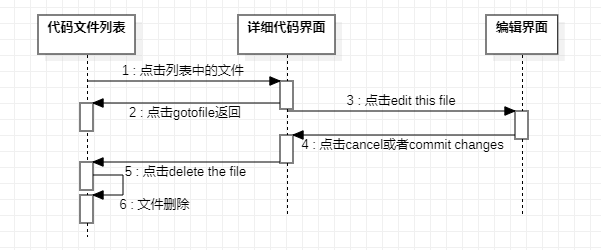
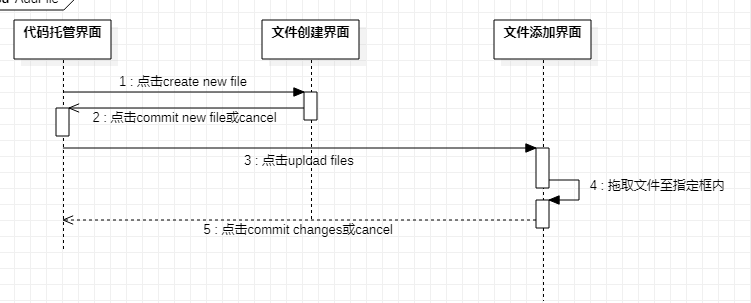
## 项目代码托管功能

顺序图展示

1.修改文件



2.新建文件

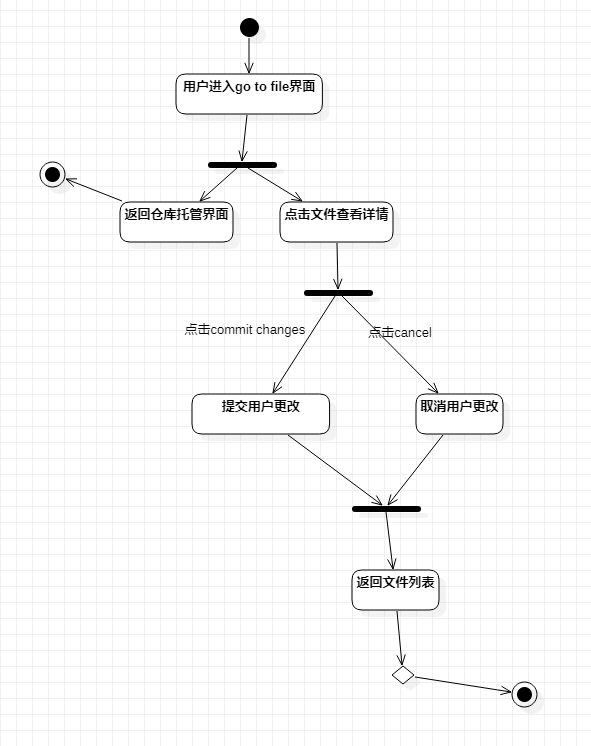


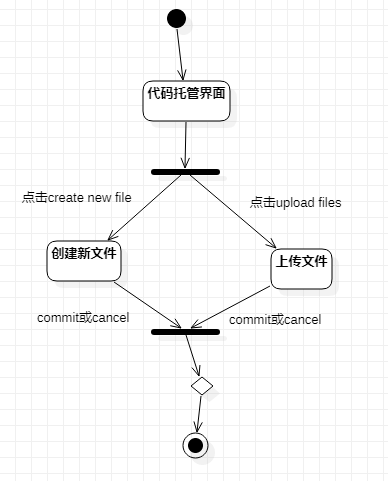
顺序图说明

|  |  |
| --- | --- |
| 对象 | user、系统界面、信息数据 |
| 业务流程 | 1.用户登录or注册进入系统  2.进入到仓库界面  3.点击Go to file 跳转到文件页面  4.点击edit this file 编辑对应的文件，点击delete the file删除相应的文件  5.然后取消或者提交修改 |
| 对象 | user、系统界面、信息数据 |
| 业务流程 | 1.用户登录or注册进入系统  2.进入到仓库界面  3.点击create new file创建新文件  4.点击commint new file或者cancel创建新文件或者取消操作  5.点击upload files上传文件，拖拽文件至指定位置  6.commit changes或者cancel |
| 建模思路 | 对于用户在github上使用代码托管功能，针对uml中的顺序图   1. 首先要考虑顺序图的对象，分别有user，系统，以及显示的文件信息。 2. 进入到系统后，再考虑一些过程，即对文件的操作 3. 可以对文件进行新建，上传文件，修改，删除等操作 4. 每次对应的操作都含有提交保存和取消。 |

Author：周泽瑜 Address：YukolQAQ.github.io

活动图展示

.



