 🙠🙟🕮🙝🙢

**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

[](http://images.google.com/imgres?imgurl=http://sites.google.com/site/truongdhbk/_/rsrc/1241529856195/config/app/images/LogoBKchuan.JPG&imgrefurl=http://sites.google.com/site/truongdhbk/chuyen-cuoi-vova&usg=__ysJd5eEVvlR1GjkbQO85K2rzgEQ=&h=1024&w=683&sz=58&hl=en&start=1&um=1&tbnid=GIEkAr2LNia2LM:&tbnh=150&tbnw=100&prev=/images?q=logo+b%C3%A1ch+khoa&hl=en&rls=com.microsoft:en-us&sa=N&um=1)

**ĐỒ ÁN**

**TỐT NGHIỆP ĐẠI HỌC**

**NGHÀNH CÔNG NGHỆ THÔNG TIN**

**ĐỀ TÀI:**

**Xây dựng ứng dụng đọc truyện trên hệ điều hành android sử dụng**

**Web Service**

Sinh viên thực hiện **: Đặng Anh Tuấn**

Lớp**: Truyền thông và mạng máy tính –K52**

Giảng viên hướng dẫn**:Th.S.Bành Thị Quỳnh Mai**

***Hà Nội,tháng 5/2014***

***Lời cảm ơn***

*Để có ngày hoàn thành đồ án tốt nghiệp này, lời đầu tiên con xin chân thành cảm ơn bố mẹ kính yêu đã tạo điều kiện cho con ăn học, kính chúc bố mẹ luôn luôn mạnh khoẻ.*

*Em xin cảm ơn cô giáo – Thạc sỹ Bành Thị Quỳnh Mai đã chỉ bảo, giúp đỡ, tạo điều kiện cho em hoàn thành tốt đồ án này.*

*Cuối cùng em xin cám ơn các thầy cô trong Viện đã giúp đỡ em trong suốt các năm theo học tại Viện, dạy bảo những kiến thức để làm nền tảng cho Đồ án này.*

***Hà Nội, ngày 20 tháng 5 năm 2014***

***Sinh viên thực hiện:***

***Đặng Anh Tuấn***

Contents

[CHƯƠNG I: ĐẶT VẤN ĐỀ 7](#_Toc388644902)

[1.1.Khảo sát thực tế 7](#_Toc388644903)

[1.2.Nội dung đề tài 7](#_Toc388644904)

[1.3.Bố cục đồ án 8](#_Toc388644905)

[Chương II: CƠ SỞ LÝ THUYẾT VÀ ĐỊNH HƯỚNG GIẢI PHÁP 9](#_Toc388644906)

[2.1.Giới thiệu hệ điều hành Android 9](#_Toc388644907)

[2.2.Kiến trúc Android 10](#_Toc388644908)

[2.3.Phát triển ứng dụng trên Android 13](#_Toc388644909)

[2.3.1.Ngôn ngữ lập trình 13](#_Toc388644910)

[2.3.2.Vấn đề phân mảnh phiên bản Android trên thị trường: 14](#_Toc388644911)

[2.3.3.Vấn đề kích thước màn hình 16](#_Toc388644912)

[2.3.4.Môi trường lập trình cho Android 16](#_Toc388644913)

[2.3.5.Các thành phần cơ bản của một project Android trên Eclipse 16](#_Toc388644914)

[2.4.Khái niệm Web Service 18](#_Toc388644915)

[2.5.Đặc điểm của Web Service 18](#_Toc388644916)

[2.6.RESTfull Web Service 18](#_Toc388644917)

[2.6.1. Sử dụng các phương thức HTTP một cách rõ ràng 19](#_Toc388644918)

[2.6.2. Phi trạng thái 19](#_Toc388644919)

[2.6.3. Đưa ra cấu trúc thư mục giống URIs 20](#_Toc388644920)

[2.6.4. Trao đổi XML, JSON, hoặc cả hai 21](#_Toc388644921)

[2.7. HTTP Progressive Download 21](#_Toc388644922)

[2.8.Định hướng giải pháp và lựa chọn công cụ xây dựng 22](#_Toc388644923)

[CHƯƠNG III: XÂY DỰNG ỨNG DỤNG ĐỌC TRUYỆN 24](#_Toc388644924)

[3.1.Phân tích các chức năng 24](#_Toc388644925)

[3.2.Biểu đồ phân cấp chức năng 25](#_Toc388644926)

[3.2.1. Biểu đồ 25](#_Toc388644927)

[3.2.2. Đặc tả chức năng 26](#_Toc388644928)

[3.3.Biểu đồ Use case 27](#_Toc388644929)

[3.3.1. Use case hiển thị danh sách truyện 27](#_Toc388644930)

[3.3.2. Use case hiển thị danh sách các chương của truyện 28](#_Toc388644931)

[3.3.3. Use case đọc truyện 28](#_Toc388644932)

[3.3.4. Use case bình luận các chương của truyện 29](#_Toc388644933)

[3.3.5. Use case yêu thích truyện 29](#_Toc388644934)

[3.3.6. Use case đăng nhập 30](#_Toc388644935)

[3.3.7. Use case đăng ký tài khoản 30](#_Toc388644936)

[3.3.8. Use case khóa tài khoản 31](#_Toc388644937)

[3.3.9. Use case mở khóa tài khoản 31](#_Toc388644938)

[3.3.10. Use case xóa comment 31](#_Toc388644939)

[3.4.Biểu đồ tuần tự 32](#_Toc388644940)

[3.4.1. Biểu đồ tuần tự chức năng Show List Story 32](#_Toc388644941)

[3.4.2. Biểu đồ tuần tự chức năng Show List Chapter 33](#_Toc388644942)

[3.4.3. Biểu đồ tuần tự chức năng Read Chapter 33](#_Toc388644943)

[3.4.4. Biểu đồ tuần tự chức năng Comment 35](#_Toc388644944)

[3.4.5. Biểu đồ tuần tự chức năng Vote Story 36](#_Toc388644945)

[3.4.6. Biểu đồ tuần tự chức năng Login 37](#_Toc388644946)

[3.4.7. Biểu đồ tuần tự chức năng Register new account 38](#_Toc388644947)

[3.4.8. Biểu đồ tuần tự chức năng Block Account 39](#_Toc388644948)

[3.4.9. Biểu đồ tuần tự chức năng Active Account 39](#_Toc388644949)

[3.4.10. Biểu đồ tuần tự chức năng Delete Comment 40](#_Toc388644950)

[3.5.Biểu đồ lớp 40](#_Toc388644951)

[3.5.1.Biểu đồ lớp phía client 40](#_Toc388644952)

[3.5.2.Biểu đồ lớp phía Server 42](#_Toc388644953)

[3.6.Biểu đồ hoạt động 43](#_Toc388644954)

[3.6.1.Biểu đồ hoạt động cho chức năng xem List Story 43](#_Toc388644955)

[3.6.2.Biểu đồ hoạt động cho chức năng xem List Chapter 44](#_Toc388644956)

[3.6.3.Biểu đồ hoạt động cho chức năng Read Chapter 44](#_Toc388644957)

[3.6.4.Biểu đồ hoạt động cho chức năng Comment 46](#_Toc388644958)

[3.6.5.Biểu đồ hoạt động cho chức năng Vote Story 47](#_Toc388644959)

[3.6.6.Biểu đồ hoạt động cho chức năng Login 48](#_Toc388644960)

[3.6.7.Biểu đồ hoạt động cho chức năng Register new account 48](#_Toc388644961)

[3.6.8.Biểu đồ hoạt động cho chức năng Block account 49](#_Toc388644962)

[3.6.9.Biểu đồ hoạt động cho chức năng Active account 50](#_Toc388644963)

[3.6.10.Biểu đồ hoạt động cho chức năng Delete Comment 50](#_Toc388644964)

[3.7.Thiết kế CSDL 51](#_Toc388644965)

[CHƯƠNG IV: TRIỂN KHAI ỨNG DỤNG 55](#_Toc388644966)

[4.1.Môi trường triển khai 55](#_Toc388644967)

[4.2. Cách thức triển khai và sử dụng hệ thống 55](#_Toc388644968)

[4.3. Kịch bản demo 55](#_Toc388644969)

[CHƯƠNG V: KẾT LUẬN 59](#_Toc388644970)

[5.1.Các kết quả đạt được 59](#_Toc388644971)

[5.1.1.Ưu điểm 59](#_Toc388644972)

[5.1.2.Nhược điểm 59](#_Toc388644973)

[5.2.Hướng phát triển trong tương lai 59](#_Toc388644974)

[**TÀI LIỆU THAM KHẢO** 60](#_Toc388644975)

[**MỤC LỤC HÌNH VẼ, BẢNG BIỂU** 60](#_Toc388644976)

# CHƯƠNG I: ĐẶT VẤN ĐỀ

# 1.1.Khảo sát thực tế

Đọc truyện, sách báo là một trong những nhu cầu giải trí và tiếp thu thông tin rất tốt. Nhu cầu đọc truyện, sách cũng rất lớn .Có rất nhiều website đọc sách, truyện miễn phí như : sstruyen.com,isach.info… để thỏa mãn nhu cầu đọc truyện,sách trên internet. Hiện nay đã có nhiều phần mềm cho phép đọc truyện trên điện thoại thông minh , nhưng vẫn còn ít ứng dụng cho phép đọc và nghe truyện online cùng một lúc.

Mặt khác số lượng người sử dụng điện thoại hiện nay là vô cùng lớn, nên việc xây dựng một phần mềm cho phép đọc truyện online chạy trên điện thoại di động là hợp lý. Do đó, em xây dựng ứng dụng đọc truyện online trên hệ điều hành Android phục vụ nhu cầu này.

# 1.2.Nội dung đề tài

**Tên đề tài**

Xây dựng ứng dụng đọc truyện trên hệ điều hành android sử dụng Web Service

**Nhiệm vụ của đề tài**

Sản phẩm của đề tài này là một hệ thống chạy trên điện thoại thông minh sử dụng hệ điều hành Android cho phép người dùng có thể đọc và nghe sách, truyện tại bất kì đâu, bất kì lúc nào nào khi có kết nối internet. Những công việc cần phải thực hiện trong trong quá trình thực hiện đề tài như sau:

* Xây dựng hệ thống server có nhiệm vụ truy cập csdl cung cấp các thông tin cần thiết.
* Xây dựng client, qua đó người dùng gửi yêu cầu tới server .

Đối với người dùng :

* Chương trình phải cung cấp các thông tin về truyện: tác giả, thể loại…
* Cho phép người dùng đọc và nghe truyện online . Với truyện audio cần phải tự động nghe theo playlist các chương của truyện.
* Cho phép người dùng vote truyện yêu thích , comment về truyện .
* Vì đây là chương trình trên điện thoại di động, nên thiết kế phải trực quan, dễ sử dụng.

Đối với người quản trị: có thêm các chức năng

* Cho phép người quản trị quản trị tài khoản người dùng: xem danh sách người dùng , khóa tài khoản người dùng , xóa comment của người dùng không hợp lệ .

# 1.3.Bố cục đồ án

Đồ án gồm các phần chính sau:

**Chương 1: Đặt vấn đề**

Trình bày tổng quát về đề tài và bố cục của đề tài

**Chương 2: Cơ sở lý thuyết và định hướng giải pháp**

Chương này trình bày vấn đề về nền tảng cơ sở lý thuyết khi thực hiện đề tài , cũng như định hướng giải pháp chung cho đề tài và môi trường, công cụ, cơ sở lý thuyết và các công nghệ sử dụng trong đề tài .

**Chương 3: Xây dựng ứng dụng đọc truyện**

Chương này trình bày chi tiết phân tích thiết kế hệ thống. Mô hình cơ sở dữ liệu của hệ thống .

**Chương 4: Triển khai ứng dụng**

Chương này trình bày chi tiết môi trường triển khai , cách thức triển khai và kịch bản demo hệ thống .

**Chương 5: Kết luận**

Rút ra kết luận chung cho đề tài, ưu điểm nhược điểm của chương trình và các vấn đề làm được và chưa làm được từ đó đưa ra định hướng phát triển đề tài.

# Chương II: CƠ SỞ LÝ THUYẾT VÀ ĐỊNH HƯỚNG GIẢI PHÁP

# 2.1.Giới thiệu hệ điều hành Android

Tháng 7/2005, Google mua lại công ty phát triển phần mềm điện thoại Android, nhưng chưa công bố sẽ sử dụng cho mục đích gì.Để trả lời cho câu hỏi đó, tháng 11/2007, sau 2 năm phát triển, Google công bố hệ điều hành điện thoại di động mã nguồn mở Android, cùng với sự thành lập “Liên minh di động mở” (Open Handset Alliance) bao gồm hơn 65 nhà sản xuất phần cứng điện thoại lớn trên thế giới như Intel, HTC, China Mobile, T-Mobile,....

