­Trường Đại học Bách Khoa Hà Nội

Viện Công nghệ Thông Tin và Truyền Thông

Báo cáo bài tập lớn

WEBSITE HỖ TRỢ HỌC TOEIC TRỰC TUYẾN

Môn: Công nghệ web tiên tiến

Hà Nội, 12/2018

Trường Đại học Bách Khoa Hà Nội

Viện Công nghệ Thông Tin và Truyền Thông

Báo cáo bài tập lớn

WEBSITE HỖ TRỢ HỌC TOEIC TRỰC TUYẾN

­­

**Nhóm 04** : Nguyễn Văn Đức 20141166

Phạm Minh Hiếu 20151359

Trần Sơn Tùng 20145145

Phạm Hải Linh 20142606

**GV hướng dẫn** : TS.Nguyễn Nhất Hải

Hà Nội, 12/20

# Mục lục

[Mục lục iii](#_Toc532532730)

[Danh mục hình vẽ vi](#_Toc532532731)

[Danh mục bảng vii](#_Toc532532732)

[Danh mục các từ viết tắt viii](#_Toc532532733)

[Danh mục thuật ngữ ix](#_Toc532532734)

[Chương 1 Giới thiệu đề tài 1](#_Toc532532735)

[1.1 Đặt vấn đề 1](#_Toc532532736)

[1.2 Định hướng giải pháp 2](#_Toc532532737)

[Chương 2 Phân tích yêu cầu 3](#_Toc532532738)

[2.1 Tổng quan chức năng 3](#_Toc532532739)

[2.1.1 Biểu đồ use case tổng quan 3](#_Toc532532740)

[2.1.2 Biểu đồ phân rã use case Học từ vựng 5](#_Toc532532741)

[2.1.3 Quy trình nghiệp vụ 6](#_Toc532532742)

[2.2 Đặc tả chức năng 9](#_Toc532532743)

[2.2.1 Đặc tả use case Đánh dấu từ vựng yêu thích 9](#_Toc532532744)

[2.2.2 Đặc tả use case Học theo trắc nghiệm từ vựng 10](#_Toc532532745)

[2.2.3 Đặc tả use case Học theo nghĩa của từ 12](#_Toc532532746)

[2.2.4 Đặc tả use case Học nghe từ 13](#_Toc532532747)

[2.2.5 Đặc tả use case Thi thử toeic 15](#_Toc532532748)

[2.2.6 Đặc tả use case Học online 16](#_Toc532532749)

[2.3 Các yêu cầu khác 17](#_Toc532532750)

[2.3.1 Về chức năng 17](#_Toc532532751)

[2.3.2 Tính dễ dùng 18](#_Toc532532752)

[2.3.3 Tính dễ bảo trì 18](#_Toc532532753)

[2.3.4 Hiệu năng 18](#_Toc532532754)

[2.3.5 Tính tin cậy 18](#_Toc532532755)

[2.3.6 An toàn thông tin 19](#_Toc532532756)

[Chương 3 Công nghệ sử dụng 20](#_Toc532532757)

[3.1 Node.js 20](#_Toc532532758)

[3.1.1 Khái niệm 21](#_Toc532532759)

[3.1.2 Đặc điểm nổi bật 21](#_Toc532532760)

[3.1.3 Ưu, nhược điểm 22](#_Toc532532761)

[3.1.4 Ứng dụng 22](#_Toc532532762)

[3.1.5 Sử dụng Node.js trong hệ thống 23](#_Toc532532763)

[3.2 ReactJS 24](#_Toc532532764)

[3.2.1 Khái niệm 24](#_Toc532532765)

[3.2.2 Đặc điểm nổi bật 24](#_Toc532532766)

[3.3 Tổng quan hệ thống 26](#_Toc532532767)

[3.4 Thiết kế chi tiết 27](#_Toc532532768)

[3.4.1 Thiết kế giao diện cho ứng dụng 27](#_Toc532532769)

[3.4.2 Thiết kế cơ sở dữ liệu 30](#_Toc532532770)

[Chương 4 Kết luận và hướng phát triển 31](#_Toc532532771)

[4.1 Kết luận 31](#_Toc532532772)

[4.1.1 Khó khăn trong khi làm sản phẩm 31](#_Toc532532773)

[4.1.2 Đánh giá chung sản phẩm 31](#_Toc532532774)

[4.1.3 Hướng phát triển 32](#_Toc532532775)

[Tài liệu tham khảo 33](#_Toc532532776)

# Danh mục hình vẽ

[**Hình 2‑1** Biểu đồ use case tổng quan 4](#_Toc532532430)

[**Hình 2‑2** Biểu đồ phân rã use case Học từ vựng. 5](#_Toc532532431)

[**Hình 2‑3** Quy trình học từ vựng của một thành viên. 6](#_Toc532532432)

[**Hình 2‑4** Quy trình thi thử. 7](#_Toc532532433)

[**Hình 2‑5** Quy trình học online. 8](#_Toc532532434)

[**Hình 3‑1** Biểu tượng Node.js 20](#_Toc532532435)

[**Hình 3‑2** Các công ty đang sử dụng Node.js trong việc triển khai các hệ thống. 23](#_Toc532532436)

[**Hình 3‑3** Biểu tượng React 24](#_Toc532532437)

[**Hình 3‑4** Tổng quan các thành phần hệ thống 26](#_Toc532532438)

[**Hình 3‑5** Biểu đồ dịch chuyển giữa các màn hình trong web site 27](#_Toc532532439)

[**Hình 3‑6** Màn hình danh sách chủ đề các từ vựng 27](#_Toc532532440)

[**Hình 3‑7** Màn hình học trắc nghiệm từ 28](#_Toc532532441)

[**Hình 3‑8** Màn hình học nghĩa từ 28](#_Toc532532442)

[**Hình 3‑9** Màn hình học online 29](#_Toc532532443)

[**Hình 3‑10** Màn hình thi thử 29](#_Toc532532444)

# Danh mục bảng

[**Bảng 2‑1** Đặc tả use case Đánh dấu từ vựng yêu thích 9](#_Toc532532446)

[**Bảng 2‑2** Đặc tả use case Học theo trắc nghiệm từ vựng 10](#_Toc532532447)

[**Bảng 2‑3** Đặc tả use case Học theo nghĩa của từ 12](#_Toc532532448)

[**Bảng 2‑4** Đặc tả use case Học nghe từ 13](#_Toc532532449)

[**Bảng 2‑5** Bảng đặc tả use case Thi thử toeic 15](#_Toc532532450)

# Danh mục các từ viết tắt

|  |  |
| --- | --- |
| API | Application Programming Interface - Giao diện lập trình ứng dụng |
| **CRUD** | Creat, Read, Update, Delete chỉ việc tạo mới, xem, cập nhật, và xóa |
| **REST** | REpresentational State Transfer – Một dạng chuyền đổi cấu trúc dữ liệu |
| **DOM** | Document Object Model – Mô hình đối tượng tài liệu |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Danh mục thuật ngữ

|  |  |
| --- | --- |
| Server | Máy chủ |
| **Web Admin** | Công cụ quản trị hệ thống website |
| **Web Service** | Dịch vụ web |
| **Framework** | Là một thư viện bao gồm các lớp đã được xây dựng hoàn chỉnh, bộ khung để phát triển các phần mềm ứng dụng |
|  |  |
|  |  |
|  |  |

# Giới thiệu đề tài

## Đặt vấn đề

Thế giới ngày càng hội nhập, mở rộng quan hệ hợp tác giữa các quốc gia vì mục tiêu chung mang lại nhiều lợi ích nhất cho các nước. Nằm trong phạm vi quốc tế, Việt Nam cũng không nằm ngoài quy luật đó. Do vậy có thể thấy vai trò ngày càng quan trọng của tiếng Anh, nó được coi là chìa khóa vàng cho mọi thành công. Chính vì vậy cụm từ tiếng Anh TOEIC hiện nay ngày càng được sử dụng rộng rãi, phổ biến và là nhiệm vụ của nhiều người là phải hoàn thành kỳ thi TOEIC. TOEIC đến Việt Nam và ngày càng phát triển nhanh chóng.

Tuy nhiên việc học tiếng anh nói chung và toeic nói riêng của đa phần các bạn sinh viên còn nhiều khó khăn đặc biệt là các bạn sinh viên kĩ thuật. Có rất nhiều nguyên nhân khiến việc học toeic trở nên khó như là lười, không đủ kiên trì,… nhưng phần lớn là do chưa có một phương pháp học hiệu quả. Hiện nay có rất nhiều cách thức học và phương tiện để học phổ biến hiện nay như là mua sách học toeic, tham gia các khóa học offline ở trung tâm hay những khóa học trên mang. Tuy nhiên có một nhược điểm chung là hầu hết đều rất tốn kém mà hiệu quả chưa thực sự cao.

