

《数据库系统原理》大作业 系统实现报告

题目名称：教务管理系统

学号及姓名：78066009 苏嘉娴

78066014 邓奇恩

78066015 谢宇隆

2020 年 12 月 5 日

一、系统结构设计

1.1 体系结构

该作业做的是一款小型的教务管理系统。用户可在这款管理系统中以相应的用户类型来使用。每个用户使用前都需要输入自己的用户号和密码，如密码正确就可登入系统，系统会自动辨别用户的类型。所有用户登入之后都可自由更改密码与登出。用户不能自己注册以在系统中创建账号，只有管理员有权限创建账号，然后把账号分发给用户使用。

所有用户都有权限进行的操作

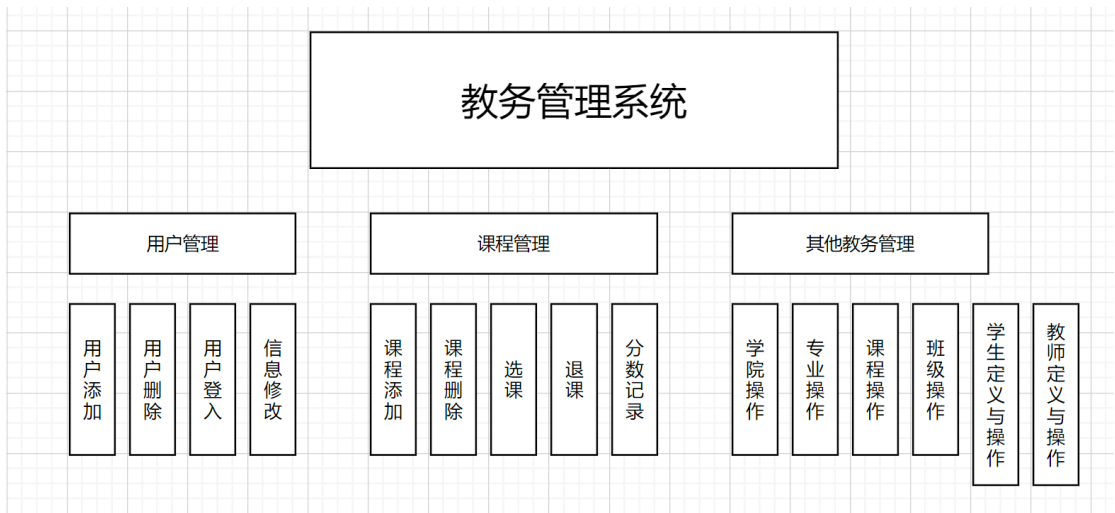
- a) 退出登录
- b) 更改密码

系统中用户类型一共有三类：

- 学生
 - a) 可查看自己的个人信息。
 - b) 可查看自身已选的所有课程，如列表为空则该学生未选课。
 - c) 在已选课程列表中可做退课操作。
 - d) 可查看目前可选的所有课程，如列表为空则该表示没有课程开放或所有课程都已选择。
 - e) 在可选课程列表中可选课。
- 教师
 - a) 可查看自己的个人信息。
 - b) 可查看自己负责教学的课程。
 - c) 可查看选择自己负责教学的课程的学生列表。
 - d) 可以给选择自己负责教学的课的学生打分数。
- 管理员
 - 可对如下所有信息做查看、增加、删除与更改操作。
 - a) 用户信息
 - b) 课程信息
 - c) 选课信息
 - d) 学院信息
 - e) 专业信息
 - f) 班级信息
 - g) 学生信息
 - h) 教师信息

1.2 功能结构

系统的功能主要有用户管理、课程管理、和其他教务系统管理三大部分，如下图所示。



用户管理包括管理员添加和删除用户的操作、三类用户的登入和信息修改如更改密码。

课程管理分成三大部分，分别是管理员负责操作的课程添加和删除，学生的选课和退课操作以及教师对学生所选的课的分数的记录。

其他教务管理有六大部分，不过由于他们全都由管理员负责管理，添加、删除、修改和查询都需要管理员的权限，虽然分布不同不过在功能实现上都是一致，因此在功能介绍中就将他们一同归类。

学生可查询自己用户信息，如所属学院、专业、班级等，还可进行选课退课操作与修改用户密码，教师可查询自己用户信息和更改学生所选择的自身负责教导的课程的分数，而管理员则可进行一切操作，如对用户的添加删除，查询和更改等，还有学院、专业、课程、班级、学生及教师都是如此。

