Bộ Giáo Dục Và Đào Tạo

**Trường Đại Học Ngoại Ngữ - Tin Học TP-Hồ Chí Minh**

**---o0o---**



**BÁO CÁO KẾT THÚC MÔN HỌC**

***ĐỀ TÀI:* XÂY DỰNG ỨNG DỤNG**

**CHO THUÊ NHÀ ĐẤT DỰA TRÊN HỆ QUẢN TRỊ MYSQL**

Khoa: Công Nghệ Thông Tin

Giảng viên hướng dẫn:  ThS. Lê Thị Minh Nguyện

Sinh viên thực hiện:

Phạm Ngọc Phú MSSV: 19DH110266

Trần Thanh Long MSSV:

Tp. Hồ Chí Minh - Tháng 1/2022

**GIỚI THIỆU**

## Ý nghĩa của đề tài

## Thành phố Hồ Chí Minh là một mảnh đất “màu mỡ” dành cho việc khởi nghiệp và kinh doanh, nơi đây cũng tồn tại nhiều trường đại học danh giá và nổi tiếng. Chính vì thế hằng năm sẽ có một lượng lớn người dân đổ xô về nơi đây để lập nghiệp và các bạn sinh viên đến đây để theo học tại các trường đại học. Bài toán giải quyết cho việc chỗ ở và nơi sinh sống là hết sức khó khăn và nan giải. Chính vì vậy nhóm chúng em đã cũng nhau thiết kế ra ứng dụng quản việc cho thuê nhà đất để đáp ứng cho người mua và người bán những tiện ích hấp dẫn nhất trong việc thuê nhà đất.

## Mục tiêu của đề tài

* Phân tích các yêu cầu của doanh nghiệp .
* Thiết kế dữ liệu phù hợp với yêu cầu của doanh nghiệp.
* Thiết kế xử lý phù hợp với yêu cầu của doanh nghiệp.
* Xây dựng cơ sở dữ liệu và trang web dựa trên các thiết kế.

# KHẢO SÁT HIỆN TRẠNG VÀ XÁC ĐỊNH YÊU CẦU

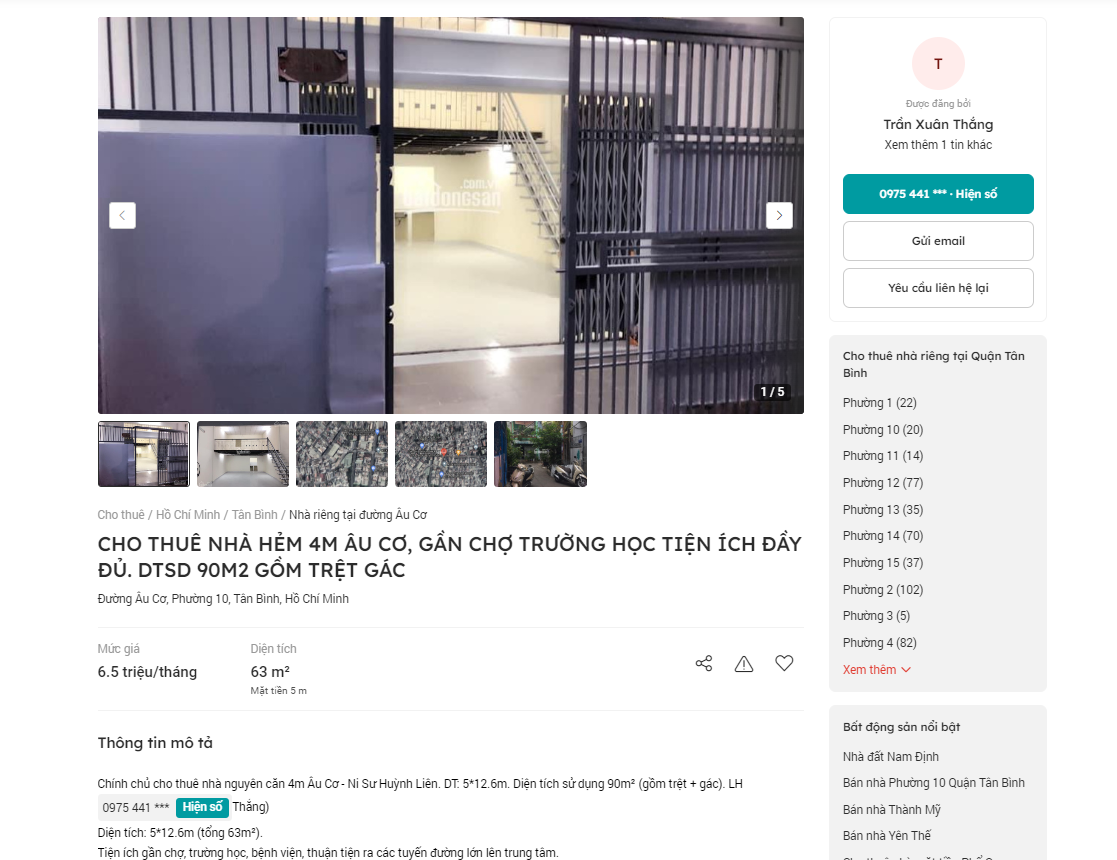
## Khảo sát nghiệp vụ

### Quản lý bài viết cho thuê nhà

Người quản lí quản lí các bài viết của bên cho thuê đăng lên:

* Các bài viết sẽ hiển thị các thông tin cơ bản của nhà được cho thuê để người thuê có thể dễ dàng theo dõi và lựa chọn cho bản thân một nơi ở hợp lí.
* Quản lí có thể theo dõi các bình luận , đánh giá , … từ đó có thể điều chỉnh lại nhà cửa trở nên hoàn hảo và đáp ứng mọi nhu cầu của người thuê.





### Đăng bài viết cho thuê nhà

Người cho thuê sẽ cho thuê nhà bằng cách đăng các bài viết cho thuê nhà lên trên website để cho người thuê dễ dàng theo dõi. Và khi đẫ cho thuê xong thì người cho thuê sẽ xóa bài viết hoặc đánh đấu là đã cho thuê.

Người thuê nhà cũng có thể biết được những căn nhà đang cho thuê bằng cách nghe nhiều người nói lại hay những tờ rơi được dán ở nhiều nơi trên đường phố.



#### Quy trình đăng một bài viết cho thuê nhà

1. Đề ra những nội dung cần thiết cho bài viết, những yêu cầu dành cho người thuê nhà, giá cả,….
2. Đăng bài viết lên website.
3. Đợi xét duyệt.

#### Quy định đăng bài viết, bình luận bài viết và đáh giá cho bài viết.

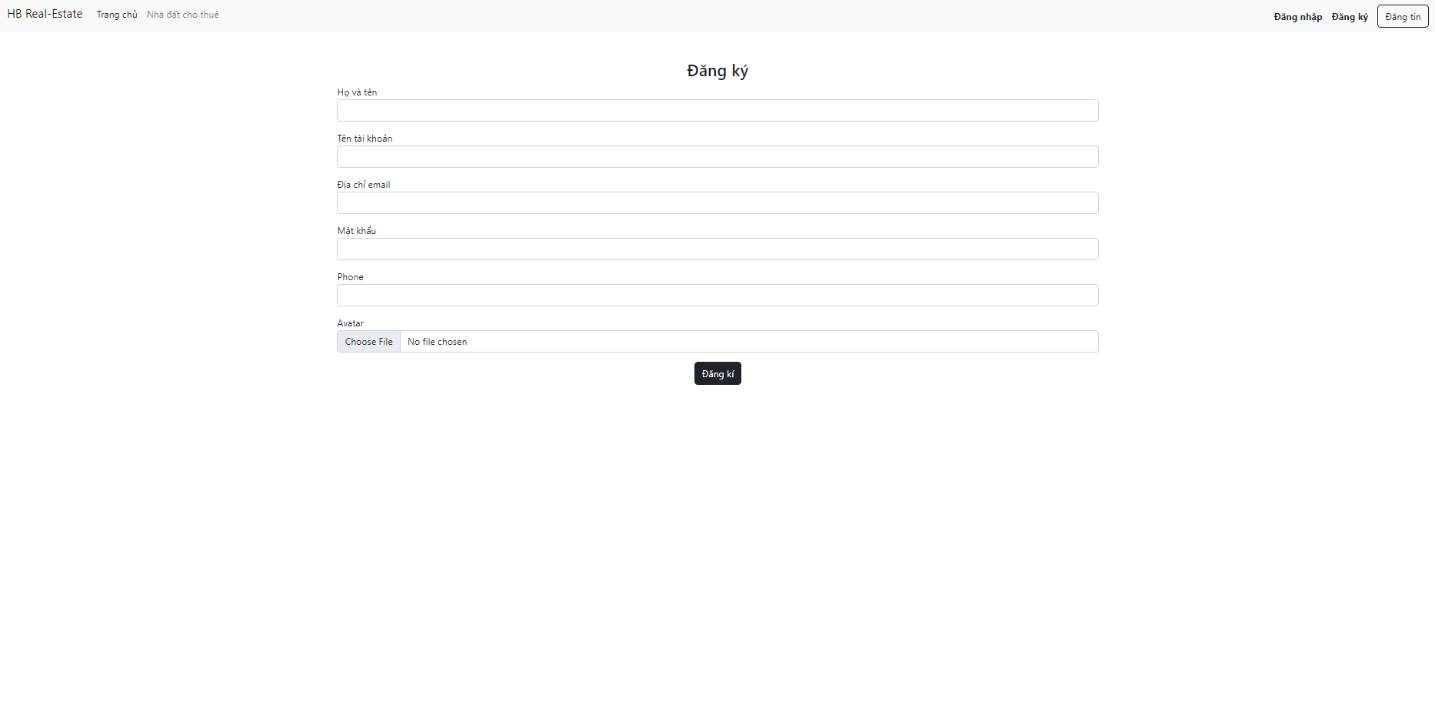
* Khi đăng bài viết thì bên cho thuê không được đăng cùng lúc 2 bài.
* Cần phải có tài khoản đã được đăng kí để có thể đăng bài viết cho thuê nhà trên website.

#### Bình luận bài viết và đáh giá cho bài viết.

* Người thuê có thể đăng bình luận cho bài viết thuê nhà nhưng chỉ được đăng 1 lần cho một bài viết.
* Người dung chỉ có thể báo cáo bài viết ( report ) một lần cho một bài viết.

### Đăng ký

Người dùng có thể đăng ký tài khoản của mình thông qua trang đăng ký của website.



#### Quy trình đăng ký

1. Người dùng vào trang đăng ký tài khoản của webstie.
2. Nhập đầy đủ thông tin và upload hình ảnh của bản than.\
3. Chọn nút đăng ký để đăng ký tài khoản

#### Quy định đăng ký

* Người phải điền đầy đủ các thông tin cá nhân.
* Tên tài khoản đăng ký không được trùng với các tài khoản đã được đăng ký trước đó.
* Mật khẩu phải đầy đủ các ký tự ( chữ hoa, chữ thường, số, ký tự đặc biệt ) và phải từ 8 ký tự trở lên.
* Số điện thoại phải đủ 10 chữ số.
* Địa chỉ email không được trùng với các email của các tài khoản đã được đăng ký và phải đúng định dạng của email.

### Duyệt bài viết

Quản trị hệ thống sẽ xem xét các bài viết có đủ tiêu chuẩn về nhà cho thuê, giá cả, có phù hợp với người thuê hay không. Sau đó sẽ duyệt để đưa lên và người thuê sẽ có thể thấy bài viết đó.

#### Quy trình duyệt bài viết

1. Quản trị hệ thống sau khi đăng nhập sẽ đi tới trang quản lí bài viết.
2. Kiểm tra các thông tin bài viết.
3. Nếu phù hợp thì xác nhận.

#### Quy định duyệt bài viết.

* Không được đăng lại một bài viết khác trùng với bài viết đã được đăng trước đó.
* Phải điền đầy đủ thông tin về nhà thuê, giá cả và các thông tin cụ thể để người thuê có thể yên tâm và dễ hiểu.

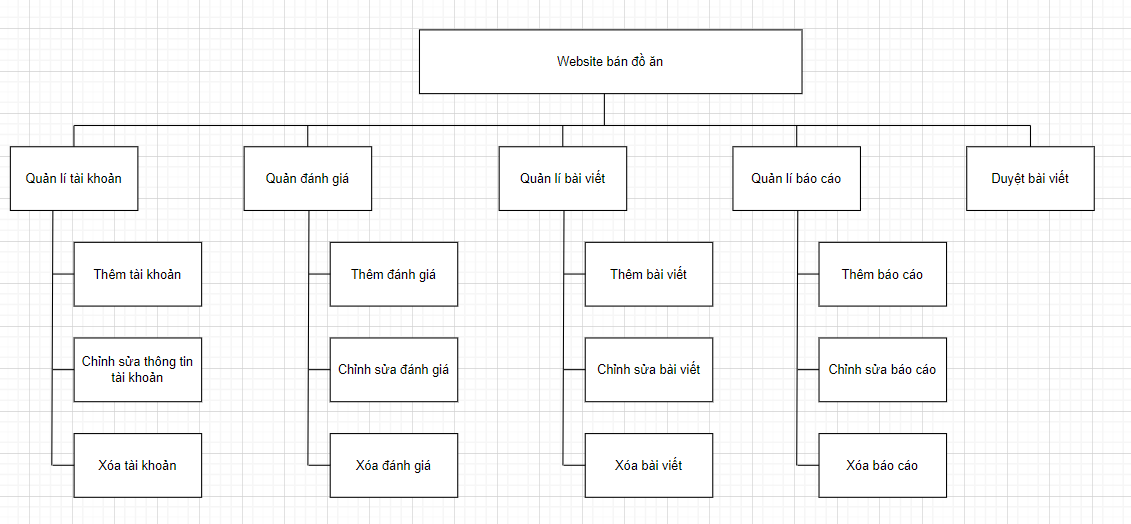
### Báo cáo

Người thuê cảm tháy có vấn đề gì với nơi thuê thì có thể báo cáo bài viết đó để cho các quản trị viên thấy và xem xét lại nơi cho thuê để giúp người thuê tránh khỏi những tình huống không đáng có.

#### Quy định báo cáo

## Yêu cầu

### Yêu cầu chức năng

****

### Yêu cầu phi chức năng

#### Quản lí tài khoản

* Đảm bảo tính bảo mật của hệ thống với người ngoài hệ thống
* Giới hạn các thông tin người dùng được truy cập theo vai trò của người dùng
* Giới hạn các giao tác người dùng có thể thực hiện theo vai trò của người dùng

#### Hiệu suất

* Thời gian phản hồi, thời gian xử lý và thời gian trả kết quả nhanh

#### Tiện dụng

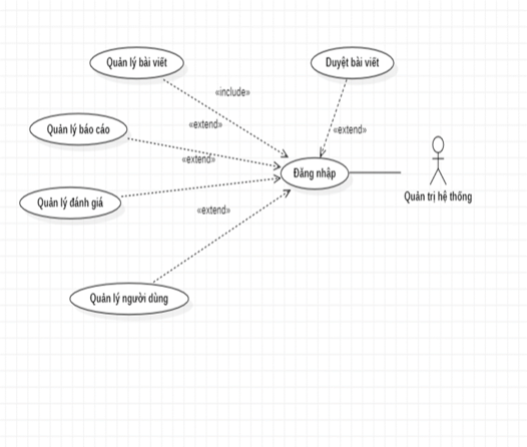
* Cung cấp đầy đủ các chức năng mà người dùng yêu cầu
* Dễ học cách sử dụng
* Hạn chế lỗi nhập liệu

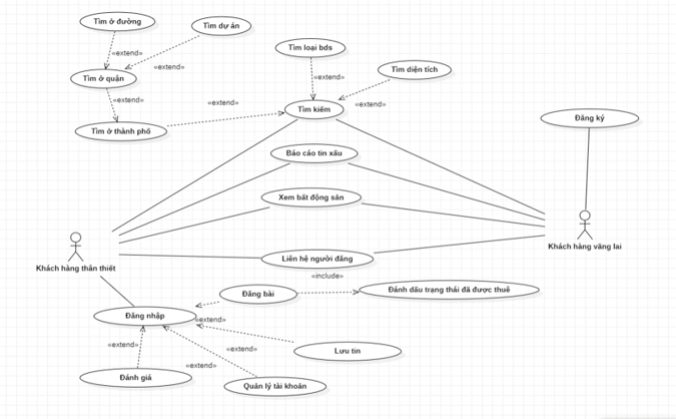
#### An toàn

* Hoạt động ổn định, chính xác
* Có thể khôi phục hệ thống, khôi phục dữ liệu sau các sự cố

# PHÂN TÍCH – THIẾT KẾ DỮ LIỆU

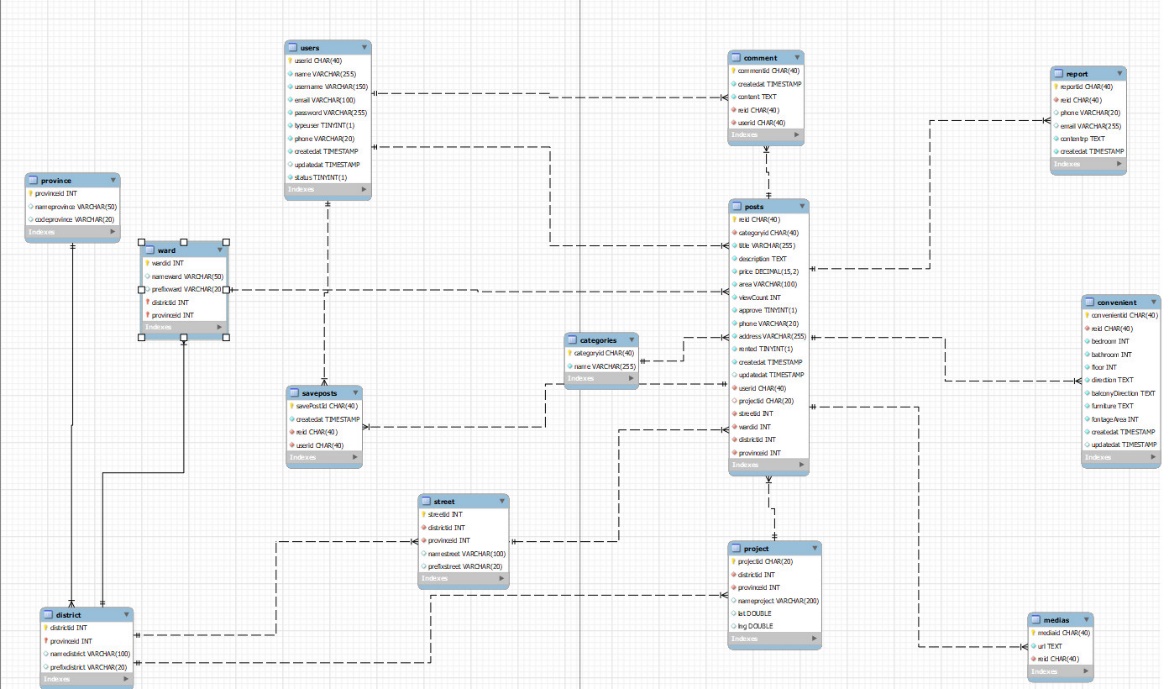
## Phân tích yêu cầu





## Phân tích dữ liệu

### Mô hình thực thể kết hợp



### Mô hình quan hệ

PROVINCE (**PROVINCEID**, nameprovince, codeprovince)

WARD (**WARDID,** nameward, prefixward, districtid, provinceid)

DISTRICT (**DISTRICTID**, provinceid, nameprovince, prefixdistrict)

USER (**USERID**, name, username, password, email, typeuser, phone, createdat, updatedat, status)

SAVEPOSTS (**SAVEPOSTID**, createdat, reid, userid)

STREET (**STREETID**, districtid, provinceid, namestreet, namestreet, prefixstreet)

CATEGORIES (**CATEGORYID**, name)

COMMENT (**COMMENTID**, createdat, content, reid, userid)

PROJECT (**PROJECTID**, districtid, provinceid, nameproject, lat, lng)

POSTS (**POSTID**, categoryid, title, description, price, area, viewcount, approve, phone, address, rented, createdat, updatedat, userid, projected, streetid, wardid, districtid, provinceid)

REPORT (**REPORTID**, reid, phone, email, contentrp, createdat)

MEDIAS (**MEDIAID**, url, reid)

CONVINIENT(**CONVIENTID**,reid,bedroom,bathroom,floor,direction,balconyDirection, furniture, fontageArea, createdat, updatedat)

#### PROVINCE

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc / Miền giá trị** |
| **PROVINCEID** | INT | PRIMARY KEY |
| nameprovince | VARCHAR(50) | NOT NULL |
| codeprovince | VARCHAR(20) | NOT NULL |

#### WARD

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc / Miền giá trị** |
| **WARDID** | INT | PRIMARY KEY |
| nameward | VARCHAR(50) | NOT NULL |
| prefixward | VARCHAR(20) | NOT NULL |
| districtid | INT | FOREIGN KEY |
| provinceid | INT | FOREIGN KEY |

#### DISTRICT

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc / Miền giá trị** |
| **DISTRICTID** | INT | PRIMARY KEY |
| provinceid | INT | FOREIGN KEY |
| nameprovince | VARCHAR(100) | NOT NULL |
| prefixdistrict | VARCHAR(20) | NOT NULL |

