

**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI  
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**



## **ĐỒ ÁN TỐT NGHIỆP**

### **TÀI LIỆU PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

**ĐỀ TÀI : Xây dựng ứng dụng học Tiếng Anh**

**Giảng viên hướng dẫn : PGS.TS Trần Đình Khang**

**Sinh viên thực hiện : Trần Đoàn Hiệp    20121720**

**Hà Nội 03/2017**

## MỤC LỤC

<b>DANH MỤC HÌNH HẠNH .....</b>	<b>4</b>
<b>DANH MỤC BẢNG BIỂU.....</b>	<b>5</b>
<b>CHƯƠNG 1. ĐỀ XUẤT – MÔ TẢ ĐỀ TÀI – CÔNG NGHỆ SỬ DỤNG .....</b>	<b>6</b>
<b>1 Hiện trạng , khó khăn , hướng giải quyết.....</b>	<b>6</b>
<b>1.1. Hiện trạng.....</b>	<b>6</b>
<b>1.2. Khó khăn khi học Tiếng Anh .....</b>	<b>6</b>
<b>1.3. Hướng giải quyết.....</b>	<b>6</b>
<b>2 . Mục đích, phạm vi của đề án.....</b>	<b>7</b>
<b>2.1 Mục đích .....</b>	<b>7</b>
<b>2.2 Phạm vi.....</b>	<b>7</b>
<b>3. Mô tả bài toán – Hướng giải quyết bài toán.....</b>	<b>7</b>
<b>3.1 Mô tả bài toán .....</b>	<b>7</b>
<b>3.2 Hướng giải quyết.....</b>	<b>8</b>
<b>4 Công nghệ sử dụng .....</b>	<b>9</b>
<b>4.1. Ngôn ngữ :.....</b>	<b>9</b>
<b>4.2. Công cụ và các thư viện khác .....</b>	<b>13</b>
<b>CHƯƠNG 2 PHÂN TÍCH THIẾT KẾ HỆ THỐNG .....</b>	<b>14</b>
<b>1 Biểu đồ use case.....</b>	<b>14</b>
<b>1.1 Use case tổng quát .....</b>	<b>14</b>
<b>1.2 Phân rã use case.....</b>	<b>15</b>
<b>1.3 Đặc tả use case .....</b>	<b>16</b>
<b>2 Biểu đồ lớp.....</b>	<b>22</b>
<b>2.1 Biểu đồ lớp tổng thể.....</b>	<b>22</b>
<b>2.2 Mô tả chi tiết.....</b>	<b>23</b>
<b>3 Biểu đồ trình tự.....</b>	<b>29</b>
<b>3.1 Chức năng đăng bài.....</b>	<b>29</b>
<b>3.2 Biểu đồ trình tự cho chức năng bình luận.....</b>	<b>29</b>

3.3 Biểu đồ trình tự cho chức năng luyện nghe .....	30
3.4 Biểu đồ trình tự cho chức năng luyện nói .....	31
3.5 Biểu đồ trình tự cho chức năng luyện viết .....	32
3.6 Biểu đồ trình tự cho chức năng Làm đề toeic .....	33
4 Biểu đồ trạng thái .....	34
4.1 Biểu đồ trạng thái của lớp Course .....	34
4.2 Biểu đồ trạng thái của lớp ToeicTest .....	35
5 Thiết kế cơ sở dữ liệu.....	36
5.1 Mô hình.....	36
5.2 Mô tả chi tiết.....	36

## DANH MỤC HÌNH ẢNH

Hình 1 : Speech to Text .....	9
Hình 2 : Nodejs .....	10
Hình 3 : Mysql .....	12
Hình 4: Angular 2 và Ionic 2 .....	13
Hình 5: Use case tổng quát .....	14
Hình 6: Phân rã use case diễn đàn trao đổi .....	15
Hình 7: Phân rã use case học từ .....	15
Hình 8: Phân rã use luyện tập .....	15
Hình 9: Phân ra use case Quản lí Khóa học.....	16
Hình 10: Biểu đồ lớp.....	22
Hình 11: Biểu đồ trình tự Chức năng đăng bài.....	29
Hình 12: Biểu đồ trình tự Chức năng bình luận.....	30
Hình 13: Biểu đồ trình tự Chức năng luyện nghe.....	31
Hình 14: Biểu đồ trình tự Chức năng luyện nói.....	32
Hình 15: Biểu đồ trình tự Chức năng luyện viết.....	33
Hình 16 : Biểu đồ trình tự Chức năng làm đề Toeic.....	34
Hình 17: Biểu đồ trạng thái của của lớp Course.....	34
Hình 18: Biểu đồ trạng thái của của lớp ToeicTest .....	35
Hình 19: Cơ sở dữ liệu của client .....	36
Hình 20: Cơ sở dữ liệu của server .....	36

## DANH MỤC BẢNG BIỂU

Bảng 1 : Đặc tả use case Học từ .....	16
Bảng 2: Đặc tả use case Luyện nghe .....	17
Bảng 3: Đặc tả use case Luyện nói .....	18
Bảng 4: Đặc tả use case Luyện viết .....	18
Bảng 5: Đặc tả use case Làm đề Toeic .....	19
Bảng 6: Đặc tả use case Xem kết quả Toeic.....	19
Bảng 7: Đặc tả use case Đăng câu hỏi lên diễn đàn .....	20
Bảng 8: Đặc tả use case Đăng câu hỏi lên diễn đàn .....	20
Bảng 9: Đặc tả use case Đăng nhập .....	21
Bảng 10: Đặc tả use case Thêm khóa học .....	21
Bảng 11: Mô tả lớp DownloadService.....	23
Bảng 12: Mô tả lớp databaseService.....	23
Bảng 13 Mô tả lớp ToecicService.....	24
Bảng 14: Mô tả lớp CourseService .....	24
Bảng 15: Mô tả lớp ForumService.....	25
Bảng 16: Mô tả lớp Course .....	25
Bảng 17: Mô tả lớp Lesson .....	26
Bảng 18: Mô tả lớp Word .....	26
Bảng 19: Mô tả lớp Post .....	27
Bảng 20: Mô tả lớp Comment.....	27
Bảng 21: Mô tả lớp User.....	28
Bảng 22: Mô tả lớp ToeicTest .....	29
Bảng 23: Mô tả bảng courses.....	37
Bảng 24: Mô tả bảng lessons .....	37
Bảng 25: Mô tả bảng words .....	38
Bảng 26: Mô tả bảng tests.....	38
Bảng 27: Mô tả bảng parts .....	38
Bảng 28: Mô tả bảng users.....	39
Bảng 29: Mô tả bảng posts.....	39
Bảng 30: Mô tả bảng comment.....	40

# **CHƯƠNG I. ĐỀ XUẤT – MÔ TẢ ĐỀ TÀI – CÔNG NGHỆ SỬ DỤNG**

## **1 Hiện trạng , khó khăn , hướng giải quyết**

### **1.1. Hiện trạng**

Tiếng Anh là một trong những ngôn ngữ phổ biến nhất trên thế giới và là ngôn ngữ bản địa của nhiều quốc gia. Nó đã trở thành ngôn ngữ thứ hai quan trọng nhất và ngày càng được nhiều người sử dụng. Các phương tiện truyền thông như Internet, tivi, báo chí, rồi hầu hết các website, sách, bản tin ... đều sử dụng tiếng Anh. Rất nhiều trường đại học lấy chứng chỉ Tiếng Anh làm điều kiện để sinh viên ra trường. Việc giao tiếp bằng Tiếng Anh cũng ngày càng phổ biến. Nói vậy để thấy, tiếng Anh rất quan trọng đối với mỗi người, là chìa khóa để mở cánh cửa tri thức, vào thế giới khoa học.

### **1.2. Khó khăn khi học Tiếng Anh**

Chúng ta đều biết Tiếng Anh rất quan trọng. Tuy nhiên, đối với nhiều người thì việc học Tiếng Anh không hề dễ dàng. Rất nhiều khó khăn trong quá trình học như: Không thể nghe được khi giao tiếp, vấn đề phát âm cũng không hề đơn giản, không thể nhớ được từ vựng lâu ... Để giải quyết vấn đề này, chúng ta cần luyện tập các kỹ năng một cách thường xuyên và đều đặn hơn.

### **1.3. Hướng giải quyết**

Ngày nay, công nghệ thông tin ngày càng phát triển mạnh mẽ và trở thành một phần quan trọng trong cuộc sống mỗi chúng ta. Sự kết hợp giữa việc học Tiếng Anh và công nghệ thông tin cũng ngày càng được nhiều người áp dụng. Phương pháp này đáp ứng cho nhu cầu học tập, tích lũy kiến thức cho mọi người một cách dễ dàng hơn. Ngoài ra, nó còn đem lại lợi ích to lớn, tiết kiệm thời gian, công sức, tiền bạc cho mỗi chúng ta.

Để giải quyết những khó khăn mà nhiều người gặp phải khi học Tiếng Anh. Em đã bắt tay xây dựng “**Ứng dụng học tiếng Anh**” trên mobile để giúp người dùng học tập và rèn luyện các kỹ năng một cách tiện lợi và thường xuyên hơn.

## 2 . Mục đích, phạm vi của đề án

### 2.1 Mục đích

Tìm hiểu về công nghệ lập trình ứng dụng hybrid cho mobile sử dụng angular/ionic 2 và tìm hiểu về nodejs.

Xây dựng lên ứng dụng có thể giúp người dùng luyện tập Tiếng Anh thông qua các kỹ năng nghe,nói,viết với các bài trắc nghiệm đơn giản.Đồng thời đánh giá năng lực,trình độ thông qua làm đề thi Toeic.

### 2.2 Phạm vi

Do thời gian không nhiều,kiến thức còn hạn hẹp nên đề án này chỉ phát triển trên nền tảng di động.