Từ năm 2007, hệ điều hành Android đã trải qua nhiều lần cập nhật, với phiên bản gần đây nhất là Gingerbread2.3, ra ngày 06/12/2010.Tính đến tháng 6/2010, đã có khoảng 70 mẫu điện thoại di động sử dụng hệ điều hành Android, chưa kể các thiết bị điện tử khác như máy tính bảng (tablet computers), ebook readers, Google TV,...

Android là hệ điều hành điện thoại di động mở nguồn mở miễn phí do Google phát triển dựa trên nền tảng của Linux. Bất kỳ một hãng sản xuất phần cứng nào cũng đều có thể tự do sử dụng hệ điều hành Android cho thiếtbị của mình, miễn là các thiết bị ấy đáp ứng được các tiêu chuẩn cơ bản do Google đặt ra (có cảm ứng chạm, GPS, 3G,...)(Xem thêm: [**Android Compatibility Definition Document**](http://source.android.com/compatibility/android-2.1-cdd.pdf)**)**

Các nhà sản xuất có thể tự do thay đổi phiên bản Android trên máy của mình một cách tự do mà không cần phải xin phép hay trả bất kì khoản phí nào nhưngphải đảm bảo tính tương thích ngược (backward compatibility) của phiên bản chế riêng đó.

Android là nền tảng cho thiết bị di động bao gồm một hệ điều hành, midware và một số ứng dụng chủ đạo. Bộ công cụ Android SDK cung cấp các công cụ và bộ thư viên các hàm API cần thiết để phát triển ứng dụng cho nền tảng Android sử dụng ngôn ngữ lập trình java.

Những tính năng mà nền tảng Android hổ trợ:

**Application framework:** Cho phép tái sử dụng và thay thế các thành phần sẳn có của Android.

**Dalvik virtual macine:** Máy ảo java được tối ưu hóa cho thiết bị di động.

**Intergrated browser:** Trình duyệt web tích hợp được xây dựng dựa trên [WebKit](http://webkit.org/) engine.

**Optimized graphics:** Hổ trợ bộ thư viện 2D và 3D dự vào đặc tả OpenGL ES 1.0.

**SQLite:** DBMS dùng để lưu trữ dữ liệu có cấu trúc.

Hổ trở các định dạng media phổ biến như: MPEG4, H.264, MP3, AAC, ARM, JPG, PNG, GIF.

Hổ trợ thoại trên nền tảng GSM (Phụ thuộc vài phần cứng thiết bị).

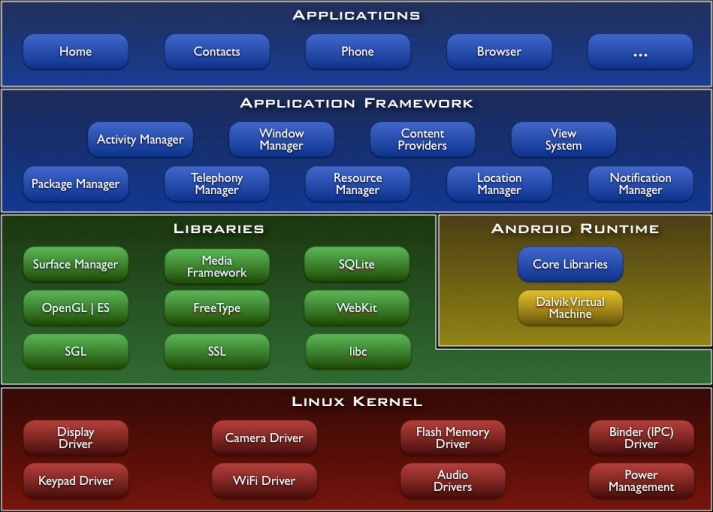
Bluetooth, EDGE, 3G và WiFi (Phụ thuộc vài phần cứng thiết bị).

Camera, GPS, la bàn và cảm biến (Phụ thuộc vài phần cứng thiết bị).

Bộ công cụ phát triển ứng dụng mạnh mẽ.

# 2.2.Kiến trúc Android

Lược đồ sau thể hiện các thành phần của hệ điều hành Android:



Hình .Mô hình kiến trúc nền tảng Android

Applications

Hệ điều hành Android tích hợp sẳn một số ứng dụng cơ bản như email client, SMS, lịch điện tử, bản đồ, trình duyệt web, sổ liên lạc và một số ứng dụng khác. Ngoài ra tầng này cũng chính là tầng chứa các ứng dụng được phát triển bằng ngôn ngữ Java.

Application Framwork

Tầng này của hệ điều hành Android cung cấp một nền tảng phát triển ứng dụng mở qua đó cho phép nhà phát triển ứng dụng có khả năng tạo ra các ứng dụng vô cùng sáng tạo và phong phú. Các nhà phát triển ứng dụng được tự do sử dụng các tính năng cao cấp của thiết bị phần cứng như: thông tin định vị địa lý, khả năng chạy dịch vụ dưới nền, thiết lập đồng hồ báo thức, thêm notification vào status bar của màn hình thiết bị…

Người phát triển ứng dụng được phép sử dụng đầy đủ bộ API được dùng trong các ứng dụng tích hợp sẳn của Android. Kiến trúc ứng dụng của Android được thiết kế nhằm mục đích đơn giản hóa việc tái sử dụng các component. Qua đó bất kì ứng dụng nào cũng có thể công bố các tính năng mà nó muốn chia sẻ cho các ứng dụng khác (VD: Ứng dụng email có muốn các ứng dụng khác có thể sử dụng tính năng gởi mail của nó). Phương pháp tương tự cho phép các thành phần có thể được thay thế bởi người sử dụng.

Tầng này bao gồm một tập các services và thành phần sau:

Một tập phong phú và có thể mở rộng bao gồm các đối tượng View được dùng để xây dựng ứng dụng như: list, grid, text box, button và thậm chí là một trình duyệt web có thể nhúng vào ứng dụng.

**Content Provider:** Cho phép các ứng dụng có thể truy xuất dữ liệu từ các ứng dụng khác hoặc chia sẽ dữ liệu của chúng.

**Resource Manager:** Cung cấp khả năng truy xuất các tài nguyên non-code như hình ảnh hoặc file layout.

**Notification Manager:** Cung cấp khả năng hiển thị custom alert trên thanh status bar.

**Activity Manager:** Giúp quản lý vòng đời của một ứng dụng.

Libraries

Hệ điều hành Android bao gồm một tập các bộ thư viện C/C++ được sử dụng bởi nhiều thành phần của Android system. Những tính năng này được cung cấp cho các lập trình viên thông qua bộ framework của Android. Dưới đây là một số thư viện cốt lõi:

**System C library:** một thể hiện được xây dựng từ BSD của bộ thư viện hệ thống C chuẩn (libc), được điều chỉnh để tối ưu hóa cho các thiết bị chạy trên nền Linux.

**Media libraries:** Bộ thư viện hổ trợ trình diễn và ghi các định dạng âm than và hình ảnh phổ biến.

**Surface manager:** Quản lý hiển thị nội dung 2D và 3D.

**LibWebCore:** Một web browser engine hiện đại được sử dụng trong trình duyệt của Android lần trong trình duyệt nhúng web view được sử dụng trong ứng dụng.

**SGL:** Engine hổ trợ đồ họa 2D.

**3D libraries:** Một thể hiện được xây dựng dựa trên các APIs của OpenGL ES 1.0. Những thư viện này sử dụng các tăng tốc 3D bằng phần cứng lẫn phần mềm để tối ưu hóa hiển thị 3D.

**FreeType:** Bitmap and vector font rendering.

**SQLite:** Một DBMS nhỏ gọn và mạnh mẽ.

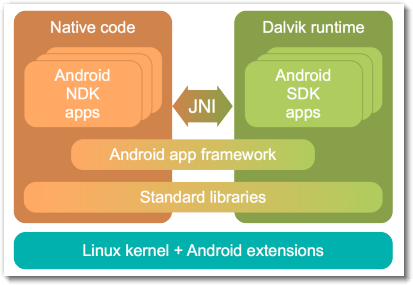
Android Runtime

Hệ điều hành Android tích hợp sẳn một tập hợp các thư viện cốt lõi cung cấp hầu hết các chức năng có sẵn trong các thư viện lõi của ngôn ngữ lập trình Java. Mọi ứng dụng của Android chạy trên một tiến trình của riêng nó cùng với một thể hiện của máy ảo Dalvik. Máy ảo Dalvik thực tế là một biến thể của máy ảo Java được sửa đổi, bổ sung các công nghệ đặc trưng của thiết bị di động. Nó được xây dựng với mục đích làm cho các thiết bị di động có thể chạy nhiều máy ảo một cách hiệu quả. Trước khi thực thi, bất kì ứng dụng Android nào cũng được convert thành file thực thi với định dạng nén Dalvik Executable (.dex). Định dạng này được thiết kế để phù hợp với các thiết bị hạn chế về bộ nhớ cũng như tốc độ xử lý. Ngoài ra máy ảo Dalvik sử dụng bộ nhân Linux để cung cấp các tính năng như thread, low-level memory management.

Linux Kernel

Hệ điều hành Android được xây dựng trên bộ nhân Linux 2.6 cho những dịch vụ hệ thống cốt lõi như: security, memory management, process management, network stack, driver model. Bộ nhân này làm nhiệm vụ như một lớp trung gian kết nối phần cứng thiết bị và phần ứng dụng.

Dưới đây là mô hinh hợp tác giữa máy ảo Dalvik và Navite code:



Hình .Mô hình hợp tác giữa máy ảo Dalvik và Navite code

JNI: Java Native Interface (Tương tự khái niệm Application Programming Interface).

Java Native Interface: là một bộ framework cho phép mã lệnh viết bằng Java chạy trên máy ảo java có thể gọi hoặc được gọi bởi một ứng dụng viết bằng **native code** (Ứng dụng được viết cho một phần cứng cụ thể và trên một hệ điều hành cụ thể) hoặc những bộ thư viện viết bằng C, C++ hoặc Assembly.

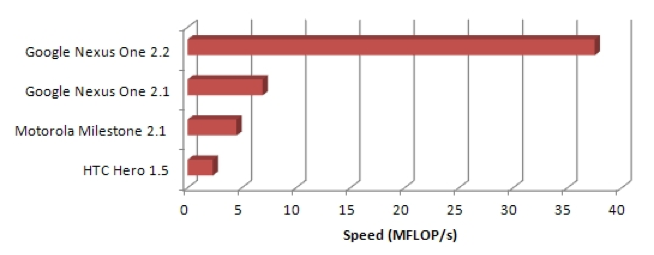
Bằng cách sử dụng JNI, Android cho phép các ứng dụng chạy trên máy ảo Dalvik có thể sử dụng những phương thức được viết bằng các ngôn ngữ cấp thấp như: C, C++, Assembly. Qua đó các nhà phát triển ứng dụng có thể xây dựng ứng dụng dựa trên các bộ thư viện viết bằng C, C++, Assembly nhằm tăng tốc độ thực thi của ứng dụng hoặc sử dụng những tính năng mức thấp mà ngôn ngữ Java không hổ trợ. Tuy nhiên người phát triển ứng dụng cần phải cân nhắc sự gia tăng độ phức tạp của ứng dụng khi quyết định sử dụng các bộ thư viện này.

# 2.3.Phát triển ứng dụng trên Android

## 2.3.1.Ngôn ngữ lập trình

Ngôn ngữ lập trình chính thức của Android là Java.Mặc dù các ứng dụng trên Android được phát triển dựa trên nền tảng Java, nhưng Android không hỗ J2ME và J2SE, là hai ngôn ngữ lập trình phổ dụng cho các thiết bị di động.

Dựa trên máy ảo Java của Sun, Google đã tinh chỉnh và phát triển nên máy ảo Dalvik để biên dịch mã Java với tốc độ biên dịch nhanh hơn và nhẹ hơn. Đến phiên bản Froyo 2.2, Android đã hỗ trợ Just-in-time Compiler (JIT) làm tăng tốc độ biên dịch Java lên gấp 2-5 lần so với các phiên bản trước.



Hình .Biểu đồ so sánh tốc độ của máy ảo Java giữa các phiên bản Android (Càng lớn càng tốt)

Bằng cách sử dụng các frameworks của hãng thứ ba (VD: PhoneGap, Titanium,...), các nhà lập trình web cũng có thể phát triển ứng dụng Android một cách dễ dàng bằng các ngôn ngữ web phổ dụng như HTML, CSS, Javascript. Tuy nhiên số lượng các ứng dụng đi theo chiều hướng này chưa nhiều.

Ngoài ra các ứng dụng của Android sử dụng SQLite để quản lý cơ sở dữ liệu.

