**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO ĐỒ ÁN**

**NHẬP MÔN CÔNG NGHỆ PHẦN MỀM**

**GVLT :** TS. Nguyễn Thị Minh Tuyền

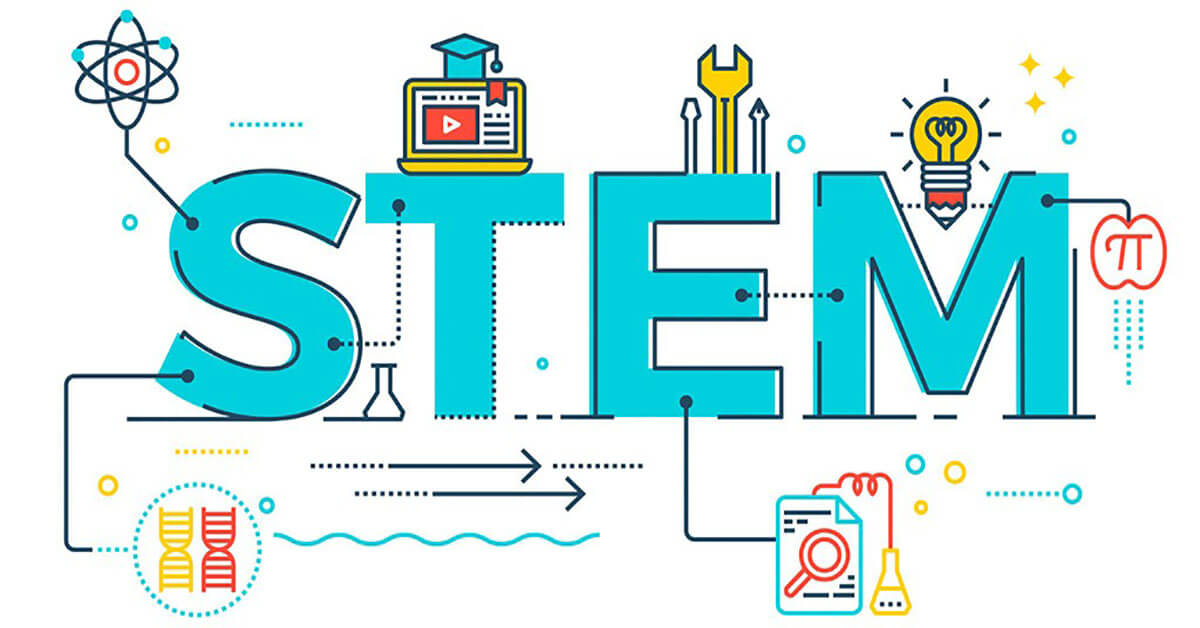
**GVHD :** ThS. Ngô Ngọc Đăng Khoa

**LỚP :** CQ2016/31

**NHÓM : 2**

**CHỦ ĐỀ : Quản lý lớp học STEM**

***Tp. Hồ Chí Minh - Tháng 12/2018***



C:\Users\tdqua_000\Dropbox\SS-Slides\DeCuong-CDIO\Template CDIO v4.2\Templates\Hinh anh\LogoTruong.png

Khoa Công nghệ Thông tin

Đại học Khoa học Tự nhiên, ĐHQG-HCM

Tháng 12/2018

**MỤC LỤC**

[1 Tổng quan & Milestone 5](#_Toc533890960)

[Thông tin nhóm 5](#_Toc533890961)

[Milestone 6](#_Toc533890962)

[2 Phát biểu bài toán sơ lược và công cụ hỗ trợ 7](#_Toc533890963)

[3 Proposal 8](#_Toc533890964)

[3.1 Đề xuất tính năng website 8](#_Toc533890965)

[Danh sách các chức năng chính của website 8](#_Toc533890966)

[Yêu cầu phần cứng để phát triển ứng dụng 9](#_Toc533890967)

[3.2 Kế hoạch phát triển website 9](#_Toc533890968)

[Phân tích yêu cầu 9](#_Toc533890969)

[Thiết kế phần mềm 9](#_Toc533890970)

[Cài đặt (implement) phần mềm 9](#_Toc533890971)

[Kiểm thử phần mềm 9](#_Toc533890972)

[Triển khai, bảo trì 9](#_Toc533890973)

[3.3 Kế hoạch nhân sự & chi phí 9](#_Toc533890974)

[4 Phân tích 10](#_Toc533890975)

[4.1 Tổng quan yêu cầu 10](#_Toc533890976)

[Danh sách các stakeholder 10](#_Toc533890977)

[Các yêu cầu chức năng 11](#_Toc533890978)

[Các yêu cầu phi chức năng 11](#_Toc533890979)

[4.2 Đặc tả yêu cầu 12](#_Toc533890980)

[Sơ đồ Use Cases 12](#_Toc533890981)

[Đặc tả Use Cases 13](#_Toc533890982)

[4.3 Prototype – Main UI 19](#_Toc533890983)

[5 Thiết kế 25](#_Toc533890984)

[5.1 Thiết kế kiến trúc 25](#_Toc533890985)

[5.2 Thiết kế dữ liệu 27](#_Toc533890986)

[Database Diagram 27](#_Toc533890987)

[Kiến trúc phần mềm 28](#_Toc533890988)

[+ **ConTroller**: Nơi lắng nghe lời gọi từ client thông qua các chuỗi url restfull, với phương thức Http tương ứng (get, post, delete,…). 29](#_Toc533890989)

[+ **Repository**: (Kho chứa) 29](#_Toc533890990)

[+ **Models**: (Ado.net entity framework) 30](#_Toc533890991)

[- Luồng xử lý: 31](#_Toc533890992)

[5.3 Thiết kế giao diện người dùng 31](#_Toc533890993)

[Đặc tả 31](#_Toc533890994)

[Danh sách các màn hình 32](#_Toc533890995)

[6 Kiểm thử 42](#_Toc533890996)

[6.1 Kế hoạch kiểm thử 42](#_Toc533890997)

[Kiểm tra các thành phần (Component Testing) 42](#_Toc533890998)

[Kiểm tra tích hợp (Intergration Testing) 42](#_Toc533890999)

[Kiểm tra hệ thống (System Testing) 43](#_Toc533891000)

[6.2 Testcases 45](#_Toc533891001)

[Danh sách 45](#_Toc533891002)

[Đặc tả 46](#_Toc533891003)

[7 Hướng phát triển 51](#_Toc533891004)

[8 Đánh giá đóng góp thành viên trong nhóm 51](#_Toc533891005)

# Tổng quan & Milestone

### Thông tin nhóm

|  |  |  |  |
| --- | --- | --- | --- |
| MSSV | Họ tên | Email | Số điện thoại |
| 1612085 | Trương Thành Danh | [toinguoitayson](mailto:leduongtuananh97@gmail.com)@gmail.com | 0987996185 |
| 1612597 | Huỳnh Duy Tân | [duytan.0422@gmail.com](mailto:duytan.0422@gmail.com) | 0363383993 |
| 1612602 | Nguyễn Quang Thạch | quangthach59@gmail.com | 0339246761 |
| 1612647 | Lê Văn Thi | thi174hcmus@gmail.com | 0334977111 |
| 1612806 | Lâm Khang Vĩ | lamkhangviiii@gmail.com | 0787718110 |

### Milestone

|  |  |  |  |
| --- | --- | --- | --- |
| Cột mốc | Công việc dự kiến | Ước lượng  (man hour) | Sản phẩm |
| 11/10/2018  Proposal | Nghiên cứu website quản lý lớp học STEM | 2 | Hiểu được ý nghĩa của phần mềm |
| Đưa ra bài toán sơ lược | 5 | Hiểu được các chức năng website, các nền tảng cho việc lập trình backend, frontend, database, kiểm thử và quản lí dự án |
| Đưa ra các chức năng & yêu cầu phần cứng để phát triển website | 3 | Chức năng & yêu cầu phần cứng |
| Đưa ra kế hoạch phát triển | 2 | Đưa ra kế hoạch phát triển phần mềm (chủ yếu đến trước giai đoạn tiến hóa) |
| Đề xuất kế hoạch nhân sự, chi phí thực hiện | 2 | Chi phí nhân sự & thời gian |
| Chọn lựa mô hình phát triển phần mềm | 1 | Chọn mô hình MVC |
| Viết proposal và báo cáo | 2 | Proposal (PDF) |
| 25/11/2018  Phân tích | Đề xuất các stakeholders & mô tả | 2 | Stakeholders & description |
| Xác định chi tiết các chi tiết | 5 | Đặc tả yêu cầu chức năng & phi chức năng |
| Đề xuất mô hình & đặc tả use cases. | 5 | Đưa ra ít nhất 10 use cases & sơ đồ Use Cases. |
| Viết báo cáo & nộp Moodle | 2 | Phân tích (PDF) |
| 25/11/2018  Thiết kế | Thiết kế mô hình quan niệm ER cho Database | 3 | ER Model |
| Thiết kế kiến trúc backend | 5 | Rest API |
| Thiết kế dữ liệu dựa trên ER Model | 5 | Đặc tả thiết kế database |
| Thiết kế giao diện người dùng (song song với công việc thiết kế client) | 5 | Danh sách, sơ đồ & đặc tả (mô tả trực quan) các màn hình ở Client |
| Viết báo cáo & nộp Moodle | 2 | Thiết kế (PDF) |
| 29/12/2018  Kiểm thử | Đề xuất kế hoạch kiểm thử | 3 | Kế hoạch kiểm thử (bao gồm Component Testing, Integration Testing, System Testing) |
| Đưa ra danh sách testcases. | 3 | Danh sách & đặc tả từng testcases. |
| Viết tài liệu kiểm thử & nộp Moodle | 2 | Kiểm thử (PDF) |
| 29/12/2018  Báo cáo | Hoàn thiện báo cáo | 5 | Báo cáo (Final – PDF) |

# Phát biểu bài toán sơ lược và công cụ hỗ trợ

**Website quản lý lớp học STEM** được phát triển để giúp cán bộ giáo viên thực hiện các nghiệp vụ quản lý trường học như: quản lý học sinh, quản lý giảng dạy, quản lý thu chi, hỗ trợ thông báo, cung cấp tài liệu học tập… giúp cán bộ giáo viên có thể ứng dụng CNTT trong trường học hướng đến trường học điện tử, kết nối trường học và phụ huynh một cách nhanh chóng, theo đó phụ huynh có thể tra cứu tình hình học tập của học sinh dễ dàng hơn.

Để xây dựng được một ứng dụng như vậy chúng ta sẽ chia thành các nhóm chính bao gồm: **backend,** **frontend**, **quản trị cơ sở dữ liệu, quản lý và kiểm thử dự án**.

Đối với **frontend** nhóm sử dụng HTML5, CSS6 (có thể thêm Bootstrap 4) và framework Angular 6. **Backend** sử dụng ASP.Net (C#) và mô hình MVC, **database** sử dụng Microsoft SQL.

Một số công cụ hỗ trợ cần cài đặt: NodeJS, Visual Studio 2017, Visual Studio Code, MS SQL 2014, Github (để lưu trữ source).

# Proposal

## Đề xuất tính năng website

### Danh sách các chức năng chính của website

*Website được chia thành 4 phần:*

***- Học sinh:***

+ Chọn tham gia các khóa học đang mở.

