代码管理平台Code Cube

前景文档

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 13/Mar/2024 | 1.0 | 增加简介、定位、涉众和用户说明、产品概述、特性、约束、质量范围、优先级、其它需求和用户手册。 | 王晨丞，龚畅玮，项百川 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 4

1.1 目的 4

1.2 范围 4

1.3 定义、首字母缩写词和缩略语 4

1.4 参考资料 4

2. 定位 4

2.1 商机 4

2.2 问题说明 4

2.3 产品定位说明 5

3. 涉众和用户说明 5

3.1 市场统计 5

3.2 涉众概要 5

3.3 用户概要 6

3.4 关键的涉众/用户需要 6

3.5 备选方案和竞争 7

3.5.1 GitHub 7

3.5.2 GitLab 7

5. 产品特性 9

5.1 注册和登录 9

5.2 创建个人资料页面 9

5.3 仓库创建 9

5.4 仓库显示与管理 9

5.5 编辑记录和发布管理 9

5.6 讨论区功能 9

5.7 代码互助 9

5.8 自然语言编程 9

6. 约束 9

7. 质量范围 10

8. 优先级 10

9. 其他产品需求 10

9.1 适用的标准 10

9.2 系统需求 10

9.3 环境需求 11

10. 文档需求 11

10.1 用户手册 11

10.2 联机帮助 11

10.3 安装指南、配置文件、自述文件 11

前景

# 简介

## 目的

本文件旨在根据最终用户的需求，定义代码管理平台Code Cube（CC）的高级要求。

## 范围

本前景文件适用于上海交通大学软件工程原理与实践课开发的代码管理平台。项目组将开发该客户端-服务器系统，以建立用户本地与数据库代码数据的对接。

通过代码管理平台，用户能管理自己的项目文件，辅助生成代码。不同用户之间能通过系统进行代码查找，浏览和保存公开的代码，发起私信或公开讨论。

## 定义、首字母缩写词和缩略语

见术语表。

## 参考资料

适用的参考文献有：

1.代码管理平台Code Cube术语表

# 定位

## 商机

现有的代码管理平台主要有github，它对于项目结构的呈现不够直观，不适用于新手了解、学习代码和用户查看代码、了解实现过程。比如文件夹页面分属不同网站，访问目录时间慢。文件内部函数声明与实现位置难以快速找到。

对于项目管理，我们提供更加方便的页面布局方式，包括仓库呈现方式、代码呈现方式，以及更方便的搜索功能、自动生成代码的功能。同时我们将保留收藏、发起提问等实用功能，致力于提高平台的便捷性。

## 问题说明

|  |  |
| --- | --- |
| 问题是 | 无法直观查看项目结构、函数的声明实现位置 |
| 影响 | 代码使用者 |
| 问题的后果 | 项目对于用户来说是一个黑盒，无法快速了解项目结构，因此在学习和修改时耗时较长。 |
| 成功的解决方案 | 1.采用三维的文件目录界面。查看项目结构更直观。  2.将目录加载在同一个页面。切换文件夹时加载速度更快  3.使用链接标明函数声明和实现。方便理解代码。 |

## 

|  |  |
| --- | --- |
| 问题是 | 界面繁杂 |
| 影响 | 代码使用者和开发者 |
| 问题的后果 | 干扰信息太多，影响用户体验。 |
| 成功的解决方案 | 采用类似windows系统页面布局。访问各种功能都更加方便直观。 |

## 产品定位说明

|  |  |
| --- | --- |
| 针对于 | 项目开发者，项目使用者。 |
| 谁 | 上传、下载、查看、交流 |
| 该（产品名） | 属于工具。 |
| 功能 | 能够让用户直观地查看项目结构，快捷地管理项目、交流反馈。 |
| 不同于 | github |
| 我们的产品 | 页面更简洁、直观，操作更便捷，功能更强大。 |

# 涉众和用户说明

本产品旨在服务于一个广泛的用户群体，核心用户群主要包括从事代码相关行业的专业人士和计算机科学或相关领域的学生。这些用户群体对代码管理、版本控制以及协作编码工有着紧迫和持续的需求。