#### USER

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc / Miền giá trị** |
| **USERID** | CHAR(40) | PRIMARY KEY |
| name | VARCHAR(255) | NOTNULL |
| username | VARCHAR(150) | NOT NULL, UNIQUE |
| password | VARCHAR(150) | NOT NULL |
| email | VARCHAR(100) | NOT NULL, UNIQUE |
| typeuser | TINYINT(1) |  |
| phone | VARCHAR(20) | NOT NULL, UNIQUE |
| createdat | TIMESTAMP |  |
| createdat | TIMESTAMP |  |
| status | TINYINT(1) |  |

#### SAVEPOSTS

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc / Miền giá trị** |
| **SAVEPOSTID** | CHAR(40) | PRIMARY KEY |
| createdat | TIMESTAMP |  |
| reid | CHAR(40) |  |
| userid | CHAR(40) |  |

#### STREET

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc / Miền giá trị** |
| **STREETID** | INT | PRIMARY KEY |
| districtid | INT | FOREIGN KEY |
| provinceid | INT | FOREIGN KEY |
| namestreet | VARCHAR(100) | NOT NULL |
| prefixstreet | VARCHAR(20) | NOT NULL |

#### CATEGORIES

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc / Miền giá trị** |
| **CATEGORYID** | CHAR(40) | PRIMARY KEY |
| name | VARCHAR(225) | NOT NULL |

#### COMMENT

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc / Miền giá trị** |
| **COMMENTID** | CHAR(40) | PRIMARY KEY |
| createdat | TIMESTAMP |  |
| content | TEXT |  |
| reid | CHAR(40) | FOREIGN KEY |
| userid | CHAR(40) | FOREIGN KEY |

#### PROJECT

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc / Miền giá trị** |
| **PROJECTID** | CHAR(20) | PRIMARY KEY |
| districtid | INT | FOREIGN KEY |
| provinceid | INT | FOREIGN KEY |
| nameproject | VARCHAR(200) | NOT NULL |
| lat | DOUBLE |  |
| lng | BOUBLE |  |

#### POSTS

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc / Miền giá trị** |
| **POSTID** | CHAR(40) | PRIMARY KEY |
| categoryid | CHAR(40) | FOREIGN KEY |
| title | VARCHAR(255) | NOT NULL |
| description | TEXT | NOT NULL |
| price | DECIMAL(15,2) | NOT NULL |
| area | VARCHAR(100) | NOT NULL |
| viewcount | INT | NOT NULL |
| approve | TINYINT(1) | NOT NULL |
| phone | VARCHAR(20) | NOT NULL |
| address | VARCHAR(255) | NOT NULL |
| rented | TINYINT(1) | NOT NULL |
| updatedat | TIMESTAMP |  |
| createdat | TIMESTAMP |  |
| userid | CHAR(40) | FOREIGN KEY |
| projected | CHAR(20) |  |
| streetid | INT | NOT NULL |
| wardid | INT | FOREIGN KEY |
| districtid | INT | FOREIGN KEY |
| provinceid | INT | FOREIGN KEY |

#### REPORT

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc / Miền giá trị** |
| **REPORTID** | CHAR(40) | PRIMARY KEY |
| reid | CHAR(40) | FOREIGN KEY |
| phone | VARCHAR(20) | NOT NULL, UNIQUE |
| email | VARCHAR(255) | NOT NULL, UNIQUE |
| contentrp | TEXT |  |
| createdat | TIMESTAMP |  |

#### MEDIAS

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc / Miền giá trị** |
| **MEDIAID** | CHAR(40) | PRIMARY KEY |
| url | TEXT |  |
| reid | CHAR(40) |  |

#### CONVINIENT

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc / Miền giá trị** |
| **CONVIENTID** | CHAR(40) | PRIMARY KEY |
| reid | CHAR(40) |  |
| bedroom | INT |  |
| bathroom | INT |  |
| floor | INT |  |
| direction | TEXT |  |
| balconyDirection | TEXT |  |
| furniture | TEXT |  |
| fontageArea | INT |  |
| createdat | TIMESTAMP |  |
| updatedat | TIMESTAMP |  |

### Chuẩn hóa mô hình quan hệ

Chú thích : Trong đoạn chuẩn hóa này chúng ta sẽ không đem createdAt, updatedAt vì nó đều là trường dữ liệu bắt buộc trong mỗi bảng

#### USERS

Tập phụ thuộc hàm F = { Userid 🡪 Username, Password, Name,Email,Password  
 , TypeUser, Phone, Status;

Username 🡪 Userid}

🡺 Bảng đạt chuẩn **3NF** (Các thuộc tính không khóa phụ thuộc hoàn toàn vào thuộc tính khóa và không phụ thuộc bắc cầu vào thuộc tính không khóa khác; Thuộc tính khóa phụ thuộc vào thuộc tính không khóa)

#### POSTS

Tập phụ thuộc hàm F = {Reid 🡪 Categoryid, Title, Description, Price, Area, Viewcount,   
 Approve, Phone, Address, Rented, Userid, Projectid,   
 Streetid, Wardid, Districtid, Provinceid }

🡺 Bảng đạt chuẩn **BCNF** (Các thuộc tính không khóa phụ thuộc hoàn toàn vào thuộc tính khóa và không phụ thuộc bắc cầu vào thuộc tính không khóa khác; Thuộc tính khóa không phụ thuộc vào thuộc tính không khóa)

#### MEDIAS

Tập phụ thuộc hàm F = { Mediaid 🡪 Url, Reid}

🡺 Bảng đạt chuẩn **BCNF** (Các thuộc tính không khóa phụ thuộc hoàn toàn vào thuộc tính khóa và không phụ thuộc bắc cầu vào thuộc tính không khóa khác; Thuộc tính khóa không phụ thuộc vào thuộc tính không khóa)

#### CATEGORIES

Tập phụ thuộc hàm F = { Categoryid 🡪 Name }

🡺 Bảng đạt chuẩn **BCNF** (Các thuộc tính không khóa phụ thuộc hoàn toàn vào thuộc tính khóa và không phụ thuộc bắc cầu vào thuộc tính không khóa khác; Thuộc tính khóa không phụ thuộc vào thuộc tính không khóa)

#### CONVENIENT

Tập phụ thuộc hàm F = { Convenientid, Reid 🡪 Bedroom, Bathroom, Floor, Direction,

Balconydirection, Funiture, FontageArea }

🡺 Bảng đạt chuẩn **BCNF** (Các thuộc tính không khóa phụ thuộc hoàn toàn vào thuộc tính khóa và không phụ thuộc bắc cầu vào thuộc tính không khóa khác; Thuộc tính khóa không phụ thuộc vào thuộc tính không khóa)

#### REPORT

Tập phụ thuộc hàm F = { Reportid 🡪 Reid, Phone, Email, Contentrp}

🡺 Bảng đạt chuẩn **BCNF** (Các thuộc tính không khóa phụ thuộc hoàn toàn vào thuộc tính khóa và không phụ thuộc bắc cầu vào thuộc tính không khóa khác; Thuộc tính khóa không phụ thuộc vào thuộc tính không khóa)

#### SAVEPOSTS

Tập phụ thuộc hàm F = {Savepostid 🡪 Reid, Userid}

🡺 Bảng đạt chuẩn **BCNF** (Các thuộc tính không khóa phụ thuộc hoàn toàn vào thuộc tính khóa và không phụ thuộc bắc cầu vào thuộc tính không khóa khác; Thuộc tính khóa không phụ thuộc vào thuộc tính không khóa)

#### COMMENT

Tập phụ thuộc hàm F = { Commentid 🡪 Content, Reid, Userid }

🡺 Bảng đạt chuẩn **BCNF** (Các thuộc tính không khóa phụ thuộc hoàn toàn vào thuộc tính khóa và không phụ thuộc bắc cầu vào thuộc tính không khóa khác; Thuộc tính khóa không phụ thuộc vào thuộc tính không khóa)

#### PROVINCE

Tập phụ thuộc hàm F = { Provinceid 🡪 Nameprovince, Codeprovince }

🡺 Bảng đạt chuẩn **BCNF** (Các thuộc tính không khóa phụ thuộc hoàn toàn vào thuộc tính khóa và không phụ thuộc bắc cầu vào thuộc tính không khóa khác; Thuộc tính khóa không phụ thuộc vào thuộc tính không khóa)

#### DISTRICT

Tập phụ thuộc hàm F = { Districtid, Provinceid 🡪 Namedistrict, Prefixdistrict }

🡺Bảng đạt chuẩn **BCNF** (Các thuộc tính không khóa phụ thuộc hoàn toàn vào thuộc tính khóa và không phụ thuộc bắc cầu vào thuộc tính không khóa khác; Thuộc tính khóa không phụ thuộc vào thuộc tính không khóa)

#### WARD

Tập phụ thuộc hàm F = { Wardid, Districtid, Provinceid 🡪 Nameward, Prefixward }

🡺Bảng đạt chuẩn **BCNF** (Các thuộc tính không khóa phụ thuộc hoàn toàn vào thuộc tính khóa và không phụ thuộc bắc cầu vào thuộc tính không khóa khác; Thuộc tính khóa không phụ thuộc vào thuộc tính không khóa)

#### STREET

Tập phụ thuộc hàm F = { Streetid, Districtid, Provinceid 🡪 Namestreet, Prefixstreet }

🡺Bảng đạt chuẩn **BCNF** (Các thuộc tính không khóa phụ thuộc hoàn toàn vào thuộc tính khóa và không phụ thuộc bắc cầu vào thuộc tính không khóa khác; Thuộc tính khóa không phụ thuộc vào thuộc tính không khóa)

#### PROJECT

Tập phụ thuộc hàm F = { Projectid, Districtid, Provinceid 🡪 Nameproject, Prefixproject }

🡺Bảng đạt chuẩn **BCNF** (Các thuộc tính không khóa phụ thuộc hoàn toàn vào thuộc tính khóa và không phụ thuộc bắc cầu vào thuộc tính không khóa khác; Thuộc tính khóa không phụ thuộc vào thuộc tính không khóa)

## Ràng buộc trọn vẹn

### Username không được trùng

* Bối cảnh: USERS
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **USERS** | + | - | + (Username) |

### Password không được để trống

* Bối cảnh: USERS
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **USERS** | + | - | + (Password) |

### Typeuser chỉ có thể là 0 hoặc 1

* Bối cảnh: USERS
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **USERS** | +(typeuser) | - | + (typeuser) |

### Status (thể hiện trạng thái chặn) chỉ có thể là 0 hoặc 1

* Bối cảnh: USERS
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **USERS** | +(status) | - | + (status) |

### Typeuser chỉ có thể là 0 hoặc 1

* Bối cảnh: USERS
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **USERS** | +(typeuser) | - | + (typeuser) |

### Thông tin địa chỉ không được sai

* Bối cảnh: POSTS
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **POSTS** | +(provinceid,wardid,districtid,projectid) | + | - |

### Số lượng bài đăng trong tháng không được quá 5 bài

* Bối cảnh: POSTS
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **POSTS** | +(reid) | + | - |

### Giá cho thuê nhà đất trên 100.000 VND

* Bối cảnh: POSTS
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **POSTS** | +(price) | + | - |

### Định dạng số điện thoại đúng

* Bối cảnh: POSTS
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **POSTS** | +(phone) | + | - |

### Lưu 1 bài đăng không được yêu thích quá 2 lần

* Bối cảnh: SAVEPOSTS
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| SAVEPOSTS | +(reid,userid) | - | +(reid,userid) |

### Khi lưu bài sẽ tăng lượt yêu thích bài post

* Bối cảnh: SAVEPOSTS, POSTS
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| SAVEPOSTS | + (reid,userid) | -(reid,userid) | - |
| POSTS | + | + | +(viewcount) |

### Khi bỏ lưu bài sẽ giảm lượt yêu thích bài post

* Bối cảnh: SAVEPOSTS, POST
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| SAVEPOSTS | + (reid,userid) | -(reid,userid) | - |
| POSTS | + | + | +(viewcount) |

### Bình luận không được bình luận hai lần trong cùng một bài viết

* Bối cảnh: COMMENT
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **COMMENT** | + (commentid,reid) | - | +  (commentid,reid) |

### Không được đăng bài viết giống nhau nếu muốn đăng thì phải đổi thuộc tính userid thành NULL.

* Bối cảnh: POSTS, USER
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **USER** | - | + | - (\*) |
| **POSTS** | - | - | + (userid) |

### Ngày đăng bài phải nhỏ hơn ngày nhà được cập nhật lại thành được thuê

* Bối cảnh: POSTS
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **POSTS** | + | - | + (createdat, updatedat) |

### Khi tố cáo bài đăng rồi không được tố cáo lần nữa dựa trên email

* Bối cảnh: REPORT
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **REPORT** | +(email) | - | + (email) |

### Khi tố cáo bài đăng rồi không được tố cáo lần nữa dựa trên số điện thoại

* Bối cảnh: REPORT
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **REPORT** | +(phone) | - | + (phone) |

### Thêm hình ảnh, videos của bài đăng không quá 5 tấm ảnh

* Bối cảnh: MEDIAS
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **MEDIAS** | +(reid,mediasid) | - | - |

# THIẾT KẾ XỬ LÝ TRONG CLIENT/ SERVER

## Các giao tác

### Thêm tài khoản

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server tạo tài khoản và truyền tham số |  |
|  | **Tạo tài khoản**   * Tạo ra mã khóa cho user dựa trên mã hóa uuid * Kiểm tra username có tồn tại * Tạo function mã hóa mật khẩu * Khai báo biến và gọi function mã hóa mật khẩu gán giá trị cho biến đã vừa khai báo * Thực thi chèn thông tin dữ liệu vào bảng USERS * Thêm user vào Role thích hợp * Cấp TSK\_Client (tài khoản khách) quyền impersonate user |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 38 Các bước xử lý giao tác Thêm tài khoản

### Đăng nhập

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server xác thực user, trả về dữ liệu, truyền tham số |  |
|  | **Đăng nhập**   * Kiểm tra username có tồn tại * Kiểm tra mật khẩu * Đọc dữ liệu từ bảng USERS |
| Nhận kết quả giao tác lưu vào trong session |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 39 Các bước xử lý giao tác Đăng nhập

### Tố cáo

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server lưu dữ liệu tố cáo |  |
|  | **Tố cáo**   * Tạo function kiểm tra dữ liệu name, phone * Kiểm tra độ dài content * Kiểm tra người dùng đã gửi tố cáo qua chưa * Thêm dữ liệu vào bảng REPORT |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 40 Tố cáo

### Cập nhật trạng thái đã thuê cho bài đằng

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server cập nhật trạng thái bài đăng |  |
|  | **Tố cáo**   * Kiểm tra bài post có bị khóa * Kiểm tra trạng thái đã được cập nhật trước đó chưa * Kiểm tra có phải chủ nhân bài đăng * Kiểm tra bài đăng có tồn tại * Cập nhật dữ liệu trong bảng POSTS |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 41 Tố cáo

### Xóa tài khoản

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server xóa tài khoản và truyền tham số |  |
|  | **Xóa tài khoản**   * Xóa tài khoản khỏi bảng TAIKHOAN * Hủy quyền impersonate user được xóa của TSK\_Client (tài khoản khách) * Xóa user trong database |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 39 Các bước xử lý giao tác Xóa tài khoản

### Cập nhật mật khẩu của tài khoản

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server cập nhật mật khẩu của tài khoản và truyền tham số |  |
|  | **Cập nhật mật khẩu của tài khoản**   * Cập nhật mật khẩu của tài khoản trong TAIKHOAN |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 40 Các bước xử lý giao tác Cập nhật mật khẩu

### Thêm giáo viên

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server thêm giáo viên và truyền tham số |  |
|  | **Thêm giáo viên**   * Đọc GIAOVIEN để tạo MaGV * Thêm giáo viên vào GIAOVIEN |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 41 Các bước xử lý giao tác Thêm giáo viên

### Xóa giáo viên

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server xóa giáo viên và truyền tham số |  |
|  | **Xóa giáo viên**   * Xóa giáo viên khỏi GIAOVIEN |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 42 Các bước xử lý giao tác Xóa giáo viên

### Cập nhật giáo viên

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server cập nhật giáo viên và truyền tham số |  |
|  | **Cập nhật giáo viên**   * Cập nhật giáo viên trong GIAOVIEN |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 43 Các bước xử lý giao tác Cập nhật giáo viên

### Thêm ca học

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server thêm ca học và truyền tham số |  |
|  | **Thêm ca học**   * Đọc CAHOC để tạo MaCH * Thêm ca học vào CAHOC |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 44 Các bước xử lý giao tác Thêm ca học

### Xóa ca học

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server xóa ca học và truyền tham số |  |
|  | **Xóa ca học**   * Xóa ca học khỏi CAHOC |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 45 Các bước xử lý giao tác Xóa ca học

### Cập nhật ca học

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server cập nhật ca học và truyền tham số |  |
|  | **Cập nhật ca học**   * Cập nhật ca học trong CAHOC |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 46 Các bước xử lý giao tác Cập nhật ca học

### Thêm chương trình học

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server thêm chương trình học và truyền tham số |  |
|  | **Thêm chương trình học**   * Thêm chương trình học vào CHUONGTRINH |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 47 Các bước xử lý giao tác Thêm chương trình học

### Cập nhật chương trình học

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server cập nhật chương trình học và truyền tham số |  |
|  | **Cập nhật chương trình học**   * Cập nhật chương trình học trong CHUONGTRINH |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 48 Các bước xử lý giao tác Cập nhật chương trình học

### Mở lớp

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server thêm lớp học và truyền tham số |  |
|  | **Thêm lớp**   * Đọc bảng LOP để tạo MaLop * Thêm lớp vào LOP |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 49 Các bước xử lý giao tác Mở lớp

### Hủy lớp

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server hủy lớp và truyền tham số |  |
|  | **Hủy lớp**   * Kiểm tra lớp đã khai giảng hay chưa * Kiểm tra có học viên nào đóng học phí cho lớp hay chưa * Hủy mọi đăng ký cho lớp được xóa khỏi DANGKY * Xóa lớp khỏi LOP |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 50 Các bước xử lý giao tác Hủy lớp

### Bố trí giáo viên cho lớp

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server bố trí giáo viên được chọn cho lớp và truyền tham số |  |
|  | **Bố trí giáo viên cho lớp**   * Kiểm tra mã lớp * Kiểm tra mã giáo viên * Cập nhật MaGV của lớp trong LOP |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 51 Các bước xử lý giao tác Bố trí giáo viên

### Đăng ký

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server lưu thông tin đăng ký và truyền tham số |  |
|  | **Đăng ký**   * Kiểm tra mã lớp * Kiểm tra lớp đã khai giảng chưa * Cập nhật MaGV của lớp trong LOP |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 52 Các bước xử lý giao tác Đăng ký

### Hủy đăng ký

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server hủy thông tin đăng ký và truyền tham số |  |
|  | **Hủy đăng ký**   * Xóa đăng ký khỏi DANGKY * Giảm sĩ số của lớp được đăng ký xuống 1 |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 53 Các bước xử lỷ giao tác Hủy đăng ký

### Thêm học viên

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server thêm học viên và truyền tham số |  |
|  | **Thêm học viên**   * Đọc HOCVIEN để tạo MaHV * Thêm học viên vào HOCVIEN |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 54 Các bước xử lý giao tác Thêm học viên

### Thu học phí (tạo hóa đơn)

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server kiểm tra mã học viên |  |
|  | **Kiểm tra mã học viên**  Tìm học viên trong HOCVIEN bằng MaHV |
| Nhận kết quả kiểm tra |  |
| Hiện thông tin học viên |  |
| * Tách chuỗi mã lớp thành các mã lớp * Lần lượt yêu cầu server kiểm tra mã lớp |  |
|  | **Kiểm tra mã lớp**  Tìm lớp trong LOP bằng mã lớp |
| Nhận kết quả kiểm tra |  |
| Hiện thông tin các lớp được nhập và tổng số tiền cần đóng |  |
| * Tách chuỗi mã đăng ký thành các mã đăng ký * Lần lượt yêu cầu server kiểm tra mã đăng ký |  |
|  | **Kiểm tra mã đăng ký**  Tìm đăng ký trong DANGKY bằng mã đăng ký |
| Nhận kết quả kiểm tra |  |
| Hiện thông tin các lớp được đăng ký và tổng số tiền cần đóng |  |
| Yêu cầu server tạo hóa đơn và truyền tham số |  |
|  | **Thu học phí**   * Kiểm tra mã học viên * Đọc HOADON để tạo MaHD * Thêm hóa đơn vào HOADON * Tách chuỗi mã đăng ký thành các mã đăng ký * Lần lượt kiểm tra mã đăng ký và thêm học viên vào các lớp đã được đăng ký, tăng tổng tiền của hóa đơn tương ứng * Lần lượt hủy đăng ký tương ứng * Tách chuỗi mã lớp thành các mã lớp * Lần lượt kiểm tra mã lớp và thêm học viên vào các lớp tương ứng, tăng tổng tiền của hóa đơn tương ứng |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 55 Các bước xử lý giao tác Thu học phí

### Chuyển lớp

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server kiểm tra mã học viên |  |
|  | **Kiểm tra mã học viên**  Tìm học viên trong HOCVIEN bằng MaHV |
| Nhận kết quả kiểm tra và hiện thông tin học viên |  |
| Yêu cầu server kiểm tra mã lớp cũ |  |
|  | **Kiểm tra mã lớp cũ**  Tìm lớp trong LOP bằng mã lớp |
| Nhận kết quả kiểm tra và hiện thông tin lớp cũ |  |
| Yêu cầu server kiểm tra mã lớp mới |  |
|  | **Kiểm tra mã lớp mới**  Tìm lớp trong LOP bằng mã lớp |
| Nhận kết quả kiểm tra và hiện thông tin lớp mới |  |
|  | **Chuyển lớp**   * Kiểm tra mã học viên * Kiểm tra mã lớp * Kiểm tra lớp mới và lớp cũ có khác nhau hay không * Kiểm tra lớp mới và lớp cũ có cùng chương trình không * Kiểm tra lớp mới đã kết thúc hay chưa * Kiểm tra học viên có trong danh sách của lớp cũ hay không * Kiểm tra học viên đã có trong danh sách của lớp mới hay không * Xóa bảo lưu của học viên cho lớp cũ (nếu có) trong BAOLUU * Xóa học viên khỏi danh sách học viên của lớp cũ trong HOC * Giảm sĩ số của lớp cũ xuống 1 trong LOP * Thêm học viên vào danh sách học viên của lớp mới trong HOC * Tăng sĩ số của lớp mới lên 1 trong LOP |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 56 Các bước xử lý giao tác Chuyển lớp

### Chấm điểm

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server cập nhật điểm của học viên và truyền tham số |  |
|  | **Chấm điểm**   * Kiểm tra điểm * Cập nhật các điểm của học viên trong HOC |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 57 Các bước xử lý giao tác Chấm điểm

### Bảo lưu

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server lưu thông tin bảo lưu của học viên và truyền tham số |  |
|  | **Bảo lưu**   * Kiểm tra mã học viên * Kiểm tra mã lớp * Kiểm tra lớp đã kết thúc hay chưa (chỉ bảo lưu cho lớp chưa kết thúc) * Kiểm tra học viên có đang học lớp được xin bảo lưu hay không * Kiểm tra học viên đã bảo lưu cho lớp hay chưa * Thêm thông tin bảo lưu vào BAOLUU |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 58 Các bước xử lý giao tác Bảo lưu

### Hủy bảo lưu

|  |  |
| --- | --- |
| **Client** | **Server** |
| Yêu cầu server hủy thông tin bảo lưu của học viên và truyền tham số |  |
|  | **Hủy bảo lưu**   * Xóa thông tin bảo lưu khỏi BAOLUU |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

Bảng 59 Các bước xử lý giao tác Hủy bảo lưu

## Phân quyền

### Các đối tượng phân quyền

* Quản lý: các nhân viên quản lý của trung tâm.
* Tư vấn: các nhân viên tư vấn của trung tâm.
* Giáo viên: các giáo viên của trung tâm.