## 3. Mô tả bài toán – Hướng giải quyết bài toán

### 3.1 Mô tả bài toán

**Đối với người sử dụng :** Để thực hiện mục tiêu giúp người dùng học Tiếng Anh một cách đơn giản và hiệu quả nhất, hệ thống ứng dụng sẽ có những chức năng cơ bản sau :

- *Chức năng học từ :* Giúp người dùng tăng cường vốn từ vựng thông qua việc học.Một từ sẽ gồm có từ tiếng anh, từ tiếng việt, phiên âm tiếng anh ,hình ảnh minh họa, âm thanh. Theo nghiên cứu, phương pháp học từ vựng Tiếng Anh bằng hình ảnh vô cùng hiệu quả trong việc ghi nhớ từ, hình ảnh sẽ giúp chúng ta hồi tưởng lại sự việc dễ dàng hơn và điều này được ứng dụng cho việc học từ vựng một cách đơn giản hơn thay vì chỉ luyện tập viết chữ như bình thường bạn đã được học.
- *Luyện tập các kỹ năng như nghe – nói – viết :*
  - + *Luyện nghe :* Giúp người dùng rèn luyện kỹ năng nghe với hình thức câu hỏi trắc nghiệm : Nghe từ và chọn nghĩa đúng.
  - + *Luyện nói :* Giúp người dùng rèn luyện kỹ năng nói thông qua việc nói qua micro của thiết bị.Sau đó ứng dụng sẽ đánh giá mức độ phát âm đúng của người nói.
  - + *Luyện viết :* Giúp người dùng rèn luyện kỹ năng viết thông qua 2 cách : Luyện viết có gợi ý (việc sắp xếp lại các ký tự cho đúng với từ tiếng anh cho trước) và luyện viết không có gợi ý (viết lại từ hoàn chỉnh với không có ký tự gợi ý).

- *Làm đề thi Toeic* : Giúp cho việc đánh giá trình độ Tiếng Anh của người dùng thông qua làm bài thi Toeic hoàn chỉnh.
- *Diễn đàn trao đổi* : Ứng dụng có một diễn đàn để giúp mọi người hỏi & giải đáp những thắc mắc liên quan đến Tiếng Anh.

**Đối với admin** : Công việc chính của admin chính là thêm những khóa học các chủ đề mới để giúp người dùng học tốt hơn. Để có thể quản lí được các khóa học , admin cần đăng nhập vào trang quản trị để được quyền thêm , xóa các khóa học.

### 3.2 Hướng giải quyết

- *Chức năng học từ* : Để giải quyết vấn đề này, ta cần chuẩn bị các bộ từ điển theo các chủ đề khác nhau ( ví dụ như 600 từ toeic, 1000 từ tiếng anh cơ bản, .v.v. ), các từ này được lưu vào database local, khi chọn từ thì sẽ query từ đó theo id trong database.
- *Luyện tập các kỹ năng như nghe – nói – viết* :
  - + *Luyện nghe* : Với hình thức câu hỏi trắc nghiệm , người dùng nghe sau đó chọn đáp án đúng.Yêu cầu đặt ra là cần một tập các câu hỏi theo hình thức trắc nghiệm 4 đáp án.Các câu hỏi sẽ được sinh ngẫu nhiên bằng cách với mỗi một từ sẽ tương ứng có 1 câu hỏi trắc nghiệm về từ đó.Một câu hỏi bao gồm 1 từ làm đáp án đúng và 3 từ nhiễu được sinh ngẫu nhiên.Ví dụ một câu hỏi sẽ có cấu trúc như sau :

```
{
  correctAnswer : word1,
  answerList : [ word1, word2, word3, word4]
}
```

+ *Luyện nói* : Với chức năng này, yêu cầu người dùng cần phát âm đúng với từ cho trước thông qua micro của thiết bị,hệ thống sẽ đánh giá xem người đó phát âm đúng hay sai ở mức độ bao nhiêu .Ở đây cần sử dụng Speech to Text plugin của cordova để chuyển âm thanh người nói sang dạng text.Rồi so sánh với từ cho sẵn.Việc so sánh cần sử dụng thuật toán tính độ tương đồng giữa 2 chuỗi để đánh giá độ tương đồng giữa 2 chuỗi là bao nhiêu.Từ đó mới đánh giá được người đó nói đúng ở mức độ bao nhiêu %.





Hình 1 : Speech to Text

+ *Luyện viết* : Có 2 cách để người dùng luyện viết: Luyện viết có gợi ý và không gợi ý. Với cách thứ nhất : Người dùng cần sắp xếp lại các kí tự đã bị xáo trộn sao cho đúng với nghĩa của từ đã cho trước. Các câu hỏi sẽ được sinh ngẫu nhiên bằng cách với mỗi một từ sẽ tương ứng có 1 câu hỏi của từ đó. Mỗi câu hỏi sẽ bao gồm 1 đáp án đúng và 1 list các kí tự bị sắp xếp lộn xộn. Ví dụ :

```
{
  correctAnswer : 'hello',
  answerList : [ 'o', 'l', 'e', 'h', 'l']
}
```

Còn với cách thứ hai: Người dùng cần nhập vào ô input text sao cho phù hợp với nghĩa tiếng việt của từ đã cho trước.

- *Làm đề thi Toeic để đánh giá trình độ Tiếng Anh của người dùng* : Đề thi có 7 Part, gồm 2 phần chính là Listening và Reading tương ứng với thời gian cho mỗi phần là 45 phút và 75 phút. Sau khi hoàn thành, hệ thống trả về kết quả của bài thi đó. Các đề thi được phân biệt nhau theo id có dạng “1001-2001-3002-4004-5004-6003-7001”, tức là : đề thi có id như trên sẽ lấy dữ liệu từ part1 có idPart = 1001, part 2 có idPart – 2001. Nếu trong ứng dụng, dữ liệu từ các part chưa có thì cần tải trên server về để có thể sử dụng để làm đề thi. Sau khi hoàn tất thì lưu lại kết quả.
- *Diễn đàn trao đổi* : Để có thể làm được chức năng này, cần có database lưu lại các bài đăng, comment của người dùng.

Tất cả các dữ liệu về từ như hình ảnh, âm thanh, chi tiết của từ và dữ liệu về các part của đề thi toeic được đặt tại server.

## 4 Công nghệ sử dụng

### 4.1. Ngôn ngữ :

#### 4.1.1 Backend : NodeJS + mysql

**NodeJS** là một mã nguồn được xây dựng dựa trên nền tảng Javascript V8 Engine. NodeJS là một mã nguồn mở được sử dụng rộng bởi hàng ngàn lập trình viên trên toàn thế giới. NodeJS có thể chạy trên nhiều nền tảng hệ điều hành khác nhau từ Window cho tới Linux, OS X nên đó cũng là một lợi thế. NodeJS cung cấp các thư viện phong phú ở dạng Javascript Module khác nhau giúp đơn giản hóa việc lập trình và giảm thời gian ở mức thấp nhất.



Hình 2 : Nodejs

Các tính năng của NodeJS :

*Không đồng bộ:* Tất cả các API của NodeJS đều không đồng bộ, nó chủ yếu dựa trên nền của NodeJS Server và chờ đợi Server trả dữ liệu về. Việc di chuyển máy chủ đến các API tiếp theo sau khi gọi và cơ chế thông báo các sự kiện của Node.js giúp máy chủ để có được một phản ứng từ các cuộc gọi API trước (Realtime).

*Chạy rất nhanh:* NodeJS được xây dựng dựa vào nền tảng V8 Javascript Engine nên việc thực thi chương trình rất nhanh

*Đơn luồng nhưng khả năng mở rộng cao:* NodeJS sử dụng một mô hình luồng duy nhất với sự kiện lặp. cơ chế tổ chức sự kiện giúp các máy chủ để đáp ứng một cách không ngăn chặn và làm cho máy chủ cao khả năng mở

rộng như trái ngược với các máy chủ truyền thống mà tạo đề hạn chế để xử lý yêu cầu. Node.js sử dụng một chương trình đơn luồng và các chương trình tương tự có thể cung cấp dịch vụ cho một số lượng lớn hơn nhiều so với yêu cầu máy chủ truyền thống như Apache HTTP Server.

**MySQL** là hệ quản trị cơ sở dữ liệu tự do nguồn mở phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng. Vì MySQL là cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Với tốc độ và tính bảo mật cao, MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên internet. MySQL miễn phí hoàn toàn cho nên bạn có thể tải về MySQL từ trang chủ. Nó có nhiều phiên bản cho các hệ điều hành khác nhau: phiên bản Win32 cho các hệ điều hành dòng Windows, Linux, Mac OS X, Unix ...

**MySQL** là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) hay còn được gọi là Relational Database Management system. Cơ sở dữ liệu quan hệ là cơ sở dữ liệu mà dữ liệu bên trong nó được tổ chức thành các bảng. Các bảng được tổ chức bằng cách nhóm dữ liệu theo cùng chủ đề và có chứa các cột và các hàng thông tin. Sau đó các bảng này được liên kết với nhau bởi bộ Database Engine khi có yêu cầu. cơ sở dữ liệu quan hệ là một trong những mô hình cơ sở dữ liệu thông dụng nhất hiện nay.

**MySQL** là một trong những ví dụ rất cơ bản về Hệ Quản trị Cơ sở dữ liệu quan hệ sử dụng Ngôn ngữ truy vấn có cấu trúc (SQL).

**MySQL** được sử dụng cho việc hỗ trợ PHP, Perl, và nhiều ngôn ngữ khác, nó làm nơi lưu trữ những thông tin trên các trang web viết bằng PHP hay Perl,...



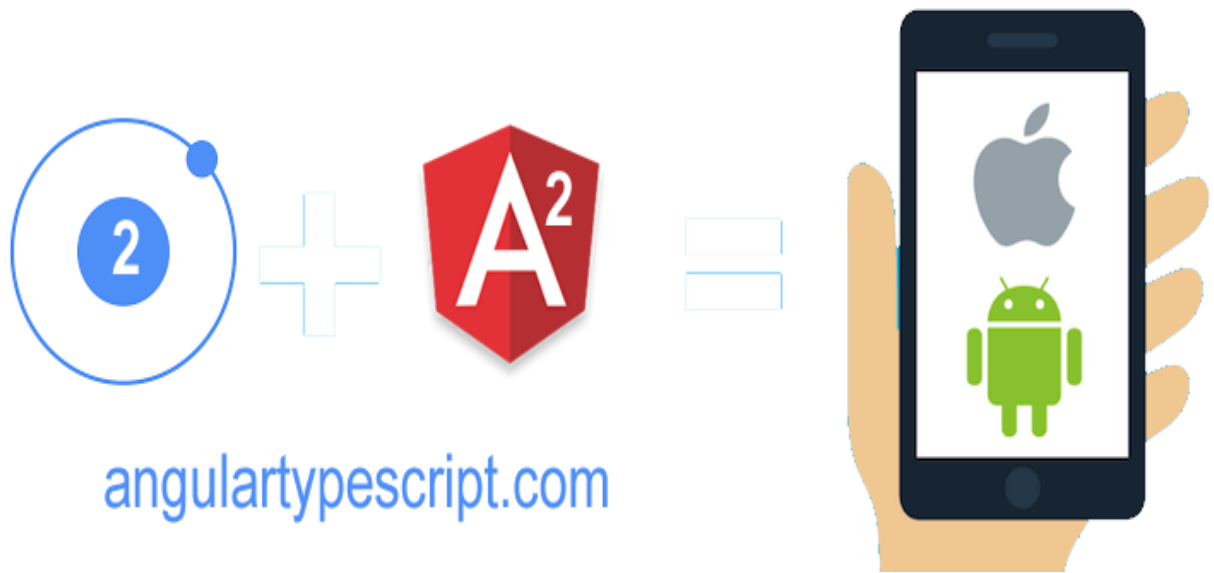
Hình 3 : Mysql

#### 4.1.2 Frontend : angular/ionic 2.

**TypeScript** là một ngôn ngữ mã nguồn mở miễn phí hiện đang được phát triển và bảo trì bởi Microsoft. Nó là tập cha của JavaScript, với các bổ sung các tùy chọn kiểu tĩnh và lớp trên cơ sở lập trình hướng đối tượng cho ngôn ngữ này. Anders Hejlsberg, kiến trúc sư ngôn ngữ C# và là người tạo ra ngôn ngữ Delphi và Turbo Pascal đã tham gia phát triển TypeScript. TypeScript có thể sử dụng để phát triển các ứng dụng chạy ở client-side (*Angular2*) và server-side (*NodeJS*).

