**仓库项目代码托管功能**

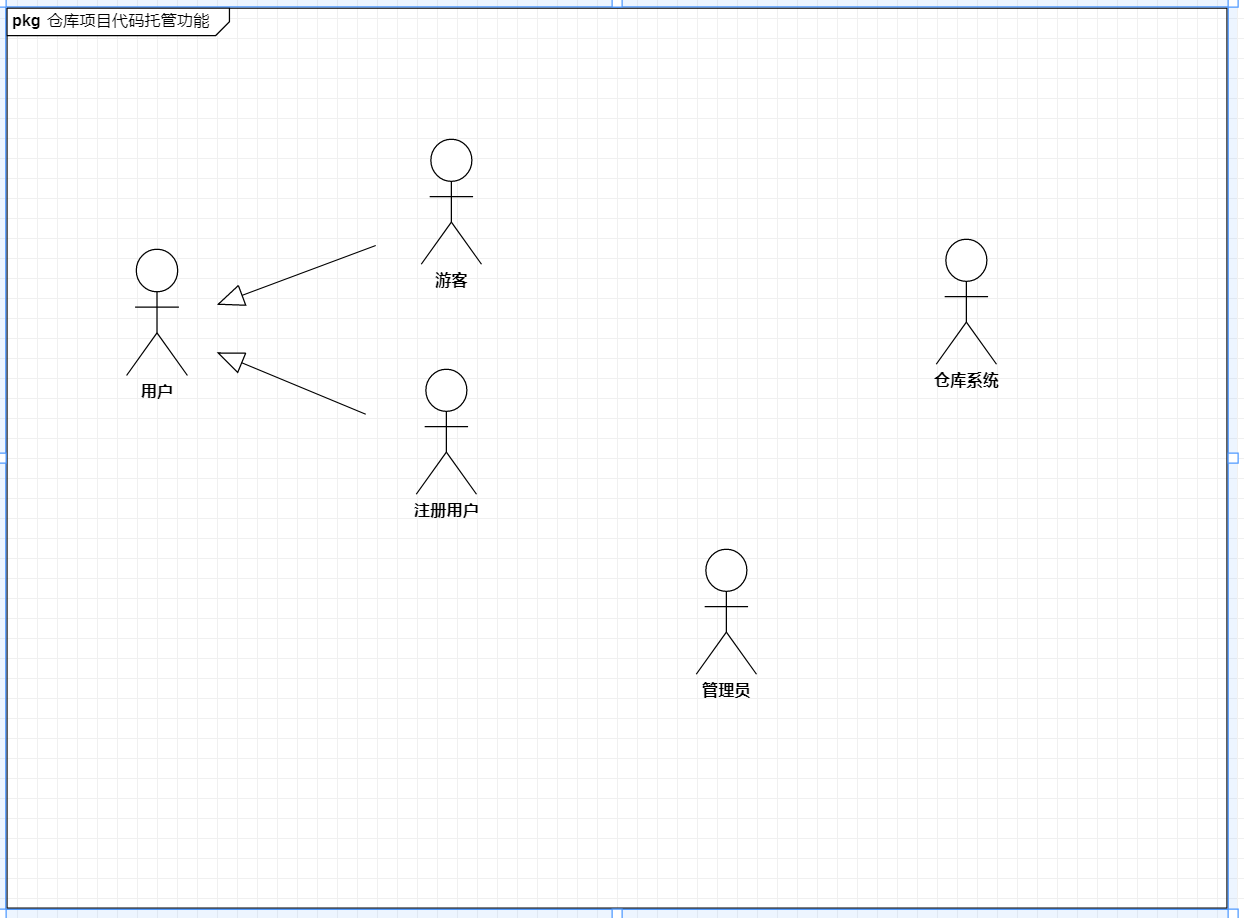


图1 确定参与者

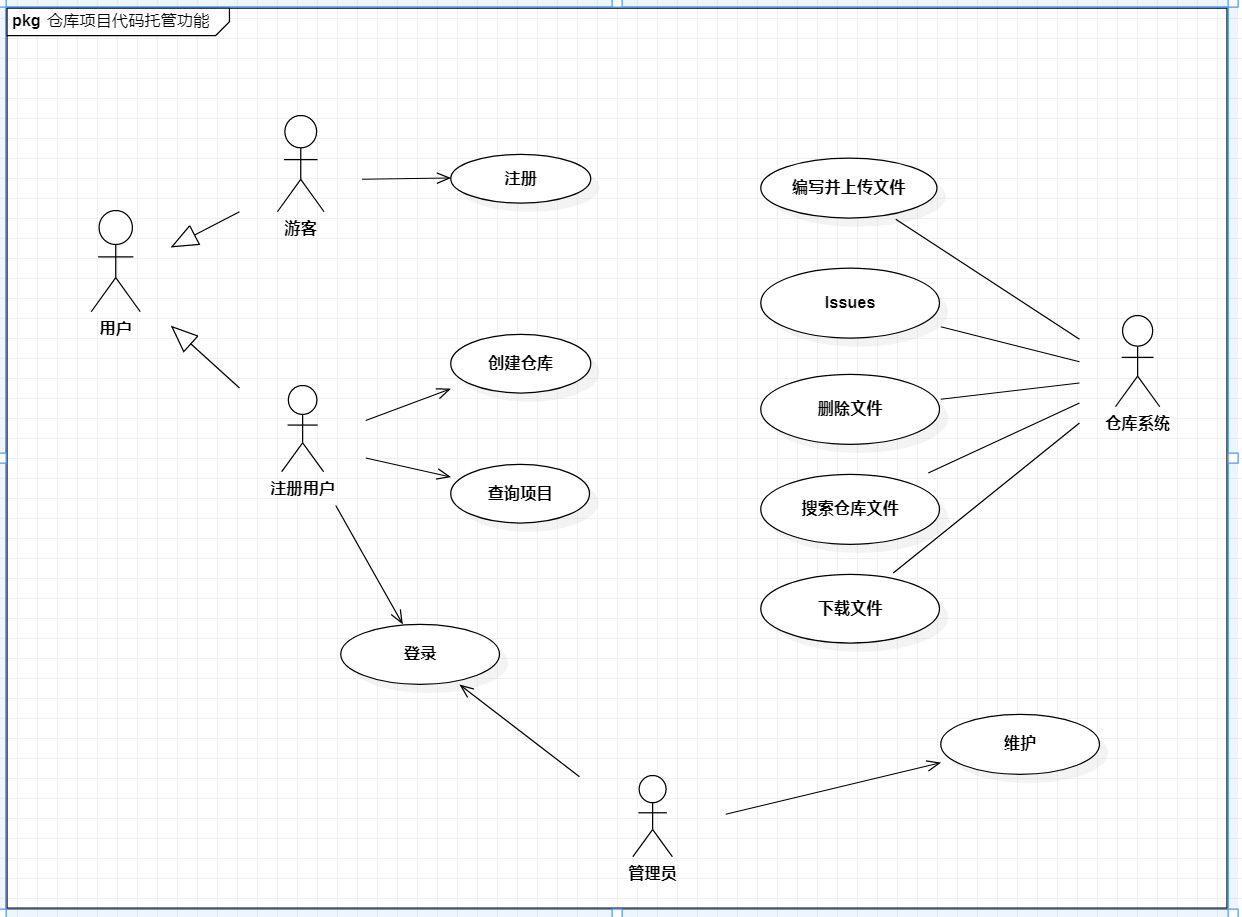


图2 确定用例

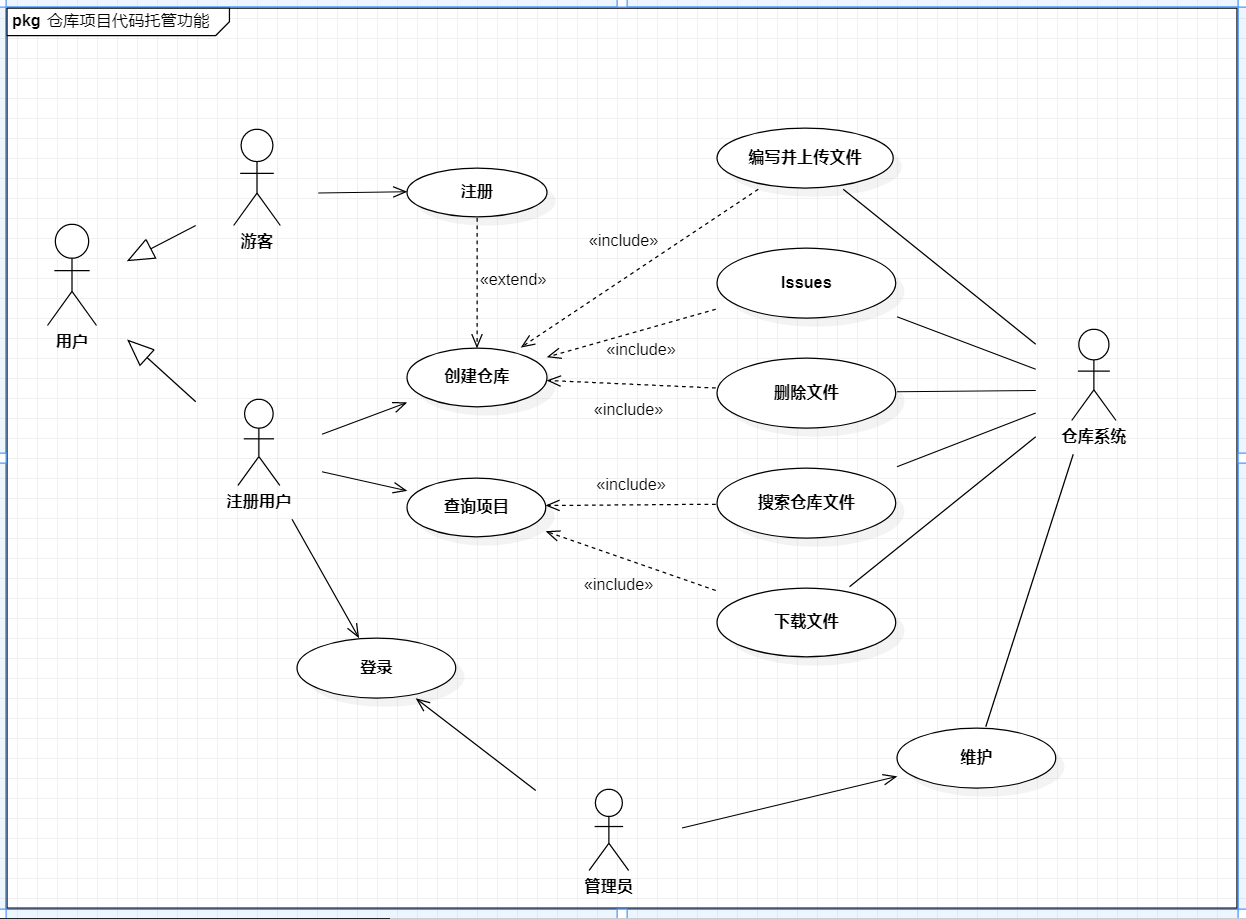


图3 最终用例图

1、用例说明

|  |  |
| --- | --- |
| 用例名称 | 用户注册 |
| 用例编号 | UC001 |
| 参与者 | 游客，系统管理员 |
| 用例描述 | 该用例描述游客可以在github网站上进行注册，成为该github网站用户。 |
| 触发器 | 当游客请求注册时，用例触发 |
