****

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH  
KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**---------------🙦

🕮🙤---------------

***Đề tài:***

**BÁO CÁO ĐỒ ÁN CNTT**

**Website quản lý chương trình đào tạo**

**Khoa Công Nghệ Thông Tin**

**GVHD: HUỲNH XUÂN PHỤNG**

**Thực hiện:**

**Lê Hoàng Nam 18110160**

**Trần Hữu Hào 18110103**

*TP. Hồ Chí Minh, ngày 18 , tháng 12, năm 2020*

**MỤC LỤC**

[**DANH MỤC BẢNG** 2](#_Toc59213217)

[**DANH MỤC HÌNH ẢNH** 3](#_Toc59213218)

[**I.** **ĐẶC TẢ** 4](#_Toc59213219)

[1. Website dùng để làm gì? 4](#_Toc59213220)

[2. Dữ liệu, thông tin đầu vào 4](#_Toc59213221)

[3. Cac tình huống sử dụng 4](#_Toc59213222)

[3.1. Mục đích 4](#_Toc59213223)

[3.2. Tính năng 4](#_Toc59213224)

[4. Giao diện dự kiến 4](#_Toc59213225)

[5. Các công nghệ/framework được sử dụng 7](#_Toc59213226)

[5.1. PostgreSQL 8](#_Toc59213227)

[5.2. React 8](#_Toc59213228)

[5.3. NodeJS/Express 8](#_Toc59213229)

[5.4. JWT(JSON Web Token) 9](#_Toc59213230)

[**II.** **PHÂN CÔNG CÔNG VIỆC** 9](#_Toc59213231)

[**III.** **THIẾT KẾ** 10](#_Toc59213232)

[1. Thiết kế model (NodeJS) 10](#_Toc59213233)

[2. Thiết kế Component (ReactJS) 13](#_Toc59213234)

[3. Thiết kế cơ sở dự liệu 16](#_Toc59213235)

[4. Tổng các bảng mô tả các Field trong 1 Table 17](#_Toc59213236)

[5. Thiết kế giao diện 19](#_Toc59213237)

[**IV.** **CÀI ĐẶT VÀ KIỂM THỬ** 20](#_Toc59213238)

[**V.** **KẾT LUẬN** 21](#_Toc59213239)

[1. Kết quả đạt được 21](#_Toc59213240)

[2. Hạn chế 22](#_Toc59213241)

[3. Ý tưởng phát triển 22](#_Toc59213242)

[**TÀI LIỆU THAM KHẢO** 23](#_Toc59213243)

# **DANH MỤC BẢNG**

[**Bảng 1** Phân công công việc 6](#_Toc59209774)

[**Bảng 2** Các thiết kế model 7](#_Toc59209775)

[**Bảng 3** Mô tả các phương thức trong model 9](#_Toc59209776)

[**Bảng 4** Các thiết kế Component 9](#_Toc59209777)

[**Bảng 5** Mô tả các phương thức thực hiện trong một Component 12](#_Toc59209778)

[**Bảng 6** Mô tả các Table trong CSDL 13](#_Toc59209779)

[**Bảng 7** Mô tả các Field trong table bomon 13](#_Toc59209780)

[**Bảng 8** Mô tả các Field trong table Chuandaura 14](#_Toc59209781)

[**Bảng 9** Mô tả các Field trong table danhgia 14](#_Toc59209782)

[**Bảng 10** Mô tả các Field trong table danhgia\_chuandaura 14](#_Toc59209783)

[**Bảng 11** Mô tả các Field trong table bomon 15](#_Toc59209784)

[**Bảng 12** Mô tả các Field trong table muctieu 15](#_Toc59209785)

[**Bảng 13** Mô tả các Field trong table ndchitiet\_cdr 15](#_Toc59209786)

[**Bảng 14** Mô tả các Field trong table noidungchitiet 16](#_Toc59209787)

[**Bảng 15** Các phần giao diện trong website 17](#_Toc59209788)

[**Bảng 16** Các tình huống kiểm thử 18](#_Toc59209789)

# **DANH MỤC HÌNH ẢNH**

[**Hình 1** Giao diện trang đăng nhập 6](#_Toc59213085)

[**Hình 2** Giao diện trang chủ 6](#_Toc59213086)

[**Hình 3** Giao diện trang tạo môn học 6](#_Toc59213087)

[**Hình 4** Giao diện danh sách môn học 7](#_Toc59213088)

[**Hình 5** Giao diện trang thông tin môn học 7](#_Toc59213089)

[**Hình 6** Giao diện form nhập liệu 8](#_Toc59213090)

[**Hình 7** Hình minh họa class diagram 17](#_Toc59213091)

1. **ĐẶC TẢ**
2. Website dùng để làm gì?

Website được xây dựng nhằm mục đích tổng hợp và quản lý các môn học trong chương trình đào tạo của khóa CNTT.

1. Dữ liệu, thông tin đầu vào

Dữ liệu đầu vào bao gồm thông tin của 1 học phần như sau:

* Môn học (Mã môn học, tên môn học, số tín chỉ,…)
* Mô tả môn học
* Mục tiêu môn học
* Chuẩn đầu ra của môn học

1. Cac tình huống sử dụng
   1. Mục đích

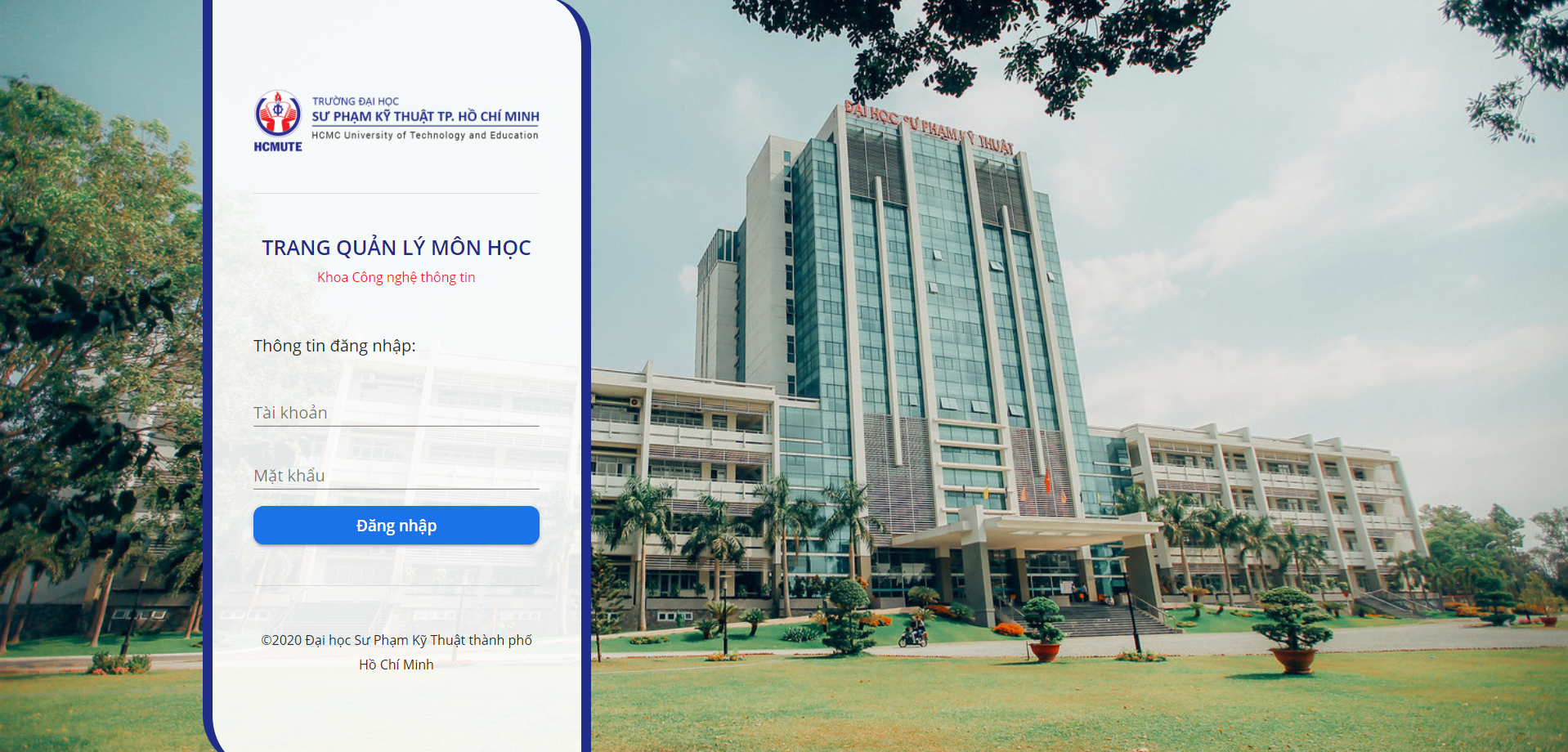
Giúp cho sinh viên, giảng viên lẫn những người làm việc trong công tác quản lý có thể đáp ứng nhu cầu học tập – giảng dạy –quản lý mọi lúc mọi nơi với nhiều chức năng.