**Angular 2** là 1 framework phát triển trên nền JavaScript của Google, kế thừa các đặc điểm của AngularJS và phát triển một phương thức tiếp cận việc xây dựng ứng dụng hoàn toàn mới, phương pháp hướng Component.

**Ionic** là một hybrid Framework được sử dụng để phát triển các ứng dụng di động dựa trên nền tảng công nghệ web HTML ( sự kết hợp giữa Angular và Cordova), được tạo bởi Max Lynch, Ben Sperry, và Adam Bradley vào 2013. Đây là một framework rất mạnh để viết các ứng dụng hybrid. Nó khắc phục các nhược điểm của Native app và Mobile webapp và kết hợp được sức mạnh của 2 nền tảng đó.



Hình 4: Angular 2 và Ionic 2

Ưu điểm :

- Dễ học, thời gian phát triển nhanh, có thể sử dụng các kỹ năng từ lập trình web.
- Đa nền tảng
- Khả năng truy cập đến các tính năng của thiết bị và hệ điều hành như bluetooth, camera,...
- Dễ dàng thiết kế giao diện cho các thiết bị có kích cỡ khác nhau.

Nhược điểm :

- Hiệu năng vẫn chưa cao và ổn định.

#### 4.2. Công cụ và các thư viện khác

- Sử dụng Sublime Text 3 để code.
- Sử dụng cordova để build ứng dụng ionic để chạy trên nền tảng Android.

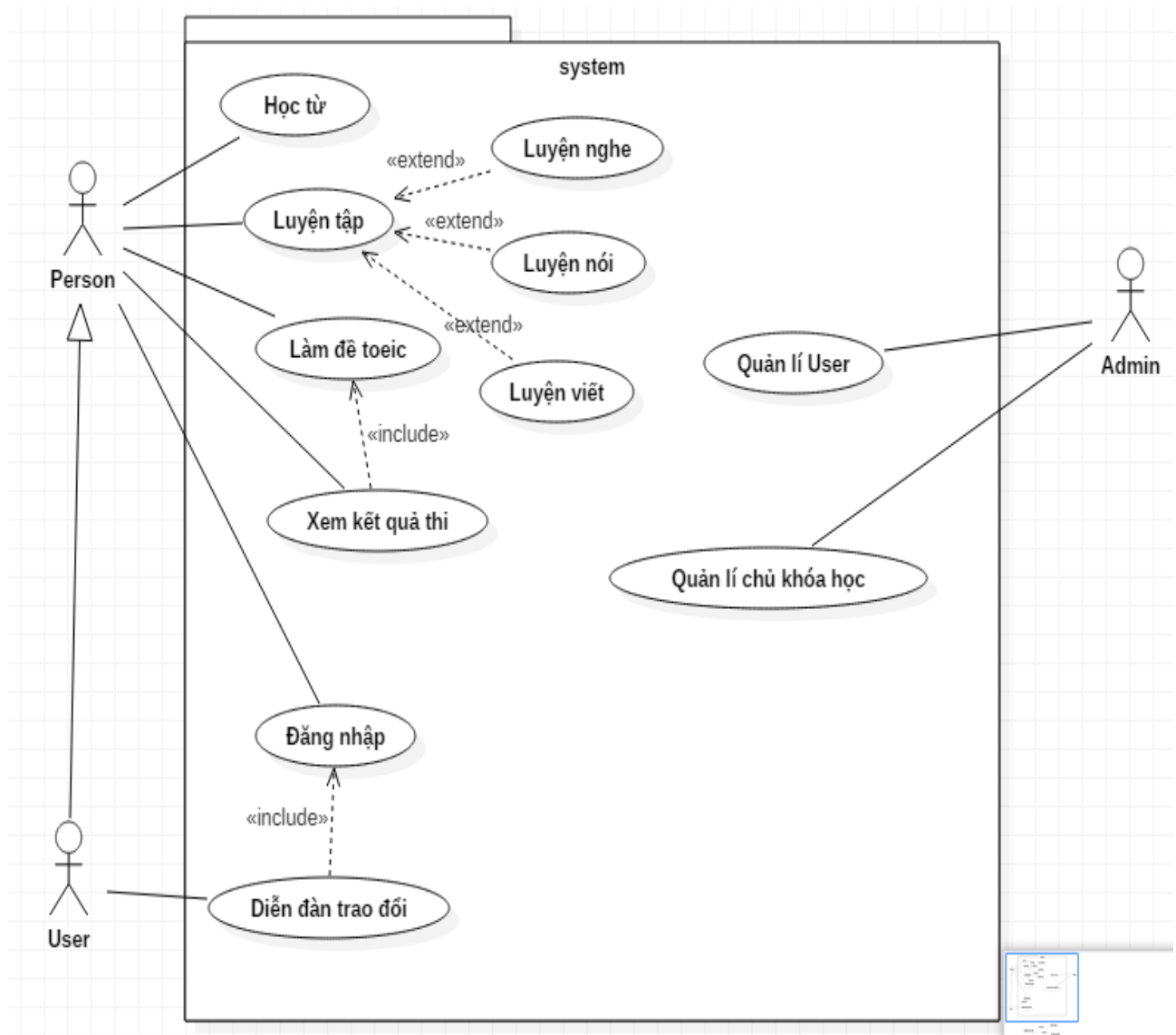
**Apache cordova** là một framework phát triển các ứng dụng cho thiết bị di động. Nó giúp cho chúng ta có thể phát triển các ứng dụng mobile bằng HTML, CSS3 và Javascripts. Hay có thể nói chúng ta có thể phát triển được ứng dụng mobile không quá phức tạp cho nhiều loại thiết bị với một bộ source code.

- Sử dụng thư viện Html DOM Parser bằng ngôn ngữ PHP để crawl dữ liệu các bài học , đề thi Toeic ... Các dữ liệu về bài học như nội dung của từ, hình ảnh, âm thanh và các bài thi Toeic được crawl từ nhiều nguồn khác nhau và được lưu trên server. Do không có server thật nên để test em đã sử dụng server localhost với nodejs.