*(Học sinh chỉ được phép tham gia các khóa học chưa đậu (dưới 5đ) hoặc chưa học).*

+ Xem bài giảng/bài tập và nộp bài trong khóa học đã đăng ký.

+ Nhận thông báo từ khóa học, thông báo chung.

+ Cập nhật thông tin cá nhân.

***- Giáo viên:***

+ Tạo khóa học.

+ Upload bài giảng/bài tập trong khóa học.

+ Download bài tập từ học sinh.

+ Set điểm cho mỗi bài tập mà học sinh đã nộp.

+ Xem danh sách học sinh và phụ huynh tương ứng.

+ Gửi thông báo cho học sinh và phụ huynh.

+ Cập nhật thông tin cá nhân.

***- Phụ huynh:***

+ Xem kết quả/tiến độ học của con.

+ Xem thông tin thanh toán.

+ Xem thông báo.

***- Quản lý:***

+ Có đầy đủ các chức năng của phụ huynh, giáo viên, học sinh.

+ Có quyền vô hiệu hóa những tài khoản không hợp lệ

+ Kiểm tra danh sách thu chi.

### Yêu cầu phần cứng để phát triển ứng dụng

**Backend:** không cần yêu cầu về phần cứng cao.

**Frontend:** không cần yêu cầu về phần cứng cao.

## Kế hoạch phát triển website

### Phân tích yêu cầu

Lựa chọn các công nghệ, nền tảng phù hợp.

Đưa ra chức năng của website và xây dựng kiến trúc tổng quan của website.

Lựa chọn nhân sự phù hợp với từng mảng, bộ phận.

### Thiết kế phần mềm

Đưa ra các flow chính cho ứng dụng.

Đưa ra thiết kế UI/UX của ứng dụng.

Xây dựng kiến trúc cho phần backend.

### Cài đặt (implement) phần mềm

Thực hiện xây dựng các API phục vụ cho website

Xây dựng ứng dụng với thiết kế đã được xây dựng và kết nối với server.

### Kiểm thử phần mềm

Thực hiện kiểm tra các chức năng của website, UI/UX.

Kiểm tra tính bảo mật.

### Triển khai, bảo trì

Deloy lên một server thật.

## Kế hoạch nhân sự & chi phí

Sau khi có được sơ lược về quy trình phát triển ứng dụng, nhóm quyết định chia công việc cụ thể như sau:

+ **Lê Văn Thi**: nhóm trưởng, backend, frontend, quản lý về mặt kỹ thuật cho nhóm

+ **Trương Thành Danh**: cài đặt hàm API sever.

+ **Huỳnh Duy Tân**: design giao diện bằng PTS.

+**Nguyễn Quang Thạch**: thiết kế Database.

+ **Lâm Khang Vĩ**: kiểm thử.

Khi chuẩn bị hoàn thành website, nhóm sẽ thực hiện deploy lên một server thật. Có nhiều tổ chức cung cấp dịch vụ này *(AWS, Google Cloud Platform, DigitalOcean, VCCloud)*. Dự kiến sử dụng *Google Cloud Platform for Student* để tiết kiệm chi phí.

Các vai trò có thể thay đổi tùy theo quá trình thực hiện nếu có các vấn đề phát sinh xảy ra.

# Phân tích

## Tổng quan yêu cầu

### Danh sách các stakeholder

|  |  |  |
| --- | --- | --- |
| **STT** | **Stakeholder** | **Mô tả** |
| 1 | Học sinh | Người học, chịu sự quản lý của nhà trường. |
| 2 | Giáo viên | Chịu trách nhiệm dạy học và quản lý học sinh. |
| 3 | Phụ huynh | Theo dõi tình hình học tập của học sinh. |
| 4 | Quản lý | Quản lí học sinh, giáo viên và chương trình đào tạo, gửi các thông báo đến phụ huynh, học sinh, giáo viên. |
| 5 | Bộ phận tài chính, kế toán | Tính toán chi phí, thu chi học phí, lương giáo viên |
| 6 | Đội ngũ IT | Cài đặt và bảo trì hệ thống |
| 7 | Các bộ phận khác | … |

### Các yêu cầu chức năng

* Về phía học sinh:
* Học sinh có thể đăng kí lớp học khi đến học kì mới.
* Học sinh có thể nhận thông báo từ nhà trường, khóa học đã đăng kí.
* Học sinh có thể nộp bài tập khi giao viên giao.
* Học sinh có thể xem điểm bài tập mà mình đã làm.
* Học sinh có thể xem học phí của học kì.
* Về phía giáo viên:
* Giáo viên có thể tạo lớp học cho học sinh đăng kí.
* Giáo viên có thể upload bài giảng/bài tập trong khóa học.
* Giáo viên có để set điểm cho mỗi bài tập mà học sinh đã nộp.
* Giáo viên có thể gửi thông báo đến học sinh, phụ huynh.
* Về phía phụ huynh:
* Phụ huynh có thể xem điểm học sinh.
* Phụ huynh có thể xem thông báo từ giáo viên, nhà trường.
* Phụ huynh có thể xem học phí của học kì.
* Về phía người quản lý:
* Người quản lý có quyền vô hiệu hóa tài khoản không hợp lệ.
* Người quản lý có đầy đủ quyền của giáo viên, học sinh
* Người quản lí có thể kiểm tra thu chi (xét học sinh đã đóng học phí hay chưa).

### Các yêu cầu phi chức năng

* Đáp ứng được lượng người dùng lớn, ví dụ cần phải đảm bảo hệ thống hoạt động ổn định khi có cùng lúc ít nhất **500 người.**
* Đảm bảo ứng dụng đạt được hiệu năng ổn định và ít bị deley trong quá trình sử dụng.
* Người dùng cần sử dụng tài khoản để đăng nhập.
* Thông tin của học sinh không được tiết lộ ra ngoài.
* Luôn cập nhật thời gian thực các thay đổi từ phía nhà quản lí
* Hệ thống server chạy được trên các dịch vụ server lớn như AWS / DigitalOccean / Google Cloud.
* Website đủ các chứng chỉ / giấy phép bắt buộc để hoạt động*.*
* Website phải dễ sử dụng với người dùng mới.

## Đặc tả yêu cầu

### Sơ đồ Use Cases

### Đặc tả Use Cases

**Use case #1**

|  |  |
| --- | --- |
| **Use case ID** | **U001** |
| Tên Use Case | Đăng ký khóa học |
| Tóm tắt | Cho phép học sinh đăng ký khóa học khi đến học kì mới |
| Tác nhân | Khi học sinh chọn “dach sách khóa học” trong mục “khóa học”. |
| Điều kiện tiên quyết | Học sinh phải đăng nhập và đang có danh sách khóa học cho học kì mới |
| Kết quả | Học sinh đăng ký được khóa học cần phải học trong học kì. |
| Kịch bản chính | Khi học sinh click vào “danh sách khóa học” trong mục “khóa học” sẽ hiển thị danh sách khóa học của học kì đó.  Học sinh click vào môn học muốn đăng ký.  Chọn ghi danh để tham gia khóa học. |
| Kịch bản phụ | Học sinh không thấy danh sách khóa học khi người quản lý chưa mở. |
| Ràng buộc phi chức năng | Học sinh chỉ được đăng ký khóa học chưa học hoặc đã học nhưng dưới 5đ.  Máy tính phải kết nối mạng |

**Use case #2**

|  |  |
| --- | --- |
| **Use case ID** | **U002** |
| Tên Use Case | Thực hiện bài tập |
| Tóm tắt | Cho phép học sinh xem và nộp bài tập khi giáo viên giao trong khóa học |
| Tác nhân | Khi học sinh chọn vào link bài tập |
| Điều kiện tiên quyết | Học sinh phải đăng nhập và có bài tập được giao trong khóa học đã đăng ký |
| Kết quả | Nộp bài tập giáo viên giao |
| Kịch bản chính | Khi học sinh chọn vào link bài tập sẽ chuyển đến trang nộp bài.  Học sinh chọn “Choose File” để chọn add file nộp và chọn button “Nộp bài”. |
| Kịch bản phụ | Học sinh nộp file không đúng định dạng sẽ thông báo lỗi |
| Ràng buộc phi chức năng | Không được nộp file rỗng hoặc nội dung không liên quan.  Máy tính phải kết nối mạng |

**Use case #3**

|  |  |
| --- | --- |
| **Use case ID** | **U003** |
| Tên Use Case | Tạo khóa học |
| Tóm tắt | Cho phép giáo viên tạo khóa học mới để học sinh tham gia |
| Tác nhân | Khi giáo viên chọn button “Thêm khóa học” |
| Điều kiện tiên quyết | Giáo viên phải đăng nhập |
| Kết quả | Tạo khóa học mới cho học sinh |
| Kịch bản chính | Khi chọn button “Thêm khóa học” sẽ chuyển đến màn hình khác để giáo viên điền thông tin về khóa học. |
| Kịch bản phụ | Bị lỗi do rớt mạng sẽ hiển thị thông báo lỗi |
| Ràng buộc phi chức năng | Khóa học được tạo phải đúng chương trình đào tạo, không được tạo tùy ý.  Máy tính phải kết nối mạng. |

**Use case #4**

|  |  |
| --- | --- |
| **Use case ID** | **U004** |
| Tên Use Case | Upload bài giảng |
| Tóm tắt | Cho phép giáo viên đăng tải bài giảng về khóa học mà giáo viên quản lý |
| Tác nhân | Giáo viên click “+” để thêm bài giảng ở màn hình khóa học quản lý |
| Điều kiện tiên quyết | Giáo viên phải đăng nhập và giáo viên có quản lý ít nhất 1 khóa học |
| Kết quả | Upload thành công bài giảng |
| Kịch bản chính | Chọn tài liệu để upload  Click “OK” để đăng tải  Thông báo đăng tải thành công |
| Kịch bản phụ | File được chọn đăng tải không đúng định dạng  Bị lỗi do rớt mạng sẽ hiển thị thông báo lỗi |
| Ràng buộc phi chức năng | Bài giảng phải đầy đủ, chính xác  Máy phải kết nối mạng |

**Use case #5**

|  |  |
| --- | --- |
| **Use case ID** | **U005** |
| Tên Use Case | Tạo hoạt động bài tập |
| Tóm tắt | Cho phép giáo viên đăng tải bài tập về khóa học mà giáo viên quản lý |
| Tác nhân | Giáo viên click “+” để thêm bài tập ở màn hình khóa học quản lý |
| Điều kiện tiên quyết | Giáo viên phải đăng nhập và giáo viên có quản lý ít nhất 1 khóa học |
| Kết quả | Nhận tín hiệu yêu cầu book hoặc book xe |
| Kịch bản chính | Chọn bài tập để upload  Click “OK” để đăng tải  Thông báo đăng tải thành công |
| Kịch bản phụ | File được chọn đăng tải không đúng định dạng  Bị lỗi do rớt mạng sẽ hiển thị thông báo lỗi |
| Ràng buộc phi chức năng | Bài tập phải bám sát nội dung giảng dạy  Máy phải kết nối mạng |