### Phân quyền trên các bảng

*Ghi chú:* S – Select, I – Insert, U – Update, D – Delete

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Quản lý** | **Tư vấn** | **Giáo viên** |
| **TAIKHOAN** | S, I, U, D |  |  |
| **GIAOVIEN** | S, I, U, D | S | S |
| **CHUONGTRINH** | S, I, U, D | S | S |
| **CAHOC** | S, I, U, D | S | S |
| **LOP** | S, I, U, D | S, U | S |
| **DANGKY** | S | S, I, U, D | S |
| **HOCVIEN** | S | S, I, U, D | S |
| **HOADON** | S | S, I, U, D | S |
| **HOC** | S | S, I, U, D | S, U |
| **BAOLUU** | S | S, I, U, D | S |

Bảng 60 Phân quyền trên các bảng

### Phân quyền trên các giao tác

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Quản lý** | **Khách hàng vãng lai** | **Khách hàng than thiêt** |
| **Đăng ký tài khoản** | X | X | X |
| **Chỉnh sửa tài khoản** | X | X |  |
| **Xóa tài khoản** | X |  |  |
| **Phân quyền tài khoản** | X |  |  |
| **Đăng bài viết** | X | X |  |
| **Chỉnh sửa bài viết** | X |  |  |
| **Lưu bài viết** | X | X |  |
| **Đánh giá bài viết** | X | X | X |
| **Liên hệ người đăng** | X | X | X |
| **Báo cáo tin xấu** | X | X | X |
| **Tìm kiếm thông tin** | X | X | X |
| **Duyệt bài viết** | X |  |  |

## Backup

* Full database backup: Mỗi chủ nhật hàng tuần vào lúc 23 giờ.
* Differenitial backup: Mỗi thứ 3, thứ 5, thứ 7 hàng tuần vào lúc 12 giờ
* Logs backups: Mỗi giờ vào phút 5, 15, 25, 35, 45, 55.

# TRIỂN KHAI THỰC NGHIỆM

## Phía Server

### Ràng buộc toàn vẹn

#### Username không được trùng

CREATE TRIGGER trg\_R1

ON TAIKHOAN

FOR INSERT, UPDATE

AS

BEGIN

DECLARE @username VARCHAR(10)

SELECT @username = Username FROM inserted

IF (SELECT COUNT(\*) FROM TAIKHOAN WHERE Username = @username) > 1

BEGIN

RAISERROR('Trùng username', 16, 1)

ROLLBACK

END

END

GO

#### Password không được để trống

CREATE TRIGGER trg\_R2

ON TAIKHOAN

FOR INSERT, UPDATE

AS

BEGIN

IF EXISTS(SELECT \* FROM inserted WHERE Password IS NULL OR Password = '')

BEGIN

RAISERROR('Mật khẩu không được để trống', 16, 1)

ROLLBACK

END

END

GO

#### Loại tài khoản chỉ được có giá trị “A”, “B”, “C”

CREATE TRIGGER trg\_R3

ON TAIKHOAN

FOR INSERT, UPDATE

AS

BEGIN

IF EXISTS(SELECT \* FROM inserted WHERE LoaiTK != 'A' AND LoaiTK != 'B' AND LoaiTK != 'C')

BEGIN

RAISERROR('Loại tài khoản không hợp lệ', 16, 1)

ROLLBACK

END

END

GO

#### Không được xóa giáo viên khi giáo viên có lớp chưa kết thúc. Nếu được xóa, đổi mã giáo viên của các lớp đã được giáo viên dạy thành NULL

CREATE TRIGGER trg\_R4

ON GIAOVIEN

INSTEAD OF DELETE

AS

BEGIN

DECLARE @magv smallint

SELECT @magv = MaGV FROM deleted

IF EXISTS(SELECT \* FROM LOP WHERE NgayKetThuc >= GETDATE() AND MaGV = @magv)

BEGIN

RAISERROR('Giáo viên có lớp chưa kết thúc', 16, 1)

RETURN

END

UPDATE LOP SET MaGV = NULL WHERE MaGV = @magv

DELETE FROM GIAOVIEN WHERE MaGV = @magv

END

GO

#### Ngày khai giảng phải nhỏ hơn ngày kết thúc

CREATE TRIGGER trg\_R5

ON LOP

FOR INSERT, UPDATE

AS

BEGIN

IF EXISTS(SELECT \* FROM inserted WHERE NgayKhaiGiang < NgayKetThuc)

BEGIN

RAISERROR('Ngày khai giảng không được nhó hơn ngày kết thúc', 16, 1)

ROLLBACK

END

END

GO

#### Sĩ số học viên của lớp học không được lớn hơn giới hạn lớp học

CREATE TRIGGER trg\_R6

ON LOP

FOR INSERT, UPDATE

AS

BEGIN

IF EXISTS(SELECT \* FROM inserted WHERE SiSo > GioiHan)

BEGIN

RAISERROR('Lớp hết chỗ', 16, 1)

ROLLBACK

END

END

GO

#### Học phí chương trình phải không nhỏ hơn 0

CREATE TRIGGER trg\_R7

ON CHUONGTRINH

FOR INSERT, UPDATE

AS

BEGIN

IF EXISTS(SELECT \* FROM inserted WHERE HocPhiCT < 0)

BEGIN

RAISERROR('Học phí không hợp lệ', 16, 1)

ROLLBACK

END

END

GO

#### Tên chương trình không được trùng

CREATE TRIGGER trg\_R8

ON CHUONGTRINH

FOR INSERT, UPDATE

AS

BEGIN

DECLARE @tenct NVARCHAR(40)

SELECT @tenct = TenCT FROM inserted

IF (SELECT COUNT(\*) FROM CHUONGTRINH WHERE TenCT = @tenct) > 1

BEGIN

RAISERROR('Tên chương trình không được trùng', 16, 1)

ROLLBACK

END

END

GO

#### Không được xóa ca học khi có lớp học trong ca học chưa kết thúc. Nếu được xóa, đổi mã ca học của các lớp có mã ca học thành NULL

CREATE TRIGGER trg\_R9

ON CAHOC

INSTEAD OF DELETE

AS

BEGIN

DECLARE @mach smallint

SELECT @mach = MaCH FROM deleted

IF EXISTS(SELECT \* FROM LOP WHERE NgayKetThuc >= GETDATE() AND MaCH = @mach)

BEGIN

RAISERROR('Ca học có lớp chưa kết thúc', 16, 1)

RETURN

END

UPDATE LOP SET MaCH = NULL WHERE MaCH = @mach

DELETE FROM CAHOC WHERE MaCH = @mach

END

GO

#### Giáo viên không cùng lúc dạy các lớp cùng ca học

CREATE TRIGGER trg\_R10

ON LOP

FOR INSERT, UPDATE

AS

BEGIN

DECLARE @magv smallint, @mach smallint, @ngaykhaigiang smalldatetime, @ngayketthuc smalldatetime

SELECT @magv = MaGV, @mach = MaCH, @ngaykhaigiang = NgayKhaiGiang, @ngayketthuc = NgayKetThuc FROM inserted

IF (SELECT COUNT(\*) FROM LOP WHERE MaGV=@magv AND MaCH=@mach AND NgayKhaiGiang <= @ngayketthuc AND NgayKetThuc >= @ngaykhaigiang) > 1

BEGIN

RAISERROR('Giáo viên trùng ca học', 16, 1)

ROLLBACK

END

END

GO

#### Một người không đăng ký một lớp 2 lần (không trùng MaLop, SDT, TenDK)

CREATE TRIGGER trg\_R11

ON DANGKY

FOR INSERT, UPDATE

AS

BEGIN

DECLARE @malop char(7), @sdt nvarchar(12), @tendk nvarchar(40)

SELECT @malop = MaLop, @sdt = SDT, @tendk = TenDK FROM inserted

IF (SELECT COUNT(\*) FROM DANGKY WHERE MaLop = @malop AND SDT = @sdt AND TenDK = @tendk) > 1

BEGIN

RAISERROR('Học viên đã đăng ký lớp', 16, 1)

ROLLBACK

END

END

GO

#### Học viên không đóng học phí cho một lớp 2 lần

CREATE TRIGGER trg\_R12

ON HOC

FOR INSERT, UPDATE

AS

BEGIN

DECLARE @malop char(7), @mahd int, @mahv int

SELECT @malop = MaLop, @mahd = MaHD FROM inserted

SELECT @mahv = MaHV FROM HOADON

IF (SELECT COUNT(\*) FROM HOC WHERE MaHD IN (SELECT MaHD FROM HOADON WHERE MaHV=@mahv)) > 1

BEGIN

RAISERROR('Học viên trong danh sách của lớp này', 16, 1)

ROLLBACK

END

END

GO

#### Điểm chuyên cần và điểm cuối kì từ 0 đến 10

CREATE TRIGGER trg\_13

ON HOC

FOR INSERT, UPDATE

AS

BEGIN

IF EXISTS(SELECT \* FROM inserted WHERE DiemChuyenCan < 0 OR DiemChuyenCan > 10 OR DiemCuoiKy < 0 OR DiemCuoiKy > 0)

BEGIN

RAISERROR('Điểm không hợp lệ', 16, 1)

ROLLBACK

END

END

GO

### Các chức năng

#### Đăng nhập

CREATE PROC sp\_DangNhap

@username varchar(10),

@password nvarchar(20)

AS

BEGIN

SELECT \* FROM TAIKHOAN WHERE Username=@username AND Password=@password

END

GO

#### Xem danh sách tài khoản

CREATE PROC sp\_LietKeTaiKhoan

AS

BEGIN

SELECT \* FROM TAIKHOAN

END

GO

#### Tạo tài khoản

CREATE PROC sp\_TaoTaiKhoan

@username varchar(10),

@password nvarchar(20),

@loai char(1)

AS

BEGIN

DECLARE @matk smallint

SET @matk = 1

BEGIN TRY

BEGIN TRAN

SELECT \* FROM TAIKHOAN with (updlock)

WHILE EXISTS( SELECT \* FROM TAIKHOAN WHERE MaTK=@matk)

SET @matk = @matk + 1

INSERT INTO TAIKHOAN(MaTK, Username, Password, LoaiTK) VALUES(@matk, @username, @password, @loai)

EXEC('CREATE USER [' + @username + '] WITHOUT LOGIN')

IF @loai='A'

BEGIN

EXEC('sp\_AddRoleMember TSK\_QuanLy, [' + @username + ']')

END

ELSE IF @loai = 'B'

BEGIN

EXEC('sp\_AddRoleMember TSK\_TuVan, [' + @username + ']')

END

ELSE

BEGIN

EXEC('sp\_AddRoleMember TSK\_GiaoVien, [' + @username + ']')

END

EXEC('GRANT IMPERSONATE ON USER::[' + @username +'] TO Client')

COMMIT

END TRY

BEGIN CATCH

ROLLBACK

DECLARE @msg nvarchar(200)

SET @msg = ERROR\_MESSAGE()

RAISERROR(@msg, 16, 1)

END CATCH

END

GO

#### Xóa tài khoản

CREATE PROC sp\_XoaTaiKhoan

@matk smallint

AS

BEGIN

DECLARE @username nvarchar(20)

SELECT @username = Username FROM TAIKHOAN WHERE MaTK=@matk

BEGIN TRY

BEGIN TRAN

DELETE FROM TAIKHOAN WHERE MaTK=@matk

EXEC('REVOKE IMPERSONATE ON USER::[' + @username +'] FROM Client')

EXEC('DROP USER [' + @username + ']')

COMMIT

END TRY

BEGIN CATCH

ROLLBACK

DECLARE @msg nvarchar(200)

SET @msg = ERROR\_MESSAGE()

RAISERROR(@msg, 16, 1)

END CATCH

END

GO

#### Cập nhật mật khẩu của tài khoản

CREATE PROC sp\_ThayDoiMatKhau

@matk smallint,

@password nvarchar(20)

AS

BEGIN

UPDATE TAIKHOAN SET Password=@password WHERE MaTK=@matk

END

GO

#### Xem danh sách giáo viên

CREATE PROC sp\_LietKeGiaoVien

AS

BEGIN

SELECT \* FROM GIAOVIEN

END

GO

#### Xem thông tin giáo viên

CREATE PROC sp\_XemGiaoVien

@magv smallint

AS

BEGIN

SELECT \* FROM GIAOVIEN WHERE MaGV=@magv

END

GO

#### Thêm giáo viên

CREATE PROC sp\_ThemGiaoVien

@ten nvarchar(40),

@sdt nvarchar(12)

AS

BEGIN

DECLARE @i smallint

SET @i = 1

BEGIN TRY

BEGIN TRAN

SELECT \* FROM GIAOVIEN WITH (updlock)

WHILE EXISTS(SELECT \* FROM GIAOVIEN WHERE MaGV=@i)

SET @i = @i + 1

INSERT INTO GIAOVIEN(MaGV, TenGV, SDTGV) VALUES(@i, @ten, @sdt)

COMMIT

END TRY

BEGIN CATCH

ROLLBACK

DECLARE @msg nvarchar(200)

SET @msg = ERROR\_MESSAGE()

RAISERROR(@msg, 16, 1)

END CATCH

END

GO

#### Xóa giáo viên

CREATE PROC sp\_XoaGiaoVien

@magv smallint

AS

BEGIN

DELETE FROM GIAOVIEN WHERE MaGV=@magv

END

GO

#### Cập nhật giáo viên

CREATE PROC sp\_CapNhatGiaoVien

@magv smallint,

@ten nvarchar(40),

@sdt nvarchar(12)

AS

BEGIN

UPDATE GIAOVIEN SET TenGV=@ten, SDTGV = @sdt WHERE MaGV=@magv

END

GO

#### Xem các lớp do giáo viên dạy

CREATE PROC sp\_XemLichSuDay

@magv smallint,

@hienthi\_moitrang smallint,

@trang smallint

AS

BEGIN

SELECT MaLop, MaCT, Lop.MaCH, CaHoc = GioHoc + ' ' + NgayHoc, GioiHan, SiSo, NgayKhaiGiang, NgayKetThuc

FROM LOP LEFT JOIN CAHOC ON LOP.MaCH = CAHOC.MaCH

WHERE MaGV=@magv

ORDER BY NgayKhaiGiang DESC

OFFSET @hienthi\_moitrang \* (@trang - 1) ROWS

FETCH NEXT @hienthi\_moitrang ROWS ONLY

END

GO

#### Xem danh sách ca học

CREATE PROC sp\_LietKeCaHoc

AS

BEGIN

SELECT \* FROM CAHOC

END

GO

#### Thêm ca học

CREATE PROC sp\_ThemCaHoc

@giohoc nvarchar(20),

@ngayhoc nvarchar(20)

AS

BEGIN

DECLARE @i smallint

SET @i = 1

BEGIN TRY

BEGIN TRAN

SELECT \* FROM CAHOC with(updlock)

WHILE EXISTS(SELECT \* FROM CAHOC WHERE MaCH=@i)

SET @i = @i + 1

INSERT INTO CAHOC(MaCH, GioHoc, NgayHoc) VALUES(@i, @giohoc, @ngayhoc)

COMMIT

END TRY

BEGIN CATCH

ROLLBACK

DECLARE @msg nvarchar(200)

SET @msg = ERROR\_MESSAGE()

RAISERROR(@msg, 16, 1)

END CATCH

END

GO

#### Xóa ca học

CREATE PROC sp\_XoaCaHoc

@mach smallint

AS

BEGIN

DELETE FROM CAHOC WHERE MaCH=@mach

END

GO

#### Cập nhật ca học

CREATE PROC sp\_CapNhatCaHoc

@mach smallint,

@giohoc nvarchar(20),

@ngayhoc nvarchar(20)

AS

BEGIN

UPDATE CAHOC SET GioHoc=@giohoc, NgayHoc=@ngayhoc WHERE MaCH=@mach

END

GO

#### Xem danh sách các chương trình học

CREATE PROC sp\_LietKeChuongTrinh

AS

BEGIN

SELECT \* FROM CHUONGTRINH

END

GO

#### Thêm chương trình học

CREATE PROC sp\_ThemChuongTrinh

@mact char(3),

@tenct nvarchar(30),

@thoigian nvarchar(10),

@hocphi float

AS

BEGIN

INSERT INTO CHUONGTRINH(MaCT, TenCT, ThoiGianHoc, HocPhiCT) VALUES(@mact, @tenct, @thoigian, @hocphi)

END

GO

#### Cập nhật chương trình học

CREATE PROC sp\_CapNhatChuongTrinh

@mact char(3),

@tenct nvarchar(30),

@thoigian nvarchar(10),

@hocphi float

AS

BEGIN

UPDATE CHUONGTRINH SET TenCT=@tenct, ThoiGianHoc=@thoigian, HocPhiCT=@hocphi WHERE MaCT=@mact

END

GO

#### Xem danh sách các lớp (lọc lớp học)

CREATE PROC sp\_TimLop

@mact char(3),

@tinhtrang smallint,

@hienthi\_moitrang smallint,

@trang smallint

AS

BEGIN

IF @mact IS NULL OR @mact = ''

BEGIN

IF @tinhtrang = 1

SELECT MaLop, LOP.MaCT, TenCT, ThoiGianHoc, HocPhiCT, LOP.MaCH, CaHoc = CAHOC.GioHoc + ' ' + CAHOC.NgayHoc, LOP.MaGV, TenGV, GioiHan, SiSo, NgayKhaiGiang, NgayKetThuc

FROM LOP LEFT JOIN CHUONGTRINH ON LOP.MaCT=CHUONGTRINH.MaCT LEFT JOIN GIAOVIEN ON LOP.MaGV=GIAOVIEN.MaGV LEFT JOIN CAHOC ON LOP.MaCH = CAHOC.MaCH

WHERE NgayKhaiGiang <= GETDATE() AND NgayKetThuc >= GETDATE()

ORDER BY NgayKhaiGiang DESC

OFFSET @hienthi\_moitrang \* (@trang - 1) ROWS

FETCH NEXT @hienthi\_moitrang ROWS ONLY

ELSE IF @tinhtrang = 2

SELECT MaLop, LOP.MaCT, TenCT, ThoiGianHoc, HocPhiCT, LOP.MaCH, CaHoc = CAHOC.GioHoc + ' ' + CAHOC.NgayHoc, LOP.MaGV, TenGV, GioiHan, SiSo, NgayKhaiGiang, NgayKetThuc

FROM LOP LEFT JOIN CHUONGTRINH ON LOP.MaCT=CHUONGTRINH.MaCT LEFT JOIN GIAOVIEN ON LOP.MaGV=GIAOVIEN.MaGV LEFT JOIN CAHOC ON LOP.MaCH = CAHOC.MaCH

WHERE NgayKhaiGiang > GETDATE()

ORDER BY NgayKhaiGiang DESC

OFFSET @hienthi\_moitrang \* (@trang - 1) ROWS

FETCH NEXT @hienthi\_moitrang ROWS ONLY

ELSE IF @tinhtrang = 3

SELECT MaLop, LOP.MaCT, TenCT, ThoiGianHoc, HocPhiCT, LOP.MaCH, CaHoc = CAHOC.GioHoc + ' ' + CAHOC.NgayHoc, LOP.MaGV, TenGV, GioiHan, SiSo, NgayKhaiGiang, NgayKetThuc

