

**期末项目设计报告**

|  |  |  |  |
| --- | --- | --- | --- |
| 题 目 | 基于GitHub的实验管理平台的详细设计 | | |
| 课程 | 软件系统分析与设计技术 | | |
| 学 院 | 信息科学与工程学院 | | |
| 专 业 | 软件工程 | 年级 | 2016级 |
| 学生姓名 | 王宇 | 学号 | 201610414421 |
| 指导教师 | 赵卫东 | 职称 | 副教授 |

|  |  |  |  |
| --- | --- | --- | --- |
| **评分项** | **评分标准** | **满分** | **得分** |
| 文档整体 | 文档内容详实、规范，美观大方 | 20 |  |
| 用例图及规约 | 用例图完整，准确，能够完全体现需求 | 10 |  |
| 顺序图与活动图 | 顺序图能够完全描述用例的设计思路和业务流程，活动图能够描述部分算法的流程。 | 10 |  |
| 类图 | 类图能够完整，准确反映业务的需数据的组织结构 | 10 |  |
| 数据库 | 数据库表的设计来自于类图，合理 | 10 |  |
| 界面设计 | 界面设计美观，清晰，合理，能够完全反映用例图的设计思路 | 10 |  |
| API接口设计 | 为每个界面设计合理的API接口，以便进行前后端分离开发 | 15 |  |
| 内容一致性 | 用例图，类图、数据库、界面相互印证，相互依赖，环环相扣 | 15 |  |
| **得分合计** | | |  |

2019 年 5 月 1 日

目录

[1.概述 4](#_Toc8224787)

[2. 系统总体结构 4](#_Toc8224788)

[3. 用例图设计 5](#_Toc8224789)

[4. 类图设计 6](#_Toc8224790)

[5. 数据库设计 7](#_Toc8224791)

[5.1 USERS表（用户表） 7](#_Toc8224792)

[5.2 TEACHERS表（老师表） 7](#_Toc8224793)

[5.3 STUDENTS表（学生表） 7](#_Toc8224794)

[5.4 GRADES表（学生实验成绩表） 8](#_Toc8224795)

[5.5 TESTS表（实验项目表） 8](#_Toc8224796)

[5.6 COURSES表（课程表） 8](#_Toc8224797)

[6. 用例及界面详细设计 9](#_Toc8224798)

[6.1“学生列表”用例 9](#_Toc8224799)

[6.1.1 用例规约 9](#_Toc8224800)

[6.1.2 业务流程（顺序图） 9](#_Toc8224801)

[6.1.3. 界面设计 10](#_Toc8224802)

[6.1.4 算法描述 11](#_Toc8224803)

[6.1.5 参照表 12](#_Toc8224804)

[6.2 评定成绩 12](#_Toc8224805)

[6.2.1 用例规约 12](#_Toc8224806)

[6.2.2 业务流程（顺序图） 12](#_Toc8224807)

[6.2.3 界面设计 13](#_Toc8224808)

[6.2.4 算法描述 15](#_Toc8224809)

[6.2.5 参照表 16](#_Toc8224810)

[6.3查看成绩 16](#_Toc8224811)

[6.3.1 用例规约 16](#_Toc8224812)

[6.3.2 业务流程（顺序图） 16](#_Toc8224813)

[6.3.3 界面设计 17](#_Toc8224814)

[6.3.4 算法描述 18](#_Toc8224815)

[6.3.5 参照表 18](#_Toc8224816)

[6.4修改密码 18](#_Toc8224817)

[6.4.1 用例规约 18](#_Toc8224818)

[6.4.2 业务流程 19](#_Toc8224819)

[6.4.3 界面设计 19](#_Toc8224820)

[6.4.4 算法描述 20](#_Toc8224821)

[6.4.5 参照表 20](#_Toc8224822)

[6.5 修改用户信息 20](#_Toc8224823)

[6.5.1 用例规约 20](#_Toc8224824)

[6.5.2 业务流程 21](#_Toc8224825)

[6.5.3 界面设计 21](#_Toc8224826)

[6.5.4 算法描述 24](#_Toc8224827)

[6.5.5 参照表 24](#_Toc8224828)

[6.6查看用户信息 24](#_Toc8224829)

[6.6.1用例规约 24](#_Toc8224830)

[6.6.2 业务流程 24](#_Toc8224831)

[6.6.3 界面设计 24](#_Toc8224832)

[6.6.4 算法描述 26](#_Toc8224833)

[6.6.5 参照表 26](#_Toc8224834)

[6.7登出 26](#_Toc8224835)

[6.7.1用例规约 26](#_Toc8224836)

[6.7.2 业务流程 27](#_Toc8224837)

[6.7.3 界面设计 27](#_Toc8224838)

[6.7.4 算法描述 28](#_Toc8224839)

[6.7.5 参照表 28](#_Toc8224840)

[6.8登录 28](#_Toc8224841)

[6.8.1用例规约 28](#_Toc8224842)

[6.8.2 业务流程 29](#_Toc8224843)

[6.8.3 界面设计 29](#_Toc8224844)

[6.8.4 算法描述 30](#_Toc8224845)

[6.8.5 参照表 31](#_Toc8224846)

[6.9选择课程 31](#_Toc8224847)

[6.9.1用例规约 31](#_Toc8224848)

[6.9.2 业务流程 31](#_Toc8224849)

[6.9.3 界面设计 32](#_Toc8224850)

[6.9.4 算法描述 34](#_Toc8224851)

[6.9.5 参照表 34](#_Toc8224852)

[6.10添加课程 34](#_Toc8224853)

[6.10.1用例规约 34](#_Toc8224854)

[6.10.2 业务流程 35](#_Toc8224855)

[6.10.3 界面设计 35](#_Toc8224856)

[6.10.4 算法描述 37](#_Toc8224857)

[6.10.5 参照表 37](#_Toc8224858)

[6.11 添加用户 37](#_Toc8224859)

[6.11.1 用例规约 37](#_Toc8224860)

[6.11.2 业务流程 38](#_Toc8224861)

[6.11.3 界面设计 38](#_Toc8224862)

