

**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 | 穆峥言 | 学号 | 201610414414 |
| 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

**[一． 概述](#_Toc8835_WPSOffice_Level1)** **[4](#_Toc8835_WPSOffice_Level1)**

**[二． 系统总体结构](#_Toc6227_WPSOffice_Level1)** **[4](#_Toc6227_WPSOffice_Level1)**

**[三． 系统用例图设计](#_Toc17739_WPSOffice_Level1)** **[4](#_Toc17739_WPSOffice_Level1)**

**[四． 系统类图设计](#_Toc10648_WPSOffice_Level1)** **[5](#_Toc10648_WPSOffice_Level1)**

**[五． 数据库设计](#_Toc4164_WPSOffice_Level1)** **[7](#_Toc4164_WPSOffice_Level1)**

[1.USERS表(用户表)](#_Toc6227_WPSOffice_Level2) [7](#_Toc6227_WPSOffice_Level2)

[2.STUDENT表(学生表)](#_Toc17739_WPSOffice_Level2) [8](#_Toc17739_WPSOffice_Level2)

[3.TEACHER表(教师表)](#_Toc10648_WPSOffice_Level2) [9](#_Toc10648_WPSOffice_Level2)

[4.ADMIN表(管理员表)](#_Toc4164_WPSOffice_Level2) [11](#_Toc4164_WPSOffice_Level2)

[5.LESSON表(课程表)](#_Toc25072_WPSOffice_Level2) [11](#_Toc25072_WPSOffice_Level2)

[6.TEST表(实验表)](#_Toc23181_WPSOffice_Level2) [13](#_Toc23181_WPSOffice_Level2)

[7.CHOOSELESSON表(选课表)](#_Toc27592_WPSOffice_Level2) [14](#_Toc27592_WPSOffice_Level2)

[8.SCOREITEMS表(评分项表)](#_Toc15128_WPSOffice_Level2) [15](#_Toc15128_WPSOffice_Level2)

[9.SCORE表(单项得分表)](#_Toc10129_WPSOffice_Level2) [16](#_Toc10129_WPSOffice_Level2)

[10.TESTAVERAGE表(实验平均分表)](#_Toc26778_WPSOffice_Level2) [17](#_Toc26778_WPSOffice_Level2)

[11.LESSONAVERAGE表(课程平均分表)](#_Toc4374_WPSOffice_Level2) [18](#_Toc4374_WPSOffice_Level2)

[12.YEARANDTERM表(学年学期表)](#_Toc6157_WPSOffice_Level2) [19](#_Toc6157_WPSOffice_Level2)

**[六． 用例及界面设计](#_Toc25072_WPSOffice_Level1)** **[20](#_Toc25072_WPSOffice_Level1)**

[1. 学生列表用例及界面](#_Toc30808_WPSOffice_Level2) [20](#_Toc30808_WPSOffice_Level2)

[2. 选择课程用例及界面](#_Toc21656_WPSOffice_Level2) [21](#_Toc21656_WPSOffice_Level2)

[3. 设定实验及评分项用例及界面](#_Toc29281_WPSOffice_Level2) [24](#_Toc29281_WPSOffice_Level2)

[4. 学年学期用例](#_Toc17302_WPSOffice_Level2) [25](#_Toc17302_WPSOffice_Level2)

[6. 登录登出用例及界面](#_Toc18872_WPSOffice_Level2) [29](#_Toc18872_WPSOffice_Level2)

[7. 结束课程用例](#_Toc7616_WPSOffice_Level2) [31](#_Toc7616_WPSOffice_Level2)

[8. 维护个人信息用例及界面](#_Toc13137_WPSOffice_Level2) [32](#_Toc13137_WPSOffice_Level2)

[9. 维护学生信息用例及界面](#_Toc11982_WPSOffice_Level2) [36](#_Toc11982_WPSOffice_Level2)

[10. 维护教师信息用例及界面](#_Toc29751_WPSOffice_Level2) [38](#_Toc29751_WPSOffice_Level2)

[11. 维护课程信息用例及界面](#_Toc30480_WPSOffice_Level2) [40](#_Toc30480_WPSOffice_Level2)

[12. 评定成绩](#_Toc17724_WPSOffice_Level2) [42](#_Toc17724_WPSOffice_Level2)

**[七． 接口设计](#_Toc23181_WPSOffice_Level1)** **[44](#_Toc23181_WPSOffice_Level1)**

[1.接口：chooseLesson](#_Toc9925_WPSOffice_Level2) [44](#_Toc9925_WPSOffice_Level2)

[2.接口：deleteLesson](#_Toc28018_WPSOffice_Level2) [45](#_Toc28018_WPSOffice_Level2)

[3.接口：deleteUser](#_Toc10559_WPSOffice_Level2) [47](#_Toc10559_WPSOffice_Level2)

[4.接口：getLessons](#_Toc4422_WPSOffice_Level2) [48](#_Toc4422_WPSOffice_Level2)

[5. 接口：getNextPreStudent](#_Toc26135_WPSOffice_Level2) [50](#_Toc26135_WPSOffice_Level2)

[6. 接口:getOneStudentResult](#_Toc25442_WPSOffice_Level2) [51](#_Toc25442_WPSOffice_Level2)

[7. 接口:getUserInfo](#_Toc20726_WPSOffice_Level2) [53](#_Toc20726_WPSOffice_Level2)

[8. 接口:insertLesson](#_Toc3533_WPSOffice_Level2) [54](#_Toc3533_WPSOffice_Level2)

[9. 接口:login](#_Toc25796_WPSOffice_Level2) [56](#_Toc25796_WPSOffice_Level2)

[10. 接口:logout](#_Toc7391_WPSOffice_Level2) [57](#_Toc7391_WPSOffice_Level2)

[11. 接口:resetPassword](#_Toc19773_WPSOffice_Level2) [58](#_Toc19773_WPSOffice_Level2)

[12. 接口:resetUserInfo](#_Toc17156_WPSOffice_Level2) [60](#_Toc17156_WPSOffice_Level2)

[13. 接口:setLesson](#_Toc23099_WPSOffice_Level2) [61](#_Toc23099_WPSOffice_Level2)

[14. 接口:setOneStudentResults](#_Toc17250_WPSOffice_Level2) [63](#_Toc17250_WPSOffice_Level2)

[15. 接口:studentList](#_Toc29236_WPSOffice_Level2) [64](#_Toc29236_WPSOffice_Level2)

[16. 接口:teacherList](#_Toc733_WPSOffice_Level2) [66](#_Toc733_WPSOffice_Level2)

[17. 接口:updateLesson](#_Toc9656_WPSOffice_Level2) [67](#_Toc9656_WPSOffice_Level2)

[18. 接口:updateUser](#_Toc31247_WPSOffice_Level2) [69](#_Toc31247_WPSOffice_Level2)

1. 概述

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。

学生的功能主要有：一是设置自己的GitHub用户名，二是选择自己的课程，三是查询自己的多个科目的多个实验成绩,学生的GitHub用户名是公开的，但成绩不公开。

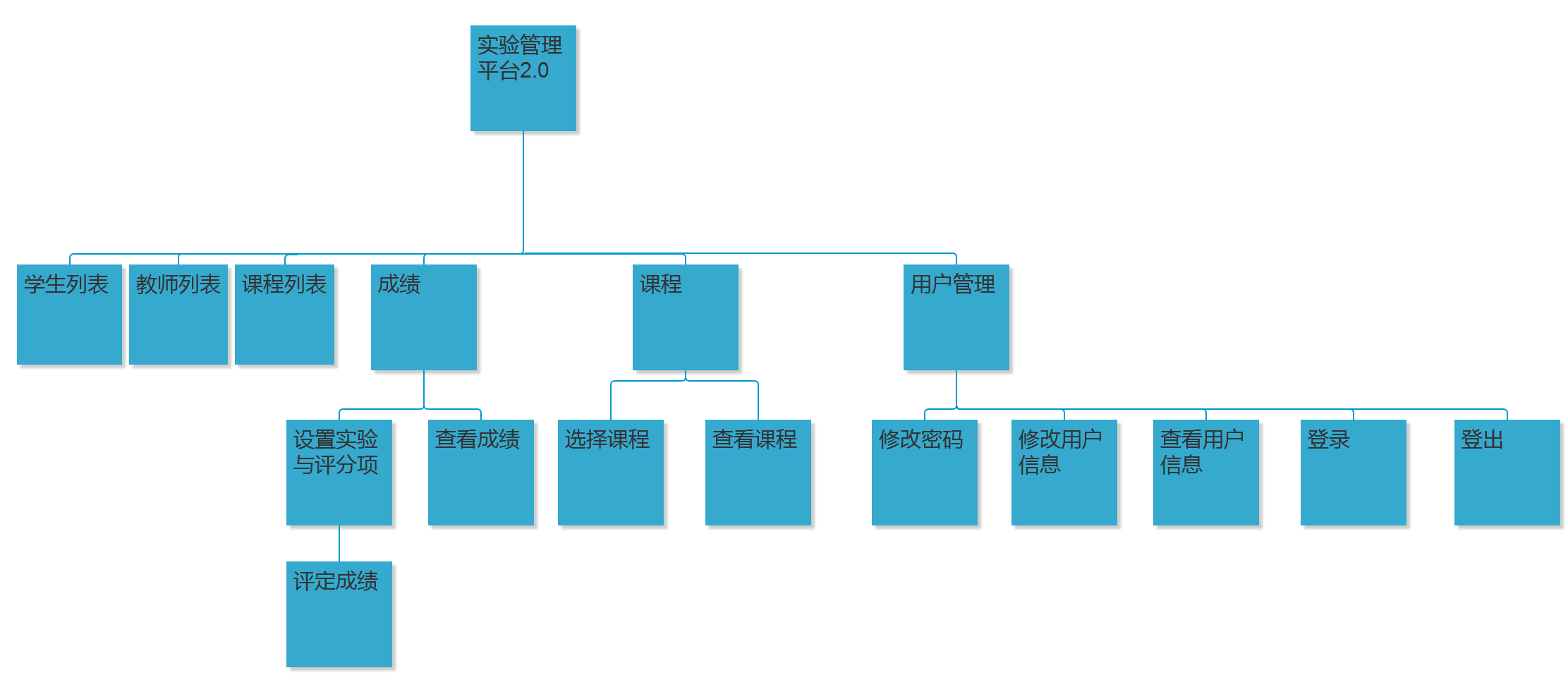
老师的功能主要有：一是设置自己的GitHub用户名，二是选择自己的课程，三是批改自己学生的成绩，四是查看自己学生的成绩。

管理员的功能主要有：一是维护课程信息，二是维护学生信息，三是维护教师信息。

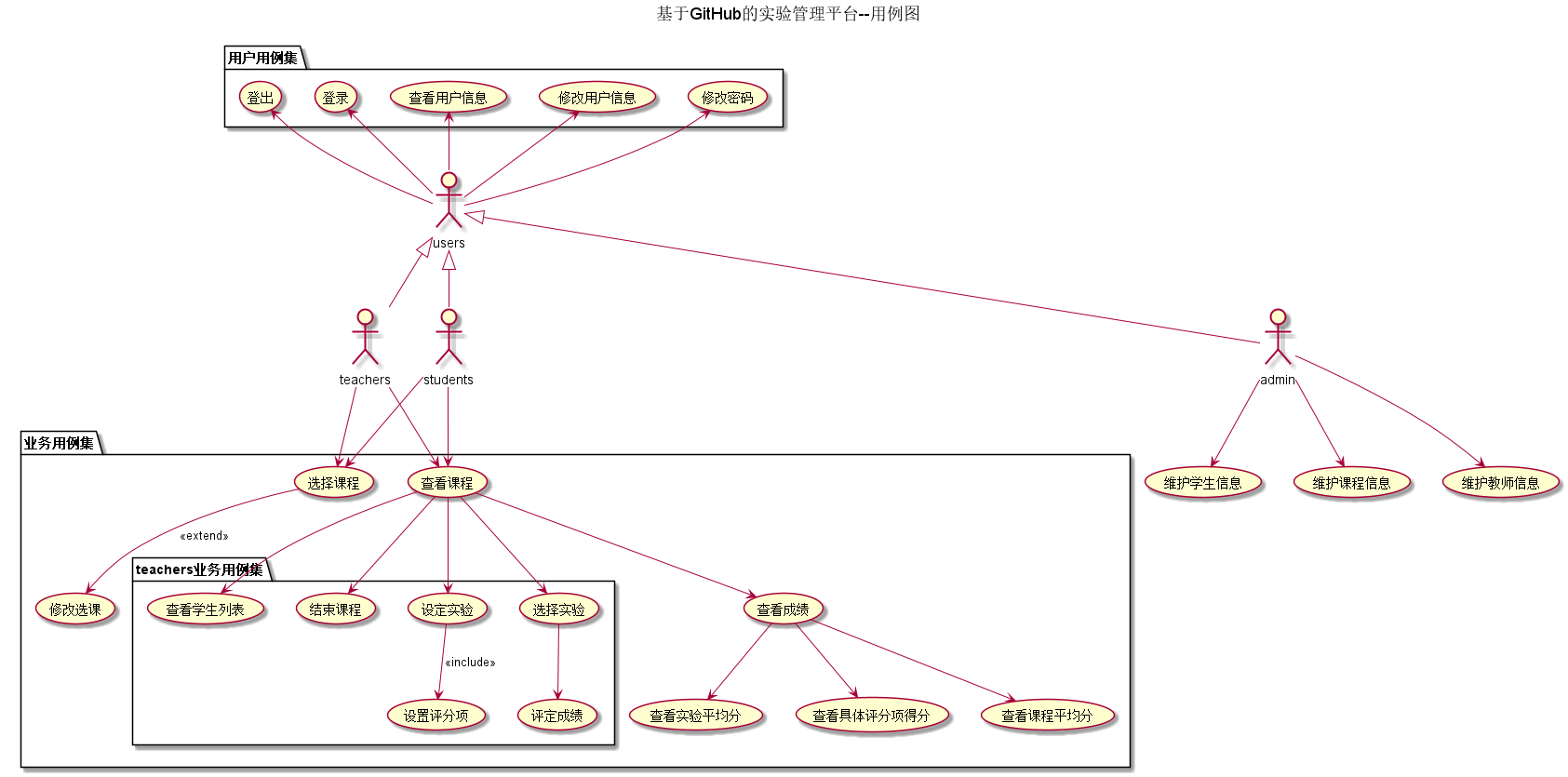
同一个课程的老师和学生都能跳转到对方的GitHub页面，以便批改实验或者查看实验情况。

