**GitSearch 需求规格说明书**

目录

[**更新历史** 4](#_Toc444850025)

[**1.引言** 4](#_Toc444850026)

[1.1文档编写目的 4](#_Toc444850027)

[1.2定义、首字母缩写和缩略语 4](#_Toc444850028)

[1.3参考文献 4](#_Toc444850029)

[**2.项目概述** 5](#_Toc444850030)

[2.1项目范围说明 5](#_Toc444850031)

[2.1.1项目目标 5](#_Toc444850032)

[2.1.2项目相关人员和用户 5](#_Toc444850033)

[2.1.3项目相关事实和假定 5](#_Toc444850034)

[2.1.4项目的边界和范围界定（系统范围用例图） 6](#_Toc444850035)

[2.2 项目实现具体功能 6](#_Toc444850036)

[2.3 项目约束 7](#_Toc444850037)

[2.4 假设和依赖 7](#_Toc444850038)

[**3.详细需求描述** 7](#_Toc444850039)

[3.1对外接口需求 7](#_Toc444850040)

[3.1.1 用户界面 7](#_Toc444850041)

[3.1.2 硬件接口 8](#_Toc444850042)

[3.1.3 软件接口 8](#_Toc444850043)

[3.1.4 通讯接口 8](#_Toc444850044)

[3.2功能需求 8](#_Toc444850045)

[3.2.1 项目排序 8](#_Toc444850046)

[3.2.3 查看项目信息 10](#_Toc444850047)

[3.2.4查看用户信息 11](#_Toc444850048)

[3.2.5搜索项目 12](#_Toc444850049)

[3.2.6搜索用户 13](#_Toc444850050)

[3.3非功能需求 15](#_Toc444850051)

[3.3.1 安全性 15](#_Toc444850052)

[3.3.2 可维护性 15](#_Toc444850053)

[3.3.3 易用性 15](#_Toc444850054)

[3.3.4 可靠性 15](#_Toc444850055)

[3.4数据需求 15](#_Toc444850056)

[3.4.1 数据定义 15](#_Toc444850057)

[3.4.2 默认数据 15](#_Toc444850058)

[3.4.3 数据格式要求 15](#_Toc444850059)

[3.5质量及其他需求 16](#_Toc444850060)

**更新历史**

|  |  |  |  |
| --- | --- | --- | --- |
| **修改人员** | **日期** | **变更原因** | **版本号** |
| 金小枫 | 2016/2/29 | 迭代一需求规格说明书初稿 | V0.5 |
| 万兴、陈欢、金小枫 | 2016/3/2 | 完善功能需求 | V1.0 |
| 万兴、陈欢、金小枫、蒙奕锟 | 2016/3/4 | 评审修改 | V1.2 |
| 金小枫、蒙奕锟 | 2016/3/4 | 评审二次修改 | V1.5 |

# 1.引言

## 1.1文档编写目的

本文档描述了GitSearch的功能需求和非功能需求。开发小组的软件系统实现与验证工作都以此文档为依据。

除特殊说明之外，本文档所包含的需求都是高优先级需求。

本说明书的内容可能在项目实施过程中发生变更，但是必须由项目小组成员发出变更请求，小组讨论，最终决定，建立持续有效的版本控制。

## 1.2定义、首字母缩写和缩略语

GitSearch指帮助广大编程爱好者搜索或查询项目及用户信息的软件系统

## 1.3参考文献

1.《软件工程与计算（卷二）》

2.《软件工程与计算（卷三）》

3.《软件需求规格说明目标（IEEE标准）》

**2.项目概述**

本项目旨在通过对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查找到自己感兴趣的项目，了解相关项目的一些基本信息。

## 2.1项目范围说明

### 2.1.1项目目标

1.能够进行项目和用户名的搜索、排序

1）根据项目名和项目所有者的登录名进行搜索

2）根据star,fork等值对项目列表进行排序

3）根据参与项目数、注册时间等对用户列表进行排序

2.能够展示单个项目和单个用户的基本信息

### 2.1.2项目相关人员和用户

|  |  |  |
| --- | --- | --- |
| **代表** | 开发人员 | 用户（老师） |
| **说明** | 进行软件开发 | 使用人员 |
| **职责** | 完成系统功能 | 使用软件 |
| **成功标准（关注点）** | 1. .能够进行项目和用户名的搜索、排序 2. 能够展示单个项目和单个用户的基本信息 | 成功使用软件 |
| **参与方式** | Design、code、test | 不直接参与项目（老师提供数据） |
| **可交付工作** | 文档、代码 | 无 |
| **意见/问题** | 无 | 无 |

