TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

──────── \* ───────

ĐỒ ÁN

**TỐT NGHIỆP ĐẠI HỌC**

NGÀNH CÔNG NGHỆ THÔNG TIN

**XÂY DỰNG GAME TRỰC TUYẾN ĐẤU TRƯỜNG KIẾN THỨC TRÊN NỀN TẢNG ANDROID**

Sinh viên thực hiện : **Nguyễn Huy Hoàng**

Lớp KTMT – K52

Giáo viên hướng dẫn: [ThS] **Dư Thanh Bình**

HÀ NỘI 5-2011

# PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

1. Thông tin về sinh viên

Họ và tên sinh viên: Nguyễn Huy Hoàng

Điện thoại liên lạc: 01678567229 Email: hoangnh29@gmail.com

Lớp: Kỹ Thuật Máy Tính – K52 Hệ đào tạo: Đại học chính quy

Đồ án tốt nghiệp được thực hiện tại: công ty phần mềm PPCLINK

Thời gian làm ĐATN: Từ ngày / /201x đến / /201x

2. Mục đích nội dung của ĐATN: Xây dựng game trực tuyến đấu trường kiến thức trên nền tảng di động Android

3. Các nhiệm vụ cụ thể của ĐATN

* Nghiên cứu lập trình ứng dụng trên Android
* Nghiên cứu xây dựng web-server bằng Apache kết hợp PHP & MySQL
* Phân tích thiết kế xây dựng game trên Android
* Phân tích thiết kế xây dựng webserver bằng PHP kết hợp cơ sở dữ liệu MySQL
* Phân tích giao tiếp giữa game trên Android với server

4. Lời cam đoan của sinh viên:

Tôi *Nguyễn Huy Hoàng* cam kết ĐATN là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của *Ths. Dư Thanh Bình*.

Các kết quả nêu trong ĐATN là trung thực, không phải là sao chép toàn văn của bất kỳ công trình nào khác.

|  |  |
| --- | --- |
|  | *Hà Nội, ngày tháng năm*  Tác giả ĐATN  *Nguyễn Huy Hoàng* |

5. Xác nhận của giáo viên hướng dẫn về mức độ hoàn thành của ĐATN và cho phép bảo vệ:

|  |  |
| --- | --- |
|  | *Hà Nội, ngày tháng năm*  Giáo viên hướng dẫn |

# LỜI CẢM ƠN

*Tôi xin gửi lời cảm ơn trân trọng nhất tới thầy giáo hướng dẫn, Thạc sĩ Dư Thanh Bình giảng viên bộ môn Kỹ Thuật Máy Tính - Viện Công Nghệ Thông Tin và Truyền Thông - Trường Đại học Bách Khoa Hà Nội, đã nhiệt tình hướng dẫn, chỉ bảo và cung cấp cho tôi nhiều kiến thức cũng như tài liệu trong suốt quá trình làm đồ án. Nhờ sự giúp đỡ tận tình của thầy giáo, tôi mới có thể hoàn thành được đồ án này.*

*Tôi cũng xin gửi lời cảm ơn chân thành sâu sắc tới thầy Lương Ánh Hoàng và các thầy cô giáo trong trường Đại học Bách Khoa Hà Nội nói chung, các thầy cô giáo trong Viện Công Nghệ Thông Tin và Truyền Thông, bộ môn Kỹ Thuật Máy Tính nói riêng, những người đã tận tình giảng dạy, truyền đạt cho tôi những bài học quý báu trong suốt 5 năm học vừa qua.*

*Tôi xin chân thành cảm ơn công ty PPCLINK đã tạo điều kiện giúp đỡ tôi về môi trường, thiết bị cũng như kỹ thuật trong suốt quá trình làm đồ án.*

*Cuối cùng, tôi xin cảm ơn gia đình, bạn bè và những người thân đã ở bên tôi trong cuộc sống và cho tôi những sự động viên lớn lao, giúp đỡ tôi trong quá trình thực hiện đồ án này.*

*Hà Nội ngày 28 tháng 5 năm 2011*

*Sinh viên:* ***Nguyễn Huy Hoàng***

Contents

[PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP 2](#_Toc324340524)

[LỜI CẢM ƠN 3](#_Toc324340525)

[TÓM TẮT NỘI DUNG ĐỒ ÁN TỐT NGHIỆP 4](#_Toc324340526)

[MỞ ĐẦU 5](#_Toc324340527)

[PHẦN 1: ĐẶT VẤN ĐỀ VÀ ĐỊNH HƯỚNG GIẢI PHÁP 6](#_Toc324340528)

[1. Nhiệm vụ đề tài 6](#_Toc324340529)

[2. Định hướng giải quyết các vấn đề đặt ra 6](#_Toc324340530)

[3. Cơ sở lý thuyết và công cụ 6](#_Toc324340531)

[3.1. Android Programming 6](#_Toc324340532)

[3.2. PHP & MySQL Programming 6](#_Toc324340533)

[3.3. IDE & Tools 6](#_Toc324340534)

[PHẦN 2: CÁC KẾT QUẢ ĐẠT ĐƯỢC 7](#_Toc324340535)

[1. Phân tích thiết kế chương trình 7](#_Toc324340536)

[1.1. Phân tích biểu đồ usecase 8](#_Toc324340537)

[1.2. Phân tích biểu đồ luồng dữ liệu 11](#_Toc324340538)

[1.3. Phân tích biểu đồ thực thể liên kết và thiết kế dữ liệu 14](#_Toc324340539)

[2. Chi tiết kết quả thực hiện, cài đặt và thử nghiệm chương trình 18](#_Toc324340540)

[3. Thử nghiệm chương trình 18](#_Toc324340541)

[4. Triển khai chương trình 18](#_Toc324340542)

[KẾT LUẬN 19](#_Toc324340543)

[1. Kết quả đánh giá 19](#_Toc324340544)

[2. Định hướng phát triển 19](#_Toc324340545)

[3. Kết luận 19](#_Toc324340546)

[TÀI LIỆU THAM KHẢO 20](#_Toc324340547)

# MỞ ĐẦU

1. Nhiệm vụ cần thực hiện

Nghiên cứu lập trình ứng dụng Android và PHP. Xây dựng game trực tuyến “Đấu trường kiến thức”.

1. Môi trường thực hiện ĐATN

Đồ án được thực hiện tại công ty phần mềm PPCLINK, một công ty đã có nhiều năm kinh nghiệm lập trình ứng dụng Mobile, và hướng đi hiện tại của công ty là phát triển các ứng dụng trên nền tảng iOS và Android.

1. Bố cục ĐATN

Bố cục đồ án tốt nghiệp bao gồm các phần chính sau đây:

Chương 1: Giới thiệu về đề tài – “Xây dựng game trực tuyến đấu trường kiến thức trên Android”

1. Nhiệm vụ đề tài
2. Định hướng giải quyết các vấn đề đặt ra
3. Cơ sở lý thuyết và công cụ

Chương 2: Cơ sở lý thuyết và công cụ lập trình

1. Giới thiệu về Android
2. Giới thiệu về PHP và MySQL
3. IDE và tools sử dụng

Chương 3: Phân tích thiết kế hệ thống

1. Phân tích hệ thống: xây dựng các biểu đồ usecase, biểu đồ luồng dữ liệu, biểu đồ thực thể liên kết.
2. Thiết kế hệ thống:

Chương 4: Cài đặt thử nghiệm và triển khai hệ thống

# CHƯƠNG 1: GIỚI THIỆU VỀ GAME TRỰC TUYẾN ĐẤU TRƯỜNG KIẾN THỨC

## Nhiệm vụ đề tài

Xây dựng game trực tuyến “Đấu trường kiến thức” trên nền tảng Android. Đấu trường kiến thức là game cho phép nhiều người chơi tham gia thi đấu kiến thức với nhau ở bất cứ nơi đâu, bất cứ khi nào.

