# 修改历史

|  |  |  |  |
| --- | --- | --- | --- |
| 修改人员 | 日期 | 修改原因 | 版本号 |
| 孙婧 | 2016.2.28 | 完成文档 | V1.0 |
| 孙婧 | 2016.2.29 | 拼写及用词规范 | V1.1 |
| 孙婧 | 2016.3.2 | 增加实时更新网络数据功能 | V1.2 |

# GitMining软件需求规格说明文档

## 引言

1.1目的

本文档描述了GitMining软件系统的相关功能需求和非功能需求。开发小组的软件实现和验证都以此文档为依据。

1.2范围

GitMining系统的目标是通过对github 网站项目和用户数据的分析展示，来帮助广大编程爱好者方便快捷的找到自己感兴趣的项目，并了解相关项目的一些基本信息。

## 总体描述

**2.1软件前景**

2.1.1背景

越来越多的编程爱好者通过github网站来进行软件项目的开发，许多编程爱好者希望能够更方便更迅速的在github网站上找到自己感兴趣的项目。所以我们开发了GitMining系统，帮助广大编程爱好者查找github网站上相关项目和项目所有者的基本信息以及对相关数据的分析结果，以此找到自己感兴趣的项目。

2.1.2业务需求

BR1：系统使用1个月后，用户量保持增加的趋势。

BR2：系统使用1个月后，80%以上编程爱好者能顺利找到感兴趣的项目。

**2.2软件功能**

SF1：提供项目和用户搜索、排序功能。

SF2：提供单个项目和单个用户基本信息展示功能。

SF3：用户需要时，软件可以向用户提供实时更新github网站上项目和项目所有者的信息的功能。

**2.3用户特征**

|  |  |
| --- | --- |
| 编程爱好者 | 本系统的使用者，通过项目名或项目所有者的登录名进行搜索，查找到相关项目的基本信息，排序信息，以及单个项目和单个用户的基本信息，并通过对各项目和项目参与者信息的比较，找到自己感兴趣的项目。 |

**2.4约束**

CON1：采用Java语言开发

CON2：系统使用的是PC端的图形界面

CON3：迭代I II不允许使用数据库

CON4：项目采用分层模型进行开发

CON5：项目后期会增加数据来源及开放式功能

CON6：将工程行为尽可能地记录在Gitlab上

CON7：每次迭代产品均必须附带部署说明文档

## 详细需求描述

**3.1功能需求**

3.1.1项目和用户搜索、排序

3.1.1.1特性描述

编程爱好者（系统用户）通过输入项目名或项目所有者的登录名对项目进行查询，并可以根据star和fork等值查询到项目列表的排序。若用户希望得到github网站上项目的实时信息，需要为用户提供联网查询功能。（等待查询结果间隙用户可执行其他任务）

优先级=低

3.1.1.2刺激响应序列

刺激：编程爱好者输入项目名或项目所有者的登录名

响应：系统显示所查找的项目

刺激：编程爱好者通过star和fork等值查找项目列表的排序情况

响应：系统通过图表的形式向用户展示项目列表的排序情况

刺激：编程爱好者希望得到github网站上最新的项目信息

响应：系统通过网络连接从github网站上获取信息展示给用户

刺激：编程爱好者查询结束，请求退出系统

响应：系统关闭，等待下一次服务

3.1.1.3相关功能需求

|  |  |
| --- | --- |
| Search.Input  Search.Input.Begin  Search.Input.End | 系统应该允许编程爱好者在搜索项目的过程中进行键盘输入和鼠标点击。  在系统用户开始搜索任务时，系统应该执行搜索任务。  在系统用户结束搜索任务时，系统应该关闭一次搜索任务。 |
| Search.Information | 系统显示编程爱好者所查找的项目信息 |
| Search.sort | 在系统用户通过star和fork等值查找项目列表的排序时，系统显示项目列表的排序信息 |
| Search.update | 用户希望得到项目的实时信息时，系统通过网络连接为用户展示项目的最新信息 |
| Search.Close | 在系统用户完成搜索任务时，系统关闭 |

3.1.2单个项目基本信息展示

3.1.2.1特性描述

编程爱好者通过输入项目名对单个项目的基本信息进行查询。

优先级=高

3.1.2.2刺激响应序列

刺激：编程爱好者输入项目名称进行查询

响应：系统显示编程爱好者所查询的项目的基本信息（项目基本信息、项目使用语言、项目贡献者、项目合作者、项目fork信息）

刺激：编程爱好者结束对单个项目信息的查询

响应：系统关闭查询单个项目信息任务

3.1.2.3相关功能需求

|  |  |
| --- | --- |
| Program.Input  Program.Input.Begin | 系统应该允许编程爱好者在搜索单个项目信息时进行键盘输入和鼠标的点击。  在系统用户开始查询时，系统开始查询任务 |
| Program.Information | 系统显示编程爱好者查找的单个项目的所有基本信息 |
| Program.Close | 系统用户结束查找，系统关闭查找任务 |
| Program.Next | 系统开始下一次查找单个项目基本信息的任务 |

3.1.3单个用户基本信息展示

3.1.2.1特性描述

编程爱好者通过输入项目用户登录名对单个用户项目的基本信息进行查询。

优先级=高

3.1.2.2刺激响应序列

刺激：编程爱好者输入项目用户的登录名进行查询

响应：系统显示编程爱好者所查询的单个用户的基本信息（用户基本信息、用户参与项目、用户创建项目）

刺激：编程爱好者结束对单个用户信息的查询

响应：系统关闭查询单个用户信息任务

3.1.2.3相关功能需求

|  |  |
| --- | --- |
| User.Input  User.Input.Begin | 系统应该允许编程爱好者在搜索单个用户信息时进行键盘输入和鼠标的点击。  在系统用户开始查询时，系统开始查询任务 |
| User.Information | 系统显示编程爱好者查找的单个用户的所有基本信息 |
| User.Close | 系统用户结束查找，系统关闭查找任务 |
| User.Next | 系统开始下一次查找单个用户基本信息的任务 |

**3.2非功能性需求**

3.2.1安全性

Safety1：本系统不需要用户注册，只提供查询任务，不允许用户修改所查询到的任何信息，提供的项目信息仅供用户对比和参考。

3.2.2可维护性

Modifiablity1：如果系统增加新的项目，系统要在一天内完成数据统计的更新。

Mod2：如果系统增加新的统计图表，开发要在一周内完成。

3.2.3易用性

Usability1：系统界面应有良好的导航。

3.2.4可靠性

Reliability1：当网络发生故障时，系统可以通过上次缓存数据运行，并提示网络异常、重连数据服务器。

**3.3数据需求**

3.3.1数据定义

DR1：系统要存储API中所有的项目和用户数据。

DR2：迭代三之后，系统可以从github网站中获取项目和用户的相关数据。

DR3：迭代三之后，系统可以为用户提供实时更新github网站上项目和用户的最新相关数据。