二、数据库基本表的定义

1. 创建数据库：数据库名为 school_database。

```
CREATE database school_database;
```

2. 用户信息表

包括用户 id，密码和用户类别（admin，student，teacher 其中一类）。主码是用户 id。

```
CREATE TABLE user(  
    user_id VARCHAR(100),  
    password VARCHAR(100),  
    user_type VARCHAR(100),  
    PRIMARY KEY (user_id)  
);
```

3. 学院信息表

包括学院 id 和学院名字。主码是学院 id。

```
CREATE TABLE department(  
    department_id VARCHAR(100),  
    name VARCHAR(100),  
    PRIMARY KEY (department_id)  
);
```

4. 专业信息表

包括专业 id、专业名、学院 id。主码是专业 id。学院 id 是外码。

```
CREATE TABLE major(  
    major_id VARCHAR(100),  
    name VARCHAR(100),  
    department_id VARCHAR(100),  
    PRIMARY KEY(major_id),  
    FOREIGN KEY (department_id) REFERENCES department(department_id) ON  
DELETE CASCADE  
);
```

5. 班级信息表

包括班级 id、年级、专业 id。主码是班级 id。专业 id 是外码。

```
CREATE TABLE class(  
    class_id VARCHAR(100),  
    year VARCHAR(100),  
    major_id VARCHAR(100),  
    PRIMARY KEY (class_id),  
    FOREIGN KEY (major_id) REFERENCES major(major_id) ON DELETE CASCADE  
);
```

6. 学生信息表

包括学生 id、学生名、班级 id、学生已选课程数量。主码是学生 id。班级 id 是外码。

```
CREATE TABLE student(  
    student_id VARCHAR(100),  
    name VARCHAR(100),  
    class_id VARCHAR(100),  
    course_cnt INT DEFAULT 0,  
    PRIMARY KEY(student_id),  
    FOREIGN KEY (class_id) REFERENCES class(class_id) ON DELETE CASCADE  
);
```

7. 教师信息表

包括教师 id、教师名、学院 id。主码是教师 id。学院 id 是外码。

```
CREATE TABLE teacher(  
    teacher_id VARCHAR(100),  
    name VARCHAR(100),  
    department_id VARCHAR(100),  
    PRIMARY KEY(teacher_id),  
    FOREIGN KEY (department_id) REFERENCES department(department_id) ON  
DELETE CASCADE  
);
```

8. 课程信息表

包括课程 id、课程名、教师 id、学院 id。主码是课程 id 与教师 id。同时课程 id 和教师 id 都是外码。

```
CREATE TABLE course(  
    course_id VARCHAR(100),  
    name VARCHAR(100),  
    teacher_id VARCHAR(100),  
    department_id VARCHAR(100),  
    PRIMARY KEY(course_id, teacher_id),  
    FOREIGN KEY(teacher_id) REFERENCES teacher(teacher_id) ON DELETE  
CASCADE,  
    FOREIGN KEY (department_id) REFERENCES department(department_id) ON  
DELETE CASCADE  
);
```

9. 选课信息表

包括课程 id、学生 id、分数。主码是课程 id 和学生 id。同时课程 id 和学生 id 是外码。分数必须大于等于 0 和小于等于 100。

```
CREATE TABLE courseSelection(  
    course_id VARCHAR(100),  
    student_id VARCHAR(100),  
    grade VARCHAR(100) CHECK (CAST(grade AS SIGNED) >= 0 AND CAST(grade AS  
SIGNED) <= 100) ,  
    PRIMARY KEY (course_id, student_id),  
    FOREIGN KEY (course_id) REFERENCES course(course_id) ON DELETE CASCADE,  
    FOREIGN KEY (student_id) REFERENCES student(student_id) ON DELETE  
CASCADE  
);
```

三、系统重要功能实现方法

1. 使用方式

按照上方的数据库基本表的定义在 MySQL 数据库建立数据库实例。然后打开两个命令行，一个的 path 是前端文件夹的主目录，另一个的 path 是后端文件夹的主目录。然后在两个命令行分别输入“npm start”，程序就可以在浏览器上运行了，运行的地址是 <http://localhost:3000/>（前端地址）。