## Định hướng giải quyết các vấn đề đặt ra



Game trực tuyến được xây dựng dựa trên mô hình client ⬄ server. Nhiệm vụ cụ thể của chúng:

* Client: cung cấp cho người dùng giao diện chơi game, các thao tác giao tiếp với hệ thống.
* Server: quản lý dữ liệu của hệ thống đồng thời đóng vai trò là cầu nối giữa các người chơi với nhau.

Hướng giải quyết

* Nghiên cứu lập trình ứng dụng trên Android🡪 xây dựng ứng dụng phía client
* Nghiên cứu xây dựng web-server bằng Apache kết hợp PHP & MySQL🡪 xây dựng ứng dụng phía server
* Phân tích thiết kế xây dựng game trên Android
* Phân tích thiết kế xây dựng webserver bằng PHP kết hợp cơ sở dữ liệu MySQL
* Phân tích giao tiếp giữa client với server

## Các công việc cụ thể

**Lý do xây dựng đề tài:**

* Mục đích cá nhân: trò chơi được xây dựng trong quá trình làm đồ án tốt nghiệp nhằm tổng kết lại các kiến thức và kinh nghiệm đã được tích lũy trong 5 năm học tập và làm việc tại trường. Đó là các kiến thực IT liên quan đến các lĩnh vực:
  + Lập trình: kĩ thuật lập trình, lập trình hướng đối tượng, lập trình nhúng và web.
  + Cơ sở dữ liệu: phân tích và xây dựng cơ sở dữ liệu tuân theo các dạng chuẩn hóa
  + An ninh mạng và bảo mật:
  + Phân tích, thiết kế hệ thống:
* Mục đích xã hội: tham gia trò chơi, người chơi sẽ trắc nghiệm lại các kiến thức xã hội như lịch sử, địa lý, văn học, toán học… Một phần củng cố kiến thức xã hội cho người chơi, mặt khác đem lại cho họ cảm giác thoải mái và thư giãn khi chơi cùng bạn bè.

**Đối tượng tham gia**: Kiến thức xã hội là vô cùng tận, ai ai cũng muốn hiểu hơn về nó, các bạn trẻ là những người có lòng ham tìm hiểu, yêu công nghệ và giao lưu kết bạn, đó là những người chơi mà trò chơi hướng đến.

**Yêu cầu phía người chơi**: Để có thể tham gia trò chơi, người chơi cần trang bị cho mình một chiếc điện thoại cài đặt hệ điều hành Android. Điện thoại đã cài đặt phần mềm chơi. Khi đó, họ có thể chơi trò chơi mọi lúc mọi nơi.

**Các chức năng trò chơi cần xây dựng**

***Chức năng chơi offline***: Chương trình sẽ đưa ra 15 câu hỏi cho người chơi. Câu hỏi đưa ra dưới dạng trắc nghiệm và có 4 phương án cho người chơi chọn lựa. Các câu hỏi được sắp xếp với độ khó tăng dần. Nhiệm vụ của người chơi là lần lượt trả lời chúng. Nếu trả lời đúng, người chơi sẽ có quyền chơi tiếp, trả lời sai trò chơi sẽ dừng lại.

Trả lời câu hỏi: người chơi sẽ chọn một trong số 4 phương án do chương trình đưa ra. Khi đã chắc chắn đó là đáp án đúng, họ sẽ gửi nó đến chương trình. Chương trình sẽ gửi lại kết quả cho người chơi. Nếu đúng, chương trình sẽ hiển thị câu hỏi tiếp theo. Sai thì dừng lại

Thời gian chơi: người chơi có 30s suy nghĩ cho một câu hỏi. Nếu khi thời gian kết thúc mà người chơi vẫn chưa xác nhận chọn một đáp án cho mình thì trò chơi sẽ dừng lại.

Cấp độ của người chơi, chương trình đưa ra khái niệm điểm. Khởi đầu trò chơi, người chơi sẽ có 0 điểm. Mỗi khi trả lời đúng một câu hỏi, điểm sẽ được cộng dồn với số điểm của câu hỏi đó. Điểm cho mỗi câu hỏi được tính như sau:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Câu | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Điểm | 10 | 10 | 10 | 10 | 20 | 50 | 100 | 200 | 500 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Câu | 10 | 11 | 12 | 13 | 14 | 15 |
| Điểm | 1000 | 2000 | 3000 | 5000 | 8000 | 10000 |

Khi kết thúc trò chơi, chương trình sẽ tự động lưu lại điểm của người chơi đạt được và hiển thị 10 lần chơi với số điểm cao nhất.

Để hỗ trợ người chơi hoàn thành đến câu thứ 15, chương trình đưa ra 4 trợ giúp hỗ trợ người chơi trả lời câu hỏi và giúp người chơi dành được nhiều điểm số hơn. Người chơi chỉ có thể sử dụng các trợ giúp từ câu thứ 6 và mỗi trợ giúp chỉ được sử dụng một lần duy nhất. 4 trợ giúp là:

* Trợ giúp 50:50: Khi sử dụng quyền trợ giúp này, chương trình tự động loại bỏ 2 phương án chọn lựa. Do đó, người chơi chỉ cần chọn 1 trong 2 phương án còn lại.
* Trợ giúp giải thoát câu hỏi trừ 25% số điểm: Khi sử dụng quyền trợ giúp này, người chơi sẽ không cần trả lời câu hỏi hiện tại, tiếp tục trả lời câu hỏi tiếp theo đồng thời số điểm của người chơi bị trừ đi 25%, người chơi sẽ không dành được điểm của câu hỏi đó.
* Trợ giúp nhân đôi số điểm: nếu trả lời đúng câu hỏi và sử dụng quyền này, người chơi sẽ dành được 2 lần số điểm của câu hỏi đó.
* Trợ giúp đổi câu hỏi khác: khi sử dụng quyền trợ giúp này, chương trình sẽ thay câu hỏi hiện tại bằng một câu hỏi có cùng cấp độ tương ứng.

Câu hỏi sử dụng trong chương trình: các câu hỏi được trình bày dưới dạng văn bản với 4 phương án trả lời. Tất cả đều là câu hỏi trắc nghiệm và chỉ có 1 phương án là đúng. Những câu hỏi này được lấy từ cơ sở dữ liệu đã được lưu trong máy.

***Chức năng chơi online, một người chơi:***để chơi được chế độ này, yêu cầu điên thoại người chơi cần có kết nối internet. Người chơi cũng tham gia trả lời các câu hỏi do hệ thống cung cấp. Các câu hỏi được đưa ra với độ khó tăng dần và không giới hạn câu hỏi như trong chế độ offline. Trò chơi sẽ dừng lại khi người chơi không trả lời được câu hỏi.

Câu hỏi thuộc thể loại trắc nghiệm và được trình bày dưới dạng multimedia. Có câu hỏi thuộc dạng văn bản, có câu thuộc dạnh hình ảnh, âm thanh và video. Câu hỏi đều được lấy từ server của hệ thống và thông qua kết nối internet gửi đến người chơi.

Thời gian trả lời câu hỏi và cách thức trả lời: giống với chế độ offline, cũng với 30s cho mỗi câu hỏi và người chơi cần xác nhận phương án chọn lựa cuối cùng của mình cho hệ thống.

Xếp hạng người chơi: mỗi người chơi bắt đầu đều có 0 điểm. Điểm sẽ được cộng dần khi người chơi vượt qua các mức câu hỏi. Cách tính điểm cho các câu giống với trong chế độ offline, và từ câu 15 trở đi, điểm cho mỗi câu sẽ tăng dần theo cấp số cộng với bội số là 10000. Khi hoàn thành trò chơi, điểm của người chơi sẽ được cập nhật lên server và server sẽ hiện thị top người chơi có điểm số tốt nhất.

***Chức năng chơi online, nhiều người chơi:***

Đây là chức năng chính của chương trình, cho phép nhiều người chơi cùng tham gia thi đấu trực tuyến với nhau.