**Use case #6**

|  |  |
| --- | --- |
| **Use case ID** | **U006** |
| Tên Use Case | Quản lý danh sách học sinh |
| Tóm tắt | Cho phép giáo viên xem thông tin học sinh và cho điểm bài tập mà học sinh đã nộp |
| Tác nhân | Khi giáo viên chọn mục “Danh sách học sinh” |
| Điều kiện tiên quyết | Giáo viên phải đăng nhập |
| Kết quả | Xem thông tin học sinh |
| Kịch bản chính | Chọn mục “Danh sách học sinh” sẽ hiển thị danh sách học sinh  Click vào học sinh để xem thông tin chi tiết |
| Kịch bản phụ | Khi mất mạng sẽ hiển thị lỗi |
| Ràng buộc phi chức năng | Giáo viên không được để lộ thông tin học sinh ra bên ngoài  Máy phải kết nối mạng |

**Use case #7**

|  |  |
| --- | --- |
| **Use case ID** | **U007** |
| Tên Use Case | Xem kết quả học sinh |
| Tóm tắt | Cho phép phụ huynh xem điểm của học sinh |
| Tác nhân | Khi phụ huynh click vào “Xem điểm” |
| Điều kiện tiên quyết | Sau khi phụ huynh đăng nhập thành công |
| Kết quả | Xem điểm tập của học sinh |
| Kịch bản chính | Phụ huynh chọn “Xem điểm” sẽ chuyển đến màn hình khác, yêu cầu nhập thời gian (học kì).  Click “Xem” để hiển thị điểm môn học của học sinh trong học kì đó |
| Kịch bản phụ | Học kì chưa kết thúc sẽ không hiển thị điểm môn học đó hoặc điểm là null |
| Ràng buộc phi chức năng | Máy phải kết nối mạng |

**Use case #8**

|  |  |
| --- | --- |
| **Use case ID** | **U008** |
| Tên Use Case | Xem học phí |
| Tóm tắt | Cho phép phụ huynh xem học phí (mức phí, đã đóng hay chưa) của học sinh ở học kì đó |
| Tác nhân | Màn hình thông tin học sinh |
| Điều kiện tiên quyết | Phụ huynh phải đăng nhập |
| Kết quả | Kiểm tra được mức học phí |
| Kịch bản chính | Xem thông tin mức học phí  Đã đóng học phí sẽ có kí hiệu đánh dấu |
| Kịch bản phụ | Không có |
| Ràng buộc phi chức năng | Máy tính phải kết nối mạng |

**Use case #9**

|  |  |
| --- | --- |
| **Use case ID** | **U009** |
| Tên Use Case | Xem thông báo từ trung tâm |
| Tóm tắt | Xem các thông báo từ nhà trường /trung tâm giảng dạy |
| Tác nhân | Màn hình thông báo chung |
| Điều kiện tiên quyết | Phụ huynh phải đăng nhập |
| Kết quả | Xem thông báo từ nhà trường/trung tâm |
| Kịch bản chính | Xem các thông báo quan trọng từ nhà tường |
| Kịch bản phụ | Không có |
| Ràng buộc phi chức năng | Máy phải có kết nối mạng |

**Use case #10**

|  |  |
| --- | --- |
| **Use case ID** | **U010** |
| Tên Use Case | Thống kê thu chi |
| Tóm tắt | Cho phép quản lý, kiểm tra việc đóng học phí của học sinh |
| Tác nhân | Quản lý chọn “Quản lý thu chi” |
| Điều kiện tiên quyết | Quản lý phải đăng nhập |
| Kết quả | Đánh dấu học sinh đã đóng hoặc chưa đóng học phí |
| Kịch bản chính | Khi học sinh đóng học phí, quản lý sẽ kiểm tra và đánh dấu học sinh vào danh sách đã đóng học phí. |
| Kịch bản phụ | Không có |
| Ràng buộc phi chức năng | Quá trình kiểm tra phải chính xác, minh bạch.  Máy phải có kết nối mạng |

**Use case #11**

|  |  |
| --- | --- |
| **Use case ID** | **U011** |
| Tên Use Case | Quản lý học sinh, giáo viên, phụ huynh |
| Tóm tắt | Cho phép quản lý chỉnh sửa thông tin học sinh, giáo viên, phụ huynh |
| Tác nhân | Quản lý thực hiện tác vụ quản lý trong mục “dành cho quản lý” |
| Điều kiện tiên quyết | Quản lý phải đăng nhập |
| Kết quả | Thực hiện thành công các tác vụ mà quản lý muốn |
| Kịch bản chính | Chọn mục “dành cho quản lý”.  Thực hiện thay đổi, chỉnh sửa thông tin học sinh, giáo viên tùy theo nhu cầu. |
| Kịch bản phụ | Không có |
| Ràng buộc phi chức năng | Quản lý không được thay đổi thông tin tùy tiện  Máy phải có kết nối mạng |

**Use case #12**

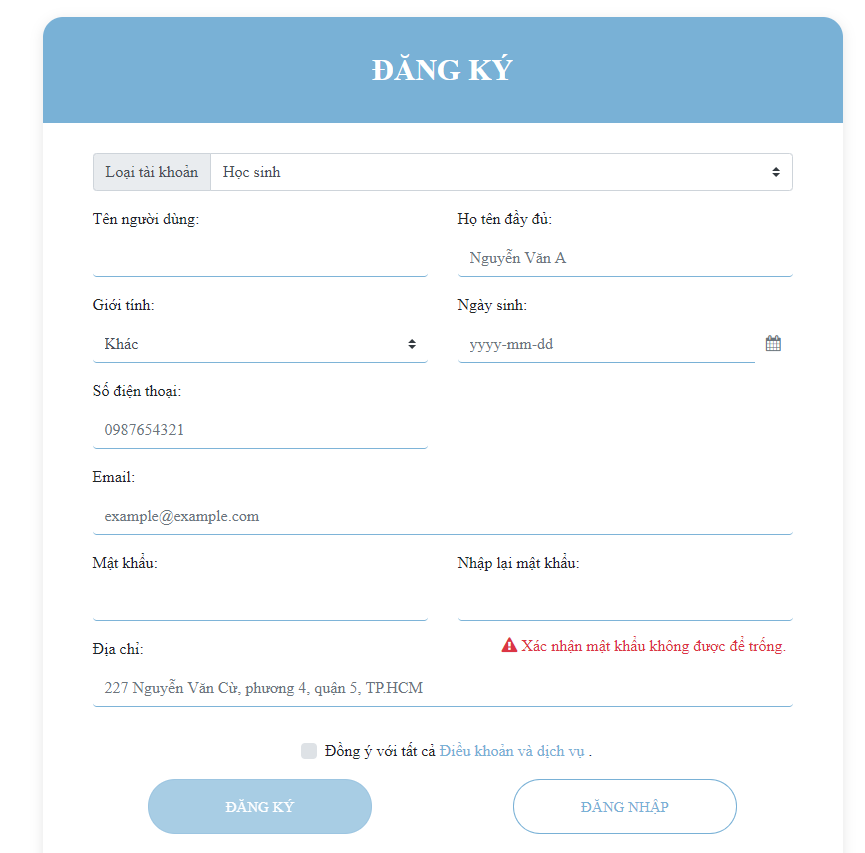
|  |  |
| --- | --- |
| **Use case ID** | **U012** |
| Tên Use Case | Đăng ký |
| Tóm tắt | Cho phép người dùng đăng ký để đăng nhập vào website |
| Tác nhân | Người dùng thực hiện chức năng “đăng ký” |
| Điều kiện tiên quyết | Người dùng truy cập và trang website và chưa đăng nhập |
| Kết quả | Đăng ký thành công, tài khoản được lưu vào hệ thống |
| Kịch bản chính | Người dùng thực hiện chức năng đăng ký  Điền các thông tin cần thiết  Đồng ý với điều khoản  Chọn đăng ký |
| Kịch bản phụ | Điền thông tin không đầy sẽ thông báo lỗi  Không đồng ý điều khoản sẽ thông báo lỗi  Tài khoản bị trùng sẽ không báo lỗi  Mật khẩu nhập lại không đúng sẽ thông báo lỗi  Mật khẩu quá ngắn sẽ thông báo lỗi |
| Ràng buộc phi chức năng | Thông tin người dùng phải được lưu giữ bí mật  Máy phải có kết nối mạng |

**Use case #13**

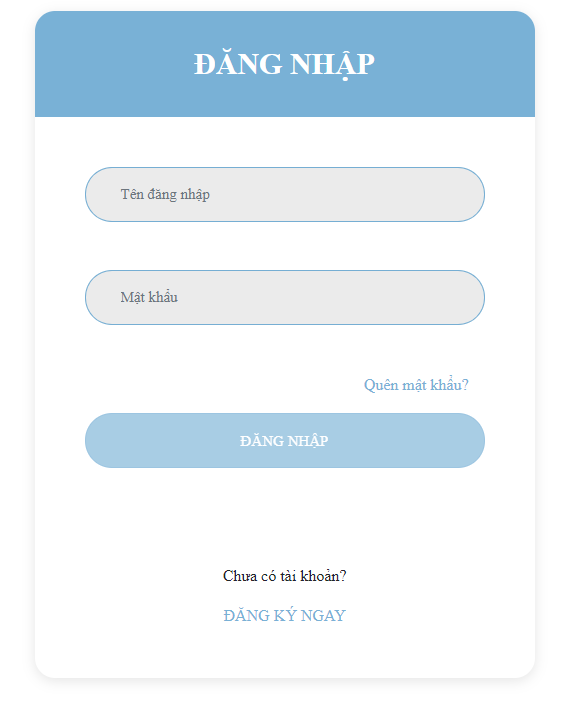
|  |  |
| --- | --- |
| **Use case ID** | **U013** |
| Tên Use Case | Đăng nhập |
| Tóm tắt | Cho phép người dùng đăng nhập để thực hiện các chức năng khác của Website |
| Tác nhân | Người dùng thực hiện chức năng “đăng nhập” |
| Điều kiện tiên quyết | Đã đăng ký tài khoản thành công |
| Kết quả | Truy cập hệ thống quản lý của Website |
| Kịch bản chính | Người dùng điền thông tin username và password  Click “Đăng nhập”.  Truy cập vào hệ thống quản lý  Tùy vào loại tài khoản sẽ có quyền thực hiện các chức năng khác nhau trong hệ thống |
| Kịch bản phụ | Username không tồn tại sẽ thông báo lỗi  Password không chính xác sẽ thông báo lỗi |
| Ràng buộc phi chức năng | Máy phải có kết nối mạng |

