

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN  
KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG**

**TRẦN VÕ KHÔI NGUYỄN - 09520193  
HUỖNH THÁI DƯƠNG - 09520049**

**ĐỒ ÁN TỐT NGHIỆP  
Xây Dựng Ứng Dụng Hỗ Trợ Học Tiếng Anh  
Cho Thiết Bị Di Động Chạy Nền Tảng Android**

**KỸ SƯ NGÀNH MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG**

**GIẢNG VIÊN HƯỚNG DẪN  
THS. TRẦN BÁ NHIỆM**

**TP. HỒ CHÍ MINH, NĂM 2014**

## MỞ ĐẦU

Hiện nay, tiếng Anh được coi là ngôn ngữ quốc tế số một trên thế giới. Hàng triệu người từ các nền văn hóa khác nhau đều nỗ lực học tiếng Anh mỗi ngày. Ở Việt Nam, tiếng Anh cũng đã chiếm được vị trí quan trọng kể từ khi đất nước bắt đầu thực hiện chính sách mở cửa hội nhập ra khu vực và thế giới. Ở nhiều lĩnh vực khác nhau trong xã hội như kinh tế, chính trị, khoa học, du lịch, vv... đều rất cần những người có trình độ tiếng Anh giỏi. Người ta học và sử dụng tiếng Anh với nhiều mục đích khác nhau để đáp ứng nhu cầu giao tiếp ngày càng lớn trong xã hội, để kiếm được công việc tốt, để được thăng chức, hay để giành được cơ hội đi du học và làm việc ở nước ngoài. Cùng với đó, sự phát triển của công nghệ di động đã bùng nổ trong vài năm qua. Các thiết bị cầm tay đã trở nên mạnh mẽ và rất phổ biến. Trong vài năm trở lại đây, hệ điều hành Android ra đời và trở thành nền tảng điện thoại thông minh phổ biến nhất thế giới, với sự kế thừa những ưu việt của các hệ điều hành ra đời trước và sự kết hợp của nhiều công nghệ tiên tiến nhất hiện nay. Những ưu điểm dễ thấy nhất của Android chính là khả năng tùy biến nhanh chóng, dễ làm quen, ứng dụng hỗ trợ phong phú, tương thích với đa cấu hình phần cứng của các nhà sản xuất. Dù đã hay chưa từng sử dụng smartphone Android thì bạn cũng không cần quá lo lắng về việc thay đổi. Hầu như tất cả những smartphone sử dụng hệ điều hành Android đều rất dễ sử dụng, đơn giản trong tùy biến theo sở thích cá nhân, cũng như có rất nhiều ứng dụng để thỏa mãn nhu cầu của từng người dùng. Đồng thời Android có mã nguồn mở đã cho phép các nhà phát triển thiết bị, mạng di động và các lập trình viên được điều chỉnh và phân phối Android một cách tự do. Bởi vậy trước nhu cầu và lợi thế đó, nhóm em quyết định nghiên cứu và xây dựng ứng dụng: **Xây dựng ứng dụng hỗ trợ học tiếng Anh trên thiết bị di động chạy nền tảng Android.**

## LỜI CẢM ƠN

Đầu tiên cho phép chúng em được gửi lời cảm ơn chân thành và sâu sắc nhất đến quý thầy cô giáo Đại Học Công Nghệ Thông Tin – ĐHQG TP.Hồ Chí Minh, đặc biệt là quý thầy cô Khoa Mạng Máy Tính Và Truyền Thông đã tận tình chỉ dạy và quan tâm trong suốt quá trình học tập và rèn luyện tại trường.

Chúng con gửi tất cả lòng biết ơn chân thành và sự kính trọng của chúng con đến cha mẹ và gia đình, những người đã sinh thành, dưỡng dục chúng con tự tin đi trên bước chân của chính mình cho đến ngày hôm nay, cha mẹ luôn bên cạnh và là chỗ dựa vững chắc cho chúng con vượt qua mỗi khi chúng con gặp khó khăn trong cuộc đời.

Chúng em trân trọng biết ơn thầy Trần Bá Nhiệm đã tận tình hướng dẫn, chỉ bảo, góp ý cho chúng em trong suốt quá trình thực hiện đề tài luận văn đề hôm nay có thể hoàn thành khóa luận tốt nghiệp của mình. Trong khoảng thời gian qua cô là người định hướng và giúp đỡ chúng em trước những khó khăn trở ngại, rồi những buổi gặp trao đổi những kiến thức nhà trường cũng như những bài học cuộc sống hàng tuần sẽ là những ký ức không bao giờ quên trong mỗi bước đi về phía trước của chúng em.

Xin chân thành cảm ơn đến những người bạn, những người mà luôn bên cạnh đồng viên, chia sẻ vui buồn và khó khăn với chúng tôi trong suốt thời gian qua.

Hồ Chí Minh, 25 tháng 02 năm 2014

**NHẬN XÉT**  
**(Của giảng viên hướng dẫn)**

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

**NHẬN XÉT**  
**(Của giảng viên phản biện)**

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

# MỤC LỤC

Chương 1	TỔNG QUAN .....	1
1.1	Giới thiệu tổng quan và lý do chọn đề tài.....	1
1.1.1	Sơ lược về tiếng Anh .....	1
1.1.2	Sự thông dụng của tiếng Anh.....	1
1.1.3	Sự ra đời và phát triển của điện thoại thông minh.....	4
1.2	Mục tiêu đề tài .....	7
1.2.1	Lý thuyết .....	7
1.2.2	Thực tiễn .....	8
1.3	Các bước nghiên cứu .....	8
1.4	Bố cục đề tài .....	8
Chương 2	CƠ SỞ LÝ THUYẾT.....	9
2.1	Hệ điều hành Android.....	9
2.1.1	Đặc điểm .....	11
2.1.2	So sánh với các hệ điều hành cùng loại khác.....	12
2.2	Kiến trúc và các thành phần.....	14
2.2.1	Kiến trúc tổng quát.....	14
2.2.2	Hệ điều hành .....	14
2.2.3	Các thành phần của một ứng dụng Android .....	19
2.3	Công nghệ web .....	29
2.3.1	Ngôn ngữ lập trình PHP.....	29

2.3.2	Hệ quản trị cơ sở dữ liệu .....	32
2.3.3	Công nghệ web liên quan khác .....	34
2.3.4	YII framework.....	36
2.3.5	Web Service .....	43
2.3.6	Công nghệ sử dụng .....	52
Chương 3	PHÂN TÍCH THIẾT KẾ.....	54
3.1	Xác định yêu cầu.....	54
3.1.1	Chức năng chính .....	54
3.1.2	Mô hình hoạt động client-server .....	56
3.2	Phân tích thiết kế.....	61
3.2.1	Thiết kế cơ sở dữ liệu.....	61
3.2.2	Thiết kế mô hình lớp.....	68
Chương 4	XÂY DỰNG ỨNG DỤNG.....	77
4.1	Client.....	77
4.1.1	Danh sách các màn hình.....	77
4.1.2	Trang đăng nhập.....	78
4.1.3	Trang chính (Home).....	80
4.1.4	Trang thông tin cá nhân .....	83
4.1.5	Trang từ điển.....	84
4.1.6	Trang thành ngữ .....	85
4.1.7	Trang ngữ pháp .....	85
4.1.8	Trang bài nghe .....	86

4.1.9	Trang video .....	87
4.1.10	Các trang về trắc nghiệm .....	88
4.1.11	Trang yêu thích .....	90
4.1.12	Trang kết quả thi .....	90
4.1.13	Các trang chức năng khác .....	91
4.2	Server .....	91
4.2.1	Trang chủ đăng nhập.....	91
4.2.2	Trang thống kê hoạt động .....	93
4.2.3	Trang thống kê câu hỏi.....	94
4.2.4	Trang thống kê video/ngữ pháp .....	95
4.2.5	Các trang quản lý bài thi .....	96
4.2.6	Trang quản lý yêu thích .....	97
4.2.7	Trang quản lý kết quả thi .....	97
4.2.8	Các trang quản lý khác.....	98
4.2.9	Trang backup cơ sở dữ liệu.....	99
4.2.10	Trang quản lý file backup .....	100
Chương 5	<b>KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN</b> .....	102
	<b>DANH MỤC TÀI LIỆU THAM KHẢO</b> .....	104



## **DANH MỤC BẢNG BIỂU, SƠ ĐỒ**

Bảng 2.1 So Sánh Các Hệ Điều Hành Điện Thoại .....	12
Bảng 2.2 Bảng So Sánh Các Công Nghệ Lập Trình Web.....	30
Bảng 2.3 Bảng So Sánh Các Hệ Quản Trị Cơ Sở Dữ Liệu .....	33
Bảng 3.1 Các bảng và thực thể trong sơ sở dữ liệu .....	63
Bảng 3.2 Danh sách các mối kết hợp .....	64
Bảng 3.3 Bảng user và các thuộc tính .....	65
Bảng 3.4 Bảng từ điển và các thuộc tính .....	66
Bảng 3.5 Bảng thành ngữ và các thuộc tính .....	66
Bảng 3.6 Bảng audio và các thuộc tính .....	66
Bảng 3.7 Bảng video và các thuộc tính .....	67
Bảng 3.8 Bảng điểm và các thuộc tính .....	67
Bảng 3.9 Bảng favorite và các thuộc tính.....	67
Bảng 3.10 Bảng lienhe và các thuộc tính .....	68
Bảng 3.11 Bảng level và các thuộc tính .....	68
Bảng 4.1 Danh sách các màn hình.....	78
Bảng 4.2 Bảng mô tả trang đăng nhập .....	79
Bảng 4.3 Bảng mô tả scrollview người dùng .....	80
Bảng 4.4 Bảng mô tả trang home .....	81
Bảng 4.5 Bảng mô tả trang thông tin cá nhân .....	84
Bảng 4.6 Bảng mô tả trang từ điển.....	84
Bảng 4.7 Bảng mô tả trang thành ngữ.....	85
Bảng 4.8 Bảng mô tả trang ngữ pháp .....	86
Bảng 4.9 Bảng mô tả trang audio .....	87
Bảng 4.10 Bảng mô tả trang video .....	88

## DANH MỤC HÌNH ẢNH

Hình 2.1 Hệ điều hành Android.....	9
Hình 2.2 Kiến trúc tổng quát .....	14
Hình 2.3 Linux kernel.....	15
Hình 2.4 Libraries.....	16
Hình 2.5 Android Runtime .....	18
Hình 2.6 Application Framework.....	19
Hình 2.7 Vòng đời của một activity .....	22
Hình 2.8 Sơ đồ chuyển trạng thái của service .....	24
Hình 2.9 Cấu trúc tĩnh của ứng dụng Yii .....	38
Hình 2.10 Luồng làm việc điển hình của một ứng dụng Yii.....	39
Hình 2.11 Yii Data Access Objects.....	42
Hình 2.12 Thành phần chính của Web Service .....	46
Hình 2.13 Kiến trúc của Web Service .....	48
Hình 3.1 Chức năng dành cho khách.....	56
Hình 3.2 Chức năng dành cho thành viên .....	57
Hình 3.3 Mô hình xử lý đăng nhập.....	58
Hình 3.4 Mô hình xử lý đăng kí .....	59
Hình 3.5 Mô hình xử lý thi trắc nghiệm.....	60
Hình 3.6 Mô hình xử lý thay đổi thông tin.....	61
Hình 3.7 Sơ đồ tổng quát mối liên hệ giữa các bảng trong cơ sở dữ liệu .....	62
Hình 3.8 Mô hình lớp của chương trình. ....	69
Hình 3.9 Mô hình MVC trong YII Framework.....	75
Hình 3.10 Mô hình MVC cụ thể cho quản lý user. ....	76
Hình 4.1 Màn hình đăng nhập .....	79
Hình 4.2 Màn hình Home .....	81
Hình 4.3 Màn hình menu left.....	82

Hình 4.4 Các tab viewpaper .....	83
Hình 4.5 Màn hình thông tin cá nhân .....	83
Hình 4.6 Màn hình từ điển.....	84
Hình 4.7 Màn hình thành ngữ.....	85
Hình 4.8 Màn hình ngữ pháp.....	86
Hình 4.9 Màn hình bài nghe .....	87
Hình 4.10 Màn hình xem video .....	88
Hình 4.11 Trang lựa chọn thi trắc nghiệm.....	89
Hình 4.12 Màn hình thi trắc nghiệm.....	90
Hình 4.13 Màn hình trang yêu thích.....	90
Hình 4.14 Màn hình kết quả thi .....	91
Hình 4.15 Trang chủ đăng nhập .....	92
Hình 4.16 Giao diện trang chủ sau khi đăng nhập .....	93
Hình 4.17 Giao diện thống kê hoạt động.....	94
Hình 4.18 Giao diện trang thống kê câu hỏi.....	95
Hình 4.19 Giao diện trang thống kê ngữ pháp .....	96
Hình 4.20 Giao diện trang quản lý bài thi .....	96
Hình 4.21 Giao diện trang quản lý yêu thích.....	97
Hình 4.22 Giao diện quản lý kết quả thi.....	98
Hình 4.23 Giao diện trang backup database .....	100
Hình 4.24 Giao diện trang quản lý file backup .....	101

## **Chương 1 TỔNG QUAN**

### **1.1 Giới thiệu tổng quan và lý do chọn đề tài**

#### **1.1.1 Sơ lược về tiếng Anh**

Tiếng Anh (English) là một ngôn ngữ thuộc nhánh miền Tây của nhóm ngôn ngữ German trong ngữ hệ Ấn-Âu. Đây là ngôn ngữ sử dụng rộng rãi nhất thế giới. Nó được sử dụng là ngôn ngữ mẹ đẻ bởi một số lượng lớn người dân từ khắp thế giới tại Liên hiệp Anh, Mỹ, Canada, Cộng hòa Ireland, New Zealand và một số quốc đảo trong vùng Caribbean. Đây là ngôn ngữ mẹ đẻ được sử dụng nhiều thứ ba trên thế giới, sau Tiếng Hoa và Tiếng Tây Ban Nha. Đây là ngôn ngữ thứ hai được sử dụng rộng rãi và là ngôn ngữ chính thức trong các tổ chức lớn bao gồm Liên minh châu Âu, Khối Thịnh vượng chung Anh và đặc biệt là Liên hiệp Quốc.

Tiếng Anh được sử dụng phổ biến nhờ vào ảnh hưởng của Mỹ và Anh trên các lĩnh vực quân sự, kinh tế, khoa học, tin học, chính trị và văn hóa. Ở nhiều nước, trong đó có Việt Nam, người ta bắt buộc hoặc mặc định phải học tiếng Anh để đi làm.

#### **1.1.2 Sự thông dụng của tiếng Anh**

##### **➤ Số người sử dụng**

Ngày nay có khoảng một tỷ người nói tiếng Anh trên khắp thế giới và con số này ngày càng tăng lên. Tiếng Anh là ngôn ngữ được sử dụng rộng rãi nhưng lại không thay thế các ngôn ngữ khác, thay vào đó nó hỗ trợ các ngôn ngữ với các yếu tố sau:

- ✓ Hơn 250 triệu người Trung Quốc học tiếng Anh.
- ✓ Trong 80 nước, tiếng Anh là ngôn ngữ thông dụng thứ hai hay được phổ biến trong việc học.

- ✓ Ở Hồng Kông, 9 trên 10 học sinh trung học đều được học tiếng Anh.
  - ✓ Ở Pháp, để bắt đầu vào trung học các học sinh phải có ít nhất 4 năm học tiếng Anh hay tiếng Đức; hầu hết 85% học sinh là chọn học tiếng Anh.
  - ✓ Tại Nhật Bản, các học sinh trung học được đòi hỏi là phải có 6 năm học tiếng Anh trước khi tốt nghiệp.
- Trong các phương tiện truyền thông và giao thông

Tiếng Anh chiếm ưu thế trong giao thông vận tải và các phương tiện truyền thông. Trong lĩnh vực du lịch và ngôn ngữ cộng đồng của hàng không quốc tế, tiếng Anh đóng vai trò chính. Phi công, tiếp viên và kể cả các nhân viên kiểm soát đều nói tiếng Anh tại các phi trường quốc tế. Cờ và các tình hiệu ánh sáng được sử dụng trong ngành hàng hải, nhưng “nếu các tàu lớn cần truyền tín hiệu cho nhau bằng các thông điệp thì họ sẽ tìm kiếm một ngôn ngữ chung và thông dụng và khi đó tiếng Anh chắc chắn sẽ là chọn lựa chính”, câu nói của một người bảo vệ bờ biển của tại Mỹ, Werner Siems.

Năm trong số các đài phát thanh nổi tiếng là CBS, NBC, ABC, BBC và CBC được 300 triệu người chọn ra là các đài phát thanh tiếng Anh phổ biến nhất. Tiếng Anh cũng là ngôn ngữ phổ biến trên các chương trình TV thuộc truyền tải vệ tinh.

- Trong thời đại công nghệ thông tin

Ngôn ngữ của thời đại thông tin là tiếng Anh. Hơn 80% nguồn dự trữ thông tin của hơn 100 triệu máy tính khắp thế giới là tiếng Anh. 85% các cuộc trao đổi qua điện thoại quốc tế được sử dụng bằng tiếng Anh, cũng như vậy số lượng mail, các cuộc điện báo và truyền tín hiệu qua dây cáp. Chương trình chỉ dẫn trên máy tính và các chương trình phần mềm thường được dùng bằng tiếng Anh.

Tiếng Đức đã là một ngôn ngữ của khoa học. Nhưng ngày nay, hơn 80% các bản ghi chép khoa học được trình bày với ngôn ngữ thứ nhất là tiếng Anh.

Tương tự, phân nửa kỹ thuật và khoa học trên thế giới cũng được phổ biến bằng tiếng Anh và còn được dùng trong các lĩnh vực y học, điện tử và kỹ thuật không gian.

➤ Trong lĩnh vực kinh doanh quốc tế

Tiếng Anh trong lĩnh vực kinh doanh quốc tế ảnh hưởng đến hầu hết các lĩnh vực trong nền kinh tế Châu Âu. Cũng vậy tiếng Anh hầu như tham gia hầu hết vào các thành phần lãnh đạo của các doanh nghiệp.

Trong nền công nghiệp thực phẩm, các biển hiệu cho sản phẩm của họ thường được dùng bằng tiếng Anh như *Made in Germany*, họ không dùng các câu như *Hergestellt in deutschland* – câu trên có nghĩa là "sản xuất tại Đức" nhưng một dùng với tiếng Anh và một dùng với tiếng Đức. Các tập đoàn của nhiều quốc gia trên thế giới thường chọn tiếng Anh như lựa chọn chính của họ. Các tập đoàn như Datsun và Nissan đều gửi điện báo với ngôn ngữ tiếng Anh. Như những năm 1985, 80% nhân viên của tập đoàn Mitsui có thể nói, đọc và viết được tiếng Anh, tập đoàn Toyota thì mở các lớp tiếng Anh tại chức cho nhân viên của mình. Các lớp tiếng Anh đã được giữ lại ở Ả Rập Saudi cho các công nhân của tập đoàn dầu hỏa Aramco và trên ba lục địa thuộc Ngân hàng Chase Manhattan.

➤ Ngôn ngữ chung

Tiếng Anh dùng như là tiếng nói chung ở nhiều nước nơi mọi người nói nhiều ngôn ngữ khác nhau. Tại Ấn Độ, nơi có gần 200 ngôn ngữ khác nhau được sử dụng và chỉ có 30% người nói ngôn ngữ chính là tiếng Hindi. Khi Rajiv Gandhi đọc diễn văn quốc gia sau khi mẹ ông ta bị ám sát, ông ta đã nói bằng tiếng Anh. Tổ chức thương mại tự do Châu Âu làm việc chủ yếu bằng tiếng Anh mặc dù 6 nước thành viên đều không trực thuộc nước Anh.

➤ Ngôn ngữ chính thức

Tiếng Anh là ngôn ngữ nửa chính thức của 20 nước Châu Phi bao gồm Sierra Leone, Ghana, Nigeria, Liberia và Nam Phi. Các sinh viên được dạy tiếng Anh tại trường Đại học Makerere ở Uganda, trường đại học của thành phố Nairobi ở Kenya và trường đại học của thành phố Dar es Salaam ở Tanzania.

Tiếng Anh là ngôn ngữ thống nhất của hội đồng thế giới Thiên chúa giáo và là một ngôn ngữ chính thức của các thể vận hội và các cuộc thi hoa hậu hoàn vũ trên thế giới.

### 1.1.3 Sự ra đời và phát triển của điện thoại thông minh

#### 1.1.3.1 Định nghĩa

Điện thoại thông minh (tiếng Anh: Smart Phone) là điện thoại tích hợp một nền tảng hệ điều hành di động, với nhiều tính năng tiên tiến về điện toán và kết nối hơn các điện thoại di động thông thường.

Các ngành công nghiệp tin rằng điện thoại thông minh đa số có một màn hình độ phân giải cao hơn so với điện thoại truyền thống và điện thoại thông minh như một máy tính di động, vì nó có một hệ điều hành riêng biệt vì có thể hiển thị phù hợp các trang website bình thường và người dùng có thể thay đổi một giao diện. và sở hữu khả năng mở ứng dụng, tiện hơn và dễ dàng cài đặt lẫn gỡ bỏ ứng dụng. Điện thoại thông minh có màn hình cảm ứng độ phân giải cao, và sẵn sàng để gọi bàn phím ảo và viết chữ tay.

Điện thoại thông minh có thể tiến hành đa tác vụ thao tác, và có một đa phương tiện mạnh mẽ, Email, truy cập Internet và hoàn toàn có thể thay đổi các thiết bị truyền thống như MP3, MP4, PDA. Điện thoại thông minh có thể thay thế xử lý các vấn đề máy tính văn phòng và các vấn đề khác, nó có thể giao tiếp với mạng duy trì một thời gian kết nối liền mạch với thời gian, đồng thời có thể vô hiệu hóa mạng bất cứ lúc nào và đồng bộ hóa dữ liệu với máy tính, máy tính xách tay và các thiết bị khác.

### 1.1.3.2 Lịch sử

Lịch sử Smart Phone khởi nghiệp từ năm 1993, cùng với thời gian, nhiều thương hiệu mới ra đời, các tên tuổi cũ chìm vào dĩ vãng. Và rồi, sự xuất hiện của iPhone năm 2007 đã đưa nền công nghiệp điện tử này bước sang một trang mới. Sau đó, công nghệ được cải thiện với sự hiện diện của các thiết bị di động sử dụng tín hiệu vô tuyến không dây, lướt web, giải trí.... Sau đây là một cái nhìn trực quan vào sự phát triển của điện thoại thông minh trong suốt bề dày lịch sử.

- Năm 1993-1994: Thiết bị không dây đầu tiên - IBM Simon xuất hiện. Chiếc smartphone đầu tiên đã ra đời từ cách đây vừa tròn 20 năm (ngày 26/11/1993) tại hội chợ COMDEX diễn ra ở Las Vegas (Mỹ), IBM đã cho ra mắt chiếc điện thoại với tên gọi IBM Simon, hay còn được biết đến với tên mã Angler. Với trọng lượng lên đến gần 0,6kg. Máy được trang bị vi xử lý tốc độ 16MHz, 1MB bộ nhớ RAM cùng 1MB ổ cứng lưu trữ. Simon cũng được trang bị một màn hình cảm ứng rộng 4.5-inch. Tuy nhiên điểm khác biệt đó là màn hình cảm ứng trên Simon chỉ hiển thị đơn sắc và hoạt động trên hệ điều hành là một biến thể của hệ điều hành DOS, có tên gọi ROM-DOS.
- Năm 1996: Thiết bị Pilot 1000/5000 đầu tiên ra mắt tháng 3/1996 thu hút sự quan tâm, nhưng chưa thành công trên thị trường, Palm (lúc bấy giờ là một bộ phận của U.S. Robotics tách ra) bắt đầu đi vào thị trường PDA (Personal Digital Assistants - thiết bị trợ giúp cá nhân). Pilot 1000 với giá bán 299 USD có bộ nhớ 128 KB, Pilot 500 bộ nhớ 512 KB có giá 369 USD là hai model đầu tiên, máy có thiết kế vỏ màu xám, thiết kế cơ bản giống các sản phẩm hiện nay. Màn hình với độ phân giải 160 x 160 pixel, thiết bị sử dụng để đồng bộ với Windows (phiên bản 3.1 hoặc 95) và hai pin AAA cho phép hoạt động từ một tuần hoặc hơn. Tuy hiện nay



nó không còn hiện diện trên thị trường và trở thành “cổ vật” nhưng chủ đề cổ này chính là “nhân vật” đã mở đường cho cuộc cách mạng tư duy về các thiết bị di động.

- Năm 1998: Nokia 9110 Communicator lộ diện Nokia 9110 Communicator đích thực là thiết bị làm nền móng cho smartphone với thiết kế bàn phím QWERTY gập cùng nhiều trò chơi mới được cập nhật, do đó đã tạo thói quen giải trí trên chiếc điện thoại cá nhân cho người dùng. Bên cạnh đó, nó có 32 bit bộ xử lý Intel 24MHz và trọng lượng chỉ 317 gram.
- Năm 1999: Ericsson R380 - điện thoại thông minh đầu tiên chính thức trình làng. Người sử dụng R380 không cần phải mang theo nhiều thiết bị di động, bởi vì tất cả các thiết bị này đều được gói gọn trong chiếc điện thoại di động nhỏ bé này. Được kết nối quốc tế trên hơn 120 nước tại 5 châu lục thông qua dịch vụ WAP cung cấp thông tin Internet. Với một màn hình cảm ứng và đồ họa phong phú, cung cấp nhiều kỹ năng tổ chức cá nhân và liên lạc, cùng dịch vụ WAP trên R380 giúp người sử dụng nhận hoặc gửi thư điện tử, hay truy cập vào những trang Web,... R380 sử dụng hệ điều hành Symbian, một hệ điều hành được thiết kế riêng cho các thiết bị thông tin không dây.
- Năm 2002: Giới thiệu BlackBerry 5810 - Email & tính năng lướt web BlackBerry vào thị trường không dây (smartphone) với thiết bị 5810 có thể gửi email và cho phép người dùng lướt Internet. Trước khi RIM trình làng dòng điện thoại mới chạy hệ điều hành BlackBerry 10, chuyên gia về thiết bị di động Sascha Segan đã điểm lại những mẫu máy mang tính bước ngoặt của hãng này, đánh dấu một số thay đổi nào đó về tính năng và thiết kế chứ không nhất thiết là các sản phẩm thành công nhất, giống như Z10 là điện thoại BlackBerry 10 đầu tiên nhưng chưa chắc là thiết bị

chạy hệ điều hành này được ưa chuộng nhất để người dùng dễ so sánh BlackBerry 2013 sẽ có những khác biệt gì so với trước đây.