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。 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。

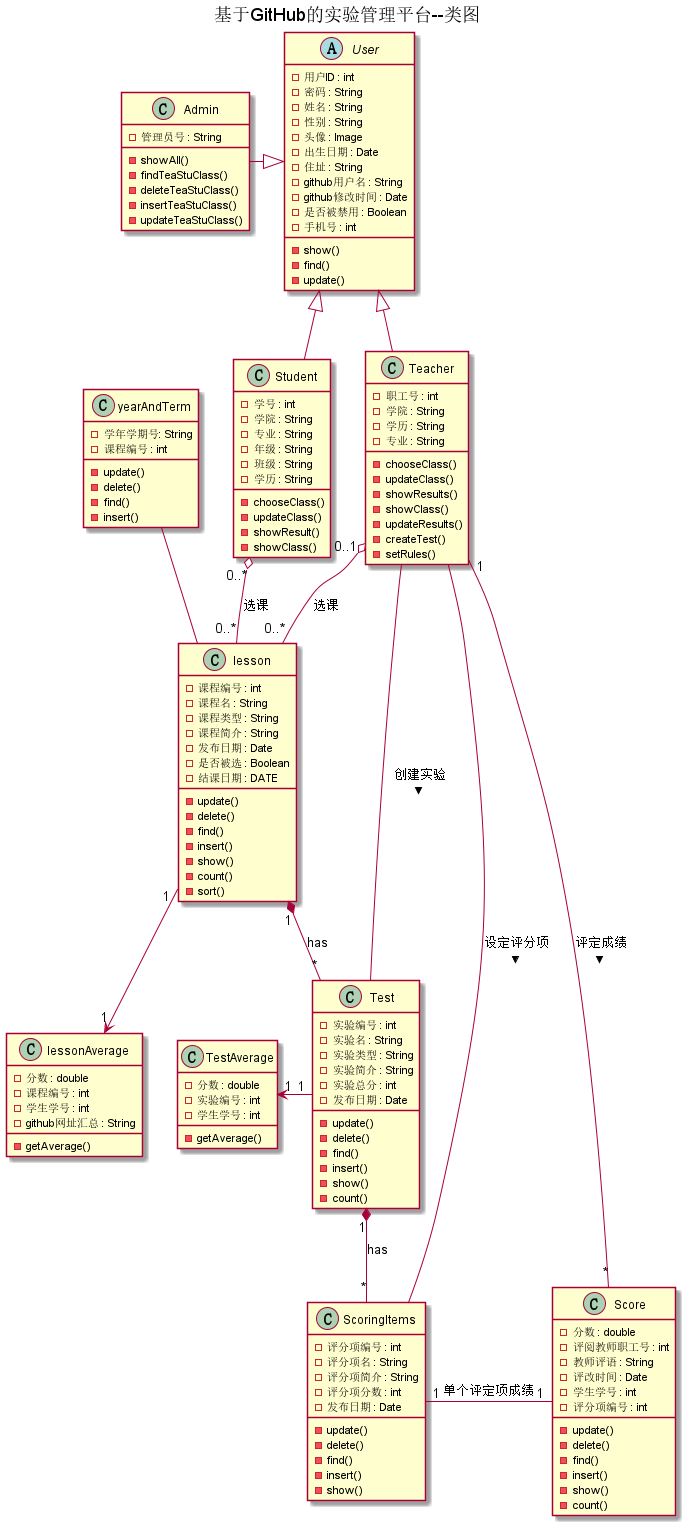
1. 系统总体结构



1. 系统用例图设计



1. 系统类图设计



1. 数据库设计

### **1.USERS表(用户表)**

| **字段** | **类型** | **主键，外键** | **可以为空** | **默认值** | **约束** | **说明** |
| --- | --- | --- | --- | --- | --- | --- |
| user\_id | NUMBER(8,0) | 主键 | 否 |  |  | 用户ID |
| password | VARCHAR2(512 BYTE) |  | 是 | 空 |  | 加密存储密码，为空表示密码就是学号 |
| user\_name | VARCHAR2(50 BYTE) |  | 否 |  |  | 用户真实姓名 |
| sex | VARCHAR2(10 BYTE) |  | 否 | 男 |  | 用户性别 |
| photo | Image |  | 是 | 空 |  | 头像为空表示为默认设置的头像 |
| birthday | DATE |  | 否 |  |  | 出生日期 |
| address | VARCHAR2(512 BYTE) |  | 是 | 空 |  | 为空默认为学校地址 |
| phone | NUMBER(15,0) |  | 否 |  |  | 预留手机号 |

### **2.STUDENT表(学生表)**

| **字段** | **类型** | **主键，外键** | **可以为空** | **默认值** | **约束** | **说明** |
| --- | --- | --- | --- | --- | --- | --- |
| user\_id | NUMBER(8,0) | 主键 | 否 |  |  | 学生学号 |
| student\_department | VARCHAR2(50 BYTE) |  | 否 |  |  | 学生所处学院 |
| student\_major | VARCHAR2(50 BYTE) |  | 否 |  |  | 学生专业 |
| student\_grade | VARCHAR2(50 BYTE) |  | 否 |  |  | 学生年级 |
| student\_class | VARCHAR2(50 BYTE) |  | 否 |  |  | 学生班级 |
| student\_degree | VARCHAR2(10 BYTE) |  | 否 | 本科 |  | 学生学历 |
| github\_username | VARCHAR2(50 BYTE) |  | 是 | 空 |  | GitHUB用户名 |
| update\_date | DATE |  | 是 | 空 |  | GitHUB用户名修改日期 |
| disable | Boolean |  | 否 |  |  | 是否禁用,值为是表示禁用,否表示正常 |

### **3.TEACHER表(教师表)**

| **字段** | **类型** | **主键，外键** | **可以为空** | **默认值** | **约束** | **说明** |
| --- | --- | --- | --- | --- | --- | --- |
| user\_id | NUMBER(8,0) | 主键 | 否 |  |  | 教师职工号 |
| teacher\_department | VARCHAR2(50 BYTE) |  | 否 |  |  | 教师所在学院 |
| teacher\_degree | VARCHAR2(10 BYTE) |  | 否 |  |  | 教师学历 |
| teacher\_major | VARCHAR2(10 BYTE) |  | 否 |  |  | 教师专业 |
| github\_username | VARCHAR2(50 BYTE) |  | 是 | 空 |  | GitHUB用户名 |
| update\_date | DATE |  | 是 | 空 |  | GitHUB用户名修改日期 |
| disable | Boolean |  | 否 |  |  | 是否禁用,值为是表示禁用,否表示正常 |

### **4.ADMIN表(管理员表)**

| **字段** | **类型** | **主键，外键** | **可以为空** | **默认值** | **约束** | **说明** |
| --- | --- | --- | --- | --- | --- | --- |
| user\_id | NUMBER(8,0) | 主键 | 否 |  |  | 管理员id |

### **5.LESSON表(课程表)**

| **字段** | **类型** | **主键，外键** | **可以为空** | **默认值** | **约束** | **说明** |
| --- | --- | --- | --- | --- | --- | --- |
| lesson\_id | NUMBER(5,0) | 主键 | 否 |  |  | 课程ID |
| lesson\_name | VARCHAR2(20 BYTE) |  | 否 |  |  | 课程名 |
| lesson\_type | VARCHAR2(10 BYTE) |  | 否 |  |  | 课程类型 |
| lesson\_describe | VARCHAR2(100 BYTE) |  | 是 | 空 |  | 课程简介 |
| lesson\_releaseDate | DATE |  | 是 | 空 |  | 课程发布时间，为空表示尚未发布 |
| lesson\_chooseOrNot | Boolean |  | 否 | False |  | 课程是否被选择，true表示被选，false表示未被选 |
| lesson\_finishTime | DATE |  | 是 | 空 |  | 为空表示未结束，每个课程自发布开始6个月之内必须结课 |

### **6.TEST表(实验表)**

| **字段** | **类型** | **主键，外键** | **可以为空** | **默认值** | **约束** | **说明** |
| --- | --- | --- | --- | --- | --- | --- |
| test\_id | NUMBER(5,0) | 主键 | 否 |  |  | 实验ID |
| lesson\_id | NUMBER(5,0) | 外键 | 否 |  |  | 实验所属的课程ID |
| test\_name | VARCHAR2(20 BYTE) |  | 否 |  |  | 实验名 |
| test\_type | VARCHAR2(10 BYTE) |  | 否 |  |  | 实验类型 |
| test\_describe | VARCHAR2(100 BYTE) |  | 是 | 空 |  | 实验简介 |
| test\_releaseDate | DATE |  | 是 | 空 |  | 实验发布时间，为空表示尚未发布 |
| test\_totalResult | NUMBER(5,0) |  | 是 | 空 |  | 实验设置的总分数 |

### **7.CHOOSELESSON表(选课表)**

| **字段** | **类型** | **主键，外键** | **可以为空** | **默认值** | **约束** | **说明** |
| --- | --- | --- | --- | --- | --- | --- |
| choose\_id | NUMBER(8,0) | 主键 | 否 |  | 自增 | 选课操作ID |
| lesson\_id | NUMBER(5,0) | 联合主键，外键 | 否 |  |  | 课程ID |
| teacher\_id | NUMBER(8,0) | 外键 | 是 | 空 |  |  |
| student\_id | NUMBER(8,0) | 外键 | 是 | 空 | teacher\_id不为空 | 为空表示尚未有学生选此课,如果teacher\_id为空，则此字段也为空 |

### **8.SCOREITEMS表(评分项表)**

| **字段** | **类型** | **主键，外键** | **可以为空** | **默认值** | **约束** | **说明** |
| --- | --- | --- | --- | --- | --- | --- |
| scoreItems\_id | NUMBER(5,0) | 主键 | 否 |  | 自增 | 评分项ID |
| scoreItems\_name | VARCHAR2(10 BYTE) |  | 否 |  |  | 评分项名 |
| scoreItems\_describe | VARCHAR2(100 BYTE) |  | 是 | 空 |  | 评分项简介 |
| scoreItems\_totalResult | NUMBER(5,0) |  | 否 | 20 |  | 评分项总分值 |
| scoreItems\_releaseDate | DATE |  | 否 |  |  | 评分项发布时间 |
| test\_id | NUMBER(5,0) | 外键 | 否 |  |  | 所属实验ID |

### **9.SCORE表(单项得分表)**

| **字段** | **类型** | **主键，外键** | **可以为空** | **默认值** | **约束** | **说明** |
| --- | --- | --- | --- | --- | --- | --- |
| score\_result | DOUBLE | 主键 | 否 |  |  | 单评分项实际得分 |
| teacher\_id | NUMBER(8,0) | 外键 | 否 |  |  | 评阅的教师职工号 |
| student\_id | NUMBER(8,0) | 外键 | 否 |  |  | 学生ID |
| scoreItems\_id | NUMBER(5,0) | 外键 | 否 |  |  | 评分项编号 |
| score\_describe | VARCHAR2(100 BYTE) |  | 是 | 空 |  | 评语 |
| score\_releaseDate | DATE |  | 否 |  |  | 评改的时间 |

### **10.TESTAVERAGE表(实验平均分表)**

| **字段** | **类型** | **主键，外键** | **可以为空** | **默认值** | **约束** | **说明** |
| --- | --- | --- | --- | --- | --- | --- |
| average\_test | DOUBLE | 主键 | 否 |  |  | 实验平均成绩 |
| student\_id | NUMBER(8,0) | 联合主键，外键 | 否 |  |  | 学生学号 |
| test\_id | NUMBER(5,0) | 联合主键，外键 | 否 |  |  | 实验编号 |

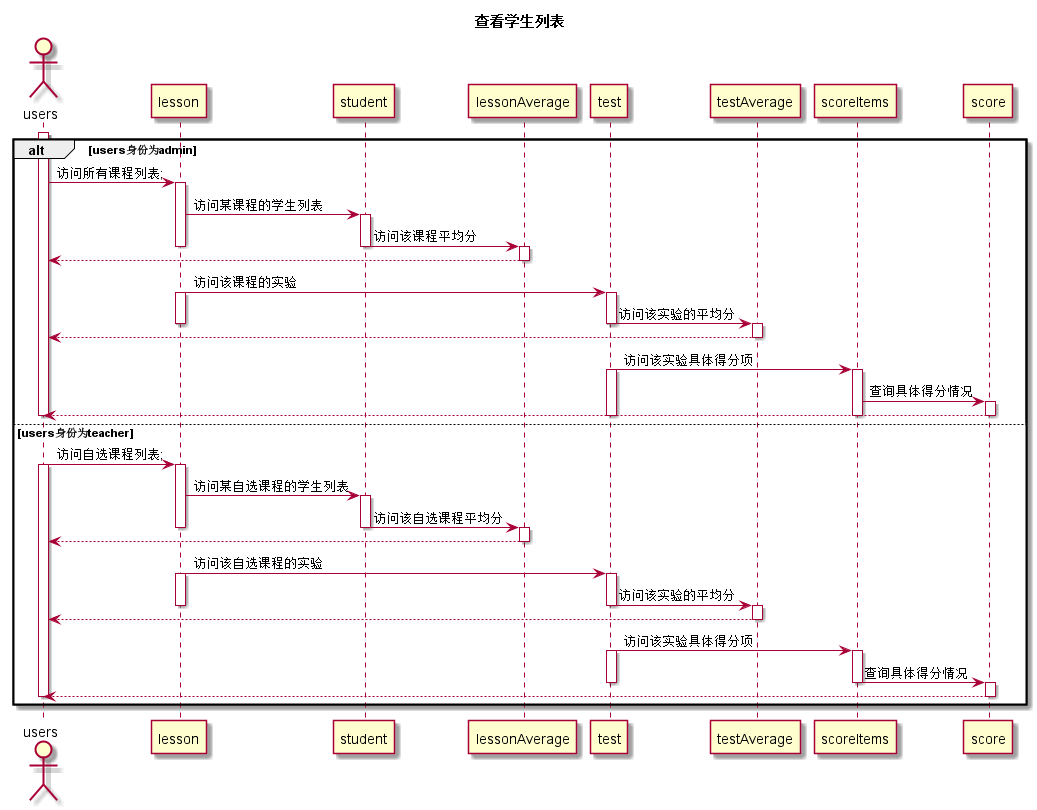
### **11.LESSONAVERAGE表(课程平均分表)**

| **字段** | **类型** | **主键，外键** | **可以为空** | **默认值** | **约束** | **说明** |
| --- | --- | --- | --- | --- | --- | --- |
| average\_lesson | DOUBLE | 主键 | 否 |  |  | 课程成绩平均值 |
| student\_id | NUMBER(8,0) | 联合主键，外键 | 否 |  |  | 学生学号 |
| lesson\_id | NUMBER(5,0) | 联合主键，外键 | 否 |  |  | 实验编号 |
| web\_sum | VARCHAR2(400 BYTE) |  | 是 | 空 |  | 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 |