Để có thể chơi được chức năng này, yêu cầu người chơi cần có một tài khoản. Tài khoản này do người chơi đăng kí với hệ thống

* Đăng kí: người chơi sẽ đăng kí một tài khoản cho mình. Tài khoản này sẽ hiển thị trong trò chơi để phân biệt với những người khác. Tài khoản yêu cầu tên và mật khẩu. Tên chỉ được bao gồm các chữ cái thường, hoa và các chữ số, nó phải là duy nhất trong hệ thống (không trùng với những tên của người chơi khác). Mật khẩu không có dấu và không được bao gồm các kí tự đặc biệt như kí tự “ ’ ”, kí tự “/”, …
* Đăng nhập: Người chơi sẽ dùng tài khoản đã đăng kí để tham gia vào hệ thống. Nếu đăng nhập thành công thì họ mới có thể chơi được chức năng này.

Sau khi đăng nhập thành công, người chơi có 2 lựa chọn: tham gia một phòng chơi hoặc tự tạo ra một phòng chơi cho mình.

Khái niệm phòng chơi: là nơi cho một nhóm những người chơi tham gia thi đấu với nhau. Những người chơi trong phòng sẽ cùng trả lời lần lượt câu hỏi do hệ thống cung cấp nhằm tìm ra người thắng cuộc. Cách thức trả lời, các trợ giúp câu hỏi, cách trình bày câu hỏi giống với chức năng offline.

Nếu người chơi tham gia một phòng chơi, người chơi trở thành một thành viên trong phòng, khác với người tạo ra phòng chơi – là người chủ phòng. Chủ phòng có quyền từ chối tham gia phòng chơi của các thành viên, có quyền phát ra tín hiệu bắt đầu chơi. Khi tạo ra phòng chơi, chủ phòng đồng thời thiết lập các thông số cho phòng chơi bao gồm số lượng tối đa thành viên trong phòng, thời gian trả lời cho một câu hỏi và số điểm cược cho phòng. Trong quá trình chơi, chủ phòng không có quyền loại một người chơi ra khỏi phòng.

Khi trả lời xong một câu hỏi, chương trình sẽ hiện ra danh sách trả lời của các người chơi khác trong phòng. Nếu người chơi trả lời sai sẽ không có quyền chơi tiếp nhưng họ có lựa chọn xem tiếp những người chơi khác thi đấu. Trò chơi kết thúc khi tìm ra được người chiến thắng cuối cùng.

Điểm cược: mỗi khi tạo một tài khoản, người chơi sẽ có 1000 điểm trong tài khoản. Khi tham gia trò chơi, chủ phòng tạo phòng chơi đồng thời thiết lập điểm cược cho phòng chơi. Mỗi người chơi khi tham gia phòng thì điểm sẽ bị trừ đi một lượng bằng số điểm cược của phòng chơi. Tổng điểm cược trong phòng được tính bằng công thức ‘điểm cược \* số thành viên’. Điểm này sẽ là phần thưởng cho người chiến thắng trong phòng chơi.

Khi người chơi muốn tham gia phòng chơi mà số điểm của họ không đủ, họ cần tăng điểm của mình lên. Người chơi có thể tăng điểm bằng cách gửi tin nhắn đến chương trình…

Câu hỏi: Để đảm bảo đồng bộ thời gian cho những người chơi, câu hỏi trong phần này là những câu hỏi văn bản dạng text. Như vậy sẽ giảm được dung lượng và thời gian download câu hỏi cho các client, không phải phụ thuộc quá nhiều vào tốc độ mạng.

***Chức năng cập nhật câu hỏi cho chế độ chơi offline:*** chức năng này cung cấp cho người chơi cập nhật dữ liệu câu hỏi đã được cài đặt trong máy người chơi. Những câu hỏi này sẽ được sử dụng trong chế độ chơi offline.

***Chức năng gửi câu hỏi đến chương trình:*** cho phép người chơi gửi câu hỏi đến chương trình. Câu hỏi được gửi đến phải đảm bảo chận thực và chính xác, chúng sẽ được người quản lý kiểm tra lại.

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ CÔNG CỤ

## Android Programming



**Android là gì?**

Android là một phần mềm đi kèm với các thiết bị di động bao gồm operating system, middleware và application. Android OS được phát triển bởi Google dựa trên nền tảng Linux

**Lịch sử phát triển**

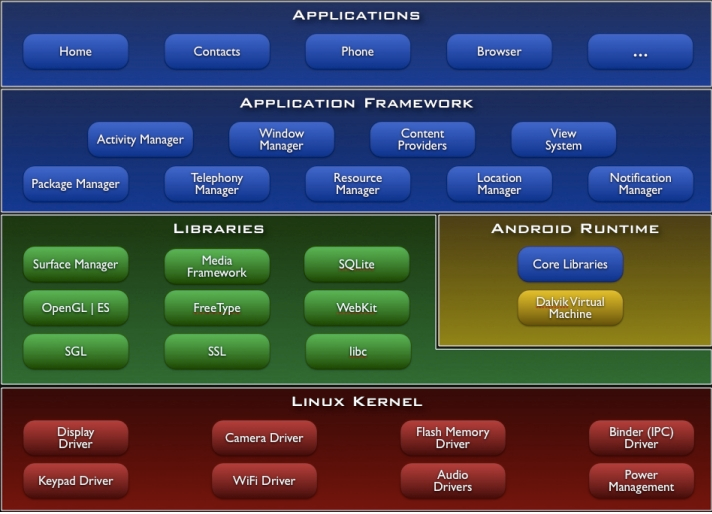
Tháng 7 năm 2005, Google mua lại Android, Inc., một công ty nhỏ mới thành lập có trụ sở ở Palo Alto, California, Mỹ. Những nhà đồng sáng lập của Android chuyển sang làm việc tại Google gồm có Andy Rubin (đồng sáng lập công ty Danger), Rich Miner (đồng sáng lập công ty Wildfire Communications), Nick Sears (từng là phó chủ tịch của T-Mobile), và Chris White (trưởng nhóm thiết kế và phát triển giao diện tại WebTV). Khi đó, có rất ít thông tin về các công việc của Android, ngoại trừ việc họ đang phát triển phần mềm cho điện thoại di động. Điều này tạo những tin đồn về việc Google có ý định bước vào thị trường điện thoại di động.

Tại Google, nhóm do Rubin đứng đầu đã phát triển một nền tảng thiết bị di động dựa trên hạt nhân Linux, được họ tiếp thị đến các nhà sản xuất thiết bị cầm tay và các nhà mạng trên những tiền đề về việc cung cấp một hệ thống mềm dẻo, có khả năng nâng cấp mở rộng cao. Một số nguồn tin cho biết trước đó Google đã lên danh sách các thành phần phần cứng và các đối tác phần mềm, đồng thời ra hiệu với các nhà mạng rằng họ sẵn sàng hợp tác ở nhiều cấp độ khác nhau. Ngày càng nhiều suy đoán rằng Google sẽ tham gia thị trường điện thoại di động xuất hiện trong tháng 12 năm 2006. Tin tức của BBC và Nhật báo phố Wall chú thích rằng Google muốn đưa công nghệ tìm kiếm và các ứng dụng của họ vào điện thoại di động và họ đang nỗ lực làm việc để thực hiện điều này. Các phương tiện truyền thông in và online cũng sớm có bài viết về những tin đồn cho rằng Google đang phát triển một thiết bị cầm tay mang thương hiệu Google. Và lại càng có nhiều suy đoán sau bài viết về việc Google đang định nghĩa các đặc tả công nghệ và trình diễn các mẫu thử với các nhà sản xuất điện thoại di động và nhà mạng.

Ngày 5 tháng 11 năm 2007, Liên minh thiết bị cầm tay mở rộng (Open Handset Alliance), một hiệp hội bao gồm nhiều công ty trong đó có Texas Instruments, Tập đoàn Broadcom, Google, HTC, Intel, LG, Tập đoàn Marvell Technology, Motorola, Nvidia, Qualcomm, Samsung Electronics, Sprint Nextel và T-Mobile được thành lập với mục đích phát triển các tiêu chuẩn mở cho thiết bị di động. Cùng với sự thành lập của OHA, họ cũng giới thiệu sản phẩm Android đầu tiên. Nó là một thiết bị di động có hệ điều hành dựa trên nhân Linux phiên bản 2.6.