## 2.3.2.Vấn đề phân mảnh phiên bản Android trên thị trường:

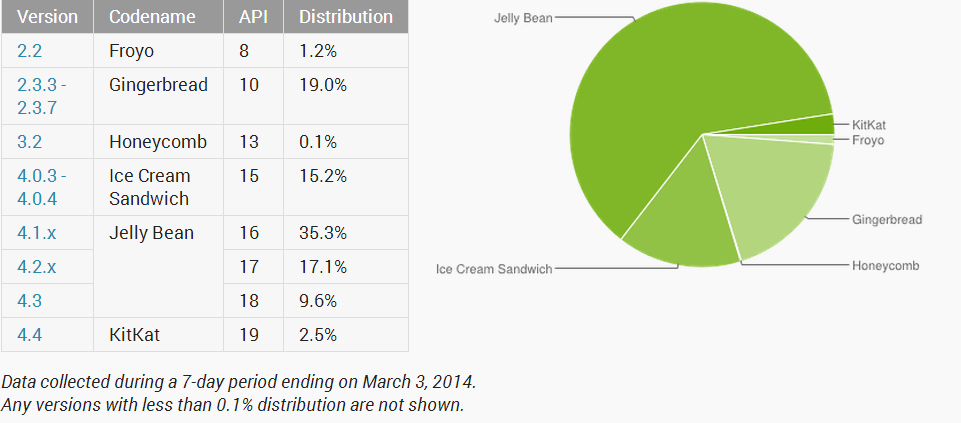
Do Android là hệ điều hành mã nguồn mở và miễn phí, bất kỳ một nhà sản xuất phần cứng nào cũng có thể sử dụng và tùy biến Android để cài đặt trên các thiết bị của mình. Điều này dẫn đến vấn để phân mảnh (fragmentation) khi trên thị trường tồn tạinhiềumẫu điện thoại Android, mỗi mẫu chạy một phiên bản Android khác nhau. Đôi khi các phiên bản Android này còn khác nhau về giao diện hay chức năng, do các nhà sản xuất tự chế thêm vào mã nguồn của Android. Một ví dụ tiêu biểu chính là giao diện Sense UI của HTC.

|  |  |
| --- | --- |
|  |  |

Hình . So sánh giao diện nguyên bản của Android và giao diện Sense của riêng hãng HTC thiết kế

Mặt khác, do việccập nhật hệ điều hành phụ thuộc hoàn toàn vào các nhà sản xuất, thế nên sau khi Google công bố phiên bản Android mới , người sẽ buộc phải chờ khá lâu mới được cập nhật.

Do vấn đề phân mảnh này, trên thị trường một lúc tốn tại nhiều phiên bản Android khác nhau, và người phát triển ứng dụng Android buộc phải xem xét đến khả năng ứng dụng do mình viết ra có thể sẽ khôngchạy đuợc hay thiếu di một số chức năng quan trọng khi chaỵ trên các máy chạy phiên bản Android cũ.



Hình .Thống kê số lượng điện thoại sử dụng các phiên bản Android

## 2.3.3.Vấn đề kích thước màn hình

Một lần nữa, do trên thị trường có rất nhiều nhà sản xuất điện thoại Android, và mặc dù Google yêu cầu tất cả mọi điệnthoại Android phải đáp ứng 1 số tiêu chí chung về phần cứng (màn hình cảm ứng, GPS, 3G,...), các nhà sản xuất hoàn toàntự do quyết định kích cỡ và độ phân giải điệnthoại.

Như vậy, các nhà phát triển ứng dụng Android phải chuẩn bị trước khả năng ứng dụng của mình sẽ chạy trên một loạt các kích cỡ màn hình lớn nhỏ khácnhau.

Tuy nhiên, có một điều chắc chắn: 100% các điện thoại Android đều sử dụng màn hình cảm ứng đa điểm, còn bàn phím vật lý hay cảm ứng quang thì tùy máy.

## 2.3.4.Môi trường lập trình cho Android

Android SDK bao gồm các công cụ riêng lẻ như: debugger, các thư viện, trình giả lập điện thoại Android, các tài liệu hỗ trợ và code mẫu. Hiện Android cung cấp bộ công cụ này trên nhiều nền tảng hệ điều hành khác nhau (Windows, Linux, Mac,...), miễn là có sẵn Java Development Kit, Apache Ant và Python2.2trởlên.

Môitrườnglậptrình (IDE) chính thức của Android là Eclipse (từ phiên bản 3.2) với sự hỗ trợ của plugin Android Development Tools (ADT). Tuy nhiên, người lập trình có thể sử dụng bất kỳ 1 IDE hay trình soạn thảo văn bản nào để viết code Java và XML rồi biên dịch nên ứng dụng hoàn chỉnh bằng cách sử dụng dòng lệnh (command lines).

Ứng dụng Android được đóng gói thành các file .apk và đuợc lưu trong thư mục /data/app của hệ điều hành Android.Java Development Kit (JDK) 5.0.

Một số công cụ hỗ trợ lập trình Android tiêu biểu:

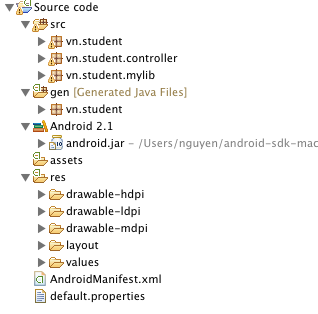
**SQLite Manager:**Là một addon của Firefox giúp quản lí cơ sở dữ liệu SQLite của Android.

**DroidDraw:**Giúp thiếtkếfileXML giao diện ứng dụng.

**Balsamiq Mockups và AdobeFireworks:**Giúp nhanh chóng phác thảo ý tưởng và giao diện sơ bộ của ứng dụng.

**StarUML:** Vẽ các lược đồ UML hỗ trợ phân tích thiết kế.

## 2.3.5.Các thành phần cơ bản của một project Android trên Eclipse



Hình .Cấu trúc thư mục và file của một dự án phần mềm Android trên Eclipse

**AndroidManifest.xml:** file XML mô tả ứng dụng và các thành phần đuợc cung cấp bởi ứng dụng (activities, services,...).

**build.xml:**Mộtfile chứa mã script Ant (ant.apache.com) nhằm compile và cài đặt ứng dụng lên máy.

**default.properties**: file property tạo bởi script Ant trên.

**bin/** : nơi chứa ứng dụng sau khi được compile.

**bin/classes/** : chứa các lớp Java đã được compile.

**bin/classes.dex** : chứa các file executable tạo bởi các lớp Java.

**bin/yourapp.ap\_** : chứa các tài nguyên của ứng dụng, đóng gói thành 1 file zip.

**bin/yourapp-debug.apk** hay **bin/yourapp-unsigned.apk** : chứa chính ứngdụng Android của ta.

**libs/** : nơi chứa các file Java JAR ứng dụng yêu cầu (third party).

**src/** : nơi chứa mã nguồnJava của ứng dụng.

**res/** : chứa các tài nguyên của ứng dụng, như các icons, GUI layouts,...

**res/drawable/** : chứa file hình ảnh (PNG, JPEG,...).

**res/layout/** : chứa UI layout, dưới dạng XML.

**res/menu/** : chi tiết các menu, dưới dạng XML.

**res/raw/** : chứa các file khác (CSV chứa thông tin account,...).

**res/values/** : chứa các strings, dimensions,...

**res/zml/** : chứa các file XML khác cần cho ứng dụng.

**assets/** : nơi chứa các files tĩnh (static) được yêu cầu đi kèm với ứng dụng.

# 2.4.Khái niệm Web Service

Web Serivce là một công nghệ cho phép client truy xuất để thực hiện mọi tác vụ như một Web Application. Về bản chất, Web service dựa trên XML và HTTP, trong đó XML làm nhiệm vụ mã hóa và giải mã dữ liệu và dùng SOAP để truyền tải. Web Service không phụ thuộc vào platform nào, do đó bạn có thể dùng Web Service để truyền tải dữ liệu giữa các ứng dụng hay giữa các platform.

# 2.5.Đặc điểm của Web Service

* Theo đặc tả của W3C (World Wide Web Consortium), dịch vụ Web có khả năng tương tác với các hệ thống khác thông qua một giao diện chung được mô tả bằng XML
* 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.
* Một Dịch vụ Web có bao gồm có nhiều mô-đun và công bố lên 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, Java, Oracle Application server hay Microsoft.NET…

# 2.6.RESTfull Web Service

REST (Representational State Transfer) đã được chọn sử dụng rộng rãi thay cho Web service dựa trên SOAP và WSDL. REST định nghĩa các quy tắc kiến trúc để bạn thiết kế Web services chú trọng vào tài nguyên hệ thống, bao gồm các trạng thái tài nguyên được định dạng như thế nào và được chuyển tải qua HTTP thông qua số lượng lớn người dùng và được viết bởi những ngôn ngữ khác nhau.

Một Web service REST sẽ tuân thủ theo bốn nguyên tắc thiết kế cơ bản sau:

* Sử dụng các phương thức HTTP một cách rõ ràng
* Phi trạng thái
* Hiển thị cấu trúc thư mục như URls
* Trao đổi JavaScript Object Notation (JSON) và XML hoặc cả hai.

### 2.6.1. Sử dụng các phương thức HTTP một cách rõ ràng

Một đặc tính quan trọng của dịch Web service RESTful là sử dụng một cách rõ ràng các phương thức HTTP theo cách một giao thức được xác định bởi RFC 2616. Ví dụ HTTP GET được xác định như là một phương thức sinh ra số liệu được sử dụng có chủ đích bởi các ứng dụng người dùng để thu thập tài nguyên, dữ liệu từ một máy chủ, hoặc thực thi một truy vấn mà máy chủ sẽ tìm kiếm và phản hồi cùng với một gói thông tin tương thích.

REST yêu cầu các nhà phát triển sử dụng phương thức HTTP một cách rõ ràng theo cách tương thích với giao thức chuẩn. Nguyên lý thiết kế REST cơ bản này thiết lập một ánh xạ 1-1 giữa các hành động tạo, đọc, cập nhật và xoá (CRUD) các quá trình vận hành và các phương thức HTTP. Theo cách ánh xạ này thì:

* Để tạo một tài nguyên trên máy chủ, bạn cần sử dụng phương thức POST.
* Để truy xuất một tài nguyên, sử dụng GET.
* Để thay đổi trạng thái một tài nguyên hoặc để cập nhật nó, sử dụng PUT.
* Để huỷ bỏ hoặc xoá một tài nguyên, sử dụng DELETE.

### 2.6.2. Phi trạng thái

Một dịch Web phi trạng thái sinh ra một phản hồi liên kết với số trang tiếp theo trong một tổng thể và để máy khách làm những gì mà nó cần để giữ giá trị này ở mức nhất định. Khía cạnh này của thiết kế dịch vụ Web RESTful có thể được tách thành hai phần trách nhiệm như là mức phân chia cao nhất mà chỉ rõ một dịch vụ phi trạng thái có thể được duy trì như thế nào.

**Máy chủ**

Tạo ra các phản hồi bao gồm các đường dẫn tới nguồn tài nguyên cho phép các ứng dụng điều hướng giữa các tài nguyên liên quan. Loại phản hồi này nhúng các liên kết. Tương tự, nếu các yêu cầu đối với máy chủ hoặc các kho tài nguyên, thì các phản hồi RESTful Web service điển hình có thể bao gồm các đường dẫn đến các máy con hoặc các tài nguyên phụ sao cho những phản hồi này được duy trì kết nối.

Tạo ra các phản hồi mà xác định chúng có thể lưu trữ hoặc không phải để nâng cao được hiệu quả bằng cách giảm số lượng yêu cầu đối với các tài nguyên trùng nhau và bằng cách loại trừ một vài yêu cầu toàn bộ. Máy chủ làm được như vậy bằng cách gộp một phản hồi phần đầu HTTP Last - Modified (lần sửa gần nhất) (giá trị ngày) và Cache-Control (bộ điều khiển lưu trữ).

**Ứng dụng máy khách**

Sử dụng phần đầu phản hồi Cache-Control (bộ điều khiển lưu trữ tạm) để xác định lưu trữ tài nguyên (lập một vùng sao chép nội bộ) hay không. Máy khách cũng đọc phần đầu phản hồi Last-Modified (lần sửa gần nhất) và gửi lại giá trị ngày vào phần đầu If-Modified-Since (nếu-sửa) để truy vấn máy chủ xem tài nguyên có thay đổi không. Việc này được gọi là truy vấn có điều kiện, và hai phần đầu đi với nhau trong phản hồi của máy chủ là mã 304 chuẩn (không sửa đổi) và bỏ qua tài nguyên thực được yêu cầu nếu nó không thay đổi. Mã phản hồi HTTP 304 có nghĩa rằng máy khách có thể sử dụng an toàn một vùng sao lưu nội bộ, lưu giữ một bản sao mới nhất của tài nguyên đại diện, hiệu quả bằng cách vượt qua yêu cầu GET tiếp theo cho đến khi tài nguyên thay đổi.