### **12.YEARANDTERM表(学年学期表)**

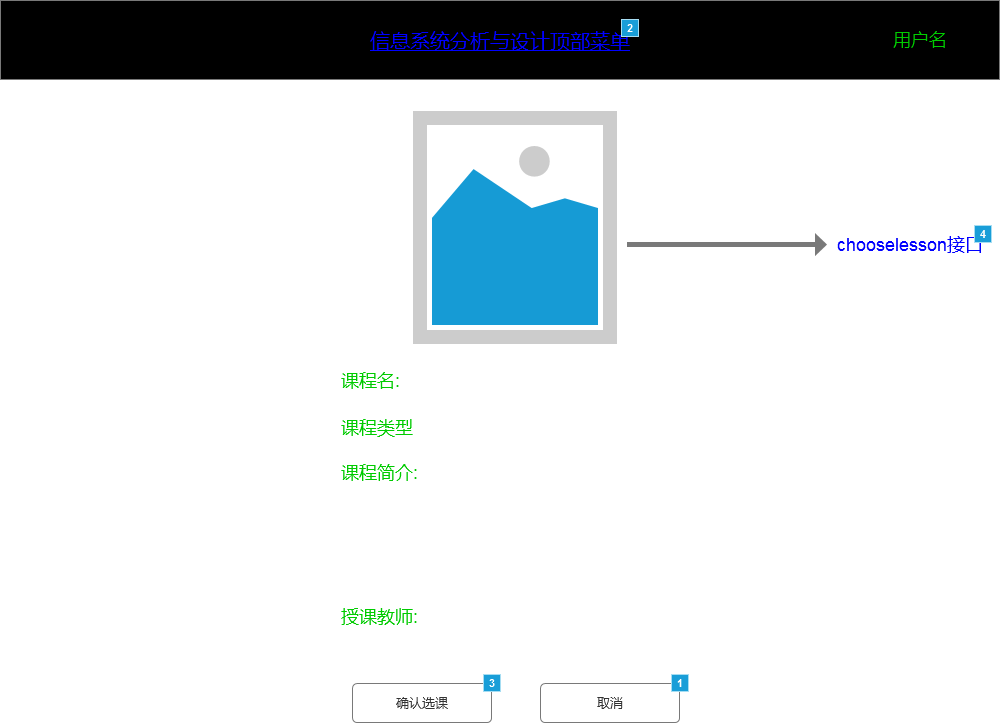
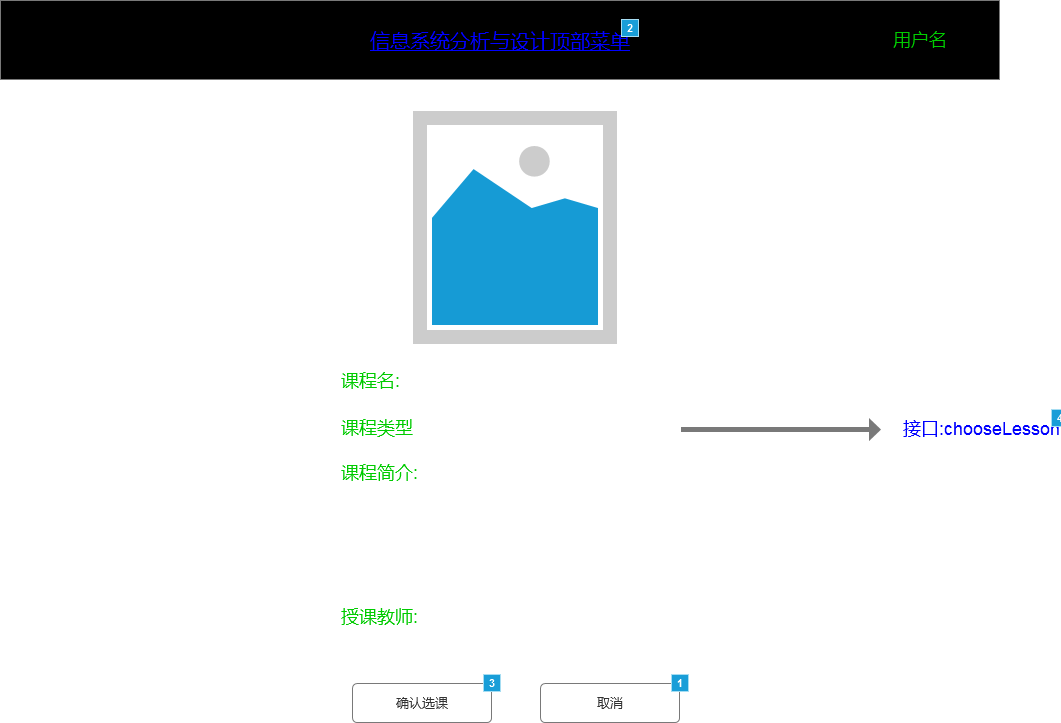
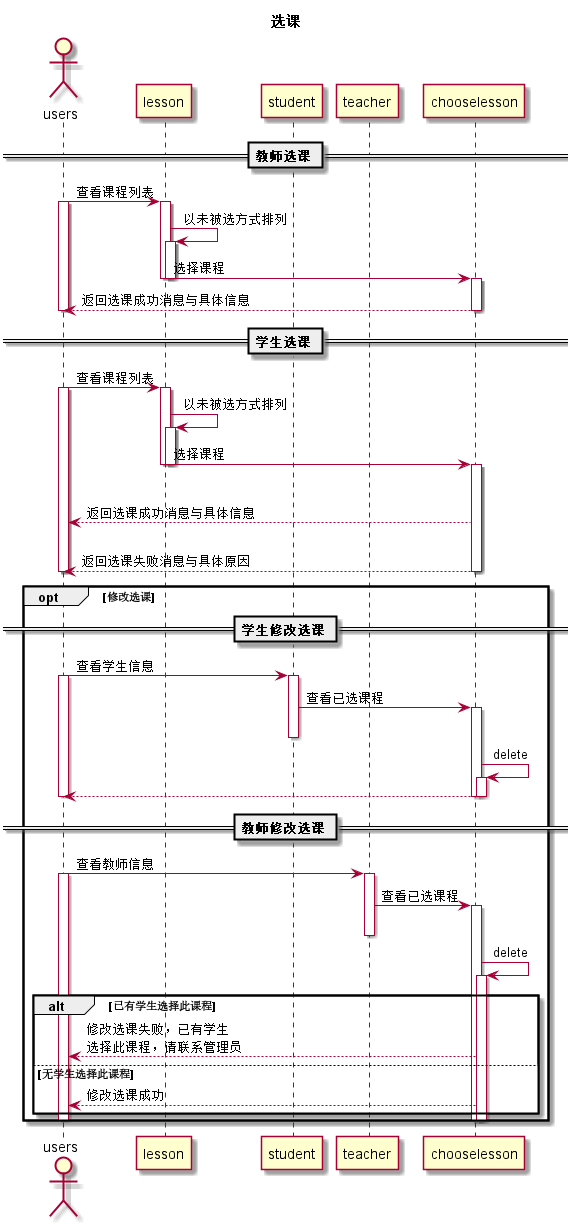
| **字段** | **类型** | **主键，外键** | **可以为空** | **默认值** | **约束** | **说明** |
| --- | --- | --- | --- | --- | --- | --- |
| year\_and\_term | VARCHAR2(20 BYTE) | 主键 | 否 |  |  | 学年和学期，中间用-分开，第一位代表学年，第二位代表学期，学期分为01(上学期),02(下学期),例如:"2019-01" |
| lesson\_id | NUMBER(5,0) | 外键 | 是 | 空 |  | 为空代表该学期尚未增加课程 |

1. 用例及界面设计
2. 学生列表用例及界面

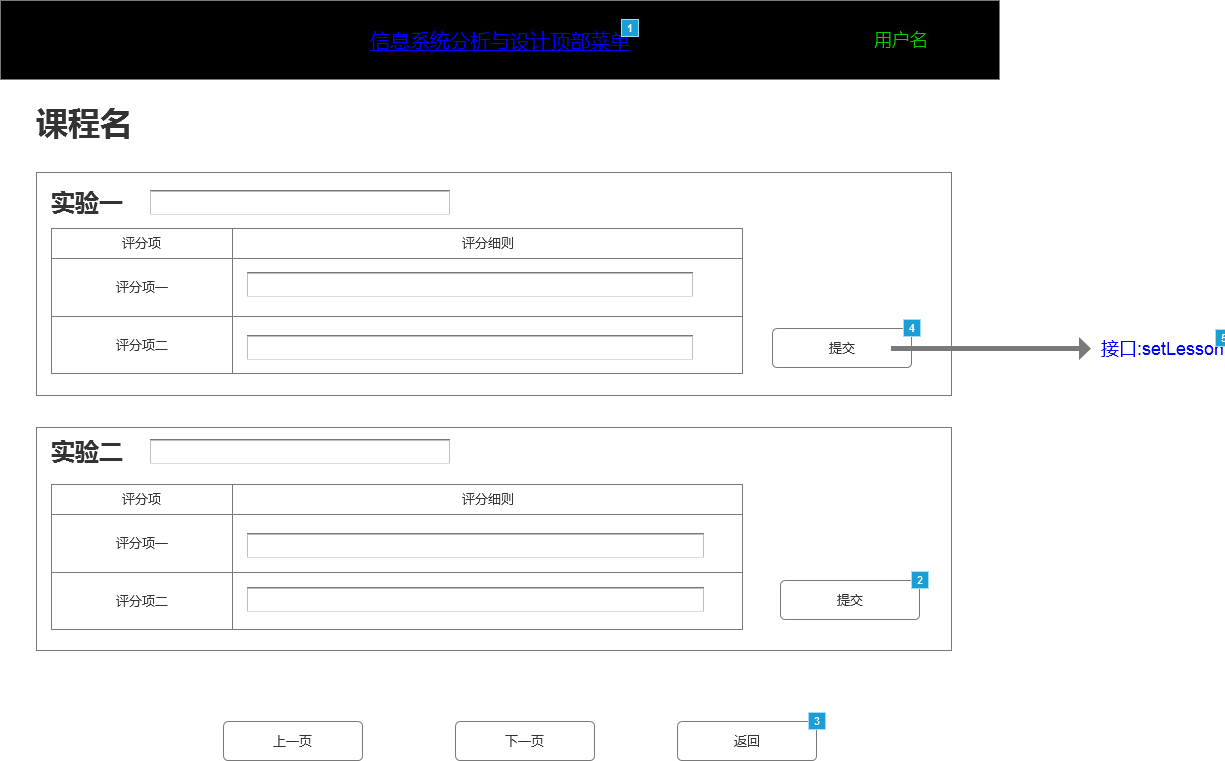
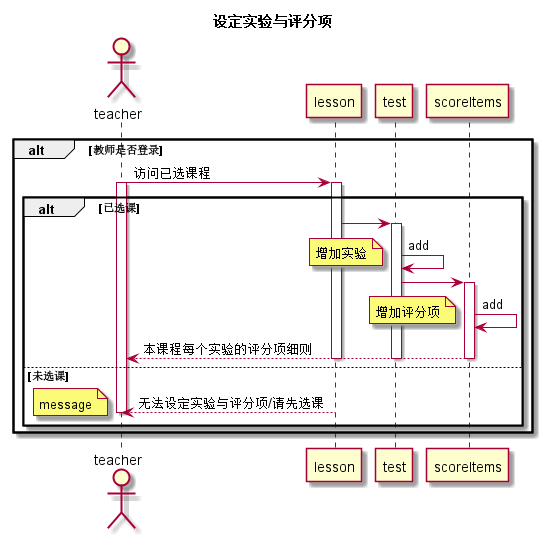