* 1. Tính năng

Website quản lý sẽ có các tính năng thêm, xóa , sửa nội dung của một học phần nằm trong chương trình đào tạo cũng như nội dung chi tiết của học phần đó.

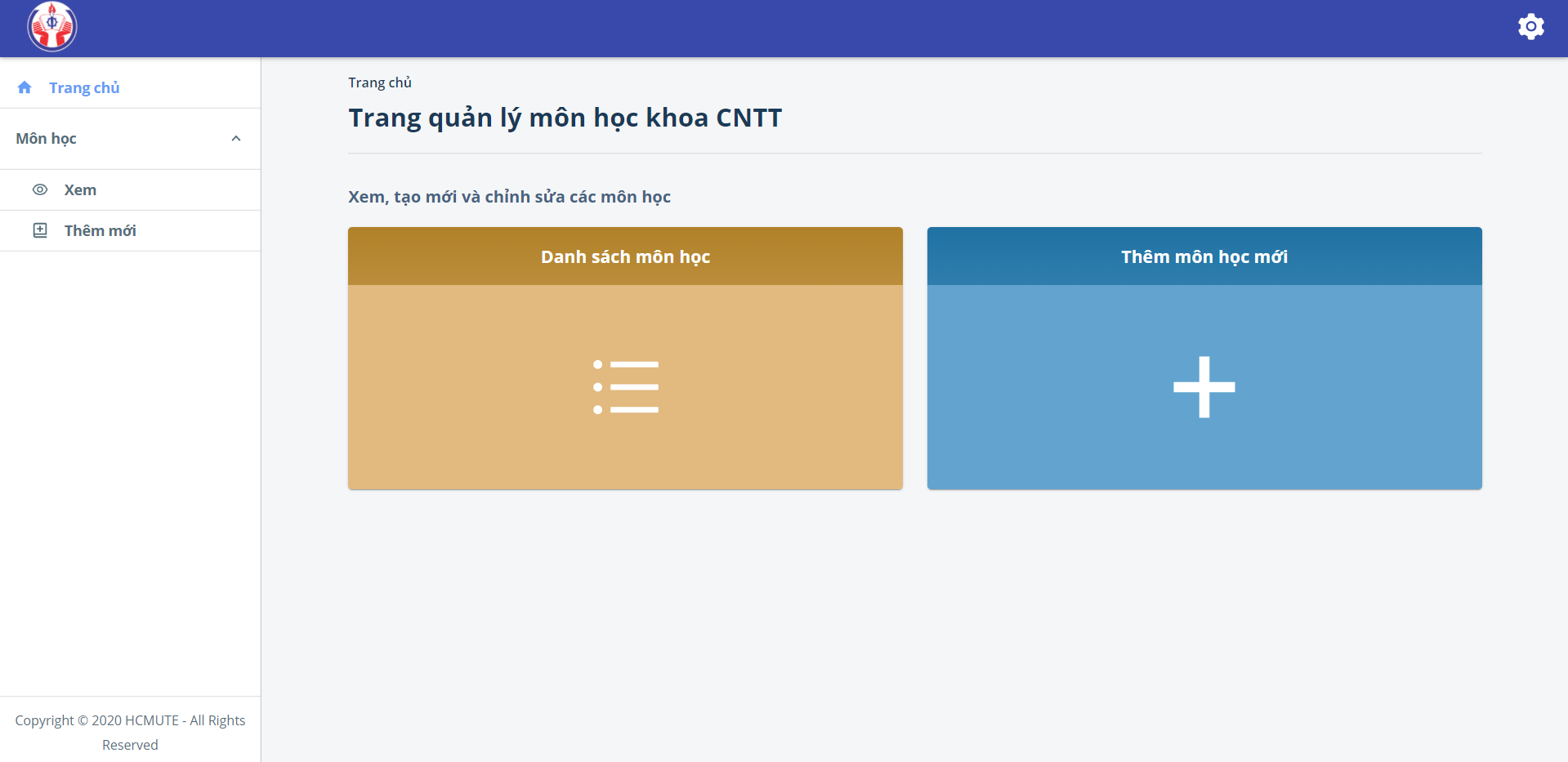
1. Giao diện dự kiến

* Trang đăng nhập



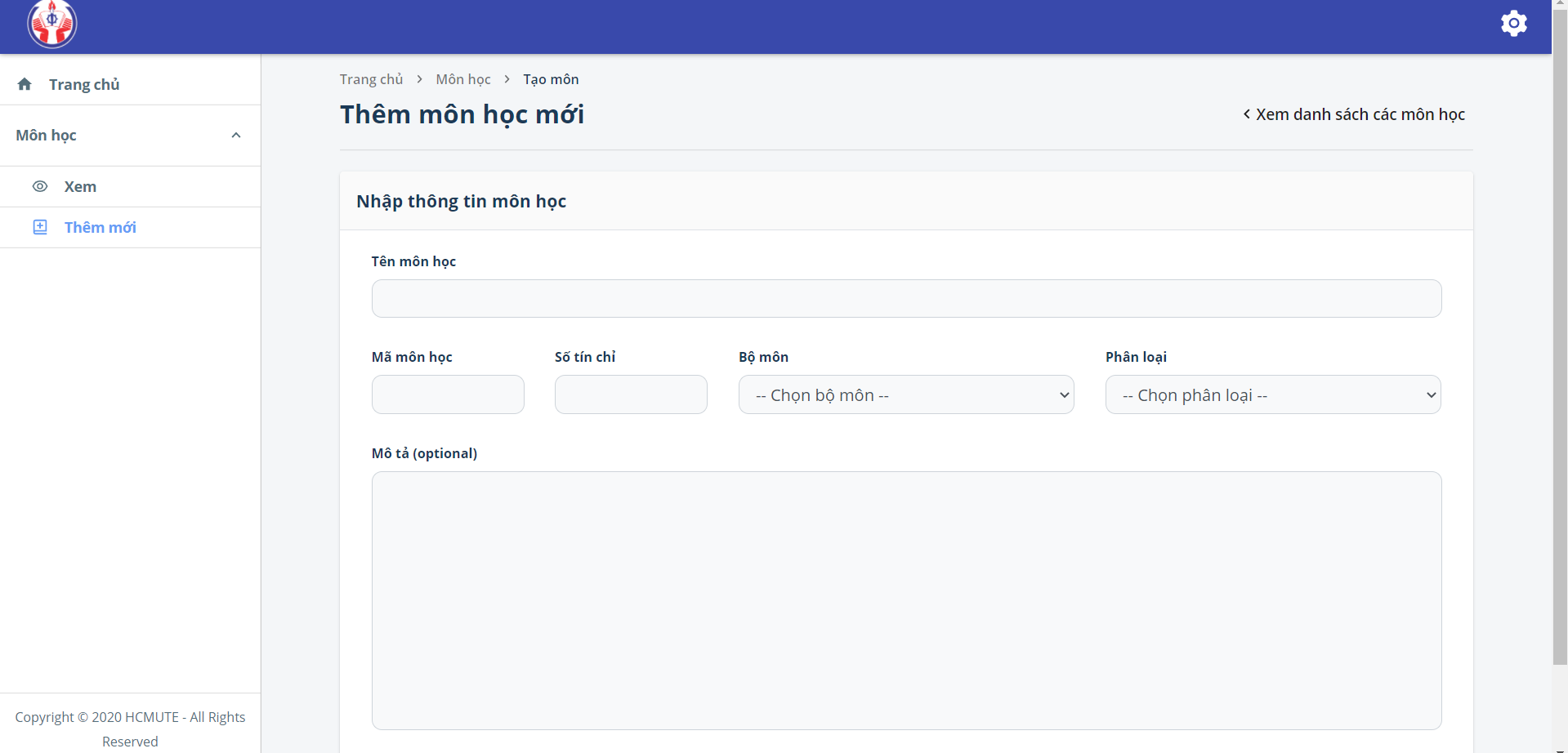
**Hình 1** Giao diện trang đăng nhập

* Trang chủ



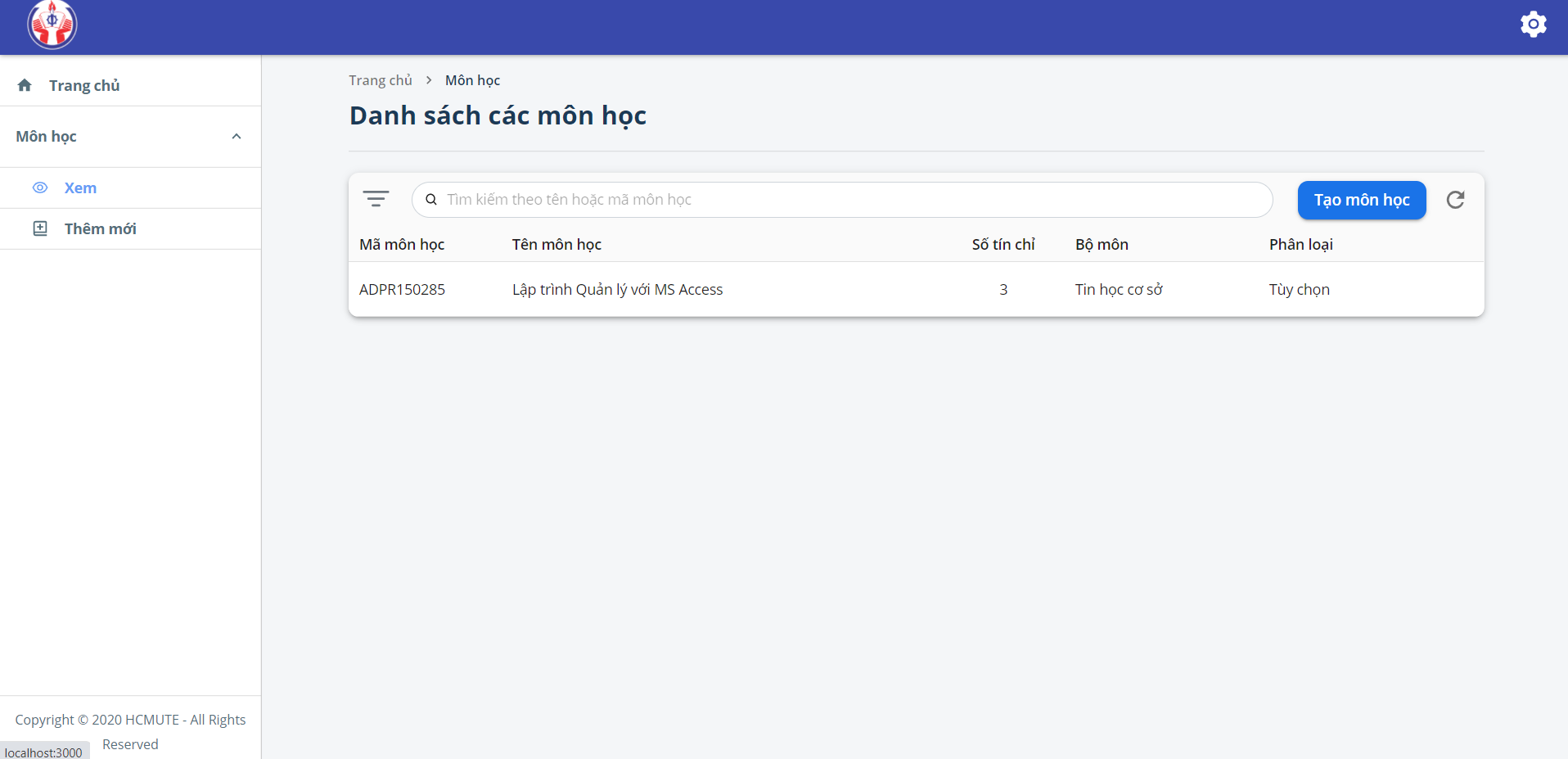
**Hình 2** Giao diện trang chủ

* Trang tạo môn học