**专业开发人员**：这一群体包括软件开发人员、工程师和技术项目经理。他们需要一个可靠、高效的平台来托管和管理代码，同时也需要强大的合作和代码审查工具来优化开发流程。他们追求的是提高生产力、简化工作流程以及确保代码的质量和安全性。

**学生和学术界**：计算机科学专业的学生或研究人员在学习和研究过程中需要接触到版本控制和代码合作工作。这个群体对易于学习和使用的工具有高需求，这样可以帮助他们更快地掌握现代软件开发的最佳实践，并促进团队项目和学术研究的合作。

## 市场统计

**目标市场和用户基础**

我们的目标市场是软件开发和计算机科学领域，特别是那些需要高效代码管理和协作工具的专业人士和学生。我们通过行业报告、市场调研和用户访谈收集数据，以确定这一群体的大小和需求。

**细分市场定位**：我们特别关注中小型企业（SMEs）和教育机构。这些细分市场对易于使用、功能丰富且经济高效的代码管理和协作平台有着迫切需求。

**组织在市场中的声誉**

当前声誉：目前，我们在该领域的影响力还未充分开发。

期望声誉：我们希望以一个创新者的身份活跃在该领域，在提供创新解决方案和支持软件开发教育方面处于行业前沿。

产品对策略目标的支持

支持组织目标：本产品将通过提供一个集成的平台来支持代码管理、协作和学习，从而帮助我们开辟在软件开发工具市场的新领域，并扩大我们在教育领域的影响。

满足市场需求：通过满足专业开发人员和学生对高效、协作和学习工具的需求，我们可以解决市场的未满足需求，同时提高我们的市场份额和品牌知名度。

## 涉众概要

|  |  |  |
| --- | --- | --- |
| **名称** | **说明** | **角色** |
| 开发者小组 | 本产品开发者小组由三人构成，分别承担项目中的不同任务 | 完成各迭代步骤，构建并完善软件架构 |
| 沈备军老师 | 项目的发起人，验收者 | 定期发起迭代请求，验收项目阶段性成果 |
| 同学 | 具有相同需求的同学 | 使用者，用户 |

## 用户概要

|  |  |  |
| --- | --- | --- |
| **名称** | **说明** | **涉众** |
| 学生 | 使用代码管理支持更模块化的代码学习，使用代码工具模仿更灵活高效的算法 | 涉众-同学 |

## 关键的涉众/用户需要

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **需要** | **优先级** | **关注的要点** | **目前的解决方案** | **提议的解决方案** | |
| 代码版本控制不足 | 高 | 团队成员在并行开发时经常出现代码冲突和覆盖 | 手动管理不同版本的代码文件，依赖成员之间的沟通来避免冲突 | | 引入一个集成的版本控制系统，如Git，以自动化处理版本冲突和历史记录跟踪 |
| 缺乏实时协作功能 | 低 | 成员间开发时间冲突，面临突发问题无法及时协同解决 | 通过电子邮件和即时消息工具进行通信，但这些工具与代码编辑环境分离 | | 集成实时协作工具和聊天功能到开发环境中或同步进行，以改善团队沟通和协作效率 |
| 不足的项目跟踪和管理 | 中 | 团队成员难以追踪项目进度和任务状态，导致管理混乱 | 使用独立的项目管理软件或Excel表格来追踪任务和进度 | | 整合任务管理和进度跟踪工具进入主要开发平台，提供一体化的项目视图和通知系统 |

## 备选方案和竞争

### GitHub

**产品概述:**

GitHub是一个广泛使用的代码托管平台，为数百万开发者和公司提供了代码管理和协作的解决方案。GitHub支持多种编程语言，提供了包括版本控制、项目管理、代码审查和团队协作的功能。

**主要优点:**

广泛的用户基础和社区支持，使新用户易于学习和获得帮助。

强大的集成能力，与多种开发工具和服务（如CI/CD流水线、项目跟踪工具等）无缝集成。

高级功能，如自动代码审查、安全性检查和大规模协作支持。

**主要缺点:**

对于非技术用户而言，界面和某些功能可能显得复杂。

高级功能和私有仓库可能需要付费订阅。

### GitLab

产品概述:

GitLab是另一个流行的代码托管和CI/CD平台，提供从代码创建到部署的全流程服务。与GitHub相比，GitLab提供了一些内置的持续集成/持续部署的解决方案。

**主要优点:**

内置CI/CD解决方案，简化了从代码提交到部署的工作流程。

开放源代码，允许更大程度的自定义和控制。

综合性的项目管理工具，适合从小型项目到大型企业的不同需求。

**主要缺点:**

初始设置和配置可能比GitHub更复杂。

尽管提供了许多高级功能，但界面可能不如GitHub直观。

1. **产品概述**
   1. **产品总体效果**

本产品主要由以下三个主要界面构成：

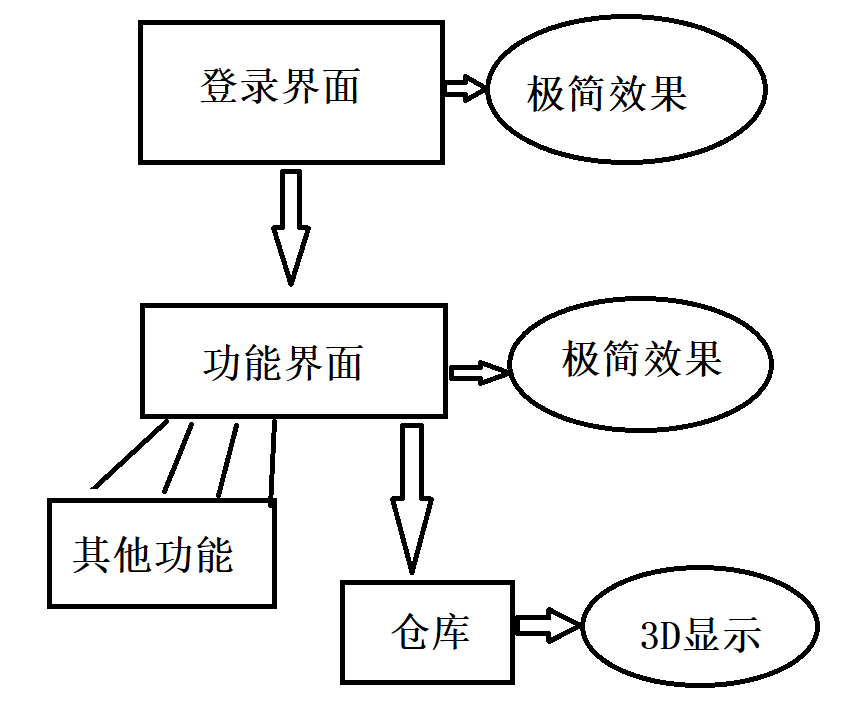
1. 登录页面。为让用户最大化便利使用，登录界面使用极简设计，并追求让用户手不离开键盘完成所有登录选项。
2. 功能界面。本产品的主界面，同样采用极简设计，力求让用户以最快的速度找到自己需要使用的功能。
3. 仓库界面。本产品的仓库界面使用3d模型，在便捷的同时增加可视化信息，让用户更快锁定相应选项。

图4.1.1 产品框架与总体效果

* 1. **功能摘要**

|  |  |
| --- | --- |
| **客户利益** | **支持特性** |
| 开发者可以安全地存储、管理和跟踪代码变化。 | 集成的Git仓库管理。  分支、合并和标签功能，支持版本控制和代码回溯。 |
| 开发团队可以在项目中高效地合作和沟通。 | 代码审查和合并请求，促进代码质量提高和团队协作。  实时讨论和注释功能，提高交流效率和解决方案的质量。 |
| 团队成员可以轻松跟踪和管理项目进度。 | 里程碑和时间表，帮助团队设置目标和截止日期。 |
| 开发者可以便捷轻松地使用相应功能。 | 极简的UI设计，帮助开发者分类不同模块。 |