2. 开发环境与技术说明

程序分为前端和后端。前端使用了 React 框架，开发语言是 JavaScript。后端使用的是 Express 框架，开发语言是 Node.js。使用的数据库管理系统是 MySQL。

开发过程中使用了前后端分离的方法。前端地址在 <http://localhost:3000/>，后端地址在 <http://localhost:3001/>。

前端通过特定的 API 地址向后端发送请求，而后端通过 Node.js 的 mysql 模块连接本地的 MySQL 数据库，向本地的 MySQL 数据库发送请求。

系统与数据库的交互大多比较简单，而复杂的查询或修改指令被设计成了数据库上的触发器、存储过程或视图，这样，后端的 sql 查询语句只要对存储过程或视图进行简单的调用就可以了实现冗长的指令操作了。

此外，为了不把用户密码的明文存储到数据库，在后端使用了对密码加密的模块 bcrypt，因此存储到数据库的密码是加密后的密码。判定用户输入的密码是否对得上数据库里加密后的密码，也是使用 bcrypt 模块来判定。

调用 Node.js 的 mysql 模块的方法如下，其中数据库的用户名是‘root’，密码是‘password’。

```
var mysql = require('mysql');

var connection = mysql.createConnection({
  host: "localhost",
  user: "root",
  password: "password",
  database: "school_database"
});

connection.connect(function(err) {
  if (err) throw err;
  console.log("Connected!");
});
```


3. 触发器、存储过程、与视图的设计与实现说明

a. 触发器 `trigger_student_course_cnt_increase`

当学生选课时，把学生的选课数量属性增加 1。

```
DELIMITER $$
CREATE TRIGGER trigger_student_course_cnt_increase
  AFTER INSERT
  ON courseSelection FOR EACH ROW
BEGIN
  UPDATE student SET course_cnt = course_cnt + 1 WHERE student_id =
new.student_id;
END$$
DELIMITER ;
```

b. 触发器 `trigger_student_course_cnt_decrease`

当学生退课时，把学生的选课数量属性减 1。

```
DELIMITER $$
CREATE TRIGGER trigger_student_course_cnt_decrease
  BEFORE DELETE
  ON courseSelection FOR EACH ROW
BEGIN
  UPDATE student SET course_cnt = course_cnt - 1 WHERE student_id =
old.student_id;
END$$
DELIMITER ;
```

c. 存储过程 `StudentSelectedCourses(In check_student_id VARCHAR(100))`

某学生已选择的课程。

```
DELIMITER $$
CREATE PROCEDURE StudentSelectedCourses(In check_student_id VARCHAR(100))
BEGIN
  SELECT courseselection.student_id, courseselection.course_id,
    course.name course_name, teacher.name teacher_name,
    department.name department_name , grade
  FROM courseselection
  JOIN course ON courseselection.course_id = course.course_id
  JOIN teacher ON course.teacher_id = teacher.teacher_id
  JOIN department ON course.department_id = department.department_id
  WHERE courseselection.student_id = check_student_id;
END$$
DELIMITER ;
```

- d. 存储过程 StudentNotSelectedCourses(In check_student_id
VARCHAR(100))
某学生未选择的课程。

```
DELIMITER $$
CREATE PROCEDURE StudentNotSelectedCourses(In check_student_id
VARCHAR(100))
BEGIN
    SELECT course.course_id,
           course.name course_name, teacher.name teacher_name,
           department.name department_name
    FROM course
    JOIN teacher ON course.teacher_id = teacher.teacher_id
    JOIN department ON course.department_id = department.department_id
    WHERE course.course_id NOT IN (SELECT course.course_id
                                   FROM course, courseselection
                                   WHERE      courseselection.course_id=course.course_id      AND
courseselection.student_id = check_student_id);
END$$
DELIMITER ;
```

- e. 存储过程 GetStudentInfo(In check_student_id VARCHAR(100))
某学生的个人信息。

```
DELIMITER $$
CREATE PROCEDURE GetStudentInfo(In check_student_id VARCHAR(100))
BEGIN
    SELECT student.student_id, student.name student_name, course_cnt,
           class.class_id,      class.year,      major.major_id,      major.name
major_name,
           department.department_id, department.name department_name
    FROM student, class, major, department
    WHERE check_student_id = student.student_id AND class.class_id =
student.class_id      AND      major.major_id      =      class.major_id      AND
department.department_id = major.department_id;
END$$
DELIMITER ;
```