Ngày 9 tháng 12 năm 2008, thêm 14 thành viên mới gia nhập dự án Android được công bố, gồm có ARM Holdings, Atheros Communications, Asustek Computer Inc, Garmin Ltd, Softbank, Sony Ericsson, Toshiba Corp, và Vodafone Group Plc.

**Kiến trúc Android OS**

**H1. Android Architecture**

Android OS được xây dựng từ 5 thành phần chính bao gồm:

* Applications: đây là những ứng dụng cơ bản mà Android cung cấp cho người dùng bao gồm: email client, SMS program, calendar, browser, contacts…. Tất cả những ứng dụng này đều sử dụng ngôn ngữ Java
* Application Framework: cung cấp nền tảng phát triển hoàn toàn mở, giúp cho nhà phát triển có thể xây dứng bất kì ứng dụng nào mà họ muốn: khai thác toàn bộ tài nguyên phần cứng, truy cập thông tin nội bộ, chạy các background services, thêm các notifications vào status bar và hơn nữa.

Tất cả những ứng dụng đều là thiết lập của các services và systems, bao gồm:

* + Views: xây dựng lên các ứng dụng như lists, grids, text boxes, buttons…
  + Content Providers: giúp ứng truy cập dữ liệu từ các ứng dụng khác (như Contacts), hay chia sẻ chính dữ liệu của chúng
  + Resource Manager: truy cập vào graphics, layout files, localized strings.
  + Notification Manager: giúp ứng dụng hiển thị các alerts trên status bar
  + Activity Manager: quản lý vòng đời của một ứng dụng
* Libraries: Android bao gồm một tập hợp thư viện C/C++ được sử dụng bởi nhiều thành phần trong hệ thống Android. Các thư viện cơ bản như: System C library, Media Libraries, Surface Manager, LibWebCore, SGL, 3D libraries, FreeType, SQLite
* Android Runtime: Dalvik virtual machine.
* Linux Kernel: nhân của Android là Linux 2.6 phục vụ bảo mật, quản lý bộ nhớ, quản lý tiến trình, mạng và driver model. Lõi này hoạt động như một tầng trừu tượng giữa phần cứng và phần mềm

**Các phiên bản của Android**

* 2.0/2.1 (Eclair): tân trang giao diện người dùng, hỗ trợ HTML5, W3C Geolocation API và Exchange ActiveSync 2.5
* 2.2 (Froyo): cải tiến về tốc độ cùng với sự tối ưu của JIT, Chrome V8 JavaScript engine, hỗ trợ thêm Wi-Fi hotspot tethering và Adobe Flash
* 2.3 (Gingerbread): tỉnh chỉnh giao diện, phát triển bàn phím mềm và tính năng copy/paste, thêm hỗ trợ cho Near Field Communication.
* 3.0/3.1 (Honeycomb): hỗ trợ màn hình lớn, thêm nhiều tính năng mới trong giao diện, hỗ trợ bộ xử lý đa lõi và tăng tốc phần cứng cho đồ họa
* 3.2 (Honeycomb): giống với 3.1 nhưng tối ưu cho 7 inch-tables

**Thành phần cơ bản tạo nên ứng dụng Android:**

* Activity: là nền của 1 ứng dụng. Khi khởi động 1 ứng dụng Android nào đó thì bao giờ cũng có 1 main Activity được gọi, hiển thị màn hình giao diện của ứng dụng cho phép người dùng tương tác.
* Service: thành phần chạy ẩn trong Android. Service sử dụng để update dữ liệu, đưa ra các cảnh báo (Notification) và không bao giờ hiển thị cho người dùng thấy.
* Content provider kho dữ liệu chia sẻ. Content Provider được sử dụng để quản lý và chia sẻ dữ liệu giữa các ứng dụng.
* Intent: nền tảng để truyền tải các thông báo. Intent được sử dụng để gửi các thông báo đi nhằm khởi tạo 1 Activity hay Service để thực hiện công việc bạn mong muốn. VD: khi mở 1 trang web, bạn gửi 1 intent đi để tạo 1 activity mới hiển thị trang web đó.
  1. Broadcast receiver thành phần thu nhận các Intent bên ngoài gửi tới. VD: bạn viết 1 chương trình thay thế cho phần gọi điện mặc định của Android, khi đó bạn cần 1 BR để nhận biết các Intent là các cuộc gọi tới.
  2. Notification đưa ra các cảnh báo mà không làm cho các Activity phải ngừng hoạt động.

**Vòng đời của ứng dụng Android**

Android có cơ chế quản lý các process theo chế độ ưu tiên. Các process có priority thấp sẽ bị Android giải phóng mà không hề cảnh báo nhằm đảm bảo tài nguyên.

***Foreground process****:* là process của ứng dụng hiện thời đang được người dùng tương tác.  
***Visible process****:* là process của ứng dụng mà activity đang hiển thị đối với người dùng (onPaused()của activity được gọi).

***Service process****:* là Service đang running.

***Background process****:* là process của ứng dụng mà các activity của nó ko hiển thị với người dùng (onStoped() của activity được gọi).

***Empty process****:* process không có bất cứ 1 thành phần nào active.  
Theo chế độ ưu tiên thì khi cần tài nguyên, Android sẽ tự động kill process, trước tiên là các empty process.

Actitvity là thành phần quan trọng nhất và đóng vai trò chính trong xây dựng ứng dụng Android. Hệ điều hành Android quản lý Activity theo dạng stack: khi một Activity mới được khởi tạo, nó sẽ được xếp lên đầu của stack và trở thành *running activity*, các Activity trước đó sẽ bị tạm dừng và chỉ hoạt động trở lại khi Activity mới được giải phóng.

Activity bao gồm 4 state:

- ***active (running):*** Activity đang hiển thị trên màn hình (foreground).  
- ***paused****:* Activity vẫn hiển thị (visible) nhưng không thể tương tác (lost focus). VD: một activity mới xuất hiện hiển thị giao diện đè lên trên activity cũ, nhưng giao diện này nhỏ hơn giao diện của activity cũ, do đó ta vẫn thấy được 1 phần giao diện của activity cũ nhưng lại không thể tương tác với nó.

- ***stop****:* Activity bị thay thế hoàn toàn bởi Activity mới sẽ tiến đến trạng thái *stop*  
- ***killed****:* Khi hệ thống bị thiếu bộ nhớ, nó sẽ giải phóng các tiến trình theo nguyên tắc ưu tiên. Các Activity ở trạng thái *stop* hoặc *paused* cũng có thể bị giải phóng và khi nó được hiển thị lại thì các Activity này phải khởi động lại hoàn toàn và phục hồi lại trạng thái trước đó.

  
**H2. Activity Life**

Vòng đời của Activity:

- *Entire lifetime:* Từ phương thức onCreate( ) cho tới onDestroy( ).

- *Visible liftetime:* Từ phương thức onStart( ) cho tới onStop( ).

- *Foreground lifetime:* Từ phương thức onResume( ) cho tới onPause( ).

## PHP & MySQL Programming

* 1. Giới thiệu PHP và môi trường lập trình web
     1. PHP là gì ?

Cái tên PHP ban đầu được viết tắt bởi cụm từ Personal Home Page, và được phát triển từ năm 1994 bởi Rasmus Lerdorf. Lúc đầu chỉ là một bộ đặc tả Perl, được sử dụng để lưu dấu vết người dùng trên các trang web. Sau đó, Rasmus Lerdorf đã phát triển PHP như là một máy đặc tả (Scripting engine). Vào giữa năm 1997, PHP đã được phát triển nhanh chóng trong sự yêu thích của nhiều người. PHP đã không còn là một dự án cá nhân của Rasmus Lerdorf và đã trở thành một công nghệ web quan trọng. Zeev Suraski và Andi Gutmans đã hoàn thiện việc phân tích cú pháp cho ngôn ngữ để rồi tháng 6 năm 1998, PHP3 đã ra đời (phiên bản này có phần mở rộng là \*.php3). Cho đến tận thời điểm đó, PHP chưa một lần được phát triển chính thức, một yêu cầu viết lại bộ đặc tả được đưa ra, ngay sau đó PHP4 ra đời (phiên bản này có phần mở rộng không phải là \*.php4 mà là \*.php). PHP4 nhanh hơn so với PHP3 rất nhiều. PHP bây giờ được gọi là PHP Hypertext PreProcesor.