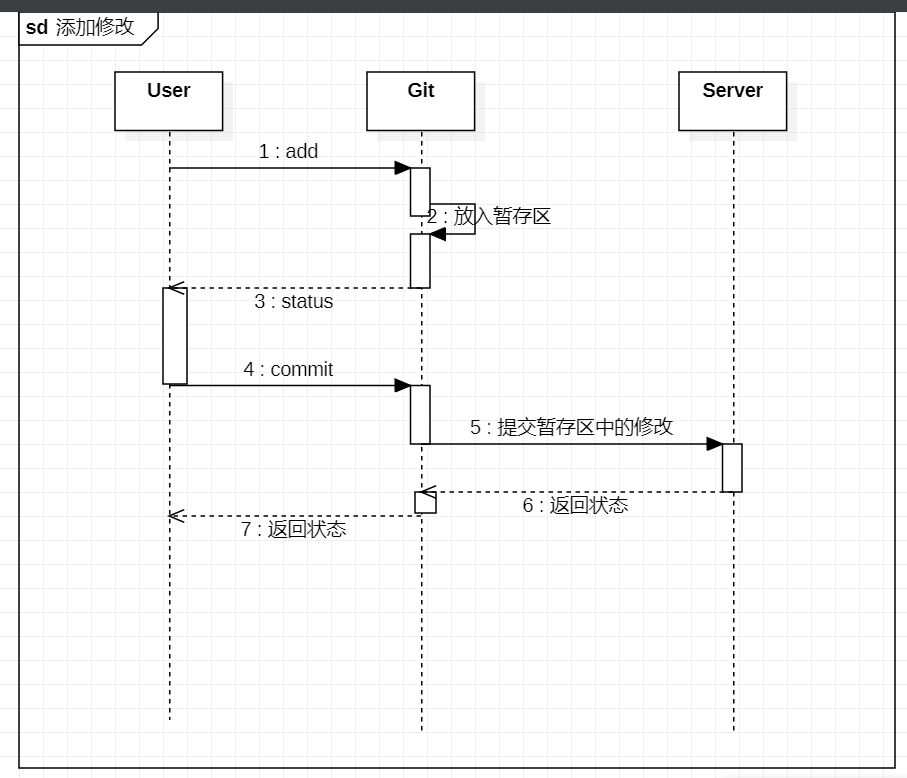
活动图说明

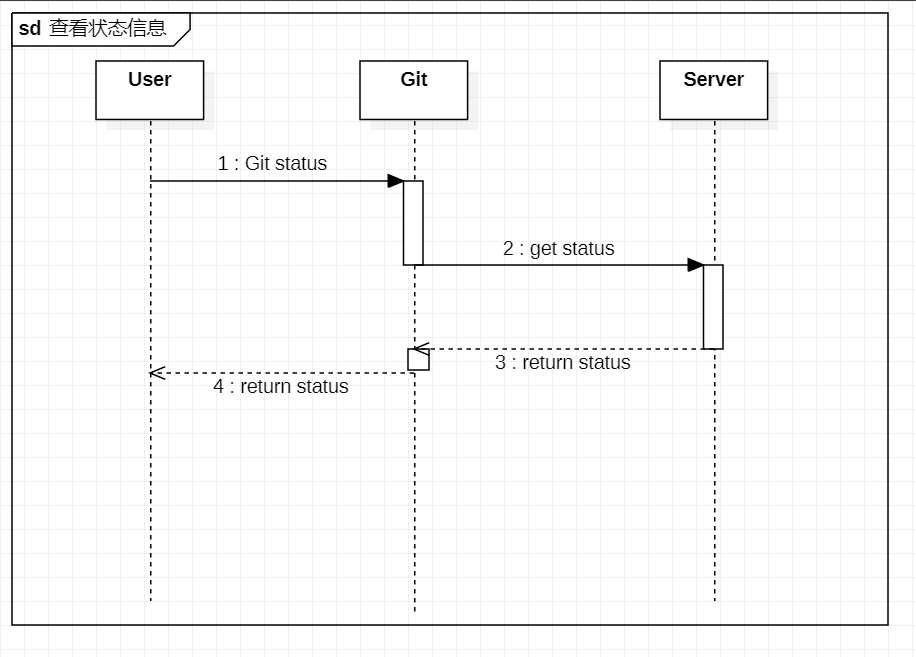
|  |  |
| --- | --- |
| 业务流程 | 1.用户登录or注册进入系统  2.进入到仓库页面  3.   1. 对已有的文件进行修改 2. 创建新文件或上传文件   4.保存相应的文件并提交  5.所有的步骤完成则结束 |
| 建模思路 | 对于用户如果想在GitHub上进行代码托管功能，针对UML中的活动图。  1.首先我们需要考虑的是有哪些主要的活动。