Website hỗ trợ học toeic giúp cho việc học toeic trở trên dễ dàng hơn, và tiết kiêm chi phí hơn. Website cung cấp hơn 600 từ vựng toeic, gần 20 khóa học các loại từ cơ bản đến nâng cao từ các chuyên gia về toeic hàng đầu với chi phí phù hợp sẽ giúp giải quyết vấn đề ở trên.

## Định hướng giải pháp

Trên cơ sở đánh giá được trình bày ở mục 1.1, giải pháp đặt ra ở đây sẽ được nghiên cứu và thực hiện trong các bước chính sau.

**Bước 1**: Xây dựng website học toeic với một ngân hàng các từ vựng, các đề thi thử toeic được tổng hợp từ những cuốn sách kinh điển theo lộ trình học. Ngoài ra hệ thống sẽ đưa lên một số khóa học toeic online để người dùng có thể học nếu như thời gian không đủ để đi học thêm tại các trung tâm.

Cụ thể hệ thống bao gồm: (i) một server bằng Nodejs để cung cấp dữ liệu cho website, (ii) một website cho người dùng và một web admin để quản lý dữ liệu trên website học toeic. (iii) một web service để phục vụ truyền dữ liệu giữa server và client.

**Bước 2**: Liên kết với các trường đại học để có thể mở thêm các mục ôn thi hay thi thử dành cho sinh viên tại trường đó. Các bạn sinh viên trường khác cũng có thể tham khảo đề của các trường mình không học.

# Phân tích yêu cầu

## Tổng quan chức năng

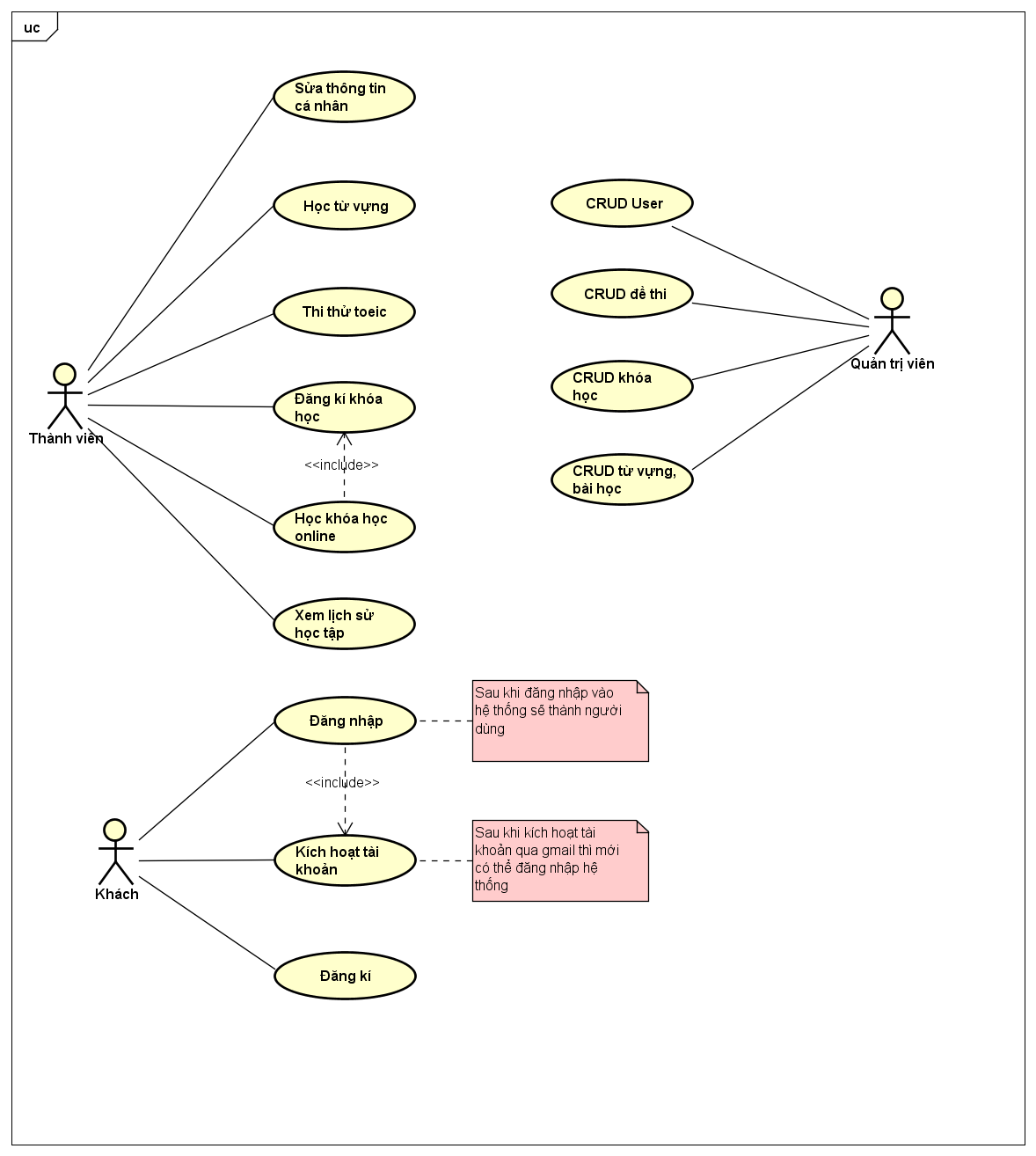
### Biểu đồ use case tổng quan

Hình 1 là biểu đồ use case tổng quan của hệ thống. Hệ thống có 3 tác nhân bao gồm: (i) Thành viên, (ii) Quản trị viên, (iii) Khách. Vai trò của các tác nhân cụ thể:

**Quản trị viên** là một người dùng sau khi đã đăng nhập của hệ thống, sẽ có nhiệm vụ quản lý server về các dữ liệu như: (i) thêm, sửa, xóa, tìm kiếm các thông tin về người dùng website, (ii) thêm, sửa, xóa, cập nhật các thông tin về các đề thi thử, (iii) thêm, sửa, xóa, cập nhật các thông tin về các khóa học online hiện có, (iv) thêm, sửa, xóa, cập nhật các thông tin về từ vựng, chủ đề.

**Khách** là người dùng của ứng dụng khi chưa đăng nhập, có thể xem thông tin tổng quan về trang web, đăng kí tài khoản, kích hoạt tài khoản qua gmail.

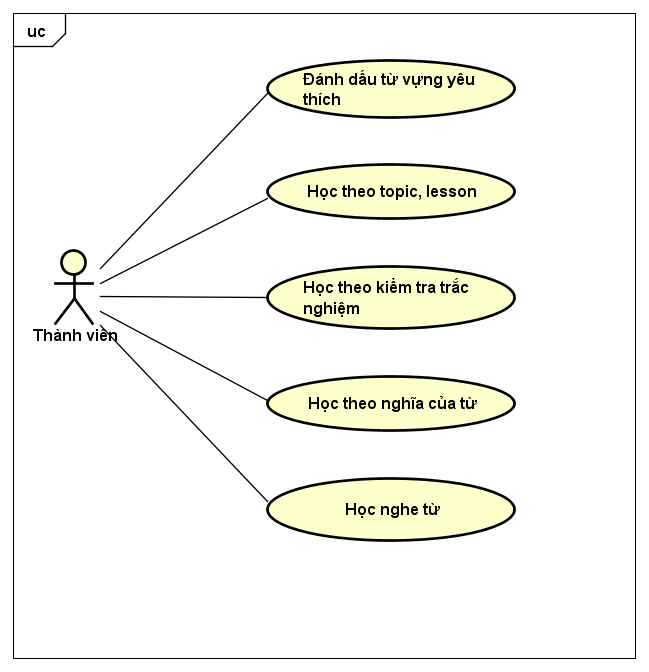
**Thành viên** là người dùng khi đã đăng nhập vào hệ thống, có thể học từ vựng, thi thử toeic, đăng khí và tham gia học các khóa học online, xem lịch sử học tập, cập nhật thông tin cá nhân.



**Hình 2‑1** Biểu đồ use case tổng quan

### Biểu đồ phân rã use case Học từ vựng

Hình 2 mô tả biểu đồ phân rã use case Học từ vựng. Người dùng khi sử dụng website có thể học từ vựng theo các chủ đề, đánh dấu các từ yêu thích, học theo các phương pháp như trắc nghệm từ, học nghe từ, hay nghĩa của từ.