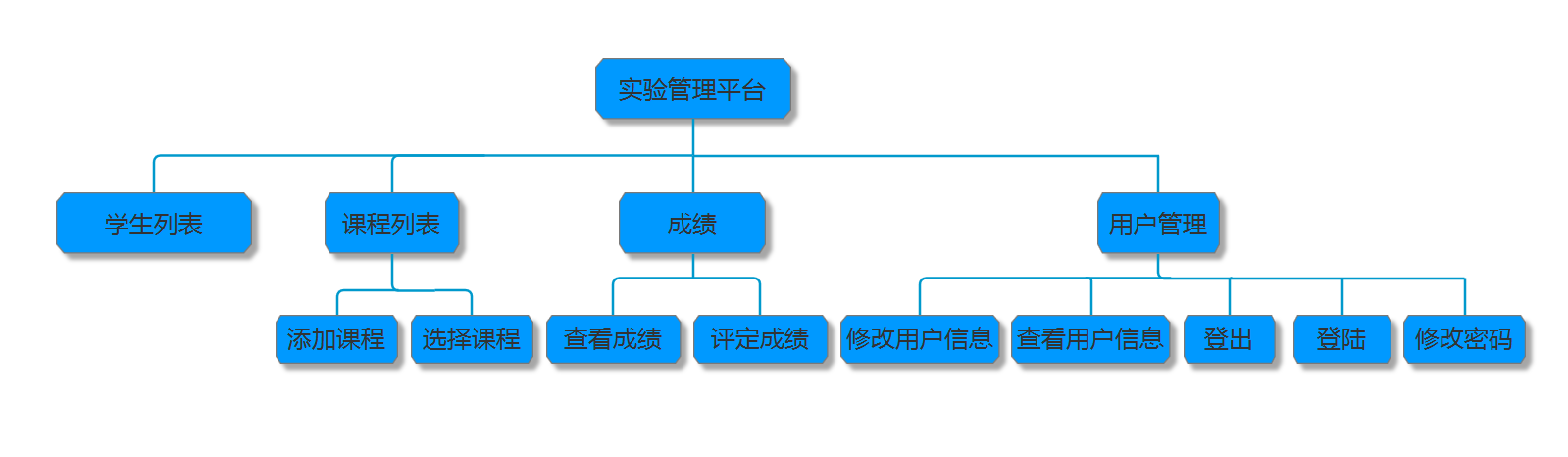
[6.11.4 算法描述 39](#_Toc8224863)

[6.11.5 参照表 39](#_Toc8224864)

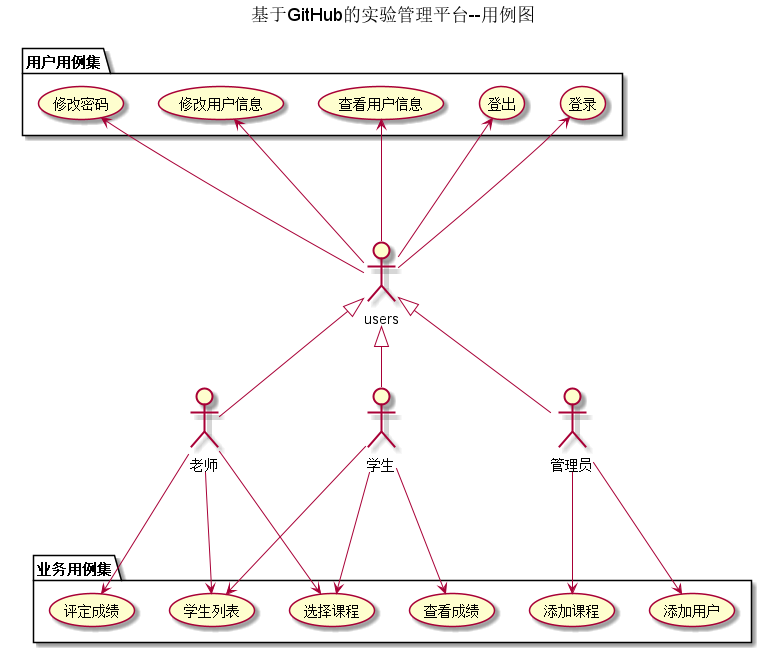
# 概述

1. 基于GitHub的实验管理平台的作用是在线管理实验成绩的Web应用系统。学生和老师的实验内容均存放在GitHUB  
   页面上。
2. 学生的功能主要有：一是设置自己的GitHub用户名，二是查询自己的实验成绩，三是选择课程。学生的GitHub用户名是公开的，但成绩不公开。
3. 老师的功能主要有：一是批改每个学生的成绩，二是查看每个学生的成绩，三是挑选选择了自己所教课程的学生。
4. 老师和学生都能通过本系统的链接方便地跳转到学生的每个GitHUB实验目录，以便批改实验或者查看实验情况。
5. 实验成绩按数字分数计算，每项实验的满分为100分，最低为0分。  
   每个实验的实验成绩细分为多个评分项，每个评分项对应各自的评分标准。
6. 老师在批改实验的时候，对每个评分项进行评分并输入对应的文字评价，系统自动计算出所有评分项的成绩之和为该实验的总成绩。  
   系统自动计算每个学生的所有实验的平均分。