- Năm 2007: Điện thoại thông minh “đúng nghĩa” - Apple lộ diện iPhone Thời điểm này, Android đang lớn lên và trở thành một trong những nền tảng được chú ý nhiều nhất. Ra mắt năm 2007, hệ điều hành mở hiện còn đáng sợ hơn iPhone, BlackBerry, Windows Mobile hay Symbian. Android không ngừng mở rộng, nền tảng này được nhiều nhà sản xuất ủng hộ với giới thiệu di động mới. Cùng ngày, các nhà phát triển càng đề mắt hơn đến Android trên các dòng máy mới với tốc độ kinh ngạc, nhưng với sự ra đời của iPhone Apple, mọi thứ dường như thay đổi mãi mãi.
- Năm 2008 - 2012: Android chiếm lĩnh thị trường điện thoại thông minh Từ năm 2007 đến năm 2008, các hệ điều hành được sử dụng nhiều nhất cho điện thoại thông minh là Symbian, Blackberry ...nhưng cùng thời điểm này Google cũng đã giới thiệu Android, một mã nguồn mở hệ thống điều hành điện thoại thông minh chiếm lĩnh toàn bộ thị trường điện thoại di động cho đến nay, không chỉ điện thoại di động mà còn rất nhiều sản phẩm khác. Theo một thông tin được cung cấp bởi Google vào ngày 22/9/2012 thì có hơn 500 triệu thiết bị chạy Android và mỗi ngày thêm 1 triệu thiết bị được cập nhật.

## 1.2 Mục tiêu đề tài

### 1.2.1 Lý thuyết

- ✓ Nghiên cứu công cụ đồ họa như photoshop.
- ✓ Nghiên cứu lập trình trên hệ điều hành android.
- ✓ Nghiên cứu ngôn ngữ lập trình java.
- ✓ Nghiên cứu xây dựng giao diện với xml trong android.
- ✓ Nghiên cứu ngôn ngữ lập trình web php.

- ✓ Nghiên cứu hệ quản trị cơ sở dữ liệu mysql.
- ✓ Nghiên cứu html, css, javascript.

### 1.2.2 Thực tiễn

- ✓ Xây dựng ứng dụng gồm client, server có khả năng giao tiếp với nhau.
- ✓ Ứng dụng hỗ trợ đầy đủ tính năng của một chương trình dạy và học tiếng Anh như: học từ mới, ngữ pháp, thi thử, nghe audio, xem video.
- ✓ Ngoài ra người dùng còn có thể lưu các thông tin cá nhân, đánh dấu các bài đã học, xem lại kết quả thi để đánh giá được sự tiến bộ trong quá trình học.
- ✓ Liên kết các mạng xã hội facebook, google+, twitter.

## 1.3 Các bước nghiên cứu

- ✓ Phân tích yêu cầu.
- ✓ Lựa chọn công nghệ.
- ✓ Nghiên cứu cơ sở lý thuyết của công nghệ đã chọn.
- ✓ Áp dụng lý thuyết vào xây dựng ứng dụng thực tiễn.
- ✓ Kiểm tra, tham khảo các ứng dụng khác để tối ưu hóa ứng dụng.

## 1.4 Bộ cục đề tài

- ✓ Cơ sở lý thuyết.
- ✓ Phân tích thiết kế.
- ✓ Giải pháp xây dựng.
- ✓ Triển khai ứng dụng.
- ✓ Kết luận và hướng phát triển

## Chương 2 CƠ SỞ LÝ THUYẾT

### 2.1 Hệ điều hành Android



Hình 2.1 Hệ điều hành Android

Android là một hệ điều hành dựa trên nền tảng Linux được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như điện thoại thông minh và máy tính bảng. Ban đầu, Android được phát triển bởi Tổng công ty Android, với sự hỗ trợ tài chính từ Google và sau này được chính Google mua lại vào năm 2005. Android ra mắt vào năm 2007 cùng với tuyên bố thành lập Liên minh thiết bị cảm tay mở: một hiệp hội gồm các công ty phần cứng, phần mềm, và viễn thông với mục tiêu đẩy mạnh các tiêu chuẩn mở cho các thiết bị di động. Chiếc điện thoại đầu tiên chạy Android được bán vào tháng 10 năm 2008.

Android có mã nguồn mở và Google phát hành mã nguồn theo Giấy phép Apache. Chính mã nguồn mở cùng với một giấy phép không có nhiều ràng buộc đã cho phép các nhà phát triển thiết bị, mạng di động và các lập trình viên nhiệt huyết được điều chỉnh và phân phối Android một cách tự do. Ngoài ra, Android còn có một cộng đồng lập trình viên đông đảo chuyên viết các ứng dụng để mở rộng chức năng của thiết bị, bằng một loại ngôn ngữ lập trình Java có sửa đổi. Vào tháng 10 năm 2012, có khoảng 700.000 ứng dụng trên Android, và số lượt tải ứng dụng từ Google Play, cửa hàng ứng dụng chính của Android, ước tính khoảng 25 tỷ lượt.

Những yếu tố này đã giúp Android trở thành nền tảng điện thoại thông minh phổ biến nhất thế giới, vượt qua Symbian vào quý 4 năm 2010, và được các công ty công nghệ lựa chọn khi họ cần một hệ điều hành không nặng nề, có khả năng tinh chỉnh, và giá rẻ chạy trên các thiết bị công nghệ cao thay vì tạo dựng từ đầu. Bản chất mở của Android cũng khích lệ một đội ngũ đông đảo lập trình viên và những người đam mê sử dụng mã nguồn mở để tạo ra những dự án do cộng đồng quản lý. Những dự án này bổ sung các tính năng cao cấp cho những người dùng thích tìm tòi hoặc đưa Android vào các thiết bị ban đầu chạy hệ điều hành khác.

Android chiếm 75% thị phần điện thoại thông minh trên toàn thế giới vào thời điểm quý 3 năm 2012, với tổng cộng 500 triệu thiết bị đã được kích hoạt và 1,3 triệu lượt kích hoạt mỗi ngày. Sự thành công của hệ điều hành cũng khiến nó trở thành mục tiêu trong các vụ kiện liên quan đến bằng phát minh, góp mặt trong cái gọi là "cuộc chiến điện thoại thông minh" giữa các công ty công nghệ.

Được xây dựng trên một nền tảng mở, và một bộ thư viện đa năng, mạnh mẽ với nguyên lý mở, Android đã nhanh chóng được cộng đồng lập trình viên di động hưởng ứng mạnh mẽ. Nền tảng Android tích hợp nhiều tính năng nổi bật:

- ✓ Android là một hệ điều hành nhân Linux, đảm bảo sự tương tác với các phần cứng, quản lý bộ nhớ, điều khiển các tiến trình tối ưu cho các thiết bị di động.
- ✓ Bộ ứng dụng khung cho phép sử dụng lại và thay thế các thành phần riêng lẻ.
- ✓ Máy ảo Dalvik được tối ưu cho các thiết bị di động, chạy các ứng dụng lập trình trên ngôn ngữ Java.
- ✓ Các thư viện cho phát triển ứng dụng mã nguồn mở bao gồm SQLite, WebKit, OpenGL và trình quản lý đa phương tiện.
- ✓ Hỗ trợ các chuẩn đa phương tiện phổ biến, thoại trên nền GSM, Bluetooth EDGE, 3G và Wifi
- ✓ Hỗ trợ Camera, GPS, la bàn, máy đo gia tốc...
- ✓ Bộ phát triển ứng dụng SDK đầy đủ gồm thiết bị giả lập, công cụ sửa lỗi, tích hợp với Eclipse SDK.

Android cung cấp một tập hợp đầy đủ các phần mềm cho thiết bị di động bao gồm: hệ điều hành, các khung ứng dụng và các ứng dụng cơ bản.

### 2.1.1 Đặc điểm

#### ➤ Tính mở

Android được xây dựng từ dưới đi lên cho phép người phát triển tạo các ứng dụng di động hấp dẫn với đầy đủ các điểm mạnh của các thiết bị cầm tay hiện có. Android hoàn toàn mở, một ứng dụng có thể gọi tới bất kể một chức năng lõi của điện thoại như tạo cuộc gọi, gửi tin nhắn hay sử dụng máy ảnh, cho phép người phát triển tạo phong phú hơn, liên kết hơn các tính năng cho người dùng. Android được xây dựng trên nhân Linux mở. Thêm nữa, nó sử dụng một máy ảo mà đã được tối ưu hóa bộ nhớ và phần cứng với môi trường di động. Android là một mã nguồn mở, nó có thể được mở rộng để kết hợp tự do giữa

các công nghệ nổi trội. Nền tảng này sẽ tiếp tục phát triển bởi cộng đồng phát triển để tạo ra các ứng dụng di động hoàn hảo.

➤ Tính ngang hàng của các ứng dụng

Với Android, không có sự khác nhau giữa các ứng dụng điện thoại cơ bản với ứng dụng của bên thứ ba. Chúng được xây dựng để truy cập như nhau tới một loạt các ứng dụng và dịch vụ của điện thoại. Với các thiết bị được xây dựng trên nền tảng Android, người dùng có thể đáp ứng đầy đủ các nhu cầu mà họ thích. Chúng ta có thể đổi màn hình nền, kiểu gọi điện thoại, hay bất kể ứng dụng nào. Chúng ta thậm chí có thể hướng dẫn điện thoại chỉ xem những ảnh mình thích.

➤ Dễ xây dựng ứng dụng

Android cung cấp bộ thư viện giao diện lập trình ứng dụng đồ sộ và các công cụ để viết các ứng dụng phức tạp. Ví dụ, Android có thể cho phép người phát triển biết được vị trí của thiết bị và cho phép các thiết bị giao tiếp với nhau để có thể tạo nên mạng xã hội chia sẻ ngang hàng rộng khắp. Thêm nữa, Android còn bao gồm một bộ công cụ đầy đủ giúp cho việc phát triển trở nên dễ dàng.

### 2.1.2 So sánh với các hệ điều hành cùng loại khác

*Bảng 2.1 So Sánh Các Hệ Điều Hành Điện Thoại*

	Ưu điểm	Nhược điểm
<b>Android</b>	Google Android là nền tảng mở, cho phép người dùng có thể tùy biến nền tảng theo ý thích, hơn nữa lại có một Liên minh thiết bị cầm tay mở hậu thuẫn, Google Android đang là đối thủ xứng tầm của iPhone của Apple. Google đang tích cực mở rộng	Hệ điều hành phân mảnh, không thống nhất trên các thiết bị, giới hạn về độ 'mở' và nhiều lỗ hổng bảo mật là những yếu điểm của Android OS.

	cộng đồng phát triển các ứng dụng cho Android. Bộ công cụ phát triển phần mềm (SDK) đầy đủ, hỗ trợ đa nền (Linux, Windows hay Mac OS) do chạy trên máy ảo Java. Thư viện ngày càng hoàn thiện, dễ dàng cho người lập trình.	
<b>Windows Phone</b>	Có thư viện API khá giống với API trên Win32, các công cụ hỗ trợ lập trình đầy đủ với Visual Studio, điều này làm cho những người phát triển trên Win32 không mất công tìm hiểu lại các API và các công cụ lập trình.	Sự có mặt của iPhone và Android là hai trở ngại lớn với Windows Mobile. Hai nền tảng này đang hoàn thiện và được người dùng rất ưa chuộng.
<b>iPhone</b>	<p><b>Màn hình cảm ứng đa điểm:</b> iPhone sử dụng hoàn toàn bằng cảm ứng và không sử dụng các nút. Với iPhone ta có thể điều khiển trên màn hình kể cả việc trượt của các ngón tay. Ta có thể phóng to ảnh bằng cách trượt hai ngón tay ra xa và thu nhỏ bằng cách ngược lại.</p> <p><b>Bộ cảm nhận gia tốc:</b> Những phản ứng nhanh chóng của bộ cảm nhận gia tốc thay đổi độ phân giải màn hình từ dọc sang ngang tự động khi ta đặt điện thoại nằm ngang. Điều này làm sinh động thêm cho các trò chơi. Âm thanh, hình ảnh hoàn hảo.</p>	Việc lập trình trên cho iPhone phải thực hiện trên hệ điều hành Mac, do đó không phải ai cũng có thể lập trình cho iPhone. Hơn thế, nếu muốn đưa chương trình ra máy thật người lập trình phải trả một khoản phí lập trình, điều này làm giảm tính cạnh tranh so với các đối thủ khác.



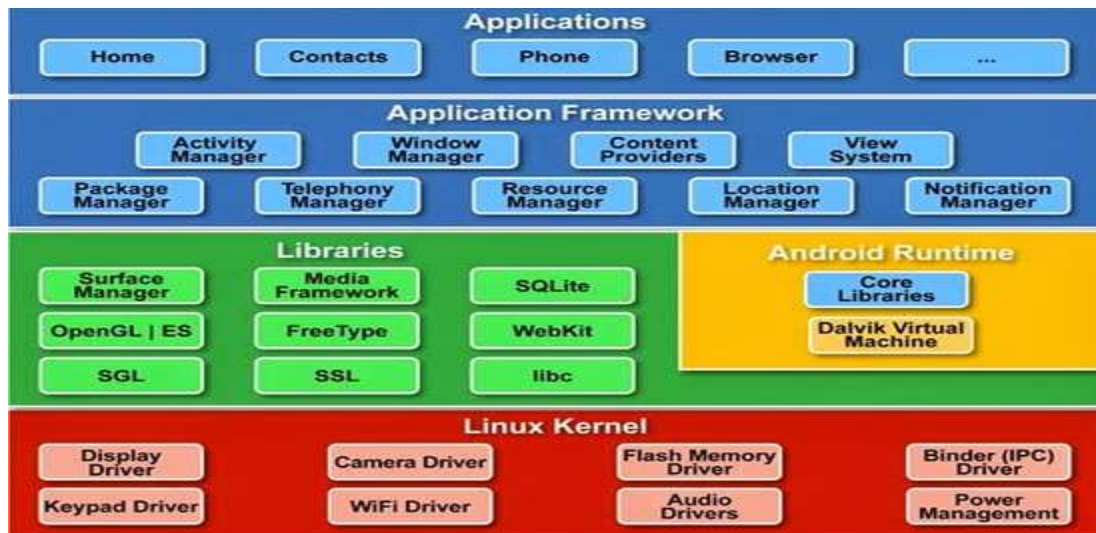
## 2.2 Kiến trúc và các thành phần

### 2.2.1 Kiến trúc tổng quát

Android bao gồm bốn thành phần sau:

- ✓ Hệ điều hành.
- ✓ Thư viện và các giao diện lập trình ứng dụng.
- ✓ Khung ứng dụng.
- ✓ Ứng dụng.

### 2.2.2 Hệ điều hành



Hình 2.2 Kiến trúc tổng quát

Android sử dụng nhân Linux 2.6 làm nhân cho các dịch vụ hệ thống như bảo mật, quản lý bộ nhớ, quản lý tiến trình (xử lý tiến trình, đa luồng), ngăn xếp mạng và trình điều khiển thiết bị (giao tiếp USB, giao tiếp hồng ngoại, không dây, v.v...). Nhân Linux này cũng có vai trò như một lớp trừu tượng giữa phần cứng và phần mềm.

Tuy được phát triển dựa vào nhân linux nhưng thực ra nhân linux đã được nâng cấp và sửa đổi rất nhiều để phù hợp với tính chất của những thiết bị

cầm tay như hạn chế về bộ vi xử lý, dung lượng bộ nhớ, kích thước màn hình, nhu cầu kết nối mạng không dây.

### 2.2.2.1 Các thành phần của nhân Linux:



*Hình 2.3 Linux kernel*

- ✓ Display Driver : Điều khiển việc hiển thị lên màn hình cũng như thu nhận những điều khiển của người dùng lên màn hình ( di chuyển, cảm ứng...).
- ✓ Camera Driver : điều khiển hoạt động của camera, nhận luồng dữ liệu từ camera về.
- ✓ Bluetooth Driver : Điều khiển thiết bị phát và thu sóng Bluetooth.
- ✓ USB Driver : Điều khiển bàn phím.
- ✓ Wifi Driver : Chịu trách nhiệm về việc thu phát sóng wifi.
- ✓ Audio Driver : Điều khiển các bộ thu phát âm thanh, giải mã các tín hiệu dạng audio thành tín hiệu số và ngược lại.
- ✓ Power Management : Giám sát việc tiêu thụ điện năng.
- ✓ M-system Driver : Quản lý việc đọc ghi... lên các thiết bị nhớ như thẻ SD, flash.
- ✓ Binder IPC Driver : Chịu trách nhiệm về việc kết nối và liên lạc với mạng vô tuyến như CDMA, GSM, 3G, 4G, E để đảm bảo những chức năng truyền thông được thực hiện.

### 2.2.2.2 Library



Hình 2.4 Libraries

Android cung cấp một số các APIs cho phát triển ứng dụng. Danh sách các API cơ bản sau được cung cấp bởi tất cả các thiết bị trên nền Android:

- ✓ `android.util`: Gói tiện ích cơ bản bao gồm nhiều lớp mức thấp như là các lớp quản lý (List, Stack...) lớp xử lý chuỗi, lớp xử lý XML.
- ✓ `android.graphics`: Cung cấp các lớp đồ họa mức thấp thực hiện các chức năng đồ họa, màu, vẽ cơ bản.
- ✓ `android.database`: Cung cấp các lớp mức thấp bắt buộc cho việc điều khiển cursor khi làm việc với các cơ sở dữ liệu.
- ✓ `android.content`: Các giao tiếp lập trình nội dung được dùng để quản lý truy cập dữ liệu và xuất bản bằng cách cung cấp các dịch vụ thao tác với tài nguyên, Content Provider, và các gói.
- ✓ `android.view`: View là lớp giao diện người dùng cơ bản nhất. Tất cả giao diện người dùng được tạo ra đều phải sử dụng một tập các View để cung cấp cho các thành phần tương tác người dùng.
- ✓ `android.widget`: Xây dựng dựa trên gói View. Những lớp widget những thành phần giao diện được tạo sẵn được sử dụng để tạo nên giao diện người dùng. Các widget bao gồm danh sách, nút bấm, hộp nhập, các kiểu trình bày (layout).
- ✓ `com.google.android.maps`: Bộ API mức cao cung cấp truy cập đến điều khiển bản đồ sẵn trong Android từ ứng dụng được xây dựng. Bao gồm cả

lớp `MapView` cũng như `Overlay` và `MapController` để tương tác với bản đồ bên trong ứng dụng.

- ✓ `android.provider`: Để tạo thuận lợi cho người phát triển truy cập đến các Content Provider tiêu chuẩn (như là dữ liệu danh bạ), gói cung cấp (Provider) bao gồm các lớp cho phép truy cập đến cơ sở dữ liệu chuẩn trong tất cả các bản phân phối Android.
- ✓ `android.telephony`: Các API điện đàm cung cấp khả năng tương tác trực tiếp với tầng điện thoại trong các thiết bị, cho phép tạo, nhận, theo dõi các cuộc gọi, tình trạng các cuộc gọi và tin nhắn SMS.
- ✓ `android.webkit`: Gói WebKit cung cấp các API để làm việc với các nội dung Web-based bao gồm một lớp `WebView` để tạo ra giao diện web, nhúng trong ứng dụng và một trình quản lý cookie.

Cùng với các API của Android, còn có một tập các thư viện C/C++ như:

- **OpenGL**: Thư viện dùng để tạo ra các đồ họa 3D dựa vào chuẩn OpenGL ES 1.0 API.
- **FreeType**: Hỗ trợ xử lý bitmap và font vector.
- **GGL**: Thư viện cơ bản, dùng để cung cấp các engine đồ họa 2D.
- **Libc**: Thư viện C chuẩn, được tối ưu cho các thiết bị Linux-based.
- **SQLite Engine**: Cơ sở dữ liệu quan hệ gọn nhẹ, dùng để lưu trữ dữ liệu của ứng dụng.
- **SSL**: Hỗ trợ sử dụng giao thức mã hóa Secure Sockets Layer trong bảo mật truyền thông Internet.

Ngoài các thư viện chuẩn của Android, để đáp ứng tiêu chí phù hợp với nhiều thiết bị khác nhau, Android còn có thể có các API phụ thuộc thiết bị như `android.location`, `android.media`, `android.opengl`, `android.hardware`, `android.bluetooth`, `android.net.wifi`, và `android.telephony`.

### 2.2.2.3 Android Runtime



*Hình 2.5 Android Runtime*

Phần này chứa các thư viện mà một chương trình viết bằng ngôn ngữ Java có thể hoạt động. Phần này có 2 bộ phận tương tự như mô hình chạy Java trên máy tính thường. Thứ nhất là các thư viện lõi (Core Library), chứa các lớp như JAVA IO, Collections, File Access. Thứ hai là một máy ảo java (Dalvik Virtual Machine).

Mặc dù cũng được viết từ ngôn ngữ Java nhưng một ứng dụng Java của hệ điều hành Android không được chạy bằng JRE (Java Runtime Environment) của Sun (nay là Oracle) (JVM) mà là chạy bằng máy ảo Dalvik (máy ảo dalvik được đặt tên sau khi một nhà phát triển phần mềm của Google tạo ra nó sau khi đi thăm một ngôi làng ở nơi tổ tiên ông sinh sống) do Google phát triển, máy ảo dalvik thực thi các tập tin dalvik (dex), khác với java là chuyển sang dạng bytecode. Định dạng này được tối ưu hóa cho bộ nhớ tối thiểu.

### 2.2.2.4 Application Framework



*Hình 2.6 Application Framework*

Kiến trúc của Android khuyến khích khái niệm thành phần sử dụng lại, cho phép công bố và chia sẻ các Activity, Service, dữ liệu, với các ứng dụng khác với quyền truy cập được quản lý bởi khai báo.

Cơ chế đó cho phép người lập trình tạo ra một trình quản lý danh bạ hoặc trình quay số điện thoại mà có các thành phần người khác có thể tạo mới giao diện và mở rộng chức năng thay vì tạo lại chúng.

Những dịch vụ sau là những dịch vụ kiến trúc cơ bản nhất của tất cả các ứng dụng, cung cấp một framework cho mọi mọi phần mềm được xây dựng:

- ✓ **Activity Manager:** Điều khiển vòng đời của các Activity bao gồm cả quản lý các tầng Activity.
- ✓ **Views:** Được sử dụng để tạo lập các giao diện người dùng cho các Activity.
- ✓ **Notification Manager:** Cung cấp một cơ chế cố định và quy củ cho việc gửi các thông báo đến người dùng.
- ✓ **Content Provider:** Cho phép ứng dụng chia sẻ dữ liệu giữa các ứng dụng.
- ✓ **Resource Manager :** Hỗ trợ các thành phần không thuộc mã nguồn như là chuỗi ký tự đồ họa được đặt bên ngoài.

### 2.2.3 Các thành phần của một ứng dụng Android

Một ứng dụng trên Android được cấu thành từ bốn thành phần cơ bản sau:

- ❖ Activities.
- ❖ Services.
- ❖ Broadcast Receivers.
- ❖ Content Provider.

Các thành phần này không nhất thiết phải có mặt đầy đủ trong ứng dụng. Chúng ta có thể xem các thành phần nào được sử dụng trong ứng dụng bằng việc xem khai báo trong file `AndroidManifest.xml`.

### 2.2.3.1 Activity

- Khái niệm

Một hoạt động là một giao diện người dùng trực quan mà người dùng có thể thực hiện trên đó mỗi khi được kích hoạt. Một ứng dụng có thể có nhiều hoạt động và chúng có thể gọi qua lại lẫn nhau. Mỗi Activity là một dẫn xuất của lớp `android.app.Activity`.

Mỗi hoạt động có một cửa sổ để vẽ lên. Thông thường cửa sổ này phủ đầy màn hình, ngoài ra nó cũng có thể có thêm các cửa sổ con khác như là hộp thoại...Nội dung của cửa sổ của hoạt động được cung cấp bởi một hệ thống cấp bậc các View (là đối tượng của lớp Views).

- Vòng đời hoạt động (circle activities)

Các hoạt động trong hệ thống được quản lý bởi một cấu trúc dữ liệu ngăn xếp. Khi có một hoạt động được khởi tạo, nó được đẩy vào trong ngăn xếp, chuyển sang trạng thái thực thi và hoạt động trước đó sẽ chuyển sang trạng thái chờ. Hoạt động này chỉ trở lại trạng thái kích hoạt khi mà hoạt động vừa khởi tạo kết thúc việc thực thi.

Một Activity có 3 trạng thái chính:

- ✓ Active hoặc running khi nó ở trên nhất màn hình và nhận tương tác người dùng.