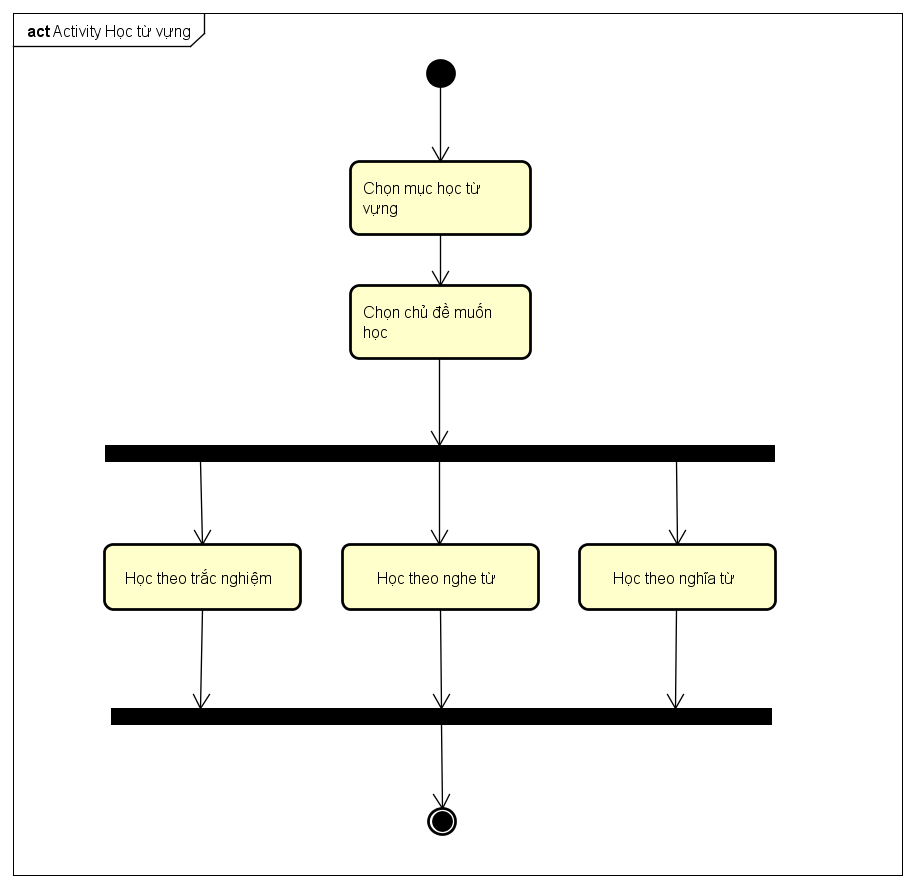


**Hình 2‑2** Biểu đồ phân rã use case Học từ vựng.

### Quy trình nghiệp vụ

#### Quy trình học từ vựng

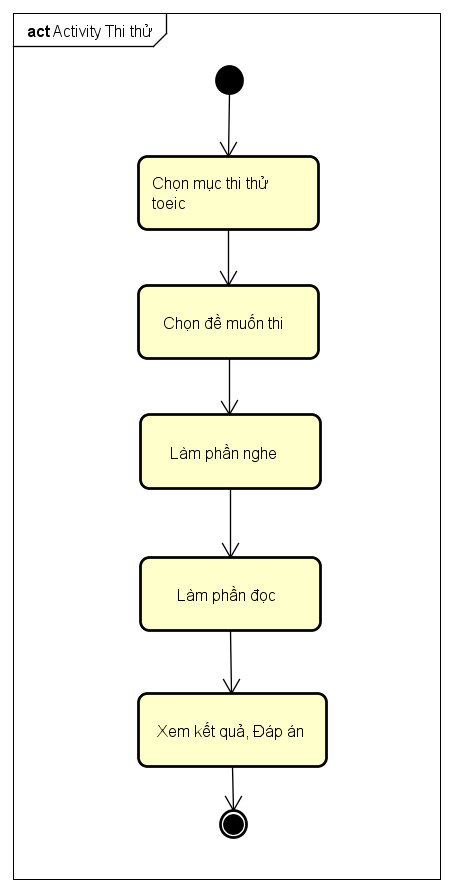
Thành viên sau khi vào website đăng nhập chọn mục học từ vựng có thể tham gia học từ vựng ngay lập tức thông qua các chủ đề. Chi tiết của quy trình được mô tả như Hình 2.3.



**Hình 2‑3** Quy trình học từ vựng của một thành viên.

#### Quy trình thi thử

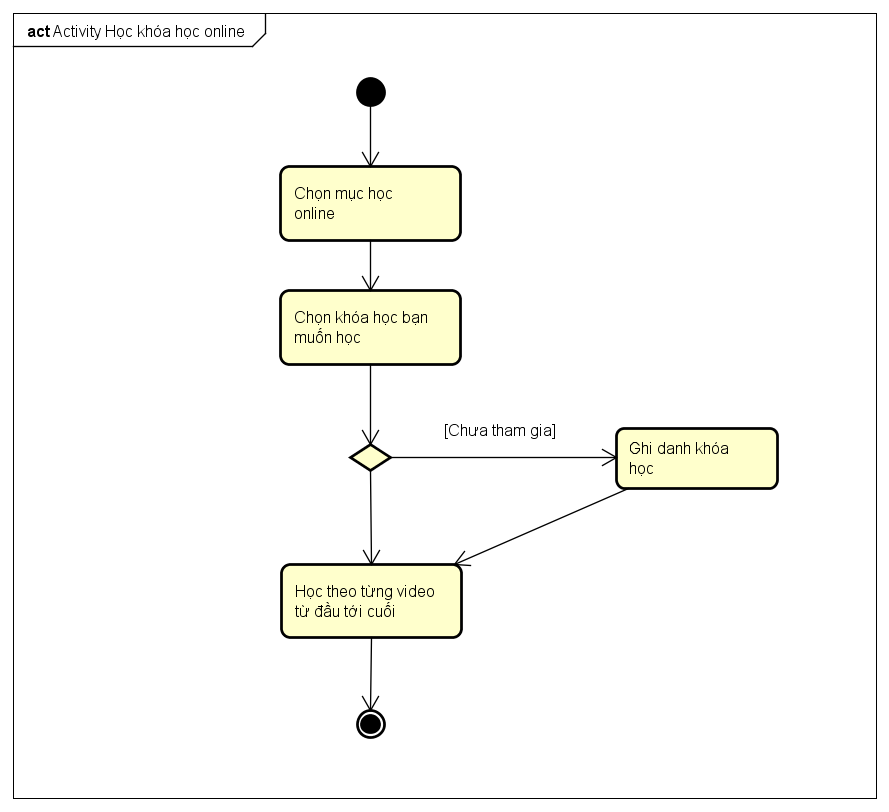
Thành viên sau khi đăng nhập vào chọn mục thi thử để có thể đánh giá năng lực qua các bài thi. Chi tiết quy trình được mô tả trong [hình 2-4](#_Quy_trình_thi).



**Hình 2‑4** Quy trình thi thử.

#### Quy trình học online

Thành viên sau khi đăng nhập vào chọn mục khóa học online để có thể tham gia các khóa học từ cơ bản đến nâng cao. Chi tiết quy trình được mô tả trong hình 2-5



**Hình 2‑5** Quy trình học online.

## Đặc tả chức năng

### Đặc tả use case Đánh dấu từ vựng yêu thích

Bảng 2.1 đặc tả chi tiết use case Đánh dấu từ vựng yêu thích với tác nhân là Thành viên (khách đã đăng nhập).

**Bảng 2‑1** Đặc tả use case Đánh dấu từ vựng yêu thích

|  |  |  |
| --- | --- | --- |
| **Tên use case** | Đánh dấu từ vựng yêu thích | |
| **Tác nhân** | Thành viên (Khách đã đăng nhập) | |
| **Tiền điều kiện** | Người dùng đã đăng nhập vào website | |
| **Luồng sự kiện chính** | | |
| **STT** | **Thực hiện** | **Hành động** |
| 1 | Người dùng | Mở website. |
| 2 | Hệ thống | Hiển thị giao diện Trang chủ của web site với mục “Học từ vựng” trên phần tiêu đề ngay đầu web. |
| 3 | Người dùng | Chọn mục Học từ vựng. |
| 4 | Hệ thống | Hiển thị ra danh sách các chủ đề với các bài học tương ứng theo từng hàng. Hàng đầu sẽ là những chủ đề được học nhiều và phổ biến. |
| 5 | Người dùng | Chọn một chủ đề bất kì |
| 6 | Hệ thống | Hiển thị ra màn hình chi tiết về chủ đề bao gồm phần đầu flashcard các từ. Sau đó là danh sách các từ được sắp xếp theo mức độ học quên nhiều hay test sai nhiều của người dùng cho đến những từ chưa học |
| 7 | Người dùng | Click dấu sao màu xám để đánh dấu thích hoặc dấu sao màu vàng để bỏ thích. |
| **Luồng sự kiện thay thế** | | |
| **STT** | **Thực hiện** | **Hành động** |
| 4a | Hệ thống | Thông báo lỗi: Không thể hiển thị dữ liệu |
| **Hậu điều kiện** | Không. | |

### Đặc tả use case Học theo trắc nghiệm từ vựng

Bảng 2-2 đặc tả chi tiết use case Học theo trắc nghiệm từ vựng.