如列举的用户登录or注册进入系统、对已有的文件进行修改、创建信息文件或上传文件等等  2.分析这个活动是否有分支的情况。在这个活动图中，我们可以将在仓库中对文件的操作分成（1）对已有的文件进行查看修改、（2）创建新文件并保存  3.将分支的活动进行汇合，如这个活动图中汇合成了提交并保存相应的修改 |

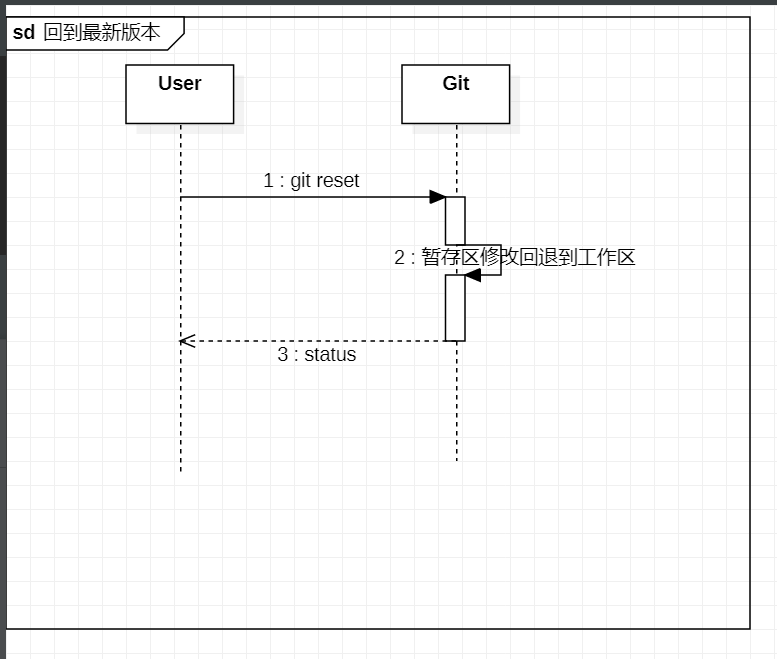
**Author：张子龙 Address：https://zhang-zi-long.github.io/zhangzilong20182123005**