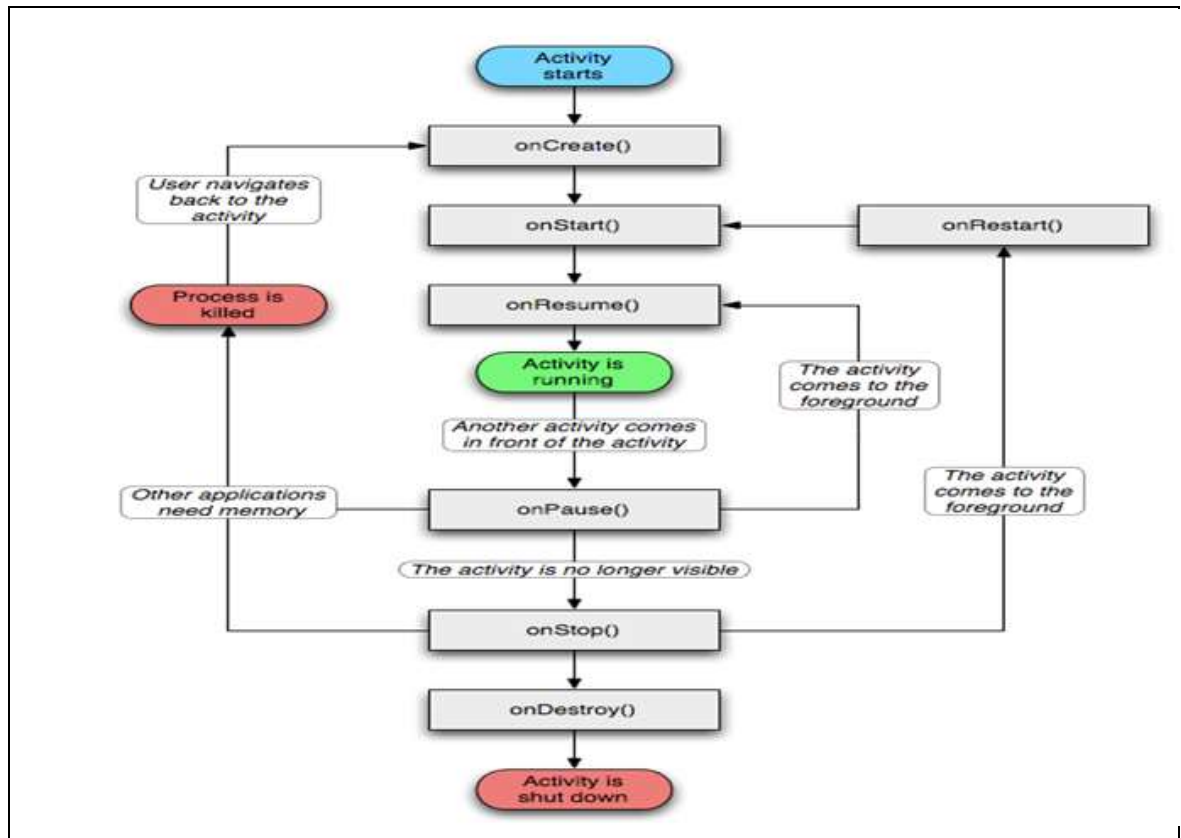
- ✓ Paused khi Activity không còn là trọng tâm trên màn hình nhưng vẫn hiện thị trước người dùng.
- ✓ Stopped khi một Activity hoàn toàn bị che khuất, nó sẽ rơi vào trạng thái Stopped. Tuy nhiên, nó vẫn còn lưu trữ toàn bộ thông tin trạng thái. Và nó thường bị hệ thống đóng lại khi có tình trạng thiếu bộ nhớ. Khi chuyển giữa các trạng thái, ứng dụng sẽ gọi các hàm callback ứng với các bước chuyển:

- void onCreate(Bundle savedInstanceState)
- void onStart()
- void onRestart()
- void onResume()
- void onPause()
- void onStop()
- void onDestroy()

Biểu đồ sau mô tả trạng thái trong vòng đời của một hoạt động. Hình chữ nhật thể hiện các phương thức Callback mà chúng ta có thể khai báo để gọi thực thi một số thao tác khi hoạt động chuyển sang trạng thái khác (phương thức Callback là phương thức được gọi lại bởi một phương thức khác khi có một sự kiện xảy ra). Các trạng thái chính của một hoạt động được thể hiện bởi các hình viên thuốc.

Vòng đời của một hoạt động có thể được thể hiện trong những quá trình sau:





Hình 2.7 Vòng đời của một activity

Toàn bộ thời gian sống của một hoạt động bắt đầu từ lời gọi đầu tiên tới phương thức `onCreate(Bundle)` tới lời gọi phương thức `onDestroy()`. Trong quá trình này, một hoạt động sẽ khởi tạo lại tất cả các tài nguyên cần sử dụng trong phương thức `onCreate()` và giải phóng chúng khi phương thức `onDestroy()` được thực thi.

Thời gian sống có thể nhìn thấy của một hoạt động bắt đầu từ lời gọi tới phương thức `onStart()`, cho tới khi phương thức `onStop()` của nó được thực thi. Toàn bộ các tài nguyên đang được sử dụng bởi hoạt động vẫn tiếp tục được lưu giữ, người dùng có thể thấy giao diện nhưng không tương tác được với hoạt động do trong quá trình này hoạt động không ở trạng thái chạy tiền cảnh.

Thời gian sống tiền cảnh của một hoạt động là quá trình bắt đầu từ khi có lời gọi tới phương thức `onResume()` và kết thúc bằng lời gọi tới phương thức

`onPause()`. Trong thời gian này, hoạt động chạy ở tiền cảnh và có thể tương tác với người dùng.

### 2.2.3.2 Service

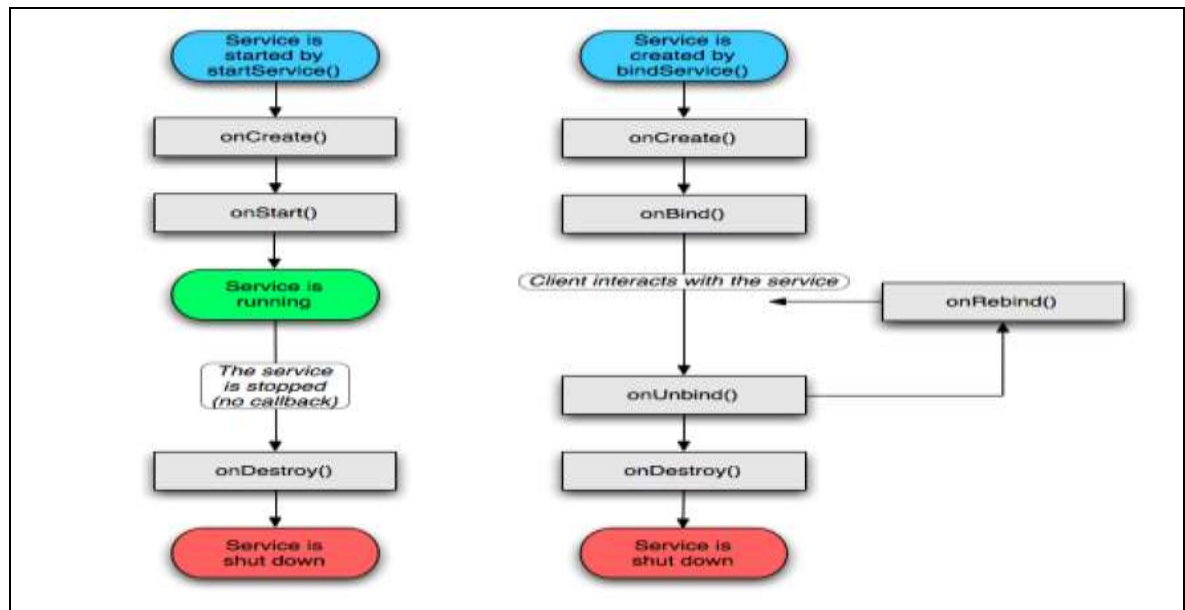
Một dịch vụ (Service) là các đoạn mã được thực thi ngầm bởi hệ thống mà người sử dụng không thấy được. Mỗi service đều được mở rộng từ lớp cơ sở là `Service` trong gói `android.app`. Có thể kết nối tới hoặc kích hoạt một Service thông qua interface mà Service đưa ra. Ví dụ như một chương trình chơi nhạc, sẽ có vài hoạt động cho phép người dùng duyệt danh sách các bài hát và lựa chọn bài nào để phát. Tuy nhiên, chức năng chơi nhạc không được thiết kế như một hoạt động bởi chúng ta sẽ muốn chuyển qua cửa sổ khác, như khi soạn tin nhắn thì bài nhạc vẫn tiếp tục được chơi. Trong trường hợp này, ứng dụng chơi nhạc sẽ khởi tạo một dịch vụ bằng cách sử dụng phương thức:

```
Context.startService().
```

Một ứng dụng có thể dễ dàng thực hiện liên kết tới một dịch vụ đang chạy (thậm chí khởi động nếu nó chưa thực thi) bằng phương thức `Context.bindService()`. Khi đó dịch vụ này sẽ cung cấp cho ứng dụng cơ chế để giao tiếp với chúng thông qua giao diện gọi là `IBinder` (đối với dịch vụ chơi nhạc có thể cho phép dừng hoặc chuyển qua bài nhạc kế tiếp).

- ✓ Vòng đời của một dịch vụ được hiểu là quá trình hoạt động từ khi nó được tạo ra cho tới khi bị loại khỏi hệ thống. Có hai cách thức để một dịch vụ có thể được chạy trong hệ thống.
- ✓ Khi hệ thống có lời gọi tới phương thức `Context.startService()`. Trong trường hợp này, dịch vụ sẽ được thực hiện liên tục cho tới khi hệ thống gọi phương thức `Context.stopService()`.
- ✓ Khi các ứng dụng gọi phương thức `Context.bindService()` để tạo kết nối với dịch vụ (dịch vụ sẽ được khởi tạo nếu tại thời điểm đó nó đang không hoạt động). Ứng dụng sẽ nhận được một đối tượng

IBinder do dịch vụ trả lại để có thể gọi các phương thức Callback phù hợp để truy cập tới các trạng thái của dịch vụ. Nếu do lời gọi `Context.bindService()` mà dịch vụ được khởi tạo thì nó sẽ được thực thi cho tới khi nào kết nối trên (tức là đối tượng IBinder) vẫn còn tồn tại.



Hình 2.8 Sơ đồ chuyển trạng thái của service

### 2.2.3.3 Bộ nhận quảng bá (Broadcast Receivers)

Bộ nhận quảng bá là một thành phần không làm gì cả nhưng nó nhận và phản hồi lại các thông báo quảng bá. Nhiều quảng bá có nguồn gốc từ mã hệ thống, ví dụ thông báo thay đổi múi giờ, pin yếu, ảnh đã chụp hay thay đổi ngôn ngữ. Các ứng dụng có thể khởi động quảng bá, ví dụ để các ứng dụng khác biết rằng dữ liệu đã được tải về xong trên thiết bị và sẵn sàng sử dụng.

Một ứng dụng có thể có bất kỳ số lượng bộ nhận quảng bá nào để nhận những thông báo quan trọng với nó. Tất cả các bộ nhận quảng bá được kế thừa từ lớp `BroadcastReceiver`.

Bộ nhận quảng bá không có giao diện. Tuy nhiên, chúng có thể khởi động một hoạt động để đáp lại thông tin mà nó nhận được, hay chúng có thể sử

dụng NotificationManager để thông báo người dùng biết. Các thông báo có thể được sự chú ý của người dùng theo các cách khác nhau như là sáng màn hình, rung thiết bị, bật âm thanh nào đấy... Thông thường, chúng đặt thông báo trên thanh trạng thái, nơi người dùng có thể nhận được thông báo.

#### 2.2.3.4 Content Provider

Các ứng dụng có thể lưu trữ dữ liệu của mình trong các tập tin hoặc sử dụng cơ sở dữ liệu SQLite sẵn có v.v... Content Provider có chức năng cung cấp một tập hợp các phương thức cho phép một ứng dụng có thể lưu trữ và lấy dữ liệu được quản lý bởi content provider đó.

Content Provider là một đặc trưng riêng của Android, nhờ đó mà các ứng dụng có thể chia sẻ dữ liệu với nhau một cách dễ dàng.

#### 2.2.3.5 Các Intent

Content Provider được kích hoạt khi chúng được gọi từ một ContentResolver. Ba thành phần khác (hoạt động, dịch vụ và bộ nhận quảng bá) được kích hoạt bởi thông điệp không đồng bộ từ các Intent. Một Intent là một đối tượng có kiểu Intent chứa nội dung của thông điệp. Với các hoạt động và dịch vụ, nó gọi tên hành động được yêu cầu và xác định URI của dữ liệu tác động tới ở giữa. Ví dụ, nó có thể truyền tải một yêu cầu cho một hoạt động hiển thị một ảnh cho người dùng hay cho phép người dùng sửa văn bản. Với bộ nhận quảng bá, đối tượng Intent gọi tên của hành động được thông báo. Ví dụ, bộ nhận quảng bá có thể thông báo các phần nó quan tâm là nút chụp ảnh đã được bấm.

Có vài phương thức cho việc kích hoạt mỗi thành phần:

Một hoạt động được khởi chạy thông qua một đối tượng Intent `Context.startActivity()` hay `Activity.startActivityForResult()`. Hoạt động đáp lại có thể theo dõi Intent được tạo ra đó bằng phương thức

`getIntent()` và cập nhật thông qua phương thức `setIntent(Intent)`. Android gọi phương thức `onNewIntent()` để bỏ qua các Intent đến trước nó.

Một hoạt động thường bắt đầu hoạt động khác. Nếu nó muốn trả lại kết quả hoạt động nó đã khởi chạy, nó sẽ gọi phương thức:

`startActivityForResult()` thay cho phương thức `startActivity()`. Ví dụ, nếu nó khởi chạy một hoạt động mà cho phép người dùng lấy một ảnh, nó có thể muốn lấy kết quả của ảnh được chọn. Kết quả được trả về trong một đối tượng Intent thông qua phương thức `onActivityResult()`.

Một dịch vụ được bắt đầu thông qua một đối tượng Intent là `Context.startService()`. Android gọi phương thức `onStart()` của dịch vụ và thông qua đối tượng Intent của nó.

Tương tự, một Intent có thể thông qua `Context.bindService()` để thiết lập một kết nối liên tục giữa các thành phần và dịch vụ đích. Dịch vụ nhận đối tượng Intent qua lời gọi `onBind()` (nếu dịch vụ chưa được chạy, `bindService()` có thể chọn bắt đầu nó). Cho ví dụ, một hoạt động có thể thiết lập kết nối với dịch vụ chơi nhạc để cập ở phần trước để nó có thể cung cấp cho người dùng giao diện sử dụng để điều khiển chơi lại. Hoạt động sẽ gọi `bindService` để thiết lập kết nối và sau đó gọi phương thức đã định nghĩa bởi dịch vụ để áp dụng chơi lại ca khúc.

Một ứng dụng có thể khởi tạo một quảng bá thông qua đối tượng Intent bằng phương thức như:

`Context.setBroadcast()`, `Context.setOrderedBroadcast()` và `Context.sendStickyBroadcast()`. Android chuyển những Intent tới tất cả các bộ nhận quảng bá nào quan tâm bằng việc gọi phương thức `onReceive()` của nó.

### 2.2.3.6 Tập tin khai báo (Manifest File)

Trước khi có thể khởi chạy một ứng dụng thành phần, nó phải xem ứng dụng bao gồm những thành phần nào. Thêm nữa, các ứng dụng khai báo các thành phần của nó trong một tập tin khai báo để đóng gói lại vào trong gói Android (tập tin .apk chứa các mã nguồn, tập tin và tài nguyên).

Tập tin này có cấu trúc của tập tin XML và luôn có tên là `AndroidManifest.xml` trong mọi ứng dụng. Tập tin này thực hiện một số chức năng như thêm và khai báo các thành phần của ứng dụng, tên các thư viện ứng dụng cần liên kết tới (ngoài thư viện chuẩn của Android) và xác định các quyền cho ứng dụng.

Tuy nhiên, nhiệm vụ chính của tập tin khai báo là khai báo các thành phần của ứng dụng. Một hoạt động có thể được khai báo như sau:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest . . . >
<application . . . >
<activity android:name="com.example.project.FreneticActivity"
android:icon="@drawable/small_pic.png"
        android:label="@string/freneticLabel"
        . . . >
</activity>
. . .
</application>
</manifest>
```

Thuộc tính “name” của phần tử `<activity>` là tên các lớp con lớp Activity đã được cài đặt, thuộc tính “icon” và “label” trỏ đến tập tin tài nguyên chứa biểu tượng và nhãn được hiển thị cho người dùng.

Các thành phần khác được khai báo theo cách tương tự: `<service>` dịch vụ, `<receiver>` bộ nhận quảng bá và `<provider>` content provider. Các hoạt

động, dịch vụ và content provider có thể cùng được khai báo trong tập tin khai báo hoặc có thể được tạo tự động trong mã (như đối tượng BroadcastReceiver) và được đăng ký với hệ thống bằng cách gọi `Context.registerReceiver()`.

### 2.2.3.7 Bộ lọc Intent

Một đối tượng Intent có thể có tên rõ ràng trong thành phần đích. Nếu có, Android sẽ tìm thành phần đó (dựa trên khai báo trong tập tin khai báo) và kích hoạt nó. Nhưng nếu đích có tên không rõ ràng, Android phải xác định thành phần nào thích hợp nhất để đáp lại intent. Nó thực hiện so sánh đối tượng Intent với bộ lọc intent trong các đích có khả năng. Một thành phần của bộ lọc intent cho Android biết loại intent thành phần nào cần xử lý. Giống các thông tin cơ bản, chúng được khai trong tập tin khai báo. Đây là một ví dụ để thêm hai bộ lọc intent cho một hoạt động.