Gửi các yêu cầu hoàn chỉnh có thể được đáp ứng độc lập bởi các yêu cầu khác. Điều này đòi hỏi máy khách sử dụng toàn bộ các phần đầu HTTP như chỉ định bởi giao diện dịch vụ mạng và để gửi các đại diện tài nguyên hoàn chỉnh trong phần giữa của yêu cầu. Máy khách gửi yêu cầu lập một vài giả thuyết về các yêu cầu trước đó, sự tồn tại của một vùng của máy chủ, khả năng của máy chủ để thêm các ngữ cảnh vào yêu cầu, hoặc về các trạng thái ứng dụng mà được giữ giữa các yêu cầu.

Sự hợp tác này giữa ứng dụng máy khách và máy chủ là cần thiết để có một phi trạng thái trong một Web service RESful. Nó nâng cao hiệu quả bằng cách tiết kiệm băng thông và tối thiểu hoá trạng thái ứng dụng về phía máy chủ.

### 2.6.3. Đưa ra cấu trúc thư mục giống URIs

Các địa chỉ Web service REST nên có tính hiện thực theo nghĩa rằng chúng dễ dàng đối với người dùng. Cấu trúc của một địa chỉ nên rõ ràng, có thể đoán được và dễ hiểu.

Một cách để đạt được mức độ sử dụng này là xác định cấu trúc thư mục giống URIs. Loại URI này có thứ bậc, có điểm khởi nguồn tại một đường dẫn đơn giản, và có nhánh đi ra là các nhánh phụ thể hiện các vùng chính của dịch vụ. Theo định nghĩa này, một URI không chỉ là một chuỗi bị cắt không giới hạn, mà còn là một cây với các nhánh chính và nhánh dọc nối với nhau tại các nút. Ví dụ, trong một thảo luận dịch vụ nhỏ thu thập các chủ đề từ Java tới bài viết, bạn có thể định nghĩa một tập hợp được cấu trúc bởi URIs giống như sau:

**http://www.myservice.org/discussion/topics/{topic}**

Phần gốc, /discussion, có một nút /topics bên dưới nó. Phía dưới là một chuỗi tên các chủ đề, như chuyện xã hội, kỹ thuật, v.v.., mỗi chủ đề chỉ ra một mạch thảo luận. Trong cấu trúc này, dễ dàng kéo các mạch thảo luận bằng cách gõ một vài thứ sau /topics/.

### 2.6.4. Trao đổi XML, JSON, hoặc cả hai

Để trao đổi thông tin giữa client và server , webservice cung cấp định dạng trao đổi dữ liệu XML hoặc JSON . Nó cho phép client đọc hiểu được các tài nguyên trên server với các thuộc tính của tài nguyên đó.

Ví dụ:

{

    "error": false,

    "tasks": [

        {

            "id": 1,

            "task": "Complete REST article by Sunday",

            "status": 0,

            "createdAt": "2014-01-08 23:35:45"

        },

        {

            "id": 2,

            "task": "Book bus tickets!",

            "status": 0,

            "createdAt": "2014-01-08 23:56:52"

        }

    ]

}

Với ví dụ trên, mảng JSON cho ta biết được thông tin về các task với các thuộc tính của nó (tên, trạng thái,status,ngày tạo)

# 2.7. HTTP Progressive Download

Sử dụng kĩ thuật Progressive Download( truyền lũy tiến) để truyền audio trong ứng dụng. Với kĩ thuật này , ta cần sử dụng 1 Web Server và File được lưu trữ trong Web Server.

Giao thức HTTP được sử dụng để truyền thông giữa Client và Server, sử dụng giao thức TCP. Với kĩ thuật này :

– Audio được truyền với tốc độ tốt nhất có thể phụ thuộc mạng.

– File được Download dần từng phần kết hợp với trình diễn

– File chỉ được trình diễn khi đã nhận đủ một lượng nhất định nào đấy (được quy định bởi Mediaplayer trong Android)

# **2.8.Định hướng giải pháp và lựa chọn công cụ xây dựng**

HTTP request

Database

WebServer

HTTP request

Client

HTTP response

Hình .Sơ đồ kiến trúc hệ thống

**Client**

**Client**

* Kiến trúc hệ thống: Từ các yêu cầu và nhiệm vụ đặt ra như trên chúng ta cần xây dựng hệ thống theo cấu trúc Client – Server dữ liệu tập trung tại Server và các Client sẽ kết nối tới Server thông qua mạng internet để trao đổi thông tin với Server
* Giao thức truyền tin giữa Client và Server: Ta sử dụng giao thức HTTP, đây là giao thức cho phép trao đổi thông tin ở dạng siêu văn bản thông qua mạng internet. Nó tương đối dễ triển khai và cài đặt, sử dụng kết hợp với JSON để truyền dữ liệu giữa client và server
* Server : là một http Server . Ta sẽ sử dụng RESTfull để cung cấp api như một web service. HTTPServer đơn giản miễn phí dễ sử dụng là Apache .
* Client : Yêu cầu client cần phải có kết nối mạng. Chúng ta sử dụng các smartphone android vì nó hỗ trợ các khả năng kết nối mạng, định vị toàn cầu, dễ lập trình và giá thành tương đối rẻ.

Viết ứng dụng trên hệ điều hành Android có thể sử dụng nhiều công cụ nhưng phổ biến nhất là Eclipse, cung cấp đầy đủ thư viện, giả lập điện thoại phục vụ cho việc phát triển ứng dụng.

* Ngôn ngữ:
* Client : Android
* Server : PHP
* Môi trường phát triển:
* Java Development Kit 7
* Android SDK 20.0.3 – Android platform 4.0 trở lên
* Eclipse Indigo Service Release 2
* Netbean
* MySQL Server 5.6
* PHP myadmin
* Các thư viện – Framework
* Slim FrameWork

# CHƯƠNG III: XÂY DỰNG ỨNG DỤNG ĐỌC TRUYỆN

## 3.1.Phân tích các chức năng

Ứng dụng Story được thiết kế để đọc và nghe truyện , dữ liệu về truyện được chia sẻ miễn phí trên mạng internet(<http://mobileworld.vn/threads/truyen-dinh-dang-txt.62959/>) và các nguồn chia sẻ truyện miễn phí khác . . Hệ thống dựa trên mô hình client-server, client làm nhiệm vụ hiện thị thông tin, còn server có nhiệm vụ truy xuất cơ sở dữ liệu, truyền cho client. Khi Client muốn thực hiện một chức năng hiển thị thông tin nào sẽ thông qua Web Service sẽ gọi hàm tương ứng với chức năng đó trên Server. Chương trình được thiết kế có các chức năng chính :

Đối với người dùng:

* -Đăng nhập.
* -Đăng ký tài khoản .
* -Danh sách truyện theo tác giả , thể loại .
* -Danh mục thể loại truyện.
* -Danh sách các chương của truyện.
* -Đọc, nghe truyện .
* -Yêu thích , bình luận các chương của truyện.

+Chức năng đăng nhập và đăng ký tài khoàn giúp người dùng có thể sử dụng các tính năng vote truyện yêu thích và bình luận truyện.

+Trong danh sách truyện theo tác giả , thể loại sẽ hiển thị các truyện theo tên truyện ,tác giả , thể loại có số vote của người dùng yêu thích từ cao xuống thấp . Khi tiến hành xem 1 truyện thì màn hình sẽ chuyển tới danh mục các chương của truyện đó.

+Khi vào màn hình đọc 1 chương của truyện , sẽ hiển thị truyện đó cùng với các icon comment cho phép xem danh sách các comments và icon vote cho phép người dùng yêu thích truyện , icon setting cho phép tùy chỉnh phông chữ của truyện . Nếu là truyện audio thì sẽ cho phép người dùng chọn repeat và tự động chuyển qua chương mới khi kết thúc nghe 1 chương truyện , cùng với các menu để điều khiển tiến, lui khi nghe truyện.

+Trong danh sách các thể loại truyện sẽ hiển thị tất cả các thể loại truyện , khi người dùng chọn 1 thể loại truyện thì sẽ hiển thị danh sách các truyện ở thể loại đó.

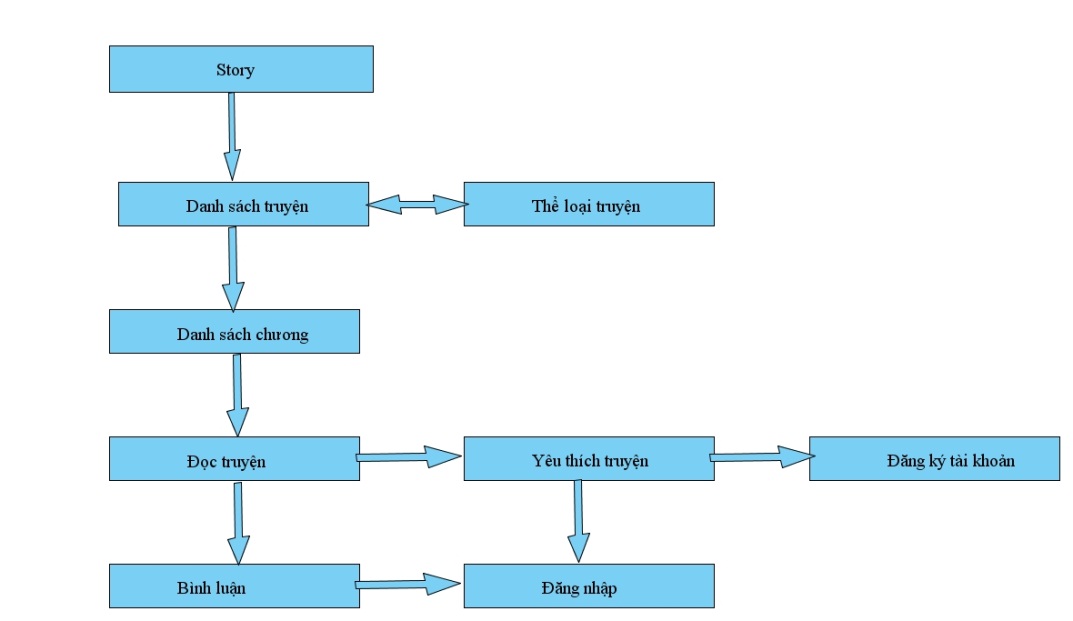
Đối với admin : admin có mọi quyền như người dùng bình thường, ngoài ra admin có thêm các chức năng như:

* -Khóa, mở khóa tài khoản người dùng .
* -Xóa comment không hợp lệ của người dùng

+Admin có thể vào mục quản trị để quản trị tài khoản người dùng : khóa, mở khóa tài khoản . Admin đồng thời có thể xem danh sách các comment của người dùng và xóa comment vi phạm .

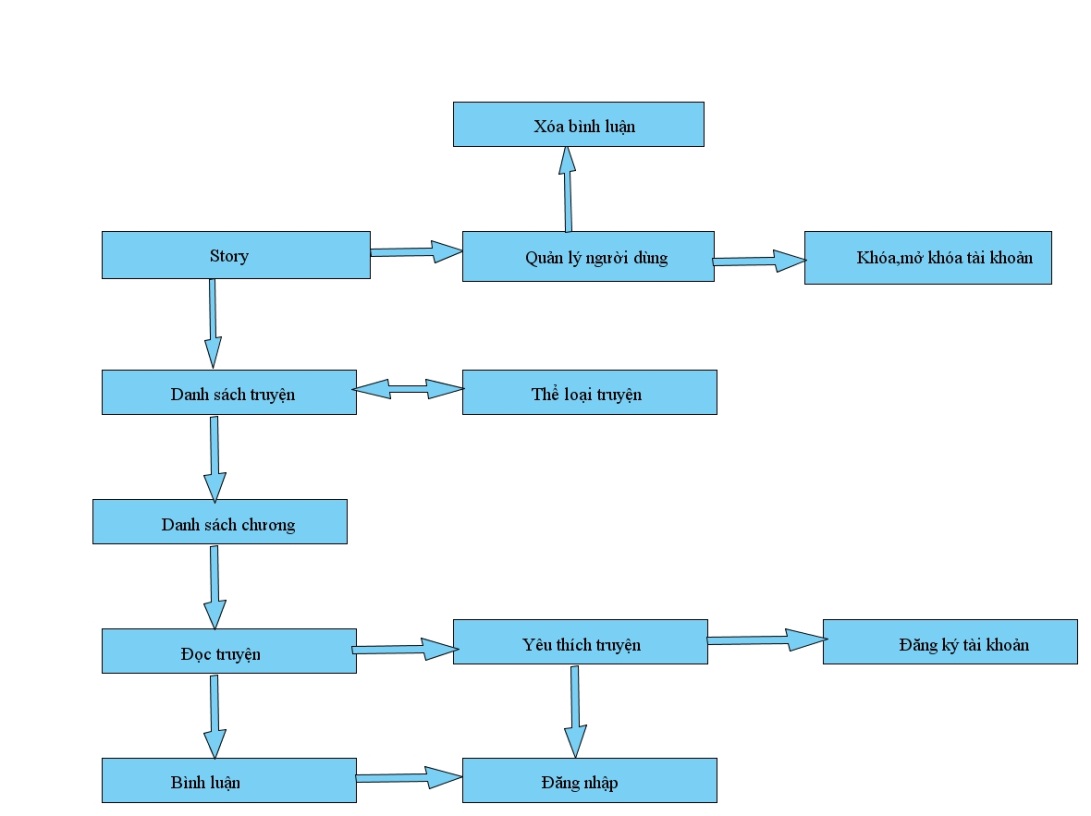
## 3.2.Biểu đồ phân cấp chức năng

## 3.2.1. Biểu đồ



Hình .Biểu đồ phân cấp chức năng đối với người dùng

Hệ thống sẽ gồm 2 chức năng chính là danh sách các truyện và đọc(nghe) truyện .



