

软件需求工程作业一 项目规划

庄毅非 何迪 李予谦 应凌凯 刘奕骁

2022 年 10 月 9 日

1 确定队伍名

经过我们小组内部的讨论，大家一致决定 fiSSure 为我们的小组名

2 代码选择及优缺点分析

2.1 项目选择

经过小组内的讨论，大家最后一致选择 doubleC 作为我们的 v1.0 选择版本

2.2 前端代码分析

2.2.1 优点

1. 模块化分明。各类配置文件,css 文件,js 文件分类清晰。各类命名清晰，文件函数大驼峰，变量小驼峰，可读性高。各种组件都由单一文件进行维护，封装性高。
2. 使用了 **Material Design** UI 库，整体观感简洁干净。
3. 各类状态清晰明确，分开处理。代码中根据状态机进行 UI 的灵活转换，便于维护。
4. 网站设计规范程度高，例如包含 **robots.txt**。
5. 组件间切换顺畅，没有停顿感。
6. 页面扩展性较高，组件间距大小会自行随页面变化而变化。

2.2.2 缺点

1. 大部分代码缺少注释，不便于后期维护。
2. 部分功能代码没有实装，例如 **TimeLine**，例如 **Commit number** 的获取等等。
3. 部分组件存在样式问题，例如 **TimeLine** 与 **Language** 模块高度未对齐。刷新动画位置错误等等。
4. 在没有做页面跳转的情况下，重定向的提示 UI 与 UI 的重新转换排列存在时间差异。
5. 侧边栏似乎与页面没有一致性，较为独立，并不会随着页面的变化为自适应。
6. 部分功能引导性较弱，未给出详细的操作方法和注释。

2.3 后端代码分析

2.3.1 优点

1. 模块分配清晰，将后端分为专门处理数据库相关操作的函数都封装在 `db` 和 `models` 文件夹中，其他模块只要导入对应的对象就可以执行其上的对象方法，实现对应的功能。
2. 使用 `mongodb` 进行数据表的定义，不仅容易找到属性对应的 `constraints`，也清晰直观。
3. 使用 `express` 框架，请求均经过注册的多层中间件，包括独立的异常处理以及 404 路由处理等，方便后续扩展使用。

2.3.2 缺点

1. 大部分代码缺少注释，不便于后期维护。
2. 格式不统一，包括换行情况和是否使用分号等。
3. 配置信息在各个文件写得比较零散，以及过于依赖环境变量。
4. 部分配置已经过期，并且将安全密钥 `github_assess_token` 写死在代码中，存在一定的安全隐患。

3 队伍角色和责任

表 1: 角色分工表

序号	角色	职责	人员
1	队长 项目经理	在预算范围内按时优质地领导项目小组完成全部项目工作内容, 并使客户满意。	庄毅非
2	产品经理	负责市场调查并根据用户的需求, 确定开发何种产品, 选择何种技术、商业模式等, 根据产品的生命周期, 协调研发、营销、运营等, 确定和组织实施相应的产品策略。	李予谦
3	设计总监	建立系统框架; 数据库设计; 概要设计; 参加技术评审;	刘奕骁
4	测试经理	组织编写测试计划和测试方案, 组织系统测试; 参加技术评审;	应凌凯
5	美工	设计网站原型	何迪
6	质量经理	带领软件质量监督组成员制定质量保证计划, 对监督组反映的质量问题进行汇总与产品经理、项目经理进行交流, 当新的问题出现时最终由质量经理决定处理方式。	全体
7	开发人员	负责进行编码工作与单元测试, 进行系统集成, 及时解决测试时出现的问题	全体
8	测试人员	编写测试方案和测试用例, 进行系统测试, 向开发组反馈 BUG。	全体
9	软件质量监督	实时对质量经理以及项目经理提供项目进度与项目实际开发时的差异提出报告, 指出差异原因和改进方法。	全体

4 项目管理策略

通过一系列讨论, 大家确定了如下的项目管理方式。

- 文档书写: 通过 overleaf LaTeX 进行文档的书写, 保证格式和样式统一
- 代码托管: 通过 github 进行代码托管和版本控制, 成员都通过 pull request 进行代码和文档的提交, 由对应的 reviewer 进行代码检查, 并决定是否进行 merge。
- 组会时间及形式: 确定组会时间为每周二晚上 7:00 左右, 形式为线下, 主持人为庄毅非, 每次预计 30 分钟。
- 周会记录方式: 每周在会中及时记录, 在会后进行对应内容的补充

5 项目计划

表 2: 项目计划时间表

项目阶段	持续时间	负责人	主要工作	输出内容
项目启动	2022.9.27- 2022.10.07	庄毅非	进行项目可行性分析, 制定项目计划	确定项目实现的技术栈以及大致方向
需求分析	2022.10.08- 2022.11.20	李予谦	确定系统运行环境, 确定系统功能及性能, 建立系统逻辑模型	分析项目需求, 确定需要实现的功能
系统设计	2022.11.20- 2022.12.01	刘奕骁	进行系统设计	搭建系统开发架构, 方便进行开发; 设计系统样式
编程实现	2022.12.02- 2022.12.22	何迪	进行系统编码	实现并部署网站
需求维护	2022.11.30- 2022.12.29	应凌凯	进行需求变更控制	把握需求变动, 及时向开发组反馈
系统测试	2022.12.22- 2023.01.01	全体	进行系统测试, 项目总结	测试最终成品项目, 保证验收通过