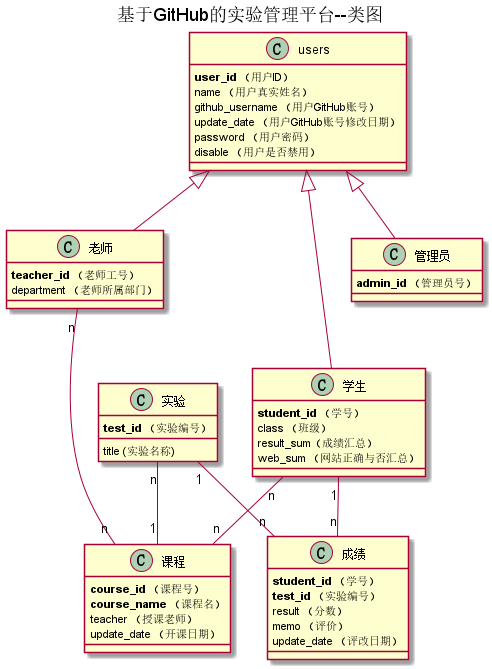
# 系统总体结构



# 用例图设计

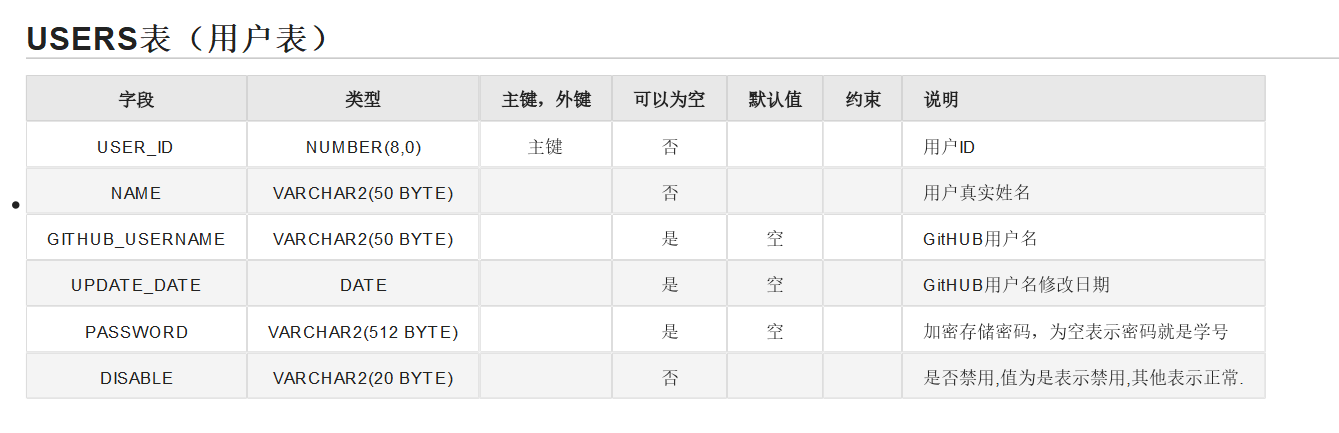


# 类图设计

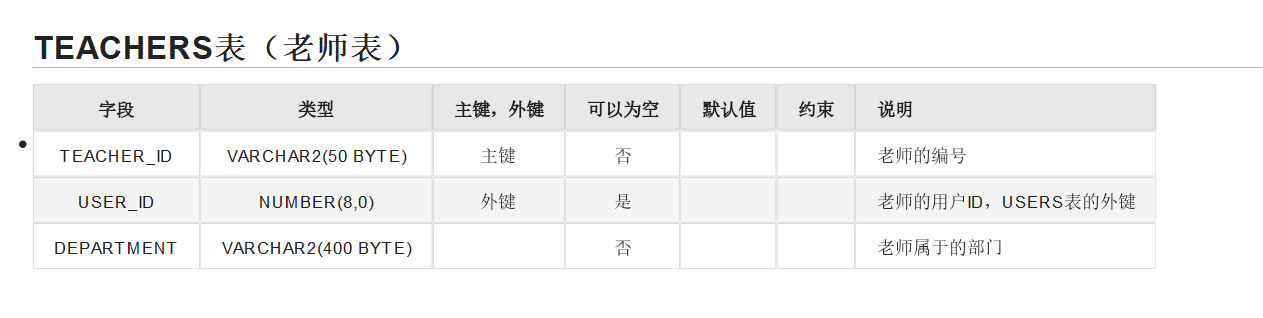


# 数据库设计

## 5.1 USERS表（用户表）



## TEACHERS表（老师表）



## STUDENTS表（学生表）



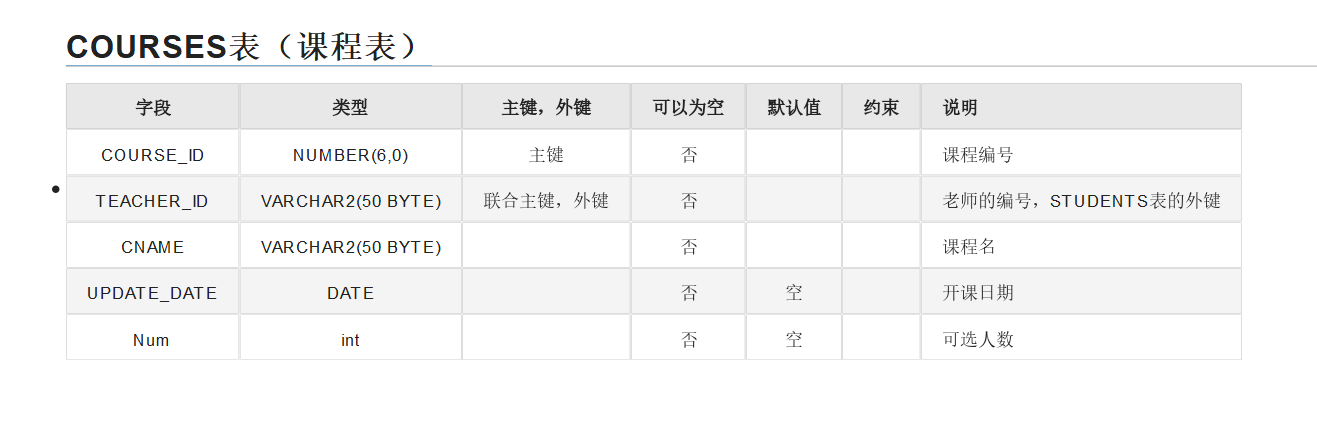
## GRADES表（学生实验成绩表）



## TESTS表（实验项目表）



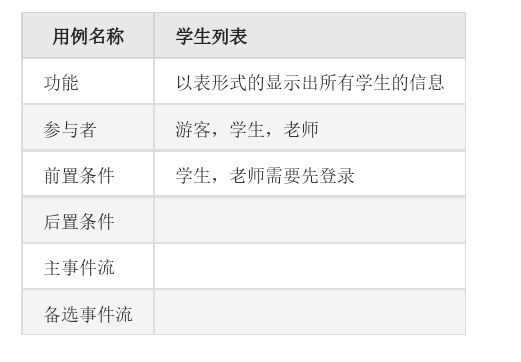
## COURSES表（课程表）



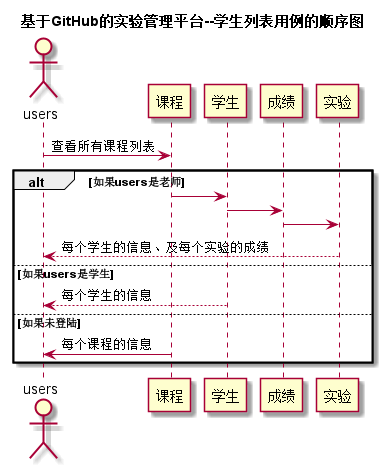
# 用例及界面详细设计

## 6.1“学生列表”用例

6.1.1 用例规约



6.1.2 业务流程（顺序图）



6.1.3. 界面设计



**接口调用：**

**# 接口：getStudents   
-** 权限：  
 学生/访客：不能看到RESULT\_SUM，WEB\_SUM  
 老师：可以看到RESULT\_SUM，WEB\_SUM。  
**-** 功能：  
 返回所有学生的列表。  
**-** API请求地址：  
 接口基本地址/v1/api/getStudents  
**-** 请求方式 ：  
 GET  
**-** 请求参数说明:  
 无  
**-** 返回实例：  
 {  
 "status": true,  
 "info": null,  
 "total": 121,  
 "data": [  
 {"WEB\_SUM": "Y,Y,Y,Y,Y,N",  