* + 1. Tại sao phải sử dụng PHP

Như chúng ta đã biết, có rất nhiều trang web được xây dựng bởi ngôn ngữ HTML (HyperText Markup Language). Đây chỉ là những trang web tĩnh, nghĩa là chúng chỉ chứa đựng một nội dung cụ thể với những dòng văn bản đơn thuần, hình ảnh,và có thể được sự hỗ trợ bởi ngôn ngữ JavaScript, hoặc Java Apple. Những trang web như vậy người ta thường gọi là client-side. Tuy nhiên, Internet và Intranets đã được sử dụng cho các ứng dụng cần tới cơ sở dữ liệu. Các trang ứng dụng như vậy được gọi là trang web động, bởi vì nội dung của chúng luôn thay đổi tùy thuộc vào dữ liệu và người sử dụng. PHP là ngôn ngữ làm được những điều như vậy. Bằng cách chạy chương trình PHP trên máy chủ Web server, bạn có thể tạo ra các ứng dụng có sự tương tác với cơ sở dữ liệu để tạo ra những trang web và đây được gọi là trang web động.

Chúng ta hãy xem xét cách hoạt động của trang web được viết bằng ngôn ngữ HTML và PHP như thế nào.

Với các trang HTML :

|  |  |
| --- | --- |
| Khi có yêu cầu tới một trang web từ phía người sử dụng (browser). Web server thực hiệu ba bước sau :  + Đọc yêu cầu từ phía browser  + Tìm trang web trên server.  + Gửi trang web đó trở lại cho browser (nếu tìm thấy) qua mạng Internet hoặc Intranet . |  |

Với các trang PHP :

|  |  |
| --- | --- |
| Khác với các trang HTML, khi một trang PHP được yêu cầu, web server phân tích và thi hành các đoạn mã PHP để tạo ra trang HTML. Điều đó được thể hiện bằng bốn bước sau :  + Đọc yêu cầu tử phía browser.  + Tìm trang web trên server.  + Thực hiện các đoạn mã PHP trên trang web đó để sửa đổi nội dung của trang.  + Gửi trở lại nội dung cho browser (đây là trang HTML có thể hiển thị được bởi trình duyệt Internet Explorer hoặc trình duyệt nào đó). |  |

Tóm lại, sự khác nhau giữa HTML và PHP là HTML không được thực hiện trên máy chủ Web server còn các trang \*.php viết bằng các đoạn mã PHP được thực hiện trên máy chủ Web server do đó nó linh động và mềm dẻo hơn.

* 1. Giới thiệu hệ quản trị csdl MySQL
     1. MySQL là gì?

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

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

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

* + 1. Các ưu điểm của MySQL:

• MySQL là một hệ quản trị nhỏ, bảo mật, và rất dễ sử dụng, thường được sử dụng cho các ứng dụng nhỏ và trung bình. Nó được sử dụng cho các ứng dụng client / server với máy chủ mạnh như UNIX, Windows NT và Windows 95/98, và đặc biệt trên máy chủ UNIX .

• MySQL hỗ trợ các điểm vào là ANSI SQL92 và ODBC mức 0-2 SQL chuẩn.

• MySQL hỗ trợ nhiều ngôn ngữ cho việc thông báo lỗi như : Czec, Dutc, English, Estonian, French, German, Hungarian, Italian, Norwegian Nynorsk, Polish, Portuguese, Spanish and Swedish.

Ngôn ngữ được hỗ trợ mặc định cho dữ liệu là ISO-8859-1 (Latin1), muốn thay đổi phải sửa trong mã nguồn.

• Ngôn ngữ lập trình sử dụng viết các hàm API để thâm nhập cơ sở dữ liệu MySQL có thể là C, Perl, PHP.. .

• Các bảng (table) trong cơ sở dữ liệu MySQL có kích thước rất lớn và được lưu ở thư mục Datas. Kích thước lớn nhất của một bảng tối thiểu là 4GB và nó còn phụ thuộc và kích thước lớn nhất của một file do hệ điều hành quy định .

• Cơ sở dữ liệu MySQL rất dễ quản lý và có tốc độ xử lý cao hơn tới ba bốn lần so với các hệ quản trị cơ sở dữ liệu khác.

• MySQL là một hệ quản trị cơ sở dữ liệu mô hình quan hệ, nó có mã nguồn mở. Nó được cung cấp miễn phí trên các máy chủ UNIX, OS/2 và cả trên Windows.

* + 1. Nhược điểm của MySQL

Bên cạnh các ưu điểm trên MySQL cũng có một vài nhược điểm :

• MySQl không cho phép thực hiện các câu lệnh SQL select truy vấn con. Ví dụ : SELECT deptno, ename, sal

FROM emp x

WHERE sal > (SELECT AVG(sal)

FROM emp

WHERE x.deptno = deptno)

ORDER BY deptno

• MySQL không hỗ trợ Stored Procedures, Triggers, Transactions, Foreign Keys, và Views như các hệ quản trị cơ sở dữ liệu khác .

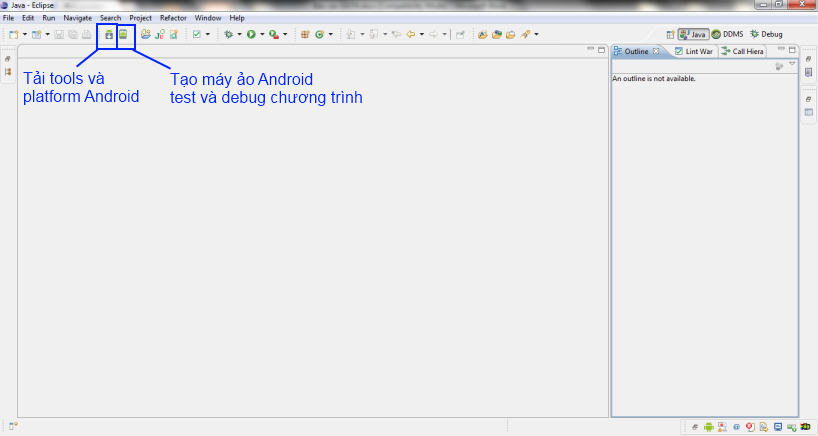
## IDE & Tools

***IDE lập trình Android***

* Môi trường lập trình: Window 7
* Công cụ lập trình: Eclipse kết hợp ADT plugin, Android SDK.
  + Android SDK sẽ cung cấp các platforms và các tools Android cho Eclipse.
  + ADT là cầu nối giữa Android SDK với Eclipse

Chi tiết cài đặt có thể xem tại [đây](http://developer.android.com/sdk/installing.html)

Hỉnh ảnh minh họa eclipse cài đặt ADT plugin và Android SDK.



***IDE lập trình PHP và MySQL***

* Môi trường lập trình: Window 7
* Công cụ: xampp🡪 xampp là gói phần mềm bao gồm apache server, php và mysql, tools thao tác cơ sở dữ liệu phpmyadmin.

PHPDesigner: IDE lập trình php

***Tools:***

* FileZilla Client: dùng để upload dữ liệu lên server.

# CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## Phân tích thiết kế chương trình

Các thành phần của hệ thống:

* Phần cứng:
  + Điện thoại cài đặt hệ điều hành Android: người dùng sẽ truy cập hệ thống trên điện thoại
  + Mạng Internet: cần có kết nối internet giữa điện thoại người dùng với hệ thống để có thể tải dữ liệu về máy
* Phần mềm:
  + Phần mềm hệ thống: được xây dựng dựa trên ứng dụng web cho phép quản lý ngân hàng câu hỏi, quản lý người dùng…
  + Phần mềm ứng dụng: được cài đặt trên điện thoại của người dùng. Người dùng sẽ thực hiện các thao tác trên phần mềm này như thi toeic, thi đại học, …
* Con người:
  + Người sử dụng thông thường: là những người sử dụng hệ thống với mục đích tham gia các bài thi test bằng việc sử dụng phần mềm ứng dụng được cài đặt vào điện thoại của họ
  + Người quản trị hệ thống: là người sẽ quản lý ngân hàng câu hỏi, thao tác thông qua phền mềm hệ thống.

### Phân tích biểu đồ usecase

Người sử dụng hệ thống: 2 loại người dùng

* User: là người sử dụng hệ thống với mục đích tham gia trò chơi
* Admin: phụ trách quản lý người dùng và quản lý ngân hàng câu hỏi
  + 1. Biểu đồ usecase của User



* 1. Chơi đấu trường kiến thức offline:

Người dùng sẽ được tham gia trò chơi đấu trường kiến thức được xây dựng dựa trên game show truyền hình “Ai là triệu phú”. Nhiệm vụ của người chơi là trả lời các câu hỏi trắc nghiệm do chương trình đưa ra.

Câu hỏi trắc nghiệm với 4 phương án trả lời trong đó chỉ có một đáp án đúng. Câu hỏi được lấy từ cơ sở dữ liệu được lưu cục bộ trong chương trình.

* 2. Đăng kí:

Tạo tài khoản mới cho người dùng. Người dùng cần đăng kí một tài khoản để có thể tham gia trò chơi dưới hình thức online. Tài khoản sẽ đại diện cho người dùng và hiển thị trên mạng.

* 3. Đăng nhập:

Người dùng sử dụng tài khoản đã đăng kí để đăng nhập vào hệ thống. Mỗi người dùng có ít nhất một tài khoản. Hệ thống yêu cầu người dùng có một tài khoản để phân biệt giữa các người chơi với nhau.

* 4. Chơi đấu trường kiến thức online / single player

Chức năng chơi này giống với chế độ chơi offline, tuy nhiên các câu hỏi cho người chơi được lấy từ server của hệ thống không như chế độ offline, câu hỏi được lấy từ cơ sở dữ liệu cục bộ được cài đặt trong máy người chơi.

* 5. Chơi đấu trường kiến thức online / multiplayer

Đây là chức năng chính của trò chơi. Người chơi sẽ không thi đấu một mình mà sẽ tham gia cùng nhiều người chơi khác. Tất cả các người chơi sẽ trả lời lần lượt các câu hỏi do hệ thống đưa ra để tìm ra người chiến thắng.

Người chơi trả lời sai sẽ bị loại khỏi trò chơi. Người chiến thắng là người duy nhất còn lại.

* 6. Download câu hỏi chơi offline

Trong chế độ chơi offline, các câu hỏi do chương trình cung cấp bị giới hạn về số lượng, không cập nhật theo thực tế. Vì vậy, hệ thống cung cấp các bộ câu hỏi cho người dùng tải về máy cập nhật vào cơ sở dữ liệu cục bộ của chương trình.

* 7. Gửi câu hỏi đến chương trình

Cho phép người dùng soạn ra các câu hỏi và gửi tới hệ thống. Các câu hỏi này sẽ chưa được cập nhật ngay vào cơ sở dữ liệu của hệ thống mà sẽ được người quản lý kiểm duyệt. Nếu hợp lệ mới được lưu vào cơ sở dữ liệu.

* + 1. Biểu đồ usecase cho Admin



* 1. Quản lý người dùng:
  + 1.1. Thêm / xóa người dùng:

Người quản trị có quyền thêm mới và xóa người dùng đã đăng kí. Có chức năng xóa người dùng nhằm loại bỏ những người dùng không hợp lệ hoặc đã lâu không hoạt động.

* + 1.2. Phân quyền cho người dùng:

Nâng cấp quyền cho người dùng thông thường thành người quản trị hệ thống tham gia quản lý câu hỏi, quản lý người dùng, …

Giảm quyền cho người người quản lý xuống thành người dùng bình thường.

* Quản lý phòng chơi

Như đã nêu trong phần 1.2 chức năng Đấu trường kiến thức online chế độ nhiều người chơi, sẽ có phòng chơi public do hệ thống tạo ra. Do đó cần có người quản lý những phòng chơi này. Quản lý về số lượng phòng chơi trong một thời điểm, thời gian chơi của phòng chơi public, hay quản lý câu hỏi cho phòng chơi này…

* Quản lý câu hỏi
  + *Thêm / sửa / xóa câu hỏi:*

Cho phép người quản lý thêm, sửa và xóa các câu hỏi.

* + *Duyệt câu hỏi từ người dùng gửi đến:*

Các câu hỏi từ người dùng gửi đến chưa được xác thực hoặc cũng có những sai xót cần người quản lý chỉnh sửa và kiểm tra.

### Phân tích biểu đồ luồng dữ liệu

* + 1. Chức năng chơi đấu trường kiến thức offline



|  |  |
| --- | --- |
| Xem câu hỏi | |
| Đầu vào | Đầu ra |
| Tín hiệu tham gia từ người chơi, dữ liệu câu hỏi từ cơ sở dữ liệu offline | Nội dung câu hỏi và các phương án trả lời hiển thị cho người chơi |
| Trả lời câu hỏi | |
| Đầu vào | Đầu ra |
| Phương án chọn lựa của người chơi | Xác nhận phương án chọn lựa của người chơi |
| Xem kết quả | |
| Đầu vào | Đầu ra |
| Đáp án của câu hỏi lấy từ cơ sở dữ liệu | Hiển thị kết quả cho user. Nếu kết quả:   * Đúng: hiển thị câu hỏi tiếp cho người chơi. * Sai:lưu điểm user vào dữ liệu bảng xếp hạng. |
| Xem xếp hạng | |
| Đầu vào | Đầu ra |
| Dữ liệu xếp hạng từ cơ sở dữ liệu | Hiển thị danh sách xếp hạng của người chơi |

Chức năng đăng kí



User cần nhập tài khoản đăng kí. Client kiểm tra dữ liệu đầu vào, nếu hợp lệ sẽ gửi tài khoản của user cho server. Server kiểm tra tài khoản đó bằng cách so sánh với những user tồn tại trong cơ sở dữ liệu. Nếu thông tin tài khoản là hợp lệ (chưa tồn tại trong cơ sở dữ liệu) server lưu tài khoản user vào cơ sở dữ liệu rồi gửi lại kết quả cho client. Client sẽ hiển thị kết quả thông báo cho user.

|  |  |
| --- | --- |
| Đăng kí | |
| Đầu vào | Đầu ra |
| * Thông tin user * Danh sách user trong hệ thống | * Lưu trử user vào cơ sở dữ liệu nếu user hợp lệ * Thông báo kết quả cho user |

* + 1. Chức năng đăng nhập



Để đăng nhập, User sẽ gửi tài khoản cho hệ thống. Hệ thống sẽ lấy về thông tin các người dùng rồi so sánh với tài khoản của người dùng. Kết quả sẽ được gửi lại cho User

|  |  |
| --- | --- |
| Đăng nhập | |
| Đầu vào | Đầu ra |
| * Thông tin user * Danh sách user trong hệ thống | * Kết quả hiển thị cho user |

* + 1. Chức năng chơi đấu trường kiến thức online / single-player



|  |  |
| --- | --- |
| Xem câu hỏi | |
| Đầu vào | Đầu ra |
| * Yêu cầu xem câu hỏi từ user * Dữ liệu câu hỏi lấy từ csdl | * Kết quả hiển thị cho user |
| Trả lời câu hỏi | |
| Đầu vào | Đầu ra |
| * Thông tin câu hỏi * Phương án chọn lựa của user | * Xác nhận phương án chọn lựa của người chơi |
| Xem kết quả | |
| Đầu vào | Đầu ra |
| * Yêu cầu xem kết quả của user * Đáp án của câu hỏi | * Kết quả hiển thị cho user * Điểm người chơi lưu vào csdl |
| Xem bảng xếp hạng | |
| Đầu vào | Đầu ra |
| * Yêu cầu xem xếp hạng từ user * Dữ liệu bảng xếp hạng lấy từ csdl | * Kết quả hiển thị cho user |