## Prototype – Main UI

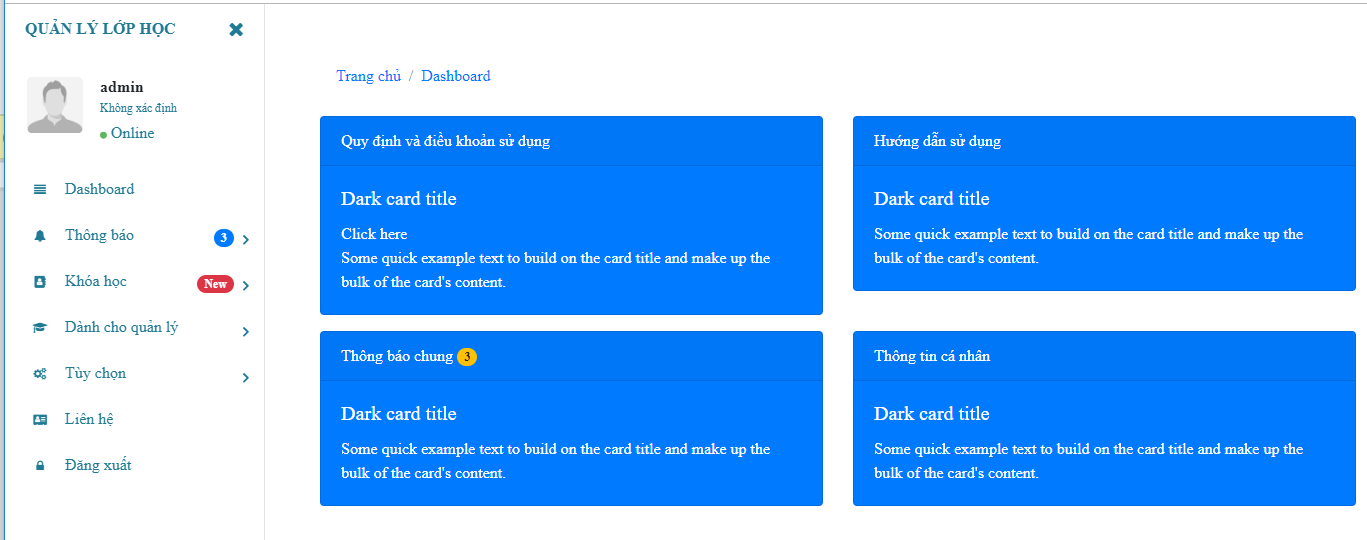
*1- Màn hình đăng ký*



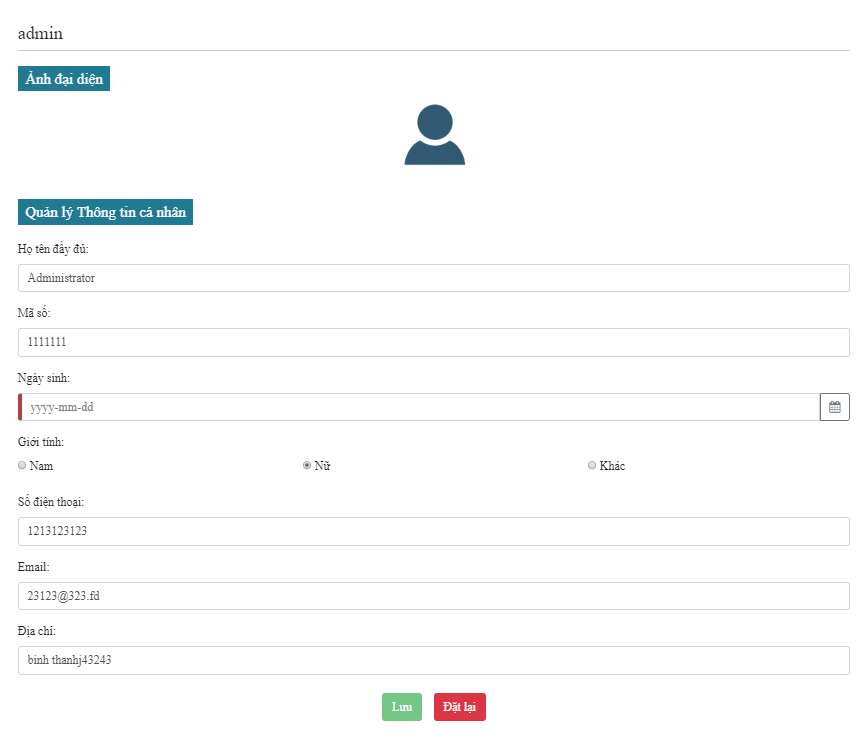
*2- Màn hình đăng nhập*



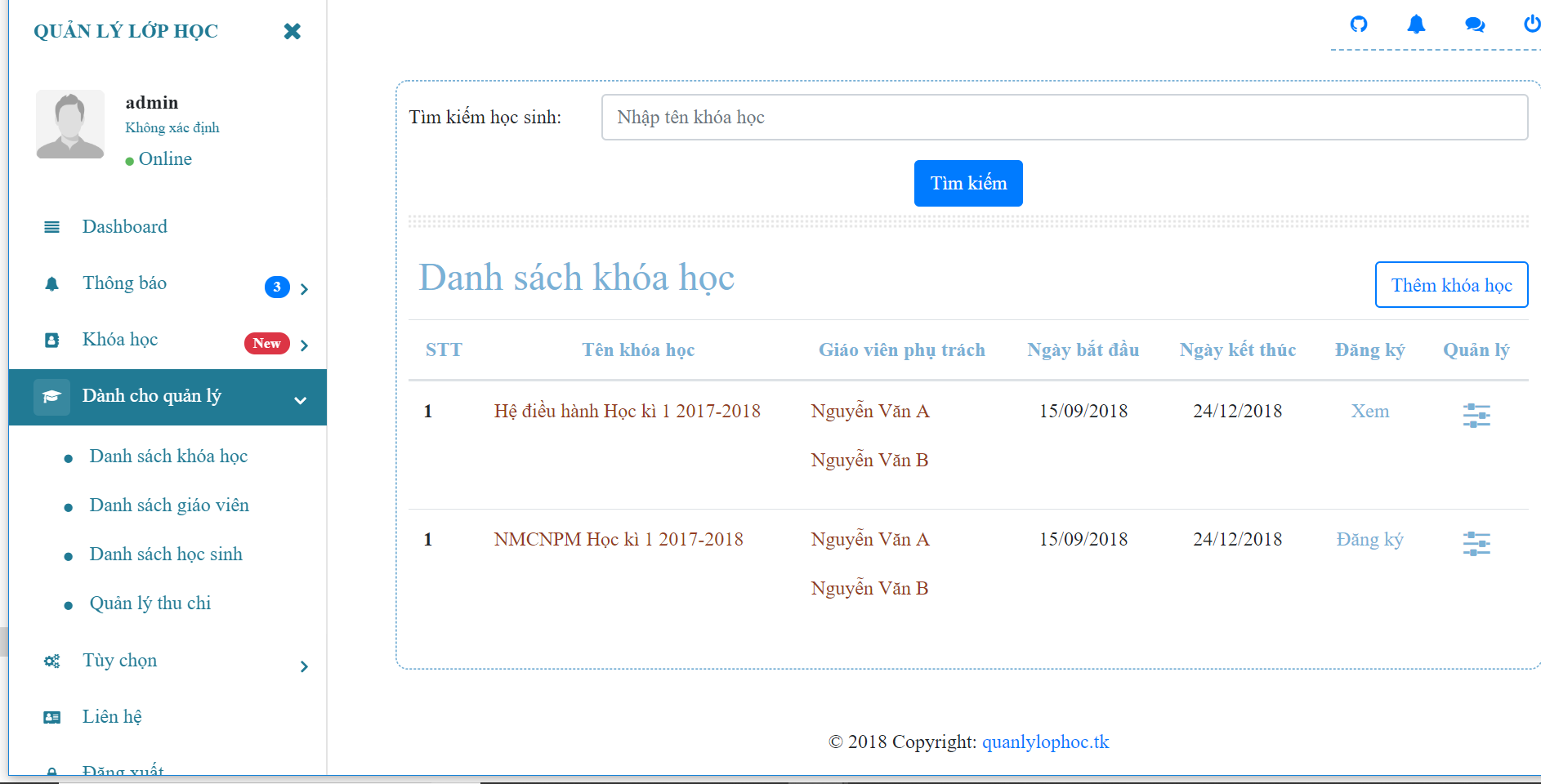
*3- Giao diện chung của trang quản lý*



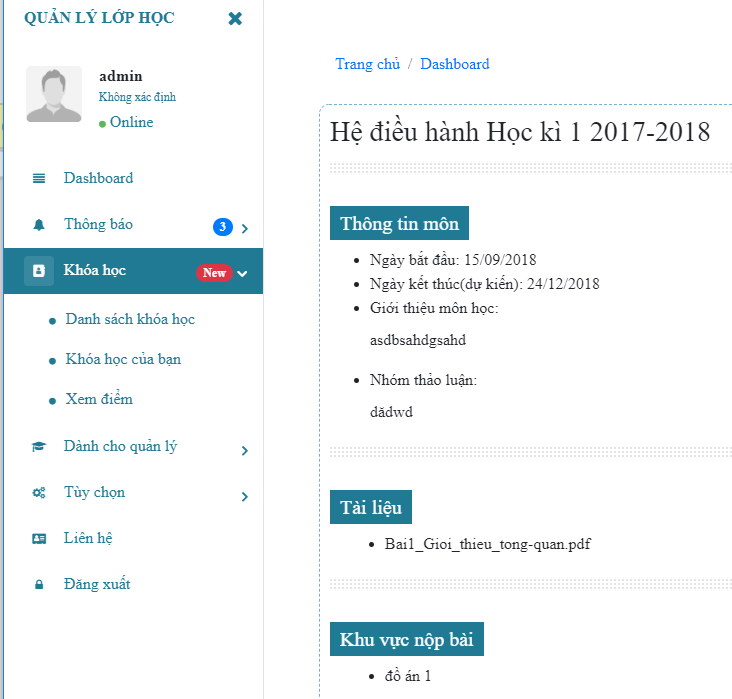
*4- Giao diện cập nhật thông tin người dùng*

z

*5- Giao diện danh sách khóa học*



*6- Giao diện quản lý của 1 khóa học*



# Thiết kế

## Thiết kế kiến trúc

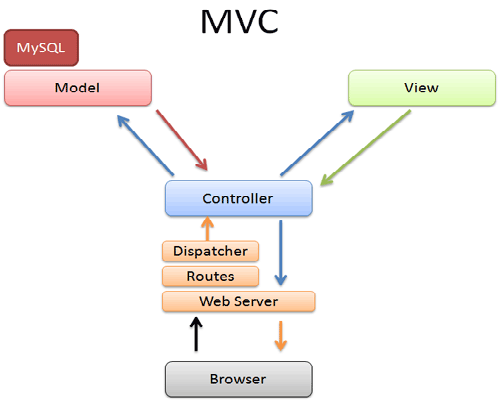
**M**odel **V**iew **C**ontroller hoặc MVC là cách gọi phổ biến, là một mô hình thiết kế phần mềm cho các ứng dụng phát triển trên nền tảng web. Một Model View Controller được tạo thành bởi 3 thành phần chính sau:

**Model** - Là thành phần thấp nhất của mô hình có nhiệm vụ duy trì dữ liệu.

**View** - Có nhiệm vụ hiển thị các phần dữ liệu đến người sử dụng.

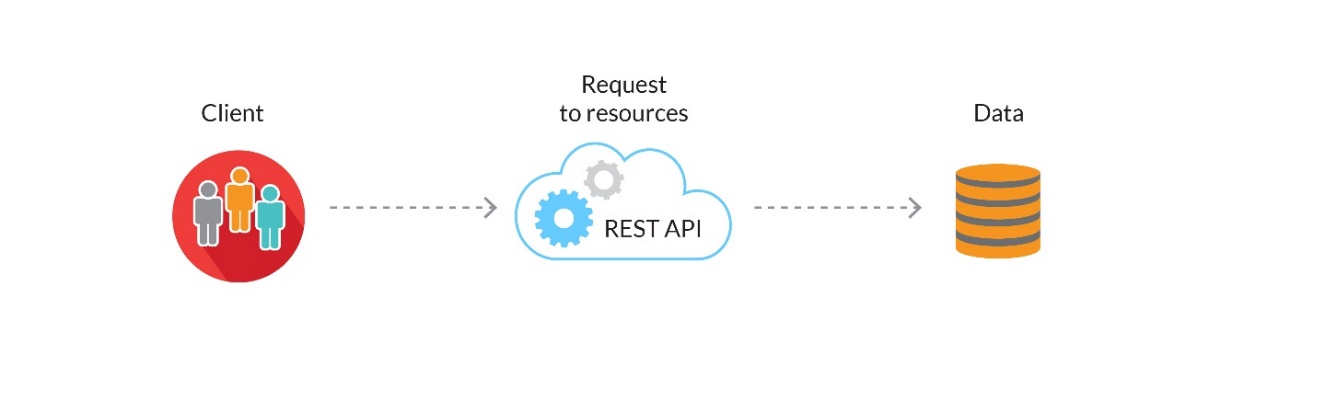
**Controller** - Là phần Code điều khiển sự tương tác giữa Model và View.