 "RESULT\_SUM": "83.75,90,80,80,85,N",  
 "GITHUB\_USERNAME": "Chinajuedui",  
 "STUDENT\_ID": "201510315203",  
 "CLASS": "软件(本)15-1",  
 "NAME": "陈松华",  
 "UPDATE\_DATE": "2018-04-02 13:48:01"},  
 {  
 ...其他学生  
 }  
 ]  
 }  
  
**-** 返回参数说明：  


6.1.4 算法描述

* WEB\_SUM解析为列表
  + WEB\_SUM是接口getStudents的返回值中的一部分，形如："WEB\_SUM": "Y,Y,Y,Y,Y,N"。需要在前端进行解析。
  + 作用是判断GitHub网址是否正确，用逗号分开，Y代表正确，N代表不正确。
  + 第1位代表总的GitHUB地址是否正确，第2位表示第1次实验的地址，第3位表示第2位实验地址，依此类推。
  + 比如：“Y,Y,Y,Y,Y,N”表示第5次实验地址不正确，其他地址正确。
  + 算法是使用使用字符串分离方法将其分开。
* RESULT\_SUM解析为列表
  + RESULT\_SUM是接口getStudents的返回值中的一部分，形如："RESULT\_SUM": "81.25,70,80,85,90,N"。需要在前端进行解析。
  + 成绩汇总（来自GRADES表），以逗号分开，第一个成绩是平均成绩,后面是每次实验的成绩，N表示未批改，平均分只计算已批改的。
  + 比如：“81.25,70,80,85,90,N”表示一共批改了4次，第5次未批改，4次的成绩分别是81.25,70,80,85,90,N，4次的平均分是81.25
  + 算法是使用使用字符串分离方法将其分开。