```

3   <intent-filter . . . >
4   <action android:name=
5           "android.intent.action.MAIN" />
6   <category android:name=
7           "android.intent.category.LAUNCHER" />
8   </intent-filter>
9   <intent-filter . . . >
10  <action android:name=
11          "com.example.project.BOUNCE" />
12  <data android:mimeType="image/jpeg" />
13  <category android:name=
14          "android.intent.category.DEFAULT" />
15  </intent-filter>

```

Bộ lọc đầu tiên – là sự kết hợp của hành động “`android.intent.action.Main`” và danh mục:

“`android.intent.category.LAUNCHER`” – là một ví dụ phổ biến. Nó đánh dấu lại một Intent và mô tả lại cho bộ khởi chạy ứng dụng, đặt vào trong danh mục các ứng dụng trên điện thoại. Nói theo cách khác, hoạt động làm lối vào cho ứng dụng sẽ được khởi chạy, người dùng sẽ thấy khi chọn khởi chạy ứng dụng.

Bộ lọc thứ hai được khai báo để hoạt động có thể thực thi trên một kiểu dữ liệu đặc biệt.

Một thành phần có thể có bao nhiêu số bộ lọc intent tùy thích, mỗi một cái lại được khai báo khác nhau cho các khả năng. Nếu nó không có một bộ lọc nào, nó có thể được kích hoạt bởi các intent khác gọi đến.

Để tạo và đăng ký một bộ nhận quảng bá trong mã nguồn, bộ lọc intent tạo bằng một đối tượng giống đối tượng `IntentFilter`. Tất cả các bộ lọc khác đều được cài đặt trong tập tin khai báo.

## 2.3 Công nghệ web

Hiện nay, web đã trở thành một sản phẩm thông tin phổ biến trong cộng đồng. Nhu cầu về việc tạo ra các trang web của tất cả mọi người là rất lớn. Những trang web hiện nay ngày càng đa dạng, về hình thức, công nghệ lẫn cách thức giao tiếp với người dùng. Sự khác nhau đó được quyết định bởi công nghệ sử dụng và nền tảng ngôn ngữ lập trình tạo ra nó. Việc lựa chọn công nghệ để phát triển là một vấn đề quan trọng, phụ thuộc vào nhiều yếu tố chẳng hạn quy mô, tốc độ xử lý, yêu cầu bảo mật ... 3 công nghệ được sử dụng phổ biến ở Việt Nam là .NET, PHP và Java.

### 2.3.1 Ngôn ngữ lập trình PHP



PHP (PHP: Hypertext Preprocessor) là một công nghệ đang nhận được sự quan tâm rất lớn từ thế giới lập trình. Do được tối ưu hóa cho các ứng dụng web, tốc độ nhanh, nhỏ gọn, cú pháp giống C và Java, dễ thực hiện và thời gian xây dựng sản phẩm tương đối ngắn hơn so với các ngôn ngữ khác nên PHP đã nhanh chóng trở thành một ngôn ngữ lập trình web phổ biến nhất thế giới.

Khi sử dụng PHP, người dùng sẽ có được tốc độ nhanh hơn nhiều so với các ngôn ngữ kịch bản khác, bởi PHP là phần mềm mã nguồn mở, được hỗ trợ bởi nhiều lập trình viên giỏi, có số lượng người dùng đông đảo. Ngoài ra, PHP chạy được trên nhiều hệ điều hành khác nhau.

Rút ngắn thời gian phát triển: PHP cho phép tách phần HTML code và phần script, do đó có thể độc lập giữa công việc phát triển mã và thiết kế. Điều này vừa giúp lập trình viên dễ dàng hơn vừa có thể làm cho chương trình mềm dẻo hơn trong việc thay đổi giao diện.

PHP không chỉ là phần mềm mã nguồn mở, mà còn thực sự miễn phí kể cả sử dụng cho mục đích thương mại. Do là phần mềm mã nguồn mở, các lỗi của PHP được công khai và nhanh chóng được sửa chữa bởi nhiều chuyên gia.

Tốc độ: nhờ vào sức mạnh của Zend Engine, khi so sánh PHP với ASP có thể thấy PHP vượt trội hơn ở một số test, vượt trội hơn tốc độ biên dịch.

Tính khả chuyên: PHP được thiết kế để chạy trên nhiều nền tảng khác nhau, có thể làm việc với nhiều phần mềm máy chủ, cơ sở dữ liệu. PHP rất ổn định và tương hợp, PHP đã vận hành khá ổn định trên các hệ điều hành kể cả Unix, Windows...Đồng thời nó cũng nối với một số máy chủ như IIS hay Apache.

*Bảng 2.2 Bảng So Sánh Các Công Nghệ Lập Trình Web*

PHP	ASP.NET	JSP
Tốc độ xử lý nhanh, hiệu quả	Tốc độ xử lý nhanh, hiệu quả	Hơi chậm

cao	cao	
Chi phí giá thành thấp (ngôn ngữ free không tốn chi phí mua bản quyền)	Chi phí giá thành cao (do một phần phải mua bản quyền)	Chi phí giá thành cao
Thời gian code và triển khai đơn giản	Thời gian code và triển khai hơi phức tạp, chậm hơn PHP	Thời gian code và triển khai hơi phức tạp, chậm hơn PHP
Số lượng nhà cung cấp hosting nhiều, dễ lựa chọn	Số lượng nhà cung cấp hosting không nhiều, vì vậy khó cho việc lựa chọn	Số lượng nhà cung cấp hosting ít, khó tìm
Số nhà cung cấp website nhiều, khả năng chọn lựa một trang web phù hợp tốt hơn	Số nhà cung cấp website nhiều, khả năng chọn lựa một trang web phù hợp tốt hơn	Số nhà cung cấp website ít, khả năng chọn lựa để xây dựng một trang web phù hợp rất khó
Khả năng mở rộng và phát triển dễ dàng và nhanh chóng	Mở rộng và phát triển website dễ dàng	Khả năng mở rộng và phát triển website khó khăn
Các công cụ và công nghệ hỗ trợ phong phú, đa dạng. Phát triển web trên nền web 2.0. Sử dụng công nghệ Ajax làm cho quá trình duyệt web nhanh	Các công cụ và công nghệ hỗ trợ phong phú, đa dạng. Phát triển web trên nền web 2.0. Sử dụng công nghệ Ajax làm cho quá trình duyệt web nhanh chóng và	Công cụ và công nghệ hỗ trợ ít, khó tìm.

chóng và thân thiện hơn đối với người dùng.	thân thiện hơn đối với người dùng.	
---	------------------------------------	--

### 2.3.2 Hệ quản trị cơ sở dữ liệu

Hệ quản trị cơ sở dữ liệu là phần mềm hay hệ thống được thiết kế để quản trị một cơ sở dữ liệu (CSDL). Cụ thể, các chương trình thuộc loại này hỗ trợ khả năng lưu trữ, sửa chữa, xóa và tìm kiếm thông tin trong một cơ sở dữ liệu. Có rất nhiều loại hệ quản trị CSDL khác nhau: từ phần mềm nhỏ chạy trên máy tính cá nhân cho đến những hệ quản trị phức tạp chạy trên một hoặc nhiều siêu máy tính.

Tuy nhiên, đa số hệ quản trị CSDL trên thị trường đều có một đặc điểm chung là sử dụng ngôn ngữ truy vấn theo cấu trúc. Các hệ quản trị CSDL phổ biến được nhiều người biết đến là MySQL, Oracle, PostgreSQL, SQL Server ... Phần lớn các hệ quản trị CSDL kể trên hoạt động tốt trên nhiều hệ điều hành khác nhau như Linux, Unix và MacOS. Ngoại trừ SQL Server của Microsoft chỉ chạy trên hệ điều hành Windows.

MySQL đã trở thành CSDL mã nguồn mở phổ biến nhất trên thế giới vì cơ chế xử lý nhanh và ổn định của nó, sự đáng tin cậy cao và dễ sử dụng. Nó được sử dụng mọi nơi – ngay cả châu Nam Cực - bởi các nhà phát triển Web riêng lẻ cũng như rất nhiều các tổ chức lớn nhất trên thế giới để tiết kiệm thời gian và tiền bạc cho những Website có dung lượng lớn, phần mềm đóng gói – bao gồm cả những nhà đi đầu trong lĩnh vực công nghệ như Yahoo!, Google, Nokia, YouTube...

#### **Ưu điểm:**

- Mã nguồn mở, miễn phí.

- Tốc độ truy cập nhanh, ổn định, mức độ bảo mật cao, dễ sử dụng.
- Hoạt động trên mọi hệ điều hành.
- Dễ quản lý.
- Cộng đồng hỗ trợ rộng lớn.

**Nhược điểm:**

- Bảo mật chưa bằng SQL Server hay Oracle.
- Chưa hỗ trợ XML.

*Bảng 2.3 Bảng So Sánh Các Hệ Quản Trị Cơ Sở Dữ Liệu*

STT	Các đặc điểm	SQL Server	Oracle	MySQL	PostgreSQL
1	Giấy phép sử dụng	Độc quyền	Độc quyền	Miễn phí	BSD
2	Hệ điều hành được hỗ trợ	Windows	Hạn chế BSD	Mọi hệ điều hành	Mọi hệ điều hành
3	Kích thước tối đa cơ sở dữ liệu	524258TB	Không giới hạn	Tùy vào hệ điều hành	Không giới hạn
4	Tốc độ truy xuất	Đòi hỏi tài nguyên lớn (CPU mạnh, RAM nhiều)	Tốc độ truy xuất nhanh	Tốc độ truy xuất nhanh	Tốc độ truy xuất nhanh
5	Tính bảo mật	Bảo mật hơn MySQL ở column level	Bảo mật cao, an toàn	Khả năng bảo mật tốt	Chưa tốt, còn lỗ hổng
6	Tính phổ biến	Hệ quản trị CSDL của Microsoft	Sử dụng cho các doanh nghiệp	Thường sử dụng để xây dựng	Thích hợp hệ thống GIS – viễn thám

			ngành lớn	dựng website	
7	Độ phức tạp	Thấp, dễ quản trị	Cao, khó quản trị	Thấp, dễ quản trị	Thấp, dễ sử dụng

(nguồn tài liệu tham khảo website)

### 2.3.3 Công nghệ web liên quan khác

#### ❖ HTML

HTML (HyperText Markup Language) là ngôn ngữ biểu diễn văn bản cho phép ta đưa vào một văn bản nhiều thuộc tính cần thiết để có thể truyền thông quảng bá trên mạng toàn cục WWW. HTML cho phép ta đưa hình ảnh đồ họa vào văn bản, thay đổi cách bày trí của văn bản, và tạo những tài liệu siêu văn bản có khả năng đối thoại tương tác với người dùng.

HTML là một ngôn ngữ tương tự như các ngôn ngữ lập trình nhưng đơn giản hơn. nó có cú pháp chặt chẽ để viết các lệnh thực hiện việc trình diễn văn bản. các từ khoá có ý nghĩa xác định được cộng đồng internet thừa nhận và sử dụng.

HTML độc lập với phần cứng và phần mềm. một tài liệu HTML được viết bằng một phần mềm soạn thảo cụ thể bất kỳ, trên một máy cụ thể nào đó đều có thể đọc được trên bất kì một hệ thống tương thích nào.

#### ❖ CSS

Khi mà công nghệ web ngày càng phát triển người ta nhận thấy việc dàn trang bằng các bảng ngày càng trở nên bất tiện. Khối lượng mã HTML lớn cùng việc khó khăn trong kiểm soát các vùng nội dung trên trang web là những lý do khiến HTML ngày càng thất thế. Nếu bạn vào các website chuyên nghiệp của nước ngoài bạn sẽ thấy việc dàn trang của họ hoàn toàn bằng các thẻ div của CSS mà không dùng các thẻ HTML, nếu có thì cũng rất ít. Công nghệ CSS được

đề cập đầu tiên bởi W3C vào năm 1996. Theo định nghĩa của W3C (Wide Web Consortium) CSS (từ viết tắt của Cascading Style Sheet tạm dịch là bảng kiểu xếp chồng) là một ngôn ngữ giúp người thiết kế web có thể thêm kiểu vào tài liệu web (như kiểu font chữ, màu sắc, khoảng cách...). Bạn có thể tạo ra kiểu một lần nhưng có thể dùng lại nhiều lần trong các tài liệu web tiếp theo.

Ưu điểm của các thẻ div là rất linh hoạt, các lớp có thể chồng lên nhau một phần hoạt nhiều phần. Đây là một điểm thú vị mà việc dàn trang bằng HTML không có được. Nhưng nhược điểm của CSS là việc kiểm soát chúng trên các trình duyệt web khác nhau khá khó khăn vì chính tính linh hoạt trên. Các trình duyệt hỗ trợ CSS đôi khi là khác nhau vì vậy nếu dàn trang không tốt thì việc hiển thị không tốt.

#### ❖ Ajax

AJAX là viết tắt của Asynchronous JavaScript And XML. Như trong tên gọi của nó, Ajax chẳng qua là sự kết hợp giữa Javascript và XML trong một thiết kế web.

Khi tìm hiểu AJAX, chúng ta thường nhầm tưởng AJAX là một ngôn ngữ lập trình mới. Thực tế, AJAX là một kỹ thuật kết hợp giữa Javascript và XML để đem lại cho người dùng một website nhanh hơn, tốt hơn và có tính tương tác cao hơn. Kỹ thuật Ajax đã xuất hiện từ lâu nhưng thuật ngữ Ajax mới được thống nhất và phổ biến rộng rãi trong giới công nghệ phần mềm từ năm 2006.

Điểm khác biệt là khi sử dụng Ajax, Javascript có thể tương tác trực tiếp với Server bằng cách sử dụng đối tượng XMLHttpRequest. Nhờ đó, trang web của bạn có thể tương tác trực tiếp với Server và nhận kết quả trả về từ Server mà không cần phải nạp lại toàn bộ trang web. So với cách truyền nhận dữ liệu thông thường, khi người dùng vào một link website, toàn bộ web sẽ được tải về trình

duyet thì khi sử dụng Ajax, chỉ một lượng rất nhỏ thông tin (đa phần ở dạng text) được truyền nhận và chính vì thế tốc độ của trang web tăng lên đáng kể. Thiết kế web với Ajax chính vì thế có thể cải thiện tốc độ truy cập web.

#### ❖ JQuery

Bất kì trang web nào cho dù sử dụng công nghệ gì (.NET, PHP, JSP...) đều phải sử dụng Javascript. jQuery là một thư viện JavaScript mạnh mẽ và ngắn gọn, đơn giản hóa việc truy xuất tài liệu HTML, xử lý sự kiện, các hiệu ứng và sự tương tác Ajax để phát triển web nhanh chóng. jQuery được thiết kế để thay đổi cách bạn viết JavaScript. jQuery thêm tương tác Ajax vào trong trang web của bạn.

jQuery đơn giản hóa cách viết Javascript và tăng tốc độ xử lý các sự kiện trên trang web, giúp tiết kiệm thời gian và công sức rất nhiều so với việc ngồi viết javascript theo cách thông thường. Đúng như khẩu hiệu của nó viết ít hơn, làm nhiều hơn, nếu không dùng jquery thì một chuyên gia cũng phải viết mất vài trang giấy mới làm được một nút “quay lại” với hiệu ứng trượt. Bên cạnh đó, việc sử dụng jquery giúp giải quyết tốt vấn đề xung đột giữa các trình duyệt web.

#### 2.3.4 Yii framework

##### ❖ Tổng quan

Yii là một framework PHP mã nguồn mở, dựa vào thành phần (component-based), hướng đối tượng, nó cho phép tái sử dụng tối đa trong lập trình web và đặc biệt có hiệu suất cao. Bởi vì Yii có trọng lượng nhẹ và được trang bị cơ chế bộ nhớ đệm tinh vi nên framework này đặc biệt phù hợp với các ứng dụng có lưu lượng cao, chẳng hạn như các cổng thông tin, diễn đàn, các hệ thống quản lý nội dung (CMS), hệ thống thương mại điện tử, ...

Giống như các framework web khác, Yii tuân theo mô hình MVC. Yii không phải là sản phẩm được phát triển từ các sản phẩm bên thứ ba, nó là kết

quả của các nhà lập trình web giàu kinh nghiệm và sự đầu tư của họ cho framework này. Yii được thiết kế cẩn thận từ đầu để đáp ứng việc phát triển các ứng dụng web một cách nghiêm túc, giúp tạo ra các sản phẩm web một cách nhanh chóng, bảo mật và chuyên nghiệp. Yii được các lập trình viên bình chọn là framework PHP tốt nhất cho đến thời điểm hiện tại<sup>[11]</sup>.

Bởi những tính năng nổi bật của Yii nên đề tài này sẽ được thực hiện trên nền Yii framework.

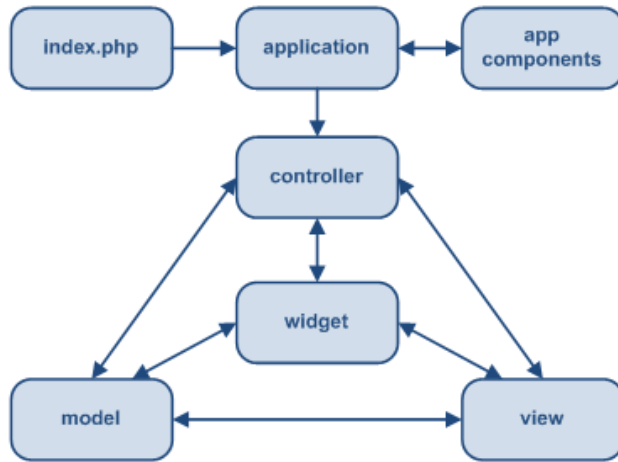
#### ❖ Các thành phần cơ bản

##### Mô hình MVC trong Yii

Cũng như các framework khác, Yii thực hiện mẫu thiết kế MVC (Model – View – Controller). MVC nhằm mục đích tách biệt tầng logic và tầng giao diện, giúp cho lập trình viên có thể dễ dàng thay đổi mỗi phần mà không làm ảnh hưởng đến phần còn lại. Trong mô hình MVC, Model đại diện cho dữ liệu và các quy tắc logic, View chứa các phần tử của giao diện người dùng (chẳng hạn như văn bản, biểu mẫu, ...), Controller quản lý giao tiếp giữa Model và View.

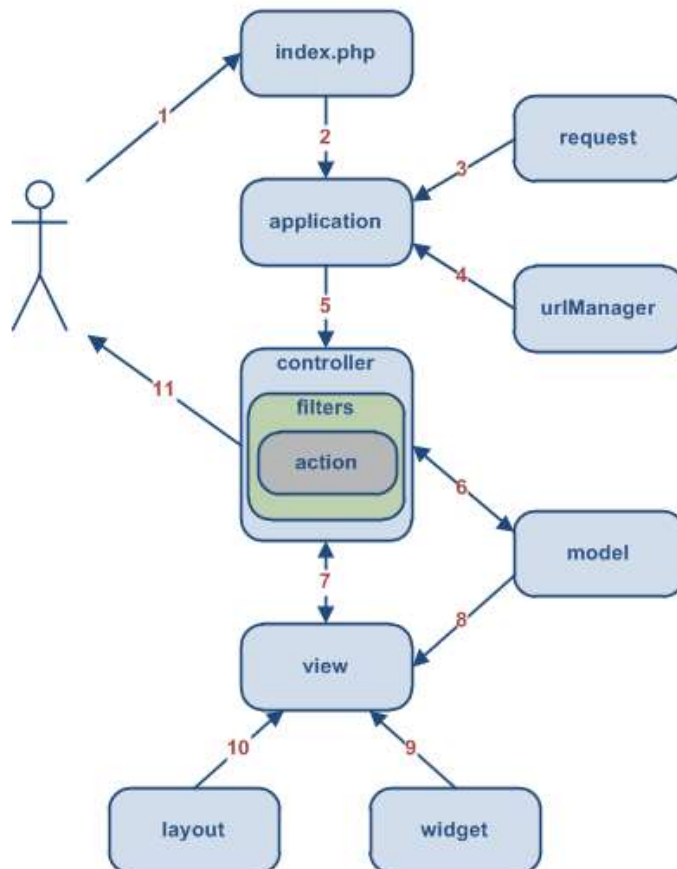
Bên cạnh thực hiện mô hình MVC, Yii cũng giới thiệu một front – controller, gọi là Application, controller này bao đóng ngữ cảnh thực thi cho quá trình xử lý yêu cầu. Application thu thập một số thông tin về yêu cầu người dùng, sau đó gửi nó đến một controller thích hợp để xử lý thêm.





Hình 2.9 Cấu trúc tĩnh của ứng dụng Yii

Hình sau đây thể hiện một luồng làm việc điển hình của một ứng dụng Yii khi nó xử lý yêu cầu người dùng:



*Hình 2.10 Luồng làm việc điển hình của một ứng dụng Yii.*

Yii xử lý yêu cầu người dùng thông qua các bước sau:

**Bước 1:** Một người dùng gửi một yêu cầu với URL `http://www.example.com/index.php?r=post/show&id=1` và máy chủ Web xử lý yêu cầu bằng cách thực thi đoạn mã khởi động `index.php`.

**Bước 2:** Đoạn mã khởi động này tạo một đối tượng Application và chạy nó.

**Bước 3:** Đối tượng Application xác định chi tiết thông tin yêu cầu người dùng từ một component gọi là request.

**Bước 4:** Đối tượng Application xác định controller và action được yêu cầu nhờ vào sự trợ giúp của một component gọi là `urlManager`. Ví dụ, controller là `post`, trở tới lớp `PostController`, và action là `show`, ý nghĩa thực sự của action này được xác định bởi controller.

**Bước 5:** Đối tượng Application tạo một thực thể của controller được yêu cầu để xử lý yêu cầu của người dùng. Controller này xác định rằng action `show` liên quan đến phương thức gọi là `actionShow` trong lớp controller. Sau đó, nó tạo và thực thi bộ lọc (ví dụ: điều khiển truy cập, ...) liên quan tới action này. Action được thực thi nếu nó được cho phép bởi bộ lọc này.

**Bước 6:** Action đọc một Post model có ID là 1 trong cơ sở dữ liệu

**Bước 7:** Action xuất ra một view gọi là `show` kèm với Post model.

**Bước 8:** View đọc và hiển thị các thuộc tính của Post model.

**Bước 9:** View thực thi một số widgets.

**Bước 10:** Kết quả mà View render được nhúng trong một layout.

**Bước 11:** Action hoàn thành việc render view và hiển thị kết quả cho người dùng.

**Đoạn mã khởi động**

Đoạn mã khởi động được viết bằng PHP để xử lý ban đầu yêu cầu của người dùng. Nó là đoạn mã PHP duy nhất mà người dùng được phép yêu cầu trực tiếp để thực thi.

Trong nhiều trường hợp, đoạn mã khởi động này được viết đơn giản như sau:

```
// remove the following line when in production mode
defined('YII_DEBUG') or define ('YII_DEBUG',true);
// include Yii bootstrap file
require_once('path/to/yii/framework/yii.php');
// create application instance and run
$configFile='path/to/config/file.php';
Yii::createWebApplication($configFile)->run();
```

Đầu tiên, nó bao gộp file yii.php của framework Yii. Sau đó, nó tạo một đối tượng Application với các thông số cấu hình nhất định và chạy đối tượng này.

### **Component**

Các ứng dụng Yii được xây dựng trên các component bởi vì tất cả các lớp trong framework này đều kế thừa từ lớp CComponent. Các component được sử dụng chủ yếu để truy cập các thuộc tính và kích hoạt/xử lý các sự kiện. Component của ứng dụng là một cách tiện lợi để chia sẻ thông tin giữa các controller của ứng dụng. Hướng tiếp cận này khá hữu ích và tiết kiệm được tài nguyên của hệ thống bởi vì chúng ta có thể sử dụng chỉ với một câu lệnh truy vấn.

### **Module**

Một module là một đơn vị phần mềm bao gồm các Model, View, Controller và các component hỗ trợ khác. Nhìn nhiều khía cạnh, một module

giống như là một ứng dụng. Sự khác nhau chính là một module không thể tự vận hành độc lập và nó phải nằm trong một ứng dụng. Người dùng có thể truy cập các controller trong một module giống như truy cập tới các controller của ứng dụng thông thường.

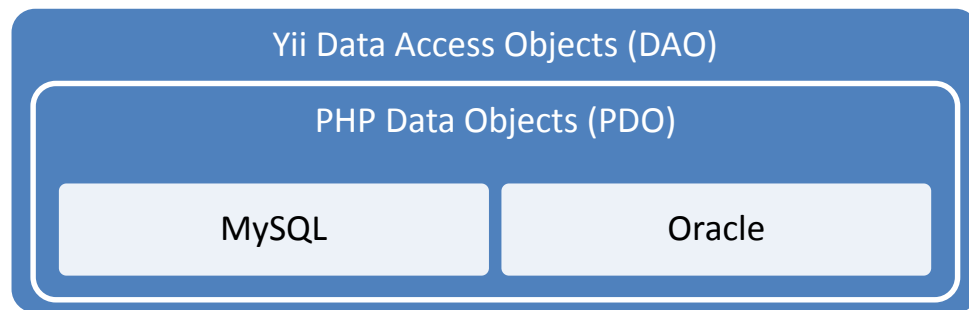
Các module thông thường khá hữu ích trong một số ngữ cảnh. Đối với ứng dụng kích cỡ lớn, chúng ta có thể chia nó thành một vài module, mỗi module được phát triển và bảo trì một cách riêng biệt. Một vài tính năng thường sử dụng, chẳng hạn quản lý người dùng, quản lý nhận xét, có thể được phát triển dưới dạng các module để chúng có thể được sử dụng lại một cách dễ dàng trong các dự án tương lai.

### **Làm việc với database**

Yii cung cấp những hỗ trợ hiệu quả để lập trình cơ sở dữ liệu. Được xây dựng bên trên phần mở rộng PHP Data Objects (PDO), Yii Data Access Objects cho phép truy xuất vào những hệ quản trị cơ sở dữ liệu khác nhau trong một interface đồng nhất. Các ứng dụng được phát triển sử dụng Yii DAO có thể được chuyển đổi một cách dễ dàng giữa các hệ quản trị cơ sở dữ liệu khác nhau mà không cần phải chỉnh sửa dữ liệu.

Yii Query Builder cung cấp một phương thức hướng đối tượng để xây dựng các câu truy vấn SQL, giúp giảm các nguy cơ tấn công kiểu SQL injection.

Yii Active Record (AR) được xây dựng theo hướng tiếp cận phổ biến Object – Relational Mapping, giúp đơn giản việc lập trình cơ sở dữ liệu. Mô hình này thể hiện một bảng dữ liệu như là một lớp và các bản ghi như là một đối tượng. Yii AR loại bỏ các công việc lặp đi lặp lại khi viết những câu lệnh SQL của các thao tác cơ bản tạo mới, đọc, cập nhật và xóa – CRUD (create, read, update, delete).



*Hình 2.11 Yii Data Access Objects*

Mặc dù các tính năng này có thể xử lý hầu hết tất cả các công việc liên quan đến cơ sở dữ liệu nhưng bạn vẫn có thể sử dụng các thư viện cơ sở dữ liệu khác trong ứng dụng Yii. Yii framework được thiết kế với khả năng được sử dụng cùng với các bộ thư viện của bên thứ ba.

### **Mở rộng Yii**

Mở rộng Yii là một hoạt động thông thường trong quá trình phát triển phần mềm. Ví dụ, khi viết một controller mới, bạn mở rộng Yii bằng cách thừa kế lớp CController, khi bạn viết một widget mới, bạn đang mở rộng CWidget hoặc một lớp Widget có sẵn. Nếu các đoạn mã mở rộng được thiết kế để được sử dụng lại bởi các lập trình viên bên thứ ba thì chúng ta có thể gọi nó là extension.

Một extension thông thường đáp ứng một mục đích nào đó. Theo thuật ngữ trong Yii, extension có thể được chia thành những loại sau đây:

- ✓ Application component
- ✓ Behaviour
- ✓ Widget
- ✓ Controller
- ✓ Action
- ✓ Filter

- ✓ Console command
- ✓ Validator: một validator là một lớp thành phần kế thừa từ lớp CValidator.
- ✓ Helper: một helper là một lớp chỉ có các phương thức tĩnh. Nó giống như các hàm toàn cục sử dụng các tên lớp như là namespace của chúng.
- ✓ Module.
- ✓ Một extension cũng có thể là một thành phần mà không thuộc bất kỳ loại nào ở trên. Yii được thiết kế sao cho tất cả các extension đó có thể mở rộng và tùy chỉnh để đáp ứng những yêu cầu riêng biệt.

## 2.3.5 Web Service

### 2.3.5.1 Giới thiệu Web Service

Theo định nghĩa của W3C (World Wide Web Consortium), dịch vụ Web là một hệ thống phần mềm được thiết kế để hỗ trợ khả năng tương tác giữa các ứng dụng trên các máy tính khác nhau thông qua mạng Internet, giao diện chung và sự gắn kết của nó được mô tả bằng XML. Dịch vụ Web là tài nguyên phần mềm có thể xác định bằng địa chỉ URL, thực hiện các chức năng và đưa ra các thông tin người dùng yêu cầu. Một dịch vụ Web được tạo nên bằng cách lấy các chức năng và đóng gói chúng sao cho các ứng dụng khác dễ dàng nhìn thấy và có thể truy cập đến những dịch vụ mà nó thực hiện, đồng thời có thể yêu cầu thông tin từ dịch vụ Web khác. Nó bao gồm các mô đun độc lập cho hoạt động của khách hàng và doanh nghiệp và bản thân nó được thực thi trên server.

Trước hết, có thể nói rằng ứng dụng cơ bản của Dịch vụ Web là tích hợp các hệ thống và là một trong những hoạt động chính khi phát triển hệ thống. Trong hệ thống này, các ứng dụng cần được tích hợp với cơ sở dữ liệu (CSDL) và các ứng dụng khác, người sử dụng sẽ giao tiếp với CSDL để tiến hành phân

tích và lấy dữ liệu. Trong thời gian gần đây, việc phát triển mạnh mẽ của thương mại điện tử và B2B cũng đòi hỏi các hệ thống phải có khả năng tích hợp với CSDL của các đối tác kinh doanh (nghĩa là tương tác với hệ thống bên ngoài – bên cạnh tương tác với các thành phần bên trong của hệ thống trong doanh nghiệp).

### 2.3.5.2 Đặc điểm của Web Service

#### ➤ Đặc điểm

- Dịch vụ Web cho phép client và server tương tác được với nhau ngay cả trong những môi trường khác nhau. Ví dụ, đặt Web server cho ứng dụng trên một máy chủ chạy hệ điều hành Linux trong khi người dùng sử dụng máy tính chạy hệ điều hành Windows, ứng dụng vẫn có thể chạy và xử lý bình thường mà không cần thêm yêu cầu đặc biệt để tương thích giữa hai hệ điều hành này.
- Phần lớn kỹ thuật của Dịch vụ Web được xây dựng dựa trên mã nguồn mở và được phát triển từ các chuẩn đã được công nhận, ví dụ như XML.
- Là sự kết hợp của việc phát triển theo hướng từng thành phần với những lĩnh vực cụ thể và cơ sở hạ tầng Web, đưa ra những lợi ích cho cả doanh nghiệp, khách hàng, những nhà cung cấp khác và cả những cá nhân thông qua mạng Internet.
- Một ứng dụng khi được triển khai sẽ hoạt động theo mô hình client-server. Nó có thể được triển khai bởi một phần mềm ứng dụng phía server ví dụ như PHP, Oracle Application server hay Microsoft.Net...
- Ngày nay dịch vụ Web đang rất phát triển, những lĩnh vực trong cuộc sống có thể áp dụng và tích hợp dịch vụ Web là khá rộng lớn như dịch vụ chọn lọc và phân loại tin tức (hệ thống thư viện có kết nối đến web portal để tìm kiếm các thông tin cần thiết); ứng dụng cho các dịch vụ du lịch (cung cấp giá vé, thông tin về địa điểm...), các đại lý bán hàng qua

mạng, thông tin thương mại như giá cả, tỷ giá hối đoái, đấu giá qua mạng... hay dịch vụ giao dịch trực tuyến (cho cả B2B và B2C) như đặt vé máy bay, thông tin thuê xe...

- Các ứng dụng có tích hợp dịch vụ Web đã không còn là xa lạ, đặc biệt trong điều kiện thương mại điện tử đang bùng nổ và phát triển không ngừng cùng với sự lớn mạnh của Internet. Bất kì một lĩnh vực nào trong cuộc sống cũng có thể tích hợp với dịch vụ Web, đây là cách thức kinh doanh và làm việc có hiệu quả bởi thời đại ngày nay là thời đại của truyền thông và trao đổi thông tin qua mạng. Do vậy, việc phát triển và tích hợp các ứng dụng với dịch vụ Web đang được quan tâm phát triển là điều hoàn toàn dễ hiểu.

#### ➤ Ưu và nhược điểm

##### **Ưu điểm:**

- Dịch vụ Web cung cấp khả năng hoạt động rộng lớn với các ứng dụng phần mềm khác nhau chạy trên những nền tảng khác nhau.
- Sử dụng các giao thức và chuẩn mở. Giao thức và định dạng dữ liệu dựa trên văn bản (text), giúp các lập trình viên dễ dàng hiểu được.
- Nâng cao khả năng tái sử dụng.
- Thúc đẩy đầu tư các hệ thống phần mềm đã tồn tại bằng cách cho phép các tiến trình/chức năng nghiệp vụ đóng gói trong giao diện dịch vụ Web.
- Tạo mối quan hệ tương tác lẫn nhau và mềm dẻo giữa các thành phần trong hệ thống, dễ dàng cho việc phát triển các ứng dụng phân tán.
- Thúc đẩy hệ thống tích hợp, giảm sự phức tạp của hệ thống, hạ giá thành hoạt động, phát triển hệ thống nhanh và tương tác hiệu quả với hệ thống của các doanh nghiệp khác.

##### **Nhược điểm:**



- Những thiệt hại lớn sẽ xảy ra vào khoảng thời gian chết của Dịch vụ Web, giao diện không thay đổi, có thể lỗi nếu một máy khách không được nâng cấp, thiếu các giao thức cho việc vận hành.
- Có quá nhiều chuẩn cho dịch vụ Web khiến người dùng khó nắm bắt.
- Phải quan tâm nhiều hơn đến vấn đề an toàn và bảo mật.

### 2.3.5.3 Kiến trúc của Web Service

Dịch vụ Web gồm có 3 chuẩn chính: SOAP (Simple Object Access Protocol), WSDL (Web Service Description Language) và UDDI (Universal Description, Discovery, and Integration). UDDI được sử dụng để đăng ký và khám phá dịch vụ Web đã được miêu tả cụ thể trong WSDL. Giao tác UDDI sử dụng SOAP để nói chuyện với UDDI server, sau đó các ứng dụng SOAP yêu cầu một dịch vụ Web. Các thông điệp SOAP được gửi đi chính xác bởi HTTP và TCP/IP.



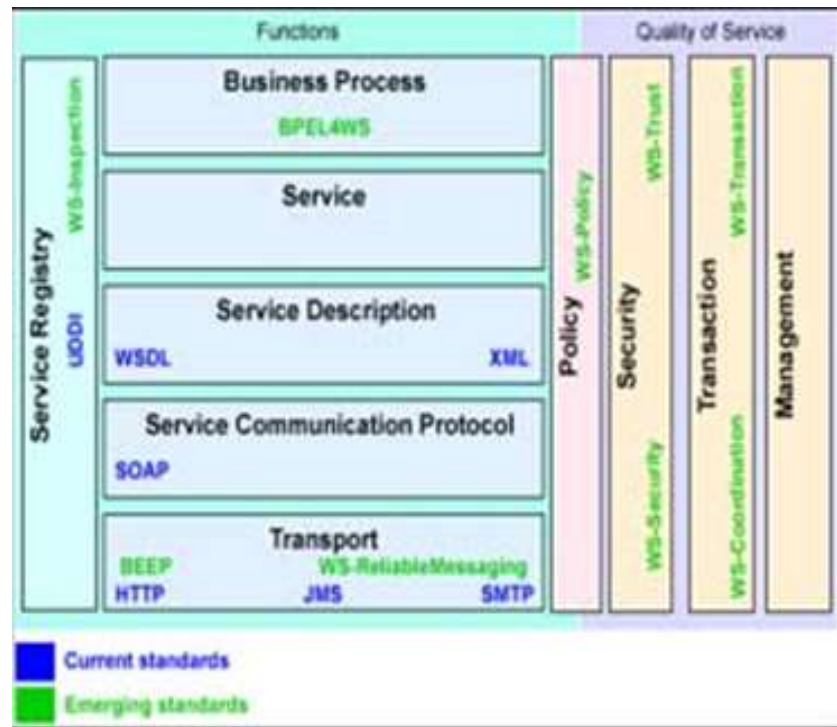
*Hình 2.12 Thành phần chính của Web Service*

Có 4 thành phần chính:

- Dịch vụ vận chuyển (Service Transport): có nhiệm vụ truyền thông điệp giữa các ứng dụng mạng, bao gồm những giao thức như HTTP, SMTP, FTP, JSM

và gần đây nhất là giao thức thay đổi khối mở rộng (Blocks Extensible Exchange Protocol- BEEP).

- Thông điệp XML: có nhiệm vụ giải mã các thông điệp theo định dạng XML để có thể hiểu được ở mức ứng dụng tương tác với người dùng. Hiện tại, những giao thức thực hiện nhiệm vụ này là XML-RPC, SOAP và REST.
- Mô tả dịch vụ: được sử dụng để miêu tả các giao diện chung cho một dịch vụ Web cụ thể. WSDL thường được sử dụng cho mục đích này, nó là một ngôn ngữ mô tả giao tiếp và thực thi dựa trên XML. Dịch vụ Web sẽ sử dụng ngôn ngữ này để truyền tham số và các loại dữ liệu cho các thao tác và chức năng mà dịch vụ Web cung cấp.
- Khám phá dịch vụ: tập trung dịch vụ vào trong một nơi được đăng ký, từ đó giúp một dịch vụ Web có thể dễ dàng khám phá ra những dịch vụ nào đã có trên mạng, tốt hơn trong việc tìm kiếm những dịch vụ khác để tương tác. Một dịch vụ Web cũng phải tiến hành đăng ký để các dịch vụ khác có thể truy cập và giao tiếp. Hiện tại, UDDI API thường được sử dụng để thực hiện công việc này.



Hình 2.13 Kiến trúc của Web Service

Trong đó, tầng giao thức tương tác dịch vụ (Service Communication Protocol) với công nghệ chuẩn là SOAP. SOAP là giao thức nằm giữa tầng vận chuyển và tầng mô tả thông tin về dịch vụ, cho phép người dùng triệu gọi một dịch vụ từ xa thông qua một thông điệp XML. Ngoài ra, để các dịch vụ có tính an toàn, toàn vẹn và bảo mật thông tin, trong kiến trúc dịch vụ Web, chúng ta có thêm các tầng Policy, Security, Transaction, Management.

#### 2.3.5.4 JSON và XML trong Web Service

##### **JSON là gì?**

- JSON là cú pháp để lưu trữ và trao đổi thông tin văn bản. Cũng giống như XML.
- JSON là nhỏ hơn so với XML, và nhanh hơn và dễ dàng hơn để phân tích (parse).

- JSON sử dụng cú pháp JavaScript để mô tả đối tượng dữ liệu, nhưng JSON là ngôn ngữ và nền tảng độc lập. Trình phân tích cú pháp(parsers) JSON và các thư viện JSON tồn tại cho nhiều ngôn ngữ lập trình khác nhau.
- JSON là viết tắt của JavaScript Object Notation.
- JSON là định dạng trao đổi dữ liệu văn bản dung lượng nhẹ.
- JSON là ngôn ngữ độc lập.
- JSON được “tự mô tả” và dễ hiểu.

### **Giống nhau và khác nhau so với XML:**

#### **Giống nhau:**

- JSON là văn bản tron (không có định dạng(màu sắc, cỡ chữ,...)).
- JSON là “tự mô tả” (người dùng có thể hiểu được).
- JSON là phân cấp (có cấu trúc cây).
- JSON có thể được phân tích cú pháp (parse) bởi JavaScript.
- Dữ liệu JSON có thể được truyền đi bằng AJAX.

#### **Khác nhau:**

- Không có thẻ kết thúc.
- Ngắn hơn.
- Nhanh hơn để đọc và ghi.
- Có thể được phân tích cú pháp bằng hàm dựng sẵn trong JavaScript là eval().
- Sử dụng mảng (Array).
- Không dùng các từ reserve.

### **Tại sao dùng JSON?**

Trong các ứng dụng AJAX, JSON là nhanh hơn và dễ dùng hơn so với XML:

#### ➤ Sử dụng XML:

- Lấy một tài liệu XML

- Sử dụng XML DOM để lập qua tài liệu
- Trích xuất các giá trị và lưu trữ trong các biến
- Sử dụng JSON:
  - Lấy một chuỗi JSON
  - Dùng hàm eval () chuyển chuỗi JSON thành đối tượng JavaScript

### **Cú pháp JSON:**

Các quy luật cú pháp JSON : Cú pháp JSON là một tập hợp con của cú pháp đối tượng JavaScript.

- Dữ liệu là các cặp tên/giá trị (name/values)
- Dữ liệu được phân cách bằng dấu phẩy
- Đối tượng nằm trong cặp ngoặc nhọn ‘ {} ’
- Dấu ngoặc vuông giữ mảng ‘ [] ’

### **Cặp Tên/Giá trị của JSON :**

Một cặp Tên/Giá trị chứa tên 1 trường (trong dấu nháy đôi “”), theo sau là dấu phẩy, cuối cùng là giá trị:

```
"firstName" : "John"
```

Các giá trị của JSON

- 1 số (nguyên (integer) hay số thực (floating point))
- 1 chuỗi (nằm trong dấu nháy đôi “”)
- 1 số luận lý (true or false)
- 1 mảng (nằm trong dấu ngoặc vuông [])
- 1 đối tượng (nằm trong dấu ngoặc nhọn {})
- Kiểu null

### **Các đối tượng JSON :**

Các đối tượng được đặt trong dấu ngoặc nhọn {}.

Đối tượng có thể chứa nhiều cặp Tên/Giá trị (Name/Values):

```
{ "firstName": "John", "lastName": "Doe" }
```

Mảng của JSON : Mảng JSON nằm trong dấu ngoặc vuông [].

Mảng có thể chứa nhiều đối tượng:

```
{
  "employees": [
    { "firstName": "John", "lastName": "Doe" },
    { "firstName": "Anna", "lastName": "Smith" },
    { "firstName": "Peter", "lastName": "Jones" }
  ]
}
```

### **JSON sử dụng cú pháp JavaScript:**

Bởi vì JSON sử dụng cú pháp JavaScript, nên không cần 1 chương trình trung gian làm việc giữa JSON và JavaScript.

Với JavaScript bạn có thể tạo 1 mảng các đối tượng và gán dữ liệu như sau:

```
var employees = [
  { "firstName": "John", "lastName": "Doe" },
  { "firstName": "Anna", "lastName": "Smith" },
  { "firstName": "Peter", "lastName": "Jones" }
];
```

Truy cập mạng : lấy giá trị đầu tiên