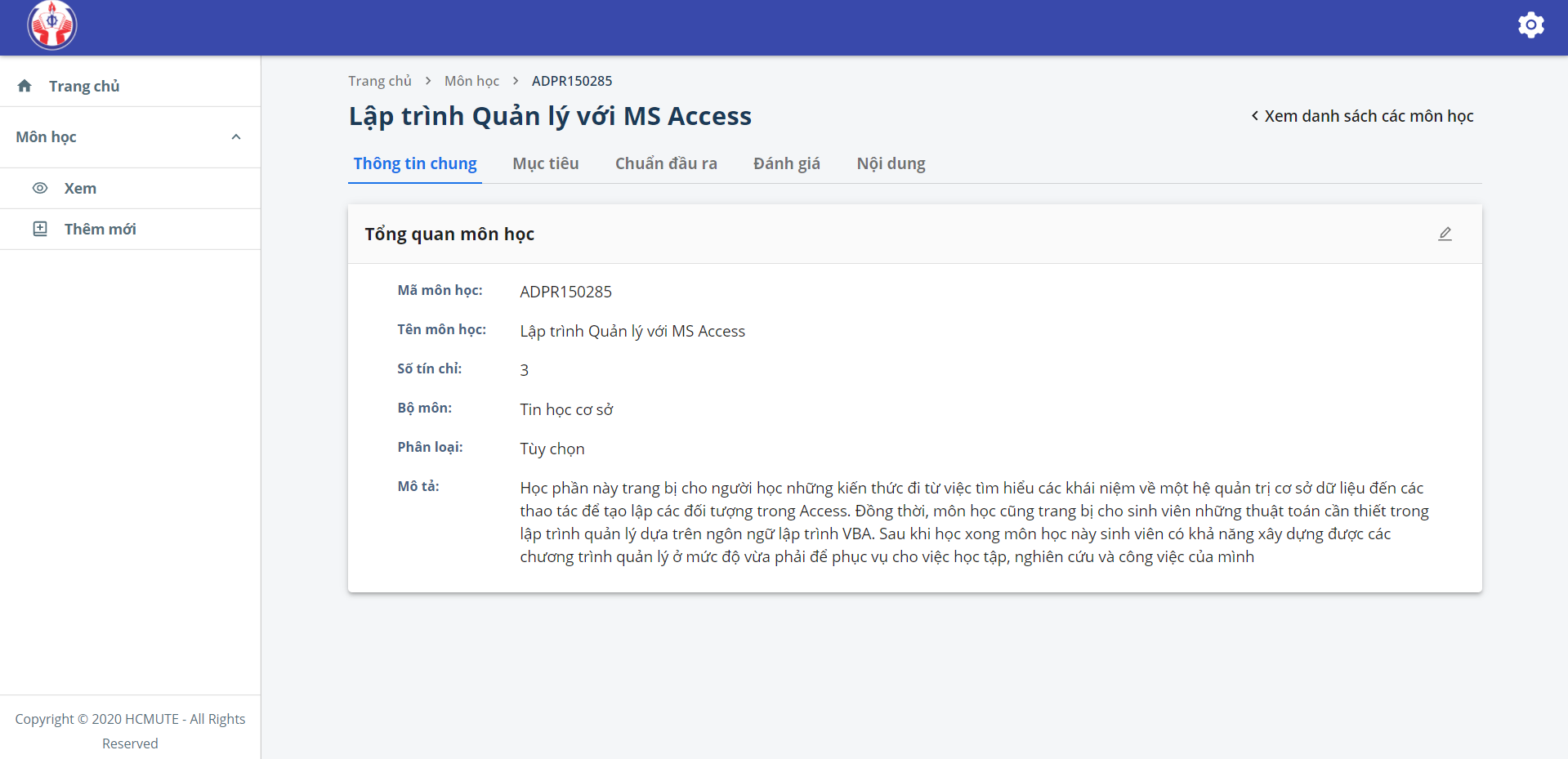
**Hình 3** Giao diện trang tạo môn học

* Danh sách môn học



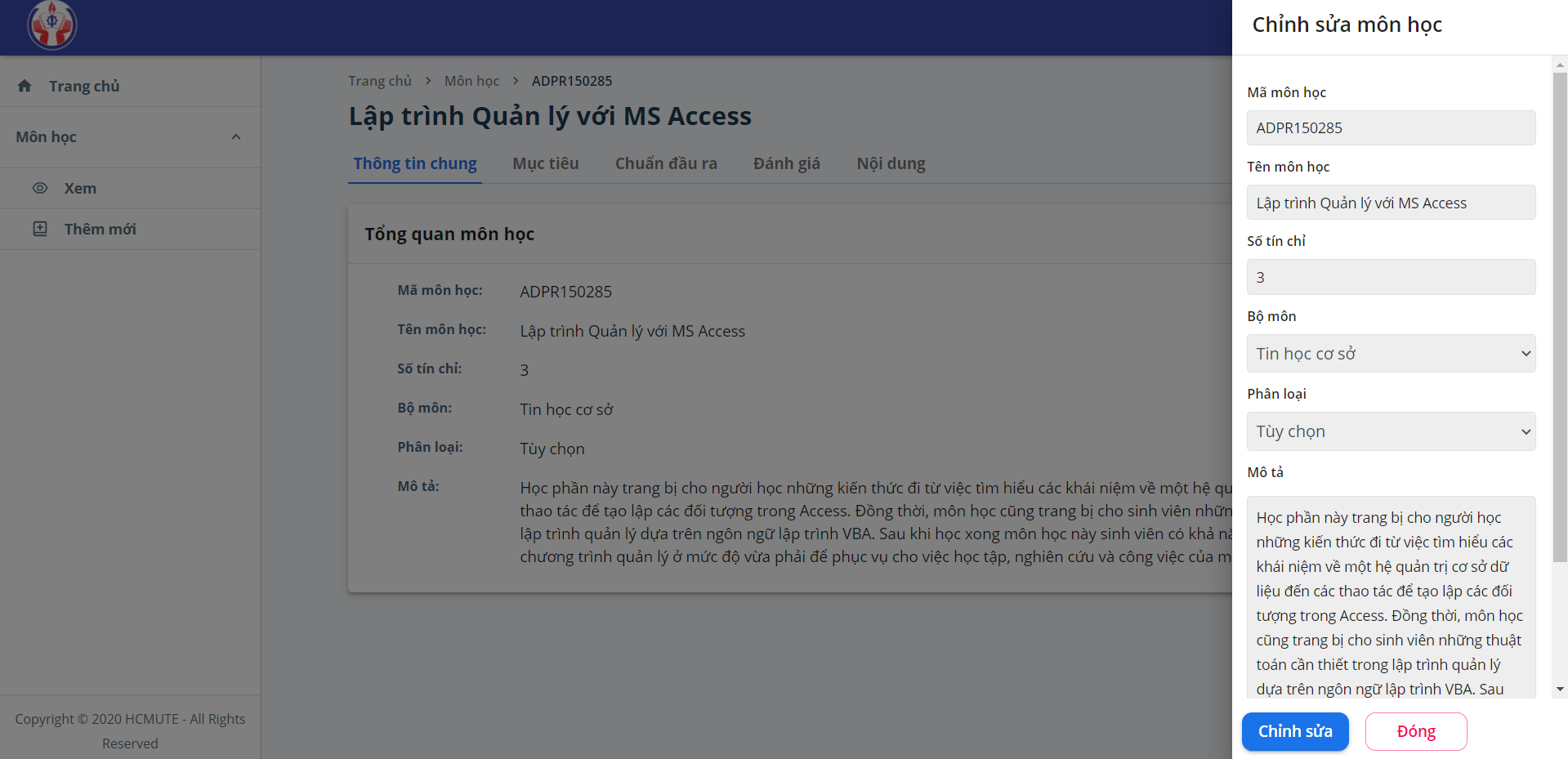
**Hình 4** Giao diện danh sách môn học

* Thông tin môn học



**Hình 5** Giao diện trang thông tin môn học

* Form thông tin môn học



**Hình 6** Giao diện form nhập liệu

1. Các công nghệ/framework được sử dụng

Trang web sử dụng công nghệ PERN Stack, được viết tắt của PostgreSQL, Express, ReactJS và NodeJS. PERN stack giúp xây dựng các ứng dụng web có hiệu năng rất cao, xử lý đa tác vụ từ client đến Server side. Trong đó:

PostgreSQL: cơ sở dữ liệu quan hệ.

Express: Web Framework nhẹ và di động

React: Thư viện Javascript để xây dựng giao diện người dùng

NodeJS: môi trường thực thi Server - side Javascript

Ngoài ra trang web còn sử dụng phương thức bảo mật JWT(JSON Web Token) để xác thực người dùng.