****6.1.5 参照表****

**STUDENTS**

**GRADES**

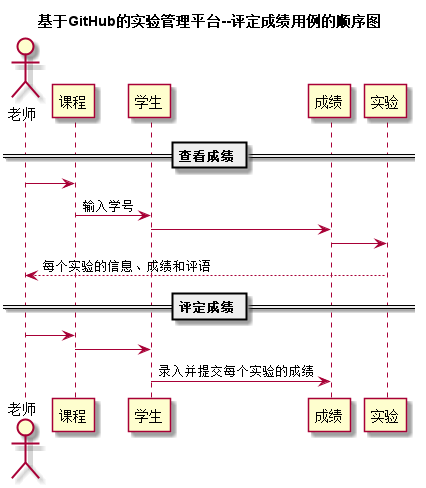
**TESTS**

## 评定成绩

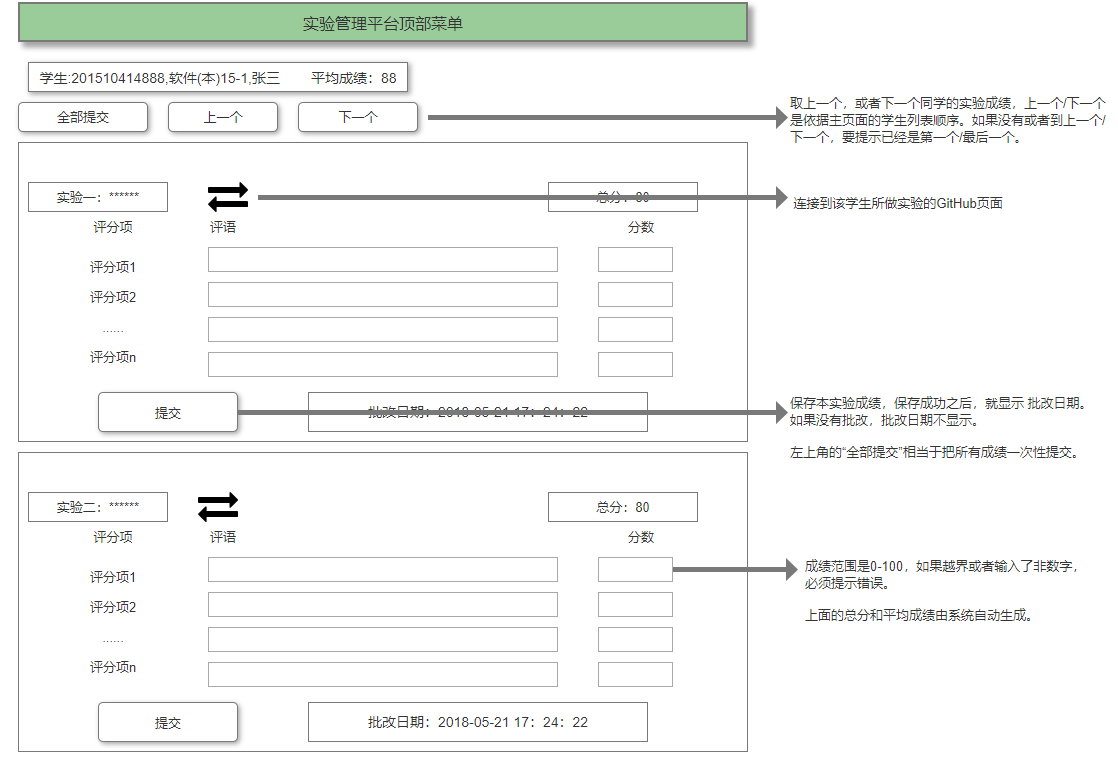
* + 1. 用例规约



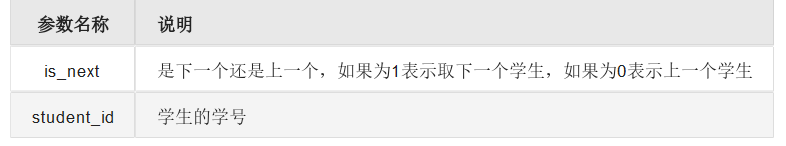
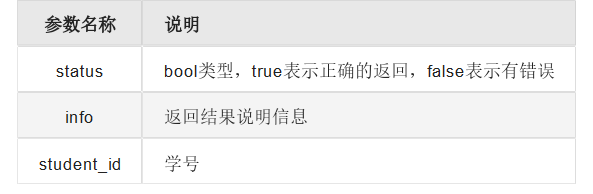
* + 1. 业务流程（顺序图）



* + 1. 界面设计



接口调用：

接口1： getNextPrevStudent  **-** 功能：  
 返回一个学生的上一个或者下一个学生的学号。  
**-** 权限：   
 老师：只有老师可以调用该API。  
**-** API请求地址：   
 接口基本地址/v1/api/getNextPrevStudent/<is\_next>/<student\_id>  
**-** 请求方式 ：  
 GET  
**-** 请求参数说明:   
  **-** 返回实例：  
 {   
 "status": true,  
 "info": null,   
 "student\_id": "201510315203"  
 }  
**-** 返回参数说明：   


接口2：getOneStudentResults  **-** 功能：  
 返回一个学生的所有实验成绩和实验评价。  
**-** 权限：  
 学生：只能查看自己的成绩，即接口参数student\_id必须等于登录学生的student\_id  
 老师：可以查看所有学生的成绩。  
**-** API请求地址：   
 接口基本地址/v1/api/getOneStudentResults/<student\_id>  
**-** 请求方式 ：

GET **-** 返回实例：   
 {   
 "status": true,  
 "info": null,   
 "student\_id": "201510315203",   
 "github\_username": "chinajuedui",   
 "class": "软件(本)15-1",   
 "name": "陈松华",   
 "total": 6,  
 "avgresult":90.5,   
 "data": [  
 {  
 "test\_id":1,  
 "web\_exists": true,   
 "result": 91,   
 "memo":"本实验做得好",  
 "update\_date": "2018-04-02 13:48:01"  
 },   
 {  
 ...其他实验  
 }  
 ]   
 }  
**-** 返回参数说明：   


* + 1. 算法描述

无

6.2.5 参照表

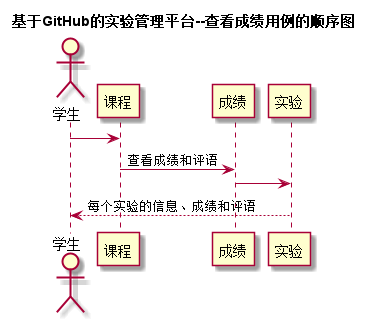
* STUDENTS
* GRADES
* TESTS

## 6.3查看成绩

6.3.1 用例规约



6.3.2 业务流程（顺序图）



6.3.3 界面设计



接口调用：

接口：getOneStudentResults  **-** 功能：  
 返回一个学生的所有实验成绩和实验评价。  
**-** 权限：  
 学生：只能查看自己的成绩，即接口参数student\_id必须等于登录学生的student\_id  
 老师：可以查看所有学生的成绩。  
**-** API请求地址：   
 接口基本地址/v1/api/getOneStudentResults/<student\_id>  
**-** 请求方式 ：

GET **-** 返回实例：   
 {   
 "status": true,  
 "info": null,   
 "student\_id": "201510315203",   
 "github\_username": "chinajuedui",   
 "class": "软件(本)15-1",   
 "name": "陈松华",   
 "total": 6,  
 "avgresult":90.5,   
 "data": [  
 {  
 "test\_id":1,  
 "web\_exists": true,   
 "result": 91,   
 "memo":"本实验做得好",  
 "update\_date": "2018-04-02 13:48:01"  
 },   
 {  
 ...其他实验  
 }  
 ]   
 }  
**-** 返回参数说明：   