Hình .Biểu đồ phân cấp chức năng đối với admin

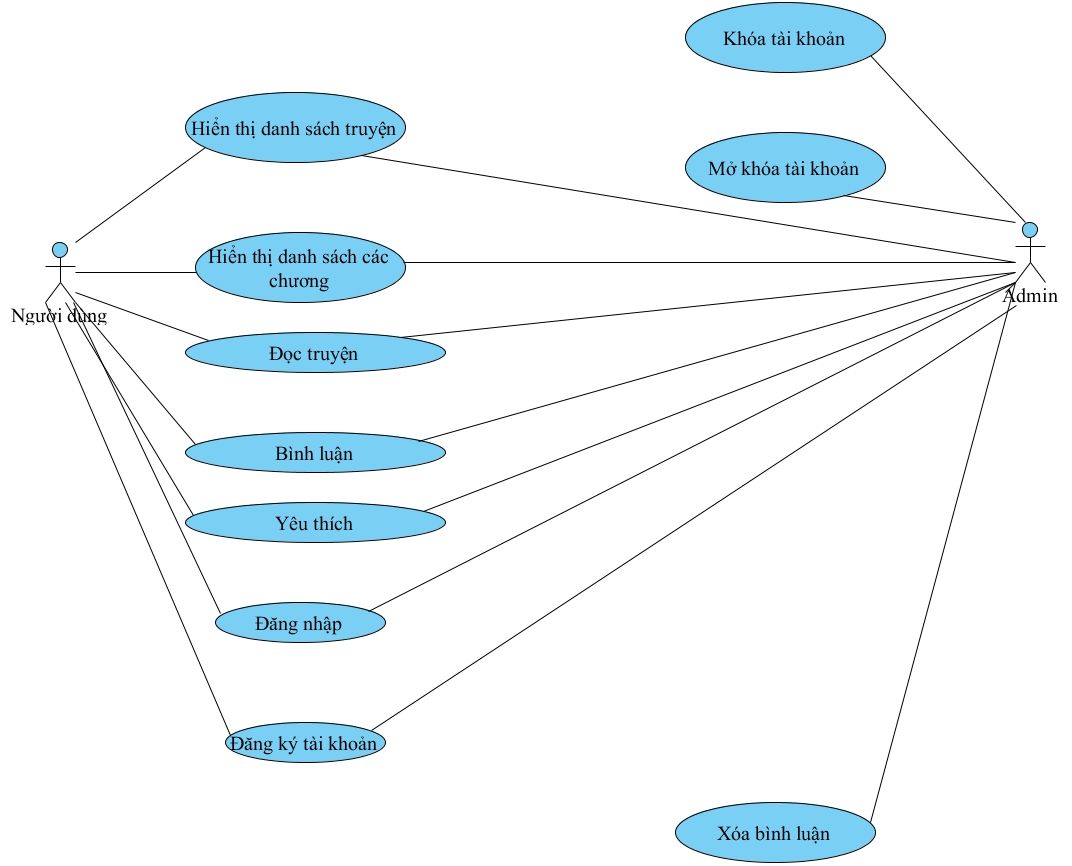
## 3.2.2. Đặc tả chức năng

* Danh sách truyện: hiển thị danh sách các truyện theo thứ tự vote từ cao xuống thấp , bao gồm thông tin về tên truyện , thể loại, tác giả, cho phép người dùng tìm kiếm truyện theo tên .
* Danh sách chương: hiển thị danh sách các chương của truyện .
* Đọc truyện: hiển thị nội dung của truyện hay nghe audio
* Bình luận: cho phép người dùng xem và bình luận
* Yêu thích truyện: cho phép người dùng yêu thích truyện
* Đăng ký tài khoản: người dùng cần đăng ký tài khoản để thực hiện các chức năng bình luận và yêu thích truyện .
* Đăng nhập: đăng nhập tài khoản đã đăng ký để thực hiện được đầy đủ các tính năng của ứng dụng .
* Thể loại truyện: người dùng chọn thể loại truyện muốn xem , màn hình sẽ hiển thị danh sách các truyện có trong thể loại đó .
* Quản lý người dùng: cho phép admin quản lý người dùng: khóa,mở khóa tài khoản, xóa bình luận của người dùng .

## 3.3.Biểu đồ Use case

Qua biểu đồ này, ta sẽ thấy được quá trình thực hiện các chức năng của chương trình cần thông qua những thao tác nào và quan hệ giữa những thao tác đó ra sao.

Theo mô hình Client-Server, mỗi chức năng trên Client sẽ tương ứng với một hàm trên server. Thông qua các Rest request, client sẽ gọi các hàm thích hợp trên server, phục vụ cho mục đích của mình.



Hình .Biểu đồ Use case của ứng dụng

### 3.3.1. Use case hiển thị danh sách truyện

|  |  |
| --- | --- |
| **Use Case** | List Story |
| **Actor** | User |
| **Brief Description** | Hiển thị danh sách truyện |
| **Main Flow** | 1. Gửi yêu cầu tới Server 2. Server truy vấn CSDL và trả về kết quả 3. Nhận và hiển thị kết quả cho người dùng |
| **Alternative Flows** | * 1. Nếu không thành công thông báo cho người dùng biết . |
| **Pre-condition** |  |

Bảng .Use case hiển thị danh sách truyện

### 3.3.2. Use case hiển thị danh sách các chương của truyện

|  |  |
| --- | --- |
| **Use Case** | List Chapter |
| **Actor** | User,Admin |
| **Brief Description** | Hiển thị danh sách các chương của truyện |
| **Main Flow** | 1.Chọn truyện muốn đọc  2.Gửi dữ liệu ID của truyện tới Server  3.Server truy vấn CSDL và trả về kết quả  4.Nhận và hiển thị kết quả cho người dùng |
| **Alternative Flows** | 4.1.Nếu không thành công thông báo cho người dùng biết . |
| **Pre-condition** |  |

Bảng .Use case hiển thị danh sách các chương của truyện

### 3.3.3. Use case đọc truyện

|  |  |
| --- | --- |
| **Use Case** | Read Chapter |
| **Actor** | User,Admin |
| **Brief Description** | Hiển thị danh sách các chương của truyện |
| **Main Flow** | 1.Gửi dữ liệu ID của chương cần đọc tới Server  2.Server truy vấn CSDL và trả về kết quả  3.Nhận và hiển thị kết quả cho người dùng |
| **Alternative Flows** | 3.1.Nếu không thành công thông báo cho người dùng biết . |
| **Pre-condition** |  |

Bảng .Use case đọc truyện

### 3.3.4. Use case bình luận các chương của truyện

|  |  |
| --- | --- |
| **Use Case** | Comment |
| **Actor** | User,Admin |
| **Brief Description** | Bình luận các chương của truyện |
| **Main Flow** | 1.Gửi dữ liệu ID của chương, comment(text),ID của User tới Server  2.Server cập nhật CSDL và trả về kết quả  3.Nhận và hiển thị kết quả cho người dùng |
| **Alternative Flows** | 3.1.Người dùng chưa đăng nhập vào hệ thống |
| **Pre-condition** | User phải đăng nhập thành công vào hệ thống |

Bảng .Use case bình luận truyện

### 3.3.5. Use case yêu thích truyện

|  |  |
| --- | --- |
| **Use Case** | Vote |
| **Actor** | User,Admin |
| **Brief Description** | User có thể yêu thích hay hủy bỏ yêu thích một truyện |
| **Main Flow** | 1.Gửi dữ liệu ID của truyện, ID của user tới Server  2.Server cập nhật CSDL và trả về kết quả  3.Nhận và hiển thị kết quả cho người dùng |
| **Alternative Flows** | 3.1.Người dùng chưa đăng nhập vào hệ thống |
| **Pre-condition** | User phải đăng nhập thành công vào hệ thống |

Bảng .Use case vote truyện

### 3.3.6. Use case đăng nhập

|  |  |
| --- | --- |
| **Use Case** | Login |
| **Actor** | User,Admin |
| **Brief Description** | Cho phéo người dùng đăng nhập vào hệ thống |
| **Main Flow** | 1. Nhập thông tin tài khoản 2. Gửi thông tin đăng nhập tới Server 3. Kiểm tra đăng nhập tại server và gửi về kết quả 4. Nhận kết quả và thông báo cho người dùng biêt |
| **Alternative Flows** | 1.1.Không được bỏ trống, yêu cầu nhập lại nếu trống  4.1.Không thành công thông báo cho người dùng biết và cho nhâp lại, nếu đúng chuyển vào giao diện chính chương trình |
| **Pre-condition** |  |

Bảng .Use case đăng nhập

### 3.3.7. Use case đăng ký tài khoản

|  |  |
| --- | --- |
| **Use Case** | Register new Account |
| **Actor** | User,Admin |
| **Brief Description** | Cho phép tạo tài khoản người dùng mới |
| **Main Flow** | 1. Nhập thông tin người dùng mới 2. Gửi thông tin đăng ký tới Server 3. Kiểm tra tồn tại account, thêm mới account tại server và gửi về kết quả 4. Nhận kết quả và thông báo cho người dùng |
| **Alternative Flows** | * 1. Không được bỏ trốngcác thông tin về username, pass,pass phải trùng nhau nếu không yêu cầu nhập lại   2. Không thành công thông báo cho người dùng biết và cho nhâp lại |
| **Pre-condition** |  |

Bảng .Use case đăng ký tài khoản

### 3.3.8. Use case khóa tài khoản

|  |  |
| --- | --- |
| **Use Case** | Block User |
| **Actor** | Admin |
| **Brief Description** | Cho phép Admin khóa tài khoản người dùng |
| **Main Flow** | 1.Chọn user cần khóa tài khoản.  2.Gửi ID User cần xóa tới Server.  3.Server cập nhật trạng thái tài khoản với ID tương ứng .  3.Nhận kết quả và thông báo cho người dùng. |
| **Alternative Flows** | 1.1.User đã bị khóa tài khoản rồi . |
| **Pre-condition** | Admin phải đăng nhập vào hệ thống |

Bảng .Use case khóa tài khoản

### 3.3.9. Use case mở khóa tài khoản

|  |  |
| --- | --- |
| **Use Case** | Active User |
| **Actor** | Admin |
| **Brief Description** | Cho phép Admin mở khóa tài khoản người dùng |
| **Main Flow** | 1.Chọn user cần mở khóa tài khoản.  2.Gửi ID User cần xóa tới Server.  3.Server cập nhật trạng thái tài khoản với ID tương ứng .  3.Nhận kết quả và thông báo cho người dùng. |
| **Alternative Flows** | 1.1.User đã được mở khóa tài khoản rồi . |
| **Pre-condition** | Admin phải đăng nhập vào hệ thống |

Bảng .Use case mở khóa tài khoản

### 3.3.10. Use case xóa comment

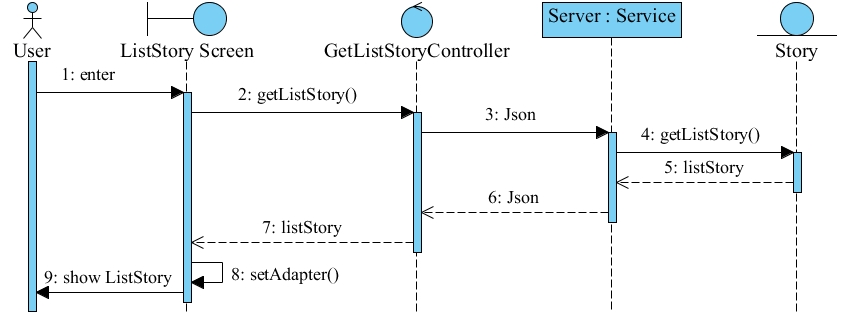
|  |  |
| --- | --- |
| **Use Case** | Block User |
| **Actor** | Admin |
| **Brief Description** | Cho phép Admin xóa comment của người dùng |
| **Main Flow** | 1.Chọn user cần xóa comment  2.Xem danh sách comment của user .  3.Gửi comment ID cần xóa lên Server  4.Server xóa comment với ID tương ứng .  5.Nhận kết quả thông báo trả về. |
| **Alternative Flows** |  |
| **Pre-condition** | Admin phải đăng nhập vào hệ thống |

Bảng .Use case xóa bình luận

## 3.4.Biểu đồ tuần tự

Biểu đồ trong phần này sẽ nói rõ hơn về quá trình thực hiện các chức năng của chương trình. Sử dụng biểu đồ này sẽ giúp cho người lập trình đơn giản hơn trong quá trình cài đặt các thiết kế.