* + 1. Chức năng chơi đấu trường kiến thức online / multi-player





Người chơi sẽ gửi tín hiệu xem danh sách phòng chơi cho chương trình. Chương trình sẽ gửi lại cho người chơi thông tin các phòng chơi đang có. Dựa vào đấy, người chơi cần chọn một phòng chơi mà mình muốn tham gia. Hệ thống sẽ cập nhật thông tin phòng chơi đó và gửi lại cho người chơi những người chơi trong phòng. Khi đã vào phòng, nếu người chơi không muốn tham gia có thể thoát khỏi phòng chơi. Trò chơi bắt đầu, tất cả người chơi sẽ trả lời một câu hỏi do chương trình đưa ra. Nhiệm vụ của người chơi là tìm ra phương án chính xác. Khi hết thời gian, chương trình sẽ gửi cho người chơi kết quả trả lời của mình và những người chơi khác trong phòng. Nếu có nhiều người chơi trả lời đúng, trò chơi sẽ tiếp tục, họ sẽ trả lời tiếp các câu hỏi của chương trình. Nếu người chơi trả lời sai sẽ bị loại khỏi trò chơi. Trò chơi kết thúc khi tìm ra được người chiến thắng là người cuối cùng còn lại trong phòng.

|  |  |
| --- | --- |
| Xem danh sách phòng chơi | |
| Đầu vào | Đầu ra |
| * Yêu cầu xem danh sách phòng chơi từ user * Thông tin các phòng chơi lấy từ csdl | * Hiển thị danh sách phòng chơi cho user |

|  |  |
| --- | --- |
| Tham gia phòng chơi | |
| Đầu vào | Đầu ra |
| * Phòng chơi chọn từ user * Thông tin phòng chơi | * Hiển thị kết quả cho user * Lưu user vào dữ liệu |

Trong chức năng này, hệ thống dựa vào thông tin phòng chơi của người chơi chọn, kiểm tra xem user có khả năng tham gia phòng chơi không, kiểm tra phòng chơi còn trống không. Nếu thỏa mãn, người chơi có đủ điều kiện tham gia phòng chơi, nếu không sẽ thông báo không thể tham gia phòng chơi này cho user.

|  |  |
| --- | --- |
| Thoát khỏi phòng chơi | |
| Đầu vào | Đầu ra |
| * Mã người chơi trong phòng lấy từ csdl * Yêu cầu thoát từ user | * Hiển thị kết quả thông báo cho user |

|  |  |
| --- | --- |
| Bắt đầu chơi | |
| Đầu vào | Đầu ra |
| * Mã người chơi trong phòng lấy từ csdl * Yêu cầu thoát từ user | * Hiển thị kết quả thông báo cho user * Cập nhật dữ liệu người chơi trong phòng |

|  |  |
| --- | --- |
| Xem người chơi trong phòng | |
| Đầu vào | Đầu ra |
| * Yêu cầu xem danh sách người chơi * Danh sách người chơi từ csdl | * Hiển thị kết quả thông báo cho user |

|  |  |
| --- | --- |
| Xem câu hỏi | |
| Đầu vào | Đầu ra |
| * Yêu cầu xem câu hỏi từ user * Thông tin câu hỏi lấy từ csdl | * Hiển thị kết quả cho user |

|  |  |
| --- | --- |
| Trả lời câu hỏi | |
| Đầu vào | Đầu ra |
| * Phương án chọn từ user | * Phương án chọn từ user lưu vào csdl |

|  |  |
| --- | --- |
| Xem kết quả | |
| Đầu vào | Đầu ra |
| * Yêu cầu xem kết quả từ user * Danh sách trả lời của các người chơi lấy từ csdl * Đáp án cho câu hỏi lấy từ dữ liệu câu hỏi | * Hiển thị kết quả cho user * Điểm người chơi lưu vào dữ liệu người chơi trong phòng |

Kết thúc chức năng này, nếu trò chơi vẫn tiếp tục, user sẽ quay lại xem câu hỏi tiếp theo và thực hiện các bước như trên, nếu trò chơi kết thúc sẽ yêu cầu xem điểm số của người chơi.

|  |  |
| --- | --- |
| Xem điểm số | |
| Đầu vào | Đầu ra |
| * Yêu cầu xem điểm từ user * Thông tin user lấy từ dữ liệu người chơi | * Hiển thị kết quả cho user |

* + 1. Chức năng download câu hỏi offline



Các bộ câu hỏi là những bản cập nhật dữ liệu offline của chương trình cho người chơi. Để có thể download những bản cập nhật này, người chơi cần xem danh sách những bộ câu hỏi và chọn bộ mà mình muốn tải về. Khi hoàn tất download các câu hỏi, chương trình sẽ tự động cập nhật vào cơ sở dữ liệu cục bộ của người chơi.

|  |  |
| --- | --- |
| Xem danh sách các bộ câu hỏi | |
| Đầu vào | Đầu ra |
| * Yêu cầu xem danh sách bộ câu hỏi từ user * Thông tin các bộ câu hỏi lấy từ cơ sở dữ liệu | * Hiển thị kết quả cho user |

|  |  |
| --- | --- |
| Download bộ câu hỏi | |
| Đầu vào | Đầu ra |
| * Bộ câu hỏi người chơi chọn * Mã bộ câu hỏi lấy từ csdl * Dữ liệu câu hỏi từ csdl | * Hiển thị kết quả cho user |

|  |  |
| --- | --- |
| Cập nhật dữ liệu offline | |
| Đầu vào | Đầu ra |
| * Yêu cầu cập nhật dữ liệu từ user | * Dữ liệu câu hỏi lưu vào csdl offline |

* + 1. Chức năng gửi câu hỏi đến chương trình



Người chơi sẽ tự soạn câu hỏi và gửi đến cho chương trình. Chương trình sẽ cập nhật nó vào cơ sở dữ liệu vào gửi thông báo lại cho người chơi

|  |  |
| --- | --- |
| Gửi câu hỏi đến chương trình | |
| Đầu vào | Đầu ra |
| * Thông tin câu hỏi từ user | * Dữ liệu câu hỏi lưu vào csdl * Kết quả thông báo cho user |

### Phân tích biểu đồ thực thể liên kết và thiết kế dữ liệu



* + 1. Bảng users

Lưu thông tin của người dùng hệ thống

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên cột | Kiểu dữ liệu | Null | Ràng buộc | Mô tả |
| 1. | user\_id | int |  | PK, identity | Mã người dùng, khóa chính, tự tăng |
| 2. | username | varchar(50) |  |  | Tên người dùng |
| 3. | password | varchar(50) |  |  | Mật khẩu người dùng |
| 4. | email | varchar(100) |  |  | Email người dùng |
| 5. | money | float |  |  | Tiền của người dùng |
| 6. | score | float |  |  | Điểm người dùng |
| 7. | register\_date | datetime |  |  | Ngày đăng kí tham gia hệ thống |
| 8. | last\_join\_date | datetime |  |  | Thời gian lần cuối tham gia hệ thống |
| 9. | status | tinyint |  |  | Phân biệt giữa người dùng và admin. Giá trị   * 0: user * 1: admin |