```
employees[0].lastName;
```

Kết quả:

Doe
-----

Dữ liệu mới được gán như sau:

employees[0].lastName = “Jonatan”;
------------------------------------

### **Trích xuất thông tin từ JSON :**

Trong Android hỗ trợ 2 kỹ thuật trích xuất thông tin và thấy thông tin về từ webservice lưu theo kiểu JSON:

- Cách 1 : Sử dụng đối tượng JSONObject. Thiết lập 1 kiểu key/value. Key phải là duy nhất và không được null. Value của JSONObject, JSONArray phải là 1 trong các kiểu cơ bản sau : String, Booleans, Integers, Longs, Doubles, NULL.
- Cách 2 : Sử dụng đối tượng GSON. Được cung cấp và sử dụng khi AndroidApplication làm việc với file JSON.

### 2.3.6 Công nghệ sử dụng

Tiêu chí lựa chọn hệ thống:

- Thân thiện, dễ sử dụng.
- Mã nguồn không quá phức tạp, đảm bảo đủ thời gian hoàn thành các chức năng cho trang web.
- Giao diện dễ tùy biến.

Từ những phân tích, so sánh trên, nhóm tác giả chọn:

- Server: ngôn ngữ PHP vì những ưu điểm vượt trội như: đơn giản, dễ thực hiện, tốc độ nhanh, miễn phí, có tính mở, nhiều công nghệ hỗ trợ...
- Client: android, java ngôn ngữ phổ biến sau web application, cộng đồng hỗ trợ đông.

- Data sử dụng trong Web Service : JSON vì ưu điểm xử lý nhanh, tốc độ cao, chính xác.
- Hệ quản trị CSDL MySQL vì những tính năng ưu việt như: miễn phí, dễ dùng, xử lý nhanh, đáng tin cậy, ổn định...
- Công nghệ hỗ trợ khác như: jquery, ajax, CSS, web service, JSON, bootstrap twitter.



## Chương 3 PHÂN TÍCH THIẾT KẾ

### 3.1 Xác định yêu cầu

#### 3.1.1 Chức năng chính

##### 3.1.1.1 Chức năng của server

- ✓ *Phân chia quyền hạn:*
  - Admin có quyền cao nhất, toàn quyền đối với tất cả.
  - Mod có quyền thấp hơn admin một bậc.
  - Các admin thứ cấp khác như: quản lý user, quản lý liên hệ, quản lý dữ liệu có những quyền hạn nhất định.
  - Giáo viên.
  - User vip.
  - User.
- ✓ *Quản lý user:*
  - Thêm, xóa, sửa người dùng.
  - Quản lý bình luận, liên hệ từ người dùng.
  - Quản lý kết quả thi.
  - Các phần được đánh dấu(bookmark, favorite).
- ✓ *Quản lý cơ sở dữ liệu:*
  - Quản lý lý thuyết tiếng Anh như: video, bài nghe, ngữ pháp...
  - Quản lý các dạng bài thi: trắc nghiệm chọn ABCD, đọc hiểu true, false, sắp xếp câu.
  - Quản lý kết quả thi, các đánh dấu yêu thích.
- ✓ *Quản lý liên hệ, phản hồi.*
  - Nhận các phản hồi của user, đáp ứng yêu cầu, trả lời thắc mắc.
  - Nâng cấp sửa chữa chương trình.

✓ Ngoài ra còn có các chức năng khác như:

- Thống kê đăng kí, lần đăng nhập.
- Thống kê số lượng thành viên
- Export file pdf danh sách thành viên.
- Tạo file backup database.

### 3.1.1.2 Chức năng của client

- ✓ Cho phép người dùng đăng kí tài khoản, đăng nhập hệ thống, lấy lại mật khẩu.
  - Đăng ký: Xử lý, kiểm tra input của người dùng, input sẽ được gửi lên server để xử lý tiếp. Nhận response từ server và thông báo cho người dùng.
  - Đăng nhập: gửi yêu cầu đăng nhập của người dùng lên server và nhận response từ server.
- ✓ Cho phép người dùng thay đổi thông tin cá nhân, thay đổi mật khẩu, thay đổi hình đại diện.
- ✓ Người dùng có thể xem từ mới, thành ngữ, audio, video và đánh dấu, ghi nhớ cho lần xem sau.
  - Đánh dấu/bỏ đánh dấu các audio, video.
  - Xem trực tiếp các video.
  - Học ngữ pháp.
- ✓ Tổ chức thi trắc nghiệm theo nhiều dạng: trắc nghiệm ABCD, trắc nghiệm TRUE/FALSE, điền từ, sắp xếp câu. Người dùng có thể lưu lại kết quả thi của mình.
- ✓ Gửi liên hệ, thắc mắc đến quản trị.
- ✓ Liên kết facebook, google+.

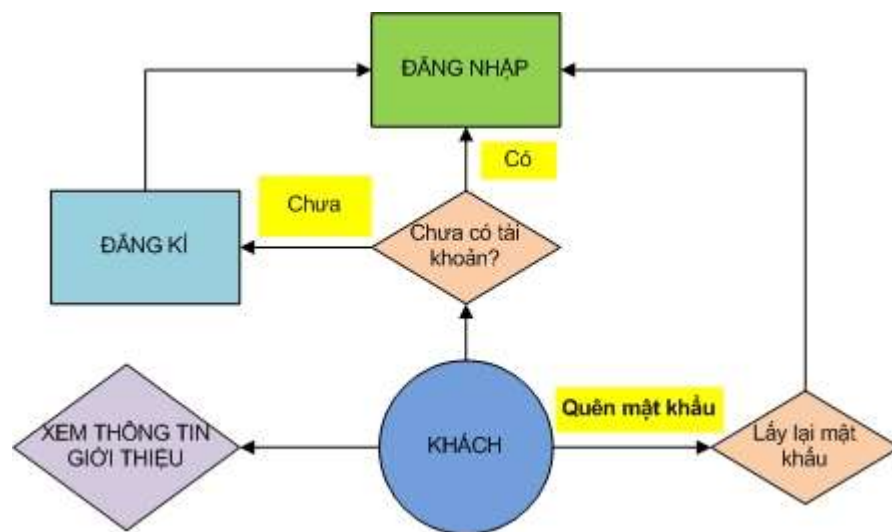
### 3.1.2 Mô hình hoạt động client-server

#### 3.1.2.1 Biểu đồ usecase

Biểu đồ usecase thể hiện số lượng chức năng mà người dùng có thể tương tác trực tiếp.

➤ Đối với khách:

Khách có thể đăng kí tài khoản để đăng nhập vào chương trình và bắt đầu sử dụng.



Hình 3.1 Chức năng dành cho khách

➤ Đối với thành viên:

Sau khi đăng nhập vào chương trình, thành viên sẽ có những chức năng nhất định.



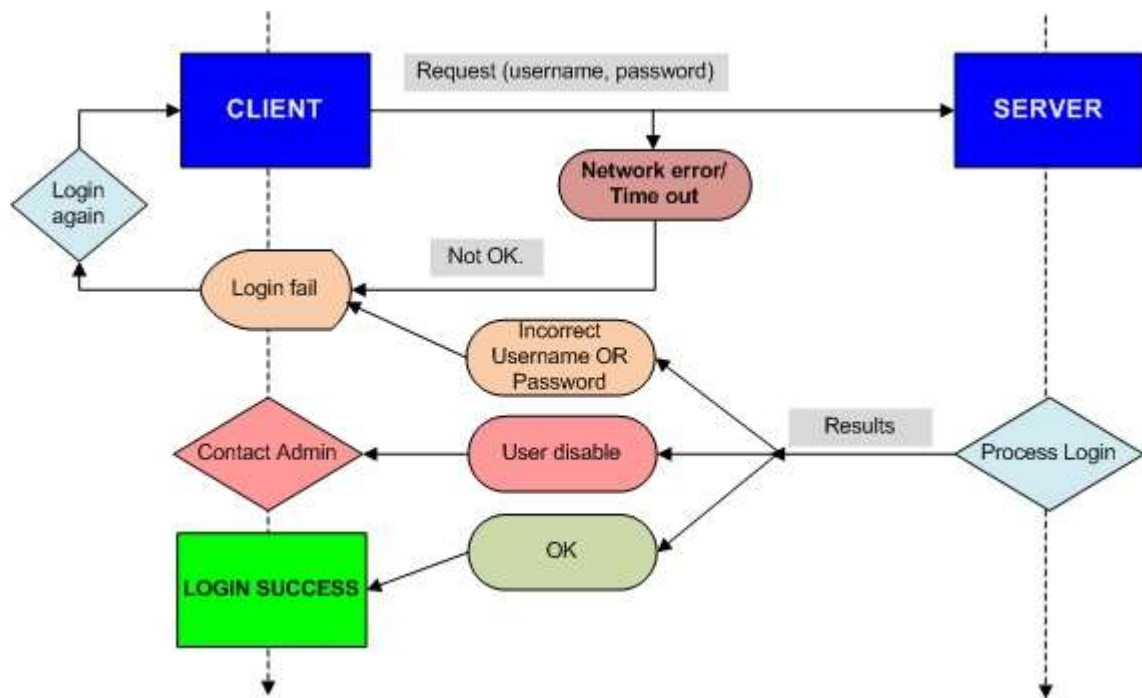
Hình 3.2 Chức năng dành cho thành viên

### 3.1.2.2 Biểu đồ xử lý

- Server nhận yêu cầu từ client, xử lý và sau đó trả lại kết quả cho client.
- Client gửi yêu cầu lên server, sau đó nhận kết quả và hiển thị cho người dùng.

Một số mô hình xử lý:

- ❖ Xử lý đăng nhập:



Hình 3.3 Mô hình xử lý đăng nhập

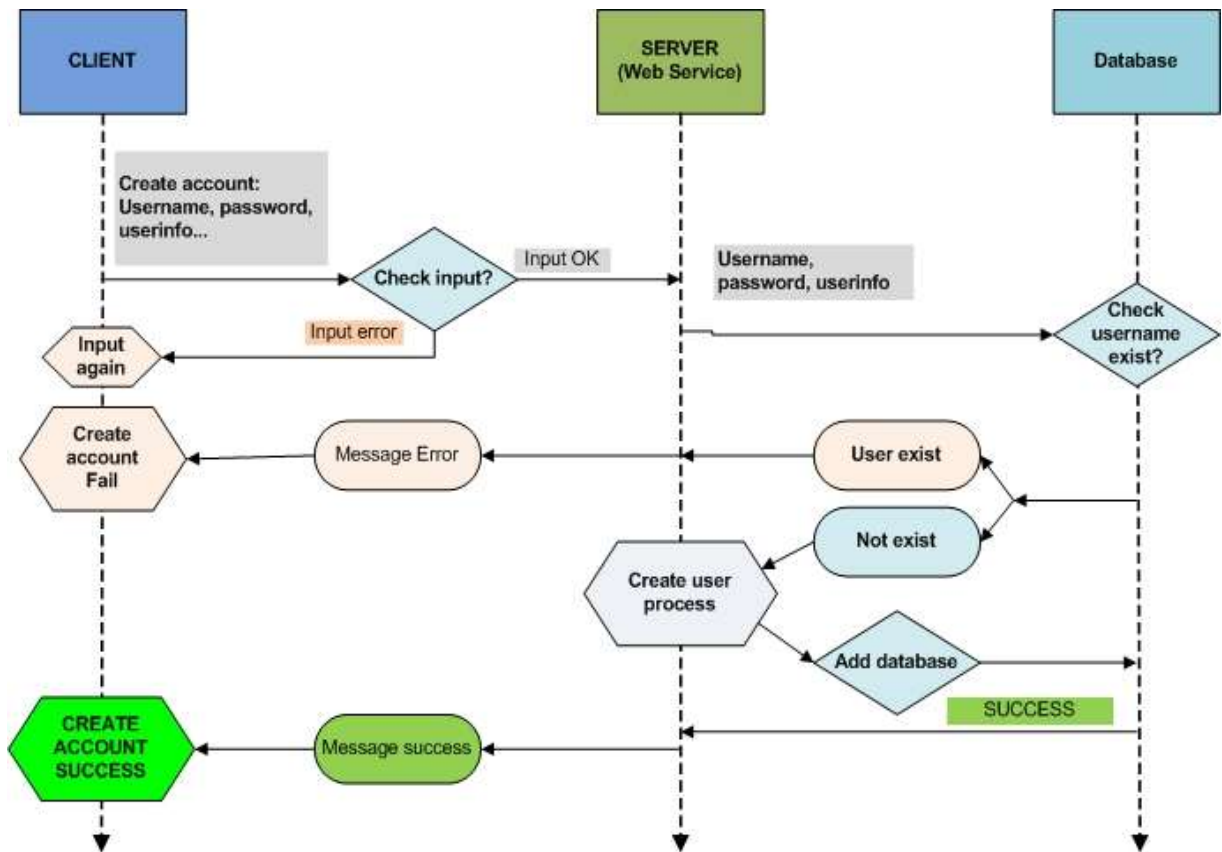
**Bước 1:** User gửi username, password đến server. Nếu có timeout hay internet lỗi thì hiển thị thông báo.

**Bước 2:** Server nhận data từ client, xử lý và gửi trả lại kết quả.

**Bước 3:** Nếu username hoặc password không đúng thì hiển thị lỗi và quay lại bước 1. Nếu tài khoản bị khóa thì hiện thông báo liên hệ với admin.

**Bước 4:** Đăng nhập thành công tiếp tục sử dụng chương trình.

❖ Xử lý đăng kí:



Hình 3.4 Mô hình xử lý đăng kí

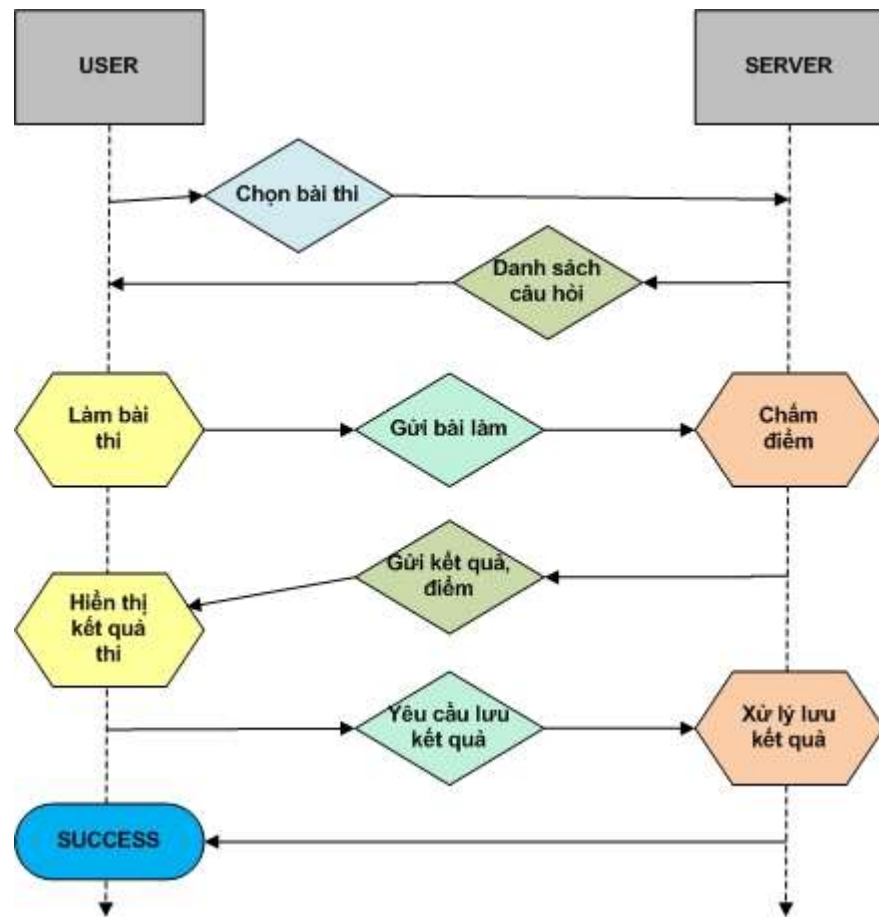
**Bước 1:** User nhập vào input, client kiểm tra input trước khi gửi lên server. Nếu input sai thì hiện thông báo nhập lại.

**Bước 2:** Gửi thông tin đăng kí lên server. Server kiểm tra username có tồn tại chưa. Nếu username tồn tại thì hiện thông báo lỗi và quay lại bước 1. Nếu username không tồn tại đến bước 3.

**Bước 3:** Xử lý đăng kí và thêm dữ liệu vào database. Gửi trả kết quả lại cho client.

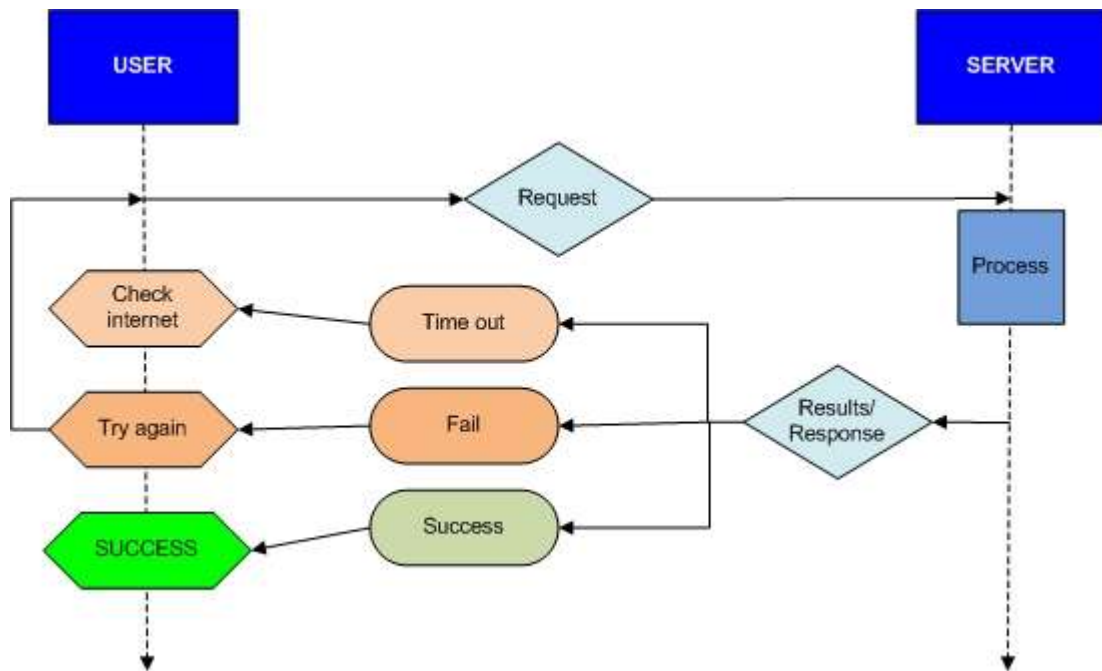
**Bước 4:** Hiện thị kết quả đăng kí thành công và chuyển user đến màn hình đăng nhập.

❖ Xử lý thi trắc nghiệm:



Hình 3.5 Mô hình xử lý thi trắc nghiệm

- ❖ Xử lý thay đổi thông tin, mật khẩu, thay đổi hình đại diện



Hình 3.6 Mô hình xử lý thay đổi thông tin

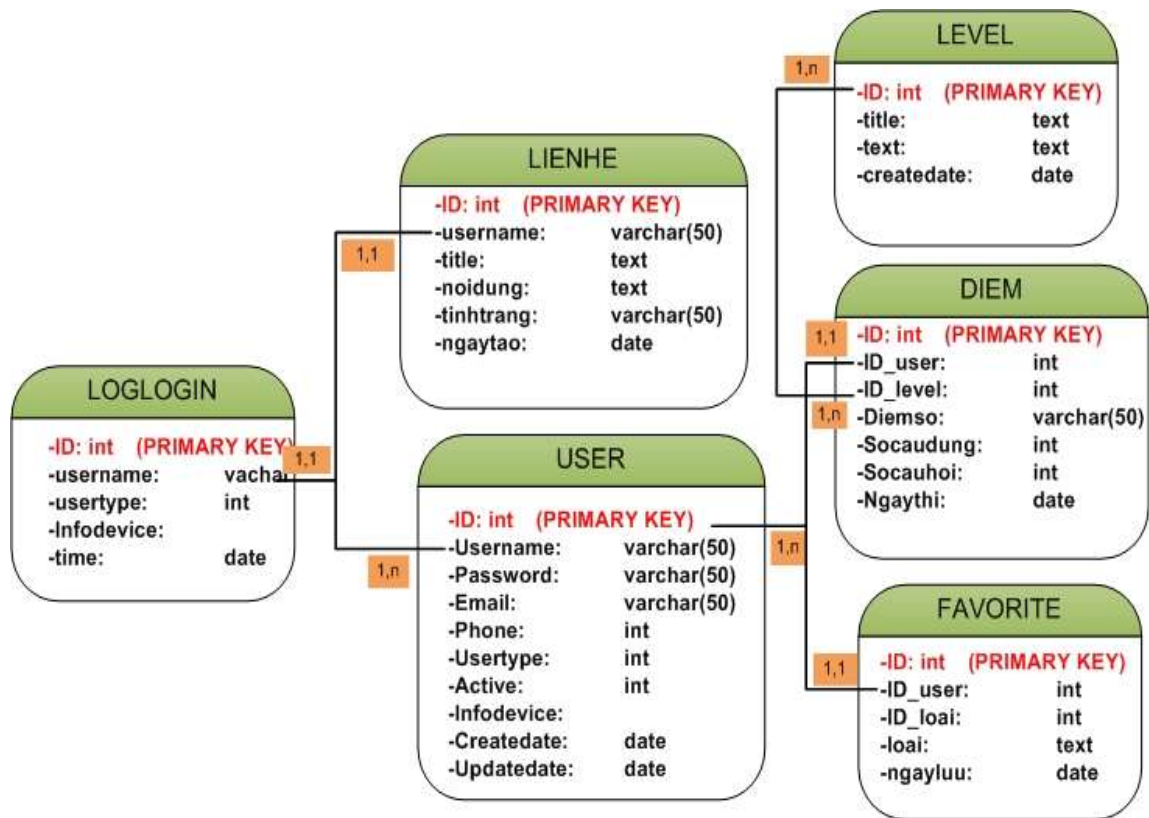
### 3.2 Phân tích thiết kế

#### 3.2.1 Thiết kế cơ sở dữ liệu

##### 3.2.1.1 Mô hình tổng quát

Mô hình cơ sở dữ liệu tổng quát thể hiện mối quan hệ giữa các bảng với nhau, thể hiện những thực thể trong cơ sở dữ liệu.





Hình 3.7 Sơ đồ tổng quát mối liên hệ giữa các bảng trong cơ sở dữ liệu

- Diễn giải sơ đồ:
  - o Danh sách các thực thể:

Bảng 3.1 Các bảng và thực thể trong sơ sở dữ liệu

<b>Thực thể</b>	<b>Mô tả</b>	<b>Thuộc tính – Chú thích</b>
<b>User</b>	<i>Thành viên</i>	<ul style="list-style-type: none"> <li>- id: mã user</li> <li>- username: tên đăng nhập</li> <li>- password: mật khẩu</li> <li>- email: địa chỉ email</li> <li>- phone: số điện thoại</li> <li>- infodevice: thông tin thiết bị</li> <li>- ngaytao: ngày tạo</li> </ul>
<b>Level</b>	<i>Các mức độ thi</i>	<ul style="list-style-type: none"> <li>- id: mã level</li> <li>- title: tên các cấp độ thi</li> <li>- text: nội dung</li> <li>- ngaytao: ngày tạo</li> </ul>
<b>Diem</b>	<i>Kết quả thi của user</i>	<ul style="list-style-type: none"> <li>- id: mã kết quả thi</li> <li>- id_user: mã user</li> <li>- id_level: id level thi</li> <li>- diem: điểm số</li> <li>- socaudung: số câu đúng</li> <li>- socauhoi: số câu hỏi</li> <li>- ngaythi: ngày thi</li> </ul>

<b>Favorite</b>	<i>Đánh dấu yêu thích của user</i>	<ul style="list-style-type: none"> <li>- id: mã</li> <li>- id_user: mã user</li> <li>- id_loai: mã của từng loại</li> <li>- loai: từ/thành ngữ/audio/video</li> <li>- ngayluu: ngày lưu</li> </ul>
<b>Loglogin</b>	<i>Ghi lại đăng nhập của user</i>	<ul style="list-style-type: none"> <li>- id: mã</li> <li>- username: user đăng nhập</li> <li>- info device: thông tin thiết bị đăng nhập</li> <li>- time: thời gian đăng nhập</li> </ul>

- Danh sách các mối kết hợp:

*Bảng 3.2 Danh sách các mối kết hợp*

<b>Tên mối kết hợp</b>	<b>Thực thể 1</b>	<b>Thực thể 2</b>	<b>Mối quan hệ</b>	<b>Ý nghĩa</b>
Kết quả thi	User	Diem	1,n	Một user có thể thi nhiều lần
Liên hệ/ Bình luận	User	Lienhe	1,n	Một user có thể gửi nhiều liên hệ/bình luận
Cấp độ thi	Level	Diem	1,1	Một kết quả thi chỉ thuộc một level cấp độ
Yêu thích	User	Favorite	1,n	Một user có thể đánh dấu nhiều yêu thích
Yêu thích	Favorite	Tudien/ Thanhngu/ Audio/ Video	1,1	Một yêu thích chỉ thuộc một loại (tudien/ thanhngu/ audio/ video)

### 3.2.1.2 Thiết kế chi tiết

#### ❖ Bảng USER:

Ý nghĩa: đặc trưng cho từng user, ghi lại thông tin của user, thông tin thiết bị mà user dùng đăng kí. Khóa chính là ID dùng phân biệt user với nhau.

Cột	Kiểu dữ liệu	Khóa chính	Ý nghĩa
ID	Int	X	Đặc trưng cho từng user, phân biệt user với nhau
Username	Text		Tên đăng nhập
Password	Text		Mã hóa 2 lần md5, mật khẩu của user
Hovaten	Text		Họ và tên user
Gioitinh	Varchar(10)		Nam/Nữ
Phone	Int		Số điện thoại
Usertype	Int		Loại tài khoản
Active	Int		User có bị khóa
Brand	text		Thông tin thiết bị user dùng đăng kí
Manufacturer	Text		Thông tin thiết bị user dùng đăng kí
Product	Text		Thông tin thiết bị user dùng đăng kí
Android_version	Text		Thông tin thiết bị user dùng đăng kí
Ngay_tao	Date		Ngày tạo

*Bảng 3.3 Bảng user và các thuộc tính*

#### ❖ Bảng TUDIEN:

Ý nghĩa: thể hiện những từ thông dụng hay gặp. Mỗi từ sẽ có phần phát âm, dịch nghĩa, nghĩa rộng và cho phép user đánh dấu yêu thích.

Cột	Kiểu dữ liệu	Khóa chính	Ý nghĩa
ID	Int	X	Mã phân biệt từng từ
Tu	Varchar(10)		Từ
Loaitu	Varchar(10)		Loại từ : (v), (n), (adj),

			(adv)
Phienam	Text		Phiên âm
Dichnghia	Text		Dịch nghĩa
Nhieunghia	Text		Nghĩa mở rộng
Linkaudio	Text		Link
Ngaytao	Date		Ngày tạo

*Bảng 3.4 Bảng từ điển và các thuộc tính*

❖ Bảng THANHNGU:

Ý nghĩa: thể hiện các thành ngữ thường gặp trong tiếng Anh. Mỗi thành ngữ sẽ có nghĩa và ví dụ giúp user dễ hiểu hơn.

Cột	Kiểu dữ liệu	Khóa chính	Ý nghĩa
ID	Int	X	Mã
Thanhngu	Text		Thành ngữ
Noidung	Text		Nội dung
Vidu	Text		Ví dụ
Ngaytao	Date		Ngày tạo

*Bảng 3.5 Bảng thành ngữ và các thuộc tính*

❖ Bảng AUDIO:

Ý nghĩa: thể hiện những bài nghe. Mỗi bài nghe có phần nội dung, phần control audio và cho phép user đánh dấu yêu thích.

Cột	Kiểu dữ liệu	Khóa chính	Ý nghĩa
ID	Int	X	Mã
Title	Text		Tiêu đề bài nghe
Noidung	Text		Nội dung
Link	Text		Link audio
Luotview	Int		Lượt xem
Ngaytao	Date		Ngày tạo

*Bảng 3.6 Bảng audio và các thuộc tính*

❖ Bảng VIDEO:

Ý nghĩa: thể hiện các video hướng dẫn học tiếng Anh. Mỗi video nói về một chủ đề riêng.

Cột	Kiểu dữ liệu	Khóa chính	Ý nghĩa
ID	Int	X	Mã
Title	Text		Tiêu đề
Mo_ta	Text		Nội dung
Link	Text		Link
Luotview	Int		Lượt xem
Ngaytao	Date		Ngày tạo

*Bảng 3.7 Bảng video và các thuộc tính*

❖ Bảng DIEM:

Ý nghĩa: lưu kết quả thi của các user.

Cột	Kiểu dữ liệu	Khóa chính	Ý nghĩa
ID	Int	X	Mã
Id_user	Int		Mã user
Id_level	Int		Level thi
Diem	Float		Điểm thi
Socaudung	Int		Số câu đúng
Socauhoi	Int		Số câu hỏi
Ngaytao	Date		Ngày tạo

*Bảng 3.8 Bảng điểm và các thuộc tính*

❖ Bảng FAVORITE:

Ý nghĩa: hiển thị các yêu thích của user.

Cột	Kiểu dữ liệu	Khóa chính	Ý nghĩa
ID	Int	X	Mã
Id_user	Int		Mã user
Id_loai	Int		Id của loại
Loai	Varchar(10)		Loại
Ngayluu	Date		Ngày lưu

*Bảng 3.9 Bảng favorite và các thuộc tính*

## ❖ Bảng LIENHE:

Ý nghĩa: các liên hệ, bình luận của user.

Cột	Kiểu dữ liệu	Khóa chính	Ý nghĩa
ID	Int	X	Mã
Username	Varchar(50)		Username
Title	Text		Tiêu đề
Noidung	Text		Nội dung
Ngaygui	Date		Ngày gửi

*Bảng 3.10 Bảng lienhe và các thuộc tính*

## ❖ Bảng LEVEL:

Ý nghĩa: các cấp độ bài thi.

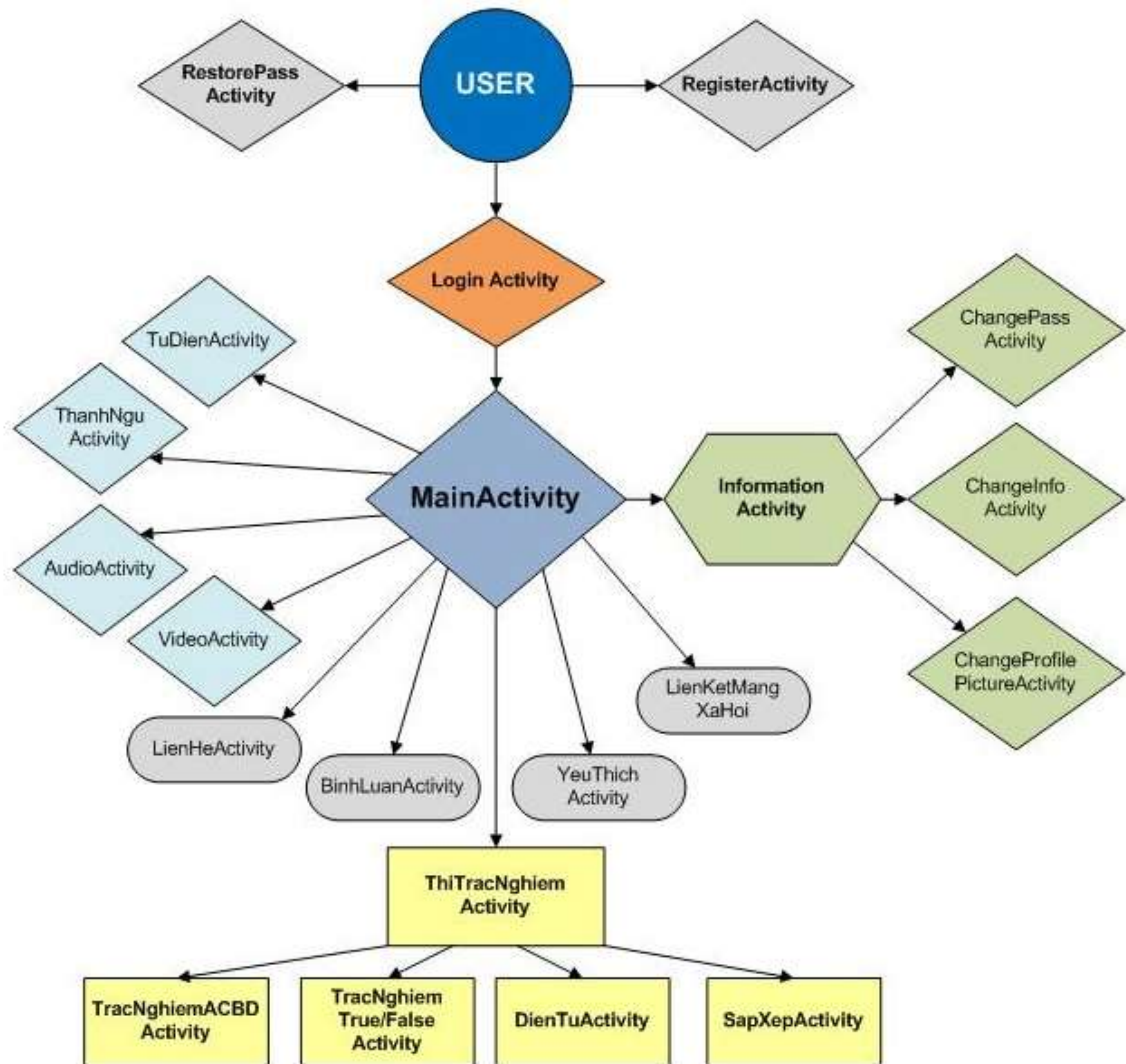
Cột	Kiểu dữ liệu	Khóa chính	Ý nghĩa
ID	Int	X	Mã
Title	Text		Tiêu đề
Noidung	Text		Nội dung
Ngaytao	Date		Ngày tạo

*Bảng 3.11 Bảng level và các thuộc tính*

### 3.2.2 Thiết kế mô hình lớp

#### 3.2.2.1 Mô hình lớp ở client - android

Mô hình lớp thể hiện mối liên hệ giữa các lớp với nhau, liên kết với nhau tạo giúp chương trình hoạt động tốt, tối ưu.



Hình 3.8 Mô hình lớp của chương trình.

❖ Lớp LoginActivity:

Là lớp chính và chạy đầu tiên khi chương trình chạy. Lớp này có nhiệm vụ xác thực user để cho phép user vào hệ thống.

Lớp này có các biến và function đặc trưng dùng xác thực user:

```
publicclass LoginActivity extends Activity
{
    EditText ed_username; //2 edittext để nhập vào username, password
    EditText ed_password;
    //Ngoài ra còn có các biến khác lưu thông tin sau khi đăng nhập thành công
```



```

public String username;
public String hovaten;
public String ngaysinh;
public String gioitinh;
public String diachi;
public String email;
public String sdt;
...
//Các function xử lý
class LoginProgress extends AsyncTask<String, String, String>
{
    @Override
    protected void onPreExecute()
    {
        ...
    }
    protected String doInBackground(String... args)
    {
        ...
    }
    protected void onPostExecute(String file_url) {
        ProgressDialog.dismiss();
    }
}

```

❖ Lớp RegisterActivity:

Lớp này dùng để khách đăng kí tài khoản, sau đó đăng nhập vào hệ thống. Lớp này đơn giản sẽ gửi input từ user đến webservice của server. Sau khi server xử lý sẽ gửi trả lại response cho client và hiển thị lên cho user biết.

```

public class RegisterActivity extends Activity
{
    //các biến
    EditText ed_name;
    EditText ed_hovaten;
    RadioGroup radGrp;
    EditText ed_pass;
    EditText ed_repeat_pass;
    EditText ed_diachi;
    EditText ed_email;
    public String username;
    public String hovaten;
    ...
    //function gửi data đến webservice
    class RegisterAccount extends AsyncTask<String, String, String>
    {
        @Override
    }
}

```

```

protectedvoid onPreExecute() {...}
protected String doInBackground(String... args) {...}
protectedvoid onPostExecute(String file_url) {...}
}
//Ngoài ra còn có các function check email, check date, check pass, check
length().

```

❖ Lớp RestorePassActivity:

Có nhiệm vụ giúp user khôi phục lại mật khẩu của mình trong trường hợp quên mật khẩu. Việc khôi phục mật khẩu phải dựa vào username, câu hỏi bí mật và câu trả lời bí mật khi đăng kí.

❖ Lớp MainActivity:

Là lớp chính của chương trình sau khi đăng nhập thành công. Lớp này xử lý đa số yêu cầu của user. Lớp này liên kết, có quan hệ đến hầu hết các lớp khác trong mô hình lớp.

```

publicclass MainActivity extends SlidingFragmentActivity
{
// chứa dữ liệu của user được gửi từ lớp LoginActivity
staticpublicintiduser;
staticpublic String username;
staticpublic String hovaten;
staticpublicintactive;
staticpublicintusertype;

...
//chứa dữ liệu data TuDien, ThanhNgu, Audio, Video
publicstatic ArrayList<TuDien>itemlistTuDien;
publicstatic ArrayList<TuDien>temp_itemlistTuDien;
publicstaticinttudien_all_item;
publicstaticinttudien_start;
publicstaticinttudien_end_of_List;
publicstaticintdang_tim_kiem;

...
//Ngoài ra còn có SlidingMenuLeft
itemlisSlidingMenu = new ArrayList<MainActivity.SampleItem>();
adapter.add(new SampleItem(" Liên Hệ ", R.drawable.icon_mail, 2));
adapter.add(new SampleItem(" Đăng Xuất ", R.drawable.icon_logout, 3));
...

//Dữ liệu Tudian, thanhngu, audio, video được hiển thị theo các Tab và
được quản lý bằng

```

```

mActionBar = getSupportActionBar();
mActionBar.setNavigationMode(ActionBar.NAVIGATION_MODE_TABS);
mPager = (ViewPager) findViewById(R.id.view_pager);
fm = getSupportFragmentManager();
mPager.setOnPageChangeListener(ViewPagerListener);
viewpageradapter = new ViewPagerAdapter(fm);
mPager.setAdapter(viewpageradapter);
...
tab = mActionBar.newTab().setText("Thông
Tin").setTabListener(tabListener).setIcon(R.drawable.icon_thong_tin_ca_nh
an_select);

mActionBar.addTab(tab);
tab = mActionBar.newTab().setText("Từ Thông
Dụng").setTabListener(tabListener).setIcon(R.drawable.icon_tu_moi_select)
;
mActionBar.addTab(tab);
...
}

```

❖ Lớp ThayDoiThongTinActivity:

Giúp cho user thay đổi thông tin cá nhân của họ.

❖ Lớp ThayDoiHinhDaiDienActivity:

Giúp user thay đổi profile picture.

❖ Lớp ThayDoiMatKhauActivity:

Giúp user thay đổi mật khẩu của họ.

❖ Lớp LienHeActivity/ BinhLuanActivity:

Hai lớp này giúp cho user có thể liên lạc với admin nếu có thắc mắc, cũng như có thể góp ý để chương trình càng tốt hơn.

```

publicclass Tab1_LienHe extends Activity
{
    private ProgressDialog pDialog;
    private FunctionCheck instantFunctionCheck;
    JSONParser jsonParser = new JSONParser();
    publicintsuccess = 0;
}

```

```

        EditText title;
        EditText message;
        Button btn;
        ...
class GuiLienHeProgress extends AsyncTask<String, String, String>
{
    @Override
    protectedvoid onPreExecute() {
        ...
    }
    protected String doInBackground(String... args)
    {
        ...
    }
    protectedvoid onPostExecute(String file_url)
    {
        ...
    }
}
}

```

❖ Lớp FavoriteActivity:

Giúp user dễ dàng vào các yêu thích đã được đánh dấu trước đó. Như vậy sẽ tiết kiệm thời gian tìm kiếm. Lớp này chia ra 4 loại yêu thích: Từ thông dụng, Thành ngữ, Audio, Video.

❖ Lớp ThiTracNghiemActivity:

Đây là lớp chính của thi trắc nghiệm. Lớp này có liên kết với từng loại thi riêng. Danh sách các bài thi sẽ được hiển thị thành list cho user chọn.

```

publicclass Tab8_ThiTracNghiem extends SherlockFragment
{
    private ProgressDialog pDialog;
    JSONParser jsonParser = new JSONParser();
    private FunctionCheck instantFunctionCheck;

    private ListView tab8_list_tracnghiem;
    private TracNghiemCustomAdapter adapter;
    private GetAllTracNghiem_first task_first;
    ...
    //List danh sách được quản lý bằng
    publicclass TracNghiemCustomAdapter extends ArrayAdapter
    <MainActivity.TracNghiem>
    {

```

```

        Context mContext;
        ArrayList<MainActivity.TracNghiem>items=null;
        intresource;
        ...