**Bảng 2‑2** Đặc tả use case Học theo trắc nghiệm từ vựng

|  |  |  |
| --- | --- | --- |
| **Tên use case** | Học theo trắc nghiệm từ vựng | |
| **Tác nhân** | Thành viên (Khách đã đăng nhập) | |
| **Tiền điều kiện** | Người dùng đang trong màn hình chi tiết chủ đề từ vựng nào đó. | |
| **Luồng sự kiện chính** | | |
| **STT** | **Thực hiện** | **Hành động** |
| 1 | Người dùng | Chọn Card “Trắc nghiệm” trên màn hình. |
| 2 | Hệ thống | Hiển thị giao diện học theo trắc nghiệm với câu hỏi đầu tiên liên quan đến từ đầu tiên. |
| 3 | Người dùng | Chọn đáp án cho câu hỏi trắc nghiệm. |
| 4 | Hệ thống | Hiển thị kết quả đúng hay sai cho từ vựng đó và đáp án ở bên dưới câu hỏi. Ở cột bên trái hiển thị kết quả số từ vựng trả lời đúng hay sai trên tổng số từ vựng có trong chủ đề. |
| 5 | Người dùng | Chọn tiếp theo để có thể chuyển sang từ vựng tiếp theo trong chủ đề rồi quay lại bước 2. |
| 6 | Hệ thống | Lưu dữ liệu trả lời đùng hay sai vào Database để tiện cho việc theo dõi học tập của người dùng. |
| **Luồng sự kiện thay thế** | | |
| **STT** | **Thực hiện** | **Hành động** |
| 2a | Hệ thống | Thông báo lỗi: Không thể hiển thị dữ liệu |
| **Hậu điều kiện** | Không. | |

### Đặc tả use case Học theo nghĩa của từ

Bảng 2-3 đặc tả chi tiết use case Học theo nghĩa của từ

**Bảng 2‑3** Đặc tả use case Học theo nghĩa của từ

|  |  |  |
| --- | --- | --- |
| **Tên use case** | Học theo trắc nghiệm từ vựng | |
| **Tác nhân** | Thành viên (Khách đã đăng nhập) | |
| **Tiền điều kiện** | Người dùng đang trong màn hình chi tiết chủ đề từ vựng nào đó. | |
| **Luồng sự kiện chính** | | |
| **STT** | **Thực hiện** | **Hành động** |
| 1 | Người dùng | Chọn Card “Điền từ” trên màn hình. |
| 2 | Hệ thống | Hiển thị giao diện học với một input text để người dùng nhập câu trả lời và một label đưa ra nghĩa của từ bằng Tiếng Việt hoặc giải thích ý nghĩa của từ bằng tiếng anh. |
| 3 | Người dùng | Điền đáp án rồi chọn trả lời. |
| 4 | Hệ thống | Hiển thị kết quả đúng hay sai cho từ vựng đó và đáp án ở bên dưới câu hỏi. Ở cột bên trái hiển thị kết quả số từ vựng trả lời đúng hay sai trên tổng số từ vựng có trong chủ đề. |
| 5 | Người dùng | Chọn tiếp theo để có thể chuyển sang từ vựng tiếp theo trong chủ đề rồi quay lại bước 2. |
| 6 | Hệ thống | Lưu dữ liệu trả lời đùng hay sai vào Database để tiện cho việc theo dõi học tập của người dùng. |
| **Luồng sự kiện thay thế** | | |
| **STT** | **Thực hiện** | **Hành động** |
| 2a | Hệ thống | Thông báo lỗi: Không thể hiển thị dữ liệu |
| **Hậu điều kiện** | Không. | |

### Đặc tả use case Học nghe từ

Bảng 2-4 đặc tả chi tiết use case Học nghe từ.

**Bảng 2‑4** Đặc tả use case Học nghe từ

|  |  |  |
| --- | --- | --- |
| **Tên use case** | Học nghe từ | |
| **Tác nhân** | Thành viên (Khách đã đăng nhập) | |
| **Tiền điều kiện** | Người dùng đang trong màn hình chi tiết chủ đề từ vựng nào đó. | |
| **Luồng sự kiện chính** | | |
| **STT** | **Thực hiện** | **Hành động** |
| 1 | Người dùng | Chọn Card “Học nghe từ” trên màn hình. |
| 2 | Hệ thống | Hiển thị giao diện học với một input text để người dùng nhập câu trả lời và một icon loa để người dùng click vào đó để nghe từ cần điền. Đồng thời ngay khi vào câu hỏi thì âm của từ sẽ tự động phát ra. |
| 3 | Người dùng | Điền đáp án rồi chọn trả lời. |
| 4 | Hệ thống | Hiển thị kết quả đúng hay sai cho từ vựng đó và đáp án ở bên dưới câu hỏi. Ở cột bên trái hiển thị kết quả số từ vựng trả lời đúng hay sai trên tổng số từ vựng có trong chủ đề. |
| 5 | Người dùng | Chọn tiếp theo để có thể chuyển sang từ vựng tiếp theo trong chủ đề rồi quay lại bước 2. |
| 6 | Hệ thống | Lưu dữ liệu trả lời đùng hay sai vào Database để tiện cho việc theo dõi học tập của người dùng. |
| **Luồng sự kiện thay thế** | | |
| **STT** | **Thực hiện** | **Hành động** |
| 2a | Hệ thống | Thông báo lỗi: Không thể hiển thị dữ liệu |
| **Hậu điều kiện** | Không. | |

### Đặc tả use case Thi thử toeic

Bảng 2-5 đặc tả chi tiết use case Thi thử toeic

Bảng 2‑5 Bảng đặc tả use case Thi thử toeic

|  |  |  |
| --- | --- | --- |
| **Tên use case** | Đặc tả use case Thi thử toeic | |
| **Tác nhân** | Người dùng | |
| **Tiền điều kiện** | Đã đăng nhập | |
| **Luồng sự kiện chính** | | |
| **STT** | **Thực hiện** | **Hành động** |
| 1 | Người dùng | Chọn Thi thử trên thanh tiêu đề |
| 2 | Hệ thống | Hiển thị giao diện danh sách các đề thi thử cho người dùng chọn. |
| 3 | Người dùng | Chọn lấy một đề thi thử bất kì. |
| 4 | Hệ thống | Hiển thị Dialog xác nhận thi thử của người dùng gồm hai nút “Bắt đầu” và “Thoát” |
| 5 | Người dùng | Chọn “Bắt đầu” rồi làm phần nghe trước. Sau 45’ sẽ chuyển sang phần đọc. |
| 6 | Người dùng | Khi hết giờ người dùng nhấn “Nộp bài” |
| 7 | Hệ thống | Hiển thị lên kết quả và đáp án |
| **Luồng sự kiện thay thế** | | |
| **STT** | **Thực hiện** | **Hành động** |
| 2a | Hệ thống | Thông báo lỗi: Không thể hiển thị dữ liệu |
| **Hậu điều kiện** | Không. | |

### Đặc tả use case Học online

|  |  |  |
| --- | --- | --- |
| **Tên use case** | Đặc tả use case Học online | |
| **Tác nhân** | Người dùng | |
| **Tiền điều kiện** | Đã đăng nhập | |
| **Luồng sự kiện chính** | | |
| **STT** | **Thực hiện** | **Hành động** |
| 1 | Người dùng | Chọn “Học khóa học” trên thanh tiêu đề |
| 2 | Hệ thống | Hiển thị giao diện danh sách các khóa học cho người dùng chọn |
| 3 | Người dùng | Chọn lấy một khóa học bất kì. |
| 4 | Hệ thống | Hiển thị thông tin khóa học bao gồm danh sách các video và trạng thái học của người dùng đó |
| 5 | Người dùng | Chọn lấy một video để học |
| 6 | Hệ thống | Hiển thị màn hình học online và play luôn từ video mà người dùng chọn. |
| **Luồng sự kiện thay thế** | | |
| **STT** | **Thực hiện** | **Hành động** |
| 2a | Hệ thống | Thông báo lỗi: Không thể hiển thị dữ liệu |
| 5a | Hệ thống | Thông báo lỗi: Không tìm thấy video |
| **Hậu điều kiện** | Không. | |

## Các yêu cầu khác

### Về chức năng

Trong các chuỗi sự kiện của các use case, tất cả các bước có thao tác với mạng internet, nếu có lỗi trong quá trình kết nối hoặc thao tác, cần có thông báo lỗi rõ ràng, tương ứng để tác nhân biết là lỗi liên quan đến mạng chứ không liên quan tới lỗi của người dùng, tránh việc người dùng hiểu nhầm về việc hệ thống gặp vấn đề.