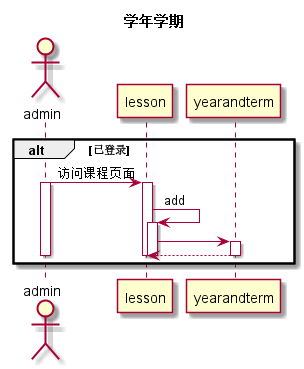
1. 选择课程用例及界面

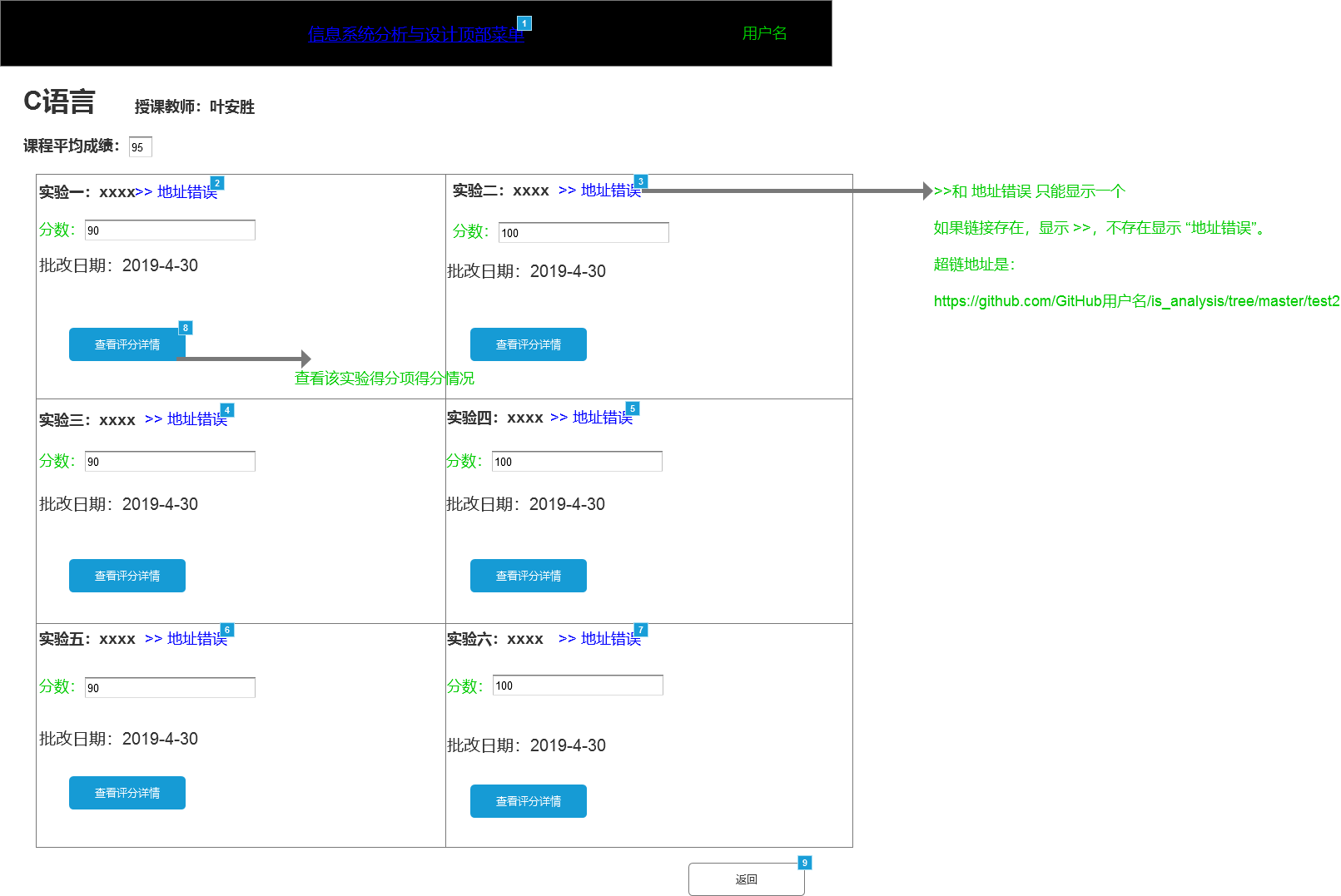
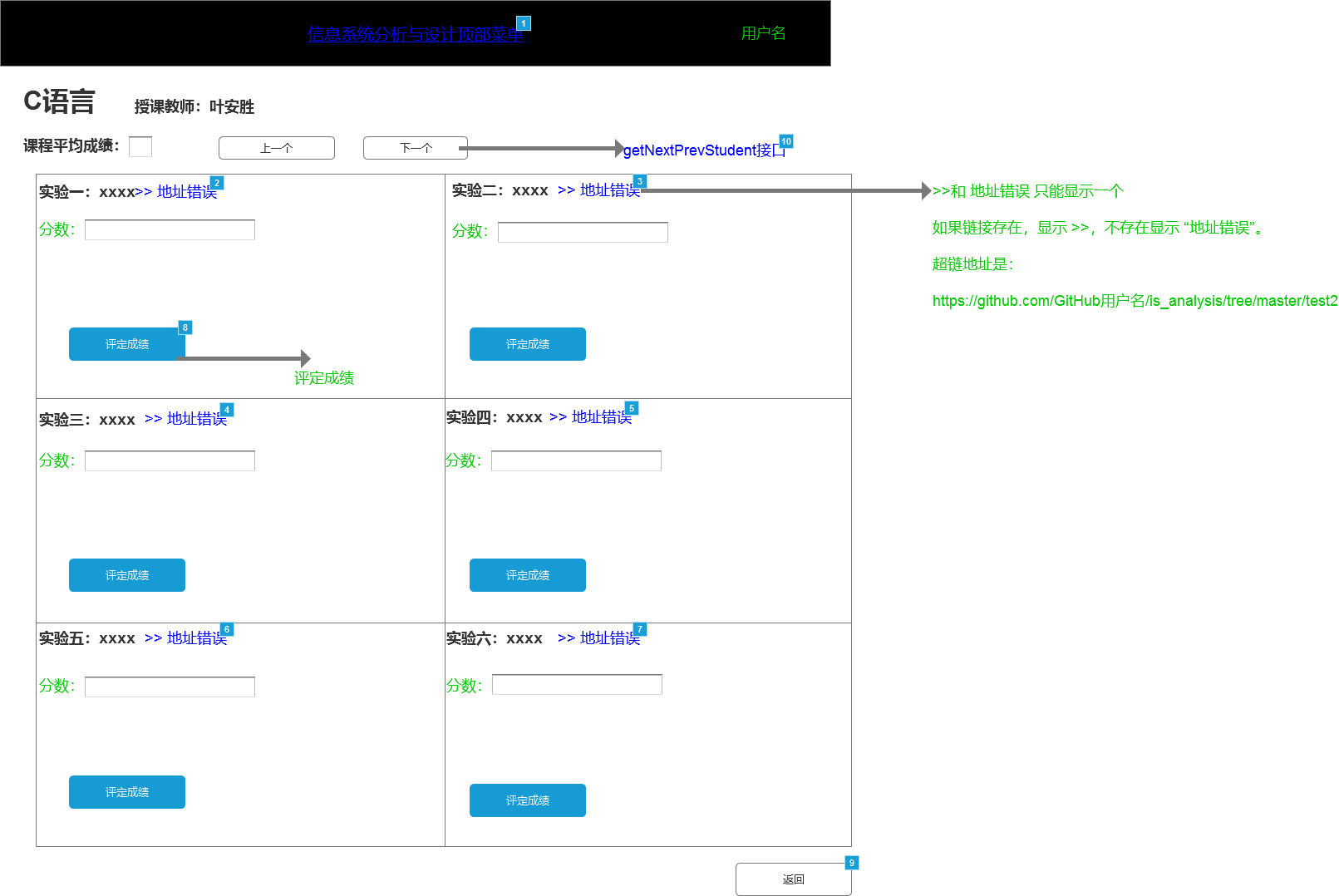
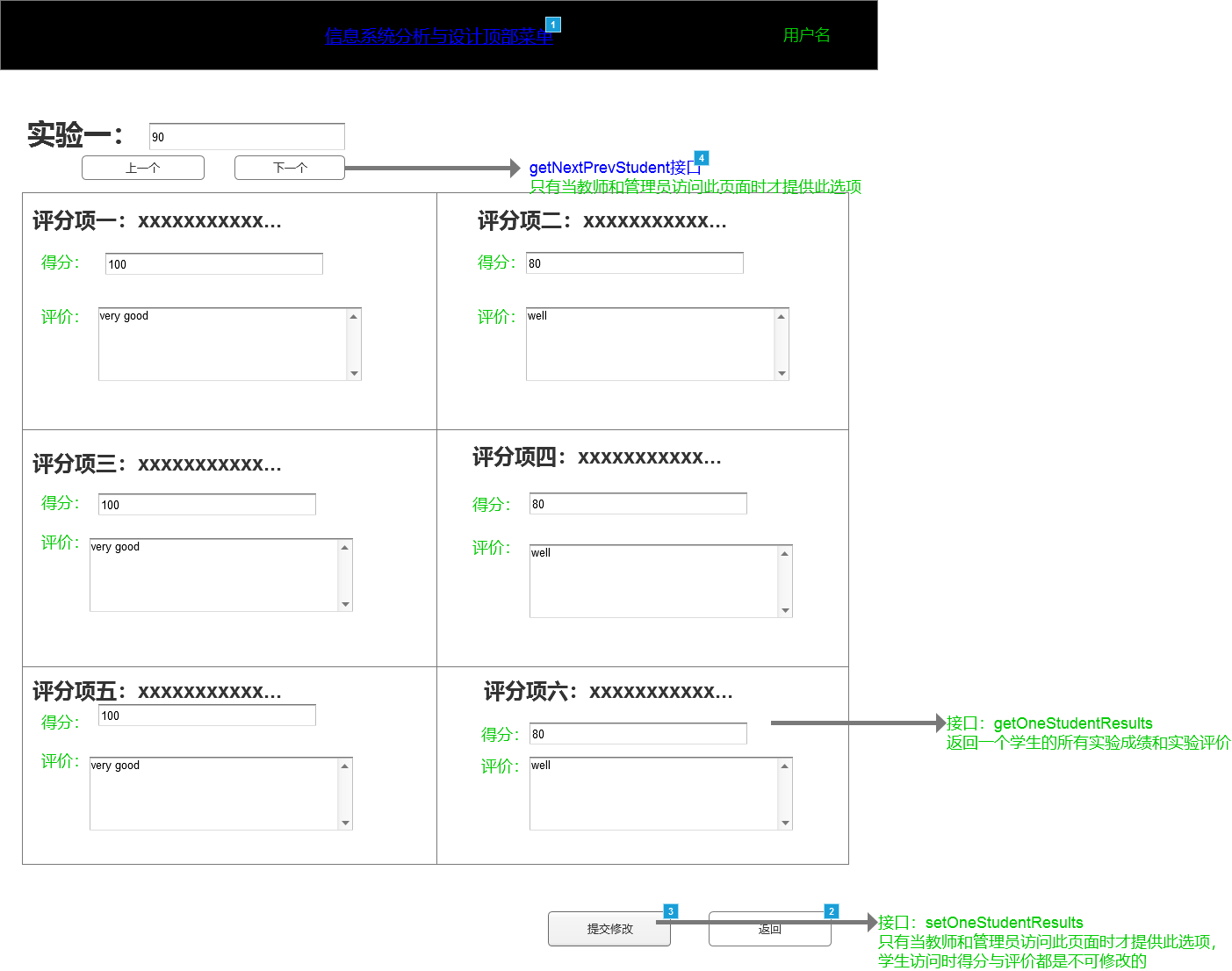
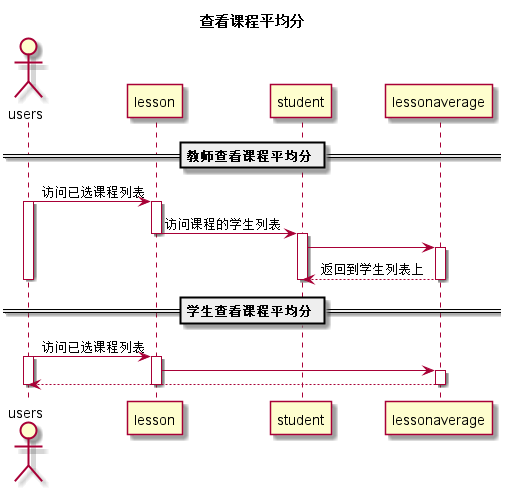
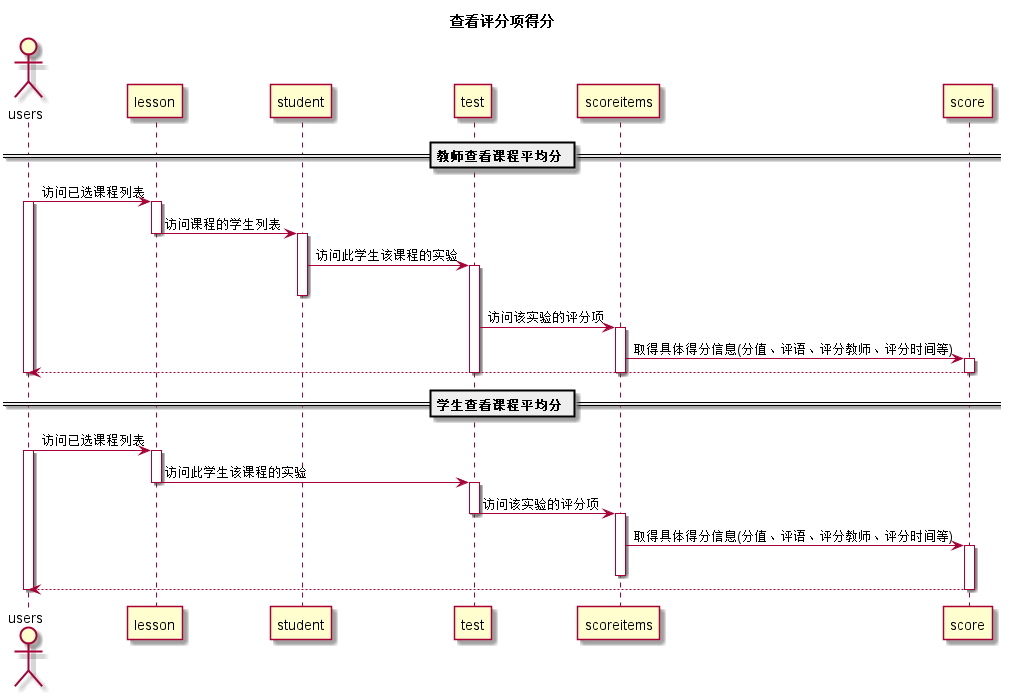
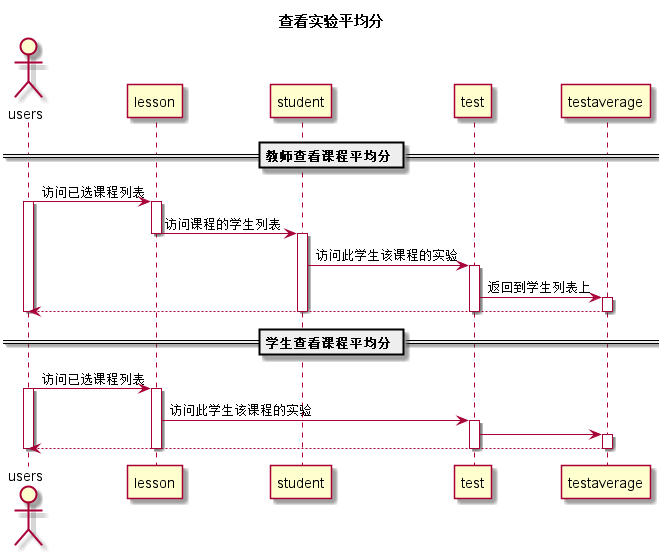