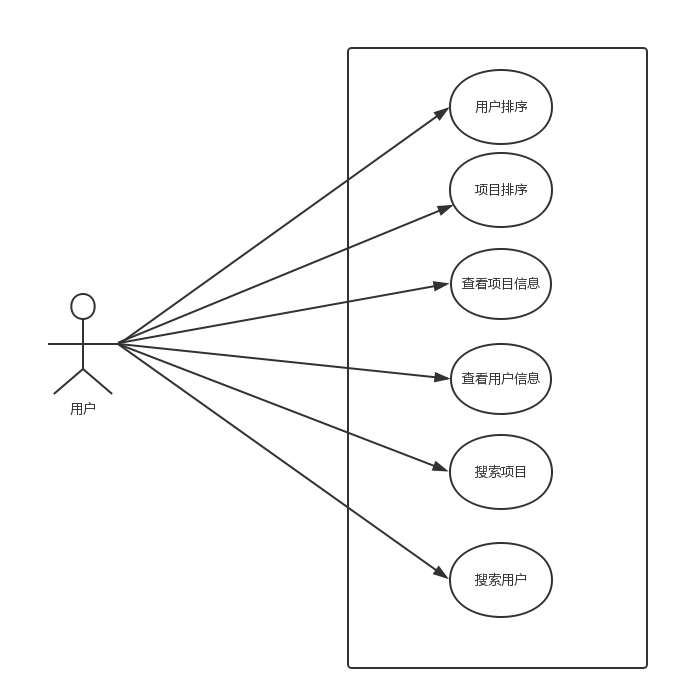
### 2.1.3项目相关事实和假定

FS1：GitSearch只在一台设备上部署，数据从api获取

FS2：GitSearch的数据由api提供

AS1：数据符合事实

### 2.1.4项目的边界和范围界定（系统范围用例图）



## 2.2 项目实现具体功能

1.用户在使用该系统时，能够在搜索框内输入项目名以搜索项目，也能够输入项目所有者的登录名搜索用户。

2.用户在使用该系统时，可以根据star,fork等值对项目列表进行排序。

3.用户在使用该系统时，可以根据参与项目数、注册时间等值对用户列表进行排序。

4.当用户要查看某个项目的信息时，应展示该项目的基本信息、项目使用语言、项目贡献者、项目合作者、项目fork信息。

5. 当用户要查看某个用户的信息时，应展示该用户的基本信息、用户参与项目、用户创建项目。

## 2.3 项目约束

**CON1**：采用Java语言开发

**CON2**：系统使用的是PC端的图形界面

**CON3**：项目采用分层模型进行开发

**CON4**：将工程行为尽可能地记录在Gitlab上

## 2.4 假设和依赖

假设每次都能从api成功获得数据

# 3.详细需求描述

## 3.1对外接口需求

### 3.1.1 用户界面

界面风格：本系统采取图形化用户界面，界面风格简洁，操作方便，支持用户用鼠标和键盘进行操作，具体可考虑实现多个风格的界面，包括windows通用图形界面,以及更多地使用look&feel进行界面定制。

界面布局:界面布局整齐合理，不会过于复杂。

### 3.1.2 硬件接口

无

### 3.1.3 软件接口

此系统可运行在windows系统上，数据从api获取，需要机器具有JRE。

### 3.1.4 通讯接口

此系统在本地运行，无通讯接口

**3.2功能需求**

**3.2.1 项目排序**

**3.2.1.1 特性描述**

用户可以根据star,fork,contributor等关键字对项目列表进行排序

**3.2.1.2 刺激/响应序列**

刺激：用户请求进行项目排序

响应：系统显示可进行排序的关键字，例如star，fork，contributor等

刺激：用户选择一项排序标准

响应：系统显示根据该值排序后的项目列表

**3.2.1.3 相关功能需求**

|  |  |
| --- | --- |
| **编号** | **需求描述** |
| Sort.Sort | 系统应该允许用户进行项目排序 |
| Sort.Select | 系统应该允许用户在进行项目排序时进行关键值选择 |
| Sort.ShowList | 系统在用户选择关键值后，显示根据该关键值排序后的项目列表 |