MVC trở lên phổ biến bởi nó phân tách ứng dụng một cách hợp lý giữa tầng giao diện người dùng và hỗ trợ các phần có liên quan. Phần controller nhận tất cả các request cho ứng dụng và sau đó làm việc với phần model để chuẩn bị dữ liệu cho phần view. Phần view sử dụng dữ liệu được chuẩn bị bởi controller và sau đó tạo ra các dữ liệu được nhìn thấy phía người sử dụng.



*Mô hình minh họa cho mô hình MVC*

Hệ thống có sử dụng **Restful API**



*Mô hình minh hoạ cho Rest API*

**REST** có những ưu điểm sau đây:

- REST sử dụng chuẩn HTTP nên nó đơn giản hơn nhiều so với trước đây. Tạo clients, phát triển các API, tài liệu dễ hiểu hơn và không có nhiều thứ mà REST không làm được.

- REST cho phép nhiều định dạng dữ liệu khác nhau.

- REST có hiệu suất tốt hơn và khả năng mở rộng.

## Thiết kế dữ liệu

### Database Diagram



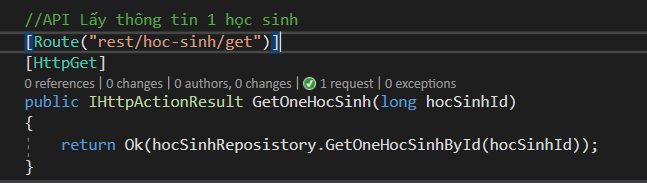
### Kiến trúc phần mềm

* + - 1. **Server**: Sử dụng công nghệ asp.net kết hợp với web api Restfull
* Dữ liệu được lưu trên database ado.net
* Mã nguồn được phân chia theo các controller xử lý các đối tượng trong cơ sở dữ liệu.

Có 2 loại class chính: Controller và Reposistory

Các Class Controller chứa lắng các lời gọi rest api từ client đế server và gọi các hàm xử lý từ các class Reposistory.

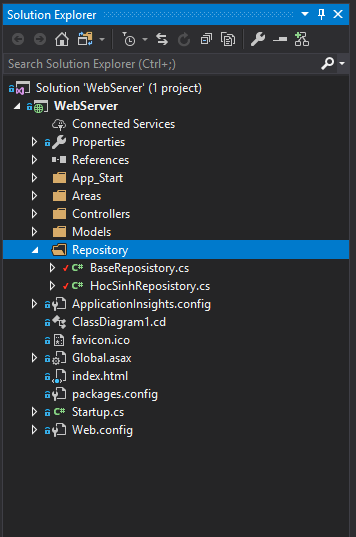
+ **ConTroller**: Nơi lắng nghe lời gọi từ client thông qua các chuỗi url restfull, với phương thức Http tương ứng (get, post, delete,…).



Và gọi các hàm xử lý từ **Repository** để trả kết quả về cho phía client.

+ **Repository**: (Kho chứa)

Ứng với mỗi đối tượng (mỗi bảng) trong database sẽ có 1 file Repository tương ứng. File này chứa tất các các hàm xử lý chính của đối tượng đó và được gọi ra ở Controller.



+ **Models**: (Ado.net entity framework)

Chứa các đối tượng dữ liệu tham chiếu đến database

Ứng với mỗi mỗi bảng trong database có 1 đối tượng được tự động tạo ra.

* + - 1. **Client**: Sử dụng *framework Angular* 6
* Phát triển theo hướng Single page Application
* Chia mã nguồn thành các component khác nhau tương ứng với các màn hình giao diện.

Mỗi component chứa các file: .html, .css hiển thị giáo diện và file .ts để xử lý logic, binding dữ liệu cũng như điều hướng trang web

* **Luồng xử lý:**

1. Phía Client(trình duyệt) gửi request gọi đến server thông qua url restfull
2. **Controller** lắng nghe request
3. **Controller** gọi vào **Repository** để thực hiện xử lý yêu cầu
4. **Repository** xử lý thông qua các hàm, bao gồm cả việc đọc ghi vào database
5. **Repository** trả kết quả về **Controller** (nếu có)
6. Controller trả kết quả về Client bằng các chuỗi json

7. Trình duyệt nhận chuỗi json và hiển thị)

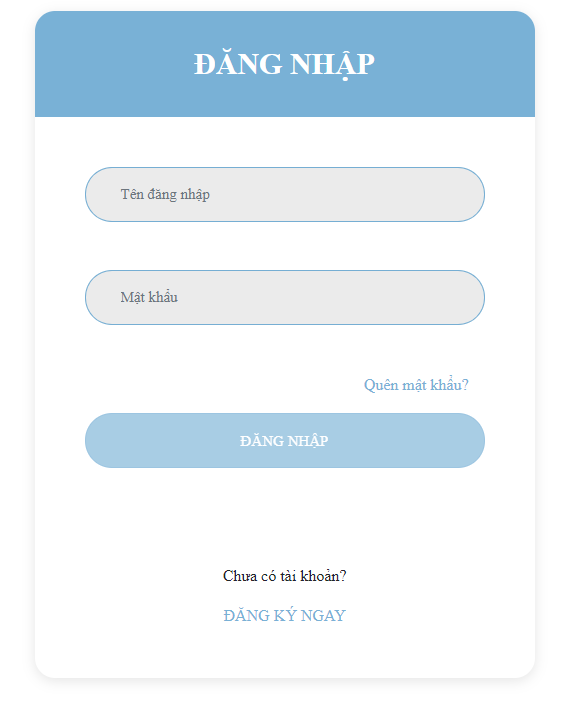
## Thiết kế giao diện người dùng

### Đặc tả

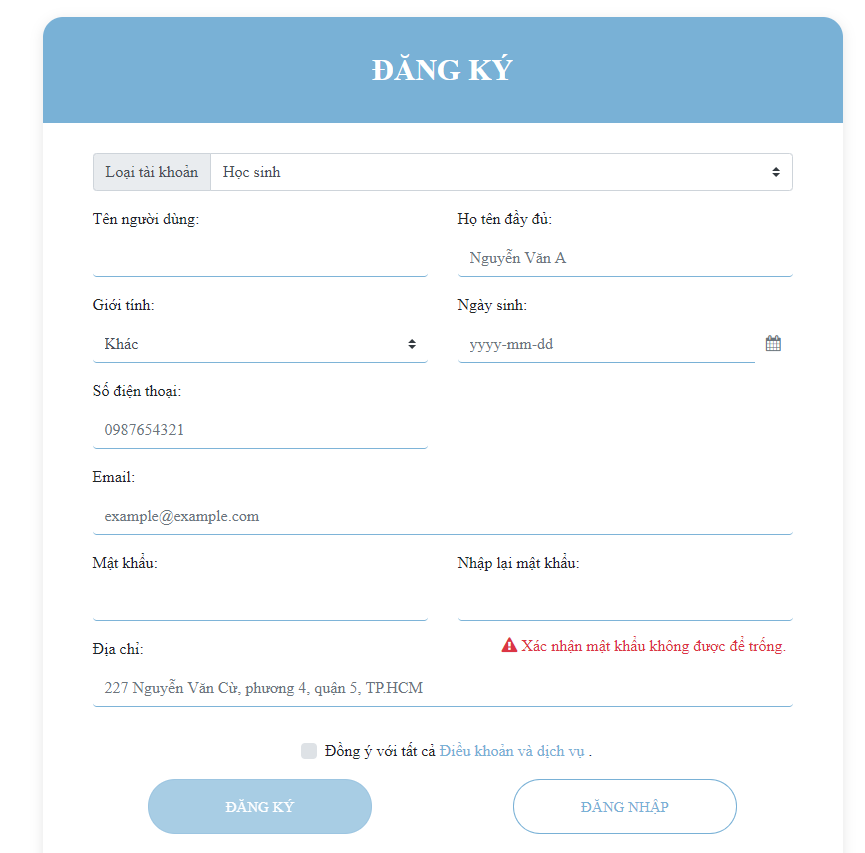
|  |  |  |
| --- | --- | --- |
| **#** | **Tên màn hình** | **Ý nghĩa/ Ghi chú** |
| 1 | Đăng nhập | Đăng nhập vào tài khoản |
| 2 | Đăng kí | Đăng kí tài khoản |
| 3 | Giao diện Dashboard | Hiển thị các mục, chức năng cần thiết |
| 4 | Giao diện thêm khóa học | Dùng cho giáo viên thêm khóa học mới |
| 5 | Giao diện khóa học | Hiển thị thông tin khóa học, bài giảng, bài tập |
| 6 | Giao diện nộp bài | Dùng để học sinh nộp bài tập của giáo viên giao |
| 7 | Giao diện thông tin cá nhân | Dùng để cập nhật và xem thông tin cá nhân |
| 8 | Giao diện xem điểm | Cho phép học sinh, phụ huynh xem điểm |
| 9 | Giao diện kiểm tra học phí | Cho phép xem học phí của học kì |
| 10 | Giao diện thông báo | Cho phép người dùng xem thông báo về khóa học / thông báo chung |