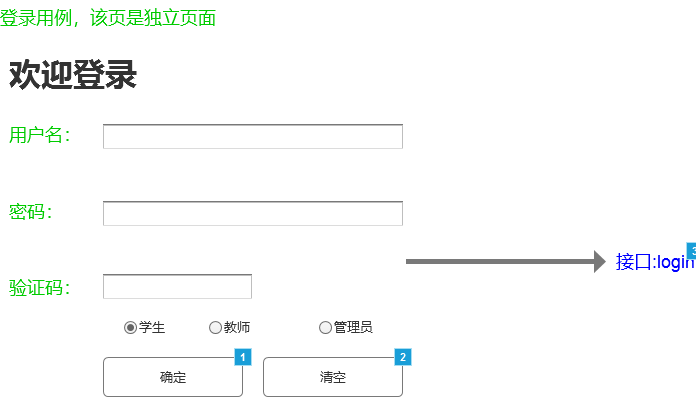
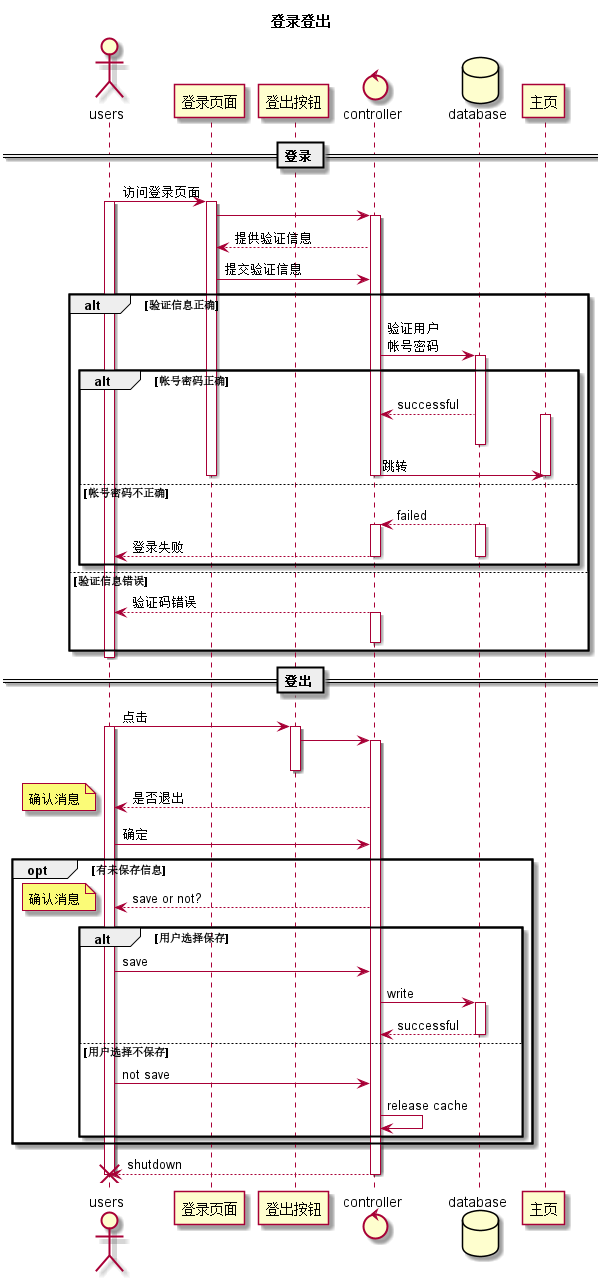
1. 设定实验及评分项用例及界面