Tương tự với các bước thao tác với CSDL phải thông báo lỗi rõ ràng.

### Tính dễ dùng

Các chức năng cần được thiết kế dễ dùng, thao tác với người sử dụng, cụ thể:

Về các chức năng của Quản trị viên phải được thiết kế dễ nhận biết và thao tác, cần có các thông báo cụ thể về lỗi sai thuộc về hệ thống hay người dùng. Giao diện đươc thiết kế phải đơn giản, phù hợp.

Về các chức năng dành cho Khách và Thành viên, cụ thể ứng dụng được thiết kế phải đảm bảo các yêu cầu : (i) Đơn giản, bắt mắt, sử dụng đồng nhất một ngôn ngữ thiết kế, màu xanh làm chủ đạo, font chữ, cỡ chữ được quy định rõ ràng; (ii) các chức năng phải đẽ phát hiện, thao tác, người dùng phải nhanh chóng tìm được các thông tin mình cần trong ứng dụng.

### Tính dễ bảo trì

Hệ thống sẽ được module hóa với từng chức năng của từng tác nhân, sử dụng thuần nhất một kiến trúc trong khi triển khai. Đảm bảo khi một phần của hệ thống gặp vấn đề không ảnh hưởng tới các thành phần khác

### Hiệu năng

Ứng dụng được thiết kế cho tác nhân Khách và Thành viên ngoài đảm bảo các yêu cầu về tính thẩm mỹ còn phải đảm bảo về mặt trải nghiệm người , chẳng hạn như : (i) ứng dụng có độ phản hồi cao, yêu cầu về thời gian không phản hồi tối thiểu là 3s, (ii) trong quá trình sử dụng trải nghiệm phải mượt mà, hạn chế giật, đơ… , (iii) tài nguyên lưu trữ, chiếm dụng của ứng dụng không được quá lớn.

### Tính tin cậy

Đảm bảo các dữ liệu được cung chính xác nhất. Dữ liệu người dùng sẽ được thiết kế và lưu trữ khoa học, tránh mất mát.

### An toàn thông tin

**Toàn vẹn**: Không cho phép tên tài khoản và mật khẩu đăng nhập được truyền trên môi trường mạng mà không được mã hóa.

**Xác thực**: Hệ thống hỗ trợ việc xác thực bằng tên tài khoản, mật khẩu đăng nhập.

# Công nghệ sử dụng

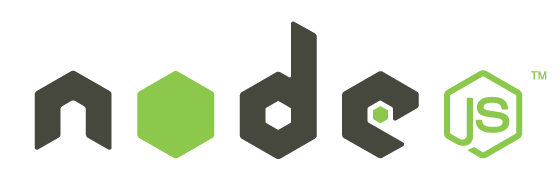
Như đã trình bày ở mục 1.3, hệ thống được xây dựng bao gồm các thành phần: (i) một server bằng Nodejs để cung cấp, xử lý dữ liệu cho website, (ii) một website cho người dùng và một web admin để quản lý dữ liệu trên website học toeic. (iii) một web service để phục vụ truyền dữ liệu giữa server và client.

Các công nghệ được áp dụng trong hệ thống được trình bày cụ thể ở bên dưới.

## Node.js

Node.js được sử dụng để duy trì server và viết web service sử dụng framework expressjs.

Hình 3.1 chính là biểu tượng của Node.js



**Hình 3‑1** Biểu tượng Node.js

### Khái niệm

Node.js là 1 nền tảng hay 1 môi trường được sử dụng để xây dựng, và phát triển các ứng dụng web, ứng dụng mạng, công cụ, thư viện... Node.js được xây dựng dựa trên Javascript Engine (V8 Engine) và được phát triển bởi Ryan Dahl năm 2009.

Node.js là một mã nguồn mở, đa nền tảng cho phát triển các ứng dụng phía Server và các ứng dụng liên quan đến mạng. Ứng dụng Node.js được viết bằng Javascript và có thể chạy trong môi trường Node.js trên nhiều hệ điều hành khác nhau như Window, Linux…

### Đặc điểm nổi bật

Các đặc điểm nổi bật của Node.js gồm:

**Thứ nhất**: không đồng bộ và phát sinh sự kiện trong quá trình sử dụng. Tất cả tập tin của Node.js sẽ không được đồng bộ hóa với nhau mà chủ yếu hoạt động trên nền của Node.js Sever cũng như chờ dữ liệu được trả về từ Sever.

**Thứ hai**: tốc độ thực thi chương trình rất nhanh do được xây dựng dựa trên nền tảng V8 Javascript Engine của Google Chrome.

**Thứ ba**: khả năng nhận và xử lý nhiều kết nối chỉ với một single-thread, vì thế nó sẽ giúp cho hệ thống của bạn tốn ít RAM hơn cũng như tăng tốc độ sử lý.

**Thứ tư**: hỗ trợ JSON API cùng cơ chế event-driven, non-blogking I/O (Input/Output). Mô hình này kết hợp với javascript tạo nên sự chọn lựa tuyệt vời nhất dành cho mọi dịch vụ Web làm bằng JSON.

### Ưu, nhược điểm

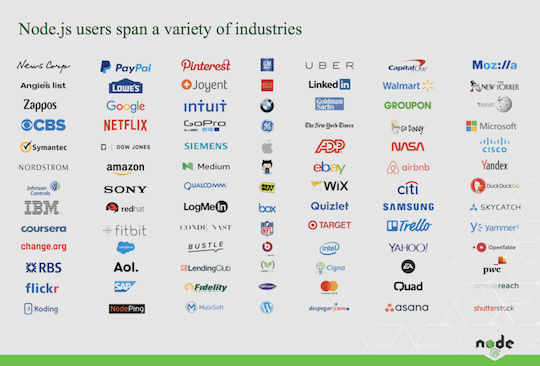
Về nhược điểm: (i) Các ứng dụng viết bới Node.js khá nặng và tốn tài nguyên, nếu cần xử lý các ứng dụng tốn tài nguyên CPU như encoding video, convert file…thì không nên dùng Node.js; (ii) Node.js có tính ổn định không cao.

Về ưu điểm: (i) Node.js chạy đa nền tảng phía Server, sử dụng kiến trúc hướng sự kiện Event-driven, cơ chế non-blocking I/O làm cho nó nhẹ và hiệu quả; (ii) Có thể chạy ứng dụng Node.js ở bất kỳ đâu trên các hệ điều hành khác nhau; (iii) Cộng đồng Node.js rất lớn và Node.js hoàn toàn miễn phí; (iv) Các ứng dụng Node.js đáp ứng tốt thời gian thực và chạy đa nền tảng, đa thiết bị; (v) Node.js sẽ tận dụng tối đa Unix để hoạt động, có thể xử lý hàng nghìn process và trả ra 1 luồng khiến cho hiệu xuất hoạt động đạt mức tối đa nhất có thể.

### Ứng dụng

Các ứng dụng thường được xây dựng bởi Node.js bao gồm: (i) Các máy chủ web socket như là Chat, Game Server…; (ii) Các chương trình upload file tốc độ cao; (iii) Các dịch vụ đám mây; (iv) RESTful API…

Hiện nay, rất nhiều công ty lớn đang sử dụng Node.js như Amazon, Ebay, Linkedin, Microsoft, Paypal, trello, Uber và còn nhiều cái tên nổi tiếng khác nữa, như Hình 12



**Hình 3‑2** Các công ty đang sử dụng Node.js trong việc triển khai các hệ thống.

### Sử dụng Node.js trong hệ thống

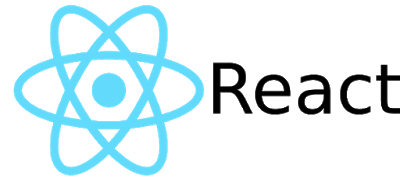
Với tất cả những tìm hiểu và nghiên cứu về ưu, nhược điểm, cũng như những gì mà Node.js làm được, nhóm em đã quyết định chọn nền tảng này làm môi trường để triển khai server với nhiệm vụ chính là xử lý dữ liệu liên quan đến việc học toeic, cung cấp web service để truyền dữ liệu giữa client và server.