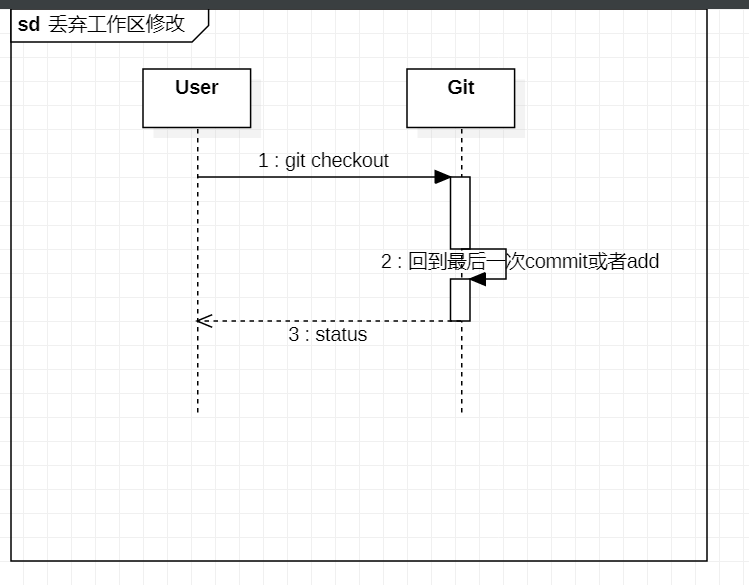
**版本管理功能**

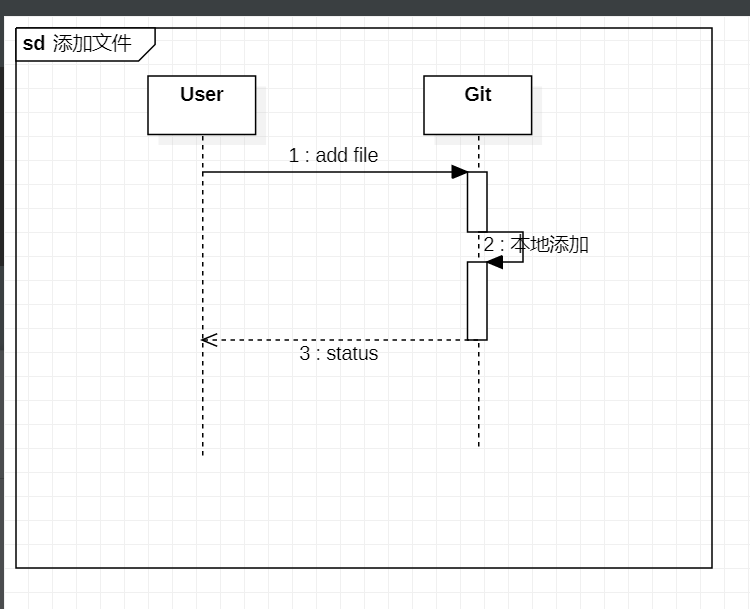
**顺序图展示**

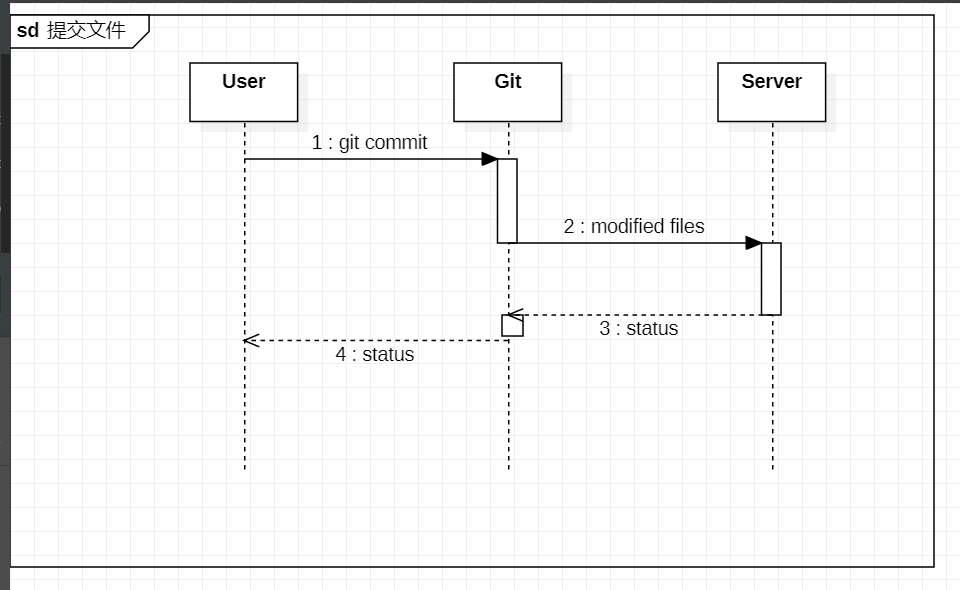


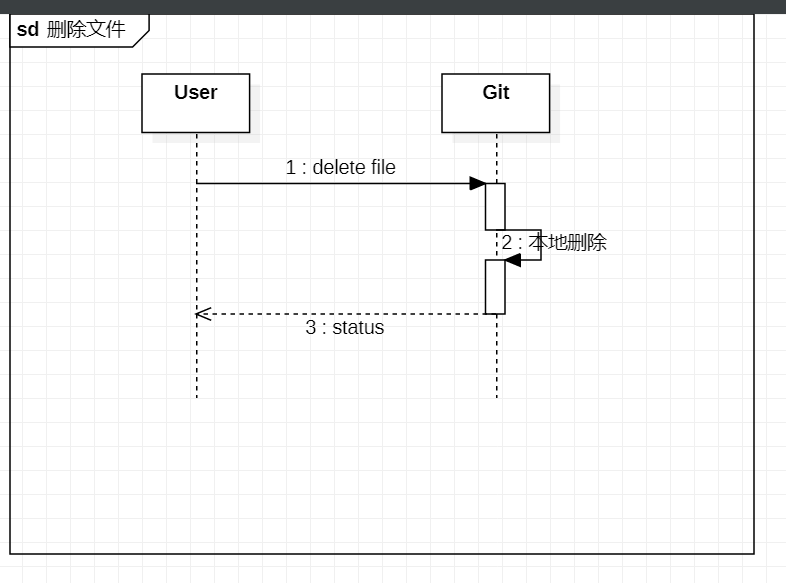


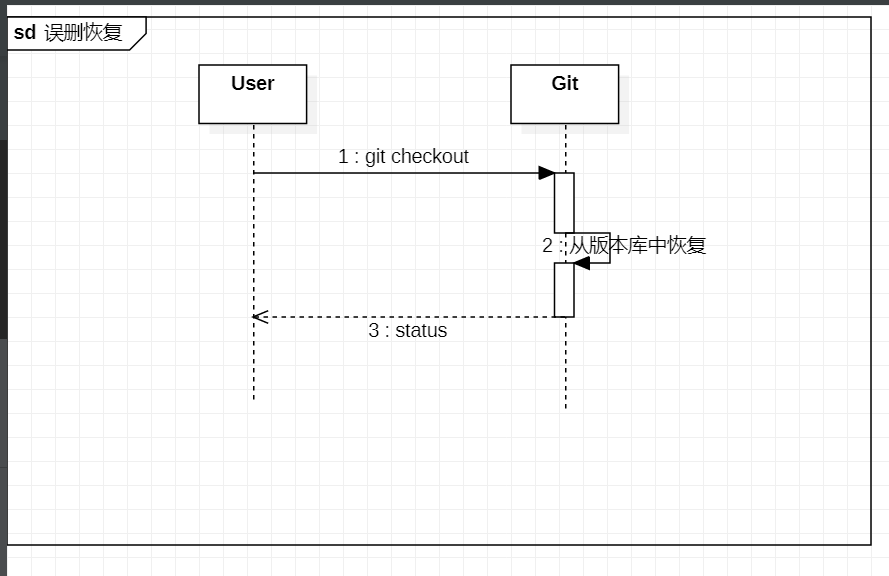


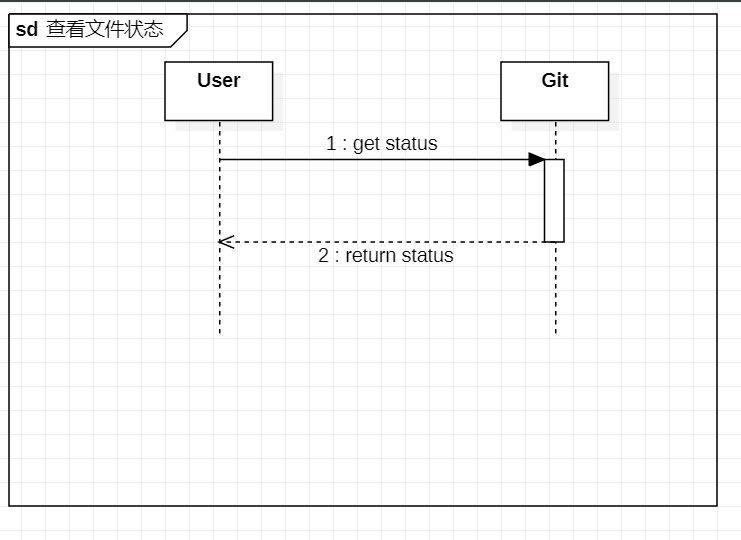












**顺序图说明**

1.管理修改

(1)添加修改