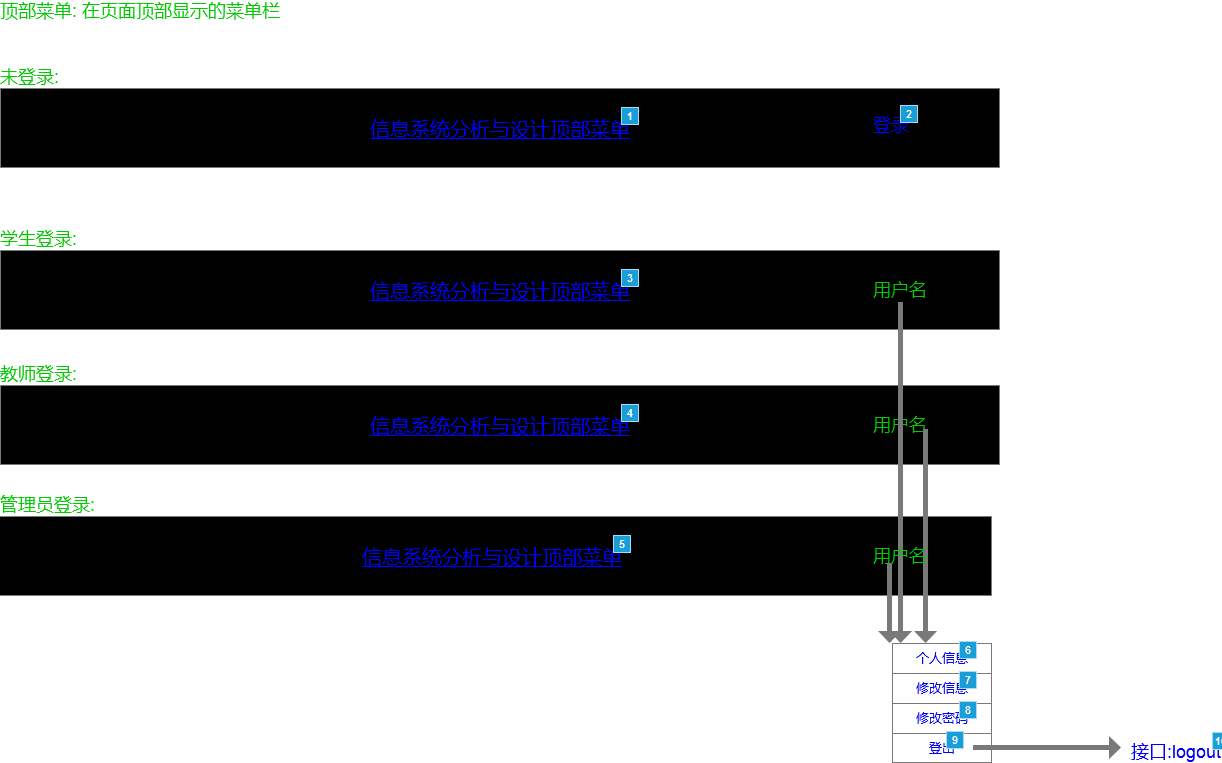


1. 学年学期用例

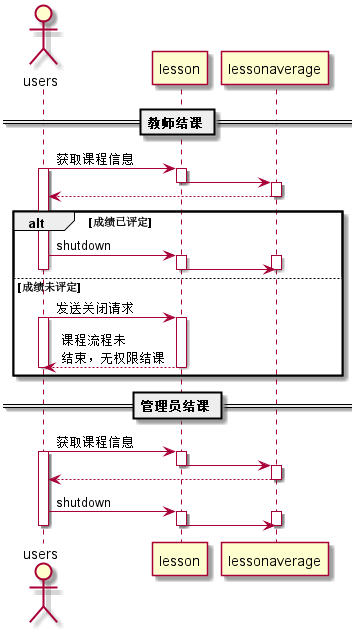


1. 查看成绩(实验平均分、课程平均分、评分项得分)用例及界面
2. 登录登出用例及界面

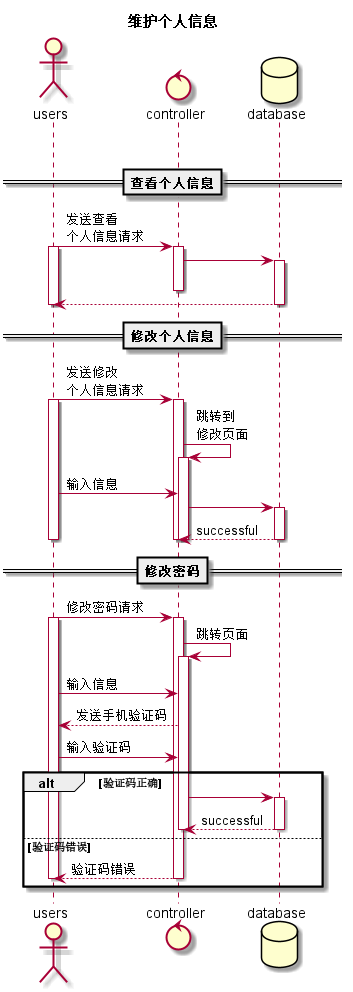




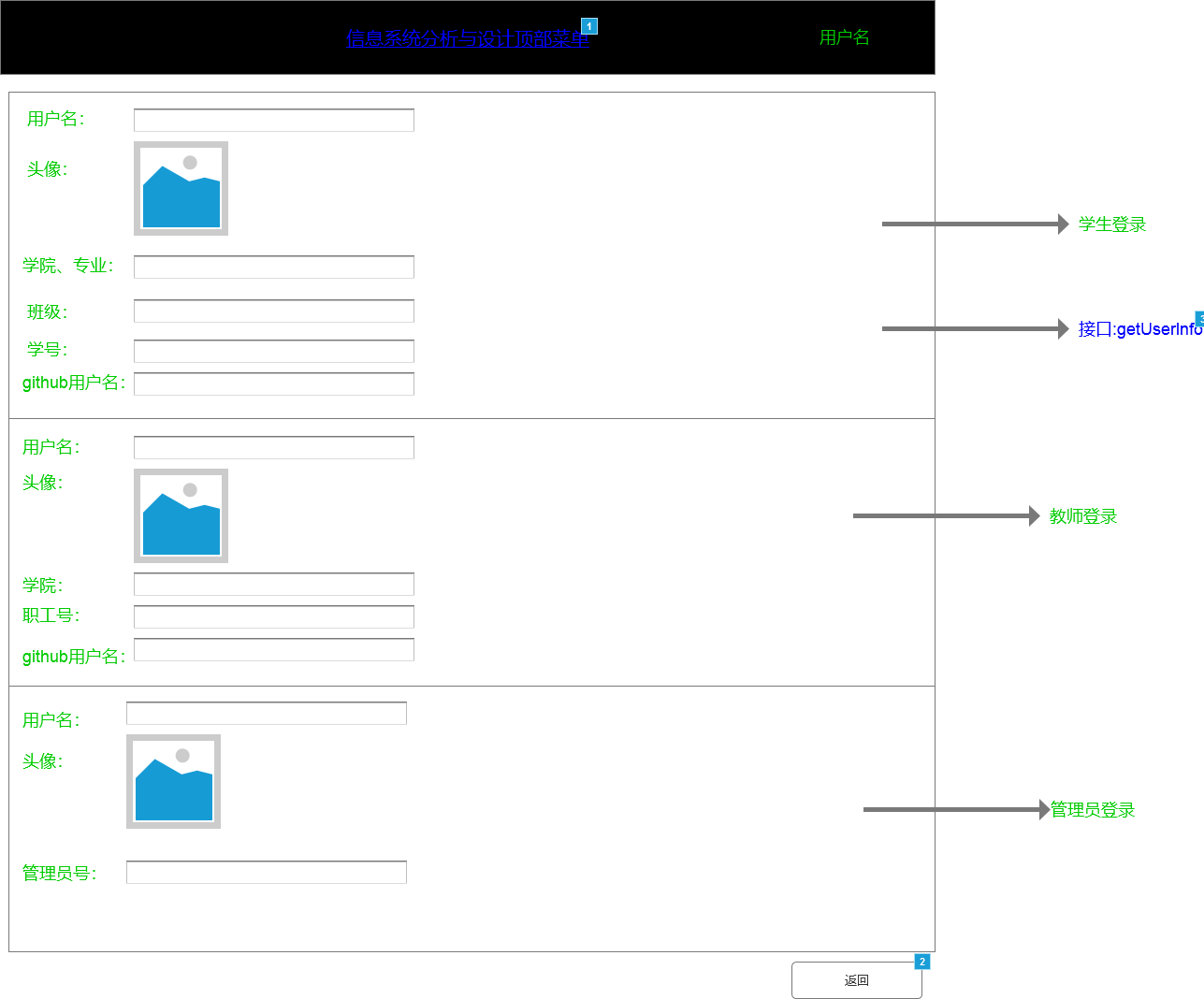
1. 结束课程用例

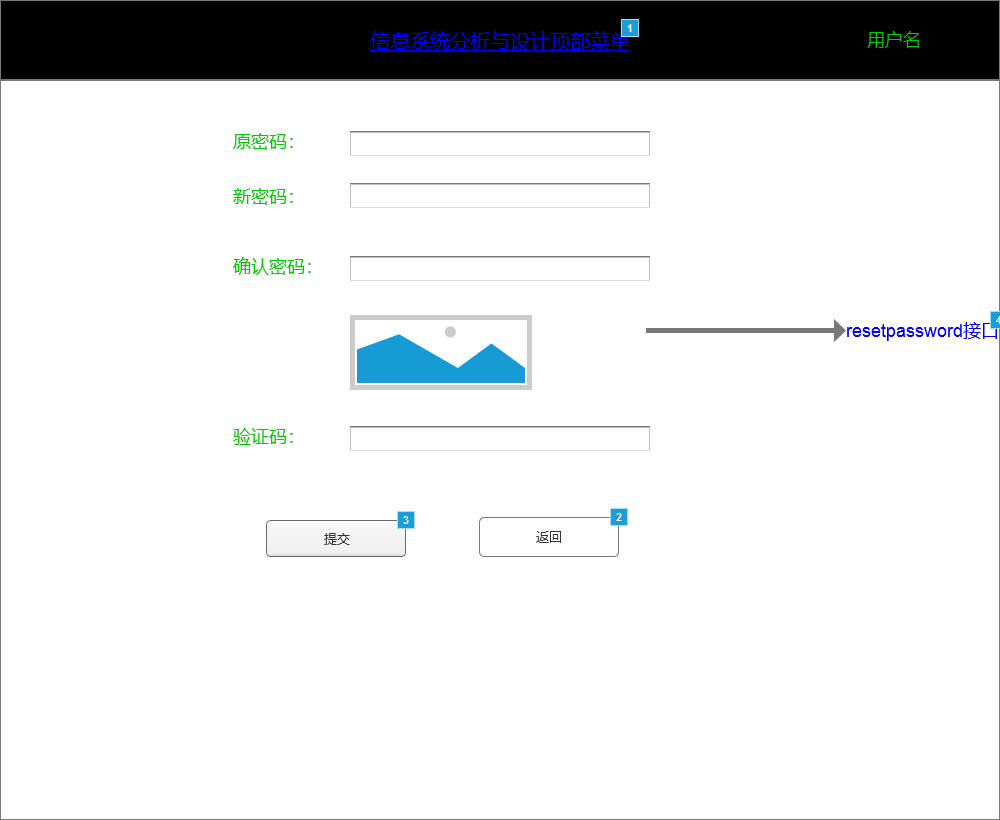


1. 维护个人信息用例及界面

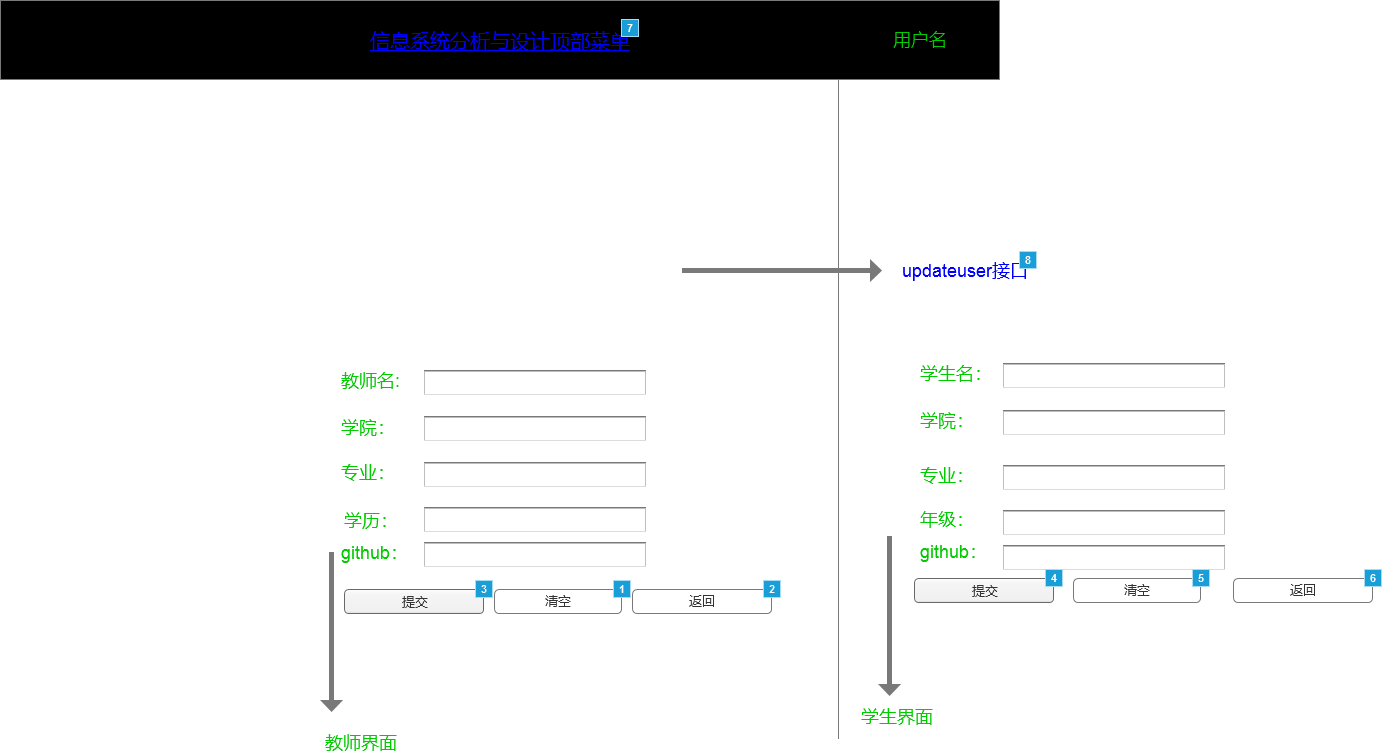
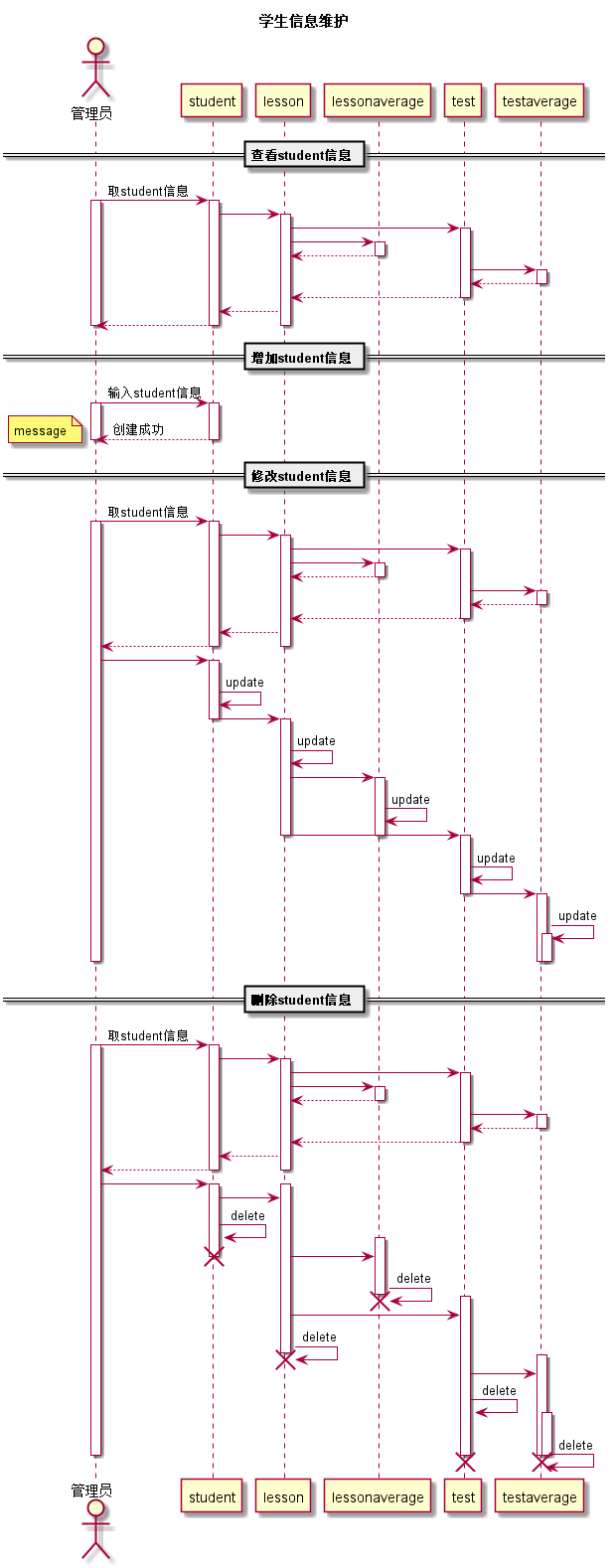


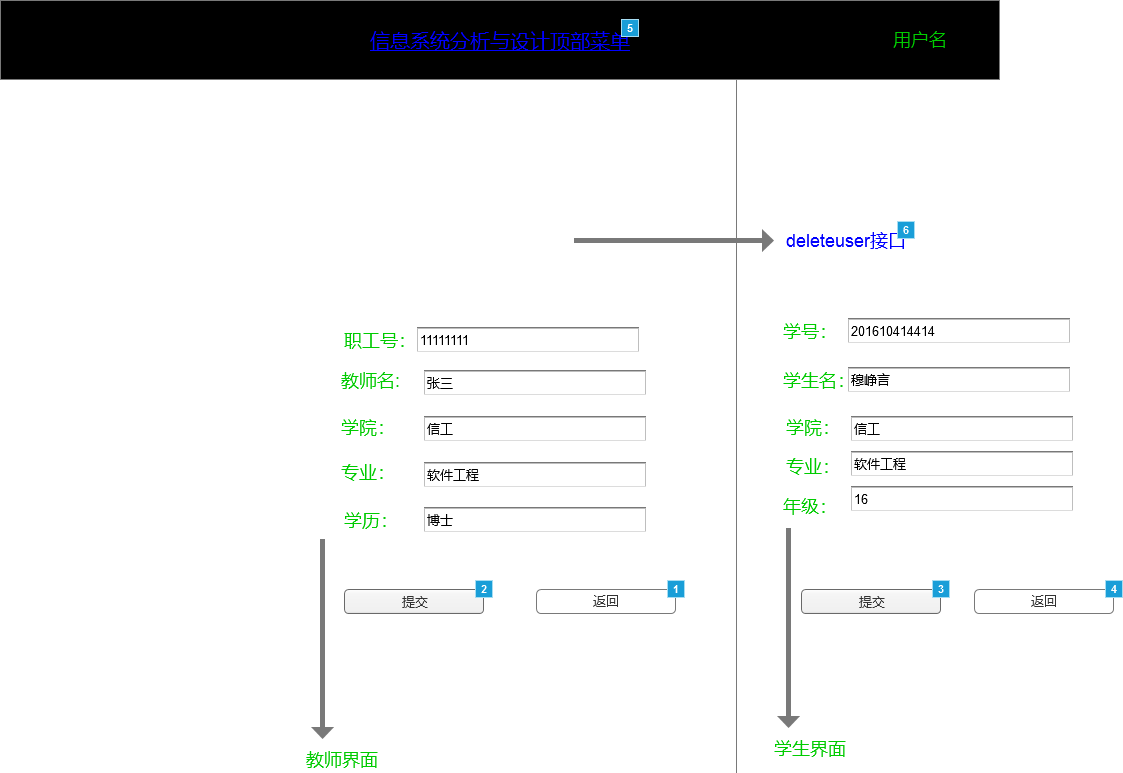




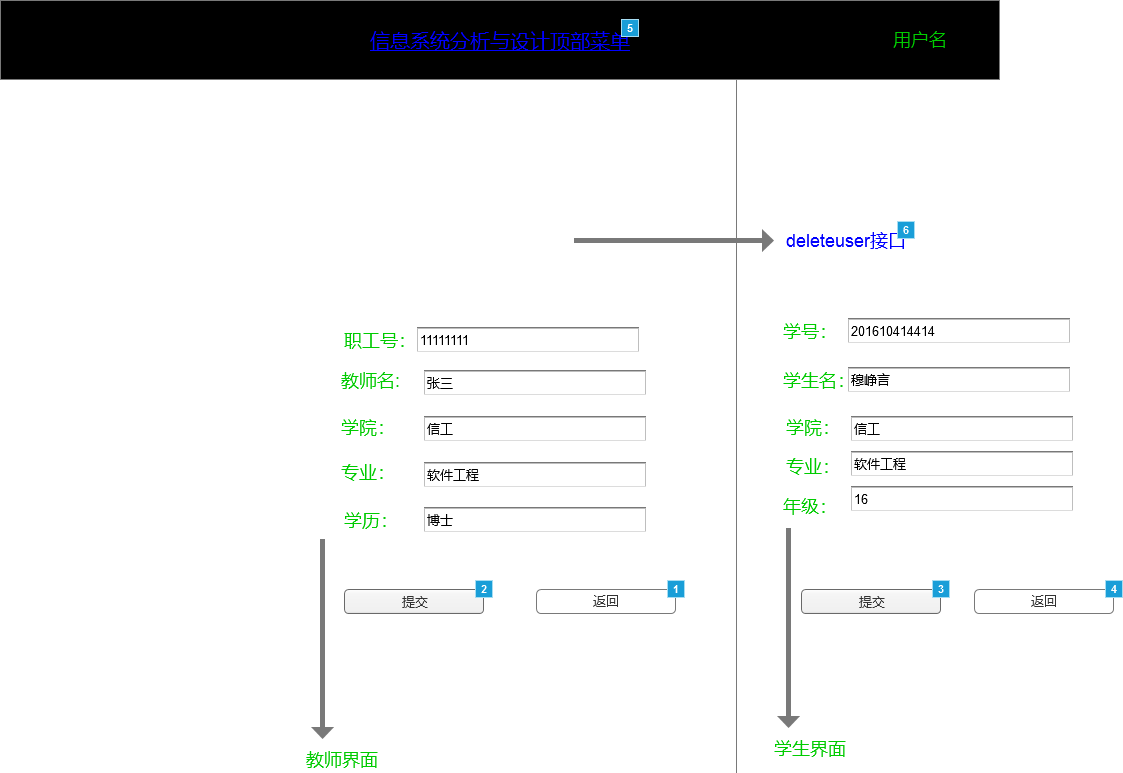
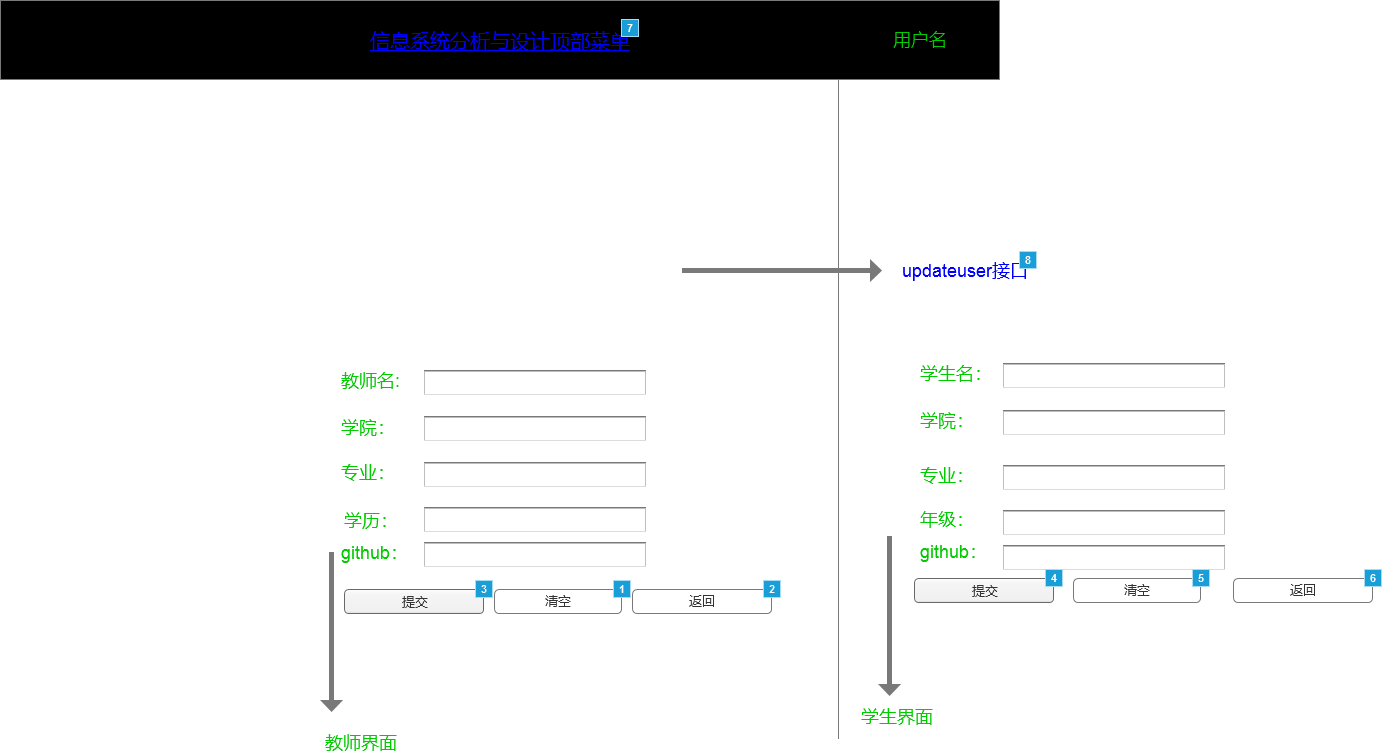
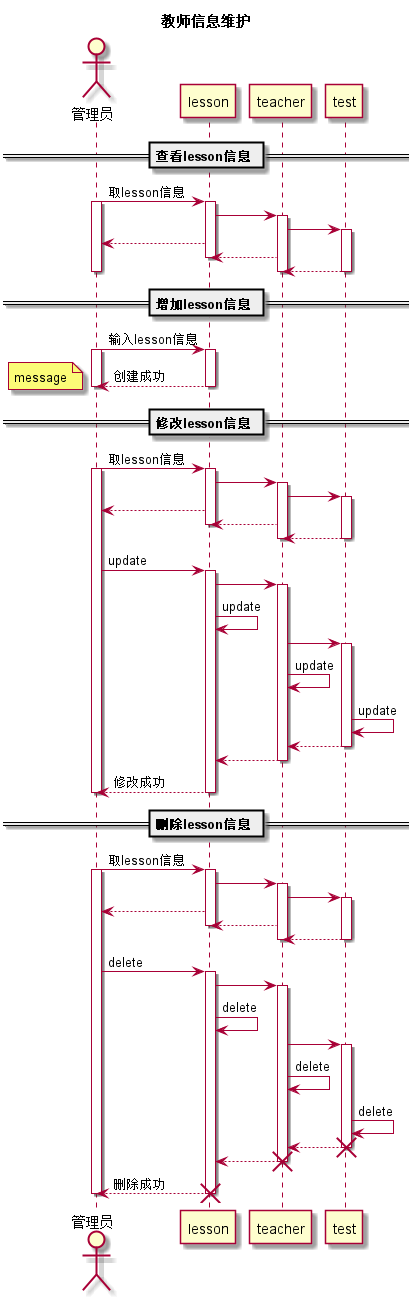