| 前置条件 | 游客访问网站，尚未注册需要进行注册登录的游客。 |
| 后置条件 | 系统管理员后台记录用户信息 |
| 基本事件流 | 1. 游客请求注册。 2. 系统显示注册页面。 3. 游客提供注册身份信息。 4. 系统检查信息是否充分。 5. 系统保存用户信息。 6. 系统显示“注册成功”。 |
| 扩展事件流 | 1. 如果用户名错误：用户名已存在或者用户名无效，则系统会给出错误原因，提醒游客重新输入。 2. 如果游客输入的密码无效时，系统会提示游客密码太长、太短或字符方面的错误，让游客重新输入密码。 3. 如果校验码错误，系统会生成一个新的校验码，让游客重新进行验证。 4. 如果游客所提供的信息已经注册过或信息错误，系统会提醒游客该信息已注册或出错原因，重新进行注册。 |
| 结论 | 当游客收到页面提示注册成功时，用例结束。 |
| 数据需求 | 游客所提供的注册身份信息，密码，验证码。 |
| 业务规则 | 当身份信息，密码 ，验证码校验正确，完成注册。 |

|  |  |
| --- | --- |
| 用例名称 | 查询项目 |
| 用例编号 | UC002 |
| 参与者 | 用户 |
| 用例描述 | 用于查询仓库中的项目 |
| 触发器 | 点击Search or jump to触发 |
| 前置条件 | 用户成功注册 |
| 后置条件 | 项目查询成功 |
| 基本事件流 | 1. 在搜索框里输入要搜索的项目 2. 在出示的项目中点击想查找的项目 |