FROM LOP LEFT JOIN CHUONGTRINH ON LOP.MaCT=CHUONGTRINH.MaCT LEFT JOIN GIAOVIEN ON LOP.MaGV=GIAOVIEN.MaGV LEFT JOIN CAHOC ON LOP.MaCH = CAHOC.MaCH

WHERE NgayKetThuc < GETDATE()

ORDER BY NgayKhaiGiang DESC

OFFSET @hienthi\_moitrang \* (@trang - 1) ROWS

FETCH NEXT @hienthi\_moitrang ROWS ONLY

ELSE

SELECT MaLop, LOP.MaCT, TenCT, ThoiGianHoc, HocPhiCT, LOP.MaCH, CaHoc = CAHOC.GioHoc + ' ' + CAHOC.NgayHoc, LOP.MaGV, TenGV, GioiHan, SiSo, NgayKhaiGiang, NgayKetThuc

FROM LOP LEFT JOIN CHUONGTRINH ON LOP.MaCT=CHUONGTRINH.MaCT LEFT JOIN GIAOVIEN ON LOP.MaGV=GIAOVIEN.MaGV LEFT JOIN CAHOC ON LOP.MaCH = CAHOC.MaCH

ORDER BY NgayKhaiGiang DESC

OFFSET @hienthi\_moitrang \* (@trang - 1) ROWS

FETCH NEXT @hienthi\_moitrang ROWS ONLY

END

ELSE

BEGIN

IF @tinhtrang = 1

SELECT MaLop, LOP.MaCT, TenCT, ThoiGianHoc, HocPhiCT, LOP.MaCH, CaHoc = CAHOC.GioHoc + ' ' + CAHOC.NgayHoc, LOP.MaGV, TenGV, GioiHan, SiSo, NgayKhaiGiang, NgayKetThuc

FROM LOP LEFT JOIN (SELECT \* FROM CHUONGTRINH WHERE MaCT=@mact) CHUONGTRINH ON LOP.MaCT=CHUONGTRINH.MaCT LEFT JOIN GIAOVIEN ON LOP.MaGV=GIAOVIEN.MaGV LEFT JOIN CAHOC ON LOP.MaCH = CAHOC.MaCH

WHERE NgayKhaiGiang <= GETDATE() AND NgayKetThuc >= GETDATE()

ORDER BY NgayKhaiGiang DESC

OFFSET @hienthi\_moitrang \* (@trang - 1) ROWS

FETCH NEXT @hienthi\_moitrang ROWS ONLY

ELSE IF @tinhtrang = 2

SELECT MaLop, LOP.MaCT, TenCT, ThoiGianHoc, HocPhiCT, LOP.MaCH, CaHoc = CAHOC.GioHoc + ' ' + CAHOC.NgayHoc, LOP.MaGV, TenGV, GioiHan, SiSo, NgayKhaiGiang, NgayKetThuc

FROM LOP LEFT JOIN (SELECT \* FROM CHUONGTRINH WHERE MaCT=@mact) CHUONGTRINH ON LOP.MaCT=CHUONGTRINH.MaCT LEFT JOIN GIAOVIEN ON LOP.MaGV=GIAOVIEN.MaGV LEFT JOIN CAHOC ON LOP.MaCH = CAHOC.MaCH

WHERE NgayKhaiGiang > GETDATE()

ORDER BY NgayKhaiGiang DESC

OFFSET @hienthi\_moitrang \* (@trang - 1) ROWS

FETCH NEXT @hienthi\_moitrang ROWS ONLY

ELSE IF @tinhtrang = 3

SELECT MaLop, LOP.MaCT, TenCT, ThoiGianHoc, HocPhiCT, LOP.MaCH, CaHoc = CAHOC.GioHoc + ' ' + CAHOC.NgayHoc, LOP.MaGV, TenGV, GioiHan, SiSo, NgayKhaiGiang, NgayKetThuc

FROM LOP LEFT JOIN (SELECT \* FROM CHUONGTRINH WHERE MaCT=@mact) CHUONGTRINH ON LOP.MaCT=CHUONGTRINH.MaCT LEFT JOIN GIAOVIEN ON LOP.MaGV=GIAOVIEN.MaGV LEFT JOIN CAHOC ON LOP.MaCH = CAHOC.MaCH

WHERE NgayKetThuc < GETDATE()

ORDER BY NgayKhaiGiang DESC

OFFSET @hienthi\_moitrang \* (@trang - 1) ROWS

FETCH NEXT @hienthi\_moitrang ROWS ONLY

ELSE

SELECT MaLop, LOP.MaCT, TenCT, ThoiGianHoc, HocPhiCT, LOP.MaCH, CaHoc = CAHOC.GioHoc + ' ' + CAHOC.NgayHoc, LOP.MaGV, TenGV, GioiHan, SiSo, NgayKhaiGiang, NgayKetThuc

FROM LOP INNER JOIN (SELECT \* FROM CHUONGTRINH WHERE MaCT=@mact) CHUONGTRINH ON LOP.MaCT=CHUONGTRINH.MaCT INNER JOIN GIAOVIEN ON LOP.MaGV=GIAOVIEN.MaGV INNER JOIN CAHOC ON LOP.MaCH = CAHOC.MaCH

ORDER BY NgayKhaiGiang DESC

OFFSET @hienthi\_moitrang \* (@trang - 1) ROWS

FETCH NEXT @hienthi\_moitrang ROWS ONLY

END

END

GO

#### Tìm lớp

CREATE PROC sp\_XemLop

@malop char(7)

AS

BEGIN

SELECT MaLop, LOP.MaCT, TenCT, ThoiGianHoc, HocPhiCT, LOP.MaCH, CaHoc = CAHOC.GioHoc + ' ' + CAHOC.NgayHoc, LOP.MaGV, TenGV, GioiHan, SiSo, NgayKhaiGiang, NgayKetThuc

FROM LOP LEFT JOIN CHUONGTRINH CHUONGTRINH ON LOP.MaCT=CHUONGTRINH.MaCT LEFT JOIN GIAOVIEN ON LOP.MaGV=GIAOVIEN.MaGV LEFT JOIN CAHOC ON LOP.MaCH = CAHOC.MaCH

WHERE MaLop = @malop

END

GO

#### Mở lớp

CREATE PROC sp\_ThemLop

@mact char(3),

@mach smallint,

@magv smallint,

@gioihan smallint,

@ngaykhaigiang smalldatetime,

@ngayketthuc smalldatetime

AS

BEGIN

DECLARE @i smallint

SET @i = 1

BEGIN TRY

BEGIN TRAN

SELECT \* FROM LOP WITH (updlock) WHERE MaCT=@mact

WHILE EXISTS(SELECT \* FROM LOP WHERE MaLop = @mact + SUBSTRING(CAST(YEAR(@ngaykhaigiang) AS CHAR(4)), 3, 2) + FORMAT(@i, 'd2'))

SET @i = @i + 1

INSERT INTO LOP(MaLop, MaCT, MaCH, MaGV, GioiHan, SiSo, NgayKhaiGiang, NgayKetThuc) VALUES(@mact + SUBSTRING(CAST(YEAR(@ngaykhaigiang) AS CHAR(4)), 3, 2) + FORMAT(@i, 'd2'), @mact, @mach, @magv, @gioihan, 0, @ngaykhaigiang, @ngayketthuc)

COMMIT

END TRY

BEGIN CATCH

ROLLBACK

DECLARE @msg nvarchar(200)

SET @msg = ERROR\_MESSAGE()

RAISERROR(@msg, 16, 1)

END CATCH

END

GO

#### Hủy lớp

CREATE PROC sp\_HuyLop

@malop char(7)

AS

BEGIN

IF EXISTS(SELECT \* FROM LOP WHERE MaLop=@malop AND NgayKhaiGiang < GETDATE())

BEGIN

RAISERROR('Không được hủy lớp đã khai giảng', 16, 1)

RETURN

END

IF EXISTS(SELECT \* FROM HOC WHERE MaLop=@malop)

BEGIN

RAISERROR('Không được hủy lớp đã có học viên', 16, 1)

RETURN

END

BEGIN TRY

BEGIN TRAN

DELETE FROM DANGKY WHERE MaLop=@malop

DELETE FROM LOP WHERE MaLop=@malop

COMMIT

END TRY

BEGIN CATCH

ROLLBACK

DECLARE @msg nvarchar(200)

SET @msg = ERROR\_MESSAGE()

RAISERROR(@msg, 16, 1)

END CATCH

END

GO

#### Bố trí giáo viên cho lớp

CREATE PROC sp\_BoTriGiaoVien

@malop char(7),

@magv smallint

AS

BEGIN

IF NOT EXISTS (SELECT \* FROM LOP WHERE MaLop=@malop)

BEGIN

RAISERROR('Không tồn tại lớp có mã đã nhập', 16, 1)

RETURN

END

IF NOT EXISTS (SELECT \* FROM GIAOVIEN WHERE MaGV=@magv)

BEGIN

RAISERROR('Không tồn tại giáo viên có mã đã nhập', 16, 1)

RETURN

END

UPDATE LOP SET MaGV=@magv WHERE MaLop = @malop

END

GO

#### Xem danh sách các đăng ký

CREATE PROC sp\_TimDangKy

@sdt varchar(12),

@loc smallint,

@hienthi\_moitrang smallint,

@trang smallint

AS

BEGIN

IF @sdt IS null OR @sdt = ''

BEGIN

IF @loc = 1

SELECT MaDK, DANGKY.MaLop, SDT, NgayDK, TenDK, NgayKhaiGiang FROM DANGKY INNER JOIN LOP ON DANGKY.MaLop = LOP.MaLop WHERE NgayKhaiGiang < GETDATE()

ORDER BY NgayKhaiGiang OFFSET @hienthi\_moitrang \* (@trang - 1) ROWS FETCH NEXT @hienthi\_moitrang ROWS ONLY

ELSE

SELECT MaDK, DANGKY.MaLop, SDT, NgayDK, TenDK, NgayKhaiGiang FROM DANGKY INNER JOIN LOP ON DANGKY.MaLop = LOP.MaLop

ORDER BY NgayKhaiGiang OFFSET @hienthi\_moitrang \* (@trang - 1) ROWS FETCH NEXT @hienthi\_moitrang ROWS ONLY

END

ELSE

BEGIN

IF @loc = 1

SELECT MaDK, DANGKY.MaLop, SDT, NgayDK, TenDK, NgayKhaiGiang FROM DANGKY INNER JOIN LOP ON DANGKY.MaLop = LOP.MaLop WHERE SDT = @sdt AND NgayKhaiGiang < GETDATE()

ORDER BY NgayKhaiGiang OFFSET @hienthi\_moitrang \* (@trang - 1) ROWS FETCH NEXT @hienthi\_moitrang ROWS ONLY

ELSE

SELECT MaDK, DANGKY.MaLop, SDT, NgayDK, TenDK, NgayKhaiGiang FROM DANGKY INNER JOIN LOP ON DANGKY.MaLop = LOP.MaLop WHERE SDT = @sdt

ORDER BY NgayKhaiGiang OFFSET @hienthi\_moitrang \* (@trang - 1) ROWS FETCH NEXT @hienthi\_moitrang ROWS ONLY

END

END

GO

#### Xem lớp đăng ký

CREATE PROC sp\_XemLopDK

@madk Int

AS

BEGIN

SELECT MaLop, LOP.MaCT, TenCT, ThoiGianHoc, HocPhiCT, LOP.MaCH, CaHoc = CAHOC.GioHoc + ' ' + CAHOC.NgayHoc, LOP.MaGV, TenGV, GioiHan, SiSo, NgayKhaiGiang, NgayKetThuc

FROM LOP LEFT JOIN CHUONGTRINH CHUONGTRINH ON LOP.MaCT=CHUONGTRINH.MaCT LEFT JOIN GIAOVIEN ON LOP.MaGV=GIAOVIEN.MaGV LEFT JOIN CAHOC ON LOP.MaCH = CAHOC.MaCH

WHERE MaLop = (SELECT MaLop FROM DANGKY WHERE MaDK=@madk)

END

GO

#### Đăng ký

@malop char(7),

@sdt nvarchar(12),

@ten nvarchar(40)

AS

BEGIN

IF NOT EXISTS(SELECT \* FROM LOP WHERE MaLop=@malop)

BEGIN

RAISERROR('Lớp không tồn tại',16,1)

RETURN

END

IF EXISTS(SELECT \* FROM LOP WHERE MaLop=@malop AND NgayKhaiGiang < GETDATE())

BEGIN

RAISERROR('Không được đăng ký lớp đã khai giảng',16,1)

RETURN

END

DECLARE @i int

SET @i = 1

BEGIN TRY

BEGIN TRAN

SELECT \* FROM DANGKY with (updlock) WHERE MaDK = @i

WHILE EXISTS(SELECT \* FROM DANGKY WHERE MaDK=@i)

SET @i = @i + 1

INSERT INTO DANGKY(MaLop, SDT, MaDK, NgayDK, TenDK) VALUES(@malop, @sdt, @i, GETDATE(), @ten)

UPDATE LOP SET SiSo = SiSo + 1 WHERE MaLop = @malop

COMMIT

END TRY

BEGIN CATCH

ROLLBACK

DECLARE @msg nvarchar(200)

SET @msg = ERROR\_MESSAGE()

RAISERROR(@msg, 16, 1)

END CATCH

END

GO

#### Hủy đăng ký

CREATE PROC sp\_HuyDangKy

@madk int

AS

BEGIN

BEGIN TRY

BEGIN TRAN

DECLARE @malop char(7)

SELECT @malop = MaLop FROM DANGKY WHERE MaDK = @madk

DELETE FROM DANGKY WHERE MaDK = @madk

UPDATE LOP SET SiSo = SiSo - 1 WHERE MaLop = @malop

COMMIT

END TRY

BEGIN CATCH

ROLLBACK

DECLARE @msg nvarchar(200)

SET @msg = ERROR\_MESSAGE()

RAISERROR(@msg, 16, 1)

END CATCH

END

GO

#### Xem danh sách học viên

CREATE PROC sp\_TimHocVien

@sdt varchar(12),

@hienthi\_moitrang smallint,

@trang smallint

AS

BEGIN

IF @sdt IS NULL OR @sdt = ''

SELECT \* FROM HOCVIEN

ORDER BY MaHV DESC

OFFSET @hienthi\_moitrang \* (@trang - 1) ROWS

FETCH NEXT @hienthi\_moitrang ROWS ONLY

ELSE

SELECT \* FROM HOCVIEN

WHERE SDTHV = @sdt

ORDER BY MaHV DESC

OFFSET @hienthi\_moitrang \* (@trang - 1) ROWS

FETCH NEXT @hienthi\_moitrang ROWS ONLY

END

GO

#### Tìm học viên

CREATE PROC sp\_XemHocVien

@mahv int

AS

BEGIN

SELECT \* FROM HOCVIEN WHERE MaHV=@mahv

END

GO

#### Thêm học viên

CREATE PROC sp\_ThemHocVien

@ten nvarchar(40),

@sdt nvarchar(12),

@email nvarchar(30)

AS

BEGIN

DECLARE @i Int

SET @i = 1

BEGIN TRY

BEGIN TRAN

SELECT \* FROM HOCVIEN WITH (updlock)

WHILE EXISTS(SELECT \* FROM HOCVIEN WHERE MaHV=@i)

SET @i = @i + 1

INSERT INTO HOCVIEN(MaHV, TenHV, SDTHV, EmailHV) VALUES(@i, @ten, @sdt, @email)

COMMIT

END TRY

BEGIN CATCH

ROLLBACK

DECLARE @msg nvarchar(200)

SET @msg = ERROR\_MESSAGE()

RAISERROR(@msg, 16, 1)

END CATCH

END

GO

#### Xem danh sách hóa đơn

CREATE PROC sp\_TimHoaDon

@sdt varchar(12),

@hienthi\_moitrang smallint,

@trang smallint

AS

BEGIN

IF @sdt IS NULL OR @sdt = ''

SELECT MaHD, HOADON.MaHV, TenHV, SDTHV, NgayLapHD, HocPhiDong

FROM HOADON INNER JOIN HOCVIEN ON HOADON.MaHV = HOCVIEN.MaHV

ORDER BY NgayLapHD DESC

OFFSET @hienthi\_moitrang\*(@trang - 1) ROWS

FETCH NEXT @hienthi\_moitrang ROWS ONLY

ELSE

SELECT MaHD, HOADON.MaHV, TenHV, SDTHV, NgayLapHD, HocPhiDong

FROM HOADON INNER JOIN (SELECT \* FROM HOCVIEN WHERE SDTHV = @sdt) HOCVIEN ON HOADON.MaHV = HOCVIEN.MaHV

ORDER BY NgayLapHD DESC

OFFSET @hienthi\_moitrang\*(@trang - 1) ROWS

FETCH NEXT @hienthi\_moitrang ROWS ONLY

END

GO

#### Tìm hóa đơn

CREATE PROC sp\_XemHoaDon

@mahd int

AS

BEGIN

SELECT MaHD, HOADON.MaHV, TenHV, SDTHV, NgayLapHD, HocPhiDong

FROM (SELECT \* FROM HOADON WHERE MaHD=@mahd) HOADON INNER JOIN HOCVIEN ON HOADON.MaHV = HOCVIEN.MaHV

END

GO

#### Xem chi tiết hóa đơn

CREATE PROC sp\_XemCacLopHoaDon

@mahd int

AS

BEGIN

SELECT HOC.MaLop, LOP.MaCT, TenCT, ThoiGianHoc, HocPhiCT, LOP.MaCH, CaHoc = CAHOC.GioHoc + ' ' + CAHOC.NgayHoc, LOP.MaGV, TenGV, GioiHan, SiSo, NgayKhaiGiang, NgayKetThuc

FROM (SELECT MaLop FROM HOC WHERE MaHD=@mahd) HOC INNER JOIN LOP ON HOC.MaLop = LOP.MaLop LEFT JOIN CHUONGTRINH CHUONGTRINH ON LOP.MaCT=CHUONGTRINH.MaCT LEFT JOIN GIAOVIEN ON LOP.MaGV=GIAOVIEN.MaGV LEFT JOIN CAHOC ON LOP.MaCH = CAHOC.MaCH

END

GO

#### Thu học phí

CREATE PROC sp\_DongHocPhi

@mahv int,

@lops nvarchar(70),

@dks nvarchar(70)

AS

BEGIN

IF NOT EXISTS(SELECT \* FROM HOCVIEN WHERE MaHV=@mahv)

BEGIN

RAISERROR('Không tồn tại học viên có mã được nhập', 16, 1)

RETURN

END

IF @lops IS NULL AND @lops = ''

BEGIN

RAISERROR('Phải đóng học phí cho ít nhất 1 lớp', 16, 1)

RETURN

END

DECLARE @malop varchar(10), @madk nvarchar(3), @mahd int, @gia int

SET @mahd = 1

BEGIN TRY

BEGIN TRAN

SELECT @gia = NULL FROM HOADON with (updlock)

WHILE EXISTS(SELECT \* FROM HOADON WHERE MaHD=@mahd)

SET @mahd = @mahd + 1

INSERT INTO HOADON(MaHD, MaHV, HocPhiDong, NgayLapHD) VALUES(@mahd, @mahv, 0, GETDATE())

IF @dks IS NOT NULL AND @dks != ''

BEGIN

DECLARE C\_MaDK CURSOR FOR SELECT \* FROM STRING\_SPLIT(@dks, ',')

OPEN C\_MaDK

FETCH NEXT FROM C\_MaDK INTO @madk

WHILE @@FETCH\_STATUS = 0

BEGIN

IF NOT EXISTS(SELECT \* FROM DANGKY WHERE MaDK = @madk)

BEGIN

RAISERROR('Không tìm thấy đăng ký được nhập',16,1)

CLOSE C\_MaDK

DEALLOCATE C\_MaDK

ROLLBACK

RETURN

END

SELECT @malop = MaLop FROM DANGKY WHERE MaDK = @madk

DELETE FROM DANGKY WHERE MaDK=@madk

INSERT INTO HOC(MaHD, MaLop, DiemChuyenCan, DiemCuoiKy)

VALUES(@mahd, @malop, 0, 0)

FETCH NEXT FROM C\_MaDK INTO @madk

END

CLOSE C\_MaDK

DEALLOCATE C\_MaDK

END

DECLARE C\_MaLop CURSOR FOR SELECT \* FROM STRING\_SPLIT(@lops, ',')

OPEN C\_MaLop

FETCH NEXT FROM C\_MaLop INTO @malop

WHILE @@FETCH\_STATUS = 0

BEGIN

IF NOT EXISTS(SELECT \* FROM LOP WHERE MaLop = @malop)

BEGIN

RAISERROR('Không tồn tại lớp có mã được nhập',16,1)

CLOSE C\_MaLop

DEALLOCATE C\_MaLop

ROLLBACK

RETURN

END

IF EXISTS(SELECT \* FROM LOP WHERE MaLop = @malop AND NgayKetThuc < GETDATE())

BEGIN

RAISERROR('Không thu học phí cho lớp đã kết thúc',16,1)

CLOSE C\_MaLop

DEALLOCATE C\_MaLop

ROLLBACK

RETURN

END

INSERT INTO HOC(MaHD, MaLop, DiemChuyenCan, DiemCuoiKy)

VALUES(@mahd, @malop, 0, 0)

FETCH NEXT FROM C\_MaLop INTO @malop

END

CLOSE C\_MaLop

DEALLOCATE C\_MaLop

COMMIT

SELECT @mahd

END TRY

BEGIN CATCH

ROLLBACK

DECLARE @msg nvarchar(200)