- f. **存储过程 GetTeacherInfo(In check_teacher_id VARCHAR(100))**
某教师的个人信息。

```
DELIMITER $$
CREATE PROCEDURE GetTeacherInfo(In check_teacher_id VARCHAR(100))
BEGIN
    SELECT teacher_id, teacher.name teacher_name, teacher.department_id,
    department.name department_name
    FROM teacher
    JOIN department ON department.department_id = teacher.department_id
    WHERE teacher_id = check_teacher_id;
END$$
DELIMITER ;
```

- g. **存储过程 StudentsTakenCourse(In check_course_id VARCHAR(100))**
某课程的学生名单。

```
DELIMITER $$
CREATE PROCEDURE StudentsTakenCourse(In check_course_id VARCHAR(100))
BEGIN
    SELECT courseselection.course_id, student.student_id, student.name
    student_name, grade
    FROM courseselection, student
    WHERE courseselection.course_id=check_course_id AND student.student_id
    = courseselection.student_id;
END$$
DELIMITER ;
```

- h. **视图 TeachingCourses**
某教师负责教授的课。

```
CREATE VIEW TeachingCourses
AS
SELECT course.teacher_id, course.course_id, course.name course_name,
department.name department_name
FROM course, department
WHERE department.department_id=course.department_id;
```

四、系统实现结果

1. 首先进入登录界面。

Welcome. Please log in.

User Id

Password

Confirm

Created for BUAA database project, by Samuel, Eric, and JiaXian.

2. 输入 user id 和 password 后点击 confirm 按钮进行登录。

Welcome. Please log in.

User Id

adminroot

Password

....

Confirm

Created for BUAA database project, by Samuel, Eric, and JiaXian.

3. 若输入的 user id 不存在，或 password 不匹配，则会弹出报错窗口。

Welcome. Please log in.

User Id

adminroo

Password

...

Confirm

localhost:3000 says
log in failed
Error: user_id does not exist

OK

Created for BUAA database project, by Samuel, Eric, and JiaXian.

Welcome. Please log in.

User Id

adminroot

Password

...

Confirm

localhost:3000 says
log in failed
Error: incorrect password

OK

4. 登录成功后进入到主页面。主页显示着用户姓名(这里用户名是 adminroot)、用户类型(这里用户类型是 admin)、以及所有用户可访问的页面。此页面显示用户可以退出登录（点击 log out 退出并跳转到登录界面）、修改密码、以及修改各个信息表的内容。

School Management System

Welcome, **adminroot**

Log Out

Change Password

Your role is: **admin**

Users Info

Students Info

Teachers Info

Course Catalog

Course Selections

Departments

Majors

Classes

Created for BUAA database project, by Samuel, Eric, and JiaXian.

5. 点击 Students Info 按钮，进入学生信息页面。在此页面中可以添加学生表项、修改学生信息（只能修改 name 和 class id，不能修改 student id）、删除学生表项、搜索学生信息表中的内容、以及点击 return 跳转到主页面。

Students Information

Insert Student

return

Search..

| StudentID | Name | ClassID | | |
|-----------|-----------|---------|-------------------|-------------------|
| 2321 | rwqty | 234 | <div>delete</div> | <div>update</div> |
| 78066014 | samuelabc | 182511 | <div>delete</div> | <div>update</div> |

Created for BUAA database project, by Samuel, Eric, and JiaXian.

6. 点击 insert student 按钮以添加学生。填入学生信息后，点击 confirm，若成功添加学生，学生表项会出现在下方的学生信息表上；若添加失败，弹出报错窗口。下方的图中是添加成功了，新添加的学生表项出现在下方的学生信息表。

Students Information

Insert Student

StudentID

78066017

Name

eric

ClassID

182511

Cancel

Confirm

| StudentID | Name | ClassID | | |
|-----------|-----------|---------|-------------------|-------------------|
| 2321 | rwqty | 234 | <div>delete</div> | <div>update</div> |
| 78066014 | samuelabc | 182511 | <div>delete</div> | <div>update</div> |
| 78066017 | eric | 182511 | <div>delete</div> | <div>update</div> |

Created for BUAA database project, by Samuel, Eric, and JiaXian.