* + 1. Bảng rooms

Lưu dữ liệu của các phòng chơi

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên cột | Kiểu dữ liệu | Null | Ràng buộc | Mô tả |
| 1. | room\_id | int |  | PK, identity | Mã phòng, khóa chính, tự động tăng |
| 2. | room\_name | varchar(100) |  |  | Tên phòng |
| 3. | owner\_id | int |  | FK | Mã người dùng tạo ra phòng chơi, tham chiếu tới bảng users |
| 4. | max\_member | int |  |  | Số lượng người chơi tối đa trong phòng |
| 5. | number\_of\_members | int |  |  | Số lượng người chơi hiện tại trong phòng |
| 6. | bet\_score | float | x |  | Số điểm đặt cọc cho của phòng |
| 7. | time\_per\_question | tinyint |  |  | Thời gian trả lời cho một câu hỏi |
| 8. | status | tinyint |  |  | Trạng thái của phòng chơi. Giá trị:   * 0: Phòng đợi * 1: Đang chơi |

* + 1. Bảng room\_members

Lưu dữ liệu người chơi trong phòng chơi

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên cột | Kiểu dữ liệu | Null | Ràng buộc | Mô tả |
| 1. | room\_member\_id | int |  | PK, identity | Mã người dùng trong phòng chơi. Khóa chính, tự động tăng |
| 2. | user\_id | int |  | FK | Mã người dùng, tham chiếu tới bảng users |
| 3. | room\_id | int |  | FK | Mã phòng chơi, tham chiếu tới bảng rooms |
| 4. | member\_type | tinyint |  |  | Loại người chơi:   * 0: member * 1: room’s owner |
| 5. | question\_id | int |  | FK | Mã câu hỏi hiện tại, tham chiếu tới bảng questions |
| 6. | score | float |  |  | Điểm của người chơi |
| 7. | last\_answer | tinyint |  |  | Câu trả lời cho câu hỏi hiện tại của người chơi |
| 8. | help | varchar | x |  | Các trợ giúp người chơi đã sử dụng |
| 9. | status | tinyint |  |  | Trạng thái người chơi trong phòng. Giá trị:   * 0: chưa sẵn sàng chơi * 1: sẵn sàng chơi |

* + 1. Bảng questions

Lưu dữ liệu câu hỏi cho hệ thống

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên cột | Kiểu dữ liệu | Null | Ràng buộc | Mô tả |
| 1. | question\_id | int |  | PK, identity | Mã câu hỏi, khóa chính, tự tăng |
| 2. | level\_id | tinyint |  | FK | Level của câu hỏi, tham chiếu tới bảng levels |
| 3. | question\_name | nvarchar(500) |  |  | Câu hỏi |
| 4. | answer\_a | nvarchar(500) |  |  | Phương án A |
| 5. | answer\_b | nvarchar(500) |  |  | Phương án B |
| 6. | answer\_c | nvarchar(500) |  |  | Phương án C |
| 7. | answer\_d | nvarchar(500) |  |  | Phương án D |
| 8. | answer | tinyint |  |  | Đáp án đúng |
| 9. | describle\_answer | nvarchar(500) | x |  | Diễn giải đáp án |

* + 1. Bảng question\_levels

Lưu dữ liệu về cấp độ của các câu hỏi

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên cột | Kiểu dữ liệu | Null | Ràng buộc | Mô tả |
| 1. | level\_id | tinyint |  | PK, identity | Mã level, khóa chính, tự động tăng |
| 2. | level\_name | nvarchar(50) |  |  | Tên level |
| 3. | level\_value | tinyint |  |  | Giá trị của level, là các số như 0, 1, 2, … |
| 4. | score | float |  |  | Điểm cho mỗi mức level |

* + 1. Bảng categories

Lưu dữ liệu về đề thi

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên cột | Kiểu dữ liệu | Null | Ràng buộc | Mô tả |
| 1. | category\_id | int |  | PK, identity | Mã của đề thi, khóa chính, tự tăng |
| 2. | subject\_name | nvarchar(200) |  |  | Tên đề thi |
| 3. | describle\_subject | nvarchar(500) |  |  | Miêu tả về đề |
| 4. | number\_of\_question | int |  |  | Số lượng câu hỏi trong đề thi |
| 5. | date\_create | datetime |  |  | Thời gian tạo đề thi |

* + 1. Bảng category\_questions

Lưu dữ liệu các câu hỏi trong đề

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên cột | Kiểu dữ liệu | Null | Ràng buộc | Mô tả |
| 1. | category\_question\_id | int |  | PK, identity | Mã câu hỏi trong một đề thi, khóa chính, tự tăng |
| 2. | category\_id | int |  | FK | Mã đề thi, tham chiếu bảng subjects |
| 3. | question\_id | int |  | FK | Mã câu hỏi, tham chiếu bảng questions |

* + 1. Bảng category\_questions

Lưu thành tích của người chơi

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên cột | Kiểu dữ liệu | Null | Ràng buộc | Mô tả |
| 1. | award\_id | int |  | PK, identity | Mã thành tích trong một đề thi, khóa chính, tự tăng |
| 2. | username | varchar(100) |  |  | Tên người chơi |
| 3. | score | float |  |  | Điểm người chơi đạt được |
| 4. | date\_record | datetime |  |  | Thời gian ghi nhật thành tích |

## Thiết kế chương trình

### Giao tiếp giữa client ⬄ server

Game trực tuyến được xây dựng theo mô hình client ⬄ server, trong đó, người chơi đóng vai trò là client yêu cầu dữ liệu từ phía server, server sẽ trả lại dữ liệu cho client xử lý và hiển thị. Dữ liệu gửi nhận giữa client và server đóng vai trò là phương thức giao tiếp và được truyền tải qua mạng.

Có nhiều kiểu định dạng dữ liệu nhưng trong chương trình, chúng ta sử dụng kiểu JSON. Đặc điểm quan trọng của nó là gọn nhẹ, dễ truyền tải và đều được hỗ trợ bởi Java và PHP.

Chi tiết về ngôn ngữ JSON có thể xem tại [đây](http://www.json.org/json-vi.html)

Request từ client đến server, dữ liệu được thêm vào trong request của giao thức HTTP dưới phương thức POST:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Message | Request (HTTP-POST) | | Response (JSON) | |
| **Name** | **Value** |  |  |
| Đăng kí | message | “register” | {  “type”: “register”  “value”: “true” (or “false”)  “message”: “…”  “info”: [  {“user\_id”= “…”, “score\_level” = “…”, “money”= “…”}  ]  } | |
| username | “value” |
| password | “value” |
| email | “value” |
| Đăng nhập | message | “login” | {  “type”: “login”  “value”: “true” (or “false”)  “message”: “…”  “info”: [  {“user\_id”= “…”, “score\_level” = “…”, “money”= “…”}  ]  } | |
| username | “value” |
| password | “value” |
| Xem danh sách phòng chơi | message | “get\_list\_room” | {  “type”: “get\_list\_room”  “value”: “true” (or “false”)  “message”: “…”  “info”: [  {“room\_id”= “…”, “room\_name” = “…”, “owner\_id”= “…”,  “username”= “…”,  “max\_member” = “…”,  “bet\_score”= “…”,  “time\_per\_question”= “..”,  },  {  …  }  ]  } | |
| Tham gia phòng chơi | message | “join\_room” | {  “type”: “join\_room”  “value”: “true” (or “false”)  “message”: “…”  “member\_id”: “value”  } | |
| room\_id | “value” |
| user\_id | “value” |
| Tạo phòng chơi | message | “create\_room” |  | |
| room\_name |  |
| owner\_id |  |
| max\_member |  |
| bet\_score |  |
| time\_per\_question |  |
| Xóa phòng chơi | message | “remove\_room” |  | |
| room\_id |  |
| Thoát phòng chơi | message | “exit\_room” |  | |
| member\_id |  |
| room\_id |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Response từ server đến client là gói dữ liệu định dạng theo kiểu JSON:

### Thiết kế giao diện

# CHƯƠNG 4: CÀI ĐẶT THỬ NGHIỆM VÀ TRIỂN KHAI HỆ THỐNG

## Cài đặt chương trình

## Thử nghiệm chương trình

## Triển khai chương trình

# KẾT LUẬN

## Kết quả đánh giá

## Định hướng phát triển

## Kết luận

# TÀI LIỆU THAM KHẢO

# DANH MỤC HÌNH VẼ