SET @msg = ERROR\_MESSAGE()

RAISERROR(@msg, 16, 1)

END CATCH

END

GO

#### Xem danh sách học viên của lớp

CREATE FUNCTION F\_XemBaoLuu(@mahd int, @malop char(7))

RETURNS char(1)

AS

BEGIN

IF EXISTS (SELECT \* FROM BAOLUU WHERE MaHD=@mahd AND MaLop=@malop)

RETURN 'Y'

RETURN 'N'

END

GO

CREATE FUNCTION F\_XemDanhSachLop(@malop char(7))

RETURNS @danhsach TABLE(MaHV int, TenHV nvarchar(40), SDT nvarchar(12), DiemChuyenCan float, DiemCuoiKy float, MaHD int, BaoLuu char(1))

AS

BEGIN

INSERT INTO @danhsach(HOADON.MaHV, TenHV, SDT, DiemChuyenCan, DiemCuoiKy, MaHD, BaoLuu)

SELECT HOADON.MaHV, TenHV, SDTHV, DiemChuyenCan, DiemCuoiKy, HOADON.MaHD, dbo.F\_XemBaoLuu(HOADON.MaHD, MaLop)

FROM (SELECT \* FROM HOC WHERE MaLop=@malop) HOC INNER JOIN HOADON ON HOC.MaHD = HOADON.MaHD INNER JOIN HOCVIEN ON HOADON.MaHV=HOCVIEN.MaHV

INSERT INTO @danhsach(TenHV, SDT, DiemChuyenCan, DiemCuoiKy)

SELECT TenDK, SDT, 0, 0

FROM DANGKY

WHERE MaLop=@malop

RETURN

END

GO

CREATE PROC sp\_XemDanhSachLop

@malop char(7)

AS

BEGIN

SELECT \*

FROM F\_XemDanhSachLop(@malop)

ORDER BY REVERSE(LEFT(REVERSE(TenHV), CHARINDEX(' ', REVERSE(TenHV)) - 1))

END

GO

#### Xem các lớp học viên đã học

CREATE PROC sp\_XemLichSuHoc

@mahv int

AS

BEGIN

SELECT HOADON.MaHV, Lop.MaLop, LOP.NgayKhaiGiang, LOP.NgayKetThuc, TenHV, SDTHV, DiemChuyenCan, DiemCuoiKy, HOADON.MaHD, BaoLuu = dbo.F\_XemBaoLuu(HOADON.MaHD, HOC.MaLop)

FROM HOC INNER JOIN HOADON ON HOC.MaHD = HOADON.MaHD INNER JOIN (SELECT \* FROM HOCVIEN WHERE MaHV = @mahv) HOCVIEN ON HOADON.MaHV=HOCVIEN.MaHV INNER JOIN LOP ON HOC.MaLop = LOP.MaLop

ORDER BY NgayKhaiGiang

END

GO

#### Chuyển lớp

CREATE PROC sp\_ChuyenLop

@mahv int,

@malop\_cu char(7),

@malop\_moi char(7)

AS

BEGIN

IF NOT EXISTS(SELECT \* FROM HOCVIEN WHERE MaHV=@mahv)

BEGIN

RAISERROR('Không tồn tại học viên có mã học viên được nhập', 16, 1)

RETURN

END

IF NOT EXISTS(SELECT \* FROM LOP WHERE MaLop=@malop\_cu OR MaLop=@malop\_moi)

BEGIN

RAISERROR('Không tồn tại lớp có mã lớp được nhập', 16, 1)

RETURN

END

IF @malop\_cu = @malop\_moi

BEGIN

RAISERROR('Lớp cũ và lớp mới phải khác nhau', 16, 1)

RETURN

END

IF SUBSTRING(@malop\_cu, 1, 3) != SUBSTRING(@malop\_moi, 1, 3)

BEGIN

RAISERROR('Chỉ được chuyển sang lớp cùng chương trình học', 16, 1)

RETURN

END

IF EXISTS(SELECT \* FROM LOP WHERE MaLop=@malop\_moi AND NgayKetThuc < GETDATE())

BEGIN

RAISERROR('Lớp xin được chuyển đến đã kết thúc', 16, 1)

RETURN

END

IF NOT EXISTS(SELECT \* FROM HOC WHERE MaLop=@malop\_cu AND MaHD IN (SELECT MaHD FROM HOADON WHERE MaHV=@mahv))

BEGIN

RAISERROR('Học viên không ở trong danh sách của lớp cũ', 16, 1)

RETURN

END

IF EXISTS(SELECT \* FROM HOC WHERE MaLop=@malop\_moi AND MaHD IN (SELECT MaHD FROM HOADON WHERE MaHV=@mahv))

BEGIN

RAISERROR('Học viên đã có trong danh sách học viên của lớp mới', 16, 1)

RETURN

END

DECLARE @mahd int

SELECT @mahd = MaHD FROM HOC WHERE MaLop=@malop\_cu AND MaHD IN (SELECT MaHD FROM HOADON WHERE MaHV=@mahv)

BEGIN TRY

BEGIN TRAN

DELETE FROM BAOLUU WHERE MaHD=@mahd AND MaLop=@malop\_cu

DELETE FROM HOC WHERE MaHD=@mahd AND MaLop=@malop\_cu

UPDATE LOP SET SiSo = SiSo - 1 WHERE MaLop=@malop\_cu

INSERT INTO HOC(MaHD, MaLop, DiemChuyenCan, DiemCuoiKy) VALUES(@mahd, @malop\_moi, 0, 0)

UPDATE LOP SET SiSo = SiSo + 1 WHERE MaLop=@malop\_moi

COMMIT

END TRY

BEGIN CATCH

ROLLBACK

DECLARE @msg nvarchar(200)

SET @msg = ERROR\_MESSAGE()

RAISERROR(@msg, 16, 1)

END CATCH

END

GO

#### Xem danh sách các học viên có thể chấm điểm (học viên đã đóng học phí và không bảo lưu)

CREATE PROC sp\_DanhSachChamDiem

@malop char(7)

AS

BEGIN

SELECT \*

FROM dbo.F\_XemDanhSachLop(@malop)

WHERE BaoLuu='N'

END

GO

#### Chấm điểm

CREATE PROC sp\_ChamDiem

@mahd int,

@malop char(7),

@diemchuyencan float,

@diemcuoiky float

AS

BEGIN

IF @diemchuyencan < 0 OR @diemchuyencan > 10 OR @diemcuoiky < 0 OR @diemcuoiky > 10

BEGIN

RAISERROR('Điểm không hợp lệ',16,1)

RETURN

END

UPDATE HOC SET DiemChuyenCan=@diemchuyencan, DiemCuoiKy=@diemcuoiky WHERE MaHD=@mahd AND MaLop=@malop

END

GO

#### Xem danh sách các thông tin bảo lưu

CREATE PROC sp\_LietKeBaoLuu

AS

SELECT BAOLUU.MaHD, NgayBaoLuu, HOADON.MaHV, TenHV, SDTHV, BAOLUU.MaLop

FROM BAOLUU INNER JOIN HOADON ON BAOLUU.MaHD = HOADON.MaHD INNER JOIN HOCVIEN ON HOADON.MaHV = HOCVIEN.MaHV

ORDER BY NgayBaoLuu

GO

#### Bảo lưu

CREATE PROC sp\_BaoLuu

@mahv int,

@malop char(7)

AS

BEGIN

IF NOT EXISTS(SELECT \* FROM HOCVIEN WHERE MaHV=@mahv)

BEGIN

RAISERROR('Không tồn tại học viên có mã học viên được nhập',16,1)

RETURN

END

IF NOT EXISTS(SELECT \* FROM LOP WHERE MaLop=@malop)

BEGIN

RAISERROR('Không tồn tại lớp có mã lớp được nhập', 16, 1)

RETURN

END

IF EXISTS(SELECT \* FROM LOP WHERE MaLop=@malop AND NgayKetThuc < GETDATE())

BEGIN

RAISERROR('Không thể bảo lưu cho lớp đã kết thúc', 16, 1)

RETURN

END

IF NOT EXISTS(SELECT \* FROM HOADON WHERE MaHV = @mahv AND MaHD IN (SELECT MaHD FROM HOC WHERE MaLop=@malop))

BEGIN

RAISERROR('Học viên không theo học lớp này', 16, 1)

RETURN

END

DECLARE @mahd int

SELECT @mahd=MaHD FROM HOADON WHERE MaHV = @mahv AND MaHD IN (SELECT MaHD FROM HOC WHERE MaLop=@malop)

IF (SELECT dbo.F\_XemBaoLuu(@mahd, @malop)) = 'Y'

BEGIN

RAISERROR('Học viên đã xin bảo lưu cho lớp này', 16, 1)

RETURN

END

INSERT INTO BAOLUU(MaHD, MaLop, NgayBaoLuu) VALUES(@mahd, @malop, GETDATE())

END

GO

#### Hủy bảo lưu

CREATE PROC sp\_HuyBaoLuu

@mahd int,

@malop char(7)

AS

BEGIN

DELETE FROM BAOLUU WHERE MaHD=@mahd AND MaLop=@malop

END

GO

#### Xem doanh thu

CREATE PROC sp\_XemDoanhThu

AS

SELECT Nam = YEAR(NgayLapHD), Thang = MONTH(NgayLapHD), DoanhThu = SUM(HocPhiDong)

FROM HOADON

GROUP BY YEAR(NgayLapHD), MONTH(NgayLapHD)

ORDER BY Nam DESC, Thang DESC

GO

### Phân quyền

#### Tạo login mặc định

IF EXISTS(SELECT \* FROM master.dbo.syslogins WHERE NAME = 'TSK\_Client')

DROP LOGIN TSK\_Client

CREATE LOGIN TSK\_Client WITH PASSWORD = 'TOEIC\_SaoKhue'