//Tùy user chọn bài thi gì thì sẽ chuyển đến bài thi đó
//
if("7".equals(iContent.getId() )
{
    Intent in = new
Intent(getSherlockActivity(),Tab8_True_False.class);
                                startActivity(in);
}
elseif("8".equals(iContent.getId() )
{
    Intent in = new
Intent(getSherlockActivity(),Tab8_Fill_Blank.class);
                                startActivity(in);
}
elseif("9".equals(iContent.getId() )
{
    Intent in = new Intent(getSherlockActivity(), Tab8_Sort.class);
                                startActivity(in);
}
else
{
    Intent in = new
Intent(getSherlockActivity(),Tab8_One_TracNghiem.class);

in.putExtra("id_user", String.valueOf(MainActivity.iduser));
in.putExtra("id_level", iContent.getId().toString());
in.putExtra("socauhoi", "10");
startActivity(in);
}
}
}
}

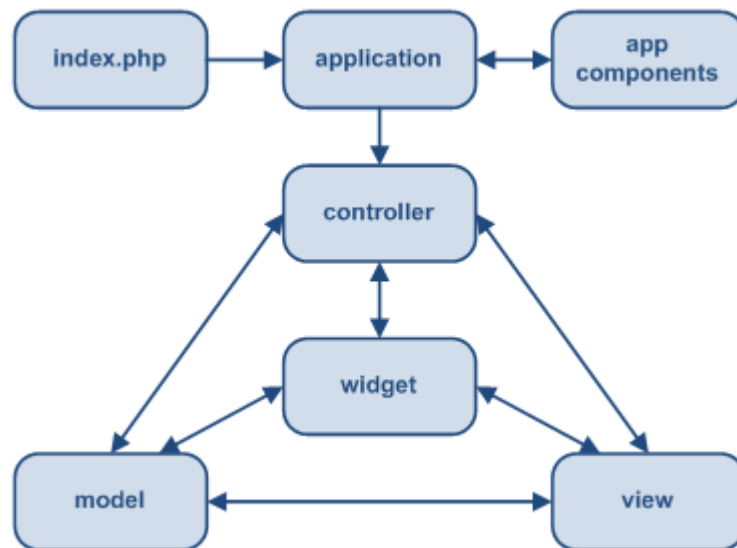
```

❖ Các lớp TracNghiemTrueFalse, FillBlankActivity, SortActivity:

Các lớp này đặc trưng cho từng loại thi riêng. Sau khi user làm bài thi xong, sẽ gửi kết quả lên server để chấm bài, sau đó nhận điểm từ server. User có thể lưu kết quả thi lại.

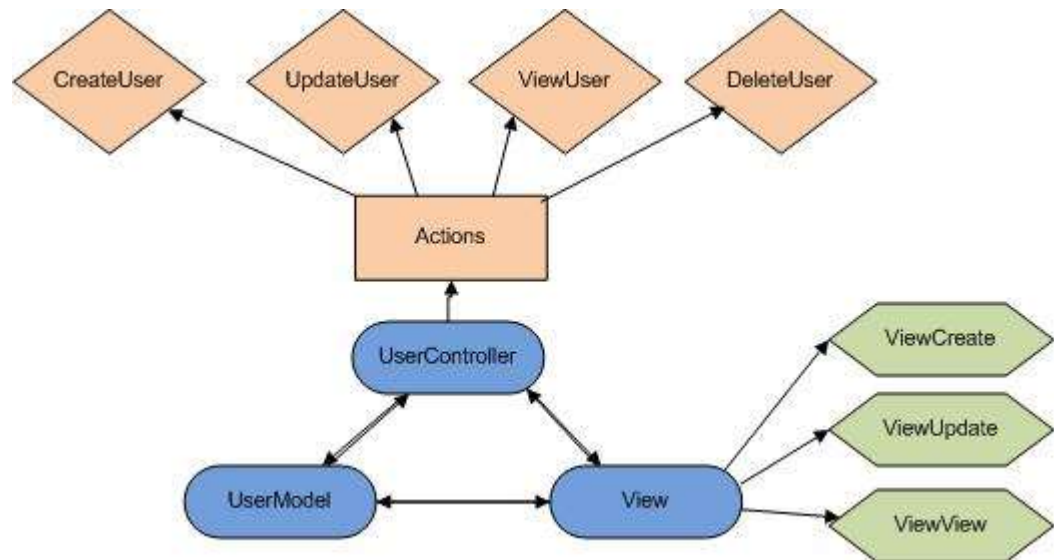
### 3.2.2.2 Mô hình lớp ở server

Ở server dùng Yii Framework nên mô hình sẽ theo chuẩn MVC (Model - View - Controller).



*Hình 3.9 Mô hình MVC trong Yii Framework*

Tùy vào từng chức năng riêng mà sẽ có mô hình đặc trưng cho chức năng đó. Như chức năng quản lý user, ta sẽ có mô hình MVC cụ thể như sau:



Hình 3.10 Mô hình MVC cụ thể cho quản lý user.

Server sẽ có các chức năng chính như:

- Quản lý user, phân quyền.
- Quản lý dữ liệu.
- Quản lý kết quả thi.
- Quản lý liên hệ/ bình luận.
- Quản lý đăng nhập.
- Quản lý yêu thích.
- Thống kê.
- Báo cáo, export pdf, backup database.

## Chương 4 XÂY DỰNG ỨNG DỤNG

### 4.1 Client

Phân tương tác với người dùng. Người dùng phải đăng ký và đăng nhập vào để có thể sử dụng được ứng dụng.

```
setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT
```

- ❖ Tùy chỉnh khả năng xoay màn hình của ứng dụng, ở đây ta chỉ cho phép ứng dụng dựng theo chiều dọc.

```
instantFunctionCheck =new FunctionCheck(getApplicationContext());
instantFunctionCheck.checkInternetConnection()
```

- ❖ Kết nối internet luôn được kiểm tra để đảm bảo ứng dụng hoạt động ổn định.

```
jsonParser.makeHttpRequest(Constant.BASE_URL_SERVER+" ", "POST", params);
JSONParser.getJSONObjectFromJString
```

- ❖ Truyền nhận dữ liệu với server với dạng “JSON” và sử dụng phương thức “POST”

#### 4.1.1 Danh sách các màn hình

STT	Tên màn hình	Ý nghĩa/ghi chú
1	Trang login	Gồm các chức năng về đăng nhập đăng ký.. cho người dùng trước khi sử dụng chương trình
2	Trang chính	Gồm menu left, actionbar tùy chọn các chức năng khác. Mặc định hiển thị thông tin các nhân



3	Trang thông tin cá nhân	Hiển thị thông tin cá nhân của người dùng cùng với các tùy chỉnh.
4	Trang từ điển	Danh mục các từ, dịch nghĩa, ví dụ, nghe từ.
5	Trang thành ngữ	Gồm các bài học thành ngữ, có thể lưu lại vào mục yêu thích.
6	Trang ngữ pháp	Gồm các bài học về ngữ pháp từ cơ bản đến nâng cao.
7	Trang bài nghe	Gồm các bài để luyện nghe tiếng Anh.
9	Trang video	Gồm các video học tiếng Anh.
10	Trang yêu thích	Trang lưu lại các bài nghe, ngữ pháp, thành ngữ để sử dụng cho những lần học sau.
11	Trang kết quả thi	Hiển thị kết quả các lần thi của người dùng.
12	Các trang khác	Các trang liên kết Facebook, Google+, chỉnh sửa thông tin cá nhân...

*Bảng 4.1 Danh sách các màn hình*

#### 4.1.2 Trang đăng nhập

Trang bắt đầu để đăng ký đăng nhập để sử dụng ứng dụng, và cung cấp thông tin về ứng dụng.

##### ❖ Giao diện



Hình 4.1 Màn hình đăng nhập

### Chi tiết màn hình

Sử dụng sliding layout với 2 scrollView có thể ẩn hiện thay thế nhau.

STT	Tên	Loại	Ý nghĩa
1	Scrollview tài khoản người dùng	Scrollview	Hiện thị các tùy chọn cho người dùng như đăng ký, đăng nhập.
2	Layout Scrollview Giới thiệu	Scrollview	Giới thiệu về ứng dụng

Bảng 4.2 Bảng mô tả trang đăng nhập

#### ➤ Scrollview tài khoản người dùng

STT	Tên	Loại	Ý nghĩa
1	ed_username	Edittext	Tùy chỉnh tên người dùng
2	ed_password	Edittext	Tùy chỉnh password
3	remember_username	Checkbox	Tùy chọn ghi nhớ username và password
4	Btnlogin	Button	Đăng nhập sau khi điền username và password

5	Btn_quen_mat_khau	Button	Chuyển đến trang trợ giúp về mật khẩu
6	Btn_register	Button	Chuyển đến trang đăng ký
7	Btn_thoat	Button	Đóng ứng dụng

Bảng 4.3 Bảng mô tả scrollview người dùng

➤ ScrollView giới thiệu

gồm các textview cung cấp các thông tin về ứng dụng.

❖ Nội dung chính

```
File file = new File(Environment.getExternalStorageDirectory() +
"/hocngoingu/remember/remember.txt");
FileWriter writer = new FileWriter(file);
writer.append(save);
writer.flush();
```

Chức năng ghi nhớ username và mật khẩu để dùng cho các lần đăng nhập tiếp theo. Username và password sẽ được lưu lại trong một file có tên remember trong đường dẫn `"/hocngoingu/remember/remember.txt"`.

```
String remember
=FunctionFolderAndFile.readFile("remember", "remember.txt");
```

Đọc user và pass từ file để thực hiện đăng nhập nếu trước đó đã chọn chức năng ghi nhớ.

#### 4.1.3 Trang chính (Home)

Gồm các tùy chọn cho người dùng để sử dụng các chức năng của chương trình.

❖ Ảnh chi tiết



Hình 4.2 Màn hình Home

❖ Mô tả chi tiết

STT	Tên	Loại	Ý nghĩa
1	Tab1_welcome_username	Textview	Hiển thị tên người dùng
2	Tab1_chuongtrinh	Textview	Hiển thị tên chương trình, lời khuyên
3	Tab1_btn_huongdan	Button	Chuyển đến trang hướng dẫn sử dụng
4	Tab1_btn_binhluan	Button	Chuyển đến trang bình luận
5	Tab1_btn_thongtintacgia	Button	Chuyển đến trang thông tin tác giả

Bảng 4.4 Bảng mô tả trang home

❖ Chi tiết màn hình

Trang chính sử dụng menu trái và actionBar để hiển thị các viewpaper.

- Menu\_left

Gồm listview và được add adapter trong code



Hình 4.3 Màn hình menu left

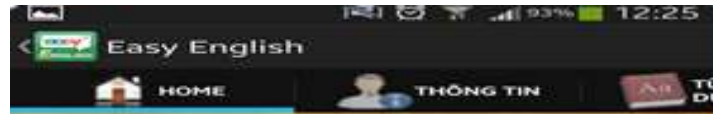
```