* 1. PostgreSQL
* PostgreSQL là một hệ thống quản trị cơ sở dữ liệu quan hệ-đối tượng (object-relational database management system) có mục đích chung, hệ thống cơ sở dữ liệu mã nguồn mở tiên tiến nhất hiện nay.
* PostgreSQL sở hữu một hệ tính năng đa dạng giúp hỗ trợ các nhà phát triển xây dựng app, các nhà quản trị bảo vệ toàn vẹn dữ liệu, và tạo ra một môi trường chịu lỗi fault-tolerant giúp quản lý dữ liệu bất kể tập dữ liệu lớn hay nhỏ. Bên cạnh hệ thống nguồn mở và miễn phí, PostgreSQL cũng có khả năng mở rộng tuyệt vời.
  1. React
* React là thư viện JavaScript phổ biến nhất để xây dựng giao diện người dùng (UI). Nó cho tốc độ phản hồi tuyệt vời khi user nhập liệu bằng cách sử dụng phương pháp mới để render trang web.
* Components của công cụ này được phát triển bởi Facebook. Nó được ra mắt như một công cụ JavaScript mã nguồn mở vào năm 2013. Hiện tại, nó đã đi trước các đối thủ chính như Angular và Bootstrap, hai thư viện JavaScript bán chạy nhất thời bấy giờ.
  1. NodeJS/Express
* Nodejs là một nền tảng (Platform) phát triển độc lập được xây dựng ở trên Javascript Runtime của Chrome mà chúng ta có thể xây dựng được các ứng dụng mạng một cách nhanh chóng và dễ dàng mở rộng.
* Phần Core bên dưới của Nodejs được viết hầu hết bằng C++ nên cho tốc độ xử lý và hiệu năng khá cao.
* Nodejs tạo ra được các ứng dụng có tốc độ xử lý nhanh, realtime thời gian thực.
* Nodejs áp dụng cho các sản phẩm có lượng truy cập lớn, cần mở rộng nhanh, cần đổi mới công nghệ, hoặc tạo ra các dự án Startup nhanh nhất có thể.
* Express là một framework giành cho nodejs. Nó cung cấp cho chúng ta rất nhiều tính năng mạnh mẽ trên nền tảng web cũng như trên các ứng dụng di động. Express hỗ rợ các phương thức HTTP và midleware tạo ra môt API vô cùng mạnh mẽ và dễ sử dụng. Có thể tổng hợp một số chức năng chính của express như sau:
* Thiết lập các lớp trung gian để trả về các HTTP request
* Định nghĩa router cho phép sử dụng với các hành động khác nhau dựa trên phương thức HTTP và URL
* Cho phép trả về các trang HTML dựa vào các tham số.
* Trong ứng dụng web của nhóm, express được sử dụng để xây dựng RESTful API
  1. JWT(JSON Web Token)
* JWT là một chuẩn mở (RFC 7519) định nghĩa một cách nhỏ gọn và khép kín để truyền một cách an toàn thông tin giữa các bên dưới dạng đối tượng JSON. Thông tin này có thể được xác minh và đáng tin cậy vì nó có chứa chữ ký số. JWTs có thể được ký bằng một thuật toán bí mật (với thuật toán HMAC) hoặc một public / private key sử dụng mã hoá RSA.
* Sử dụng JWT là cách tốt để áp dụng cơ chế bảo mật đối với các dịch vụ API RESTful mà có thể được sử dụng để truy cập vào cơ sở dữ liệu.
* JWT là một phương tiện đại diện cho các yêu cầu chuyển giao giữa hai bên Client – Server, các thông tin trong chuỗi JWT được định dạng bằng JSON.
* JWT được dùng để xác thực trong mọi tác vụ của người dùng để đảm bảo tính bảo mật.

1. **PHÂN CÔNG CÔNG VIỆC**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Tên SV** | **Công việc thực hiện** | **Ước tính phầm trăm đóng góp** |
| 1 | Lê Hoàng Nam | * Thiết kế database, model. * Thiết kế một số component cho môn học * Viết báo cáo * … | 40% |
| 2 | Trần Hữu Hào | * Thiết kế giao diện * Thiết kế controller * Thiết kế một số component * Viết báo cáo * … | 60% |

**Bảng 1** Phân công công việc

1. **THIẾT KẾ**
2. Thiết kế model (NodeJS)

Sinh viên thực hiện: Lê Hoàng Nam

|  |  |  |
| --- | --- | --- |
| **TT** | **Tên model** | **Mục đích** |
| 1 | account | Đối tượng đại diện cho bảng account trong CSDL, chứa tài khoản website |
| 2 | bomon | Đối tượng đại diện cho bảng bomon trong CSDL, chứa thông tin bộ môn |
| 3 | monhoc | Đối tượng đại diện cho bảng monhoc trong CSDL, chứa thông tin môn học |
| 4 | chuandaura | Đối tượng đại diện cho bảng chuandaura trong CSDL, chứa thông tin chuẩn đầu ra |
| 5 | danhgia | Đối tượng đại diện cho bảng danhgia trong CSDL, chứa thông tin đánh giá |
| 6 | danhgia\_chuandaura | Đối tượng đại diện cho bảng danhgia\_chuandaura trong CSDL, chứa các chuẩn đầu ra của đánh giá |
| 7 | noidungchitiet | Đối tượng đại diện cho bảng noidungchitiet trong CSDL, chứa thông tin nội dung |
| 8 | ndchitiet\_cdr | Đối tượng đại diện cho bảng ndchitiet\_cdr trong CSDL, chứa các chuẩn đầu ra của nội dung |