GO

CREATE USER Client FOR LOGIN TSK\_Client

GO

#### Tạo Roles

CREATE ROLE TSK\_QuanLy

GO

CREATE ROLE TSK\_TuVan

GO

CREATE ROLE TSK\_GiaoVien

GO

#### Cấp quyền cho các Roles

GRANT CONTROL ON DATABASE::TOEIC\_SaoKhue TO TSK\_QuanLy

GRANT SELECT ON TAIKHOAN TO Client

GRANT SELECT, INSERT, UPDATE, DELETE ON TAIKHOAN TO TSK\_QuanLy

GRANT SELECT, INSERT, UPDATE, DELETE ON CAHOC TO TSK\_QuanLy

GRANT SELECT ON CAHOC TO TSK\_TuVan, TSK\_GiaoVien

GRANT SELECT, INSERT, UPDATE, DELETE ON CHUONGTRINH TO TSK\_QuanLy

GRANT SELECT ON CHUONGTRINH TO TSK\_TuVan, TSK\_GiaoVien

GRANT SELECT, INSERT, UPDATE, DELETE ON LOP TO TSK\_QuanLy

GRANT SELECT, UPDATE ON LOP TO TSK\_TuVan

GRANT SELECT ON LOP TO TSK\_GiaoVien

GRANT SELECT, INSERT, UPDATE, DELETE ON GIAOVIEN TO TSK\_QuanLy

GRANT SELECT ON GIAOVIEN TO TSK\_TuVan, TSK\_GiaoVien

GRANT SELECT ON HOCVIEN TO TSK\_QuanLy, TSK\_GiaoVien

GRANT SELECT, INSERT, UPDATE, DELETE ON HOCVIEN TO TSK\_TuVan

GRANT SELECT ON DANGKY TO TSK\_QuanLy, TSK\_GiaoVien

GRANT SELECT, INSERT, UPDATE, DELETE ON DANGKY TO TSK\_TuVan

GRANT SELECT ON HOADON TO TSK\_QuanLy

GRANT SELECT, INSERT, UPDATE, DELETE ON HOADON TO TSK\_TuVan

GRANT SELECT ON HOC TO TSK\_QuanLy

GRANT SELECT, INSERT, UPDATE, DELETE ON HOC TO TSK\_TuVan

GRANT SELECT, UPDATE ON HOC TO TSK\_GiaoVien

GRANT SELECT ON BAOLUU TO TSK\_QuanLy, TSK\_GiaoVien

GRANT SELECT, INSERT, UPDATE, DELETE ON BAOLUU TO TSK\_TuVan

GO

GRANT EXECUTE ON sp\_DangNhap TO Client

GO

GRANT EXEC ON sp\_LietKeTaiKhoan TO TSK\_QuanLy

GO

GRANT EXEC ON sp\_TaoTaiKhoan TO TSK\_QuanLy

GO

GRANT EXEC ON sp\_XoaTaiKhoan TO TSK\_QuanLy

GO

GRANT EXEC ON sp\_ThayDoiMatKhau TO TSK\_QuanLy

GO

GRANT EXEC ON sp\_LietKeGiaoVien TO TSK\_QuanLy, TSK\_TuVan, TSK\_GiaoVien

GO

GRANT EXEC ON sp\_XemGiaoVien TO TSK\_QuanLy, TSK\_TuVan, TSK\_GiaoVien

GO

GRANT EXEC ON sp\_XemGiaoVien TO TSK\_QuanLy, TSK\_TuVan, TSK\_GiaoVien

GO

GRANT EXEC ON sp\_XoaGiaoVien TO TSK\_QuanLy

GO

GRANT EXEC ON sp\_CapNhatGiaoVien TO TSK\_QuanLy

GO

GRANT EXEC ON sp\_XemLichSuDay TO TSK\_QuanLy, TSK\_TuVan, TSK\_GiaoVien

GO

GRANT EXEC ON sp\_LietKeCaHoc TO TSK\_QuanLy, TSK\_TuVan, TSK\_GiaoVien

GO

GRANT EXEC ON sp\_ThemCaHoc TO TSK\_QuanLy

GO

GRANT EXEC ON sp\_XoaCaHoc TO TSK\_QuanLy

GO

GRANT EXEC ON sp\_CapNhatCaHoc TO TSK\_QuanLy

GO

GRANT EXEC ON sp\_LietKeChuongTrinh TO TSK\_QuanLy, TSK\_TuVan, TSK\_GiaoVien

GO

GRANT EXEC ON sp\_ThemChuongTrinh TO TSK\_QuanLy

GO

GRANT EXEC ON sp\_CapNhatChuongTrinh TO TSK\_QuanLy

GO

GRANT EXEC ON sp\_TimLop TO TSK\_QuanLy, TSK\_TuVan, TSK\_GiaoVien

GO

GRANT EXEC ON sp\_XemLop TO TSK\_QuanLy, TSK\_TuVan, TSK\_GiaoVien

GO

GRANT EXEC ON sp\_ThemLop TO TSK\_QuanLy

GO

GRANT EXEC ON sp\_HuyLop TO TSK\_QuanLy

GO

GRANT EXEC ON sp\_BoTriGiaoVien TO TSK\_QuanLy

GO

GRANT EXEC ON sp\_TimDangKy TO TSK\_TuVan

GO

GRANT EXEC ON sp\_XemLopDK TO TSK\_TuVan

GO

GRANT EXEC ON sp\_DangKy TO TSK\_TuVan

GO

GRANT EXEC ON sp\_HuyDangKy TO TSK\_TuVan

GO

GRANT EXEC ON sp\_TimHocVien TO TSK\_TuVan, TSK\_GiaoVien, TSK\_QuanLy

GO

GRANT EXEC ON sp\_XemHocVien TO TSK\_TuVan, TSK\_GiaoVien, TSK\_QuanLy

GO

GRANT EXEC ON sp\_ThemHocVien TO TSK\_TuVan

GO

GRANT EXEC ON sp\_TimHoaDon TO TSK\_TuVan, TSK\_QuanLy

GO

GRANT EXEC ON sp\_XemHoaDon TO TSK\_TuVan, TSK\_QuanLy

GO

GRANT EXEC ON sp\_DongHocPhi TO TSK\_TuVan

GO

GRANT EXEC ON sp\_XemCacLopHoaDon TO TSK\_TuVan, TSK\_QuanLy

GO

GRANT SELECT ON F\_XemDanhSachLop TO TSK\_TuVan, TSK\_QuanLy, TSK\_GiaoVien

GO

GRANT EXECUTE ON F\_XemBaoLuu TO TSK\_TuVan, TSK\_QuanLy, TSK\_GiaoVien

GO

GRANT EXECUTE ON sp\_XemDanhSachLop TO TSK\_TuVan, TSK\_QuanLy, TSK\_GiaoVien

GO

GRANT EXECUTE ON sp\_XemLichSuHoc TO TSK\_TuVan, TSK\_QuanLy, TSK\_GiaoVien

GO

GRANT EXECUTE ON sp\_ChuyenLop TO TSK\_TuVan

GO

GRANT EXECUTE ON sp\_DanhSachChamDiem TO TSK\_GiaoVien

GO

GRANT EXECUTE ON sp\_ChamDiem TO TSK\_GiaoVien

GO

GRANT EXECUTE ON sp\_LietKeBaoLuu TO TSK\_TuVan

GO

GRANT EXECUTE ON sp\_BaoLuu TO TSK\_TuVan

GO

GRANT EXECUTE ON sp\_HuyBaoLuu TO TSK\_TuVan

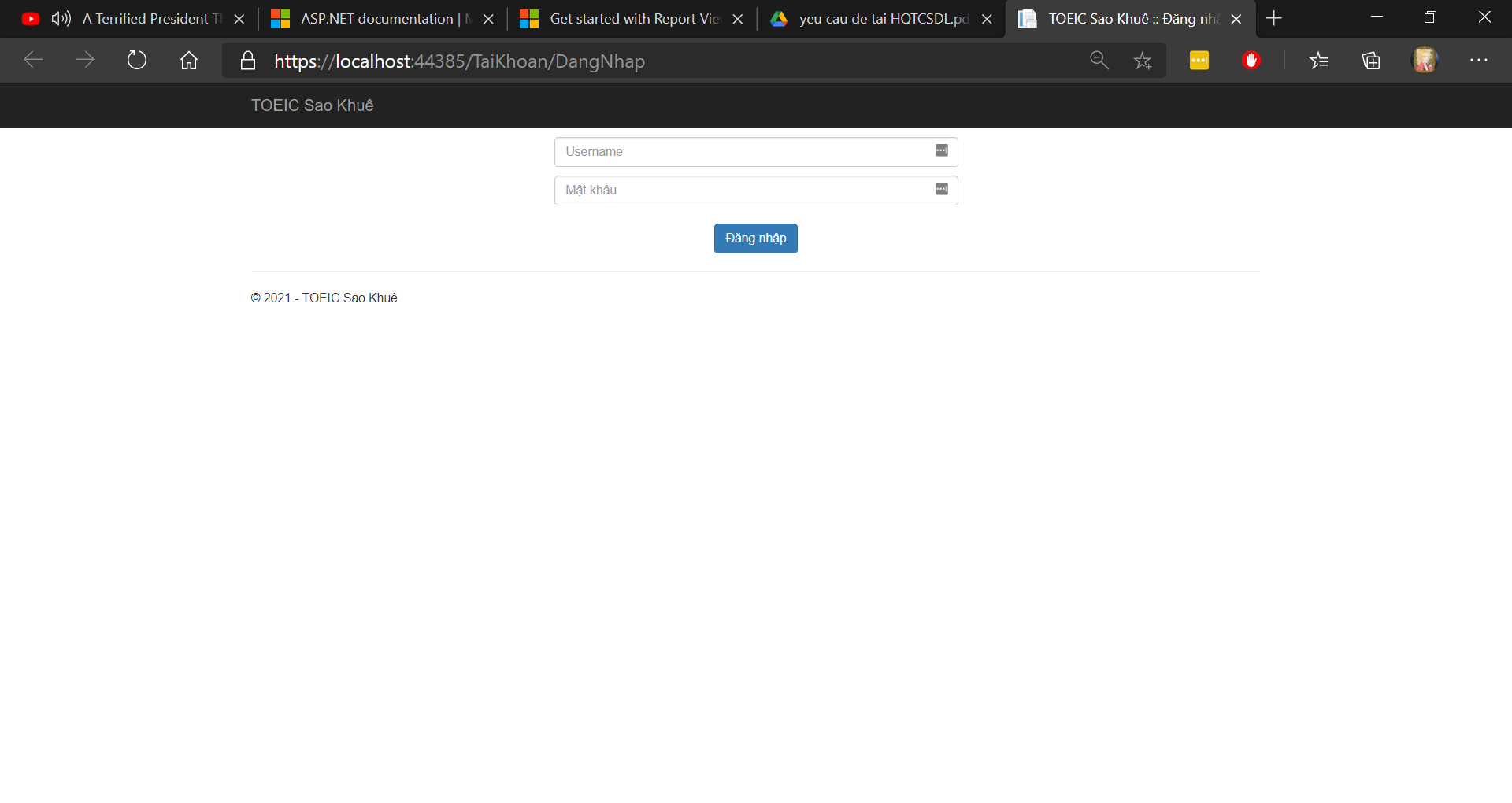
GO

GRANT EXEC ON sp\_XemDoanhThu TO TSK\_QuanLy

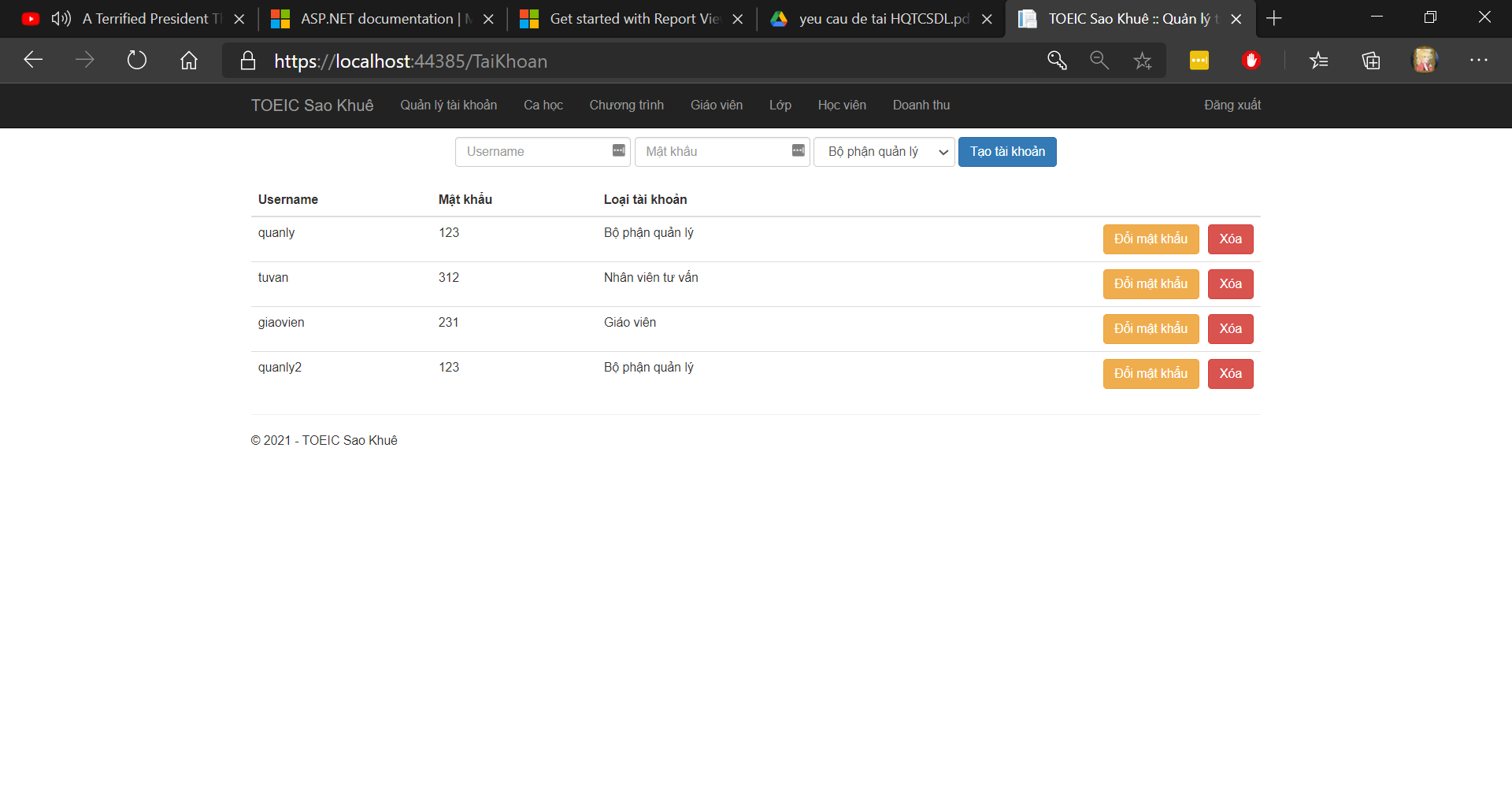
GO

## Phía Client

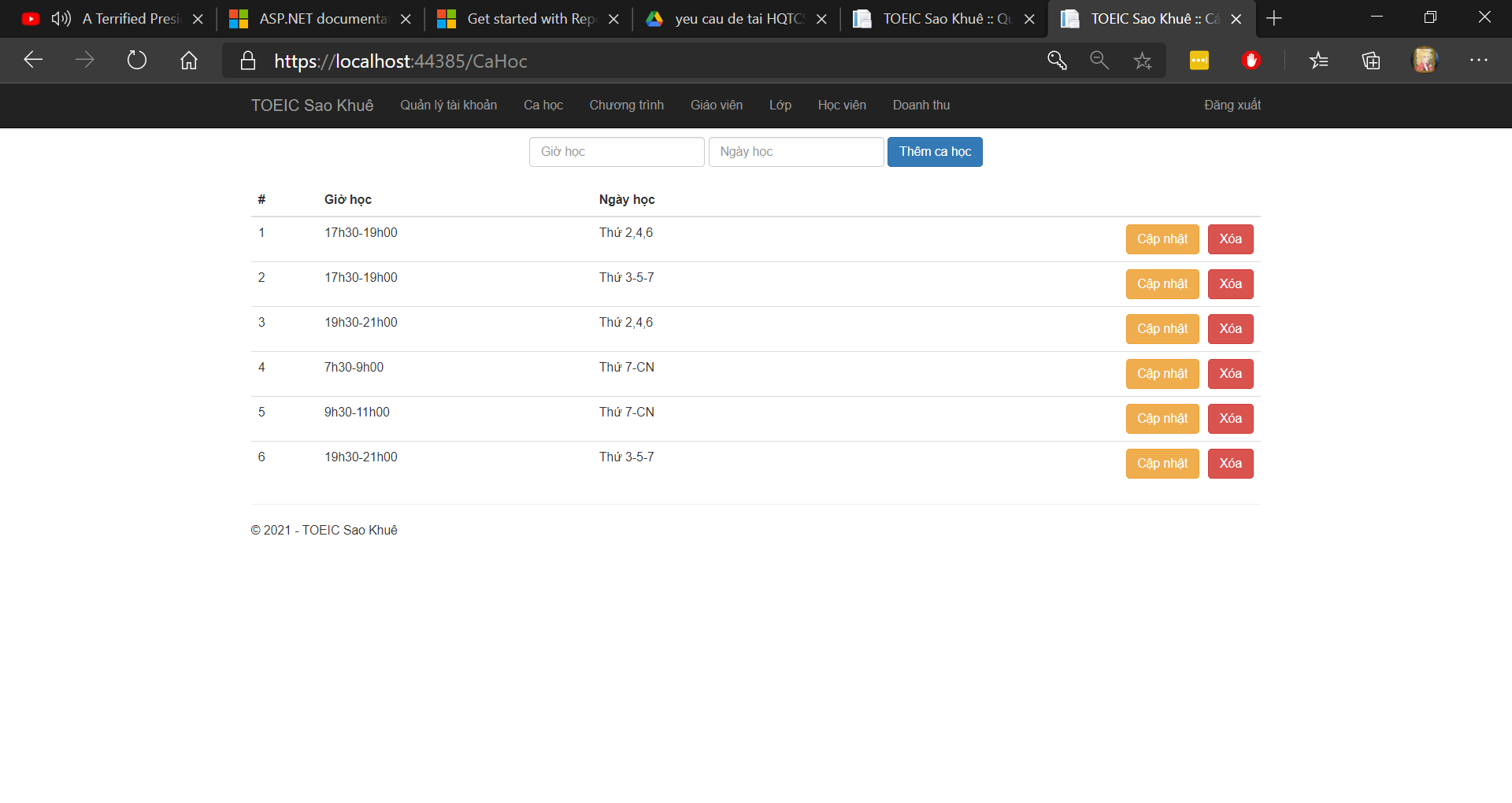
### Giao diện thực hiện các chức năng



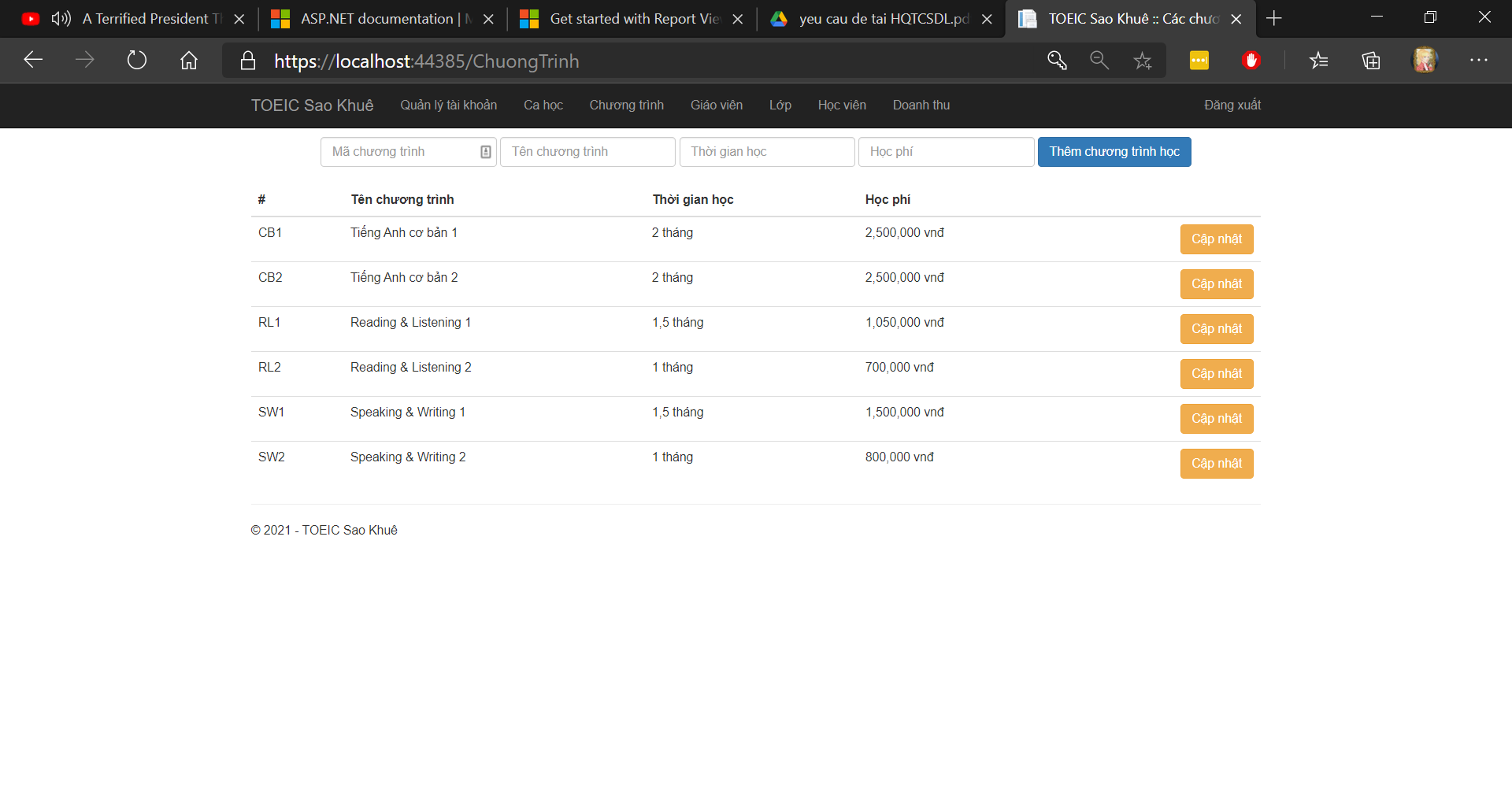
Hình 13 Giao diện Đăng nhập



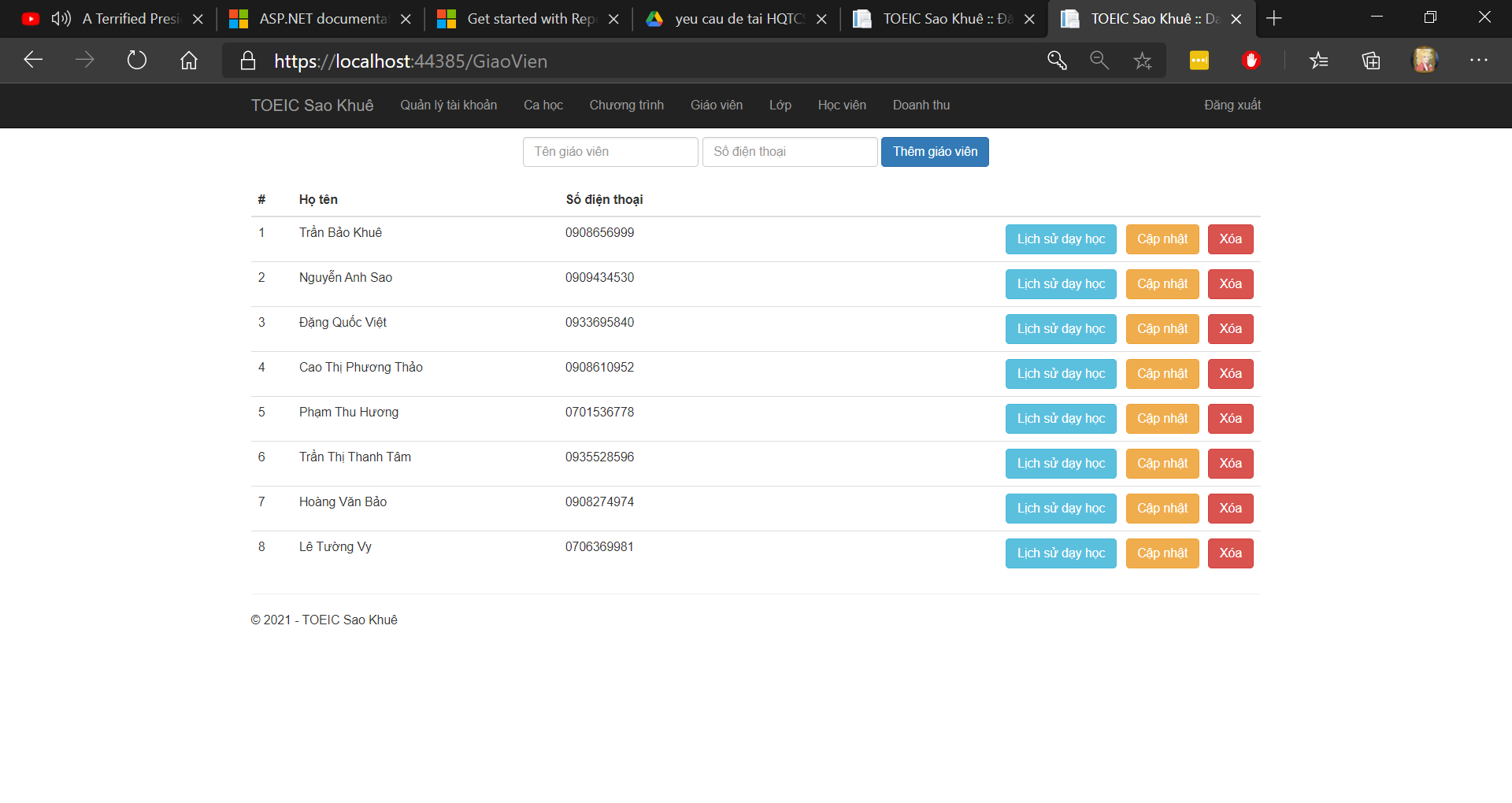
Hình 14 Giao diện Quản lý tài khoản



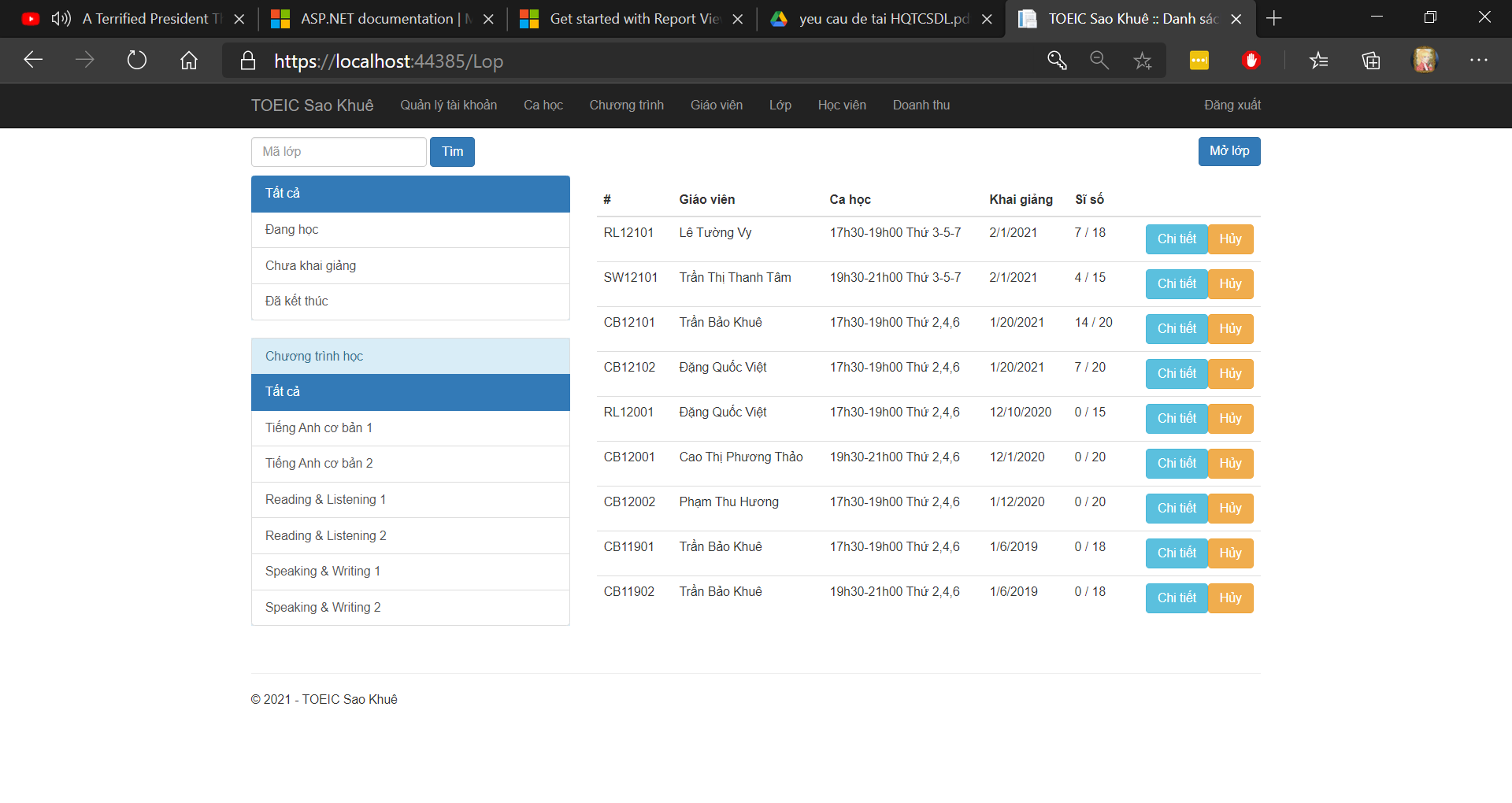
Hình 15 Giao diện Quản lý ca học



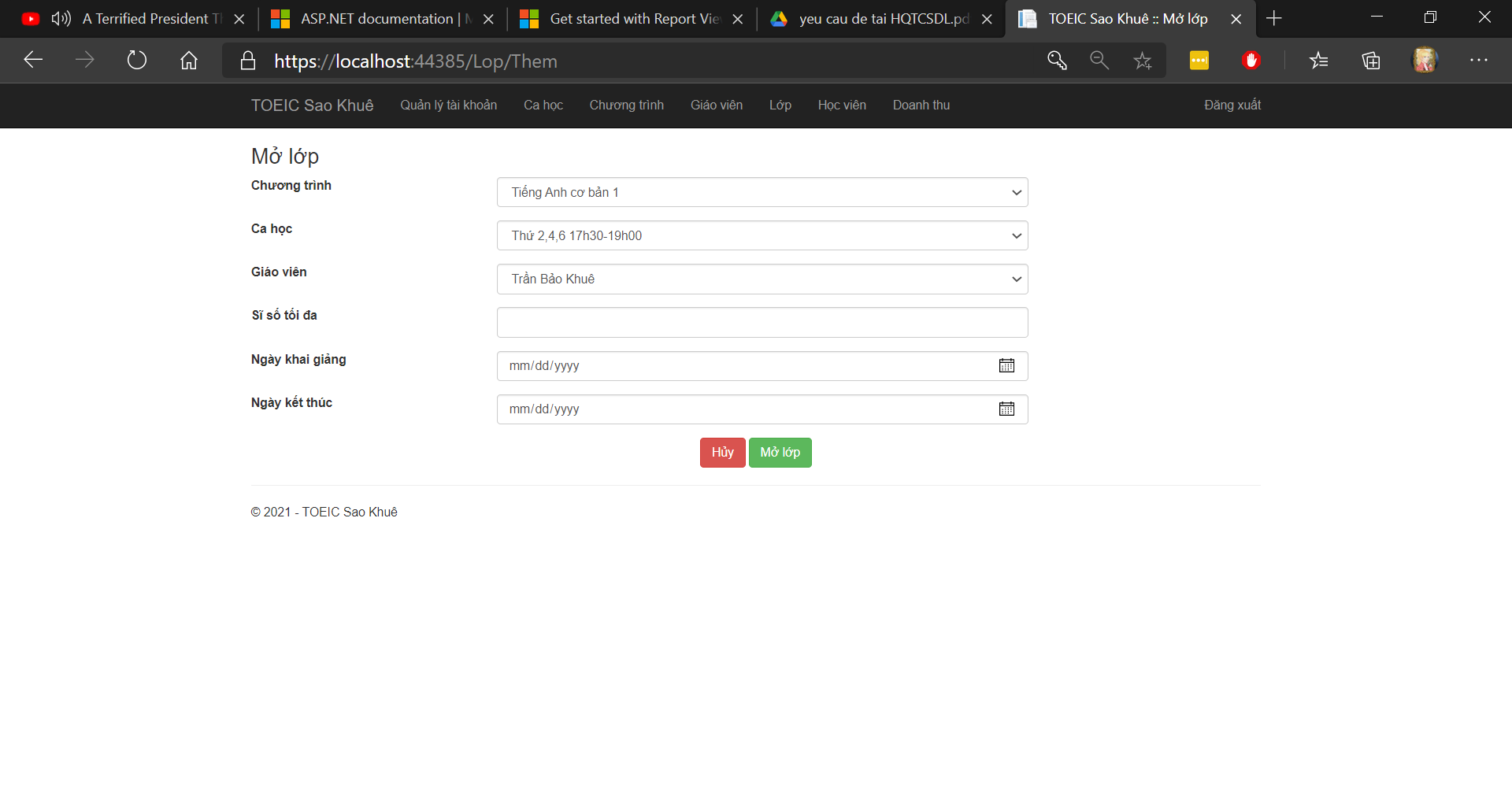
Hình 16 Giao diện Quản lý chương trình học