|  |  |
| --- | --- |
| 用例名称 | 创建仓库 |
| 用例编号 | UC003 |
| 参与者 | 仓库系统，用户，管理员 |
| 用例描述 | 该用例描述一个用户创建一个仓库的行为 |
| 触发器 | 当仓库被创建时，用例触发 |
| 前置条件 | 用户已完成注册 |
| 后置条件 | 仓库注册成功 |
| 基本事件流 | 1. 注册完成后，点击界面右上角+号 2. 点击new repositories 3. 进入新的界面后，在repository name框内输入仓库名称 4. 在description（optional）输入仓库介绍 5. 勾选public 6. 勾选Add a README file 和 Add .gitignore |

|  |  |
| --- | --- |
| 用例名称 | 登录 |
| 用例编号 | UC004 |
| 参与者 | 用户，管理员 |
| 用例描述 | 用户通过注册时的用户名和密码访问系统。 |
| 触发器 | 当用户进入系统页面时 |
| 前置条件 | 用户需要首先完成注册操作 |
| 后置条件 | 用户可以访问自己的仓库 |
| 基本事件流 | 1. 用户打开github页面 2. 用户输入用户名和密码 3. 用户提交输入信息 4. 管理员审核信息 5. 管理员对用户信息进行操作 6. 登录成功 |

|  |  |
| --- | --- |
| 用例名称 | 维护 |
| 用例编号 | UC005 |
| 参与者 | 管理员，仓库系统 |
| 用例描述 | 定期做好数据备份，保证仓库正常运行或恢复仓库运行的基础后盾，实时发布和更新仓库内容 |
| 触发器 | 当仓库中有新的内容发布、删除、修改用例触发 |
| 前置条件 | 进入仓库的一方需要完成登录操作 |
| 后置条件 | 仓库列表中显示新的内容 |
| 基本事件流 | 1、参与者将仓库中新的内容提交给系统；  2、对仓库中时间久远的内容进行数据备份；  3、监控维护在内容更新时间里仓库的正常运行。 |
| 扩展事件流 | A1如果在仓库内容更新时，出现其他内容丢失，管理员可以系统在数据备份中找回丢失内容；  A2如果仓库出现卡顿或死机，管理员可以通过后台对系统进行维护。 |
| 结论 | 当仓库内容发布、修改、删除完毕，仓库的运行正常，数据有效备份，用例结束 |
| 补充约束 | 当仓库中没有用户时，也要对仓库进行正常的维护 |

|  |  |
| --- | --- |
| 用例名称 | 写并上传文件 |
| 用例编号 | UC006 |
| 参与者 | 用户 |
| 用例描述 | 用户在仓库中写并上传文件 |
| 触发器 | 点击Add file触发 |
| 前置条件 | 用户成功注册仓库 |
| 后置条件 | 文件上传成功 |
| 基本事件流 | 1. 点击Add file 2. 如果要写文件则点击Create new file 3. 如果要上传文件则点击Upload files 4. 在Create new file中可编写文件 5. 在Upload file中可选择要上传的文件 6. 点击Commit new file即可 |

|  |  |
| --- | --- |
| 用例名称 | Issue |
| 用例编号 | UC007 |
| 参与者 | 仓库系统，用户，管理员 |
| 用例描述 | 该用例用于发现软件的BUG并报告，有事想向作者询问、探讨，  事先列出今后准备实施的任务。 |
| 触发器 | 当用户需要创建issue时触发。 |
| 前置条件 | 发现bug并报告，需要回复讨论，提出问题。 |
| 后置条件 | 后台进行管理。 |
| 基本事件流 | 1. 创建issue。 2. 标题和描述概括这个问题时关于什么的。 3. 点击submit提交问题。 4. 用户可以到仓库中查看问题。 5. 问题已解决可以关闭这个issue，代表已经解决。 6. 可以对issue添加labels（标签），帮助用户对问题进行分类和过滤。 7. 可以对issue关联到milestone（里程碑）。 8. Comments（评论）允许有权访问仓库的任何人对该问题进行回复。 |
| 扩展事件流 | 1. 问题查看搜索可以通过 Milestones，Labels 和 Assignees，对问题单进行分类和过滤。 2. 每个问题都可以有一个责任人 —— 一个人负责解决该问题的人。通过问题顶部的灰色栏，以同样的方式选择受责任人。 3. Notifications（通知）：是在 GitHub 的问题单中提及其他 GitHub 用户的方式。我们可以在任何问题的描述或评论中使用该功能，也可以通过 @username（@用户名）向另一个 GitHub 用户发送通知。 4. References（参考）：通常情况下，issue 会依赖于其他 issues，或者至少与它们相关。如果你想将它们关联起来，可以通过输入 #issue编号 来引用相关的问题。 5. 为特点代码添加issue，可以为仓库或者pull request中的特定代码添加issue。 |
| 结论 | 创建问题后，可以搜索查看问题，通过与别人合作讨论从而解决问题。 |
| 数据需求 | 需要问题内容概述，问题的责任人，项目的管理者。 |
| 业务规则 | 对问题的提出加标签，报告，里程碑，评论，回复，直到最后的解决问题结束。 |

