致迭代后期前后端代码需要频繁的更改和返工。 3. 对前后端的用作不够熟悉，导致在debug时缺乏有效的手段和明确的思路，往往需要上网花费大量时间查找资料。 4. 负责不同部分的成员的之间的沟通不足，导致在对接时接口出现差异，需要返工重新提供接口。 5. 目前的案例或问答的访问还没有加入权限设置，可以直接通过url访问，后续将会加上后端的权限验证来保证网站和信息的安全。 6. 原来在写静态界面时由于未考虑后续的渲染问题，个别组件需要返工或重做，导致界面的美观性有所下降，后续将逐步美化。 | | | |
| 经验和教训：   1. 数据库在创建的时候就应该充分考虑到该数据库可能需要的所有字段和数据类型，提前做好部署，防止后续数据库变更导致后端、前端整体出现变化。 2. 要充分考虑到不同数据之间的关联，创建好相关表单，保证程序运行的高效稳定。 3. 继续学习Spring Boot和Vue，提高工作效率，减少编码过程中由于不熟悉技术消耗的时间。 4. 组员之间加强沟通，对于当前遇到的问题及时沟通，共同解决问题，避免信息差造成的返工。 | | | |