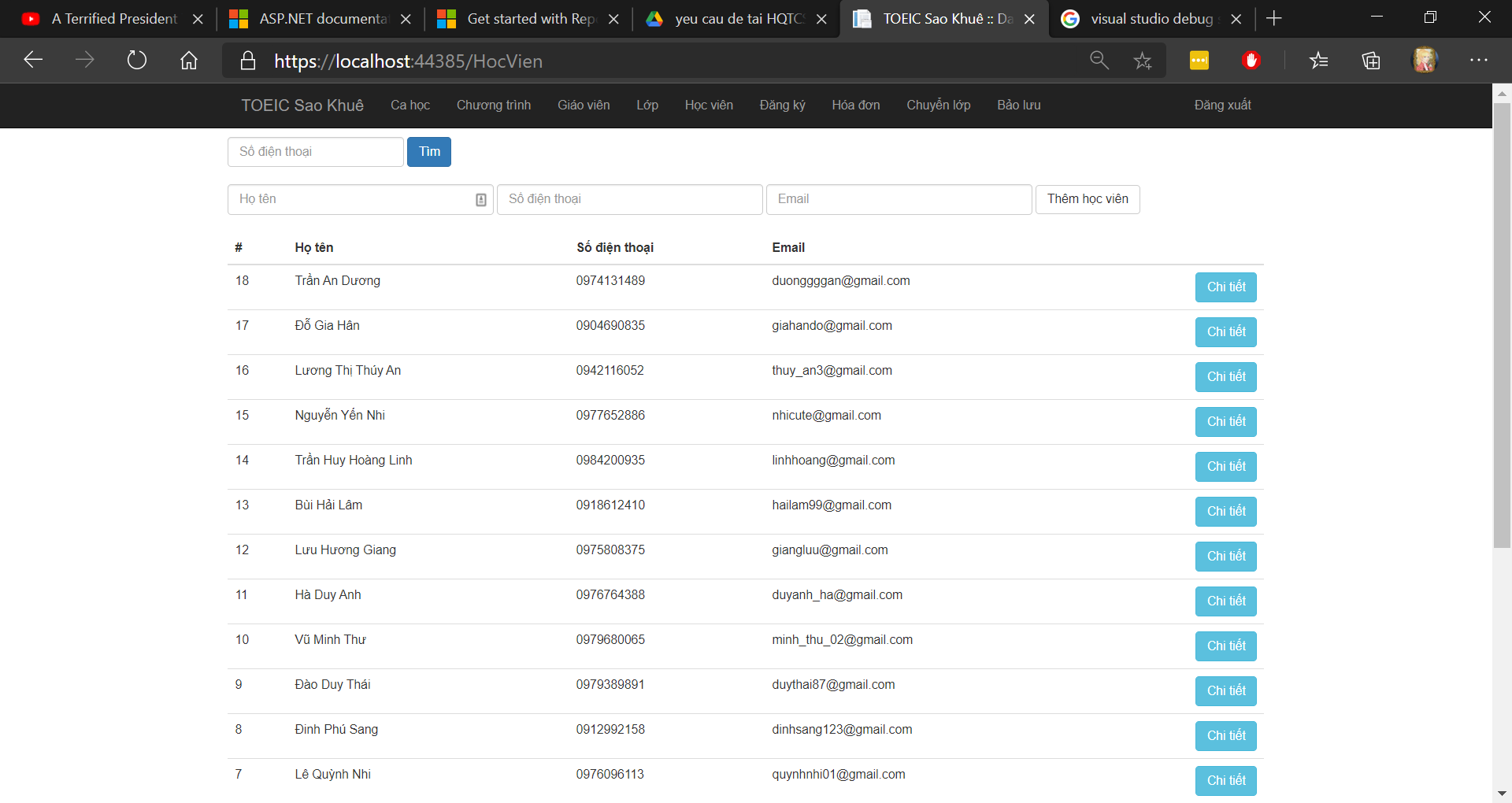
Hình 17 Giao diện Quản lý giáo viên



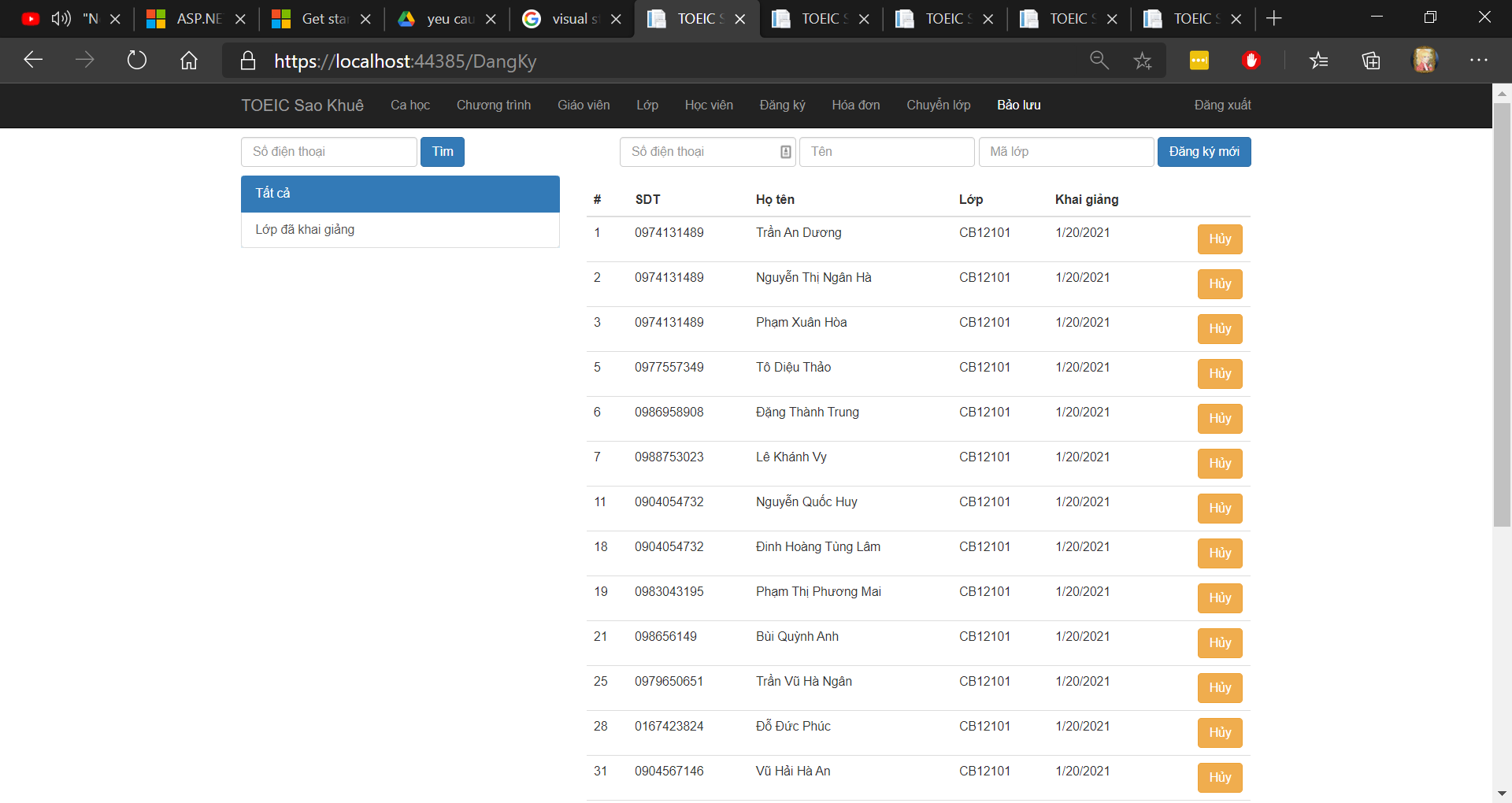
Hình 18 Giao diện Quản lý lớp



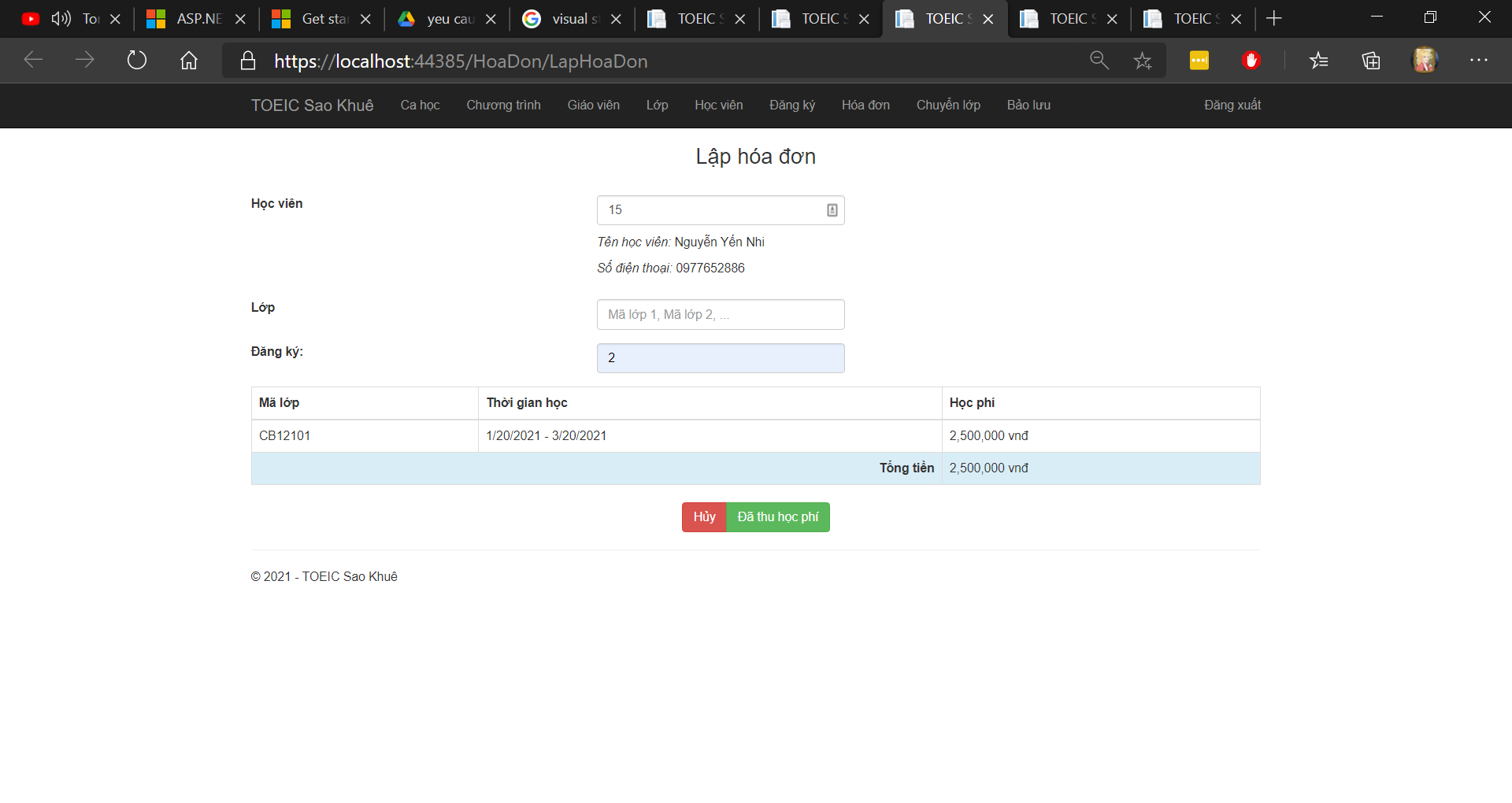
Hình 19 Giao diện Mở lớp



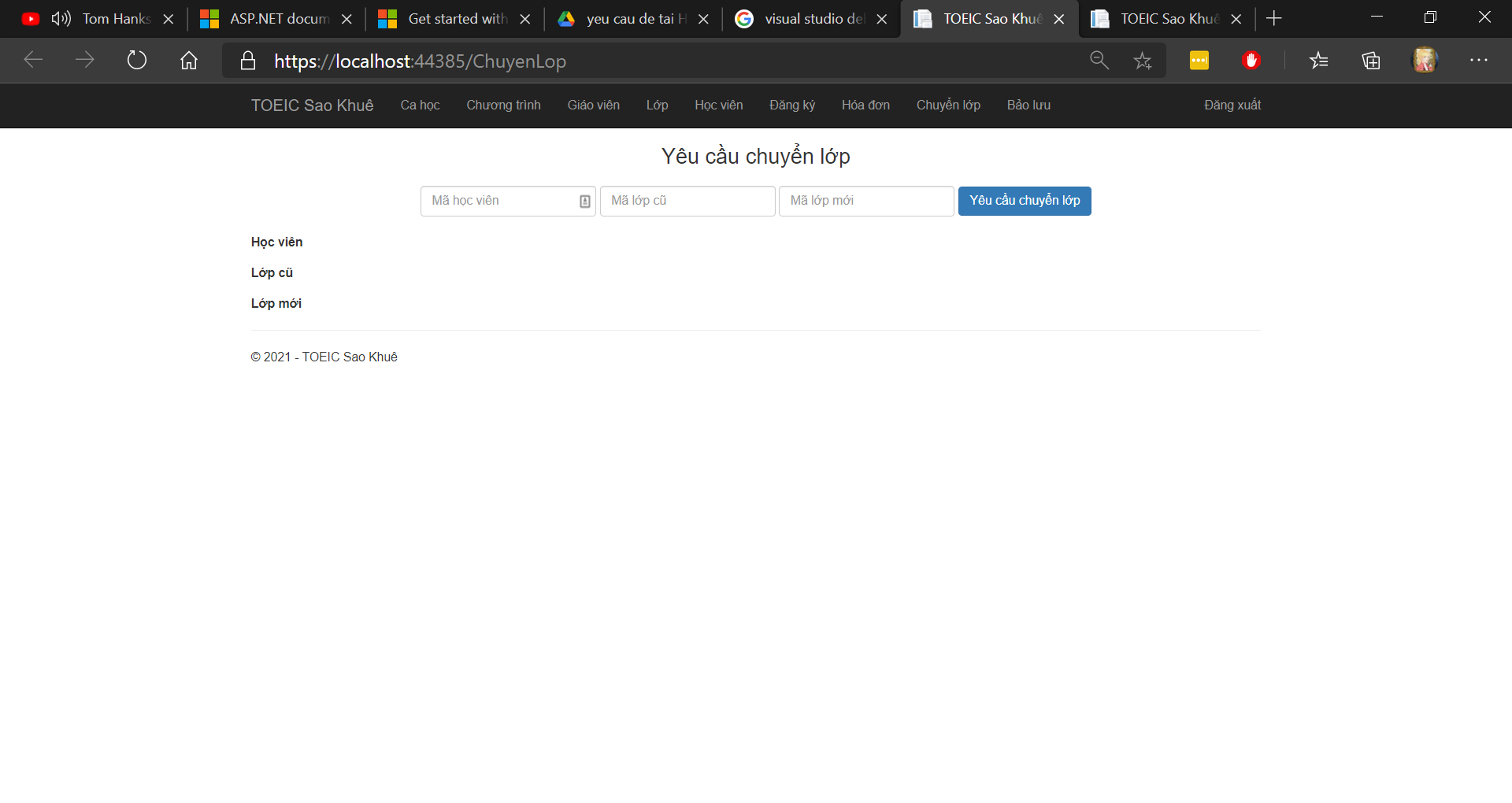
Hình 20 Giao diện Quản lý học viên



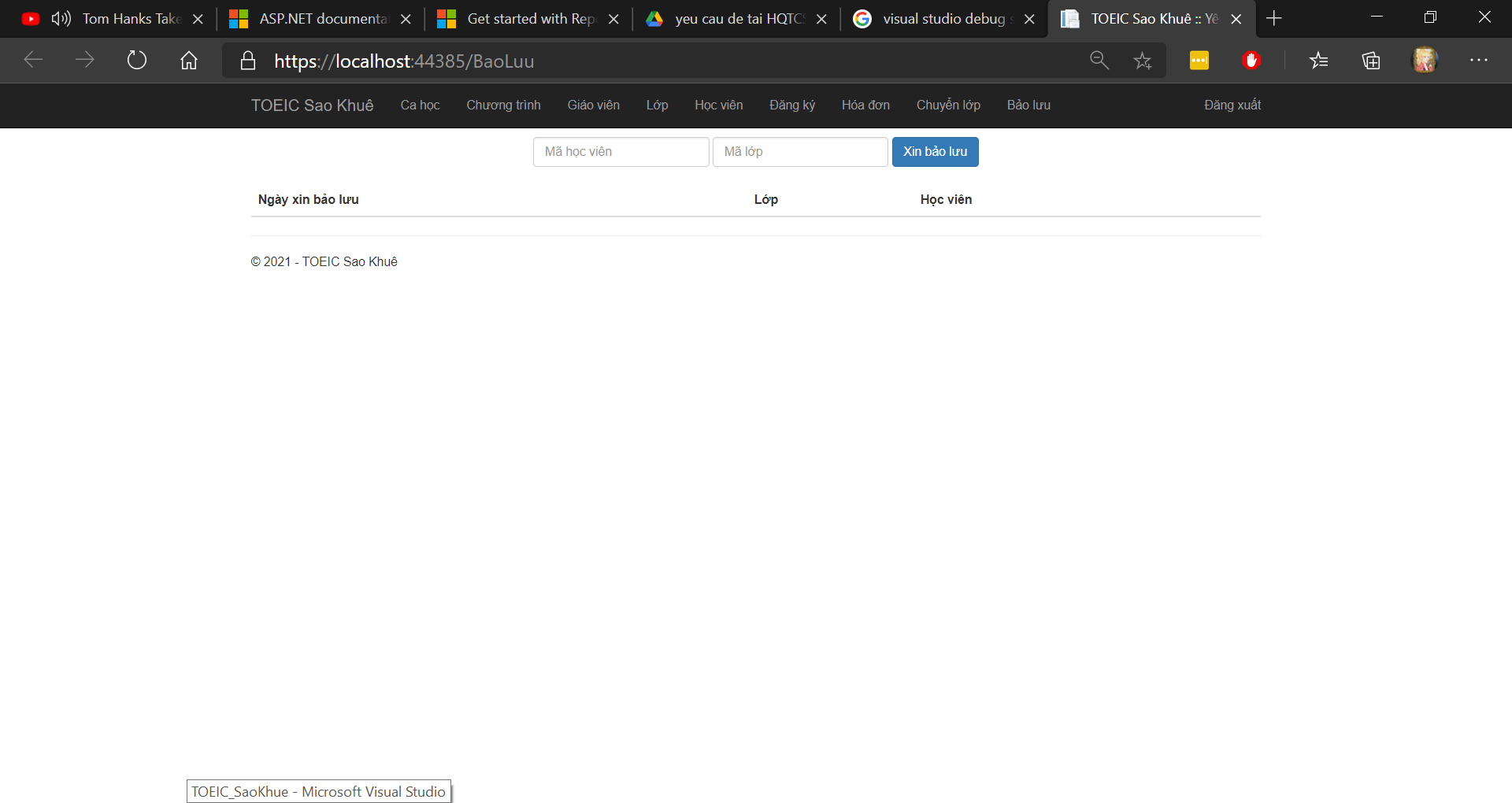
Hình 21 Giao diện Quản lý đăng ký



Hình 22 Giao diện Lập hóa đơn



Hình 23 Giao diện Chuyển lớp

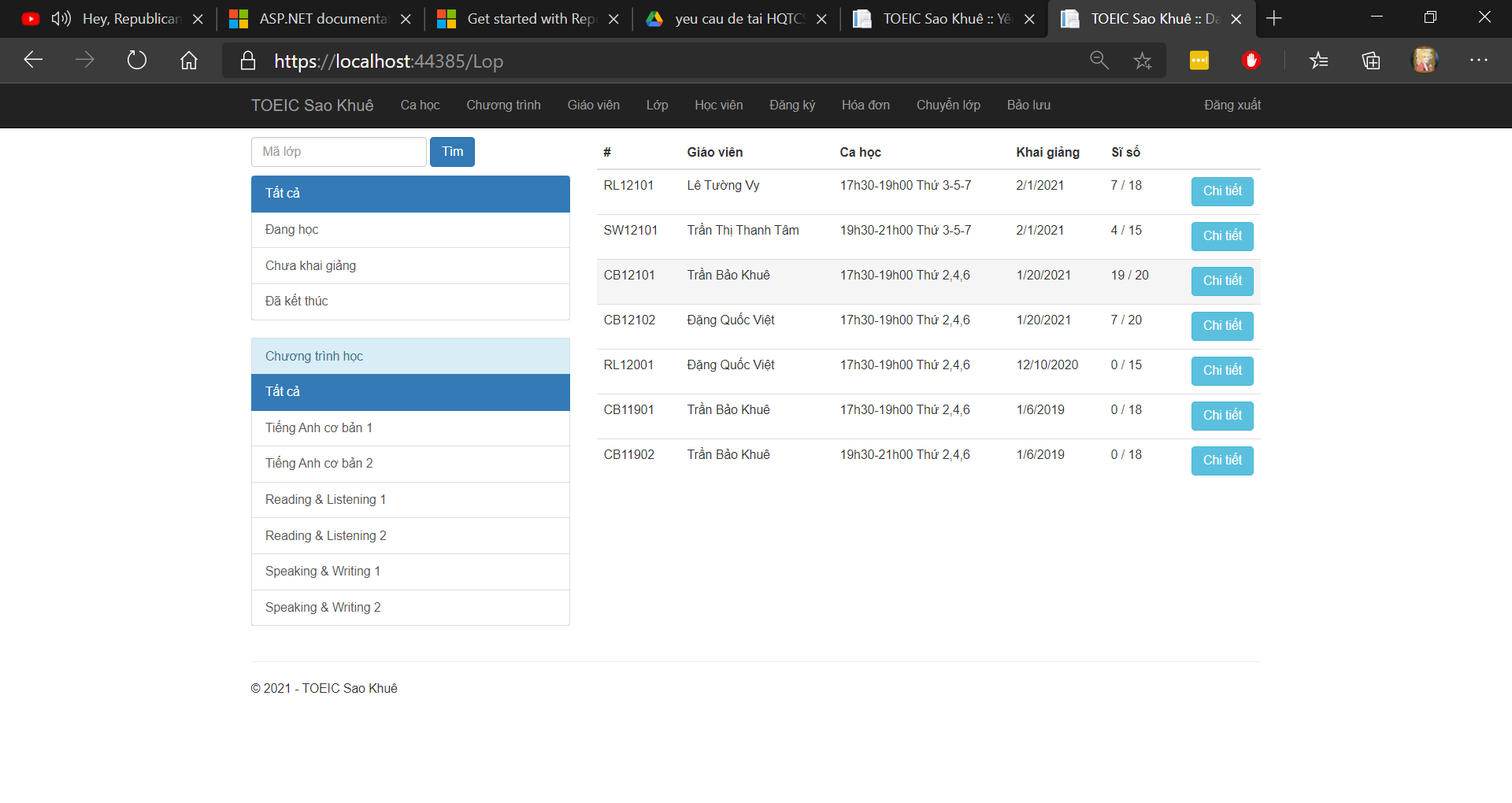


Hình 24 Giao diện Bảo lưu

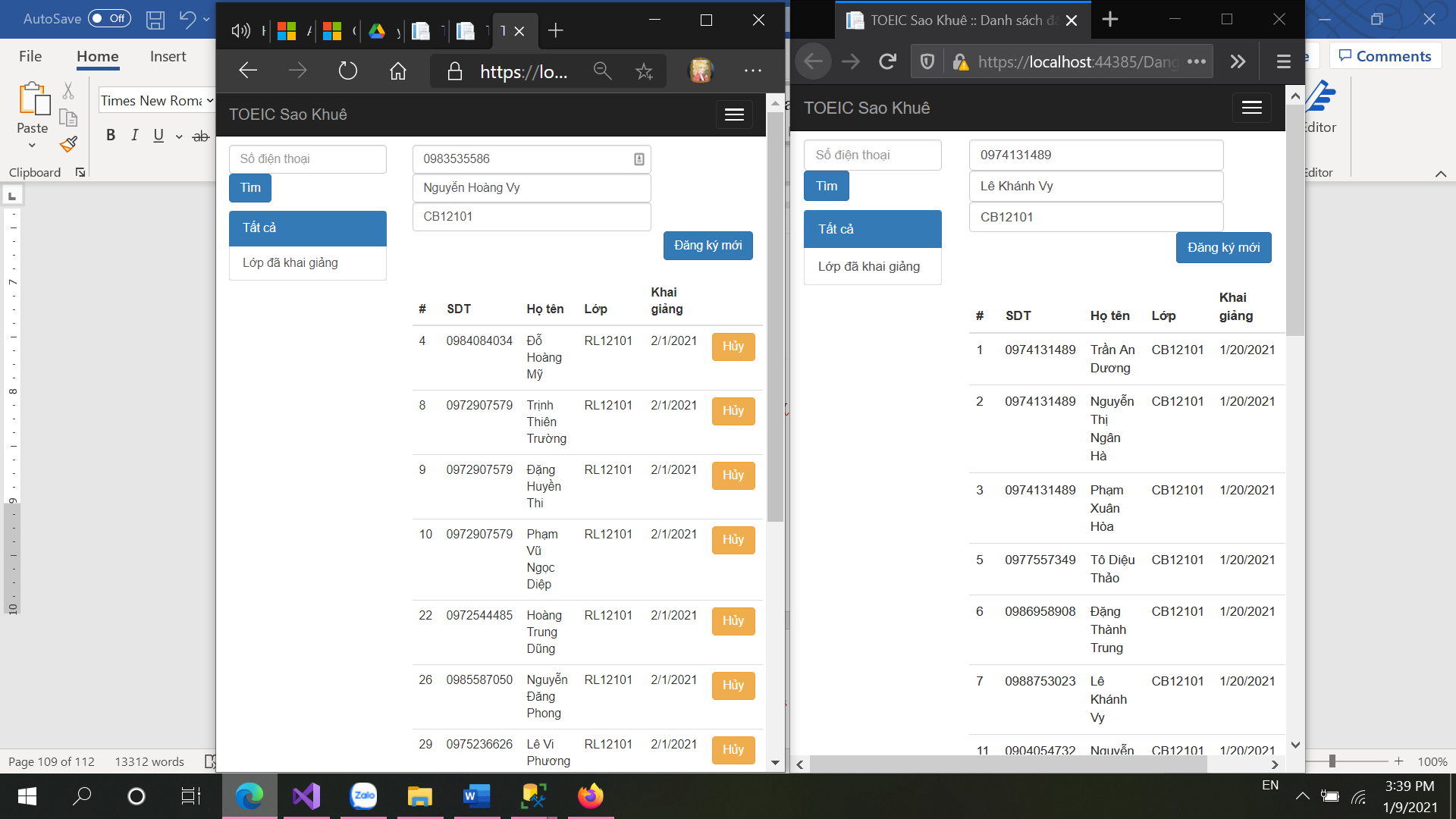
### Giả lập xử lý đồng thời

#### Đăng ký

Giả sử hai nhân viên tư vấn đồng thời đăng ký cho hai học viên khác nhau một lớp chỉ còn một suất.

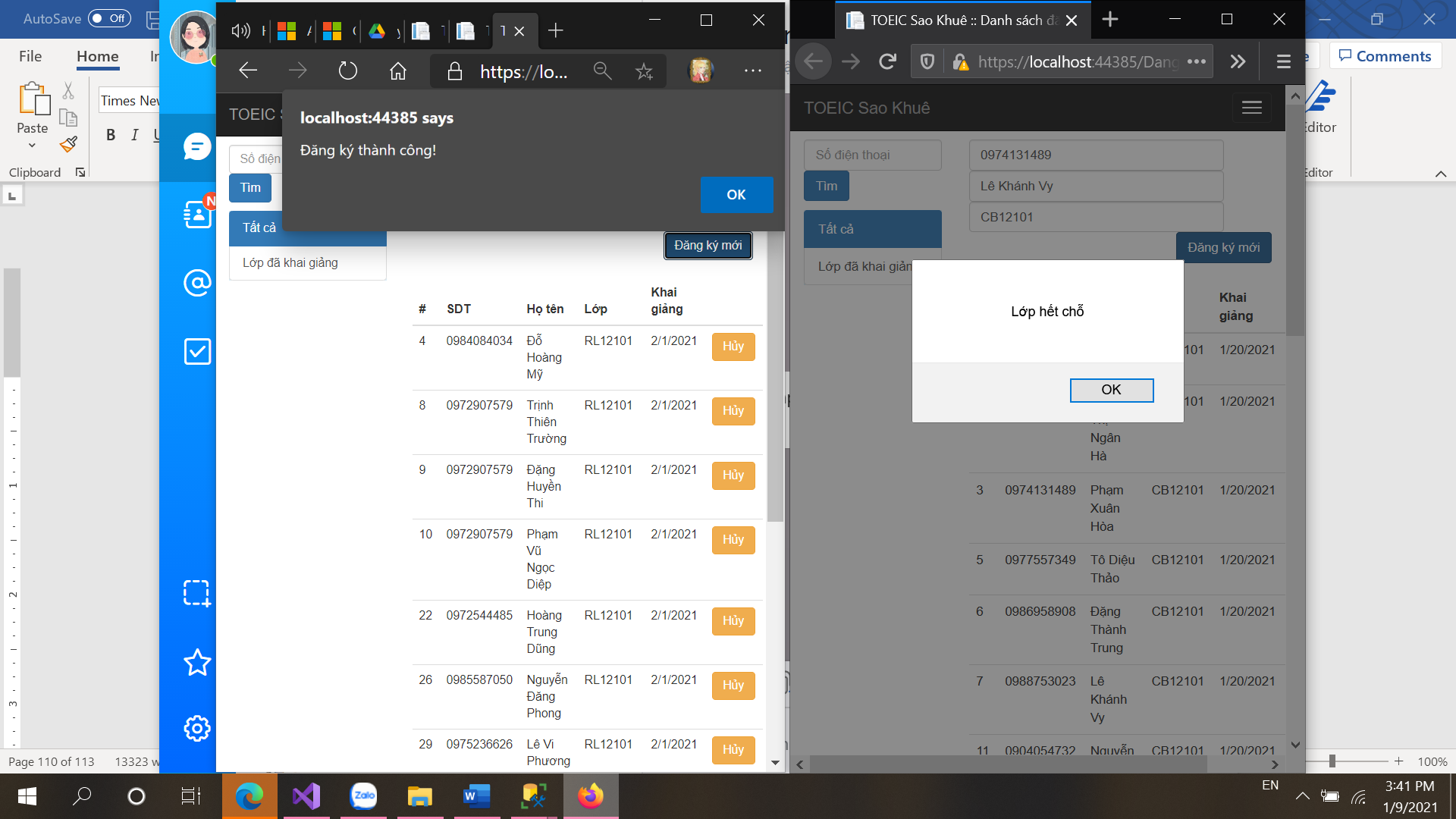


Hình 25 Giả lập xử lý đồng thời – Đăng ký (1)



Hình 26 Giả lập xử lý đồng thời – Đăng ký (2)

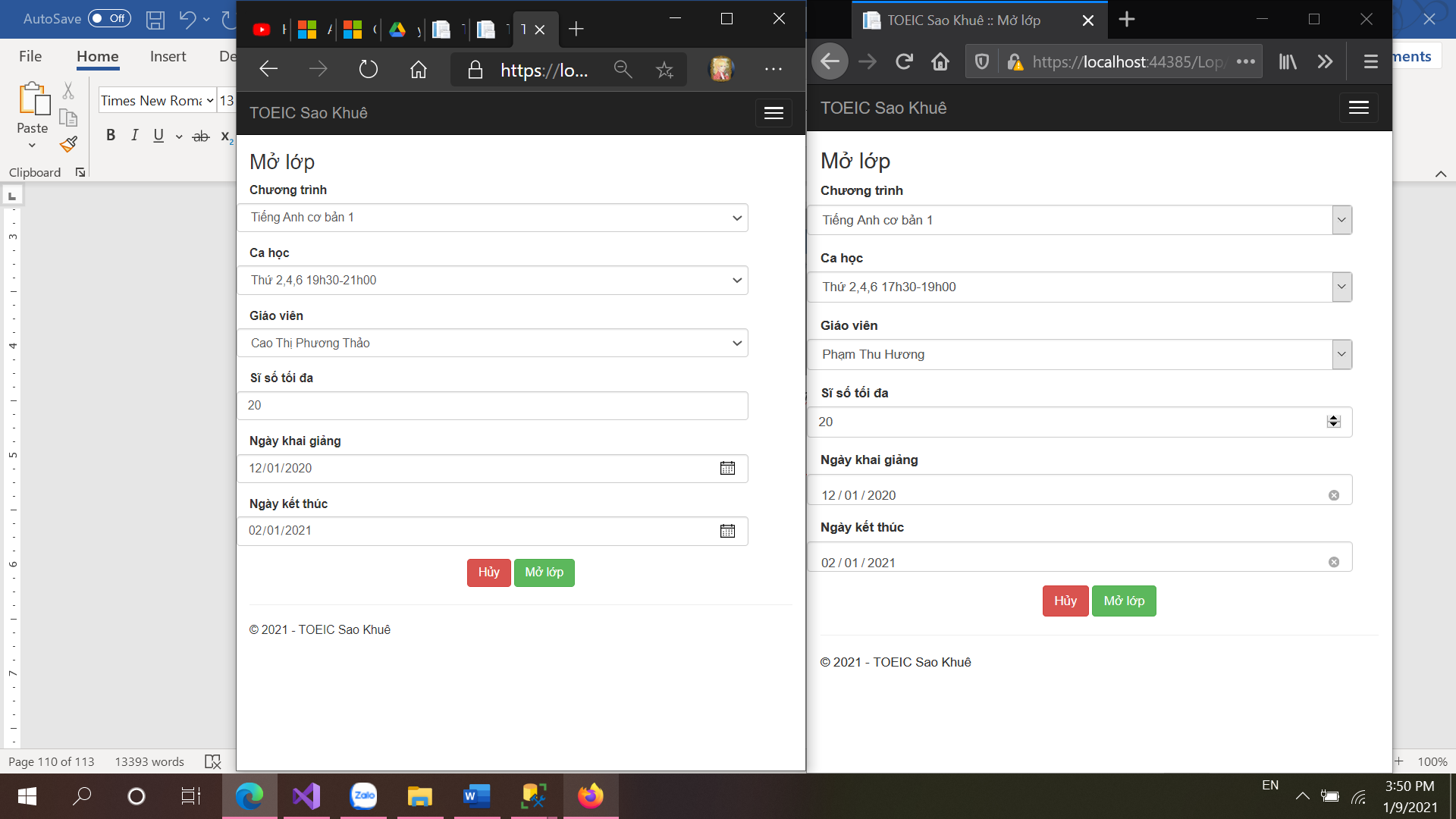
**Kết quả:** Nhân viên tư vấn lưu thông tin đăng ký trước sẽ lưu đăng ký thành công còn nhân viên tư vấn lưu thông tin đăng ký sau sẽ nhận thông báo lỗi.



Hình 27 Giả lập xử lý đồng thời - Đăng ký (3)

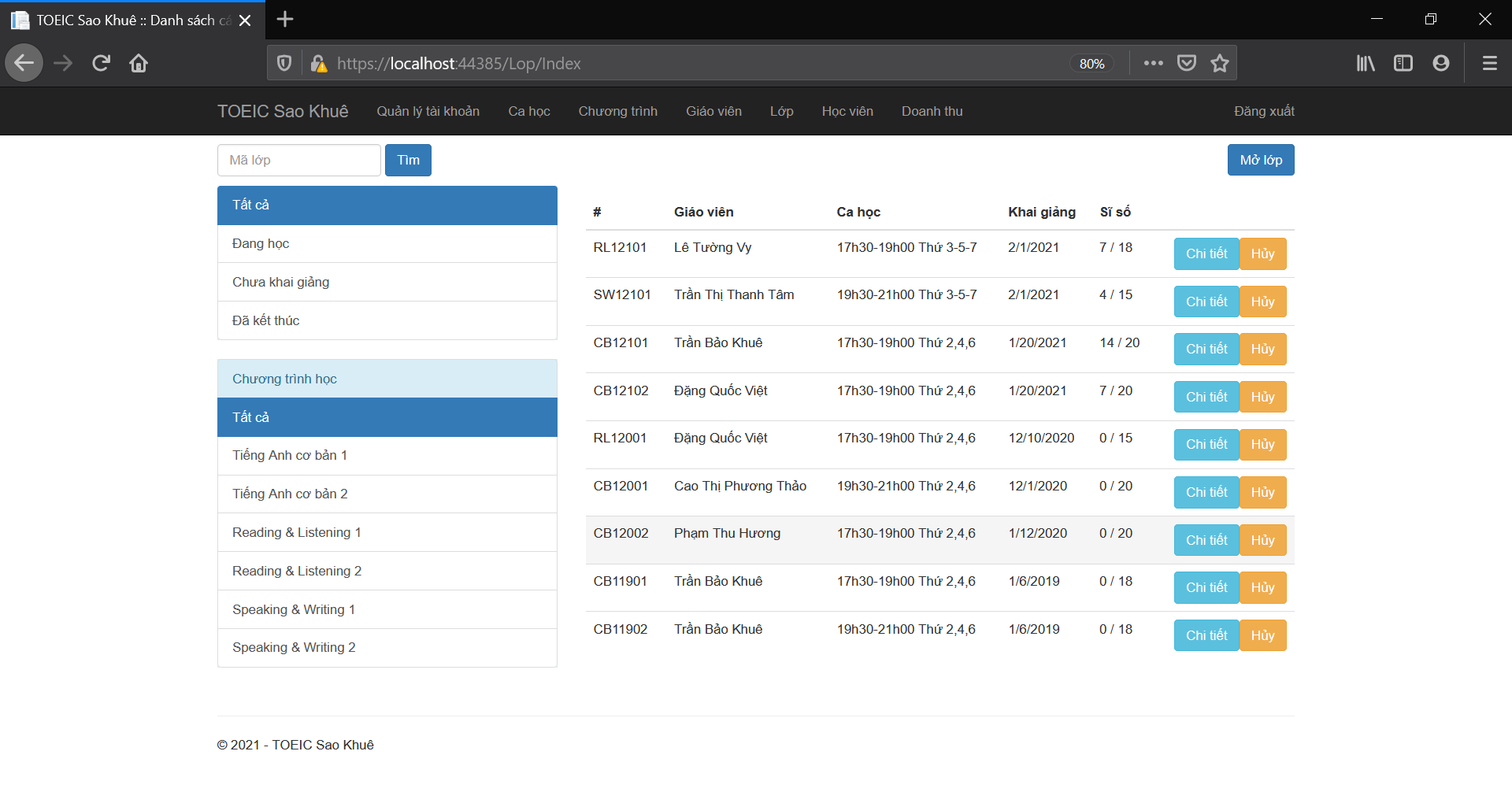
#### Mở lớp

Giả sử hai nhân viên quản lý đồng thời mở lớp cho cùng một chương trình học, cùng năm khai giảng.



Hình 28 Giả lập xử lý đồng thời - Mở lớp (1)

**Kết quả:** Cả hai lớp được mở thành công với mã lớp đúng.



Hình 29 Giả lập xử lý đồng thời - Mở lớp (2)

# KẾT LUẬN

## Kết quả đạt được

### Phía server

* Xây dựng cấu trúc dữ liệu phù hợp với yêu cầu của người dùng.
* Xây dựng các ràng buộc toàn vẹn nhằm giảm thiểu sai sót nhập liệu của người dùng.