### 3.4.1. Biểu đồ tuần tự chức năng Show List Story

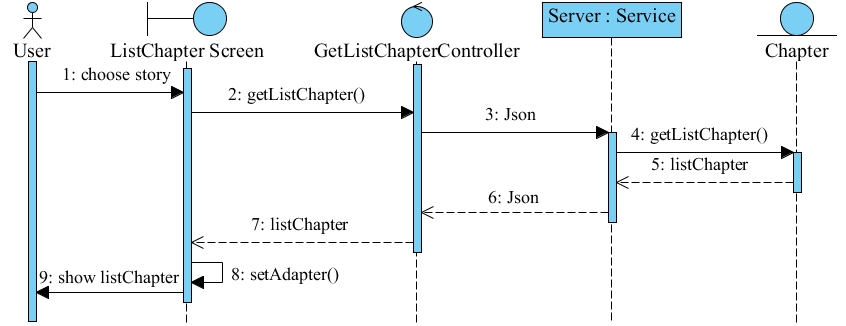


Hình .Biểu đồ tuần tự chức năng Show List Story

Các bước được thực hiện tuần tự như sau:

* 1.User vào ứng dụng
* 2-3-4: lấy tất cả các truyện theo số lượng vote giảm dần
* 5-6-7: Trả về list truyện
* 8: gán adapter vào listview
* 9:hiển thị list Story cho User

### 3.4.2. Biểu đồ tuần tự chức năng Show List Chapter



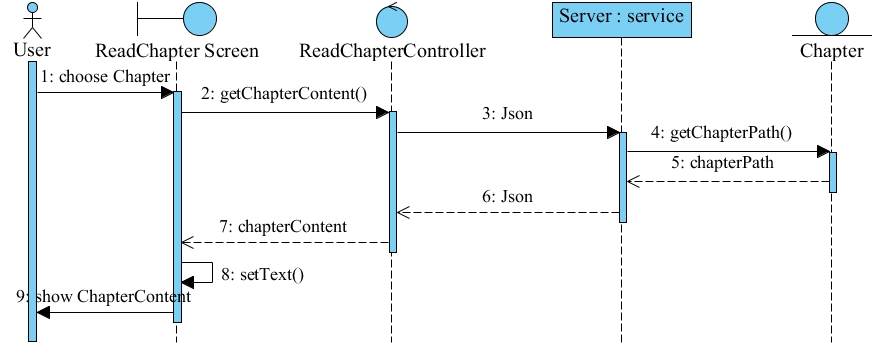
Hình .Biểu đồ tuần tự chức năng Show List Chapter

Các bước được thực hiện tuần tự như sau:

* 1.User chọn truyện muốn đọc
* 2-3-4: lấy tất cả các chương của truyện đó
* 5-6-7: Trả về list các chương
* 8: gán adapter vào listview
* 9:hiển thị list Chapter cho User

### 3.4.3. Biểu đồ tuần tự chức năng Read Chapter

#### 3.4.3.1.Đọc truyện dạng text

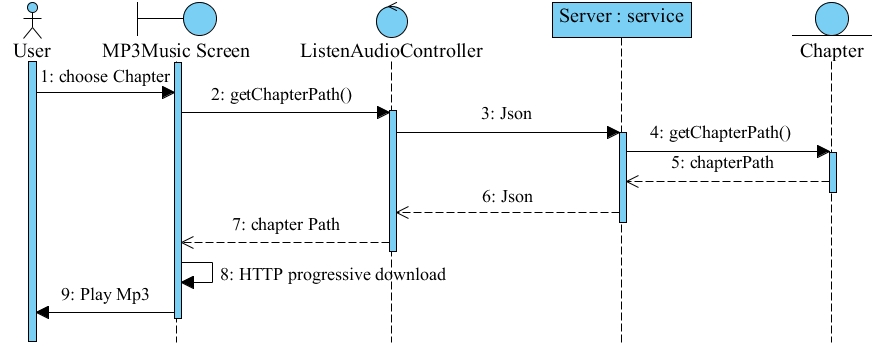


Hình .Biểu đồ tuần tự chức năng Read Chapter(text)

Các bước được thực hiện tuần tự như sau:

* 1.User chọn chapter muốn đọc
* 2-3: truyền ID của chapter lên Server
* 4-5-6:Server truy vấn CSDL để lấy được chapter Path và từ đó có được nội dung của chapter đó, sau đó trả về cho client
* 7-8: set Text cho Textview
* 9:hiển thị nội dung Chapter cho User

#### 3.4.3.2.Đọc truyện dạng mp3

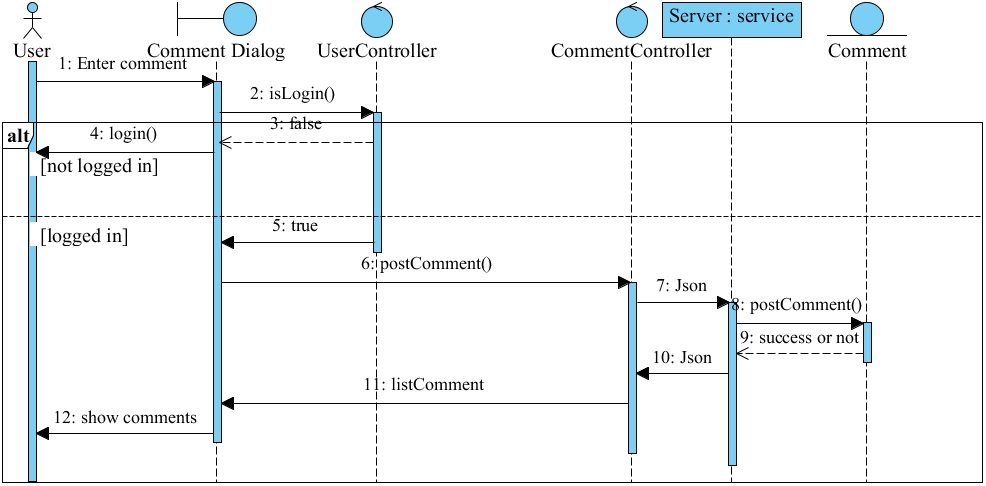


Hình .Biểu đồ tuần tự chức năng Read Chapter(audio)

Các bước được thực hiện tuần tự như sau:

* 1.User chọn chapter muốn đọc
* 2-3: truyền ID của chapter lên Server
* 4-5-6:Server truy vấn CSDL và trả về chapter path cho client
* 7-8: thực hiện HTTP progressive download
* 9:Play music

### 3.4.4. Biểu đồ tuần tự chức năng Comment

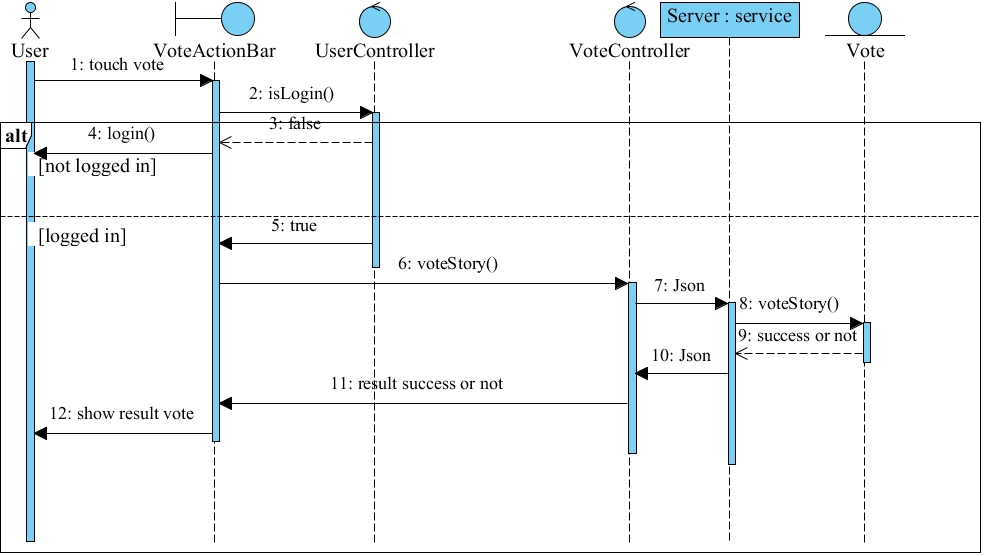


Hình .Biểu đồ tuần tự chức năng Comment

Các bước được thực hiện tuần tự như sau:

* 1.User comment về chapter và ấn submit
* 2: Kiểm tra xem User đã đăng nhập hay chưa
* 3-4: Nếu User chưa đăng nhập thì yêu cầu User đăng nhập
* 5-6-7-8: Nếu User đã đăng nhập , gửi comment đó tới phía Server để lưu vào CSDL
* 9-10-11: trả về kết quả comment thành công hay chưa và trả lại danh sách comment
* 12: hiển thị danh sách comment cho User

### 3.4.5. Biểu đồ tuần tự chức năng Vote Story

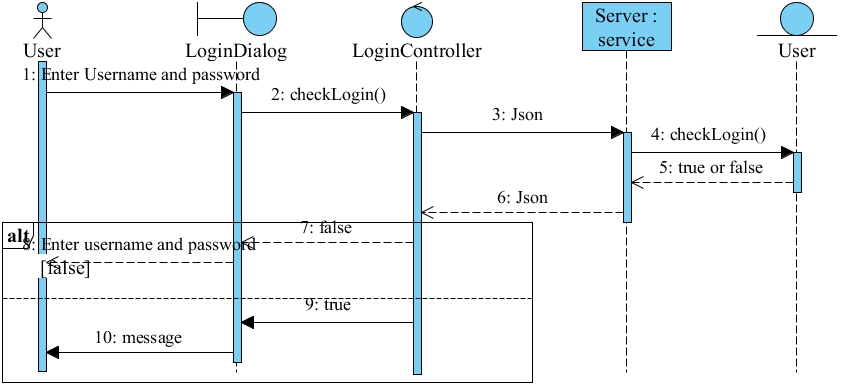


Hình .Biểu đồ tuần tự chức năng Vote Story

Các bước được thực hiện tuần tự như sau:

* 1.User touch vote story trên action bar
* 2: Kiểm tra xem User đã đăng nhập hay chưa
* 3-4: Nếu User chưa đăng nhập thì yêu cầu User đăng nhập
* 5-6-7-8: Nếu User đã đăng nhập , gửi vote đó tới phía Server để lưu vào CSDL
* 9-10-11-12: trả về kết quả vote thành công hay chưa và thông báo kết quả cho User

### 3.4.6. Biểu đồ tuần tự chức năng Login

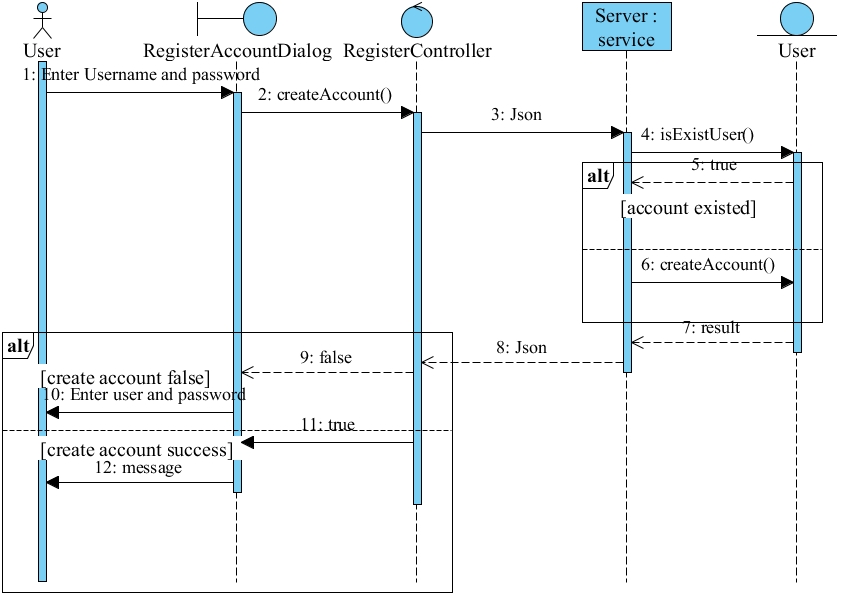


Hình .Biểu đồ tuần tự chức năng Login

Các bước được thực hiện tuần tự như sau:

* 1.User nhập username và password
* 2-3-4: Kiểm tra xem User đã đăng nhập đúng username và password hay không
* 5-6: Trả về kết quả đăng nhập
* 7: Nếu User đăng nhập sai, yêu cầu đăng nhập lại
* 9-10: Nếu User đăng nhập đúng , tắt login dialog và hiển thị kết quả đăng nhập thành công .