**Bảng 2** Các thiết kế model

Sinh viên thực hiện: Trần Hữu Hào

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Phương thức** | **Mục đích** | **Tên file, số thứ tự dòng chứa khai báo** |
| 1 | login(req, res)  Phương thức POST cho route /api/login | Đăng nhập bằng tài khoản và mật khẩu, nếu thành công máy chủ sẽ gửi về cookie chứa token cho client | account.controller.js (4) |
| 2 | logout(req, res)  Phương thức GET cho route /api/logout | Đăng xuất tài khoản và xóa token trên máy client | account.controller.js (32) |
| 3 | verify(req, res)  Phương thức GET cho route /api/verify | Đăng nhập bằng token | account.controller.js (36) |
| 4 | verify(req, res, next)  Middleware cho tất cả các HTTP request methods đến máy chủ | Middleware xác thực token mỗi khi client gửi yêu cầu đến máy chủ | auth.js (3) |
| 5 | read(req, res)  Phương thức GET cho route /api/bomon | Lấy danh sách các bộ môn | bomon.controller.js (6) |
| 6 | update(req, res)  Phương thức PUT cho route /api/bomon/:mabomon | Chỉnh sửa 1 bộ môn | bomon.controller.js (17) |
| 7 | read(req, res)  Phương thức GET cho route /api/monhoc | Lấy danh sách các môn học | monhoc.controller.js (7) |
| 8 | readOne(req, res)  Phương thức GET cho route /api/monhoc/:mamh | Lấy thông tin cụ thể 1 môn học | monhoc.controller.js (29) |
| 9 | create(req, res)  Phương thức POST cho route /api/monhoc | Tạo 1 môn học mới | monhoc.controller.js (47) |
| 10 | update(req, res)  Phương thức PUT cho route /api/monhoc/:mamh | Chỉnh sửa 1 môn học | monhoc.controller.js (61) |
| 11 | delete(req, res)  Phương thức DELETE cho route /api/monhoc/:mamh | Xóa 1 môn học | monhoc.controller.js (82) |
| 12 | readAll(req, res)  Phương thức GET cho route /api/monhoc/:mamh/muctieu | Lấy danh sách thông tin các mục tiêu của môn học | muctieu.controller.js (7) |
| 13 | readList(req, res)  Phương thức GET cho route /api/monhoc/:mamh/muctieu/list | Lấy danh sách mã mục tiêu của môn học | muctieu.controller.js (27) |
| 14 | create(req, res)  Phương thức POST cho route /api/monhoc/:mamh/muctieu | Thêm mục tiêu | muctieu.controller.js (45) |
| 15 | update(req, res)  Phương thức POST cho route /api/monhoc/:mamh/muctieu/:muctieu | Chỉnh sửa mục tiêu | muctieu.controller.js (57) |
| 16 | delete(req, res)  Phương thức DELETE cho route /api/monhoc/:mamh/muctieu/:muctieu | Xóa mục tiêu | muctieu.controller.js (73) |
| 17 | read(req, res)  Phương thức GET cho route /api/monhoc/:mamh/chuandaura | Lấy danh sách thông tin các chuẩn đầu ra của môn học | chuandaura.controller.js (7) |
| 18 | readList(req, res)  Phương thức GET cho route /api/monhoc/:mamh/chuandaura/list | Lấy danh sách mã chuẩn đầu ra của môn học | chuandaura.controller.js (37) |
| 19 | create(req, res)  Phương thức POST cho route /api/monhoc/:mamh/chuandaura/:muctieu | Thêm chuẩn đầu ra | chuandaura.controller.js (57) |
| 20 | update(req, res)  Phương thức PUT cho route /api/monhoc/:mamh/chuandaura/:muctieu/:cdr | Chỉnh sửa chuẩn đầu ra | chuandaura.controller.js (70) |
| 21 | delete(req, res)  Phương thức DELETE cho route /api/monhoc/:mamh/chuandaura/:muctieu/:cdr | Xóa chuẩn đầu ra | chuandaura.controller.js (89) |
| 22 | read(req, res)  Phương thức GET cho route /api/monhoc/:mamh/danhgia | Lấy danh sách đánh giá của môn học | danhgia.controller.js (9) |
| 23 | create(req, res)  Phương thức POST cho route /api/monhoc/:mamh/danhgia | Tạo đánh giá mới | danhgia.controller.js (45) |
| 24 | update(req, res)  Phương thức PUT cho route /api/monhoc/:mamh/danhgia/:hinhthuc | Chỉnh sửa đánh giá | danhgia.controller.js (73) |
| 25 | delete(req, res)  Phương thức DELETE cho route /api/monhoc/danhgia/:hinhthuc | Xóa đánh giá | danhgia.controller.js (115) |
| 26 | read(req, res)  Phương thức GET cho route /api/monhoc/:mamh/noidung | Lấy danh sách nội dung của môn học | ndchitiet.controller.js (9) |
| 27 | create(req, res)  Phương thức POST cho route /api/monhoc/:mamh/noidung | Tạo nội dung mới | ndchitiet.controller.js (34) |
| 28 | update(req, res)  Phương thức PUT cho route /api/monhoc/:mamh/noidung | Chỉnh sửa nội dung | ndchitiet.controller.js (61) |
| 29 | delete(req, res)  Phương thức POST cho route /api/monhoc/:mamh/noidung/delete | Xóa nội dung | ndchitiet.controller.js (101) |

**Bảng 3** Mô tả các phương thức trong model

1. Thiết kế Component (ReactJS)

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Tên Component** | **Mục đích** | **Sinh viên phụ trách viết** |
| 1 | Appbar | Hiện thị các chức năng của website | Trần Hữu Hào |
| 2 | CourseContent | Nhập thông tin chi tiết của môn học và có xóa, sửa tùy ý | Lê Hoàng Nam |
| 3 | CourseEvualate | Nhập thông tin đánh giá cho môn học và có thể xóa, sửa | Lê Hoàng Nam |
| 4 | CourseGoal | Nhập thông tin mục tiêu cho môn học và có thể xóa sửa | Lê Hoàng Nam |
| 5 | CourseOutCome | Nhập thông tin chuẩn đầu ra cho môn học | Trần Hữu Hào |
| 6 | CourseInfo | Hiện thị thông tin của môn học sao khi ta tạo mới một môn học | Trần Hữu Hào |
| 7 | CourseList | Hiển thị danh sách tất cả các môn học, tìm kiếm môn học có trong danh sách, lọc danh sách môn học | Trần Hữu Hào |
| 8 | CreateCourse | Tạo mới môn học | Trần Hữu Hào |
| 9 | Homepage | Giao diện quản lý | Trần Hữu Hào |
| 10 | LoginPage | Trang đăng nhập | Trần Hữu Hào |