Nodejs có cộng đồng hỗ trợ cực lớn, việc tải các module, package hỗ trợ các chức năng khác nhau rất dễ dàng. Cùng với đó có rất nhiều bài viết hướng dẫn khác nhau về cách sử dụng Nodejs.

Server đã được xây dựng trên môi trường Nodejs với việc sử dụng các package chính như: (i) expressjs hỗ trợ cấu trúc code, dễ dàng tạo các API từ các phương thức GET, POST, PUT, DELETE theo tiêu chuẩn của RESTful API (iv) thư viện mysql giúp làm việc với hệ quản trị cơ sở dữ liệu mysql.

## ReactJS

ReactJS dùng để viết UI cho website và web admin.



**Hình 3‑3** Biểu tượng React

### Khái niệm

React được Jordan Walke viết ra lần đầu tiên. Sau đó được sử dụng trên trang Newsfeed của Facebook (năm 2011) và Instagram (2012). Đến 05/2013 thì trở thành Open-source và được bảo hộ, phát triển bởi Facebook và Instagram.

React là một thư viện Javascript dùng để xây dựng giao diện người dùng.

### Đặc điểm nổi bật

Có 3 đặc điểm chính để nói về React đó là:

**Thứ nhất**: Just the UI: Chỉ sử dụng React để xây dựng giao diện người dùng, hiểu đơn giản thì trong mô hình MVC, React được coi như là phần V. Vì vậy nếu muốn xây dựng một trang web hoàn chỉnh thì không thể dùng mỗi React được, mà nó cần thêm phần M và C nữa.

**Thứ hai**: Virtual DOM: công nghệ DOM ảo giúp tăng hiệu năng cho ứng dụng.

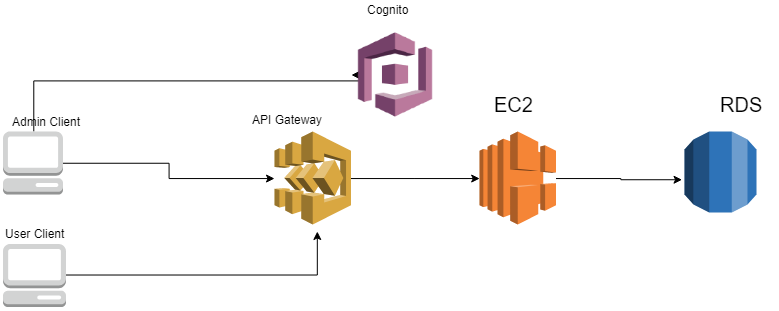
**Thứ ba**: Data Flow: React sử dụng luồng dữ liệu 1 chiều giúp kiểm soát dữ liệu dễ dàng hơn.

Với các đặc điểm ở trên, React dùng để xây dựng các ứng dụng lớn mà dữ liệu của chúng thay đổi liên tục theo thời gian. Dữ liệu thay đổi thì hầu hết kèm theo sự thay đổi về giao diện.

**Thứ tư:** Mọi thứ trong React đều là JavaScript Component và nó sử dụng JSX (JavaScript + HTML là một dạng ngôn ngữ cho phép viết các mã HTML trong Javascript) có đặc điểm nhanh hơn, an toàn hơn, dễ dàng hơn.

**Thứ năm**: Trong React sử dụng khái niệm Component. Mọi thứ trong React đều có thể coi là Component. Việc này giúp cho React thực hiện tốt các nguyên tắc trong OOP như: Composable( Đóng gói ), Reusable(Tái sử dụng), Maintainable (dễ bảo trì), Testable(dễ dàng cho test và fix bug).