## CHƯƠNG 2 PHÂN TÍCH THIẾT KẾ HỆ THỐNG

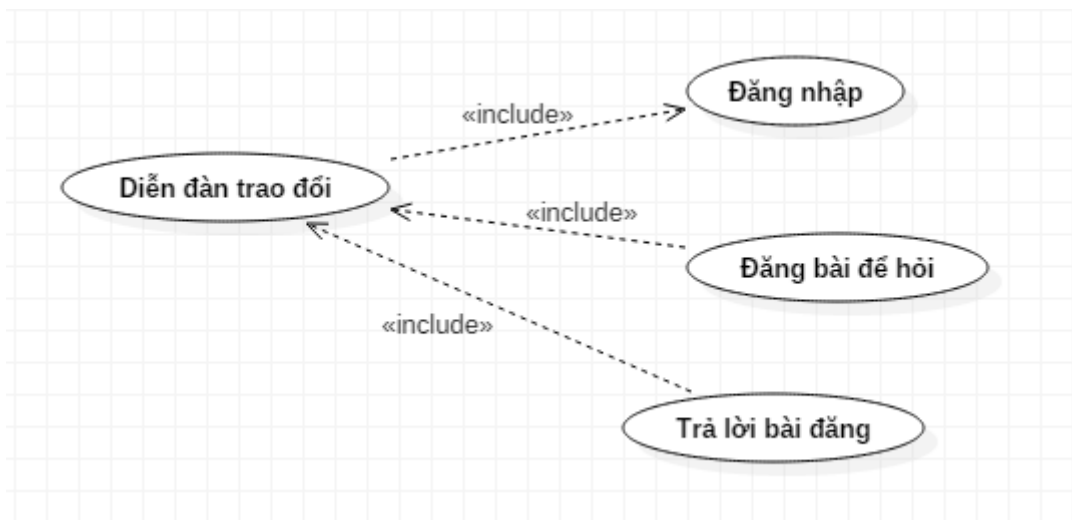
### 1 Biểu đồ use case

#### 1.1 Use case tổng quát

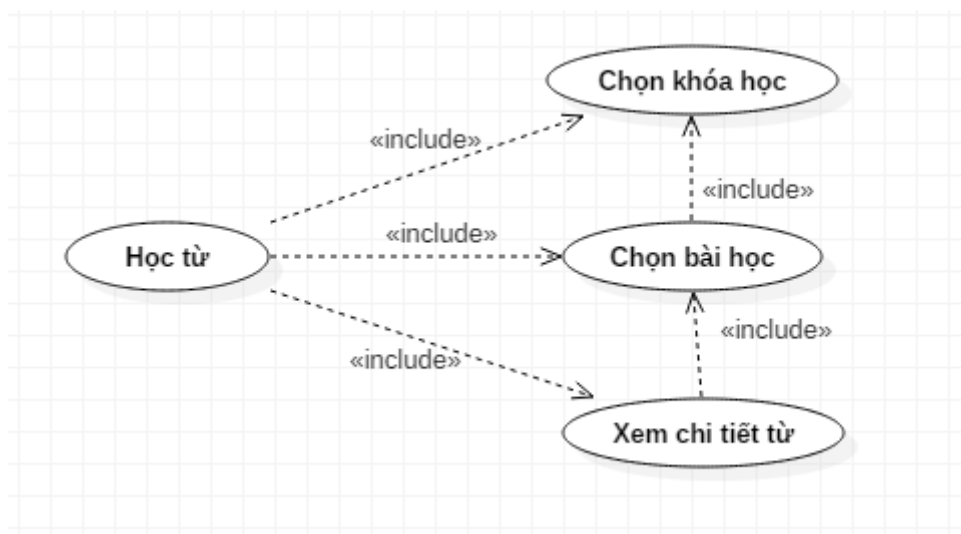


Hình 5: Use case tổng quát

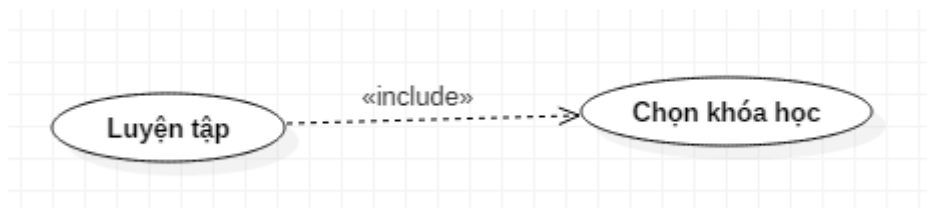
## 1.2 Phân rã use case



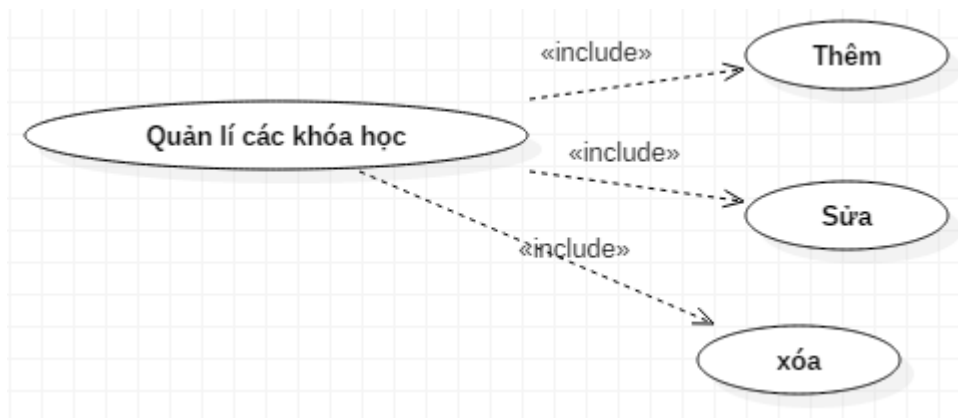
Hình 6: Phân rã use case diễn đàn trao đổi



Hình 7: Phân rã use case học từ



Hình 8: Phân rã use luyện tập



Hình 9: Phân ra use case Quản lí Khóa học

### 1.3 Đặc tả use case

#### 1.3.1 Use case Học từ

Tên use case	Học từ
Tác nhân	Person
Mô tả	Người sử dụng học những từ có trong các khóa học
Điều kiện	Không có
Luồng chính	<ol style="list-style-type: none"> <li>1. Người sử dụng chọn 1 khóa học bất kì.</li> <li>2. Ứng dụng kiểm tra <ul style="list-style-type: none"> <li>- Nếu khóa học chưa được download, thông báo yêu cầu người dùng download khóa học</li> <li>- Ngược lại, thì tiếp tục</li> </ul> </li> <li>3. Ứng dụng hiển thị thông tin khóa học và danh sách bài học của khóa học đó.</li> <li>4. Người sử dụng chọn 1 bài học bất kì.</li> <li>5. Ứng dụng hiển thị danh sách các từ của bài học đó.</li> </ol>

Bảng 1 : Đặc tả use case Học từ



### 1.3.2 Use case Luyện nghe

Tên use case	Luyện nghe
Tác nhân	Person
Mô tả	Người sử dụng luyện tập kỹ năng nghe thông qua câu hỏi trắc nghiệm 4 đáp án
Điều kiện	Không có
Luồng chính	<ol style="list-style-type: none"><li>1. Người sử dụng chọn 1 khóa học bất kì.</li><li>2. Ứng dụng kiểm tra<ul style="list-style-type: none"><li>- Nếu khóa học chưa được download, thông báo yêu cầu người dùng download khóa học</li><li>- Ngược lại, thì tiếp tục</li></ul></li><li>3. Ứng dụng hiển thị thông tin khóa học và danh sách bài học của khóa học đó.</li><li>4. Người dùng chọn luyện tập.</li><li>5. Ứng dụng hiển thị view các kỹ năng luyện tập</li><li>6. Chọn Luyện nghe</li></ol>

Bảng 2: Đặc tả use case Luyện nghe

### 1.3.3 Use case Luyện nói

Tên use case	Luyện nói
Tác nhân	Person
Mô tả	Người sử dụng luyện tập kỹ năng phát âm bằng cách phát âm đúng với từ cho trước.
Điều kiện	Không có

Luồng chính	<ol style="list-style-type: none"> <li>1. Người sử dụng chọn 1 khóa học bất kì.</li> <li>2. Ứng dụng kiểm tra <ul style="list-style-type: none"> <li>- Nếu khóa học chưa được download, thông báo yêu cầu người dùng download khóa học</li> <li>- Ngược lại, thì tiếp tục</li> </ul> </li> <li>3. Ứng dụng hiển thị thông tin khóa học và danh sách bài học của khóa học đó.</li> <li>4. Người dùng chọn luyện tập.</li> <li>5. Ứng dụng hiển thị các kỹ năng luyện tập</li> <li>6. Chọn Luyện nói</li> </ol>
-------------	---

Bảng 3: Đặc tả use case Luyện nói

#### 1.3.4 Use case Luyện viết

Tên use case	Luyện viết
Tác nhân	Person
Mô tả	Người sử dụng luyện tập kỹ năng phát âm thông qua câu hỏi sắp xếp lại thành từ chính xác
Điều kiện	Không có
Luồng chính	<ol style="list-style-type: none"> <li>1. Người sử dụng chọn 1 khóa học bất kì.</li> <li>2. Ứng dụng kiểm tra <ul style="list-style-type: none"> <li>- Nếu khóa học chưa được download, thông báo yêu cầu người dùng download khóa học</li> <li>- Ngược lại, thì tiếp tục</li> </ul> </li> <li>3. Ứng dụng hiển thị thông tin khóa học và danh sách bài học của khóa học đó.</li> <li>4. Người dùng chọn luyện tập.</li> <li>5. Ứng dụng hiển thị các kỹ năng luyện tập</li> <li>6. Chọn Luyện viết</li> </ol>

Bảng 4: Đặc tả use case Luyện viết

### 1.3.5 Use case Làm đề Toeic

Tên use case	Làm đề Toeic
Tác nhân	Person
Mô tả	Người sử dụng làm đề thi Toeic hoàn chỉnh
Điều kiện	Không có
Luồng chính	<ol style="list-style-type: none"><li>1. Người sử dụng chọn Toeic Test trên menu</li><li>2. Ứng dụng hiển thông tin danh sách các đề thi Toeic.</li><li>3. Người dùng chọn một đề thi bất kì.</li><li>4. Ứng dụng kiểm tra đề thi<ul style="list-style-type: none"><li>- Nếu đề thi đã được làm thì hiện thông báo đề thi đã được hoàn thành</li><li>- Ngược lại ,thì tiếp tục</li></ul></li></ol>

Bảng 5: Đặc tả use case Làm đề Toeic

### 1.3.6 Use case Xem kết quả

Tên use case	Xem kết quả
Tác nhân	Person
Mô tả	Người sử dụng xem kết quả thi Toeic của mình khi đề thi đã được hoàn tất
Điều kiện	Không có
Luồng chính	<ol style="list-style-type: none"><li>1. Người sử dụng chọn Toeic Test trên menu</li><li>2. Ứng dụng hiển thông tin danh sách các đề thi Toeic.</li><li>3. Người dùng chọn một đề thi bất kì.</li><li>4. Người dùng chọn xem kết quả</li><li>5. Ứng dụng kiểm tra<ul style="list-style-type: none"><li>- Nếu đề thi đã được làm thì hiển thi kết quả thi</li><li>- Ngược lại ,thông báo đề thi chưa hoàn tất</li></ul></li></ol>

Bảng 6: Đặc tả use case Xem kết quả Toeic

### 1.3.7 Use case Đăng câu hỏi lên diễn đàn

Tên use case	Đăng câu hỏi lên diễn đàn
Tác nhân	User
Mô tả	Người dùng đăng các câu hỏi cần thắc mắc lên diễn đàn
Điều kiện	Person cần đăng nhập thành công
Luồng chính	<ol style="list-style-type: none"><li>1. Người dùng chọn Hỏi &amp; đáp trên menu</li><li>2. Ứng dụng hiển thị form post câu hỏi và danh sách các câu hỏi được người dùng đăng lên</li><li>3. Nhập thông tin câu hỏi và nhấn <b>ĐĂNG BÀI</b></li><li>4. Ứng dụng kiểm tra<ul style="list-style-type: none"><li>- Nếu không nhập gì , ứng dụng yêu cầu nhập đủ thông tin trong text input</li><li>- Ngược lại , đăng bài thành công</li></ul></li></ol>

Bảng 7: Đặc tả use case Đăng câu hỏi lên diễn đàn

### 1.3.8 Use case Comment trả lời câu hỏi

Tên use case	Comment trả lời câu hỏi
Tác nhân	User
Mô tả	Người dùng comment trả lời các câu hỏi trên diễn đàn
Điều kiện	Person cần đăng nhập thành công
Luồng chính	<ol style="list-style-type: none"><li>1. Người dùng chọn Hỏi &amp; đáp trên menu</li><li>2. Ứng dụng hiển thị form post câu hỏi và danh sách các câu hỏi được người dùng đăng lên</li><li>3. Người dùng chọn 1 câu hỏi trong danh sách câu hỏi</li><li>4. Người dùng nhập nội dung cần trả lời và nhấn <b>OK</b></li><li>5. Ứng dụng kiểm tra<ul style="list-style-type: none"><li>- Nếu không nhập gì , ứng dụng yêu cầu nhập đủ thông tin trong text input</li><li>- Ngược lại , comment thành công</li></ul></li></ol>