6.3.4 算法描述

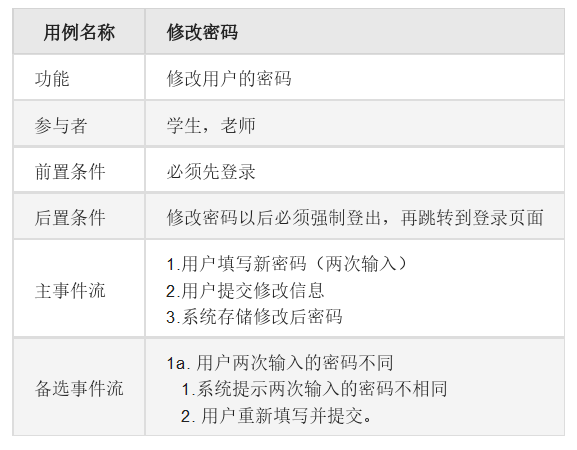
无

6.3.5 参照表

* STUDENTS
* GRADES
* TESTS

## 6.4修改密码

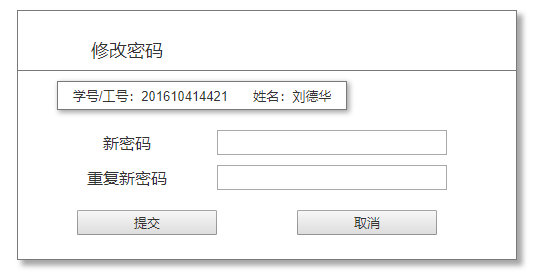
6.4.1 用例规约



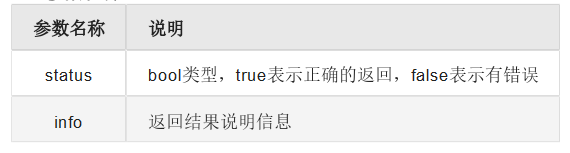
6.4.2 业务流程

无

6.4.3 界面设计



接口调用：

接口：setPassword  
- 功能：  
 修改（设置）用户的密码。  
- 权限：  
 学生/老师：修改自己的密码，必须先登录。   
- API请求地址：   
 接口基本地址/v1/api/setPassword  
- 请求方式 ：  
 POST  
- 请求实例：   
 {  
 "user\_id":1234,  
 "password":AF#W@#AAAASDF  
 }  
- 请求参数说明:   
  
- 返回实例：   
 {   
 "status": true,  
 "info": null,   
 }  
- 返回参数说明：   


6.4.4 算法描述

无

6.4.5 参照表

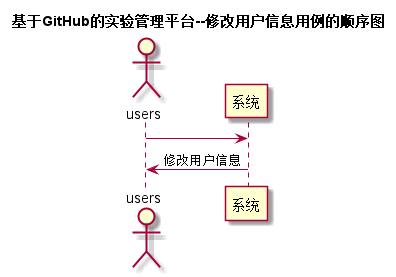
* USERS

## 6.5 修改用户信息

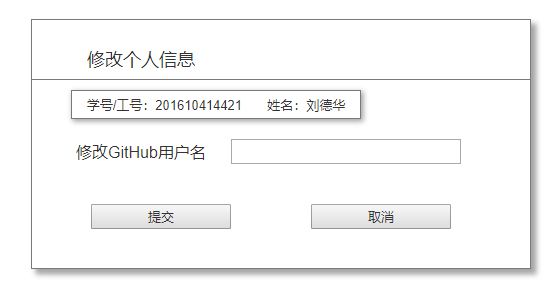
6.5.1 用例规约



6.5.2 业务流程

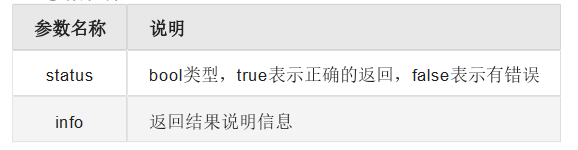


6.5.3 界面设计



接口调用：

接口：getUserInfo **-** 功能：  
 查看用户的所有信息。  
**-** 权限：  
 学生/老师：查看自己的信息，必须先登录，不能查看其他用户的信息。   
**-** API请求地址：   
 接口基本地址/v1/api/getUserInfo/<user\_id>  
**-** 请求方式 ：  
 GET  
**-** 请求参数说明:   
 **-** 返回实例：   
 {   
 "status": true,  
 "info": null,  
 "ID":"2101201281",   
 "name":"张三",  
 "class\_dept":"软件工程1班"  
 "github\_username":"ABCD",  
 "type":"学生"   
 }  
**-** 返回参数说明：   


接口：setUserInfo  **-** 功能：  
 修改用户的GitHub用户名。  
**-** 权限：  
 学生/老师：修改自己的密码，必须先登录。   
**-** API请求地址：   
 接口基本地址/v1/api/setUserInfo  
**-** 请求方式 ：  
 POST  
**-** 请求实例：   
 {  
 "id":"21048329823",  
 "github\_username":"ABCDE",   
 }  
   
**-** 请求参数说明:   
 **-** 返回实例：   
 {   
 "status": true,  
 "info": null  
 }  
**-** 返回参数说明：   


6.5.4 算法描述

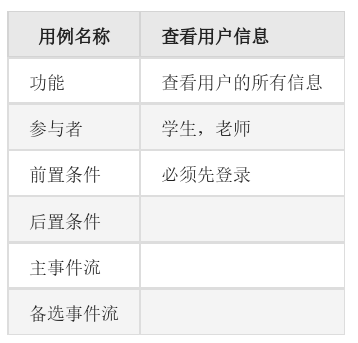
无

6.5.5 参照表

* USERS

## 6.6查看用户信息

6.6.1用例规约



6.6.2 业务流程

无

6.6.3 界面设计



接口调用：

接口：getUserInfo **-** 功能：  
 查看用户的所有信息。  
**-** 权限：  
 学生/老师：查看自己的信息，必须先登录，不能查看其他用户的信息。   
**-** API请求地址：   
 接口基本地址/v1/api/getUserInfo/<user\_id>  
**-** 请求方式 ：  
 GET  
**-** 请求参数说明:   
 **-** 返回实例：   
 {   
 "status": true,  
 "info": null,  
 "ID":"2101201281",   
 "name":"张三",  
 "class\_dept":"软件工程1班"  
 "github\_username":"ABCD",  
 "type":"学生"   
 }  
**-** 返回参数说明：   


6.6.4 算法描述

无

6.6.5 参照表

* STUDENTS
* TEACHERS
* USERS

## 6.7登出

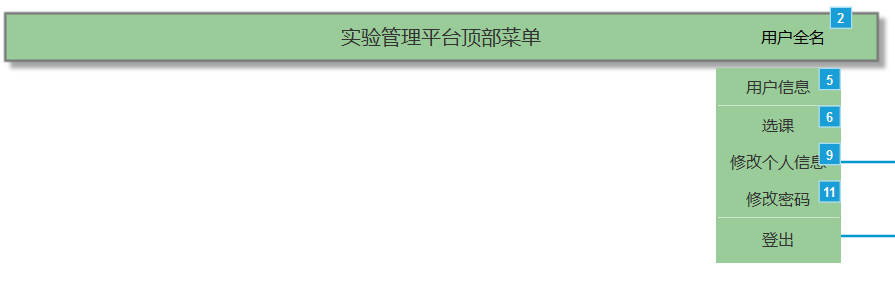
6.7.1用例规约



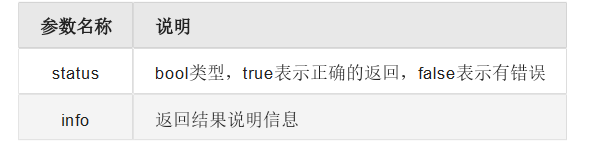
6.7.2 业务流程

无

6.7.3 界面设计



接口调用：

接口：logout   
- 功能：  
 已经登录的用户登出平台。  
- 权限：  
 学生/老师已经登录。   
- API请求地址：   
 接口基本地址/v1/api/logout  
- 请求方式 ：  
 POST  
- 请求实例：   
 {  
 "user\_id":324111  
 }  
- 请求参数说明:   
  