|  |  |
| --- | --- |
| 用例名称 | 删除文件 |
| 用例编号 | UC008 |
| 参与者 | 仓库系统，用户，管理员 |
| 用例描述 | 用户删除文件时的操作 |
| 触发器 | 当用户点击 delete this repository 时触发 |
| 前置条件 | 用户需要首先完成注册登录操作 |
| 后置条件 | 仓库中不在显示被删除的文件 |
| 基本事件流 | 1. 在已经建立好的仓库中找到要操作的仓库 2. 点击setting选项 3. 下拉页面，找到danger zone 4. 点击右边红色方框中的delete this repository 5. 输入文件名 6. 输入密码 7. 删除成功 |

|  |  |
| --- | --- |
| 用例名称 | 搜索仓库文件 |
| 用例编号 | UC009 |
| 参与者 | 仓库系统 |
| 用例描述 | 用户或游客在搜索中搜索仓库中的文件，系统根据搜索的关键字在仓库中进行筛选 |
| 触发器 | 当用户或游客在搜索栏中输入关键字后回车，用例触发 |
| 前置条件 | 输入搜索关键字的一方需要完成登录 |
| 后置条件 | 1、用户在搜索栏中输入需要搜索的关键字；  2、系统根据搜索栏中的关键字对仓库中的文件进行筛选；  3、系统根据搜索到的含有关键字的文件进行排列返回  4、列出文件显示 |
| 基本事件流 | A1当搜索栏中的关键字不被识别，系统应及时返回找不到与关键字匹配的任何问题；  A2如果搜索栏的关键字识别后没有文件显示，系统应及时返回含有关键字的文件内容 |
| 扩展事件流 | 当搜索出文件或没有匹配文件时，用例结束 |
| 结论 | 如果仓库中有相对应的文件则显示文件；否者返回其他文件或没有匹配文件。 |
| 补充约束 | 只有当搜索栏中有输入的关键字，才会对文件进行筛选。 |

|  |  |
| --- | --- |
| 用例名称 | 下载文件 |
| 用例编号 | UC010 |
| 参与者 | 仓库系统，用户，管理员 |
| 用例描述 | 对仓库中的文件进行下载 |
| 触发器 | 点击download后，用例触发 |
| 前置条件 | 文件搜索成功 |
| 后置条件 | 文件下载成功 |
| 基本事件流 | 1. 在搜索框内搜索想要的文件 2. 点击code 3. 点击download |

1. 成员博客地址

刘雪婷：lnut.github.io/liuxueting20192123013

许婷：xuting20192154005.github.io/Xuting.github.io

王卓蕾：wangzhuoleiyaya.github.io/wangzhuolei20192123034.GitHub.io

代兴甜：xiaojuyoupiqi.github.io/daixingtian20192123074

杨雪婷：yangxutn.github.io/Yangxueting20192123043

陈贞如：shuyich.github.io/chenzhenru20192123005.github.io

1. 成员分工

成员一起讨论参与者、仓库项目代码托管的功能及其用例图

刘雪婷：写创建仓库和下载的用例描述

许婷：写注册和Issue的用例描述

王卓蕾：写维护和搜索仓库文件的用例描述

代兴甜：写登录和删除文件的用例描述

杨雪婷：绘制用例图，整合word

陈贞如：写查询项目库和编写并上传文件的用例描述