SampleAdapter_Custom adapter = new
SampleAdapter_Custom(getApplicationContext(),R.layout.menu_sliding_left,
itemlisSlidingMenu);
adapter.add(new SampleItem("tùy chọn", R.drawable.icon_logout, 4));
adapter.add(new SampleItem("Tin Nhắn ", R.drawable.icon_mail, 1));
adapter.add(new SampleItem("Liên hệ ", R.drawable.icon_lienhe, 2));
adapter.add(new SampleItem("đăng xuất ", R.drawable.icon_logout, 3));
adapter.add(new SampleItem("yêu thích", R.drawable.icon_logout, 4));
adapter.add(new SampleItem("từ thông dụng ", R.drawable.icon_tu_moi, 5));
adapter.add(new SampleItem("thành ngữ", R.drawable.icon_thanh_ngu, 6));
adapter.add(new SampleItem("Bài Nghe ", R.drawable.icon_audio_tab, 7));
adapter.add(new SampleItem("Video ", R.drawable.icon_video_tab, 8));
adapter.add(new SampleItem("CHIA SẺ ", R.drawable.icon_logout, 4));
adapter.add(new SampleItem("Facebook ", R.drawable.icon_facebook, 9));
adapter.add(new SampleItem("Twitter ", R.drawable.icon_twitter, 10));
adapter.add(new SampleItem("Google+ ", R.drawable.icon_google, 11));
lv_sliding_menu_id.setAdapter(adapter);

```

- ActionBar

Gồm các tab viewpager để chuyển đến các trang chức năng khác. Mặc định ở đây là tab home.



Hình 4.4 Các tab viewpaper

#### 4.1.4 Trang thông tin cá nhân

##### ❖ Ảnh chi tiết



Hình 4.5 Màn hình thông tin cá nhân

##### ❖ Mô tả chi tiết

Gồm các text view thông tin cá nhân như họ tên, ngày sinh, giới tính... và các button

STT	Tên	Ý nghĩa
1	Btn_thaydoi_thongtin	Chuyển đến trang thay đổi thông tin
2	Btn_thaydoi_matkhau	Chuyển đến trang thay đổi mật khẩu
3	Btn_thaydoi_hinhdaidien	Chuyển đến trang thay đổi hình đại diện
4	Btn_dangxuat	Đăng xuất khỏi tài khoản và chuyển lại ra trang bắt

đầu ứng dụng

*Bảng 4.5 Bảng mô tả trang thông tin cá nhân*

#### 4.1.5 Trang từ điển

❖ Ảnh chi tiết



*Hình 4.6 Màn hình từ điển*

❖ Mô tả chi tiết

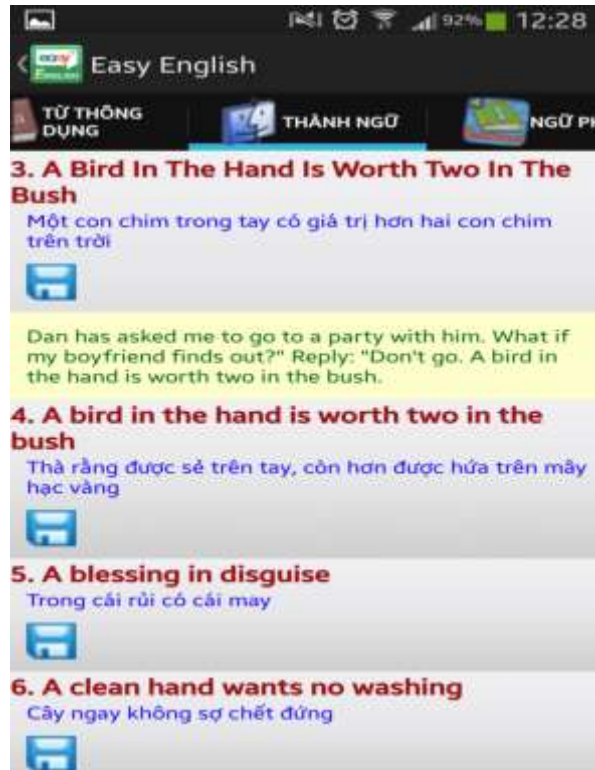
STT	Tên	Loại	Ý nghĩa
1	Tab3_tudien_tu	Textview	Hiển thị từ vựng
2	Tab3_tudien_loai	Textview	Hiển thị loại từ
3	Tab3_tudien_phienam	Textview	Hiển thị phiên âm của từ
4	Tab3_tudien_dichnghia	Textview	Hiển thị các nghĩa của từ
5	Tab3_tudien_save	Button	Lưu lại từ vào mục yêu thích
6	Tab3_tudien_listen	Button	Nghe phát âm của từ
7	Tab3_tudien_nhieunghia	Textview	Hiển thị các nghĩa khác của từ

*Bảng 4.6 Bảng mô tả trang từ điển*



#### 4.1.6 Trang thành ngữ

##### ❖ Ảnh chi tiết



Hình 4.7 Màn hình thành ngữ

##### ❖ Mô tả chi tiết

STT	Tên	Loại	Ý nghĩa
1	Tab4_thanhngu_thanhngu	Textview	Hiện thị thành ngữ
2	Tab4_thanhngu_noidung	Textview	Hiện thị dịch nghĩa của thành ngữ
3	Tab4_thanhngu_save	Button	Lưu lại vào mục thành ngữ yêu thích
4	Tab4_thanhngu_vidu	Textview	Hiện thị các ví dụ cho thành ngữ

Bảng 4.7 Bảng mô tả trang thành ngữ

#### 4.1.7 Trang ngữ pháp

##### ❖ Ảnh chi tiết





Hình 4.8 Màn hình ngữ pháp

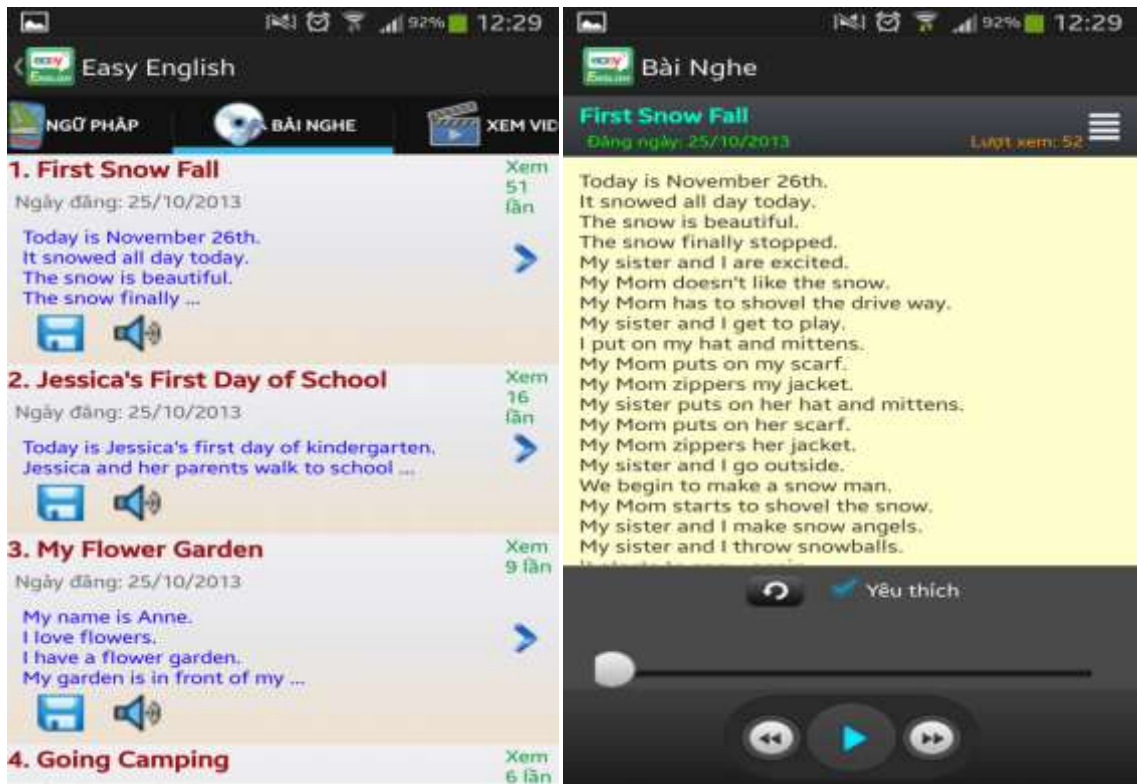
❖ Mô tả chi tiết

STT	Tên	Loại	Ý nghĩa
1	Tab5_nguphap_tieude	Textview	Hiển thị tiêu đề ngữ pháp
2	Tab5_nguphap_tomtat	Textview	Hiển thị tóm tắt nội dung của ngữ pháp
3	Tab5_nguphap_next_btn	Button	Chuyển đến trang tiếp
4	Tab5_nguphap_luot_view	Textview	Hiển thị lượt xem của bài ngữ pháp
5	Textview	Textview	Xem thêm chi tiết của bài ngữ pháp đó

Bảng 4.8 Bảng mô tả trang ngữ pháp

4.1.8 Trang bài nghe

❖ Ảnh chi tiết



Hình 4.9 Màn hình bài nghe

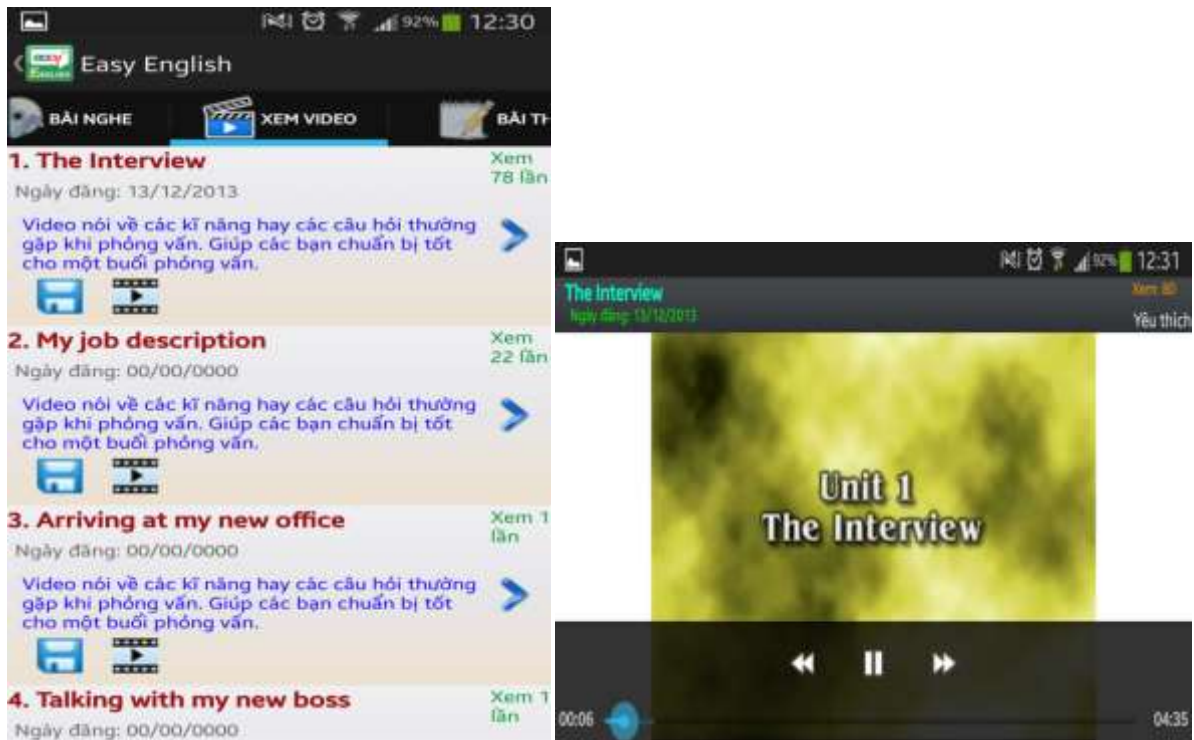
❖ Mô tả chi tiết

STT	Tên	Loại	Ý nghĩa
1	Tab6_audio_title	Textview	Hiển thị tiêu đề của bài nghe
2	Tab6_audio_text	Textview	Hiển thị nội dung tóm tắt của bài nghe
3	Tab6_audio_luotview	Textview	Hiển thị lượt nghe của bài nghe
4	Tab6_audio_ngaytao	Textview	Hiển thị ngày đăng của bài nghe
5	Tab6_audio_next_btn	Button	Chuyển đến trang nghe audio
6	Tab6_audio_save	Button	Chuyển bài nghe vào mục bài nghe yêu thích
7	Tab6_audio_listen	Button	Chuyển đến trang nghe audio

Bảng 4.9 Bảng mô tả trang audio

4.1.9 Trang video

❖ Ảnh chi tiết



Hình 4.10 Màn hình xem video

❖ Mô tả chi tiết

STT	Tên	Loại	Ý nghĩa
1	Tab7_video_title	Textview	Hiển thị tiêu đề video
2	Tab7_video_mo_ta	Textview	Hiển thị giới thiệu sơ về video
3	Tab7_video_luotview	Textview	Hiển thị lượt xem video
4	Tab7_video_ngaytao	Textview	Hiển thị ngày đăng video
5	Tab7_video_next_btn	Button	Chuyển đến trang để xem video
6	Tab7_video_save	Button	Lưu vào mục video yêu thích
7	Tab7_video_watch	Button	Chuyển đến trang để xem video

Bảng 4.10 Bảng mô tả trang video

4.1.10 Các trang về trắc nghiệm

4.1.10.1 Trang lựa chọn hình thức thi

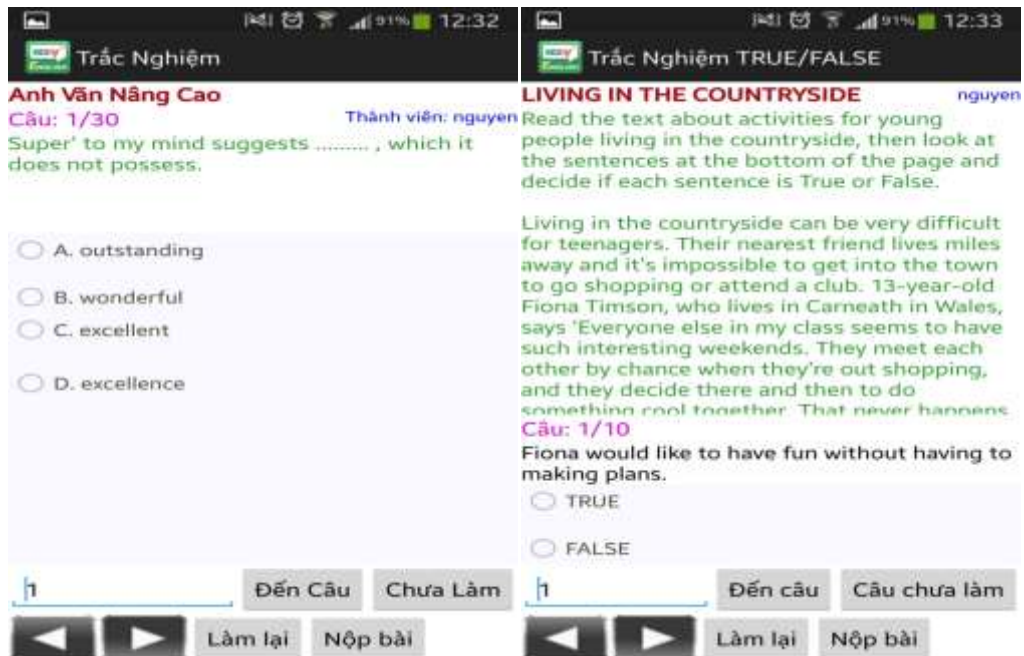
❖ Ảnh chi tiết



Hình 4.11 Trang lựa chọn thi trắc nghiệm

#### 4.1.10.2 Trang thi trắc nghiệm

##### ❖ Ảnh chi tiết



Hình 4.12 Màn hình thi trắc nghiệm

#### 4.1.11 Trang yêu thích

##### ❖ Ảnh chi tiết



Hình 4.13 Màn hình trang yêu thích

#### 4.1.12 Trang kết quả thi

##### ❖ Ảnh chi tiết





Hình 4.14 Màn hình kết quả thi

#### 4.1.13 Các trang chức năng khác

Ngoài ra còn có các trang chức năng khác như:

- ✓ Các trang liên kết Facebook, Twitter, Google+.
- ✓ Các trang thay đổi thông tin cá nhân, ảnh đại diện.
- ✓ Trang hướng dẫn sử dụng.

## 4.2 Server

Server có các chức năng chính như quản lý người dùng, đề thi, các bài học...Phân chia quyền hạn cho những người quản lý, tùy vào quyền mà họ được phép làm gì. Ví dụ: Admin có toàn quyền, trong khi đó Mod có quyền thấp hơn Admin, còn Mod User chỉ có quyền quản lý User.

### 4.2.1 Trang chủ đăng nhập

Trang chủ Tài khoản Đăng nhập

# Easy English

Trang Chủ Thống kê Quản Lý Bài Thi Yêu thích Kết quả thi Data Khác

Trang Chủ > Đăng Nhập

## Đăng Nhập

Tên Đăng Nhập \*

Mật Khẩu \*

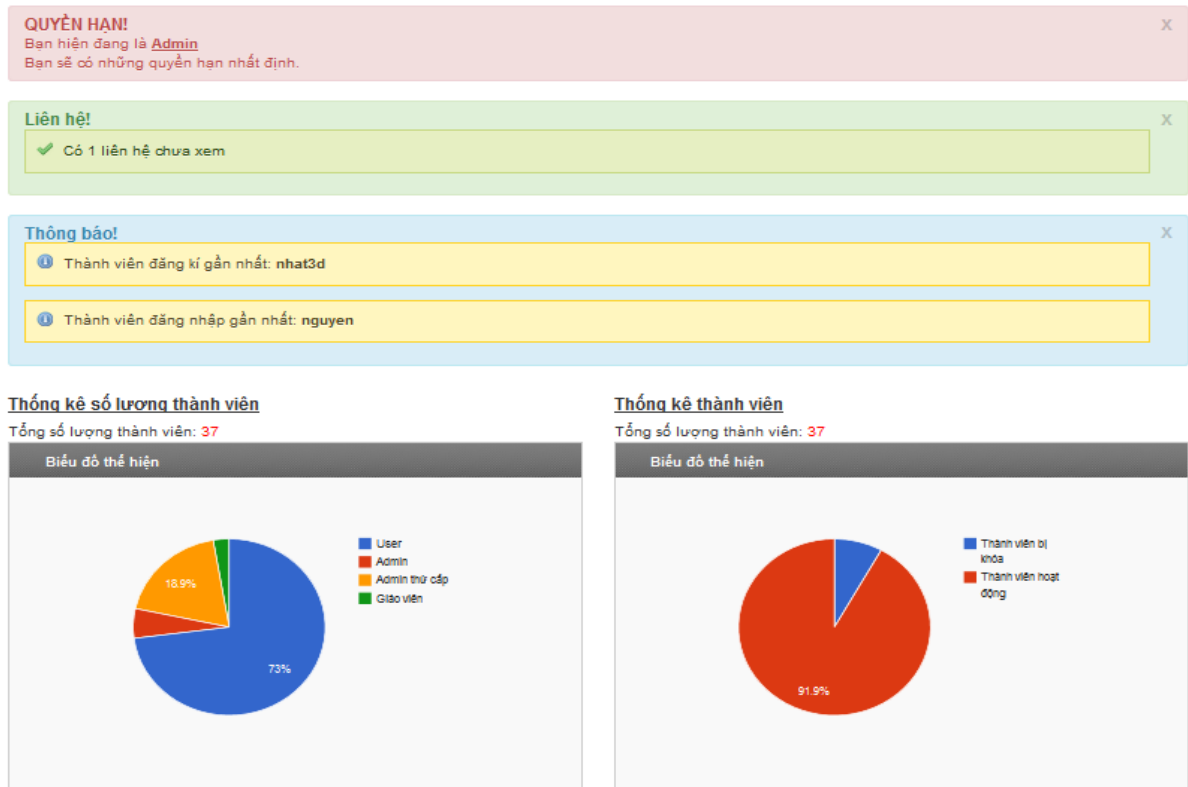
Ghi Nhớ Đăng Nhập

Đăng Nhập

Copyright © 2014 EASY ENGLISH  
All Rights Reserved.  
Powered by Yii Framework

Hình 4.15 Trang chủ đăng nhập

## TRANG QUẢN TRỊ



Hình 4.16 Giao diện trang chủ sau khi đăng nhập

➤ Chức năng trang:

- Hiện thị các thông báo như: liên hệ, thành viên đăng ký, đăng nhập gần nhất.
- Biểu đồ thống kê số lượng thành viên, thành viên bị khóa.

### 4.2.2 Trang thống kê hoạt động



### Thống kê hoạt động

ID	Username	Brand	Manufacturer	Model	Device	Android Version	Time login
107	nguyen	samsung	samsung	GT-I9500	ja3g	4.3	2014-02-13 00:47:49
106	nguyen	samsung	samsung	GT-I9500	ja3g	4.3	2014-02-13 00:25:09
105	nhat3d	generic	Genymotion	HTC One - 4.2.2 - API 17 - 1080x1920	vbox86p	4.2.2	2014-02-12 23:13:58
104	nhat3d	generic	Genymotion	HTC One - 4.2.2 - API 17 - 1080x1920	vbox86p	4.2.2	2014-02-12 23:13:45
103	nhat3d	generic	Genymotion	HTC One - 4.2.2 - API 17 - 1080x1920	vbox86p	4.2.2	2014-02-12 23:10:23
102	duyphan	alps	alps	S88	lcs77_cu_ics2	4.0.4	2014-02-12 06:18:40
101	nguyen	samsung	samsung	GT-I9500	ja3g	4.3	2014-02-12 05:44:33
100	nguyen	samsung	samsung	GT-I9500	ja3g	4.3	2014-02-12 05:43:23
99	nguyen	samsung	samsung	GT-I9500	ja3g	4.3	2014-02-12 05:43:05
98	nguyen	samsung	samsung	GT-I9500	ja3g	4.3	2014-02-12 05:42:41

Hình 4.17 Giao diện thống kê hoạt động

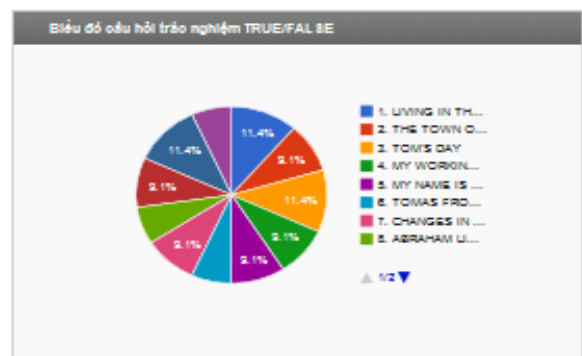
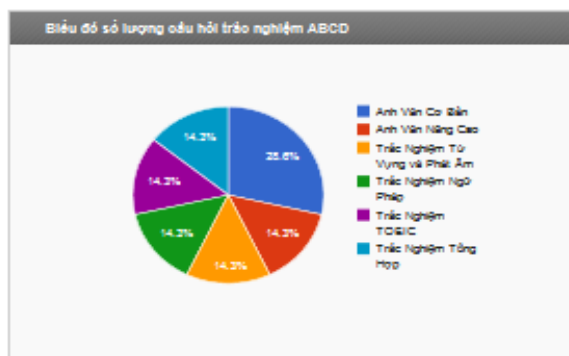
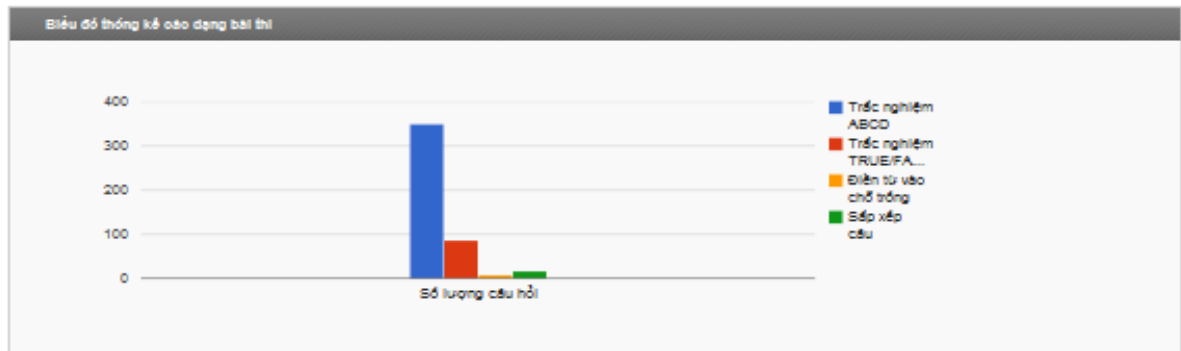
➤ Chức năng trang:

- Thống kê lượt đăng nhập gần nhất.
- Thống kê đăng ký gần đây.
- Thống kê lần thi gần nhất.
- Thống kê thành viên đăng nhập nhiều nhất.
- Thống kê thành viên thi nhiều nhất.

#### 4.2.3 Trang thống kê câu hỏi

## Thống kê câu hỏi

<p><b>Câu hỏi trắc nghiệm ABCD</b></p> <p>Tổng số câu hỏi: <b>350 câu</b>          Tổng số cấp độ: <b>6 cấp độ</b>          Số lượng câu hỏi mỗi cấp độ: <b>Biểu đồ dưới</b></p>
<p>Câu hỏi trắc nghiệm TRUE/FALSE</p>
<p>Câu hỏi điền từ vào chỗ trống</p>
<p>Câu hỏi sắp xếp</p>



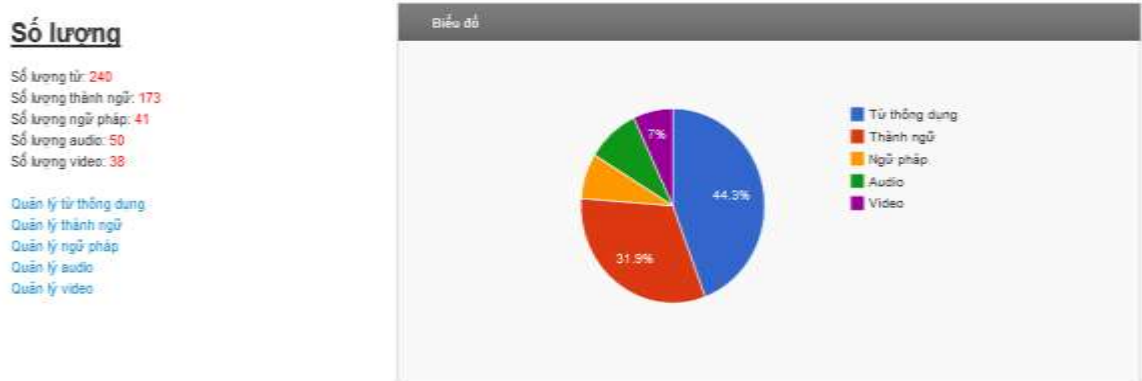
Hình 4.18 Giao diện trang thống kê câu hỏi

➤ Chức năng trang:

- Thông tin tổng số câu hỏi, cấp độ, số lượng câu mỗi cấp độ.
- Biểu đồ thống kê các dạng bài thi.
- Biểu đồ số lượng câu hỏi trắc nghiệm ABCD, TRUE/FALSE.

### 4.2.4 Trang thống kê video/ngữ pháp

## Thống kê số lượng audio/video/ngữ pháp...



Hình 4.19 Giao diện trang thống kê ngữ pháp

### 4.2.5 Các trang quản lý bài thi

Trang Chủ | Thống kê | Quản Lý | Bài Thi | Yêu thích | Kết quả thi | Data

Trang Chủ » Levels » Manage

## Quản lý các dạng bài

Tìm kiếm nâng cao

Các dạng bài thi

- Dạng thi trắc nghiệm ABCD
- Dạng thi điền vào chỗ trống
- Dạng thi sắp xếp câu
- Dạng thi Read And True/False question

ID	Tiêu đề	Nội dung	Ngày tạo	Ngày sửa	
1	Link: Anh Văn Cơ Bản	Đây là những câu trắc nghiệm cơ bản. Bài kiểm tra nhằm củng cố, bổ sung kiến thức cơ bản, thiết yếu cho các bạn mới học tiếng anh.	11/11/2013	11/11/2013	
2	Link: Anh Văn Nâng Cao	Các câu hỏi với độ khó tương đối. Bài kiểm tra cung cấp cho các bạn các kiến thức nâng cao, sâu rộng hơn trong tiếng anh.	09/11/2013	10/11/2013	

Hình 4.20 Giao diện trang quản lý bài thi

#### ➤ Chức năng:

- Quản lý dạng bài thi.


- Quản lý dạng thi trắc nghiệm ABCD.
- Quản lý dạng thi điền vào chỗ trống.
- Quản lý dạng thi sắp xếp câu.
- Quản lý dạng thi TRUE/FALSE.

#### 4.2.6 Trang quản lý yêu thích

## Quản lý yêu thích

Tìm kiếm nâng cao

Hiển thị 1-20 của 32 kết quả.

ID	ID - Username	Thuộc loại	STT loại	Ngày lưu	
	Chọn username	Chọn loại			
2	nguyen	Bài nghe	3	27/10/2013	 
5	giaovien	Bài nghe	5	00/00/0000	 
6	giaovien	Video	6	26/10/2013	 
7	nguyen	Từ thông dụng	5	25/10/2013	 
8	nguyen	Từ thông dụng	10	21/10/2013	 
9	nguyen	Từ thông dụng	15	20/10/2013	 
37	nguyen	Từ thông dụng	1	16/12/2013	 
32	nguyen	Thành ngữ	155	16/12/2013	 
18	nguyen	Từ thông dụng	144	07/10/2013	 
19	hocsinh4	Từ thông dụng	1	07/10/2013	 

Hình 4.21 Giao diện trang quản lý yêu thích

#### ➤ Chức năng trang

















- Tìm kiếm theo số thứ tự loại, ngày thi.
- Thêm xóa, sửa các phần yêu thích của user.

#### 4.2.7 Trang quản lý kết quả thi

# Kết quả thi

Tìm kiếm nâng cao

Hiển thị 1-9 của 9 kết quả.

ID	Username	Level name	Điểm	Số câu hỏi	Số câu đúng	Ngày thi	
	Chọn us <input type="text"/>	Chọn level <input type="text"/>					
25	nguyen	Trắc Nghiệm TOEIC	6.0	10	6	15/1/2014	 
22	nguyen	Anh Văn Cơ Bản	4.0	10	4	14/1/2014	 
23	nguyen	Dạng Câu Hỏi TRUE & FALSE	2.5	8	2	14/1/2014	 
24	nguyen	Điền Từ Thích Hợp Vào Chỗ Trống	3.0	10	3	14/1/2014	 
26	nguyen	Sắp Xếp Các Câu Thành Một Đoạn Văn	10.0	8	8	15/1/2014	 
27	nguyen	Anh Văn Cơ Bản	9.0	10	9	17/1/2014	 
28	duong1	Anh Văn Nâng Cao	4.0	10	4	24/01/2014	 
29	abodef	Điền Từ Thích Hợp Vào Chỗ Trống	0.0	10	0	25/01/2014	 
30	nguyen	Anh Văn Cơ Bản	1.0	30	3	13/02/2014	 

Hình 4.22 Giao diện quản lý kết quả thi

## 4.2.8 Các trang quản lý khác

The screenshot shows the 'Easy English' website's user management interface. The main heading is 'Quản lý User'. A dropdown menu is open, listing various management options: Quản lý User, Quản lý Bài Nghe, Quản lý Video, Quản lý Từ Thông Dụng, Quản lý Thành Ngữ, Quản lý Ngữ Pháp, Quản lý Bình luận, Quản lý Liên Hệ, and Quản lý Login. Below the menu, there is a search bar and a table of users with columns for ID, Tên tài khoản, Quyền hạn, Khóa User, and others. The table shows three users: admin, giaovien, and nguyen.

ID	Tên tài khoản	Quyền hạn	Khóa User	Kh	Ngày	Avatar	Brand	Model	Product	Device	Actions	
1	admin	Admin	Không bị khóa	Nguyễn	20/09/2013							
2	giaovien	Giáo viên	Không bị khóa	Trần Văn A	20/09/2013							
3	nguyen	User	Không bị khóa	Trần Võ Khôi	23/09/2013		nguyen.jpg	samsung	GT-I9500	ja3gxx	ja3g	

➤ **Gồm:**

- Quản lý user.
- Quản lý bài nghe.
- Quản lý video.
- Quản lý từ thông dụng.
- Quản lý thành ngữ.
- Quản lý ngữ pháp.
- Quản lý bình luận.
- Quản lý liên hệ.
- Quản lý login.

#### 4.2.9 Trang backup cơ sở dữ liệu

## Backup Database

Table Audio	<input type="text" value="yes"/>	Table User	<input type="text" value="yes"/>
Table Video	<input type="text" value="yes"/>	Table Login	<input type="text" value="yes"/>
Table Từ	<input type="text" value="yes"/>	Table Bình luận	<input type="text" value="yes"/>
Table Thành ngữ	<input type="text" value="yes"/>	Table Liên Hệ	<input type="text" value="yes"/>
Table Ngữ pháp	<input type="text" value="yes"/>	Table Điểm Thi	<input type="text" value="yes"/>
		Table Yêu Thích	<input type="text" value="yes"/>

---

Table Level	<input type="text" value="yes"/>
Table Level Question	<input type="text" value="yes"/>
Table Level Fill Blank	<input type="text" value="yes"/>
Table Level Sort	<input type="text" value="yes"/>
Table Level TRUE/FALSE	<input type="text" value="yes"/>
Table Level Question TRUE/FALSE	<input type="text" value="yes"/>

*Hình 4.23 Giao diện trang backup database*

➤ Chức năng trang:

- Tùy chọn các dữ liệu cần backup.

### 4.2.10 Trang quản lý file backup

## List file backup

Hiển thị 1-10 của 12 file backup

Thời gian (dd/mm/yyyy hh:ii:ss) ↕	Tên file	Kích thước	Chọn
06-02-2014 23:54:31	backup_06-02-2014_23-54-31.sql	1.45 MB	<a href="#">Download</a> <a href="#">Delete</a>
06-02-2014 14:23:56	backup_06-02-2014_14-23-56.sql	1.45 MB	<a href="#">Download</a> <a href="#">Delete</a>
27-01-2014 15:12:19	backup_27-01-2014_15-12-19.sql	1.45 MB	<a href="#">Download</a> <a href="#">Delete</a>
27-01-2014 15:12:16	backup_27-01-2014_15-12-16.sql	1.45 MB	<a href="#">Download</a> <a href="#">Delete</a>
27-01-2014 15:12:13	backup_27-01-2014_15-12-13.sql	1.45 MB	<a href="#">Download</a> <a href="#">Delete</a>
27-01-2014 15:12:10	backup_27-01-2014_15-12-10.sql	1.45 MB	<a href="#">Download</a> <a href="#">Delete</a>
27-01-2014 15:12:07	backup_27-01-2014_15-12-07.sql	1.45 MB	<a href="#">Download</a> <a href="#">Delete</a>
27-01-2014 15:12:03	backup_27-01-2014_15-12-03.sql	1.45 MB	<a href="#">Download</a> <a href="#">Delete</a>
27-01-2014 15:11:51	backup_27-01-2014_15-11-51.sql	1.45 MB	<a href="#">Download</a> <a href="#">Delete</a>
27-01-2014 15:01:58	backup_27-01-2014_15-01-58.sql	1.45 MB	<a href="#">Download</a> <a href="#">Delete</a>

1 2

*Hình 4.24 Giao diện trang quản lý file backup*

- Chức năng trang:
  - Download dữ liệu đã backup.
  - Delete file dữ liệu backup.



## Chương 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Sau quá trình nghiên cứu, tìm hiểu và thực hiện nhóm đã xây dựng thành công ứng dụng phục vụ cho việc dạy và học tiếng Anh.

Về mặt lý thuyết đồ án trình bày một số nghiên cứu về nền tảng android, so sánh với các nền tảng trên điện thoại khác để thấy được ưu nhược điểm của chúng. Ngoài ra nhóm còn nghiên cứu về các công nghệ web, framework Yii để phục vụ cho việc phát triển ứng dụng tốt hơn trong việc quản trị, kết nối dữ liệu.

Về thực nghiệm ứng dụng đã có khả năng áp dụng trong thực tế, mang lại lợi ích cho người dùng.

### ❖ Kết quả đạt được

- Ứng dụng đã hoàn thành các yêu cầu được đặt ra.
- Chương trình cung cấp tương đối đầy đủ các tính năng của một ứng dụng dạy và học.
- Chương trình có giao diện thân thiện, đẹp mắt dễ sử dụng.
- Client và server liên kết rõ ràng, độc lập để có thể dễ dàng trong việc quản lý, phát triển.

### ❖ Hạn chế

- Chương trình bắt buộc người dùng phải có kết nối internet từ điện thoại.
- Chương trình chưa được sử dụng offline.
- Tốc độ chương trình chưa được tối ưu.
- Chưa chạy thử trên hệ thống lớn, chương trình có vài chục user và vài kết nối cùng lúc.

### ❖ Hướng phát triển

- Chương trình có thể phát triển trên nền tảng khác như Windows Phone, IOS.
- Thêm các tính năng khác như kết nối các user, so sánh kết quả, danh sách thành viên điểm cao, kết quả cao làm động lực trong việc học.
- Thương mại hóa account, quảng cáo.
- Liên kết các trung tâm ngoại ngữ, cung cấp danh sách thành viên cho các trung tâm.

## DANH MỤC TÀI LIỆU THAM KHẢO

### Tiếng Việt:

1. Cleverlearn Vietnam English Language Center. *3000 từ vựng tiếng Anh thông dụng nhất của Oxford*, 2013.
2. Nguyễn Anh Tuấn. *Slide môn học Phát triển ứng dụng trên thiết bị di động*, Đại Học Công Nghệ Thông Tin.
3. Võ Thành Nhật - Phan Thành Trung. *Luận văn tốt nghiệp tìm hiểu hệ điều hành android và xây dựng ứng dụng minh họa*, Đại Học Khoa Học Tự Nhiên HCM, 2010.
4. Trần Hữu Phước. *Tìm hiểu và xây dựng ứng dụng Phone Book Safe Manager trên Android*. [Download].  
<http://vi.scribd.com/doc/56123159/Android#download>.
5. *Slide tài liệu android tiếng việt*. [Download].  
<http://www.kienthuc.name.vn>.
6. *YII 1.1 Application Development Cookbook - Vietnamese*, by longt8x - phamducbaotk10@gmail.com, August 28, 2012.
7. *Yii Framework - Design pattern and Database and Form*, by Team Justlook - Công ty truyền thông kỹ thuật số Toàn Cầu Xanh.

### Tiếng Anh:

1. Rick Rogers. *Android Application Development*. Gravenstein Highway North, Sebastopol, CA 95472: O'Reilly Media Inc, May 2009.

2. Kyle Merrifield Mew. *Android 3.0 Application Development Cookbook*. B27 6PA, UK: 32 Lincoln Road Olton Birmingham, July 2011.
3. Wei-Meng Lee. *Beginning Android™ Application Development*. Indianapolis, Indiana: Wiley Publishing, Inc., 2011.
4. Mark L. Murphy. *Beginning Android 2*. 233 Spring Street, 6th Floor, New York, Mar 2010.
5. W.FRANK ABLESON and CHARLIE COLLINS. *Unlocking Android*. ROBI SEN, 2009.
6. Mark Hancock. *English Pronunciation In Use*. 40 West 20th Street, New York, USA: Cambridge University Press, 2003.
7. Anne Taylor and Casey Malarcher. *Starter TOEIC 3th Edition - Building TOEIC Test-taking Skills*, 2006.
8. Qiang Xue and Xiang Wei Zhuo. *The Definitive Guide to Yii 1.1*. New York, 2008 - 2012.
9. Jeffery Winesett. *Agile Web Application Development with Yii 1.1 and PHP5*. 32 Lincoln Road, Olton, Birmingham, B27 6PA, UK, August 2010.
10. Alexander Makarov *Yii 1.1 Application Development Cookbook*. Livery Place, 35 Livery Street: Birmingham B3 2PB, UK, August 2011.

**Website:**

1. Yii Framework Vietnam Forum. [Online.] <http://yiivn.com/>, Oct , 2014.

2. Yii PHP Framework Forum International - Best for Web 2.0 Development. [Online] <http://www.yiiframework.com/>, August, 2014.
3. Izweb - Thiết kế Web theo chuẩn, <http://www.izwebz.com/>, Oct, 2014.
4. Android Developers, <http://developer.android.com/develop/index.html>, Oct, 2014.
5. Android Tag StackOverFlow, <http://stackoverflow.com/questions/tagged/android>, Oct, 2014.
6. OpenSource ActionBarSherlock Android Library, <http://actionbarsherlock.com/>, Oct, 2014.
7. OpenSource SlidingMenu Android Library, <https://github.com/jfeinstein10/SlidingMenu>, Oct, 2014.
8. OpenSource SlidingUpPanelLayout Android, <https://github.com/umano/AndroidSlidingUpPanel>, Oct, 2014.
9. Diễn đàn Android Việt Nam, <http://forum.vietnamandroid.com/>, August, 2014.
10. Ngữ Pháp Tiếng Anh, <http://tienganh360.com/category/trac-nghiem-ngu-phap-tieng-anh>, Dec 2014.
11. Website Trung tâm học tiếng Anh mỗi ngày, <http://hoctienganhmoingay.edu.vn/>, August, 2014.
12. Lớp Ngoại Ngữ, <http://lopngoaingu.com/>, Oct, 2014.
13. Học tiếng Anh qua video, <http://www.englishvideolesson.com/>, Oct, 2014.
14. Youtube học tiếng Anh cùng Mr Ducan - kênh của hoctienganhpro, <http://www.youtube.com/user/hoctienganhpro?feature=watch>, August, 2014.