- 返回实例：   
 {   
 "status": true,  
 "info": null,   
 }  
- 返回参数说明：   
   
 

6.7.4 算法描述

无

6.7.5 参照表

* USERS

## 6.8登录

6.8.1用例规约



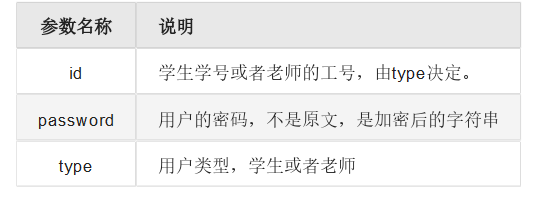
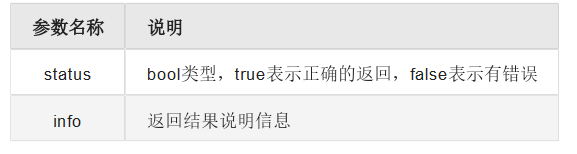
6.8.2 业务流程

无

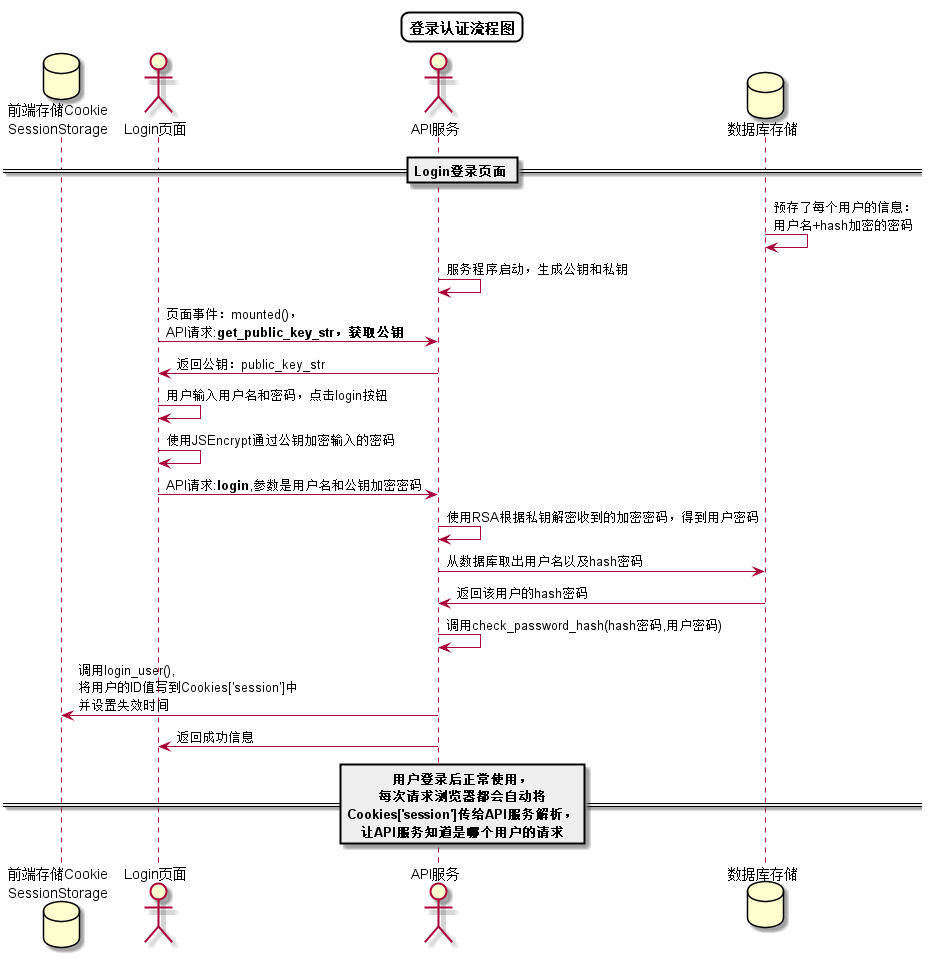
6.8.3 界面设计



接口调用：

接口：login [返回](../README.md)  
用例： [登录](../用例/登录用例.md)   
- 功能：  
 登录到平台。  
- 权限：  
 访客。   
- API请求地址：   
 接口基本地址/v1/api/login  
- 请求方式 ：  
 POST  
- 请求实例：   
 {  
 "id":"21048329823",  
 "password":"ABCDE",  
 "type":"学生"  
 }  
- 请求参数说明:   
  
- 返回实例：   
 {   
 "status": true,  
 "info": null,   
 }  
- 返回参数说明：   


6.8.4 算法描述



6.8.5 参照表

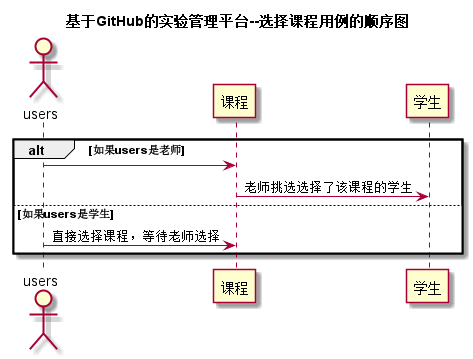
* USERS

## 6.9选择课程

6.9.1用例规约



6.9.2 业务流程



6.9.3 界面设计







接口调用：

接口：selectCourses  
- 功能：  
 返回课程列表   
- 权限：  
 学生：可以选择所有课程  
 老师：只有老师只能挑选已经选择了该课程的学生。  
- API请求地址：   
 接口基本地址/v1/api/selectCourses/<course\_id>/<student\_id>  
- 请求方式 ：  
 GET  
- 请求参数说明:   
   