**3.2.1.4 用例描述**

**用例编号**：UC001

**用例名称**：项目排序

**参与者**： 用户

**前置条件**：用户需要对项目进行排序

**后置条件**：显示按条件排序的项目列表

**主事件流**：

1.系统显示可以排序的类型

2.用户选择排序类型

3.系统根据该类型的排列顺序由大到小显示

**次要事件流**：

2a.如果排序发生错误，系统显示系统故障

**业务规则：**无

**非功能性需求**：

系统应保证所公布的信息与原来的信息完全一致

系统应在2秒内处理完用户的请求

**3.2.2 用户排序**

**3.2.2.1 特性描述**

用户可以根据star,fork,contributor等关键字对用户列表进行排序

**3.2.2.2 刺激/响应序列**

刺激：用户请求进行用户排序

响应：系统显示可进行排序的关键字，例如参与项目数，注册时间等

刺激：用户选择一项排序标准

响应：系统显示根据该值排序后的用户列表

**3.2.2.3 相关功能需求**

|  |  |
| --- | --- |
| **编号** | **需求描述** |
| Sort.Sort | 系统应该允许用户进行用户排序 |
| Sort.Select | 系统应该允许用户在进行用户排序时进行关键值选择 |
| Sort.ShowList | 系统在用户选择关键值后，显示根据该关键值排序后的用户列表 |

**3.2.2.4 用例描述**

**用例编号**：UC002

**用例名称**：用户排序

**参与者**： 用户

**前置条件**：用户需要对用户进行排序

**后置条件**：显示按条件排序的用户列表

**主事件流**：

1.系统显示可以排序的类型

2.用户选择排序类型

3.系统根据该类型的排列顺序由大到小显示

**次要事件流**：

3a.如果排序发生错误，系统显示系统故障

**业务规则：**无

**非功能性需求**：

系统应保证所公布的信息与原来的信息完全一致

系统应在2秒内处理完用户的请求

**3.2.3 查看项目信息**

**3.2.3.1 特性描述**

用户可以通过本系统查询单个项目的信息，包括该项目的基本信息、项目使用语言、项目贡献者、项目合作者、项目fork信息。

**3.2.3.2 刺激/响应序列**

刺激：用户点击某个项目

响应：系统显示该项目的详细信息

**3.2.3.3 相关功能需求**

|  |  |
| --- | --- |
| **编号** | **需求描述** |
| RepositoryCheck. RepositoryCheck | 系统应该允许用户查看项目信息 |
| RepositoryCheck.ShowRepositoryInfo | 系统在用户点击某个项目后，显示该项目的详细信息 |

**3.2.3.4 用例描述**

**用例编号**：UC002

**用例名称**：查看项目信息

**参与者**： 用户

**前置条件**：用户需要查看某一项目

**后置条件**：显示项目详细信息

**主事件流**：

1. 用户点击某一项目
2. 系统显示该项目信息（包括使用语言、项目贡献者、项目合作者、项目fork信息）

**次要事件流**：

无

**业务规则**：无

**非功能性需求**：

系统应保证所公布的信息与原本的信息完全一致

系统应在2秒内处理完用户的请求

**3.2.4查看用户信息**

**3.2.4.1 特性描述**

用户可以通过本系统查询用户的信息，包括该用户的基本信息、用户参与项目、用户创建项目。

**3.2.4.2 刺激/响应序列**

刺激：用户点击用户名

响应：系统显示相应用户信息

**3.2.4.3 相关功能需求**

|  |  |
| --- | --- |
| UserCheck.UserCheck | 系统应该允许用户查看用户信息 |
| UserCheck.ShowUserInfo | 系统在用户点击某个用户后显示该用户的详细信息 |