* 1. **假设与依赖关系**

**假设:** 代码协作工具市场将继续增长，开发者对此类工具的需求将增加。

依赖关系: 持续的市场需求和增长。

潜在变更: 如果市场需求突然减少，我们可能需要重新评估并调整市场策略。

**假设：**部分功能在构建时新增，对此类功能进行构建。

依赖关系：用户需求与团队创新。

潜在变更：如果研发时间不够，可能需要削减部分功能。

# 产品特性

本小节列出了Code Cube的各种产品特性，旨在为程序开发提供一个全面的代码管理和协作解决方案。

## 注册和登录

提供一个简单、安全的用户注册和登录过程，支持使用JACCOUNT登录，以提高用户入门的便捷性，确保用户能够快速开始使用平台，同时保障他们的账户安全。

## 创建个人资料页面

允许用户创建和编辑个人资料，上传头像，并添加个人技能标签。这使得用户能够展示自己的专业技能，促进了社区内的互动和合作。

## 仓库创建

用户可以轻松创建和管理代码仓库，上传项目文件，如代码、readme文档、requirements等，支持版本控制和项目分支管理。

## 仓库显示与管理

展示用户代码仓库的重要信息，如更新时间、star数量、权限等，方便用户快速获取项目动态。

## 编辑记录和发布管理

提供编辑记录功能，允许用户在代码更新时添加备注。通过发布管理功能，用户可以编辑和发布release信息，让社区成员了解最新动态。

## 讨论区功能

提供一个社区讨论区，使用户能够搜索和下载公开仓库，反馈漏洞，提出需求，促进知识共享和问题解决。

## 代码互助

用户可以根据项目需求，寻找并匹配有类似项目技术、特定技能标签的同学，以私聊的方式进行项目交流与合作。

## 自然语言编程

利用自然语言处理技术，实现通过自然语言指令生成代码，进行项目编写，并且将代码组织成完整项目，简化编程任务，提高开发效率。

# 约束

本产品作为Web网站，需要在浏览器中运行使用

# 质量范围

本小节定义了产品的易用性、可靠性、性能、可支持性以及其他类似特征的质量范围。

易用性：平台提供用户手册、在线帮助，为用户使用提供帮助

可靠性：平台应实现7×24h可用，可用性99%

性能：平台应支持1000个并发用户同时交互，2000人同时在线，允许5000人次用户注册，允许每个用户创建、管理100个代码仓库，响应时间不超过5秒。

可支持性：平台符合Google编码规范，便于修改bug；兼容Edge、Chrome、Safari、Firefox浏览器，兼容不同桌面分辨率

# 优先级

预计Code Cube将通过2-3个主要Release发布上线。在该产品的开发过程中，产品特性（参考本文件第7节）将用于衡量特性的相对重要性，并对发布内容进行整体规划，确定优先级。

Release1需要实现如下的产品特性：

* 注册和登录
* 创建个人资料页面
* 仓库创建
* 仓库显示与管理
* 编辑记录和发布管理

Release2需要包括：

* 讨论区功能
* 代码互助
* 自然语言编程

Release3的功能实现尚未确定，预计将包括对已有功能的进阶、增强，如改进自然语言编程功能。

# 其他产品需求

## 适用的标准

符合Window平台一致性标准。

## 系统需求

* 支持的操作系统：Windows、macOS、Linux。
* 浏览器兼容性：最新版本的Chrome、Firefox、Safari、Edge。

## 环境需求

* 使用条件：确保在网络稳定的环境下使用，以保障最佳体验。
* 维护和更新：定期进行系统维护和更新，确保服务质量。

# 文档需求

## 用户手册

提供详细的用户手册，包括产品概述、开始指南、功能详解、常见问题解答等，支持在线和PDF格式。

## 联机帮助

开发一个综合的在线帮助系统，包括搜索功能、教程、使用技巧等，以帮助用户快速解决问题。

## 安装指南、配置文件、自述文件

提供详细的安装指南和配置文件，以及包含新特性、已知问题和变通方法的自述文件，帮助用户顺利完成安装和配置。