Bảng 8: Đặc tả use case Đăng câu hỏi lên diễn đàn

### 1.3.9 Use case Đăng nhập

Tên use case	Đăng nhập
Tác nhân	Person
Mô tả	Người dùng đăng nhập
Điều kiện	Không có
Luồng chính	<ol style="list-style-type: none"><li>1. Người sử dụng chọn Đăng nhập trên menu</li><li>2. Ứng dụng kiểm tra xem người dùng đã đăng nhập hay chưa.<ul style="list-style-type: none"><li>- Nếu đã đăng nhập, bỏ qua</li><li>- Ngược lại , hiển thị view đăng nhập</li></ul></li><li>3. Người dùng chọn đăng nhập bằng facebook hoặc đăng nhập qua tài khoản đã đăng kí</li><li>4. Ứng dụng kiểm tra<ul style="list-style-type: none"><li>- Nếu nhập sai tài khoản (hoặc chưa đăng nhập facebook trên điện thoại), ứng dụng thông báo lỗi</li><li>- Ngược lại , ứng dụng thông báo đăng nhập thành công</li></ul></li></ol>

Bảng 9: Đặc tả use case Đăng nhập

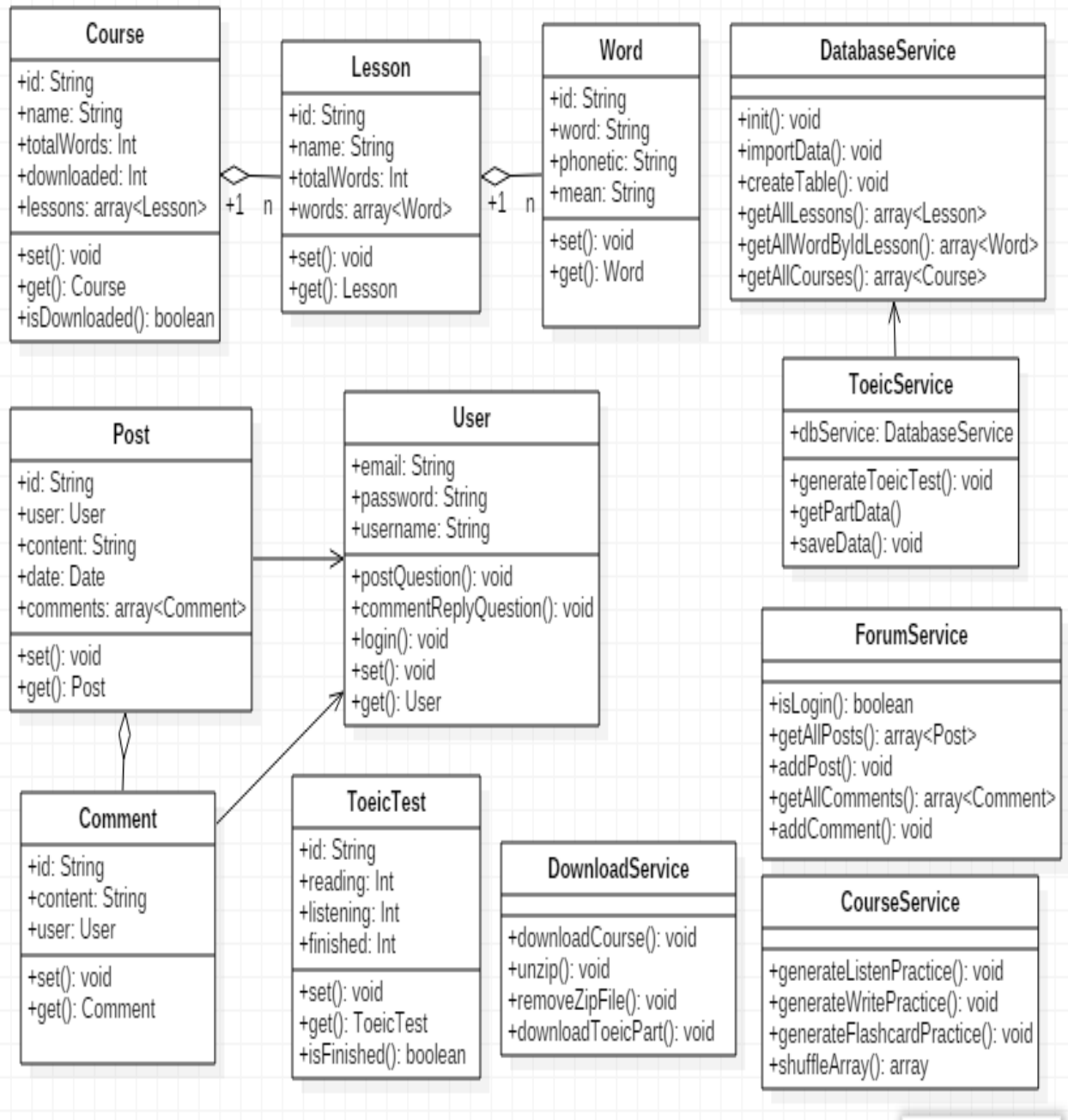
### 1.3.10 Use case Thêm khóa học

Tên use case	Thêm khóa học
Tác nhân	Admin
Mô tả	Admin vào trang quản trị để thêm khóa học
Điều kiện	Admin phải đăng nhập vào trang quản trị
Luồng chính	<ol style="list-style-type: none"><li>1. Admin đăng nhập trang quản trị</li><li>2. Ứng dụng kiểm tra xem người dùng đã đăng nhập hay chưa.<ul style="list-style-type: none"><li>- Nếu đã đăng nhập, bỏ qua</li><li>- Ngược lại , hiển thị view đăng nhập</li></ul></li><li>3. Người dùng chọn đăng nhập bằng facebook hoặc đăng nhập qua tài khoản đã đăng kí</li></ol>

Bảng 10: Đặc tả use case Thêm khóa học

## 2 Biểu đồ lớp

### 2.1 Biểu đồ lớp tổng thể



Hình 10: Biểu đồ lớp

## 2.2 Mô tả chi tiết

### 2.2.1. Lớp DownloadService

Tên Phương Thức	Mô Tả
downloadCourse()	Khi khóa học chưa được tải, người dùng cần tải dữ liệu về để cho những lần sử dụng sau
downloadToeicPart()	Khi các part trong đề thi chưa tải, người dùng cần tải dữ liệu về để làm bài thi toeic
unZip()	Khi dữ liệu download về là 1 file .zip ,phương thức này để giải nén file đó
removeFileZip()	Sau khi import dữ liệu vào database hoàn tất, sẽ xóa file .zip vừa down

Bảng 11: Mô tả lớp DownloadService

### 2.2.2. Lớp DatabaseService

Tên Phương Thức	Mô Tả
init ()	Khởi tạo database
importData()	Lấy dữ liệu insert vào Database
createTable()	Tạo các bảng trong database
getAllCourses()	Trả về tất cả các khóa học trong database
getAllLessons()	Trả về tất cả Bài học theo từng khóa học
getAllWords()	Trả về tất cả các từ theo từng bài

Bảng 12: Mô tả lớp databaseService

### 2.2.3.Lớp ToeicService

Tên Phương Thức	Mô Tả
generateToeicTest()	Sinh các đề thi toeic 1 cách ngẫu nhiên
getPartData()	Lấy dữ liệu của từng Part
savaData()	Lưu dữ liệu lại vào database

Bảng 13 Mô tả lớp ToecicService

### 2.2.4.Lớp CourseService

Tên Phương Thức	Mô Tả
generateListenPractise()	Sinh các câu hỏi trong luyện nghe 1 cách ngẫu nhiên
generateWritePractise()	Sinh các câu hỏi trong luyện viết 1 cách ngẫu nhiên
generateFlashcardPractise()	Sinh các câu hỏi trong flashcard 1 cách ngẫu nhiên
shuffleArray()	Xáo trộn các phần tử của mảng

Bảng 14: Mô tả lớp CourseService

### 2.2.5 Lớp ForumService

Tên Phương Thức	Mô Tả
isLogin ()	Kiểm tra xem người dùng đã đăng nhập hay chưa
getAllPosts()	Lấy về tất cả các bài đăng
addPost()	Đăng l bài lên diễn đàn
getAllComments()	Lấy về tất cả các comment của bài đăng



addComment()	Bình luận về bài đăng
--------------	-----------------------

Bảng 15: Mô tả lớp ForumService

### 2.2.6.Lớp Course

Tên Thuộc tính & Phương Thức	Mô Tả
id	Thuộc tính id : Mã khóa học
name	Thuộc tính name : Tên khóa học
lessons	Thuộc tính lessons : danh sách bài học trong khóa
totalWords	Thuộc tính totalWords : số từ trong khóa học
downloaded	Thuộc tính downloaded : Khóa học được tải hay chưa
set()	Phương thức set() : Thiết lập giá trị cho thuộc tính
get()	Phương thức get() : Lấy giá trị của thuộc tính

Bảng 16: Mô tả lớp Course

### 2.2.7.Lớp Lesson

Tên Thuộc tính & Phương Thức	Mô Tả
id	Thuộc tính id : Mã bài học
name	Thuộc tính name : Tên bài học
totalWords	Thuộc tính totalWords : số từ trong bài học
words	Thuộc tính words : danh sách các từ trong bài học

set()	Phương thức set() : Thiết lập giá trị cho thuộc tính
get()	Phương thức get() : Lấy giá trị của thuộc tính

Bảng 17: Mô tả lớp Lesson

### 2.2.8.Lớp Word

Tên Thuộc tính & Phương Thức	Mô Tả
id	Thuộc tính id : Mã từ
idLesson	Thuộc tính idLesson : Mã bài học
word	Thuộc tính word : Từ tiếng anh
mean	Thuộc tính mean : Từ tiếng việt
phonetic	Thuộc tính phonetic : phiên âm của từ tiếng anh
set()	Phương thức set() : Thiết lập giá trị cho thuộc tính
get()	Phương thức get() : Lấy giá trị của thuộc tính

Bảng 18: Mô tả lớp Word

### 2.2.9.Lớp Post

Tên Thuộc tính & Phương Thức	Mô Tả
id	Thuộc tính id : Mã bài post
content	Thuộc tính content : Nội dung của bài post

date	Thuộc tính date : Thời gian đăng bài
user	Thuộc tính user : người đăng bài
comments	Thuộc tính comments : danh sách tất cả bình luận của bài đăng
set()	Phương thức set() : Thiết lập giá trị cho thuộc tính
get()	Phương thức get() : Lấy giá trị của thuộc tính