**3.2.4.4 用例描述**

**用例编号**：UC003

**用例名称**：查看用户信息

**参与者**： 用户

**前置条件**：用户需要查看某一用户

**后置条件**：显示用户详细信息

**主事件流**：

1. 用户点击某用户名
2. 系统显示该用户信息（包括该用户的基本信息、用户参与项目、用户创建项目）

**次要事件流**：

无

**业务规则**：无

**非功能性需求**：

系统应保证所公布的信息与原来的信息完全一致

系统应在2秒内处理完用户的请求

**3.2.5搜索项目**

**3.2.5.1 特性描述**

在需要搜索某项目时，用户输入关键词即可找到相应项目列表

**3.2.5.2刺激/响应序列**

刺激：用户请求搜索项目

响应：系统显示所有项目的列表

刺激：用户输入关键词并进行搜索

响应：系统显示相应项目列表

**3.2.5.3相关功能需求**

|  |  |
| --- | --- |
| RepositoryFinding. ProgramFinding  RepositoryFinding.Input | 系统应该允许用户搜索项目  系统应该允许用户进行键盘输入 |
| RepositoryFinding.Invalid | 在用户输入错误标示后，系统提示输入错误并要求重新输入 |
| RepositoryFinding.ShowRepositoryList | 在用户确定搜索后，系统根据输入信息显示相应的项目列表 |

**3.2.5.4用例描述**

**用例编号**：UC004

**用例名称**：搜索项目

**参与者**： 用户

**前置条件**：用户需要搜索某项目

**后置条件**：显示符合条件的项目列表

**主事件流**：

1. 用户输入想要寻找的项目名
2. 系统显示符合条件的项目列表

**次要事件流**：

1a.用户输入了非法字符

系统提示所输项目名不符规范

2a.如果所寻找的项目不存在，则显示不存在。

**业务规则**：无

**非功能性需求**：

系统应保证所公布的信息与原来的信息完全一致

系统应在2秒内处理完用户的请求

**3.2.6搜索用户**

**3.2.6.1 特性描述**

在需要搜索一个或多个用户时，用户输入关键词即可搜索相应的一个或多个用户

**3.2.6.2 刺激/响应序列**

刺激：用户请求搜索一个或多个用户

响应：系统显示所有用户信息的列表

刺激：用户输入关键词并选择搜索

响应：系统显示相应用户列表

**3.2.6.3相关功能需求**

|  |  |
| --- | --- |
| UserFinding.UserFinding | 系统应该允许用户在进行用户搜索 |
| UserFinding.Input | 系统应该允许用户进行键盘输入 |
| UserFinding.Invalid | 在用户输入非法字符后，系统提示输入错误并要求重新输入 |
| UserFinding.ShowUserList | 在用户确认搜索之后，系统根据输入信息显示相应用户列表 |

**3.2.6.4 用例描述**

**用例编号**：UC005

**用例名称**：搜索用户

**参与者**： 用户

**前置条件**：用户需要搜索某一用户

**后置条件**：显示符合条件的用户列表

**主事件流**：

1. 用户输入想要寻找的用户
2. 系统显示满足要求的用户列表

**次要事件流**：

1a. 用户输入了非法字符

系统提示所输用户名不符规范

2a.如果所寻找的用户不存在，则显示不存在。

**业务规则**：无

**非功能性需求**：

系统应保证所公布的信息与原来的信息完全一致

系统应在2秒内处理完用户的请求

**3.3非功能需求**

**3.3.1 安全性**

本系统的所有数据均从api获取，能保障数据的安全性

**3.3.2 可维护性**

Modifiability1：如果系统要增加项目的新的属性，仅需少量代码便可完成。

Modifiability2：如果系统要增加用户的新的属性，仅需少量代码便可完成。

**3.3.3 易用性**

Usability1：不需要用户使用手册或系统使用培训，用户也能够使用本系统所有功能

Usability2：当总数据发生改变时，可以很快更新数据信息，保持数据及时性

**3.3.4 可靠性**

Reliability1：系统用JAVA语言编写，运行稳定

Reliability2：系统数据从api获取，数据安全且不会被其他未知程序修改

Reliability3：系统无法正常运行时，数据也不会丢失

## 3.4数据需求

**3.4.1 数据定义**

无

**3.4.2 默认数据**

数据均从API获取

**3.4.3 数据格式要求**

Format1：从api获得String或Json数据

## 3.5质量及其他需求

该系统必须在2周内设计、编码、测试完成

系统展现的数据必须与所提供的数据一致，不能出现差错

系统应在显示、查询、排序请求2秒内完成正确响应