### 3.4.7. Biểu đồ tuần tự chức năng Register new account

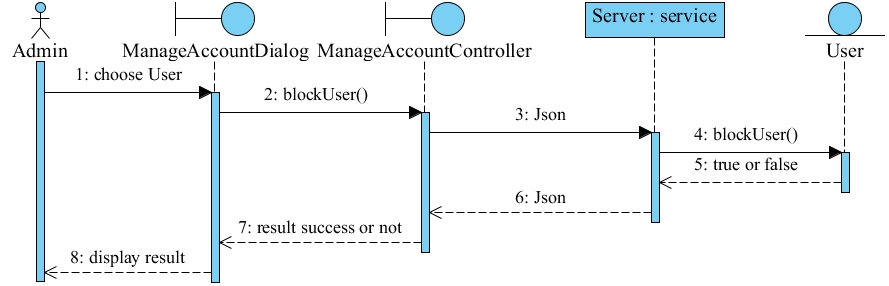


Hình .Biểu đồ tuần tự chức năng Register account

Các bước được thực hiện tuần tự như sau:

* 1.Người dùng nhập username và password
* 2-3-4: Kiểm tra xem tồn tại User như vậy trong hệ thống hay không
* 5: Nếu đã tồn tại tài khoản đó , trả về thông báo
* 6: Nếu chưa tồn tại tài khoản đó , tạo tài khoản mới và trả về thông báo
* 7-8: Trả về kết quả dạng Json
* 9-10: Nếu tạo tài khoản chưa thành công , yêu cầu người dùng đăng ký lại.
* 11-12: Nếu tạo tài khoản thành công , thông báo kết quả cho người dùng.

### 3.4.8. Biểu đồ tuần tự chức năng Block Account

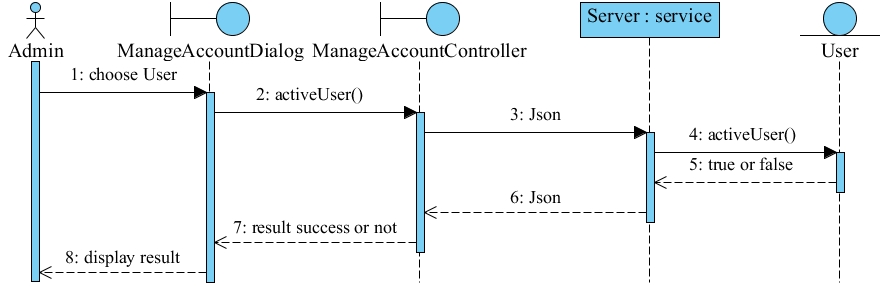


Hình .Biểu đồ tuần tự chức năng Block Account

Các bước được thực hiện tuần tự như sau:

* 1.Admin chọn user và chọn khóa tài khoản .
* 2-3-4: ID của user được gửi tới Server để cập nhật trạng thái user.
* 5-6-7-8: Thông báo kết quả của việc khóa tài khoản có thành công hay không.

### 3.4.9. Biểu đồ tuần tự chức năng Active Account

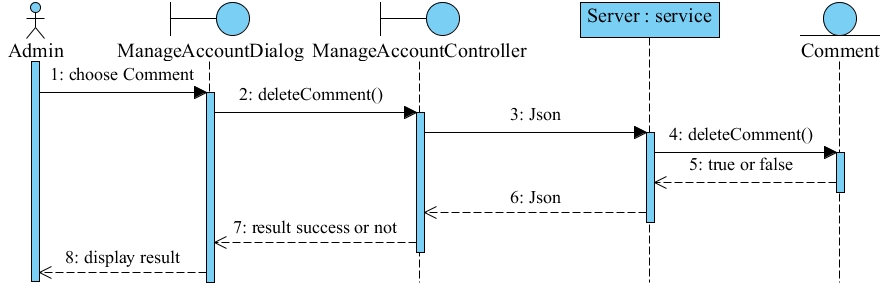


Hình .Biểu đồ tuần tự chức năng Active Account

Các bước được thực hiện tuần tự như sau:

* 1.Admin chọn user và chọn mở khóa tài khoản .
* 2-3-4: ID của user được gửi tới Server để cập nhật trạng thái user.
* 5-6-7-8: Thông báo kết quả của việc mở khóa tài khoản có thành công hay không.

### 3.4.10. Biểu đồ tuần tự chức năng Delete Comment



Hình .Biểu đồ tuần tự chức năng Delete Comment

Các bước được thực hiện tuần tự như sau:

* 1.Admin chọn comment và chọn xóa .
* 2-3-4: ID của comment được gửi tới Server để xóa .
* 5-6-7-8: Thông báo kết quả của việc xóa comment có thành công hay không.

## 3.5.Biểu đồ lớp

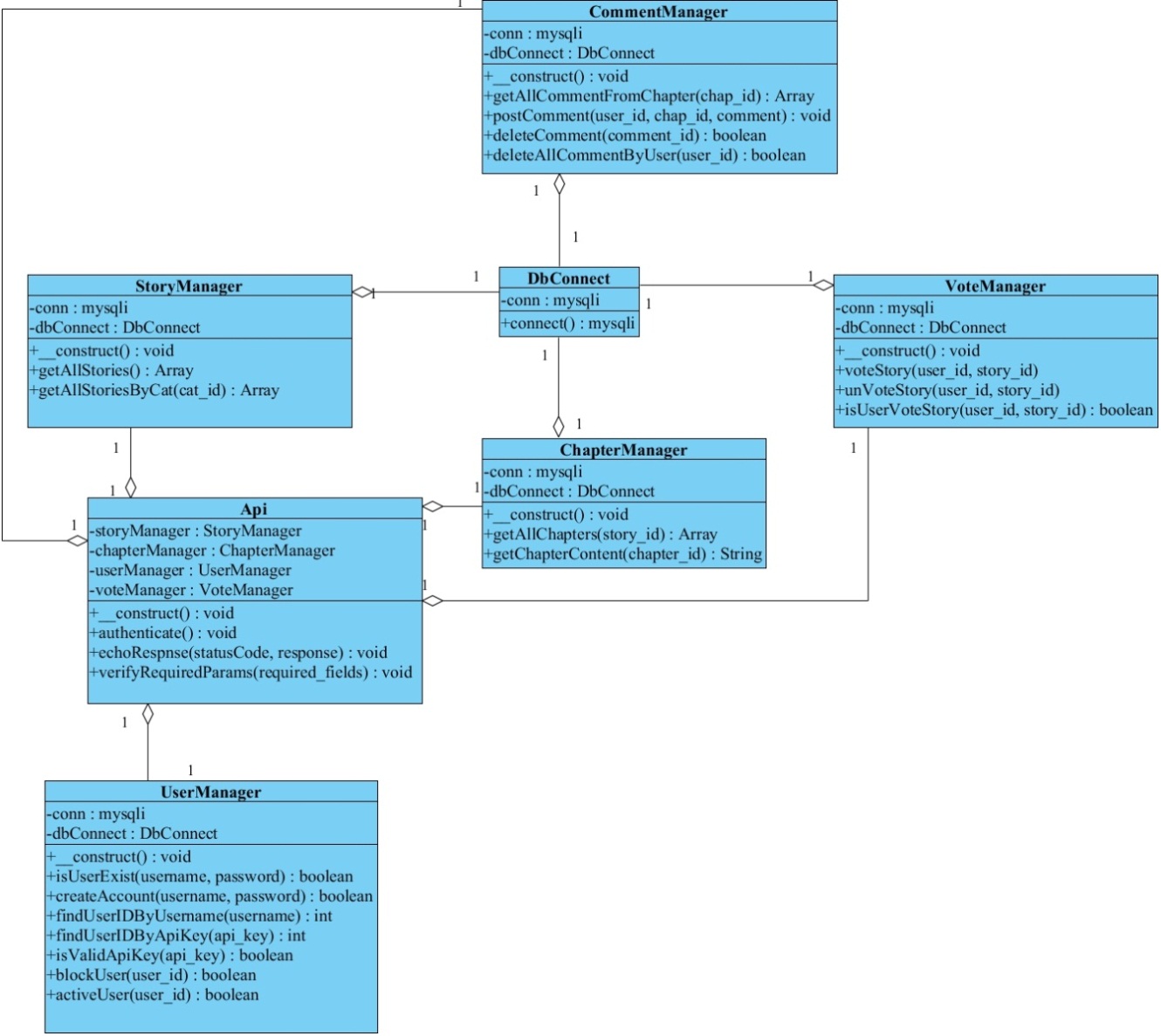
### 3.5.1.Biểu đồ lớp phía client



Hình 22.Biểu đồ lớp phía Client

* **Lớp AllStoriesActivity** : Activity chịu trách nhiêm hiển thị danh sách truyện.
* **Lớp AllChapterActivity**: Activity chịu trách nhiêm hiển thị danh sách các chương của truyện.
* **Lớp ReadChapterActivity**: Activity chịu trách nhiêm hiển thị nội dung chương của truyện text.
* **AndroidBuildingMusicPlayer**: Activity chịu trách nhiêm play audio của truyện audio , quản lý play list.
* **ManageUserActivity:**  Activity chịu trách nhiêm quản lý tài khoản người dùng :khóa, mở khóa,xóa comment.
* **LoginDialog**: Dialog chịu trách nhiêm về hiển thị form đăng nhập .
* **RegisterDialog**: Dialog chịu trách nhiêm về hiển thị form đăng ký .
* **AllCommentDialog**: Dialog chịu trách nhiêm về hiển thị danh sách comment, gửi comment bình luận.
* **JSONParser**: Lớp này chịu trách nhiêm về HTTP request tới Server,gửi và phân tích file Json.

### 3.5.2.Biểu đồ lớp phía Server

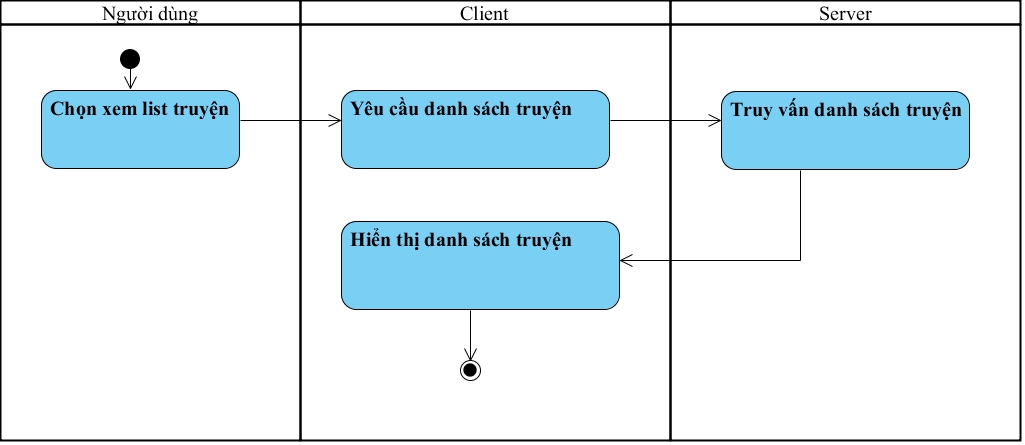


Hình .Biểu đồ lớp phía Server

* **Lớp UserManager** : Lưu trữ dữ liệu của người dùng trên Server, bao gồm các thông tin về ID, UserName, Password,và các thao tác với User
* **Lớp StoryManager:** chịu trách nhiệm việc lấy danh sách về truyện bao gồm tên truyện và tác giả , thể loại của truyện đó , và lấy danh sách truyện theo từng thể loại.
* **Lớp ChapterManager:** chịu trách nhiệm việc lấy danh sách về các chương của truyện, nội dung của từng chương.
* **Lớp VoteManager:** chịu trách nhiệm việc quản lý vote,hay hủy bỏ vote của user về truyện ưa thích .
* **Lớp CommentManager:** chịu trách nhiệm việc quản lý bình luận, của user về truyện.
* **Lớp DbConnect:** chịu trách nhiệm mở đóng kết nối với CSDL
* **Lớp Api:** Chịu trách nhiệm về RESTfull request từ client và trả về Json cũng như mã HTTP status tương ứng.

## 3.6.Biểu đồ hoạt động

### 3.6.1.Biểu đồ hoạt động cho chức năng xem List Story

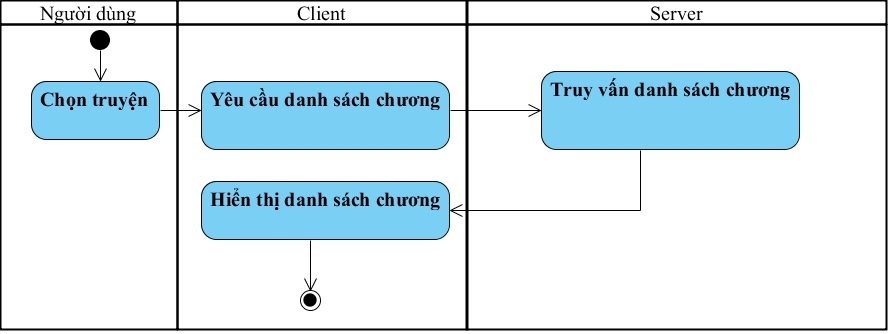


Hình .Biểu đồ hoạt động chức năng xem List Story

Mô tả:

* Người dùng chọn xem danh sách truyện.
* Client request tới Server yêu cầu danh sách truyện .
* Server truy vấn CSDL và trả về danh sách truyện tương ứng
* Client hiển thị danh sách truyện cho người dùng.