1. 维护学生信息用例及界面

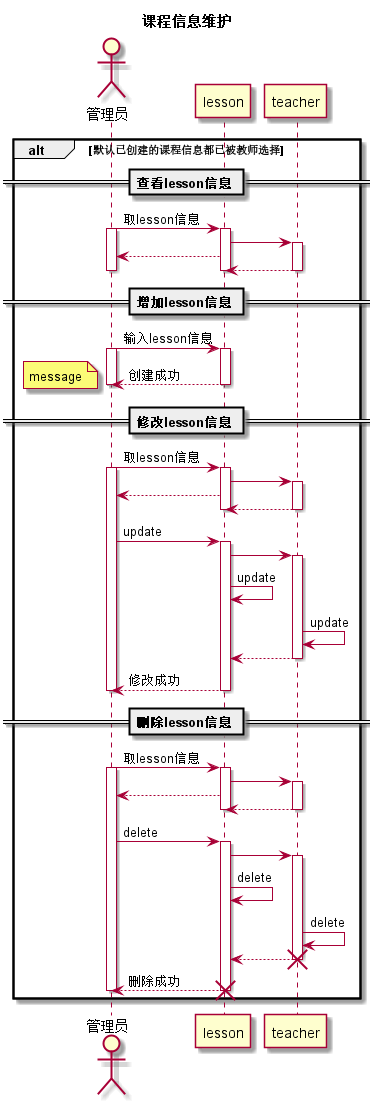




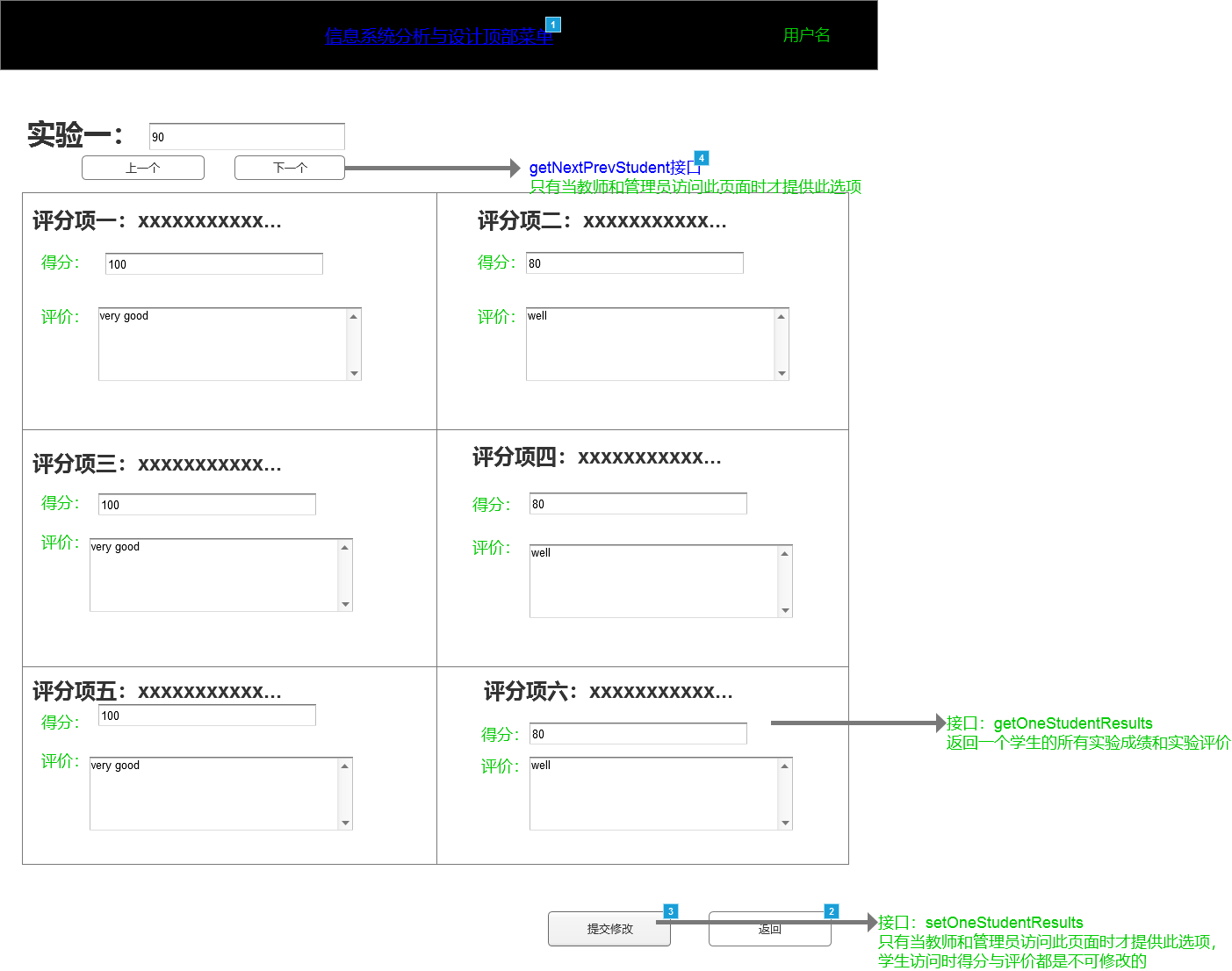
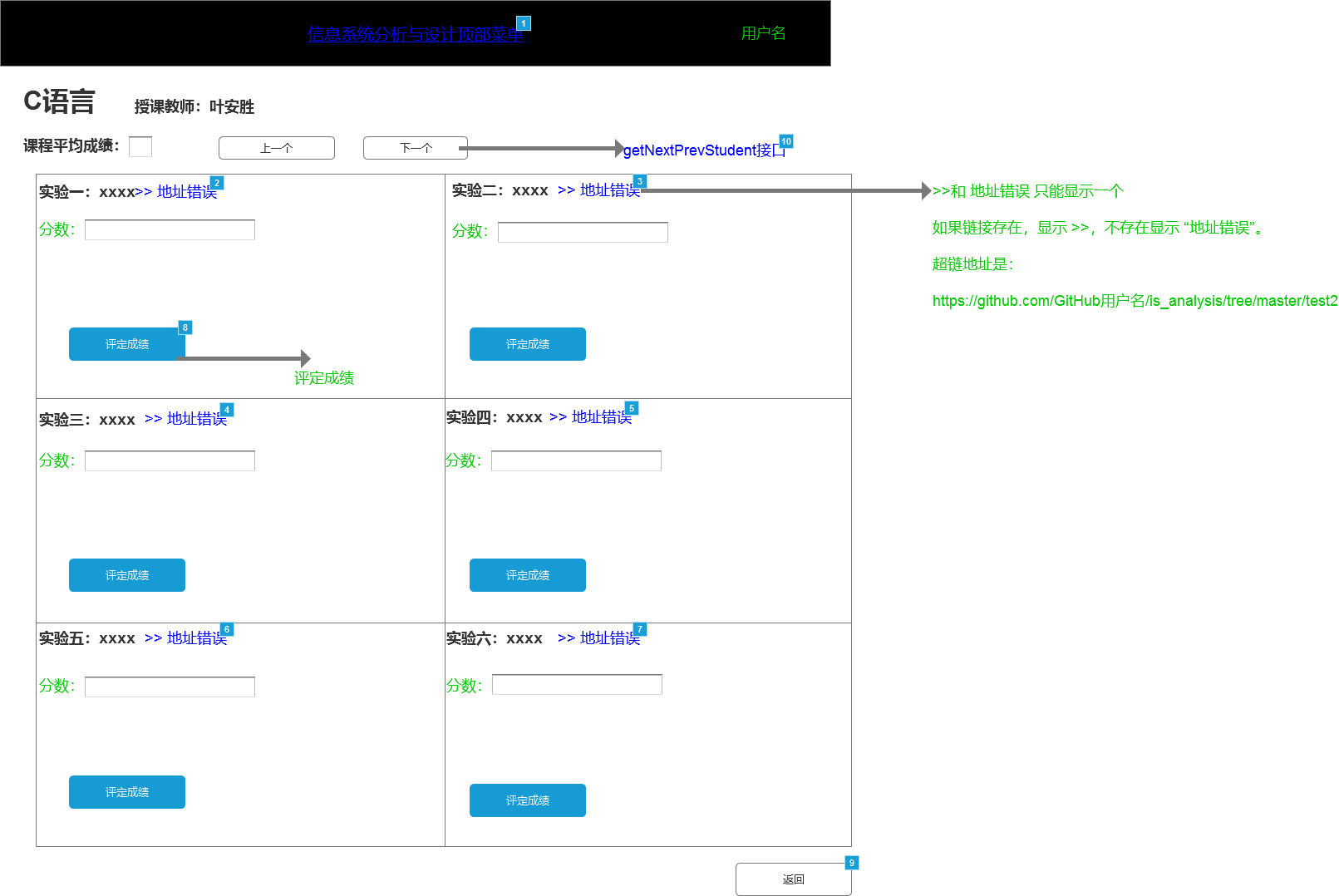
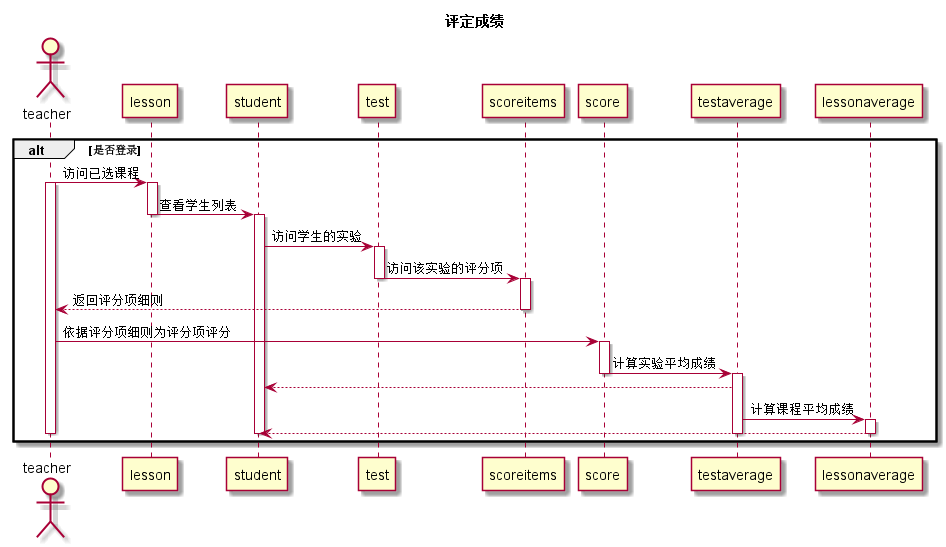
1. 维护教师信息用例及界面



1. 维护课程信息用例及界面



1. 评定成绩



1. 接口设计

1.接口：chooseLesson

用例:[选课](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E9%80%89%E8%AF%BE.md" \o "选课)

* 功能： 选择课程
* 权限： 学生/教师登录后。
* API请求地址： 接口基本地址/v1/api/chooseLesson
* 请求方式 ： POST
* 请求实例：

{

"user\_id":"21048329823",

"type":"学生"

"lesson\_id":"1231654"

}

* 请求参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| user\_id | 学生学号、老师的工号或者管理员的管理员号，由type决定 |
| lesson\_id | 课程id |
| type | 用户类型，学生、老师或者管理员 |

* 返回实例：

{

"status": true,

"info": null,

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |

2.接口：deleteLesson

用例:[维护课程信息](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E7%BB%B4%E6%8A%A4%E8%AF%BE%E7%A8%8B%E4%BF%A1%E6%81%AF.md" \o "维护课程信息)

* 功能： 删除课程信息
* 权限： 管理员登录。
* API请求地址： 接口基本地址/v1/api/deleteLesson
* 请求方式 ： POST
* 请求实例：

{

lesson\_id":"11654631",

}

* 请求参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| lesson\_id | 课程id |

* 返回实例：

{

"status": true,

"info": null,

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |

3.接口：deleteUser

用例:[维护学生信息](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E7%BB%B4%E6%8A%A4%E5%AD%A6%E7%94%9F%E4%BF%A1%E6%81%AF.md" \o "维护学生信息)[维护教师信息](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E7%BB%B4%E6%8A%A4%E6%95%99%E5%B8%88%E4%BF%A1%E6%81%AF.md" \o "维护教师信息)

* 功能： 删除用户信息
* 权限： 管理员登录。
* API请求地址： 接口基本地址/v1/api/deleteUser
* 请求方式 ： POST
* 请求实例：

{

"user\_id":"201610414414",

}

* 请求参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| user\_id | 用户id，对应学生学号，教师职工号 |