Bảng 19: Mô tả lớp Post

### 2.2.10.Lớp Comment

Tên Thuộc tính & Phương Thức	Mô Tả
id	Thuộc tính id : Mã bình luận
content	Thuộc tính content : Nội dung của lời bình luận
user	Thuộc tính user : Người bình luận
set()	Phương thức set() : Thiết lập giá trị cho thuộc tính
get()	Phương thức get() : Lấy giá trị của thuộc tính

Bảng 20: Mô tả lớp Comment

### 2.2.11.Lớp User

Tên Thuộc tính & Phương Thức	Mô Tả
email	Thuộc tính email : email của user đăng kí tài khoản
password	Thuộc tính password : mật khẩu của user đăng kí tài khoản

username	Thuộc tính username : Tên đầy đủ của người dùng khi đăng kí tài khoản
postQuestion()	Phương thức postQuestion() : đăng bài lên diễn đàn
commentReplyQuestion()	Phương thức commentReplyQuestion() : Bình luận trên diễn đàn
Login()	Phương thức login : đăng nhập vào hệ thống
set()	Phương thức set() : Thiết lập giá trị cho thuộc tính
get()	Phương thức get() : Lấy giá trị của thuộc tính

Bảng 21: Mô tả lớp User

### 2.2.12.Lớp ToeicTest

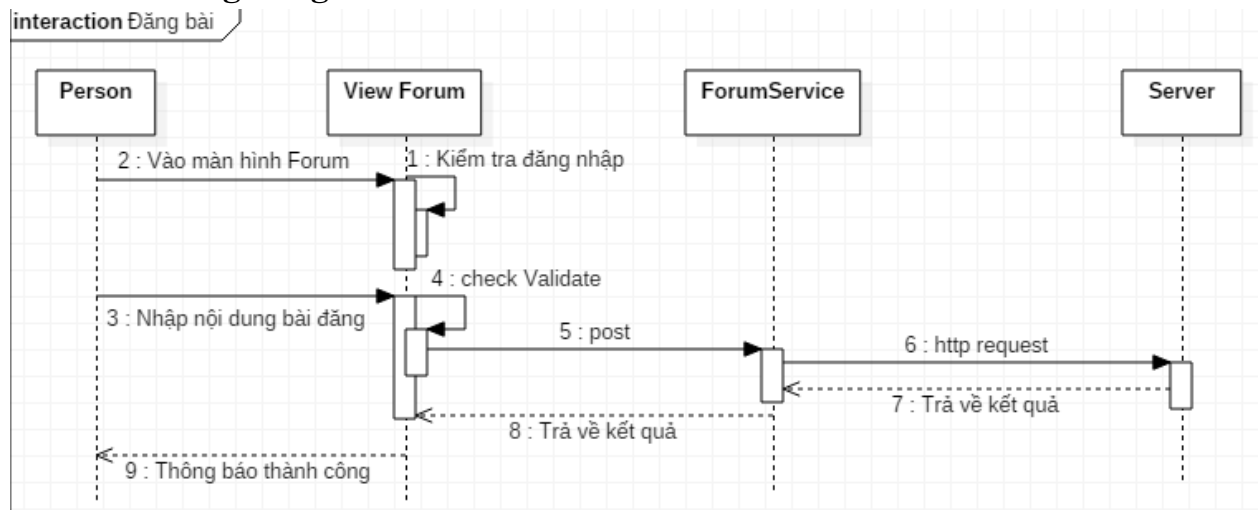
Tên Thuộc tính & Phương Thức	Mô Tả
id	Thuộc tính id : Mã đề thi
reading	Thuộc tính reading : số điểm đọc của người dùng
listening	Thuộc tính listening : số điểm nghe của người dùng
finished	Thuộc tính finished : đề thi đã hoàn thành hay chưa
commentReplyQuestion()	Phương thức commentReplyQuestion() : Bình luận trên diễn đàn
isFinish()	Phương thức isFinish : kiểm tra xem đề thi đã hoàn thành hay chưa
set()	Phương thức set() : Thiết lập giá trị cho thuộc tính

get()	Phương thức get() : Lấy giá trị của thuộc tính
-------	--

Bảng 22: Mô tả lớp ToeicTest

### 3 Biểu đồ trình tự

#### 3.1 Chức năng đăng bài

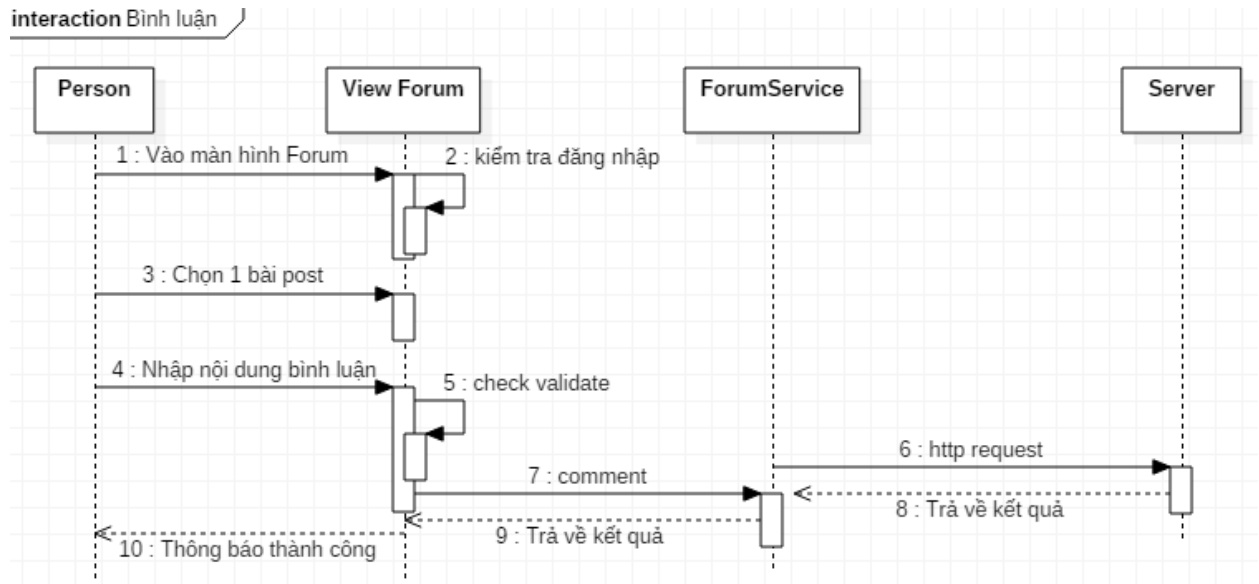


Hình 11: Biểu đồ trình tự Chức năng đăng bài

Mô tả :

- Người dùng vào màn hình diễn đàn trên Menu
- Nếu người dùng chưa đăng nhập thì yêu cầu đăng nhập
- Nhập nội dung bài đăng.
- Hệ thống kiểm tra đầu vào hợp lệ hay không
- Nếu hợp lệ thì gọi function post() trong ViewForum.
- post() function sẽ gọi api trên server sử dụng http request.
- Server trả về dữ liệu
- Thông báo đăng bài thành công .

#### 3.2 Biểu đồ trình tự cho chức năng bình luận

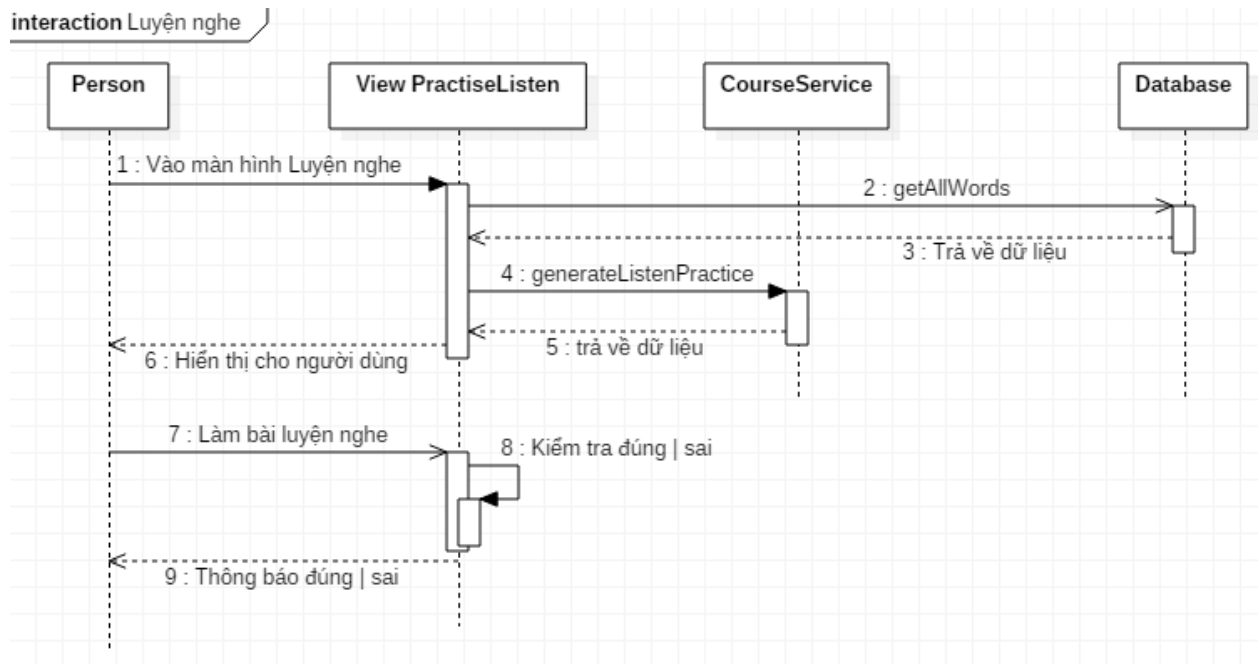


Hình 12: Biểu đồ trình tự Chức năng bình luận

Mô tả :

- Người dùng vào màn hình diễn đàn trên Menu
- Nếu người dùng chưa đăng nhập thì yêu cầu đăng nhập
- Chọn một bài đăng bất kì.
- Nhập nội dung bình luận
- Hệ thống kiểm tra đầu vào hợp lệ hay không
- Nếu hợp lệ thì gọi function comment() trong ViewForum.
- comment ()function sẽ gọi api trên server sử dụng http request.
- Server trả về dữ liệu
- Thông báo bình luận thành công .