7. 在主页面中选择 users info，进入到用户信息页面。此页面会显示系统已录入的用户信息（数据库中的密码是加密后的密码，数据库并不会存储用户的密码明文），并且可以在此页面添加用户、修改用户信息、搜索用户信息表、以及点击 return 返回主页面。

User Information

Insert User

return

Search..

Press

F11

 to exit full screen

| UserID | Password | UserType | | |
|-----------|---|----------|-------------------|-------------------|
| 2321 | \$2b\$10\$aIv5vhJzGJA3fTZywcnsSeLCXhVGVsWx41u82XBSL19.RaceO.z7G | student | <div>delete</div> | <div>update</div> |
| 78066014 | \$2b\$10\$fxrujVmtslBE89GVrBPY3OXZNktjXaqihrzeORJ36Ewk/cfJXCV. | student | <div>delete</div> | <div>update</div> |
| adminroot | 1234 | admin | <div>delete</div> | <div>update</div> |
| J1064 | \$2b\$10\$OJ5bzBiGdsLgFo9zXvSlmOATCJM.17yJi0gNNnYbEoYcvdzGpd6tG | teacher | <div>delete</div> | <div>update</div> |
| J1067 | \$2b\$10\$65v8Np3rNDpH6Sxnjt9B...wbZxWNn1GzK.aRFiksVdY6gtiKccxC | teacher | <div>delete</div> | <div>update</div> |

8. 点击 insert user 按钮以添加用户。把要添加的用户信息填入，然后需要选择用户类别，这里选择了 student 类。接着点击 confirm 按钮确定添加，若添加成功，该用户信息会出现在下方的用户信息表中。

User Information

Insert User

UserlD

78066017

Password

....

Confirm Password

....

UserType

student

admin

student

teacher

| UserlD | Password | UserType | | |
|-----------|---|----------|--------|--------|
| 2321 | \$2b\$10\$aIV5vhJzGJA3fTZywcnsSeLCXhVGVsWx41u82XBSL19.RaceO.z7G | student | delete | update |
| 78066014 | \$2b\$10\$fxrujVmtslBE89GvRBPY3OXZNktjXaqihrzeORJ36Ewk/cfJXCV. | student | delete | update |
| 78066017 | \$2b\$10\$KRBop5RgIP2WDWz.KFWluuRvD4eLiHsR/OajxJVNmhX/VaO9QmsYm | student | delete | update |
| adminroot | 1234 | admin | delete | update |
| J1064 | \$2b\$10\$OJ5bzBiGdsLgFo9zXvSlmOATCJM.17yJi0gNNnYbEoYcvdzGpd6tG | teacher | delete | update |
| J1067 | \$2b\$10\$65v8Np3rNDpH6Sxnjt9B...wbZxWNn1GzK.aRFiksVdY6gtiKccxC | teacher | delete | update |

Created for BUAA database project, by Samuel, Eric, and JiaXian.

9. 在主页面中点击 change password，进入更改密码页面。添入旧密码以及新密码后，若更改成功，跳转到主页面，并弹出窗口显示更改密码成功。

User id: **adminroot**

Old password

....

New password

...

Confirm new password

...!

Confirm

return

Created for BUAA database project, by Samuel, Eric, and JiaXian.

School Management System

Welcome, **adminroot**

Log Out

Change Password

Your role is: **admin**

Users Info

Students Info

Teachers Info

Course Catalog

Course Selections

Departments

Majors

Classes

Created for BUAA database project, by Samuel, Eric, and JiaXian.

localhost:3000 says
Successfully change password!
OK

10. 点击 log out 退出登录。现使用一个学生账号登录。

Welcome. Please log in.

User Id

2321

Password

....

Confirm

Created for BUAA database project, by Samuel, Eric, and JiaXian.

11. 登录后进入到学生主页面。主页面显示着学生用户 id（2321）和用户类型（student）。学生可以进行退出登录、修改密码、查看学生个人信息、查看已选择的课程列表和查看未选择的课程列表。

School Management System

Welcome, 2321

Log Out

Change Password

Your role is: student

Student Info

Selected Courses

Courses Catalog

Created for BUAA database project, by Samuel, Eric, and JiaXian.

12. 点击 student info, 进入学生个人信息页面。个人信息页面中显示着学生的 id、名字、班级 id、进入学校的年份、专业 id、专业名、学院 id、学院名、以及已选课程的数量。点击 return 返回主页面。

Student Info

Student Id : **2321**

Student Name : **rwqty**

Class Id : **234**

Year : **2016**

Major Id : **0512**

Major Name : **aircraft design**

Department Id : **05**

Department Name : **aerospace**

Number of selected courses: **3**

return

Created for BUAA database project, by Samuel, Eric, and JiaXian.