## Tổng quan hệ thống



**Hình 3‑4** Tổng quan các thành phần hệ thống

**Server**:

**Web Admin:**

**User Client:**

**Api Gateway**:

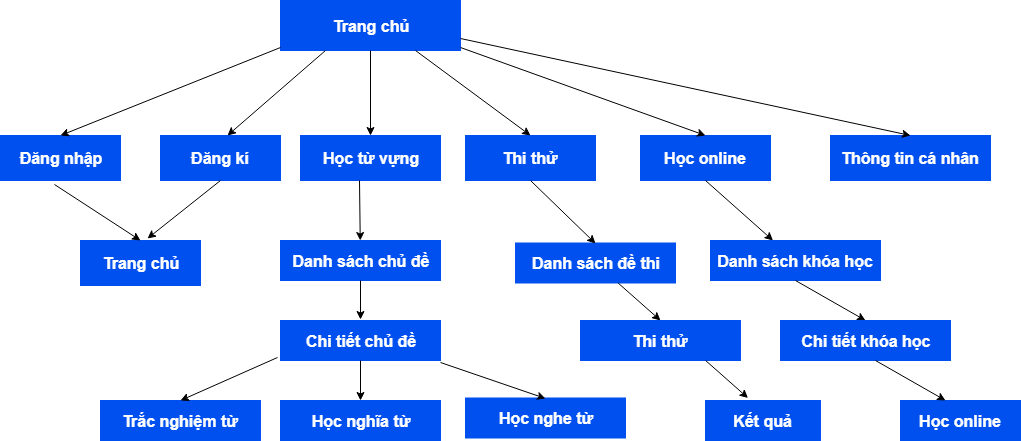
**Cognito:**

**EC2:**

**RDS:**

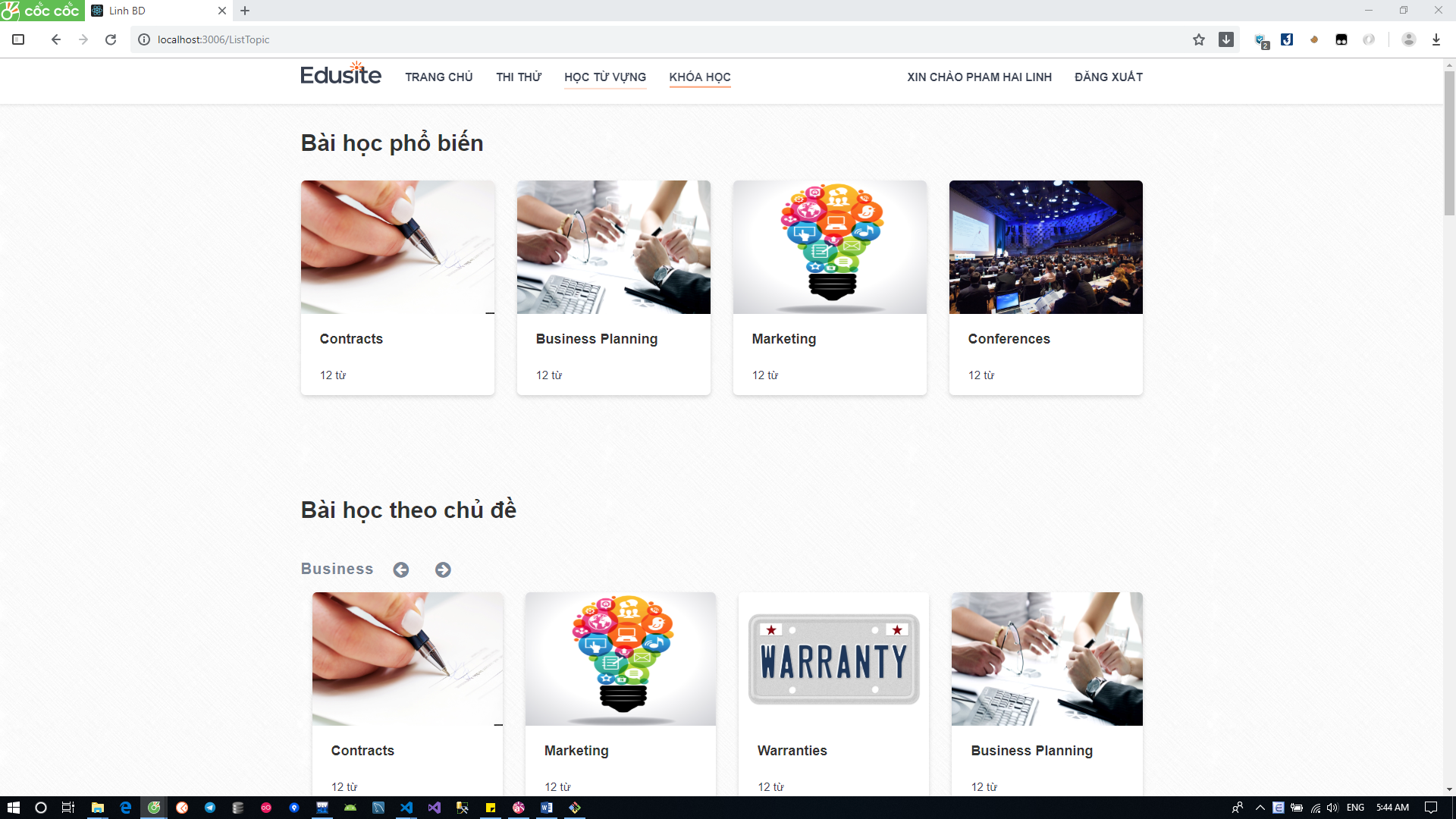
## Thiết kế chi tiết

### Thiết kế giao diện cho ứng dụng



**Hình 3‑5** Biểu đồ dịch chuyển giữa các màn hình trong web site

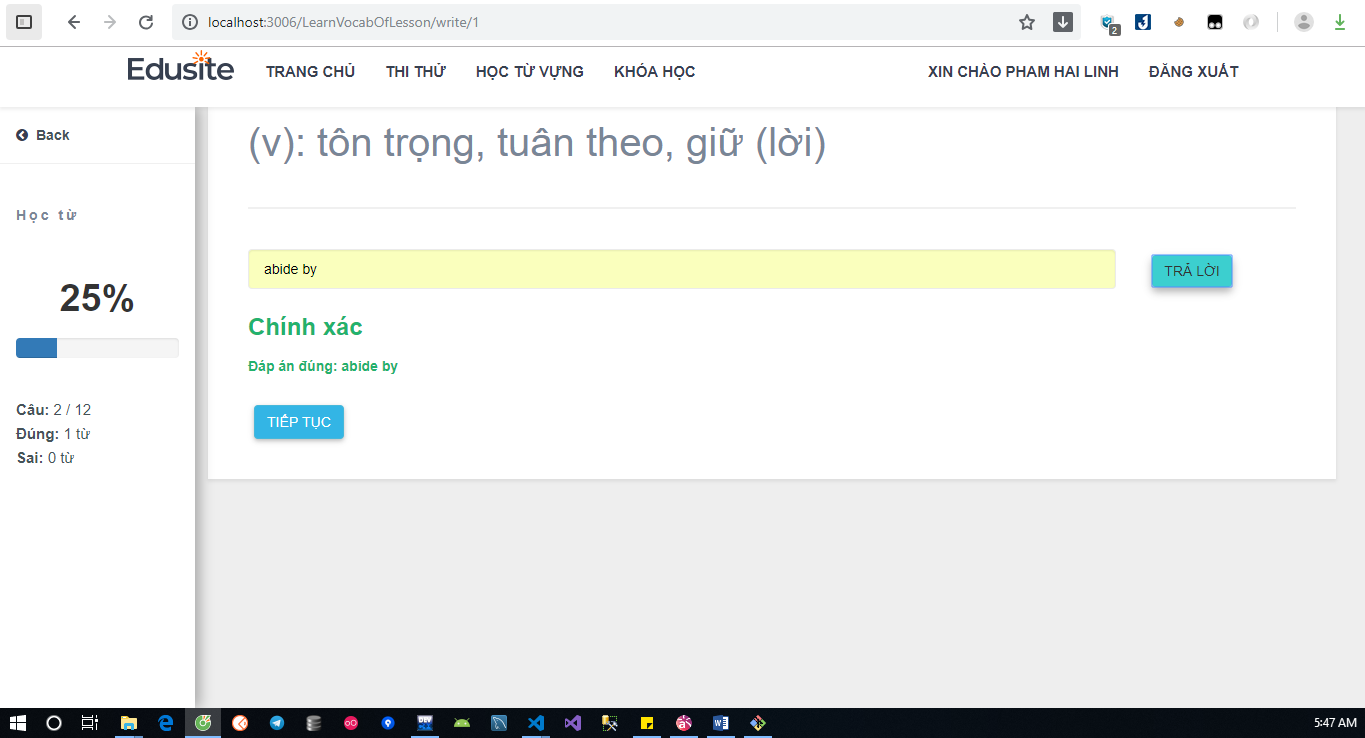
Một số màn hình chính của web:



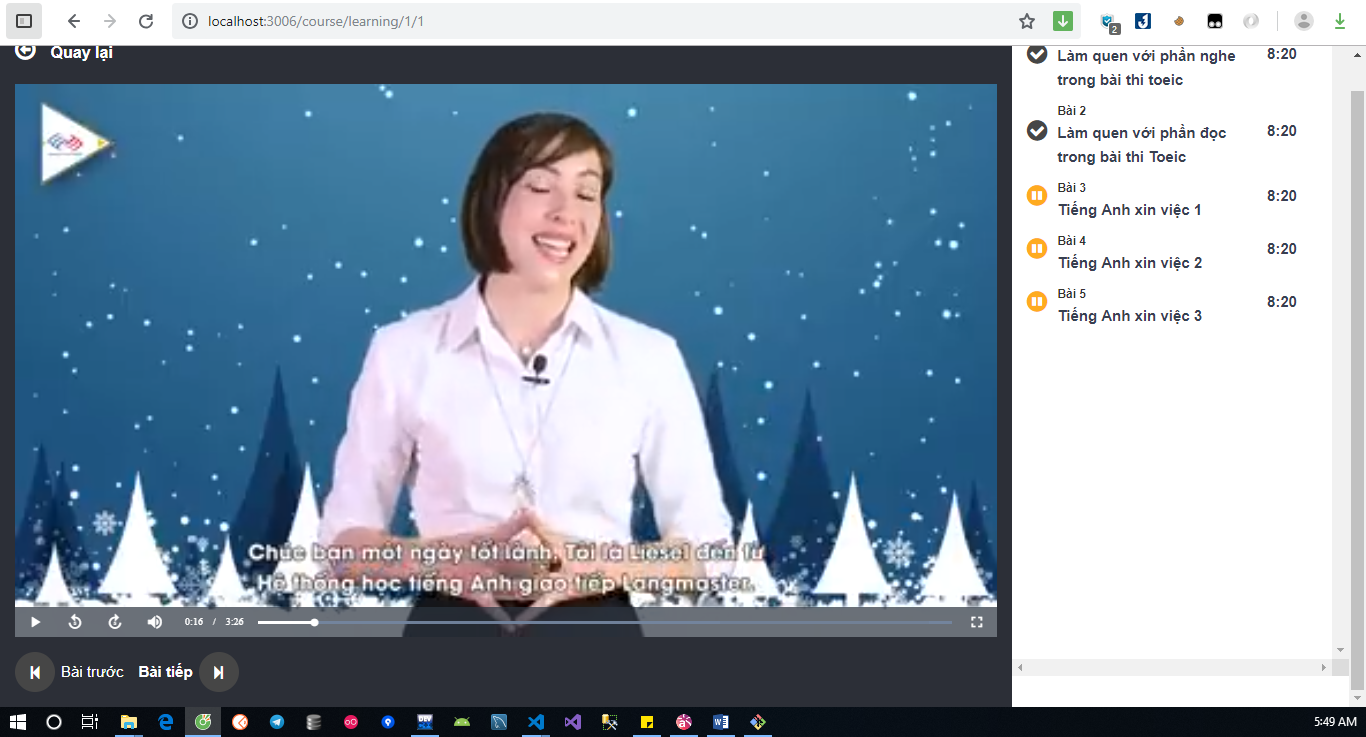
**Hình 3‑6** Màn hình danh sách chủ đề các từ vựng