### 3.3 Biểu đồ trình tự cho chức năng luyện nghe

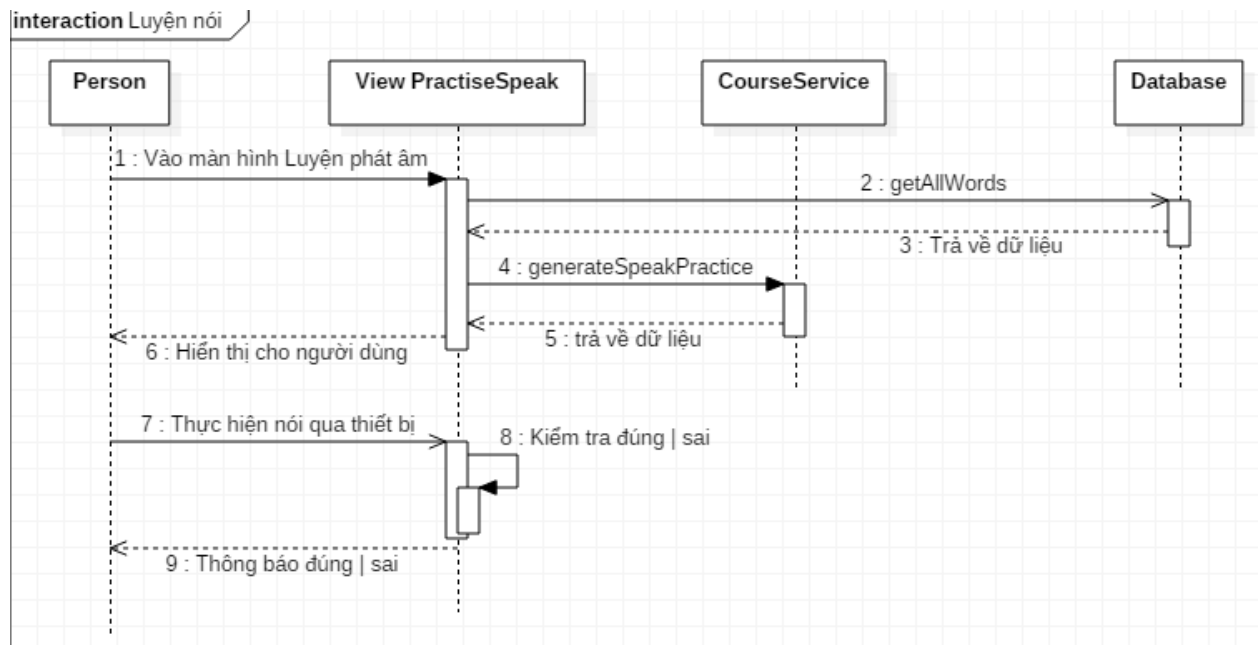


Hình 13: Biểu đồ trình tự Chức năng luyện nghe

Mô tả :

- Người dùng vào màn hình Luyện Nghe
- Hệ thống lấy tất cả các từ của khóa học trong database
- Hệ thống sinh câu hỏi ngẫu nhiên bằng cách gọi function `generateListenPractice()`.
- Hệ thống trả về dữ liệu.
- Hiển thị bài luyện tập nghe lên màn hình.
- Người dùng làm bài luyện nghe bằng cách chọn trắc nghiệm
- Hệ thống kiểm tra người dùng chọn đúng hay sai
- Thông báo cho người dùng.

### 3.4 Biểu đồ trình tự cho chức năng luyện nói



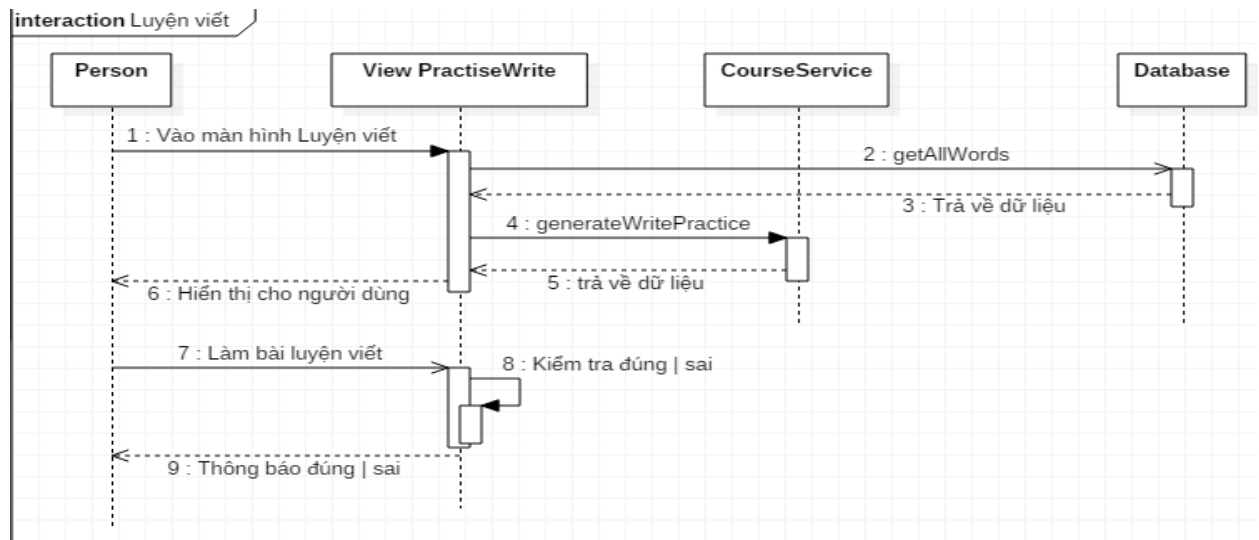
Hình 14: Biểu đồ trình tự Chức năng luyện nói

Mô tả :

- Người dùng vào màn hình Luyện Phát Âm
- Hệ thống lấy tất cả các từ của khóa học trong database
- Người dùng làm bài luyện nói bằng cách nói vào thiết bị
- Hệ thống kiểm tra người dùng nói đúng hay sai
- Thông báo cho người dùng.

### 3.5 Biểu đồ trình tự cho chức năng luyện viết



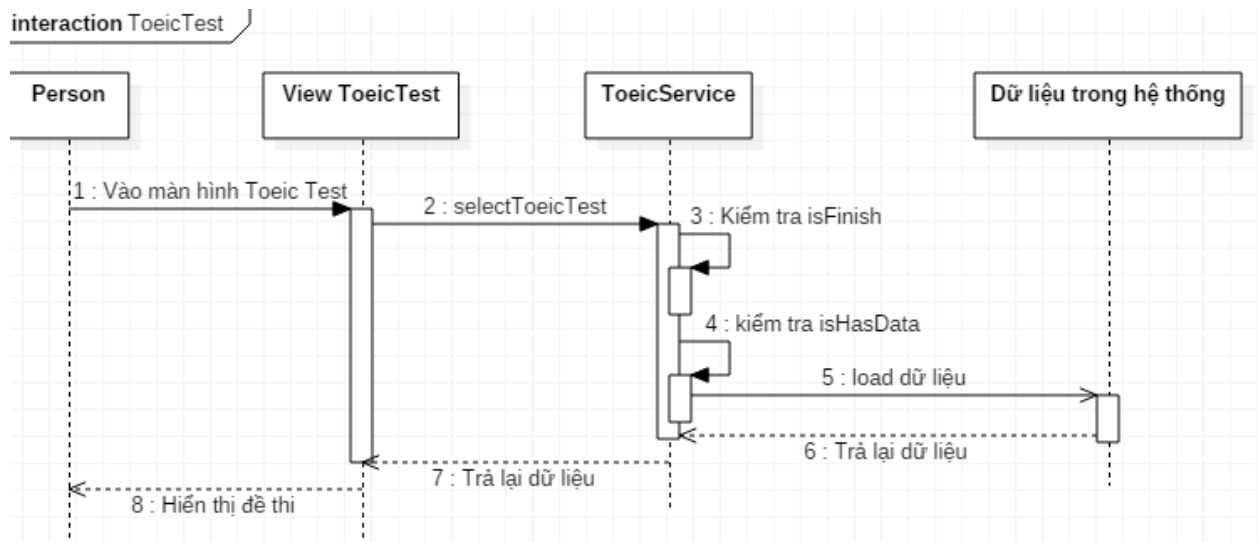


Hình 15: Biểu đồ trình tự Chức năng luyện viết

Mô tả :

- Người dùng vào màn hình Luyện Viết
- Hệ thống lấy tất cả các từ của khóa học trong database
- Hệ thống sinh câu hỏi ngẫu nhiên bằng cách gọi function generateWritePractice.
- Hệ thống trả về dữ liệu.
- Hiển thị bài luyện tập viết lên màn hình.
- Người dùng làm bài luyện viết
- Hệ thống kiểm tra người dùng làm đúng hay sai
- Thông báo cho người dùng.

### 3.6 Biểu đồ trình tự cho chức năng Làm đề toeic



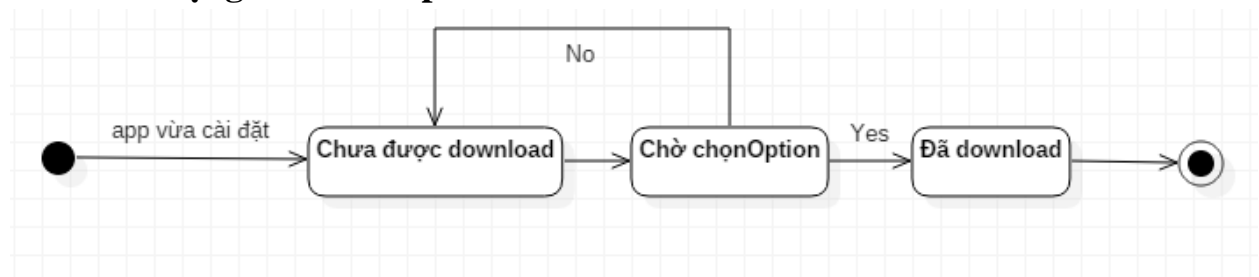
Hình 16 : Biểu đồ trình tự Chức năng làm đề Toeic

Mô tả :

- Người dùng vào màn hình ToeicTest
- Người dùng chọn đề thi toeic.
- Hệ thống kiểm tra xem đề thi đã được làm hay chưa.
- Nếu đã làm thì thông báo cho người dùng
- Hệ thống kiểm tra đã có dữ liệu đề thi trong máy hay chưa
- Nếu chưa có thông báo yêu cầu tải dữ liệu cho người dùng
- Load dữ liệu từ trong máy
- Máy trả về dữ liệu
- Hiển thị đề thi cho người dùng.