- 返回实例：   
 {   
 "info": ture,   
 "course\_id": "00001"   
 "student\_id": "201510315203"  
 }  
- 返回参数说明：   
 

6.9.4 算法描述

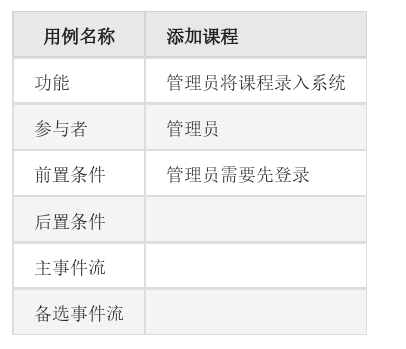
无

6.9.5 参照表

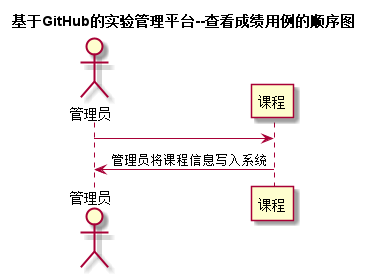
* STUDENTS
* TEACHERS
* COURSES

## 6.10添加课程

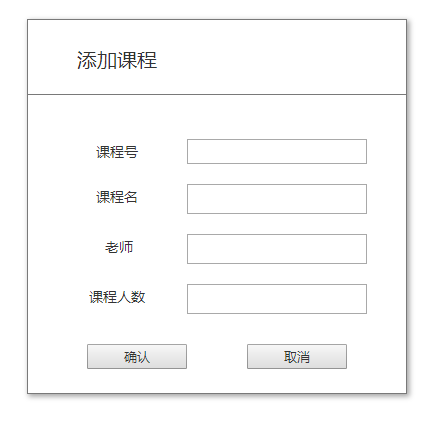
6.10.1用例规约



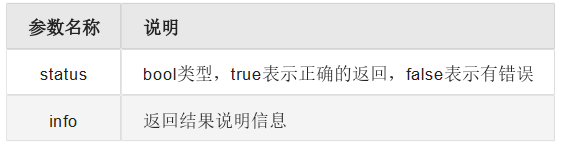
6.10.2 业务流程



6.10.3 界面设计



接口调用：

接口：addCourse  **-** 功能：  
 将课程信息添加进数据库。  
**-** 权限：  
 管理员。   
**-** API请求地址：   
 接口基本地址/v1/api/addCourse  
**-** 请求方式 ：  
 POST  
  
**-** 请求实例：   
 {  
 "course\_id":"000001",  
 "Cname":"计算机网络",  
 "teacher\_id":"21231356651"  
 }   
**-** 请求参数说明:   
 **-** 返回实例：   
 {   
 "status": true,  
 "info": null,   
 }  
**-** 返回参数说明：   
 

6.10.4 算法描述

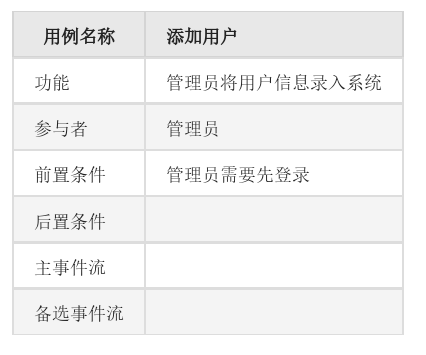
无

6.10.5 参照表

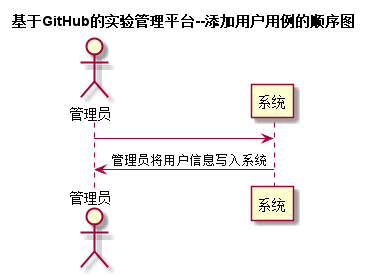
* COURSES
* TEACHERS

## 6.11 添加用户

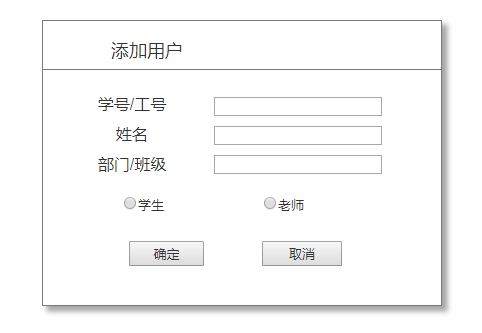
6.11.1 用例规约



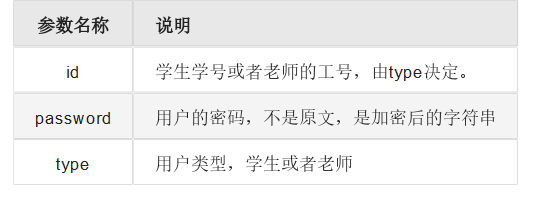
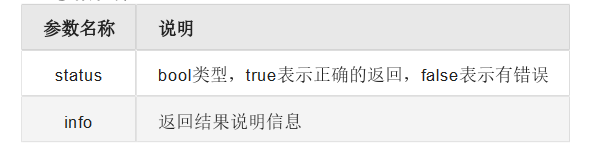
6.11.2 业务流程



6.11.3 界面设计



接口调用：

接口：addUsers   
- 功能：  
 将用户信息添加进数据库。  
- 权限：  
 管理员。   
- API请求地址：   
 接口基本地址/v1/api/addUsers  
- 请求方式 ：  
 POST  
- 请求实例：   
 {  
 "id":"21048329823",  
 "password":"ABCDE",  
 "type":"学生"  
 }  
- 请求参数说明:   
  
- 返回实例：   
 {   
 "status": true,  
 "info": null,   
 }  
- 返回参数说明：   


6.11.4 算法描述

无

6.11.5 参照表

* USERS