13. 在主页面中点击 selected courses 按钮，进入学生已选课程页面。在已选课程页面中可以搜索已选课程信息，以及进行退课操作。

Student Selected Course

Search..

| Courseld | CourseName | TeacherName | DepartmentName | Grade | |
|----------|------------|-------------|----------------|-------|------------------------|
| CS129 | java | Edwin | aerospace | 0 | <div>Drop course</div> |
| EE122 | Wave | Wong | aerospace | 0 | <div>Drop course</div> |
| EE236 | Algorithm | Wong | CS | 0 | <div>Drop course</div> |

return

Created for BUAA database project, by Samuel, Eric, and JiaXian.

14. 尝试把表中的第二个课程 EE122 退掉。点击 drop course 按钮后，弹出确认窗口，点击 OK，若操作成功，下方的已选课程表中就不存在这门课了，并且被退掉的这门课会出现在可选课程页面中。

Student Selected Course

Search..

localhost:3000 says
confirm to drop this course?

OK

Cancel

| Courseld | CourseName | TeacherName | DepartmentName | Grade | |
|----------|------------|-------------|----------------|-------|------------------------|
| CS129 | java | Edwin | aerospace | 0 | <div>Drop course</div> |
| EE122 | Wave | Wong | aerospace | 0 | <div>Drop course</div> |
| EE236 | Algorithm | Wong | CS | 0 | <div>Drop course</div> |

Student Selected Course

Search..

| Courseld | CourseName | TeacherName | DepartmentName | Grade | |
|----------|------------|-------------|----------------|-------|------------------------|
| CS129 | java | Edwin | aerospace | 0 | <div>Drop course</div> |
| EE236 | Algorithm | Wong | CS | 0 | <div>Drop course</div> |

15. 在学生主页面中点击 course catalog 按钮，进入可选课程页面（此学生未选择的课程）。可以在此页面搜索课程信息， 以及进行选课操作。

Course Catalog

Courses that still can be selected, selected courses are not included in this table.

Search..

| Courseld | CourseName | TeacherName | DepartmentName | |
|----------|------------|-------------|----------------|-------------|
| EE122 | Wave | Wong | aerospace | Take course |
| EE125 | Electronic | Wong | aerospace | Take course |

return

Created for BUAA database project, by Samuel, Eric, and JiaXian.

16. 点击第一门课 EE122 的 take course 按钮进行选课操作。跳出确认按钮，点击确认后，该课就从可选课程表中删除了，同时该课会出现在已选课程表中。

Course Catalog

Courses that still can be selected, selected courses are not in

Search..

| Courseld | CourseName | TeacherName | DepartmentName | |
|----------|------------|-------------|----------------|-------------|
| EE122 | Wave | Wong | aerospace | Take course |
| EE125 | Electronic | Wong | aerospace | Take course |

localhost:3000 says
confirm to take this course?
OK Cancel

18. 接下来我们对学生账户进行登出，然后在登录界面使用一个教师账户进行登录。登录后教师主页面显示此用户 id (J1067) 和用户类型 (teacher)。此用户可以登出、修改密码、查看教师个人信息和查看此教师教授的所有课程。

School Management System

Welcome, **J1067**

Log Out

Change Password

Your role is: **teacher**

Teacher Info

Teaching Courses

Created for BUAA database project, by Samuel, Eric, and JiaXian.

19. 点击 teacher info 按钮进入教师个人信息页面。此页面显示教师 id、教师名字、教师属于的院系和院系名字。点击 return 会返回教师主页面。

Teacher Info

Teacher Id : **J1067**

Teacher Name : **Lee**

Department Id : **05**

Department Name : **aerospace**

return

Created for BUAA database project, by Samuel, Eric, and JiaXian.

20. 在教师主页面点击 teaching courses 按钮，进入教师的教授课程信息页面。此教师没有开课，所以课程信息页面中没有课程信息列表。

Teaching Courses

Currently no teaching courses.

return

Created for BUAA database project, by Samuel, Eric, and JiaXian.