用户使用git添加后，git会将修改内容防护暂存区，在用户commit后，才会将暂存区中的修改提交

(2)查看状态信息

用户使用git获取状态信息，git请求服务器获得状态信息返回给用户

2.撤销修改

(1)回到最新版本

用户使用reset功能，git会将暂存区的修改回退到工作区

(2)丢弃工作区的修改

Git会将当前工作区回到最后一次commit或者add

3.文件管理

(1)添加文件

Git会将文件添加到本地的仓库

(2)提交文件

Git将修改的文件提交给服务器应用

(3)删除文件

Git删除本地的文件

(4)误删恢复

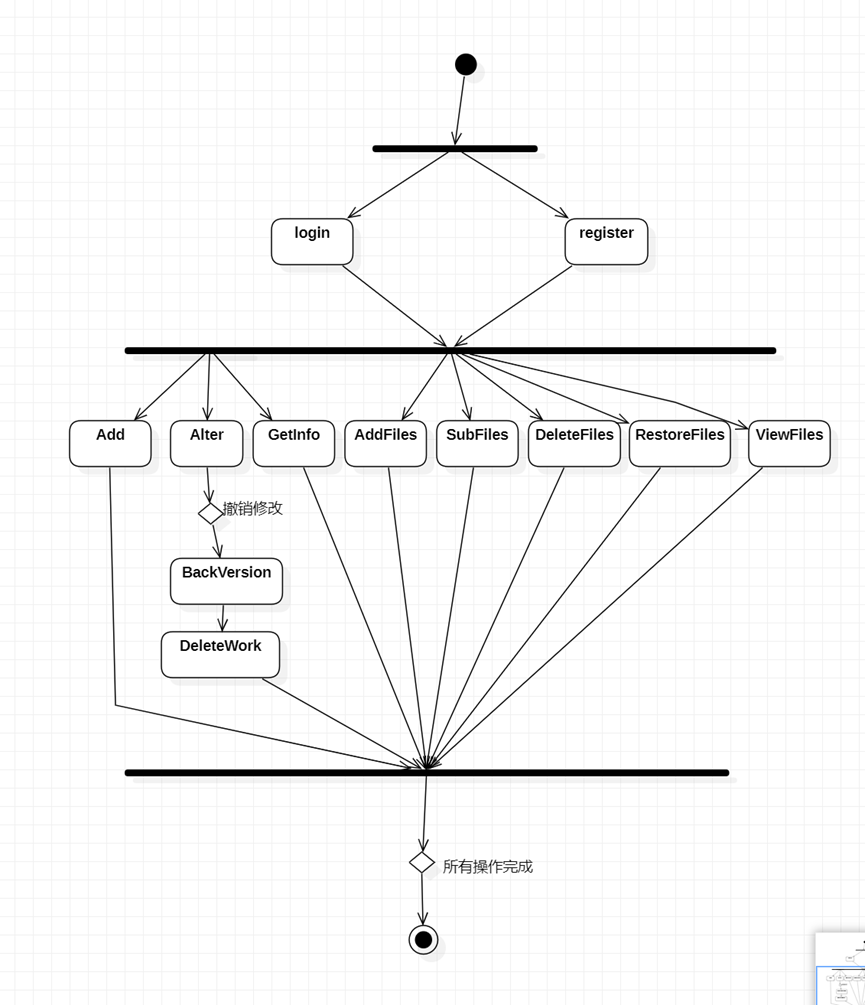