### Danh sách các màn hình

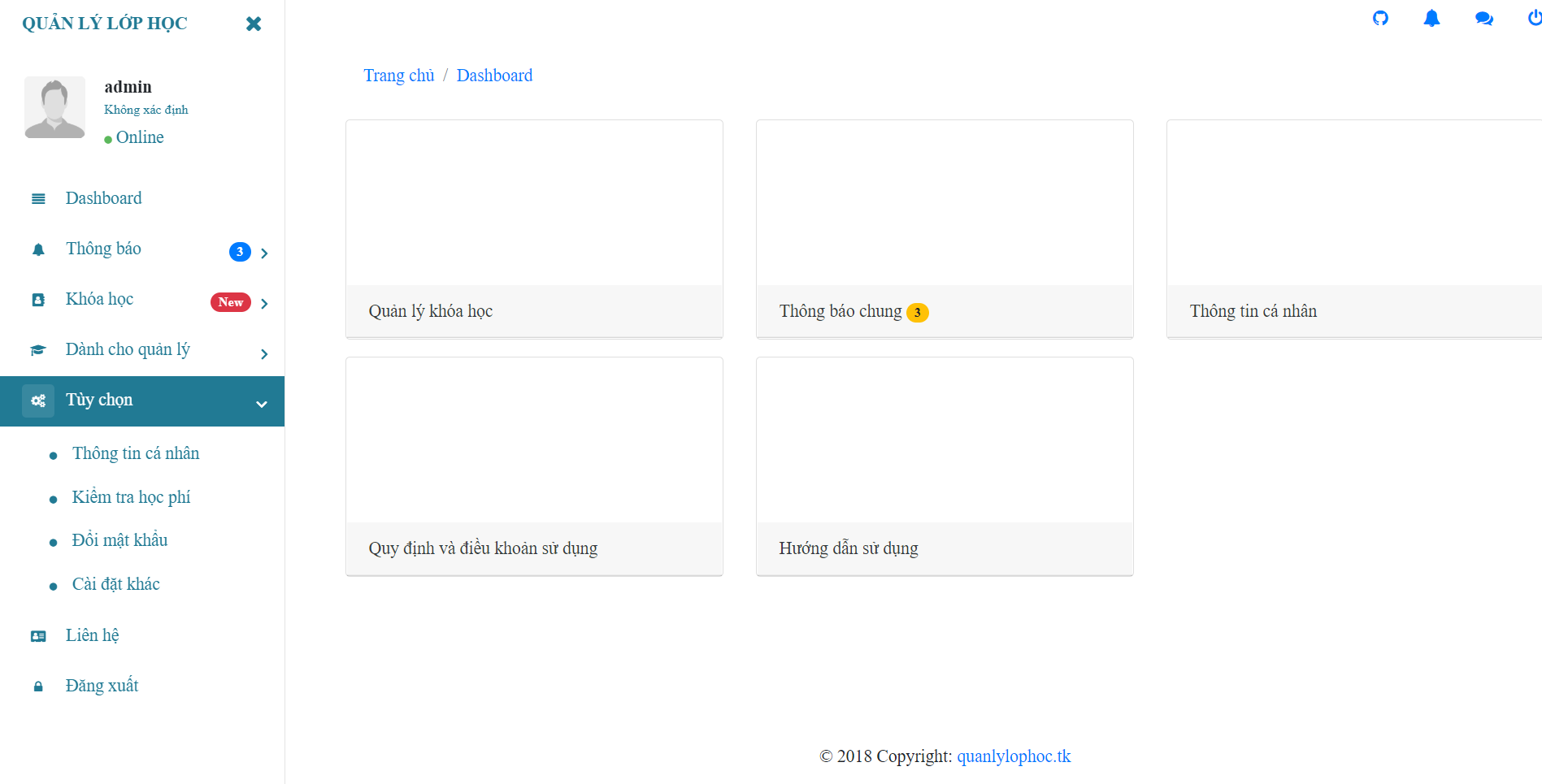
*Màn hình “1”*



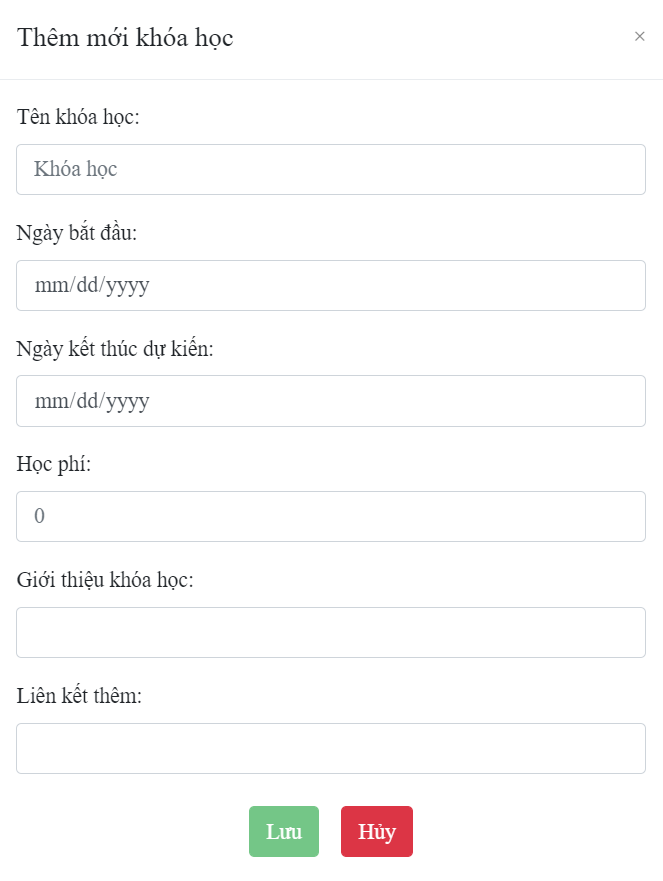
*Màn hình “2”*

**

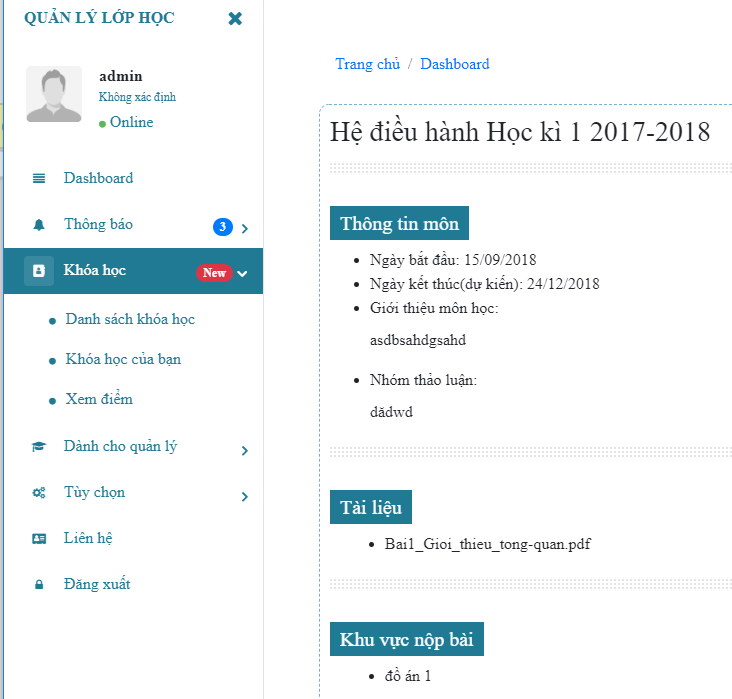
*Màn hình “3”*



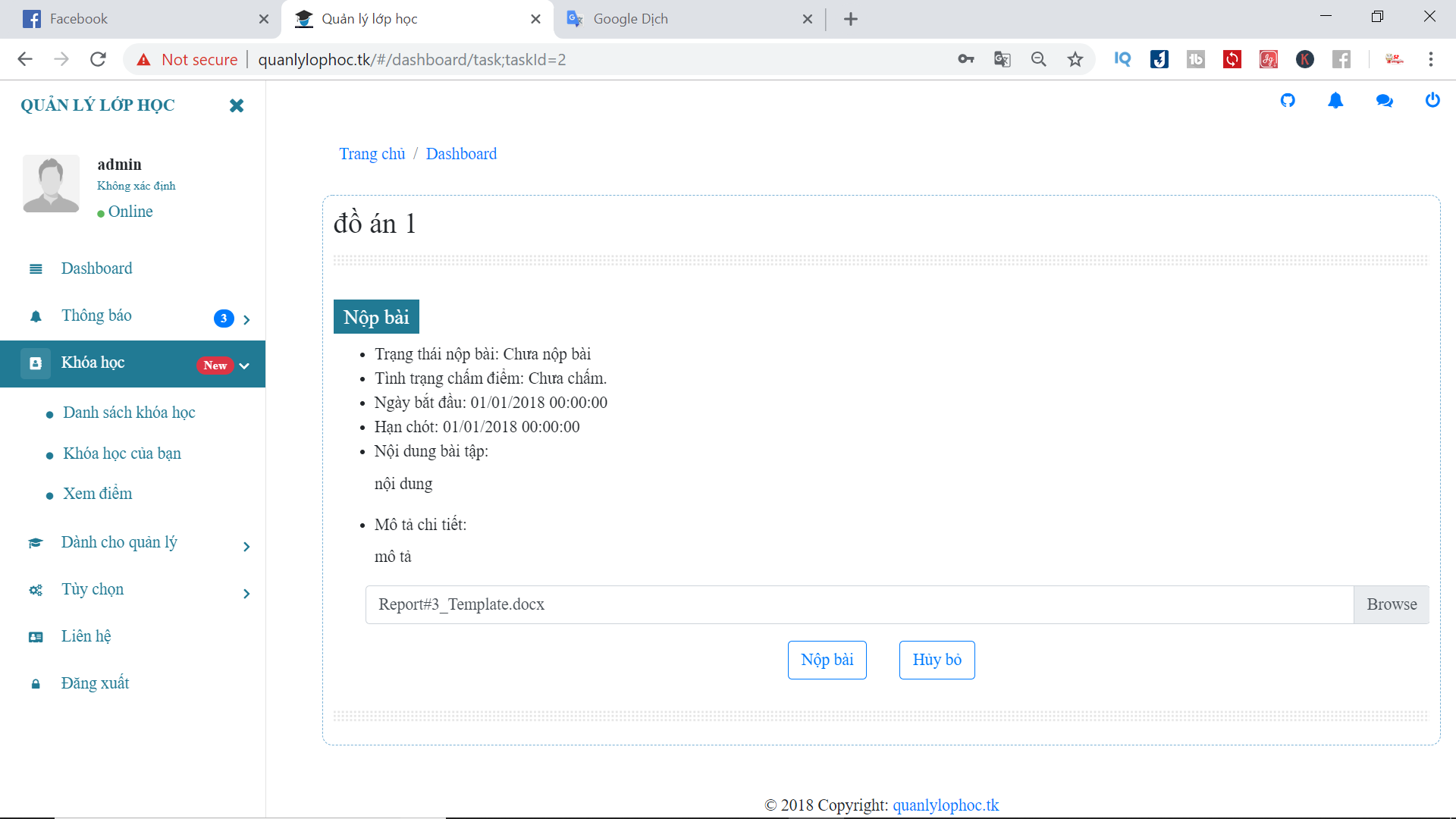
*Màn hình “4”*



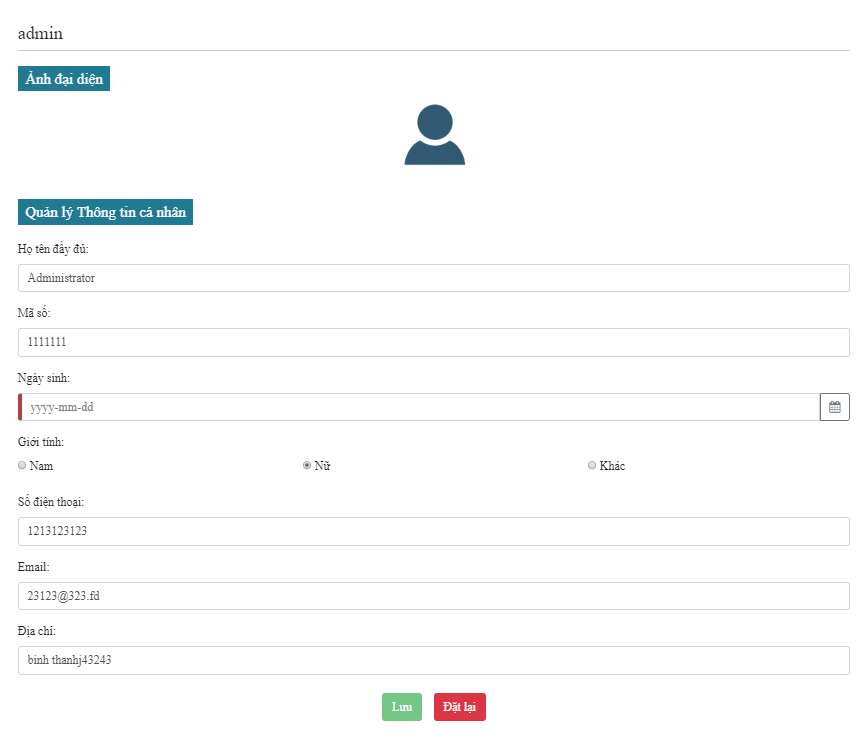
*Màn hình “5”*



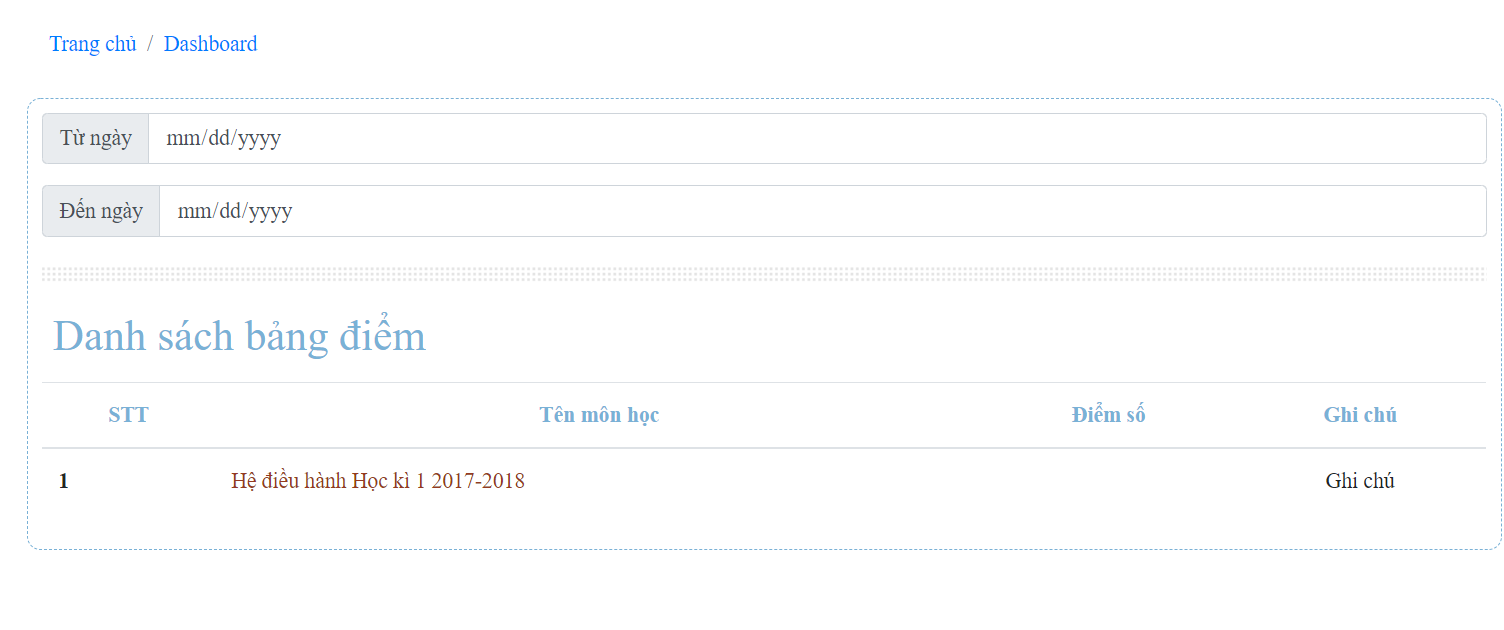
*Màn hình “6”*



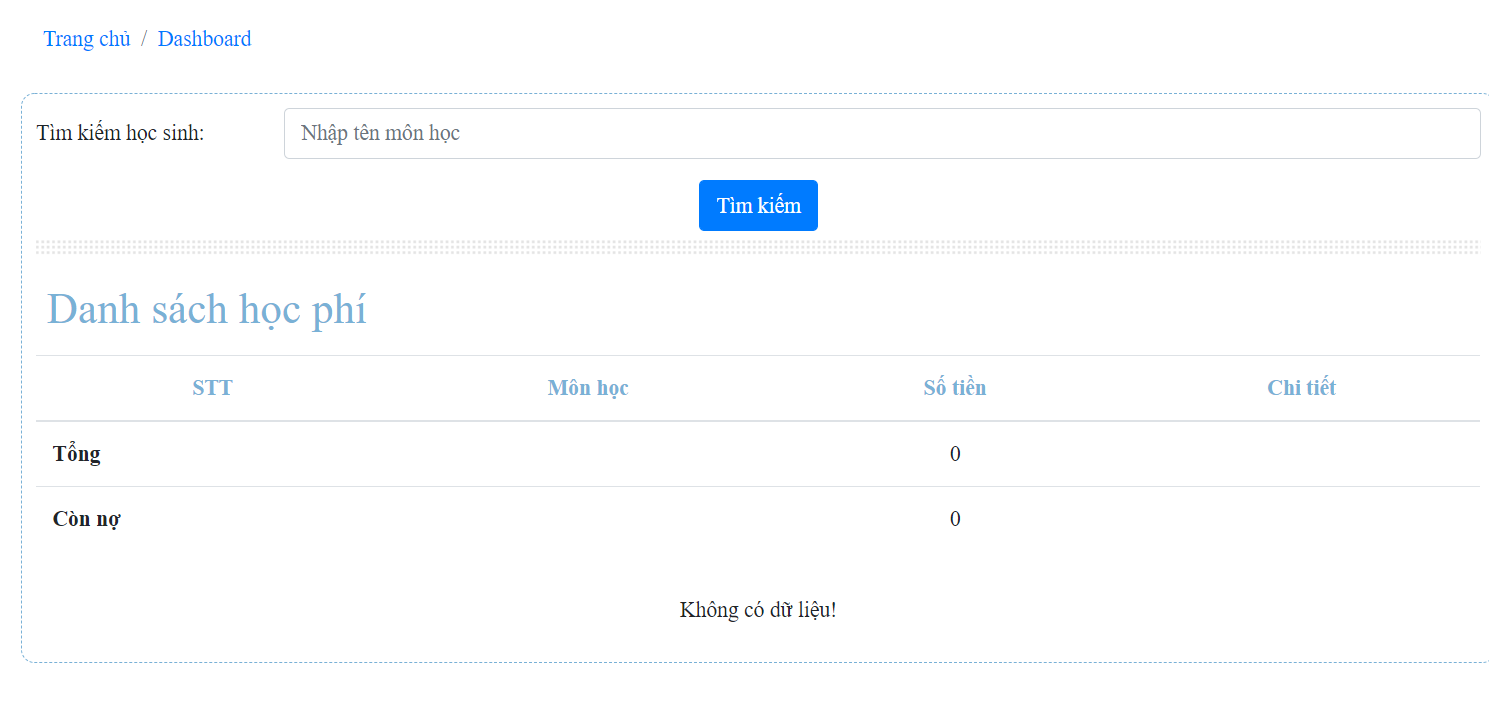
*Màn hình “7”*



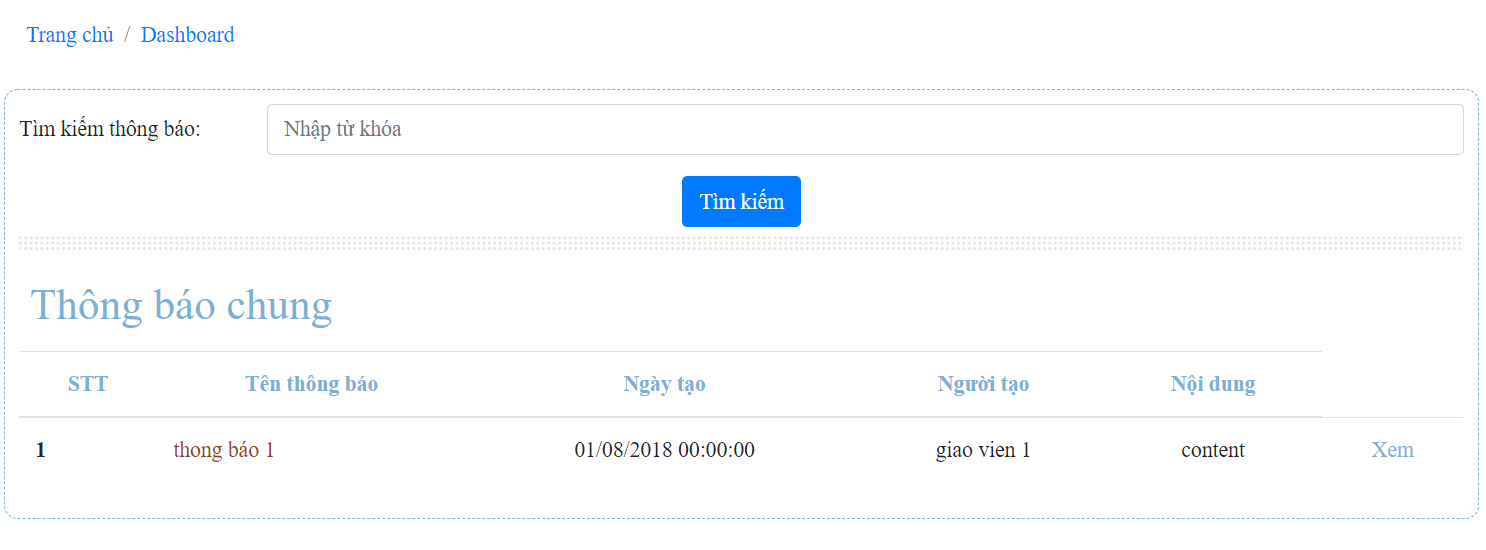
*Màn hình “8”*



*Màn hình “9”*



*Màn hình “10”*



# Kiểm thử

## Kế hoạch kiểm thử

### Kiểm tra các thành phần (Component Testing)

* **Mục tiêu kiểm tra:**
* Các thành phần (component).
* Chuyển đổi dữ liệu giữa các thành phần.
* **Kết quả mong muốn đạt được:**
* Tìm kiếm và vá được các lỗi ở mức module, lớp.
* Kết hợp các module ở mức cơ bản, kiểm tra kết quả của việc kết hợp.
* **Nền tảng thực hiện kiểm tra:**
* Tài liệu mô tả các thành phần.
* Thiết kế chi tiết các thành phần.
* Code.
* **Quá trình thực hiện kiểm tra:**
* Test các phần nhỏ nhất của chương trình: Các hàm (function), lớp, component.
* Kết hợp các hàm, kiểm tra sự kết nối giữa các hàm.