21. 切换到另一个有开课的教师的账号，并查看其开课信息。可以看到 teaching courses 页面上显示着此教师开设了 3 门课，其中 2 门课是为 aerospace 学院开的，1 门课是为 CS 学院开的。

Teaching Courses

- EE122 Wave, offered by department of aerospace
- EE125 Electronic, offered by department of aerospace
- EE236 Algorithm, offered by department of CS

Select a course to see details below.

EE122 Wave

List of students taking EE122:

Search..

| Courseld | StudentId | StudentName | Grade | |
|----------|-----------|-------------|-------|-------------------------|
| EE122 | 2321 | rwqty | 0 | <div>Update grade</div> |

return

Created for BUAA database project, by Samuel, Eric, and JiaXian.

23. 可以选择查看其中一门课的学生列表。并且可以搜索学生信息、以及点击 upgrade grade 按钮为学生打分。点击 upgrade grade 并确认后，学生的分数就成功更新了。

Select a course to see details below.

EE122 Wave

EE122 Wave

EE125 Electronic

EE236 Algorithm

Search..

| Courseld | StudentId | StudentName | Grade |
|----------|-----------|-------------|-------|
| EE122 | 2321 | rwqty | 0 |

Update grade

Teaching Courses

- EE122 Wave, offered by department of aerospace
- EE125 Electronic, offered by department of aerospace
- EE236 Algorithm, offered by department of CS

Select a course to see details below.

EE122 Wave

List of students taking EE122:

Search..

localhost:3000 says

confirm update?

OK

Cancel

| Courseld | StudentId | StudentName | Grade |
|----------|-----------|-------------|-------|
| EE122 | 2321 | rwqty | 90 |

cancel

confirm

| Courseld | StudentId | StudentName | Grade |
|----------|-----------|-------------|-------|
| EE122 | 2321 | rwqty | 90 |

Update grade

五、总结

邓奇恩:

我觉得这个数据库大作业对我来说是一个很大的挑战, 因为对开发网页没有什么经验, 也是第一次使用 javascript, 和 react.js, node.js 这些框架写程序, 由于对语言和框架都不熟悉, 因此写程序时总是会冒出奇怪的 bug, 然后要花费好多时间上网查解决方法。

然后系统前端的 UI 做得还是比较简约的, 没有花费特别多时间在设计用户界面上, 但是我觉得设计得还是比较符合用户习惯的, 用户很容易就知道该如何操作了。

虽然这次大作业我们的选题是教务管理系统, 是一个很俗套的题目, 但是这是我第一次开发带数据库的网页系统, 我觉得收获还是很大的, 对数据库的指令熟悉了很多, 然后也累积了一些搭建网页的经验。

谢宇隆:

通过本次数据库课程设计, 我在一个较短的时间内学到了很多知识和技巧, 虽然这个系统较为简单。在上这门课之前我完全不了解从前端到后端再到数据库之间是以怎样的方式来运作, 然而这次课程设计给了我机会接触许多这方面的新知识的。在开发过程中我学习了从基础的前后端语法知识再到各种框架的结构以及思想, 学到了许多关于数据库开发的大量知识。

除此之外, 我体会了团队协同开发的重要性, 也积累了宝贵的经验。我想感谢我的同伴, 在我们都毫无头绪时他带着我们开始讨论整个作业的方向。之后我们分工明确, 积极交流, 使我们能够在短时间内高效的地协同完成此次的任务。此外, 这次课程设计也让我体会到数据库设计的重要性, 一个设计规范的数据库可以在应用中让开发人员省去很多不必要的操作, 良好的设计能够保证数据库的安全性和数据完整性, 从而使整个项目更加健壮。

总的来说, 体验了这样一个完整的项目设计, 我受益匪浅。

苏嘉娴:

通过这次的实验让我从一开始对数据库的不熟悉到对数据库的深入了解, 学习了许多关于数据知识如: mysql、odbc, 以及前端与后端的认识。在这次的课程设计中我也学习了许多框架, 并运用所学到的知识开发网页系统。在开发的过程中我认为团队的合作是不可缺的, 还有对前端、后端的知识。我们需要足够了解这些知识才能开发出完整的网页系统。因此, 在一开始开发的时候面临了许多困难, 但随着不断的学习, 排解了所遇到的难题后开始对完善我们所设计的网页系统。通过这次的课程设计, 我学到许多书本上所学不到的知识, 也让我了解到团队合作的重要性。