Git将从版本库中恢复指定的文件

(5)查看文件状态

Git将本地的文件状态返回给用户

**Author：张丛祥 Address：wfcpc.github.io**

**活动图展示**

****

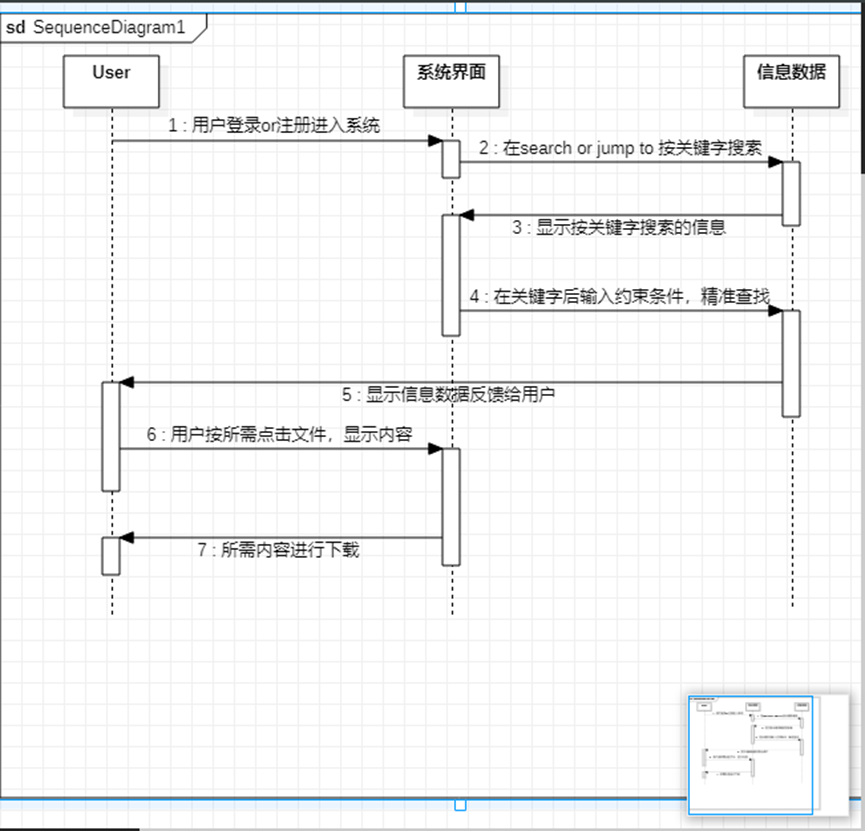
**活动图说明**

|  |  |
| --- | --- |
| 业务流程 | 1. 用户登录或注册进入系统 2. 进入系统后选择一项功能：Add（添加）、Alter（修改）、GetInfo（查看状态）、AddFiles（添加文件）、SubFiles（提交文件）、DeleteFiles（删除文件）、RestoreFiles（恢复文件）、ViewFiles（查看文件状态） 3. 若需要撤销修改，则执行BackVersion（回到最新版本）和DeleteWork（删除工作区修改） 4. 所有操作完成后结束程序 |
| 建模思路 | 对于用户如果想在GitHub上进行版本管理功能，针对UML中的活动图   1. 首先需要考虑有哪些主要活动，如用户登录、用户注册，添加，修改，查看状态等等 2. 分析活动中是否有分支情况。在此活动图中，进入系统后可选择不同的操作，所以可以进行分支，如：Add、Alter、GetInfo…… 3. 当在Alter功能中选择撤销修改时，先进行BackVersion（回到最新版本），再进行DeleteWork（删除工作区修改） 4. 确认所有操作完成后结束程序 |