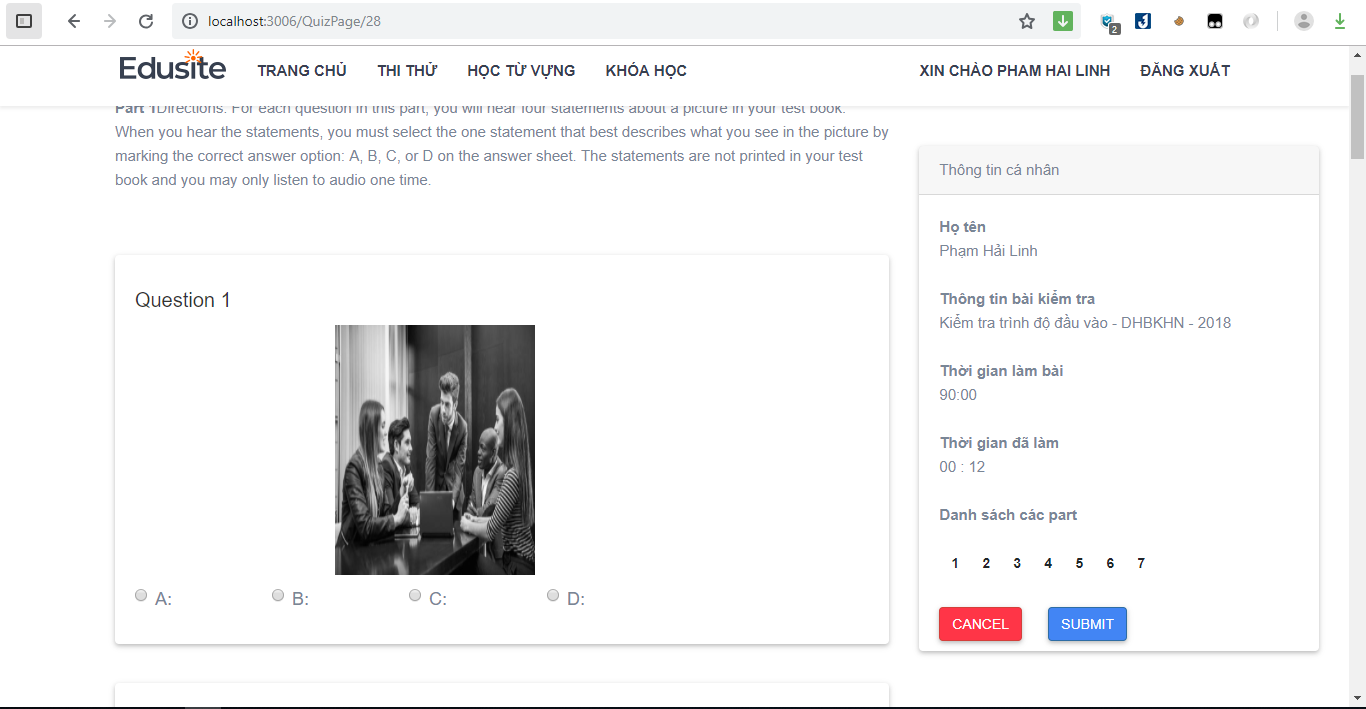
**Hình 3‑7** Màn hình học trắc nghiệm từ



**Hình 3‑8** Màn hình học nghĩa từ

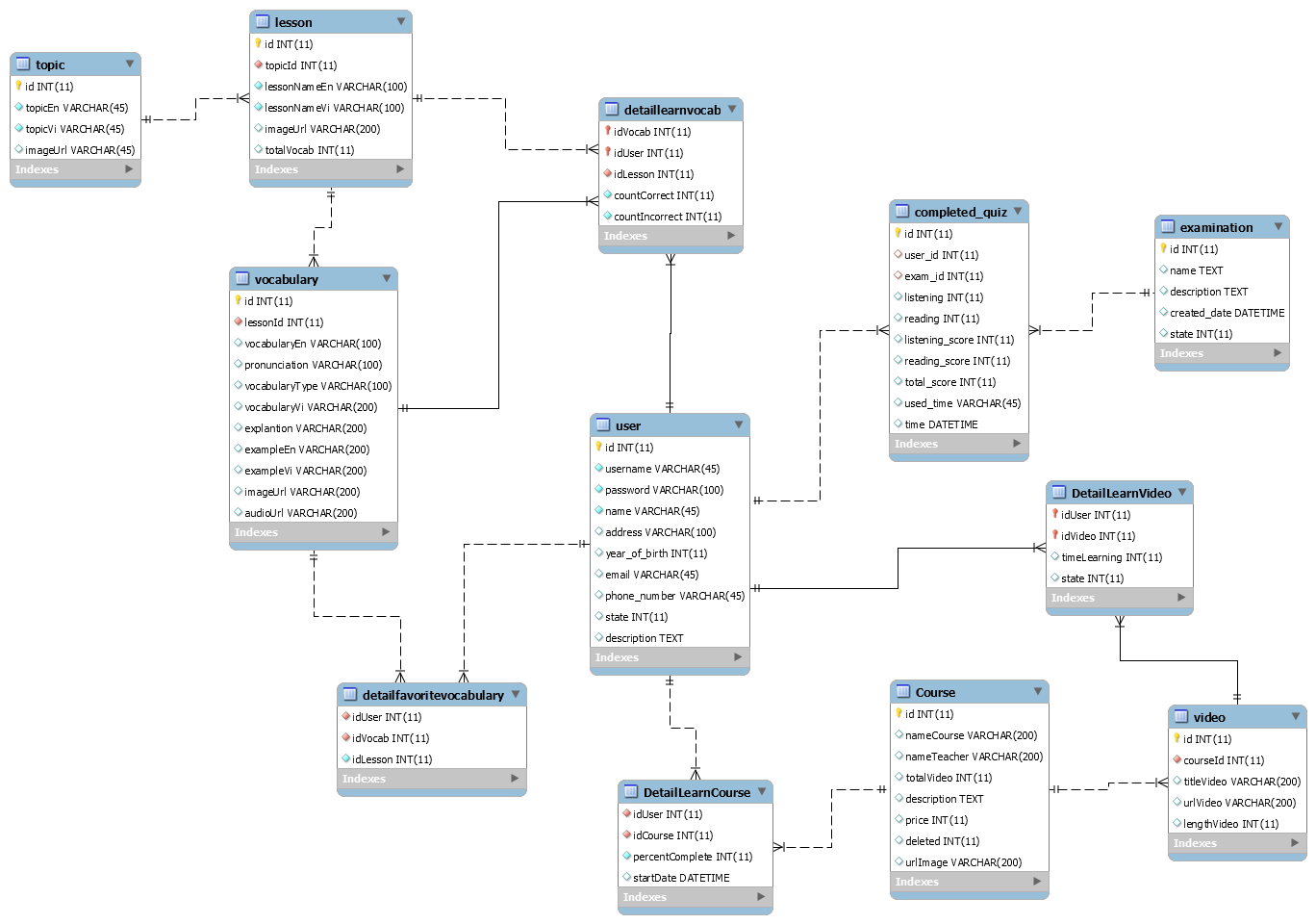


**Hình 3‑9** Màn hình học online



**Hình 3‑10** Màn hình thi thử

### Thiết kế cơ sở dữ liệu



**Hình 3‑11** Biểu đồ mô tả cơ sở dữ liệu cho website

# Kết luận và hướng phát triển

## Kết luận

### Khó khăn trong khi làm sản phẩm

Cả nhóm đều chưa bao giờ làm web nên chưa có nhiều kinh nghiệm trong thiết kế về giao diện nên nhiều chỗ còn hơi sơ sài.

Khó khăn trong việc sử dụng GIT và vẫn có nhiều lần các thành viên bị conflict.

Lần đầu tiếp xúc với những công nghệ ReactJS và NodeJS

### Đánh giá chung sản phẩm

**Ưu điểm:** Hoàn thiện đầy đủ các chức năng theo như yêu cầu. Đã tìm hiểu và config được một số công nghệ cloud của AWS như là APIGateway, config được server EC2 và deploy được app lên.

Sử dụng công nghệ phù hợp được với yêu cầu bài toán đặt ra và có khả năng mở rộng, bảo trì được dễ dàng.

**Nhược điểm:** Giao diện nhiều chỗ còn hơi đơn giản và sơ sài, chưa đồng nhất giữa các màn hình.

Chưa áp dụng được mô hình Micro service kết hợp với Docker vào trong hệ thống. Chưa biết cách dùng CI/CD trong khi làm việc.

### Hướng phát triển

Trong quá trình xây dựng hệ thống, em đã rút ra một số kinh Hướng phát triển

Như đã trình bày ở trên, sản phẩm đưa ra của đồ án này còn một số hạn chế, chính vì vậy việc phát triển hệ thống tương lai sau này trước mắt là sẽ cố gắng khắc phục những hạn chế còn tồn đọng, giải quyết cụ thể bằng việc:

**Thứ nhất**: cải thiện giao diện theo một thiết kế chuẩn để tăng trải nghiệm người dùng.

**Thứ hai:** mở rộng web không chỉ là học về toeic mà mà về cả toefl, ielts và xa hơn là tiếng anh giao tiếp.

**Thứ ba**: Tiến tới việc liên kết với các trường học, trung tâm để mở rộng về dữ liệu hiện đang có.

# Tài liệu tham khảo

1. https://nodejs.org/en/docs/
2. https://reactjs.org/docs/