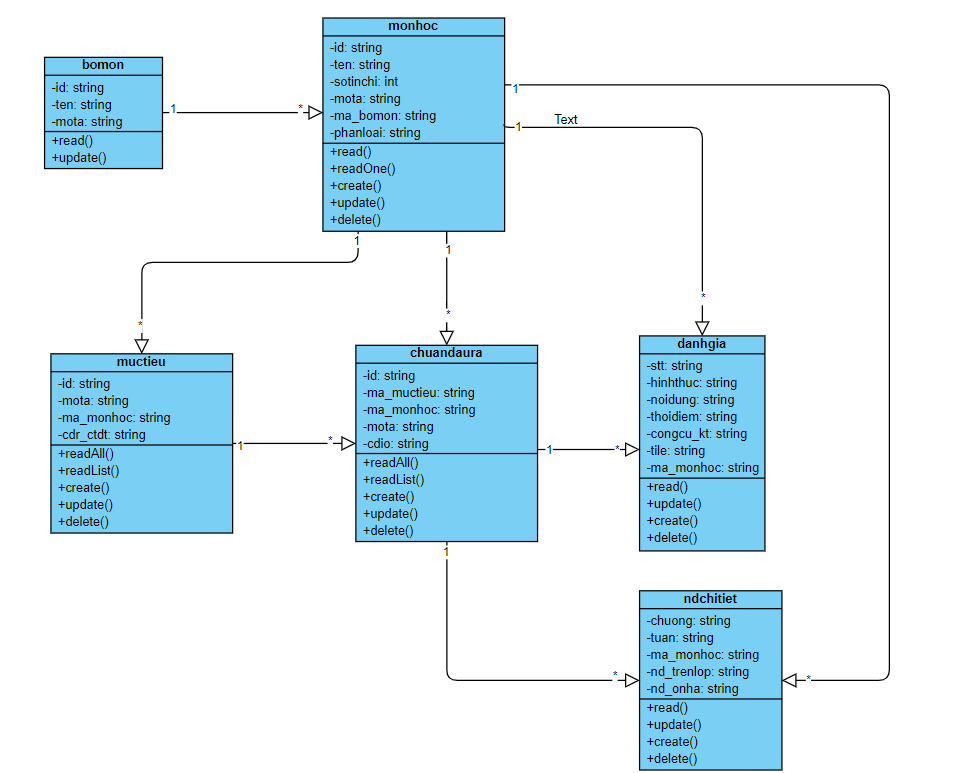
**Bảng 4** Các thiết kế Component

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TT** | **Phương thức** | **Mục đích** | **Tên file, số thứ tự dòng chứa khai báo** | **SV phụ trách viết** |
| 1 | getCourses() | Gửi yêu cầu đến máy chủ lấy danh sách các môn học | CourseApi.js (9) | Trần Hữu Hào |
| 2 | createCourse() | Gửi yêu cầu đến máy chủ thêm môn học | CourseApi.js (4) | Trần Hữu Hào |
| 3 | deleteCourse() | Gửi yêu cầu đến máy chủ xóa môn học | CourseApi.js (14) | Trần Hữu Hào |
| 4 | updateCourse() | Gửi yêu cầu đến máy chủ chỉnh sửa môn học | CourseApi.js (19) | Trần Hữu Hào |
| 5 | getCourseInfo() | Gửi yêu cầu đến máy chủ lấy thông tin 1 môn học | CourseApi.js (25) | Trần Hữu Hào |
| 6 | getCourseGoal() | Gửi yêu cầu đến máy chủ lấy danh sách mục tiêu | CourseApi.js (30) | Lê Hoàng Nam |
| 7 | getCourseGoalList() | Gửi yêu cầu đến máy chủ lấy danh sách mã mục tiêu | CourseApi.js (34) | Lê Hoàng Nam |
| 8 | createCourseGoal() | Gửi yêu cầu đến máy chủ thêm mục tiêu | CourseApi.js (38) | Lê Hoàng Nam |
| 9 | updateCourseGoal() | Gửi yêu cầu đến máy chủ chỉnh sửa mục tiêu | CourseApi.js (46) | Lê Hoàng Nam |
| 10 | deleteCourseGoal() | Gửi yêu cầu đến máy chủ xóa mục tiêu | CourseApi.js (42) | Lê Hoàng Nam |
| 11 | getCourseOutcome() | Gửi yêu cầu đến máy chủ lấy danh sách chuẩn đầu ra | CourseApi.js (51) | Trần Hữu Hào |
| 12 | getCourseOutcomeList() | Gửi yêu cầu đến máy chủ lấy danh sách mã chuẩn đầu ra | CourseApi.js (55) | Trần Hữu Hào |
| 13 | createCourseOutcome() | Gửi yêu cầu đến máy chủ thêm chuẩn đầu ra | CourseApi.js (58) | Trần Hữu Hào |
| 14 | updateCourseOutcome() | Gửi yêu cầu đến máy chủ chỉnh sửa chuẩn đầu ra | CourseApi.js (62) | Trần Hữu Hào |
| 15 | deleteCourseOutcome() | Gửi yêu cầu đến máy chủ xóa chuẩn đầu ra | CourseApi.js (66) | Trần Hữu Hào |
| 16 | getCourseEvualate() | Gửi yêu cầu đến máy chủ lấy danh sách đánh giá môn học | CourseApi.js (71) | Lê Hoàng Nam |
| 17 | createCourseEvualate() | Gửi yêu cầu đến máy chủ thêm đánh giá môn học | CourseApi.js (75) | Lê Hoàng Nam |
| 18 | updateCourseEvualate() | Gửi yêu cầu đến máy chủ chỉnh sửa đánh giá môn học | CourseApi.js (79) | Lê Hoàng Nam |
| 19 | deleteCourseEvualate() | Gửi yêu cầu đến máy chủ xóa đánh giá môn học | CourseApi.js (83) | Lê Hoàng Nam |
| 20 | getCourseContent() | Gửi yêu cầu đến máy chủ lấy danh sách nội dung môn học | CourseApi.js (88) | Lê Hoàng Nam |
| 21 | createCourseContent() | Gửi yêu cầu đến máy chủ thêm nội dung môn học | CourseApi.js (91) | Lê Hoàng Nam |
| 22 | updateCourseContent() | Gửi yêu cầu đến máy chủ chỉnh sửa nội dung môn học | CourseApi.js (94) | Lê Hoàng Nam |
| 23 | deleteCourseContent() | Gửi yêu cầu đến máy chủ xóa nội dung môn học | CourseApi.js (97) | Lê Hoàng Nam |
| 24 | loginCall() | Gửi yêu cầu đăng nhập bằng tài khoản và mật khẩu đến máy chủ | UserAPI.js (4) | Trần Hữu Hào |
| 25 | logoutCall() | Gửi yêu cầu đăng xuất | UserAPI.js (8) | Trần Hữu Hào |
| 26 | verifyToken() | Xác thực token để đăng nhập | UserAPI.js (13) | Trần Hữu Hào |
| 27 | useBreadcrumbs(routeConfig) | Phân tích đường dẫn trang web theo routeConfig  VD: /course/ABCDEF1234  => Trang chủ/Môn học/ABCDEF1234 | useBreadcrumbs.js (6) | Trần Hữu Hào |
| 28 | handleLogin() | Thực hiện đăng nhập | LoginPage/index.js | Trần Hữu Hào |
| 29 | handleLogout() | Thực hiện đăng xuất | Appbar/index.js | Trần Hữu Hào |
| 30 | removeAccents() | Xóa dấu cho chuỗi | utils/index.js | Trần Hữu Hào |
| 31 | handleSubmitCreate() | Thực hiện tạo mới môn học hoặc thông tin của môn học như mục tiêu, chuẩn đầu ra,... | CourseDetails/\*/index.js | chung |
| 32 | handleSubmitEdit() | Thực hiện chỉnh sửa môn học hoặc thông tin của môn học như mục tiêu, chuẩn đầu ra,... | CourseDetails/\*/row.js | chung |
| 33 | handleSubmitDelete() | Thực hiện xóa môn học hoặc thông tin của môn học như mục tiêu, chuẩn đầu ra,... | CourseDetails/\*/row.js | chung |

**Bảng 5** Mô tả các phương thức thực hiện trong một Component

1. Thiết kế cơ sở dự liệu

* Mô hình class Diagram



**Hình 7** Hình minh họa class diagram

Sinh viên thực hiện: Lê Hoàng Nam

|  |  |  |
| --- | --- | --- |
| **TT** | **Tên bảng** | **Mục đích** |
| 1 | bomon | Lưu thông tin bộ môn |
| 2 | chuandaura | Lưu trữ chuẩn đầu ra |
| 3 | danhgia | Lưu thông tin đánh giá |
| 4 | danhgia\_chuandaura | Lưu thông tin đánh giá của nhiều chuẩn đầu ra |
| 5 | monhoc | Lưu thông tin môn học |
| 6 | muctieu | Lưu thông tin mục tiêu của môn học |
| 7 | ndchitiet\_cdr | Lưu nội dung chi tiết của nhiều chuẩn đầu ra |
| 8 | noidungchitiet | Lưu thông tin về nội dung đào tạo của môn học |
| 9 | account | Lưu tài khoản truy cập đến website |

**Bảng 6** Mô tả các Table trong CSDL

1. Tổng các bảng mô tả các Field trong 1 Table

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Tên trường** | **Kiểu dữ liệu** | **Mục đích** |
| 1 | id | char | Phân biệt các bộ môn với nhau |
| 2 | ten | char | Chứa tên bộ môn |
| 3 | mota | char | Chứa thông tin mô tả môn học |

**Bảng 7** Mô tả các Field trong table bomon

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Tên trường** | **Kiểu dữ liệu** | **Mục đích** |
| 1 | id | char | Phân biệt chuẩn đầu ra |
| 2 | Ma\_muctieu | char | Chứa mã mục tiêu môn học |
| 3 | Ma\_monhoc | char | Chứa mã môn học |
| 4 | mota | char | Chứa thông tin mô tả của môn học |
| 5 | cdio | char | Chứa mã chuẩn đầu ra CDIO |

**Bảng 8** Mô tả các Field trong table Chuandaura

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Tên trường** | **Kiểu dữ liệu** | **Mục đích** |
| 1 | hinhthuc | char | Chứa hình thức đánh giá môn học |
| 2 | noidung | char | Chứa nội dung đánh giá môn học |
| 3 | thoidiem | char | Chứa thời gian kiểm tra, đánh giá môn học |
| 4 | congcu\_kt | char | Chứa công cụ kiểm tra, đánh giá môn học |
| 5 | tile | numeric | Chứa tỉ lệ phần trăm của các hình thức đánh giá |
| 6 | phanloai | char | Chứa thông tin phân loại đánh giá |
| 7 | ma\_monhoc | char | Chứa mã môn học để đánh giá |