### Kiểm tra tích hợp (Intergration Testing)

* **Mục tiêu kiểm tra:**
* Việc thực thi liên quan đến database.
* Cơ sở hạ tầng thực hiện: Mạng, thiết bị.
* Giao tiếp (interface).
* **Kết quả mong muốn đạt được:**
* Các component kết hợp được với nhau.
* Database hoạt động được với chương trình.
* Chương trình hoạt động được trên nền tảng cơ sở hạ tầng thực hiện.
* **Nền tảng thực hiện kiểm tra:**
* Thiết kế phần mềm và hệ thống.
* Kiến trúc phần mềm.
* Các tiến trình công việc (Workflows).
* Các trường hợp sử dụng (Use cases).
* **Quá trình thực hiện kiểm tra:**
* Component Intergration Testing: Tương tác giữa các component với nhau, thực hiện sau **[Component Testing].**
* Functional Testing: Kiểm thử các nhiệm vụ chức năng (chẳng hạn: Chức năng đăng nhập, đăng kí).
* Kiểm tra sự gắn kết module chỉ cần quan tâm sự giao tiếp, không quan tâm chức năng của từng module nữa (đã thực hiện ở bước 1).

### Kiểm tra hệ thống (System Testing)

* **Mục tiêu kiểm tra:**
* Có được hướng dẫn vận hành hệ thống, cho người sử dụng.
* Có được cấu hình cho hệ thống để hoạt động ổn định.
* **Kết quả mong muốn đạt được:**
* Hệ thống chạy được trên nền tảng tương ứng, thực hiện tốt chức năng của phần mềm.
* **Nền tảng thực hiện kiểm tra:**
* Bảng mô tả yêu cầu chi tiết phần mềm.
* Use cases.
* Bảng mô tả chi tiết chức năng.
* Các báo cáo phân tích rủi ro đã có, đã khắc phục từ 2 bước trước.
* **Quá trình thực hiện kiểm tra:**
* Tạo ra các test có môi trường gần giống với môi trường người sử dụng nhất (đặt tester vào vị trí của end-user).
* Test bao gồm: Rủi ro liên quan đến mô tả yêu cầu phần mềm, các quy trình hoạt động của phần mềm, các chức năng của phần mềm mà người sử dụng dùng đến.
* Kiểm tra các yêu cầu chức năng & phi chức năng của phần mềm.

## 6.2 Testcases

### Danh sách

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Tên test case** | **Đối tượng test** | **Ý nghĩa** |
| 1 | Đăng nhập | Màn hình đăng nhập | Kiểm tra việc login vào hệ thông & trao đổi dữ liệu vừa client & server. |
| 2 | Đăng ký | Màn hình đăng kí | Kiểm tra việc register vào hệ thông & trao đổi dữ liệu vừa client & server. |
| 3 | Thêm khóa học | Màn hình thêm khóa học | Kiểm tra việc thêm khóa học và lưu dữ liệu vào hệ thống |
| 4 | Thông báo | Màn hình thông báo | Kiểm tra học sinh có nhận được thông báo hay không |
| 5 | Nộp bài | Màn hình nộp bài | Kiểm tra bài nộp của học sinh có được lưu trên sever hay chưa |
| 6 | Chấm điểm | Màn hình chấm điểm | Kiểm tra việc chấm điểm của giáo viên |
| 7 | Cập nhật thông tin | Màn hình thông tin cá nhân | Kiểm tra hệ thống có lưu dữ liệu hay không |
| 8 | Đổi mật khẩu | Màn hình đổi mật khẩu | Kiểm tra việc thay đổi mật khẩu & trao đổi dữ liệu vừa client & server. |
| 9 | Hiệu năng, delay time | Website | Kiểm tra độ mượt của các thao tác |
| 10 | Test UI/UX | Website | Kiểm tra UI/UX, mức độ thân thiện, khả năng hiển thị trên nhiều loại thiết bị khác nhau |
| 11 | Stress Database | Database | Kiểm tra giới hạn đáp ứng của Database |

### Đặc tả

***Testcase #1***

|  |  |
| --- | --- |
| **Test case** |  |
| Related Use case | Đăng nhập vào Website |
| Context | Click vào button Đăng nhập |
| Input Data | Thông tin đăng nhập |
| Expected Output | Đăng nhập thành công hoặc thất bại (phải có phản hồi) |
| Test steps | - Nhập thông tin đăng nhập.  - Nhấn chọn Login. |

***Testcase #2***

|  |  |
| --- | --- |
| **Test case** |  |
| Related Use case | Đăng kí tài khoản mới |
| Context | Click vào button Đăng ký |
| Input Data | Thông tin đăng ký |
| Expected Output | Đăng ký thành công hoặc thất bại (phải có phản hồi).  Đăng ký thành công phải đăng nhập được. |
| Test steps | - Chọn Login.  - Nhập thông tin đăng ký.  - Nhấn chọn Login. |

***Testcase #3***

|  |  |
| --- | --- |
| **Test case** |  |
| Related Use case | Tạo khóa học |
| Context | Giáo viên tạo khóa học mới khi bắt đầu học kì |
| Input Data | Thông tin khóa học |
| Expected Output | Tạo thành công hoặc thất bại (phải có phản hồi)  Tạo thành công thì học sinh phải đăng ký được vào khóa học. |
| Test steps | - Click vào “thêm khóa học”  - Nhập thông tin khóa học  - Click “Tạo” |

***Testcase #4***

|  |  |
| --- | --- |
| **Test case** |  |
| Related Use case | Thông báo chung/ thông báo khóa học |
| Context | Thông báo được hiển thị khi người quản lí hoặc giáo viên tạo |
| Input Data | Nội dung thông báo |
| Expected Output | Tất cả học sinh phải nhận được thông báo nếu là thông báo chung, học sinh nhận được thông báo khóa học khi tham gia khóa học đó |
| Test steps | - Quản lý, giáo viên thực hiện tạo thông báo  - Nhập nội dung thông báo  - Click “Tạo” |

***Testcase #5***

|  |  |
| --- | --- |
| **Test case** |  |
| Related Use case | Thực hiện bài tập |
| Context | Khi học sinh muốn thực hiện bài tập mà giáo viên giao |
| Input Data | File bài nộp |
| Expected Output | Nộp bài thành công hay thất bại (phải có phản hồi).  Nộp thành công khi reload lại phải thấy bài đã nộp. |
| Test steps | - Chọn bài tập muốn làm  - Chọn “Choose file” để chon file nộp  - Xác nhận |

***Testcase #6***

|  |  |
| --- | --- |
| **Test case** |  |
| Related Use case | Quản lý học sinh |
| Context | Giáo viên nhận được bài nộp và chấm điểm |
| Input Data | File bài nộp của học sinh |
| Expected Output | Điểm số bài làm được lưu vào hệ thống |
| Test steps | - Vào được màn hình chấm điểm  - Chọn học sinh  - Chọn bài nộp  - Set điểm |

***Testcase #7***

|  |  |
| --- | --- |
| **Test case** |  |
| Related Use case | Thông tin |
| Context | Người dùng muốn thay đổi thông tin cá nhân |
| Input Data | Các thông tin muốn cập nhật |
| Expected Output | Thông báo thay đổi thành công  Reload lại trang, thông tin phải được giữ lại |
| Test steps | Vào trang cá nhân  Điền thông tin thay đổi  Nhấn lưu |

***Testcase #8***

|  |  |
| --- | --- |
| **Test case** |  |
| Related Use case | Đổi mật khẩu |
| Context | Người dùng muốn thay đổi mật khẩu |
| Input Data | Mật khẩu mới |
| Expected Output | Thông báo đổi thành công hay thất bại (phải có phản hồi)  Đổi thành công phải đăng nhập lại được |
| Test steps | - Vào màn hình thay đổi mật khẩu  - Nhập mật khẩu mới  - Nhấn thay đổi |

***Testcase #9 (System Test)***

|  |  |
| --- | --- |
| **Test case** | **Hiệu năng, delay time** |
| Context | Sử dụng trên nhiều trình duyệt khác nhau, thiết bị khác nhau |
| Input Data |  |
| Expected Output | Trang web load với tốc độ chấp nhận cho phép. |
| Test steps | Kiểm tra các Component. Đo thời gian phản hồi bằng Developer Tool. |

***Testcase #10 (System Test)***

|  |  |
| --- | --- |
| **Test case** | **UI/UX** |
| Context | Sử dụng trên nhiều trình duyệt khác nhau, thiết bị khác nhau |
| Input Data |  |
| Expected Output | Giao diện không bị rời rạc, long lẽo, mất cân đối |
| Test steps | - Truy cập website trên nhiều thiết bị.  - Đăng nhập  - Thực hiện các chức năng cho phép  - Kiểm tra font chữ, kích thước.  - Kiểm tra độ dãn của các đối tượng |

***Testcase #11 (System Test)***

|  |  |
| --- | --- |
| **Test case** | **Stress Database** |
| Context | Có nhiều Query gửi đến Database một lúc |
| Input Data | Query |
| Expected Output | Database vẫn xử lí được với thời gian cho phép |
| Test steps | - Tạo ra nhiều Query.  - Kiểm tra phản hồi của Database |

# Hướng phát triển

Hiện tại Website quan lý lớp học STEM cũng chỉ đang ở mức sơ khai, đáp ứng những nhu cầu cơ bản nhất đối với người dùng. Trong tương lai, nếu có điều kiện & khả năng, nhóm sẽ hoàn thiện thêm những tính năng như sau:

**1. Cải tiến giao diện:** chỉnh sửa, thiết kế lại giao diện để phù hợp với người dùng hơn.

**2. Phân hóa bộ phân quản lý:** hiện tại website chỉ có 1 cấp quản lý duy nhất, tương lai có thể chia việc quản lý thành từng cấp nhỏ hơn. VD: bộ phận quản lý thu chi, bộ phận quản lý nhân sự, bộ phận quản lý chương trình đào tạo

**3. Tính năng đánh giá môn học:** cho phép học sinh đánh giá môn học cuối học kì để quản lý có thể có những thay đổi phù hợp hơn về các giảng dạy cũng như chương trình đào tạo.

# Đánh giá đóng góp thành viên trong nhóm

|  |  |  |
| --- | --- | --- |
| Thành viên | Công việc đóng góp | Mức độ đóng góp |
| Lê Văn Thi | Quản lí dự án.  Xây dựng giao diện các màn hình  Xây dựng kiến trúc phần mềm | 28% |
| Trương Thành Danh | Code API các Sever | 19% |
| Huỳnh Duy Tân | Design giao diện bằng PTS  Cung cấp Testcases  Viết báo cáo | 19% |
| Nguyễn Quang Thạch | Phác thảo Class Diagram  Thiết kế Database | 19% |
| Lâm Khang Vĩ | Insert databse | 15% |