* 返回实例：

{

"status": true,

"info": null,

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |

4.接口：getLessons

用例:[维护课程信息](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E7%BB%B4%E6%8A%A4%E8%AF%BE%E7%A8%8B%E4%BF%A1%E6%81%AF.md" \o "维护课程信息)

* 功能： 修改（设置）用户的密码
* 权限： 学生/老师/管理员：修改自己的信息，必须先登录。
* API请求地址： 接口基本地址/v1/api/getLessons
* 请求方式 ： GET
* 请求实例：

{

"year\_and\_term":2019-01,

"lesson\_chooseOrNot":false,

}

* 请求参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| year\_and\_term | 学年学期 |
| lesson\_name | 课程名 |
| lesson\_chooseOrNot | 课程是否被选 |

* 返回实例：

{

"status": true,

"info": null,

"data": [

{"lesson\_type": "公共基础课",

"lesson\_describe": "xxxxxxxxx",

"lesson\_name": "大学英语基础课程",

"lesson\_releaseDate": "2018-04-30",

"user\_name": "本尼迪克特",

},

{

...其他课程

}

]

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |
| total | 返回课程数目 |
| data | 所有课程的数组 |
| lesson\_type | 课程的类型 |
| lesson\_describe | 课程的简介 |
| lesson\_releaseDate | 课程的发布时间 |
| user\_name | 授课教师的姓名 |

1. 接口：getNextPreStudent

用例:[评定成绩](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E8%AF%84%E5%AE%9A%E6%88%90%E7%BB%A9.md" \o "评定成绩)

* 功能： 返回一个学生的上一个或者下一个学生的学号
* 权限： 教师/管理员登录后。
* API请求地址： 接口基本地址/v1/api/getNextPrevStudent/<is\_next>/<user\_id>
* 请求方式 ： GET
* 请求参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| user\_id | 学生学号、老师的工号或者管理员的管理员号，由type决定 |
| is\_next | 是下一个还是上一个，如果为1表示取下一个学生，如果为0表示上一个学生 |

* 返回实例：

{

"status": true,

"info": null,

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |
| user\_id | 学号 |

1. 接口:getOneStudentResult

用例:[评定成绩](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E8%AF%84%E5%AE%9A%E6%88%90%E7%BB%A9.md" \o "评定成绩)[查看评分项得分](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E6%9F%A5%E7%9C%8B%E8%AF%84%E5%88%86%E9%A1%B9%E5%BE%97%E5%88%86.md" \o "查看评分项得分)

* 功能： 返回一个学生的所有评分项得分和评价
* 权限： 老师/管理员登录。
* API请求地址： 接口基本地址/v1/api/getOneStudentResults
* 请求方式 ： GET
* 请求参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| user\_id | 用户id，对应学生学号 |

* 返回实例：

{

"status": true,

"info": null,

"user\_id": "201610414414",

"github\_username": "mzy1997",

"grade": "16级",

"major":"软件工程"

"class":"4班",

"user\_name": "穆峥言",

"total": 6,

"average\_test":90,

"data": [

{

"scoreItems\_id":1,

"scoreItems\_name":"xxx"

"web\_exists": true,

"score\_result": 91,

"score\_describe":"本实验做得好",

"score\_releaseDate": "2018-04-02"

},

{

...其他评分项

}

]

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |
| user\_id | 用户id，对应学生学号 |
| github\_username | github用户名 |
| grade | 年级 |
| major | 专业 |
| class | 班级 |
| user\_name | 用户姓名 |
| total | 总评分项数目 |
| average\_test | 实验平均分 |
| scoreItems\_id | 评分项id |
| scoreItems\_name | 评分项名 |
| web\_exists | 网址是否存在 |
| score\_result | 评分项得分 |
| score\_describe | 评价 |
| score\_releaseDate | 评改时间 |

1. 接口:getUserInfo

用例:[维护个人信息](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E7%BB%B4%E6%8A%A4%E4%B8%AA%E4%BA%BA%E4%BF%A1%E6%81%AF.md" \o "维护个人信息)

* 功能： 删除用户信息
* 权限： 管理员登录。
* API请求地址： 接口基本地址/v1/api/getUserInfo/
* 请求方式 ： GET
* 请求参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| user\_id | 用户id，对应学生学号，教师职工号 |

* 返回实例：

{

"status": true,

"info": null,

"user\_id":"2101201281",

"user\_name":"张三",

"student\_class":"软件工程1班"

"github\_username":"ABCD",

"type":"学生"

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |
| user\_id | 用户id，对应学生学号，教师职工号，管理员管理员号 |
| user\_name | 用户姓名 |
| github\_username | github用户名 |
| student\_class | 学生所在班级 |
| type | 用户类型 |

1. 接口:insertLesson

用例:[维护课程信息](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E7%BB%B4%E6%8A%A4%E8%AF%BE%E7%A8%8B%E4%BF%A1%E6%81%AF.md" \o "维护课程信息)

* 功能： 增加课程信息
* 权限： 管理员登录。
* API请求地址： 接口基本地址/v1/api/insertLesson
* 请求方式 ： POST
* 请求实例：

{

"lesson\_id":123456

"lesson\_name":"大学英语",

"lesson\_type":"公共基础课"

"lesson\_describe":"本书为远程教育大学英语基础课教材，由十课组成;每课包括课文，词汇，注释，语音或语法，练习，补充课文等。文章语言规范，语法讲解透彻，增强了教材的实用性。"

"user\_name":"本尼迪克特"

}

* 请求参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| lesson\_id | 课程id |
| lesson\_name | 课程名 |
| lesson\_type | 课程类型 |
| lesson\_describe | 课程简介 |
| user\_name | 授课教师名 |

* 返回实例：

{

"status": true,

"info": null,

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |

1. 接口:login

用例:[登录登出](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E7%99%BB%E5%BD%95%E7%99%BB%E5%87%BA.md" \o "登录登出)

* 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:

|  |  |
| --- | --- |
| 参数 | 说明 |
| id | 学生学号、老师的工号或者管理员的管理员号，由type决定 |
| password | 用户的密码，不是原文，是加密后的字符串 |
| type | 用户类型，学生、老师或者管理员 |

* 返回实例：

{

"status": true,

"info": null,

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |

1. 接口:logout

用例:[登录登出](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E7%99%BB%E5%BD%95%E7%99%BB%E5%87%BA.md" \o "登录登出)

* 功能： 已经登录的用户登出平台
* 权限： 学生/老师/管理员
* API请求地址： 接口基本地址/v1/api/login
* 请求方式 ： POST
* 请求实例：

{

"user\_id":324111

}

* 请求参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| user\_id | 已经登录用户的user\_id值，对应学生学号、教师职工号、管理员管理员号 |

* 返回实例：

{

"status": true,

"info": null,

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |

1. 接口:resetPassword

用例:[维护个人信息](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E7%BB%B4%E6%8A%A4%E4%B8%AA%E4%BA%BA%E4%BF%A1%E6%81%AF.md" \o "维护个人信息)

* 功能： 修改（设置）用户的密码
* 权限： 学生/老师/管理员：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:

|  |  |
| --- | --- |
| 参数 | 说明 |
| user\_id | 用户的ID号，对应学生学号、教师职工号、管理员管理员号 |
| password | 用户的密码，不是原文，是加密后的字符串 |

* 返回实例：

{

"status": true,

"info": null,

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |

1. 接口:resetUserInfo

用例:[维护个人信息](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E7%BB%B4%E6%8A%A4%E4%B8%AA%E4%BA%BA%E4%BF%A1%E6%81%AF.md" \o "维护个人信息)

* 功能： 修改（设置）用户的密码
* 权限： 学生/老师/管理员：修改自己的信息，必须先登录。
* API请求地址： 接口基本地址/v1/api/resetUserInfo
* 请求方式 ： POST
* 请求实例：

{

"user\_name":mzy1997,

"photo":xxxxx,

"github\_username":mzy1997

}

* 请求参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| user\_name | 用户名 |
| photo | 头像 |
| github\_username | github用户名 |

* 返回实例：

{

"status": true,

"info": null,

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |

1. 接口:setLesson

用例:[设定实验与评分项](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E8%AE%BE%E5%AE%9A%E5%AE%9E%E9%AA%8C%E4%B8%8E%E8%AF%84%E5%88%86%E9%A1%B9.md" \o "设定实验与评分项)

* 功能： 选择课程
* 权限： 学生/教师登录后。
* API请求地址： 接口基本地址/v1/api/setLesson
* 请求方式 ： POST
* 请求实例：

{

"test\_id":"1554601",

"test\_name":"顺序图实战"

"data": [

{"scoreItems\_id": "15546011",

"scoreItems\_describe": "xxxxxxxxx",

"scoreItems\_name": "画图得分",

},

{

...其他评分项

}

]

}

* 请求参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| test\_id | 实验id |
| test\_name | 实验名称 |
| data | 评分项数组 |
| scoreItems\_id | 评分项id |
| scoreItems\_describe | 评分项简介 |
| scoreItems\_name | 评分项名称 |

* 返回实例：

{

"status": true,

"info": null,

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |

1. 接口:setOneStudentResults

用例:[评定成绩](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E8%AF%84%E5%AE%9A%E6%88%90%E7%BB%A9.md" \o "评定成绩)

* 功能： 设置一个学生的部分评分项成绩和评语  
  输入参数result不为空，自动设置update\_date为当前日期，表示同时设置批改日期  
  输入参数result为空，自动设置update\_date为空，表示未批改。
* 权限： 老师：可以查看选自己课的所有学生的成绩，管理员可以查看所有课程的所有学生的成绩
* API请求地址： 接口基本地址/v1/api/setOneStudentResults
* 请求方式 ： POST
* 请求实例：

{

"user\_id": "201610414414",

"total": 6,

"data": [

{

"scoreItems\_id":1,

"score\_result": 91,

"score\_describe":"本实验做得好",

},

{ ...其他实验 }

]

}

* 请求参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| user\_id | 学号 |
| total | 本次批改的所有评分项的总数，小于等于全部评分项的总数 |
| data | 评分项的成绩和评语 |
| scoreItems\_id | 评分项id |
| score\_result | 评分项得分 |
| score\_describe | 评语 |

* 返回实例：

{

"status": true,

"info": null,

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |

1. 接口:studentList

用例:[学生列表](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E5%AD%A6%E7%94%9F%E5%88%97%E8%A1%A8.md" \o "学生列表)

* 功能： 查看学生列表
* 权限： 老师/管理员登录。
* API请求地址： 接口基本地址/v1/api/studentList/
* 请求方式 ： GET
* 请求参数说明:

|  |  |  |
| --- | --- | --- |
| 参数 | 说明 | 可以为空 |
| lesson\_id | 课程id | 是，为空代表查询所有学生，不为空查询对应课程的学生列表 |

* 返回实例：

{

"status": true,

"info": null,

"total": 120,

"data": [

{

"user\_id": "201510315203",

"student\_class": "软件(本)15-1",

"user\_name": "陈松华",

"update\_date": "2018-04-02 13:48:01"

"github\_username":"mzy1997"

"lesson\_average":95},

{

...其他学生

}

]

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |
| total | 返回学生总数 |
| data | 所有学生数组 |
| user\_id | 学号 |
| student\_class | 学生班级 |
| user\_name | 学生姓名 |
| update\_date | 学生github修改时间 |
| github\_username | github用户名 |
| lesson\_average | 课程平均分 |

1. 接口:teacherList

用例:[教师列表](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E6%95%99%E5%B8%88%E5%88%97%E8%A1%A8.md" \o "教师列表)

* 功能： 查看教师列表
* 权限： 管理员登录。
* API请求地址： 接口基本地址/v1/api/teacherList
* 请求方式 ： GET
* 返回实例：

{

"status": true,

"info": null,

"total": 60,

"data": [

{

"user\_id": "123454656",

"user\_name": "本尼迪克特",

"teacher\_department": "信工"

{

...其他教师

}

]

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |
| total | 返回教师总数 |
| data | 所有教师数组 |
| user\_id | 职工号 |
| user\_name | 教师姓名 |
| teacher\_department | 教师所在学院 |

1. 接口:updateLesson

用例:[维护课程信息](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E7%BB%B4%E6%8A%A4%E8%AF%BE%E7%A8%8B%E4%BF%A1%E6%81%AF.md" \o "维护课程信息)

* 功能： 更新课程信息
* 权限： 管理员登录。
* API请求地址： 接口基本地址/v1/api/updateLesson
* 请求方式 ： POST
* 请求实例：

{

"lesson\_id":123456

"lesson\_name":"大学英语",

"lesson\_type":"公共基础课"

"lesson\_describe":"本书为远程教育大学英语基础课教材，由十课组成;每课包括课文，词汇，注释，语音或语法，练习，补充课文等。文章语言规范，语法讲解透彻，增强了教材的实用性。"

"user\_name":"本尼迪克特"

}

* 请求参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| lesson\_id | 课程id |
| lesson\_name | 课程名 |
| lesson\_type | 课程类型 |
| lesson\_describe | 课程简介 |
| user\_name | 授课教师名 |

* 返回实例：

{

"status": true,

"info": null,

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |

1. 接口:updateUser

用例:[维护学生信息](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E7%BB%B4%E6%8A%A4%E5%AD%A6%E7%94%9F%E4%BF%A1%E6%81%AF.md" \o "维护学生信息)[维护教师信息](https://github.com/mzy1997/is_analysis/blob/master/test6/%E7%94%A8%E4%BE%8B/%E7%BB%B4%E6%8A%A4%E6%95%99%E5%B8%88%E4%BF%A1%E6%81%AF.md" \o "维护教师信息)

* 功能： 更新学生或教师信息,如果数据库内无此user\_id，则增加新字段
* 权限： 管理员登录。
* API请求地址： 接口基本地址/v1/api/updateUser
* 请求方式 ： POST
* 请求实例：

{

"user\_id":201610414414

"user\_name":"穆峥言",

"student\_department":"信工"

"student\_major":"软件工程"

"user\_grade":"16级"

"github\_username":mzy1997

"type":"学生"

}

* 请求参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| user\_id | 用户id，对应学生学号，教师职工号 |
| type | 用户类型 |
| user\_name | 用户名 |
| student\_department | 学生所属学院 |
| student\_major | 学生专业 |
| user\_grade | 学生年级 |
| github\_username | github用户名 |

* 返回实例：

{

"status": true,

"info": null,

}

* 返回参数说明:

|  |  |
| --- | --- |
| 参数 | 说明 |
| status | bool类型，true表示正确的返回，false表示有错误 |
| info | 返回结果说明信息 |