**Bảng 9** Mô tả các Field trong table danhgia

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Tên trường** | **Kiểu dữ liệu** | **Mục đích** |
| 1 | ma\_monhoc | char | Chứa mã môn học dùng để đánh giá |
| 2 | hinhthuc\_ | char | Chứa hình thức đánh giá |
| 3 | ma\_cdr | char | Chứa mã chuẩn đầu ra môn học |
| 4 | ma\_muctieu | char | Chứa mã mục tiêu đánh giá môn học |

**Bảng 10** Mô tả các Field trong table danhgia\_chuandaura

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Tên trường** | **Kiểu dữ liệu** | **Mục đích** |
| 1 | id | char | Phân biệt bộ môn |
| 2 | ten | char | Chứa tên bộ môn |
| 3 | sotinchi | numeric | Chứa số tin chỉ môn học |
| 4 | mota | char | Chứa mô tả môn học |
| 5 | ma\_bomon | char | Chứa mã bộ môn |
| 6 | phanloai | char | Chứa thông tin để phân loại bộ môn |

**Bảng 11** Mô tả các Field trong table bomon

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Tên trường** | **Kiểu dữ liệu** | **Mục đích** |
| 1 | id | char | Phân biệt mục tiêu |
| 2 | mota | char | Chứa thông tin mô tả mục tiêu |
| 3 | cdr\_ctdt | char | Chứa mã chuẩn đầu ra theo chương trình đào tạo |
| 4 | ma\_monhoc | char | Chứa mã môn học |

**Bảng 12** Mô tả các Field trong table muctieu

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Tên trường** | **Kiểu dữ liệu** | **Mục đích** |
| 1 | ma\_monhoc | char | Chứa mã môn học |
| 2 | chuong | text | Chứa các chương trong nội dung giảng dạy |
| 3 | trenlop\_onha | char | Chứa nội dung học tập trên lớp và ở nhà |
| 4 | ma\_cdr | char | Chứa mã chuẩn đầu ra môn học |
| 5 | ma\_muctieu | char | Chứa mục tiêu môn học |

**Bảng 13** Mô tả các Field trong table ndchitiet\_cdr

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Tên trường** | **Kiểu dữ liệu** | **Mục đích** |
| 1 | chuong | text | Chứa các chương trong nội dung giảng dạy |
| 2 | tuan | char | Chứa thông tin tuần học |
| 3 | ma\_monhoc | char | Chứa mã môn học |
| 4 | nd\_trenlop | char | Chứa nội dung giảng dạy trên lơp |
| 5 | nd\_onha | char | Chứa nội dung học ở nhà |

**Bảng 14** Mô tả các Field trong table noidungchitiet

1. Thiết kế giao diện

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Màn hình/cửa sổ/Dialog** | **Mục đích** | **Giải thích** |
| 1 | Trang đăng nhập | Truy cập vào website quản lý | Đăng nhập bằng tài khoản và mật khẩu, nếu thành công máy chủ sẽ gửi về cookie chứa token cho client |
| 2 | Trang chủ | Hiện thị các chức năng của trang web | Ỏ trang chủ người dùng có thể sử dụng chức năng xem danh sách môn học hoặc tạo mới môn học. Ở góc trên cùng bên phải khi click vào sẽ hiện ra thông tin của user cho phép đăng xuất |
| 3 | Danh sách môn học | Xem danh sách, lọc, tìm kiếm, thêm, xóa môn học | Hiển thị các môn học đã nhập vào, người dùng có thể tìm kiếm dựa trên mã môn học. Có thể xem chi tiết về môn học |
| 4 | Chi tiết môn học | Hiển thị chi tiết về môn học | Ở giao diện này người dùng có thể thêm các thông tin chi tiết về môn học như: Mục tiêu, chuẩn đầu ra, đánh giá, nội dung. Người dùng có thể tạo mới hoặc xóa, sửa các thông tin chi tiết đó |

**Bảng 15** Các phần giao diện trong website

1. **CÀI ĐẶT VÀ KIỂM THỬ**

|  |  |  |
| --- | --- | --- |
| **TT** | **Tình huống** | **Mục đích** |
| 1 | Đăng nhập sai tài khoản  Web sẽ báo lỗi | Kiểm tra xác thực tài khoản |
| 2 | Xóa cookie chứa token và thực hiện các chức năng của trang web như xem danh sách, thêm mới,...  Web sẽ báo lỗi | Kiểm tra việc xác thực token trong mọi tác vụ |
| 3 | Nhập form đúng và gửi đi  Máy chủ sẽ gửi status 200, trang web sẽ thông báo thành công và reload | Kiểm tra chức năng chính của trang web |
| 4 | Không nhập form và nhấn submit  Trang web sẽ cảnh báo và không gửi lên máy chủ | Kiểm tra ở phía client trước khi gửi lên máy chủ, làm nhẹ việc cho máy chủ |
| 5 | Nhập form đầy đủ nhưng trùng khóa chính hoặc sai kiểu dữ liệu  Phía máy chủ gửi về lỗi | Đảm bảo ràng buộc cơ sở dữ liệu |

**Bảng 16** Các tình huống kiểm thử

1. **KẾT LUẬN**
2. Kết quả đạt được

Trong quá trình thực hiện đề tài nhóm em đã cố gắng hết sức để tìm hiều chương trình đào tạo khoa CNTT để xây dựng và cài đặt website nhưng vì thời gian có hạn nên chưa thể giải quyết được toàn bộ các vấn đề. Em mong nhận được sự cảm thông của thầy.

* Về công nghệ:
* Tìm hiểu và nắm bắt được các công cụ thiết kế web.
* Biết được cách thiết kế web động cũng như cách tổ chức cơ sở dữ liệu.
* Hiểu sâu hơn về React FrameWork.
* Hiểu sâu và vận dụng thành thạo mô hình MVC, các kiến thức hỗ trợ trong quá trình xây dựng website: CSS, HTML, Javascript, PostgreSQL.
* Tích lũy thêm kinh nghiệm về xây dựng kế hoạch cho việc phát triển một phần mềm.
* Về ứng dụng:
* Xây dựng thành công website quản lý môn học đáp ứng được như cầu quản lý.
* Giao diện thân thiện với người dùng.
* Cho phép người dùng tra cứu chương trình học của mình.

1. Hạn chế

* Thời gian thực hiện còn khá ít nên một số vấn đề chưa được giải quyết.
* Khi đưa vào hoạt động thực tế trên host chưa đáp ứng nhu cầu về tốc độ tải.
* Website chỉ đáp ứng nhu cầu quản lý ở mức dữ liệu nhỏ.
* Một số chức năng của website chỉ đáp ứng một phần nào yêu cầu của người dùng, theo sáng tạo.

1. Ý tưởng phát triển

* Hoàn thiện các chức năng còn hạn chế của hệ thống.
* Phân loại người truy cập giữa admin và user.
* Xây dựng hệ thống kiểm định AUN-QA cho website.

# 

# **TÀI LIỆU THAM KHẢO**

* Tài liệu React, Nodejs, Express, <https://fullstackopen.com/en/>
* Hướng dẫn React, <https://hocwebchuan.com/tutorial/reactjs/>
* Mẫu một project express, https://github.com/gothinkster/node-express-realworld-example-app/
* Hướng dẫn PostgreSQL, <https://www.postgresqltutorial.com/>
* Giới thiệu JWT, <https://topdev.vn/blog/jwt-la-gi/>
* Hướng dẫn sử dụng JWT trong nodejs, <https://codetheworld.io/nodejs-xac-thuc-nguoi-dung-su-dung-jwt-va-co-che-refresh-token.html>