* Xây dựng các stored procedure và function đáp ứng yêu cầu truy vấn và giao tác của người dùng, đảm bảo việc xử lý giao tác được thực hiện nhanh và có thể xử lý đồng thời.
* Phân quyền trên database.
* Lập lịch backup tự động cho database.

### Phía client

Xây dựng trang web phục vụ đầy đủ các chức năng tùy loại người dùng yêu cầu.

## Kết quả chưa đạt được

* Mã hóa dữ liệu nhằm tăng tính bảo mật.
* Giao diện chưa thực sự bắt mắt.

## Hướng phát triển trong tương lai

* Mã hóa dữ liệu để tăng tính bảo mật.
* Thiết kế giao diện bắt mắt hơn.
* Cho phép học viên đăng ký lớp online.

# TÀI LIỆU THAM KHẢO

1. Microsoft, Microsoft SQL documentation   
   https://docs.microsoft.com/vi-vn/sql/?view=sql-server-ver15
2. Microsoft, ASP.NET documentation   
   https://docs.microsoft.com/vi-vn/aspnet/core/?view=aspnetcore-5.0
3. Stack Exchange, stackoverflow.com  
   https://stackoverflow.com/

# BẢNG PHÂN CÔNG

|  |  |  |
| --- | --- | --- |
|  | **Nguyễn Sang Hoàng Kim** | **Phạm Thị Quỳnh Giang** |
| **Khảo sát nghiệp vụ** | X | X |
| **Xác định các yêu cầu cho đề tài** | X |  |
| **Thiết kế lược đồ Use case** |  | X |
| **Phân tích dữ liệu** | X |  |
| **Phân tích ràng buộc toàn vẹn** | X | X |
| **Thiết kế các bước thực hiện giao tác** | X |  |
| **Thiết kế phân quyền** | X | X |
| **Xây dựng lịch backup** |  | X |
| **Xây dựng các ràng buộc toàn vẹn và các giao tác trên server** | X | X |
| **Phân quyền trên server** | X |  |
| **Lập lịch backup tự động** |  | X |
| **Xây dựng các chức năng và giao diện cho client** | X |  |
| **Test hệ thống** | X | X |
| **Viết báo cáo** | X | X |