## 4 Biểu đồ trạng thái

### 4.1 Biểu đồ trạng thái của lớp Course

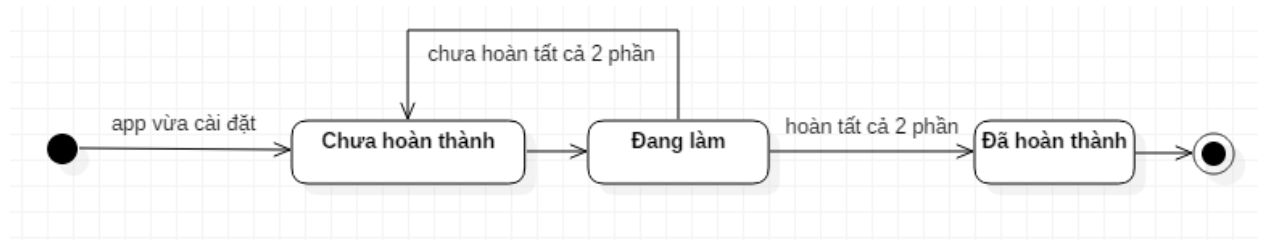


Hình 17: Biểu đồ trạng thái của của lớp Course

Mô tả :

- Khi ứng dụng được mới được cài đặt, các khóa học (lớp Course) ở trạng thái là chưa được download,
- Khi người dùng nhấn vào một khóa học bất kì thì khóa học chuyển sang trạng thái chờ chọn option.option này yêu cầu download hay không
- Nếu chọn **YES** thì chuyển sang trạng thái đã download, nếu chọn **NO** thì chuyển trạng chưa download

#### 4.2 Biểu đồ trạng thái của lớp ToeicTest



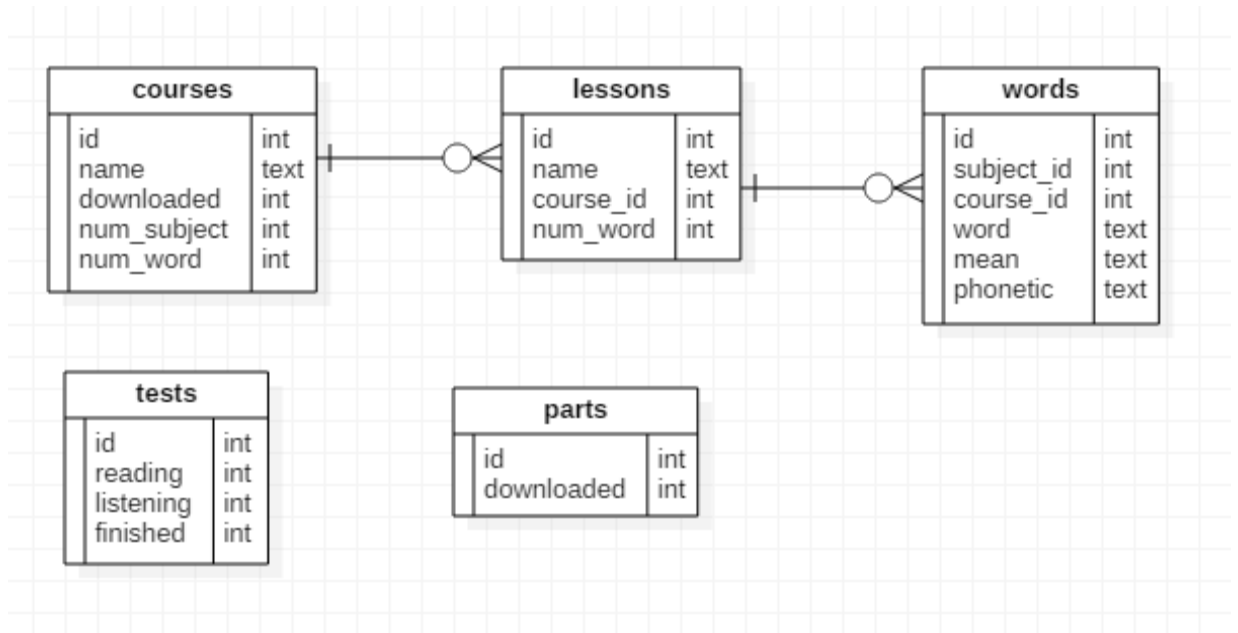
Hình 18: Biểu đồ trạng thái của của lớp ToeicTest

Mô tả :

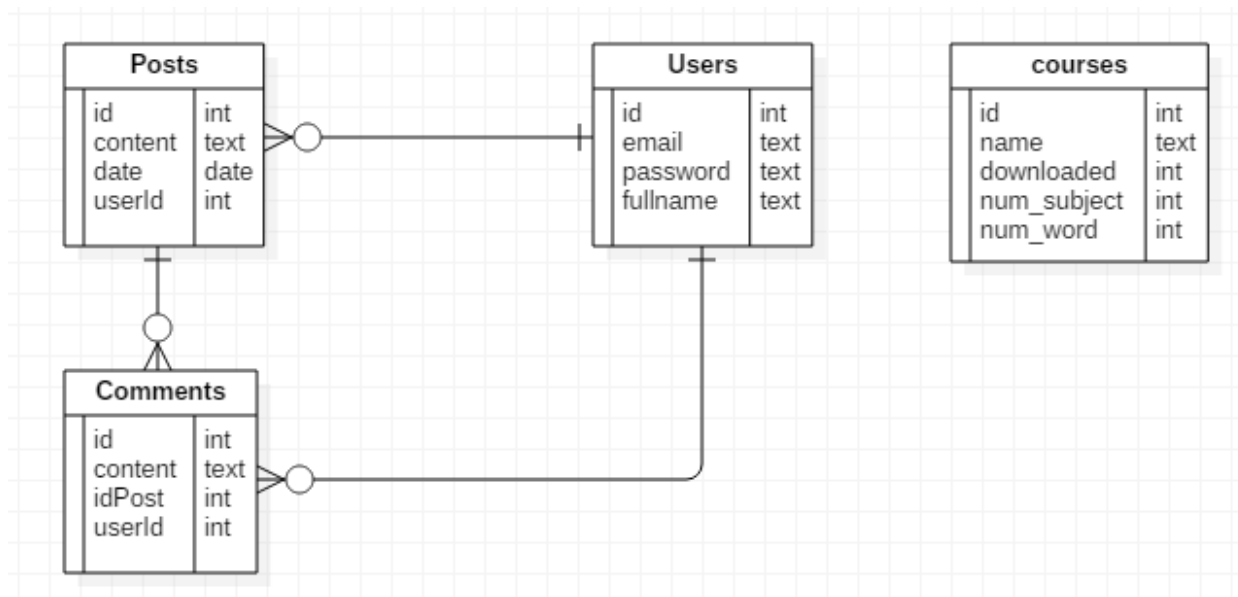
- Khi ứng dụng được mới được cài đặt, các đề thi(lớp ToeicTest) ở trạng thái là chưa hoàn thành
- Khi người dùng nhấn vào đề thi để làm thì ToeicTest chuyển sang trạng thái đang làm.
- Nếu đang làm dở mà thoát ứng dụng , tức là chưa hoàn tất thì chuyển trạng thái chưa hoàn thành, nếu hoàn thành cả 2 phần listening và reading và đã lưu lại bài làm thì chuyển trạng thái đã hoàn thành.

## 5 Thiết kế cơ sở dữ liệu

### 5.1 Mô hình



Hình 19: Cơ sở dữ liệu của client



Hình 20: Cơ sở dữ liệu của server

### 5.2 Mô tả chi tiết

#### 5.2.1 Bảng courses

Bảng courses dùng để lưu trữ thông tin về khóa học

Tên Cột	Kiểu dữ liệu	Mô Tả
id	int	Mã khóa học
name	Text	Tên khóa học
downloaded	int	Khóa học đã được download hay chưa
num_subject	int	Số lượng bài học trong khóa học
num_word	int	Số lượng từ trong khóa học

Bảng 23: Mô tả bảng courses

### 5.2.2 Bảng lessons

Bảng lessons dùng để lưu trữ thông tin về bài học tại client

Tên Cột	Kiểu dữ liệu	Mô Tả
id	int	Mã bài học
name	Text	Tên bài học
course_id	int	Mã của khóa học
num_word	int	Số lượng từ trong bài học

Bảng 24: Mô tả bảng lessons

### 5.2.3 Bảng words

Bảng words dùng để lưu trữ thông tin về các từ tại client

Tên Cột	Kiểu dữ liệu	Mô Tả
id	int	Mã của từ

subject_id	int	Mã bài học
course_id	int	Mã khóa học
word	Text	Từ tiếng anh của từ
mean	Text	Từ tiếng việt của từ
phonetic	Text	Phiên âm của từ

Bảng 25: Mô tả bảng words

#### 5.2.4 Bảng tests

Bảng tests dùng để lưu trữ thông tin về đề thi toeic tại client

Tên Cột	Kiểu dữ liệu	Mô Tả
id	int	Mã đề thi toeic
reading	int	Số điểm phần đọc của đề thi
listening	int	Số điểm phần nghe của bài thi
finished	int	Đề thi đã hoàn thành hay chưa

Bảng 26: Mô tả bảng tests

#### 5.2.5 Bảng parts

Bảng parts dùng để lưu trữ thông tin về Part tại client

Tên Cột	Kiểu dữ liệu	Mô Tả
id	int	Mã của part
downloaded	int	Dữ liệu của Part đã được download hay chưa

Bảng 27: Mô tả bảng parts

### 5.2.2 Bảng Users

Bảng Users dùng để lưu trữ thông tin về người dùng tại server

Tên Cột	Kiểu dữ liệu	Mô Tả
id	int	Mã người dùng
email	Text	Email đăng kí tài khoản
password	Text	Mật khẩu của tài khoản
fullname	Text	Tên đầy đủ của người dùng

Bảng 28: Mô tả bảng users

### 5.2.2 Bảng Posts

Bảng Posts dùng để lưu trữ thông tin về đăng tại server

Tên Cột	Kiểu dữ liệu	Mô Tả
id	int	Mã bài đăng
content	Text	Nội dung bài đăng
date	Date	Thời gian đăng bài
userId	int	Mã người dùng đăng bài

Bảng 29: Mô tả bảng posts

### 5.2.2 Bảng Comments

Bảng lessons dùng để lưu trữ thông tin về bài học tại client

Tên Cột	Kiểu dữ liệu	Mô Tả
id	int	Mã bình luận
content	Text	Nội dung bình luận
idPost	int	Mã của bài đăng
userId	int	Mã của người bình luận

Bảng 30: Mô tả bảng comment