**Author：张杰 Address：https://github.com/Dummer-2000/zhangjie20182123001**

**代码查找功能**

**顺序图展示**

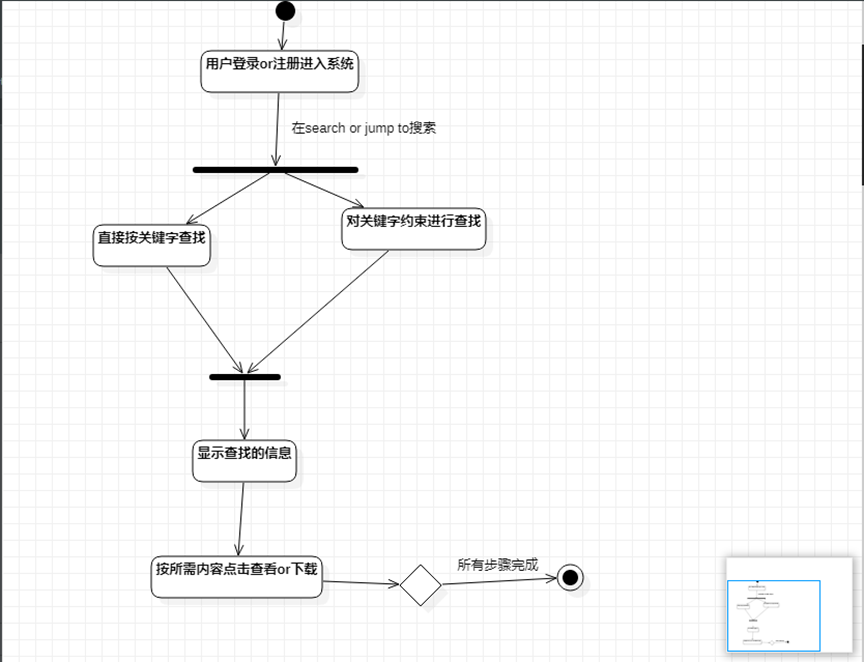
****

**顺序图说明**

|  |  |
| --- | --- |
| 对象 | user、系统界面、信息数据 |
| 业务流程 | 1.用户登录or注册进入系统  2.在search or jump to 按关键字搜索  3.显示按关键字搜索的信息  4.在关键字后输入约束条件，精准查找  5.显示信息数据反馈给用户  6.用户按所需点击文件，显示内容  7.所需内容进行下载 |
| 建模思路 | 对于用户如果想在GitHub上进行代码查找功能，针对UML中的顺序图。  1.首先需要考虑顺序图的对象，分别有user、系统，以及所显示的数据信息。  2.进入系统，然后考虑一些过程，即查找代码的一些消息。  3.在消息中，我们可以在search or jump to 按关键字搜索  4.在输入关键字后，可以发现In this repository、In this user、All GitHub,我们可以按提示的约束进行查找代码，也可以自己在搜索框自己设置一些约束，如star>500等进行约束，这样更加准确。  5.查找之后按照所需打开文件，进行下载代码 |

**Author：冯高杰 Address：https://github.com/FGJtalent/fenggaojie20182123008**

**活动图展示**

****

**活动图说明**

|  |  |
| --- | --- |
| 业务流程 | 1.用户登录or注册进入系统  2.在搜索框进行查找  （1）直接按关键字查找  （2）对关键字进行约束查找  3.对查找后的代码信息进行显示  4.按所需的代码进行点击查看或者下载  5.所有的步骤完成则结束 |
| 建模思路 | 对于用户如果想在GitHub上进行代码查找功能，针对UML中的活动图。  1.首先我们需要考虑的是有哪些主要的活动。如列举的用户登录or注册进入系统、在搜索框进行查找、信息显示等等  2.分析这个活动是否有分支的情况。在这个活动图中，我们可以将在搜索框进行查找进行分流，分成（1）直接按关键字查找、（2）对关键字进行约束查找  3.将分支的活动进行汇合，如这个活动图中汇合成了显示信息  4.对于显示的信息，我们可以查看，或者是下载。 |

**Author：冯高杰 Address：https://github.com/FGJtalent/fenggaojie20182123008**