### 3.6.2.Biểu đồ hoạt động cho chức năng xem List Chapter



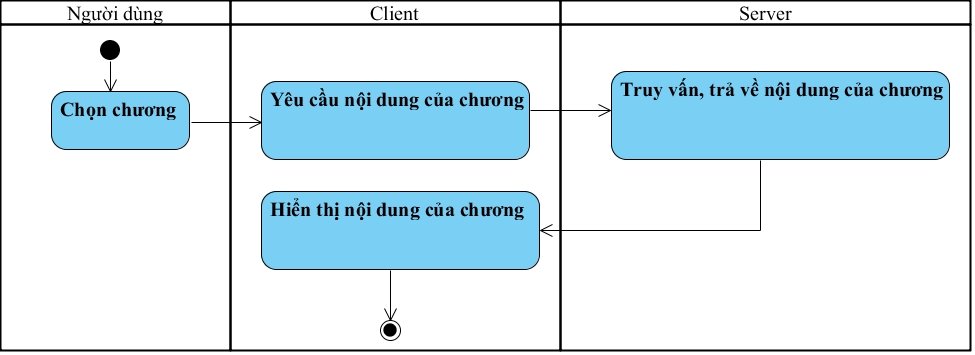
Hình .Biểu đồ hoạt động chức năng xem List Chapter

Mô tả:

* Người dùng chọn truyện.
* Client request tới Server yêu cầu danh sách chương của truyện .
* Server truy vấn CSDL và trả về danh sách chương tương ứng
* Client hiển thị danh sách chương cho người dùng.

### 3.6.3.Biểu đồ hoạt động cho chức năng Read Chapter

#### 3.6.3.1.Đối với truyện text

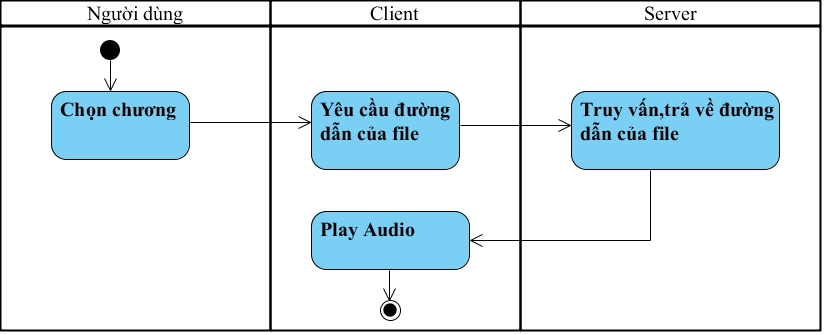


Hình .Biểu đồ hoạt động chức năng xem đọc truyện text

Mô tả:

* Người dùng chọn chương.
* Client request tới Server yêu cầu nội dung của chương .
* Server truy vấn CSDL và trả về nội dung chương tương ứng
* Client hiển thị nội dung chương cho người dùng.

#### 3.6.3.2.Đối với truyện audio

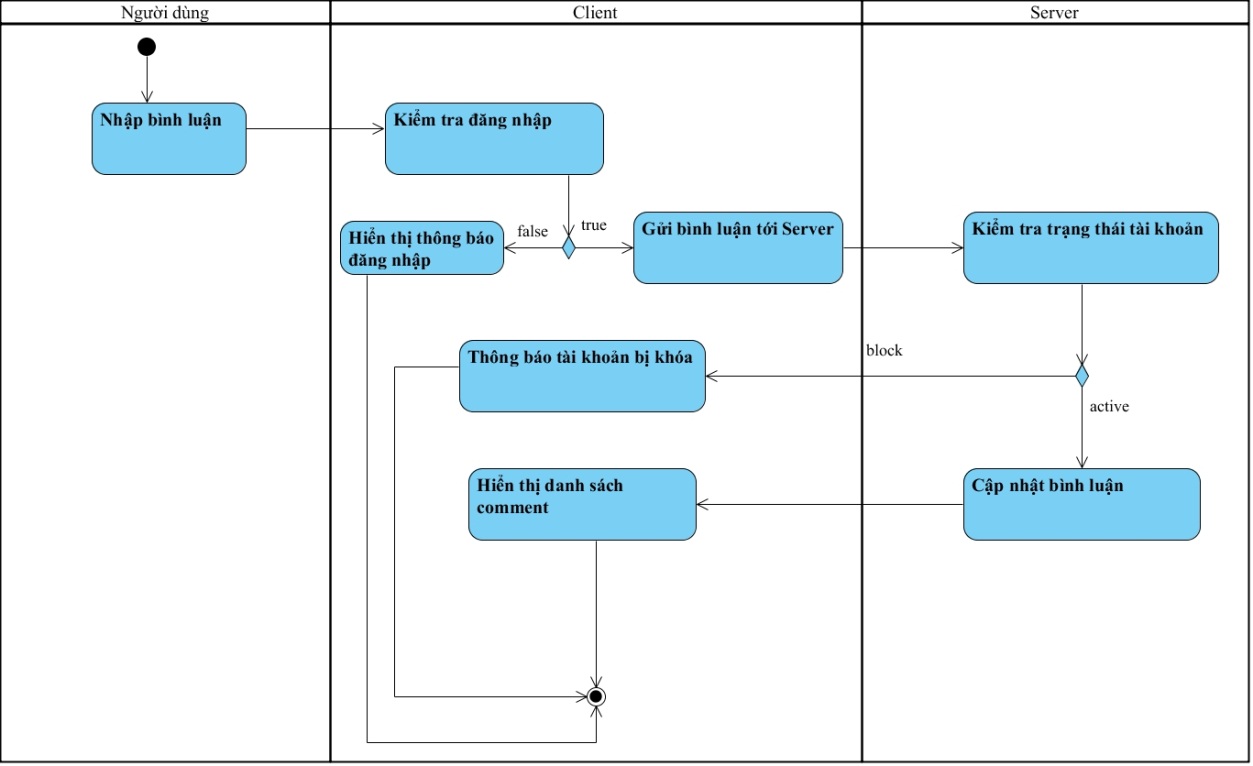


Hình .Biểu đồ hoạt động chức năng xem đọc truyện audio

Mô tả:

* Người dùng chọn chương.
* Client request tới Server yêu cầu đường link của chương .
* Server truy vấn CSDL và trả về kết quả .
* Client play audio .

### 3.6.4.Biểu đồ hoạt động cho chức năng Comment

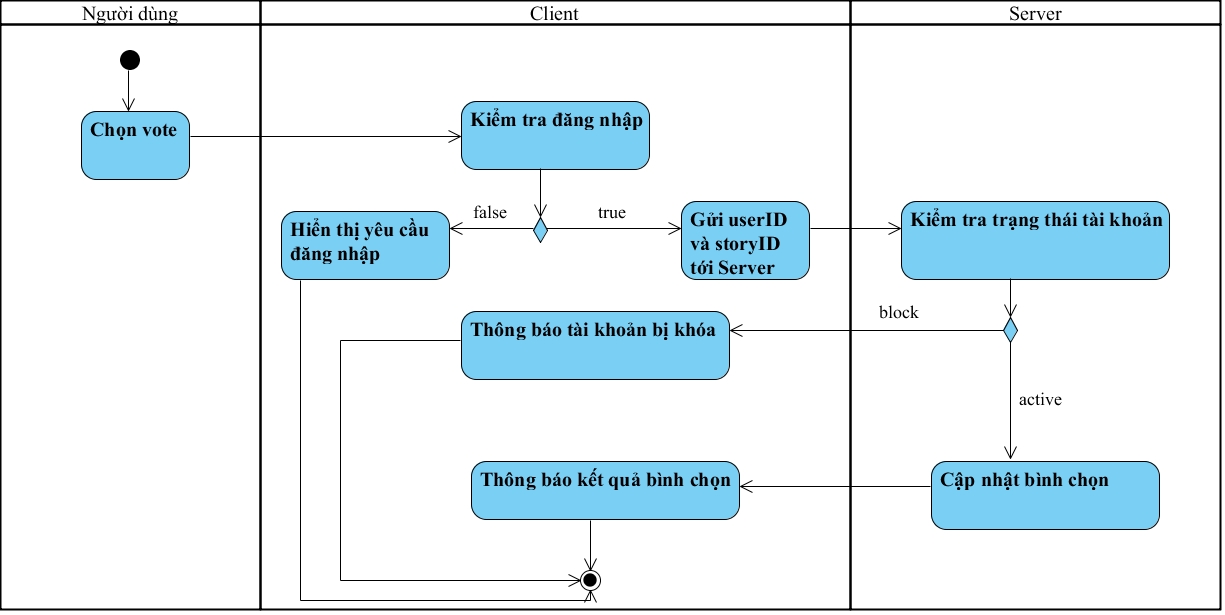


Hình .Biểu đồ hoạt động chức năng bình luận

Mô tả:

* Người dùng nhập comment và ấn submit
* Client tiến hành kiểm tra đăng nhập để biết người dùng đăng nhập chưa .Nếu chưa, yêu cầu người dùng đăng nhập.Ngược lại , gửi comment lên Server để xử lý .
* Server tiến hành kiểm tra trạng thái tài khoản của người dùng . Nếu tài khoản không bị khóa, tiến hành cập nhật bình luận .
* Client phân tích trả lời từ server . Nếu bình luận thành công, hiển thị danh sách bình luận. Nếu tài khoản người dùng bị khóa , hiển thị thông báo cho người dùng.

### 3.6.5.Biểu đồ hoạt động cho chức năng Vote Story

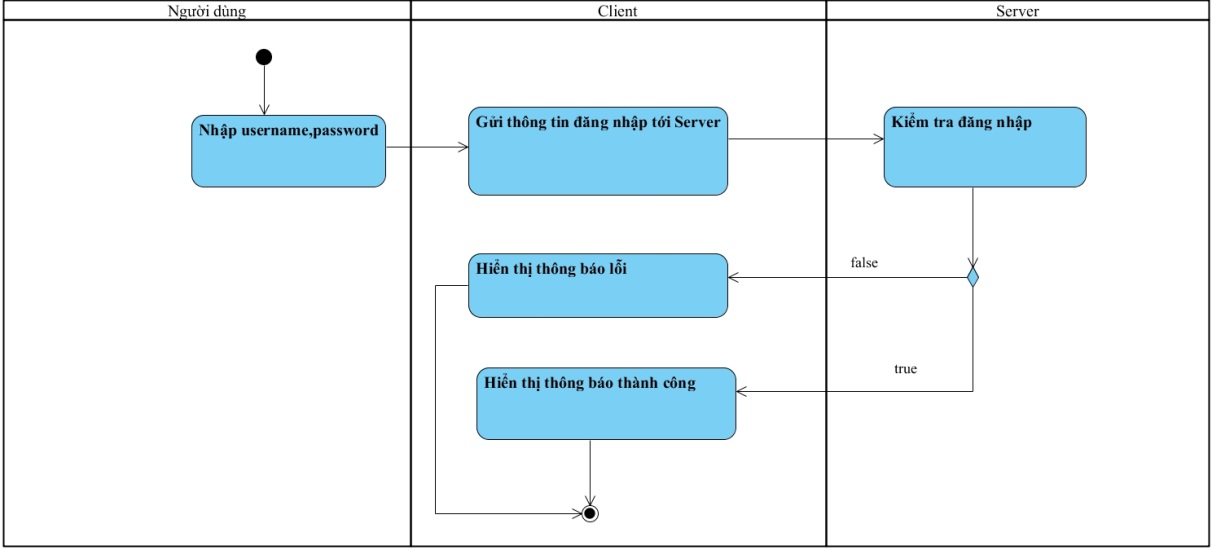


Hình .Biểu đồ hoạt động chức năng vote truyện yêu thích

Mô tả:

* Người dùng nhập chọn vote
* Client tiến hành kiểm tra đăng nhập để biết người dùng đăng nhập chưa .Nếu chưa, yêu cầu người dùng đăng nhập.Ngược lại , gửi userID và StoryID lên Server để xử lý .
* Server tiến hành kiểm tra trạng thái tài khoản của người dùng . Nếu tài khoản không bị khóa, tiến hành cập nhật vote .
* Client phân tích trả lời từ server . Nếu vote thành công, hiển thị thông báo thành công. Nếu tài khoản người dùng bị khóa , hiển thị thông báo cho người dùng.

### 3.6.6.Biểu đồ hoạt động cho chức năng Login

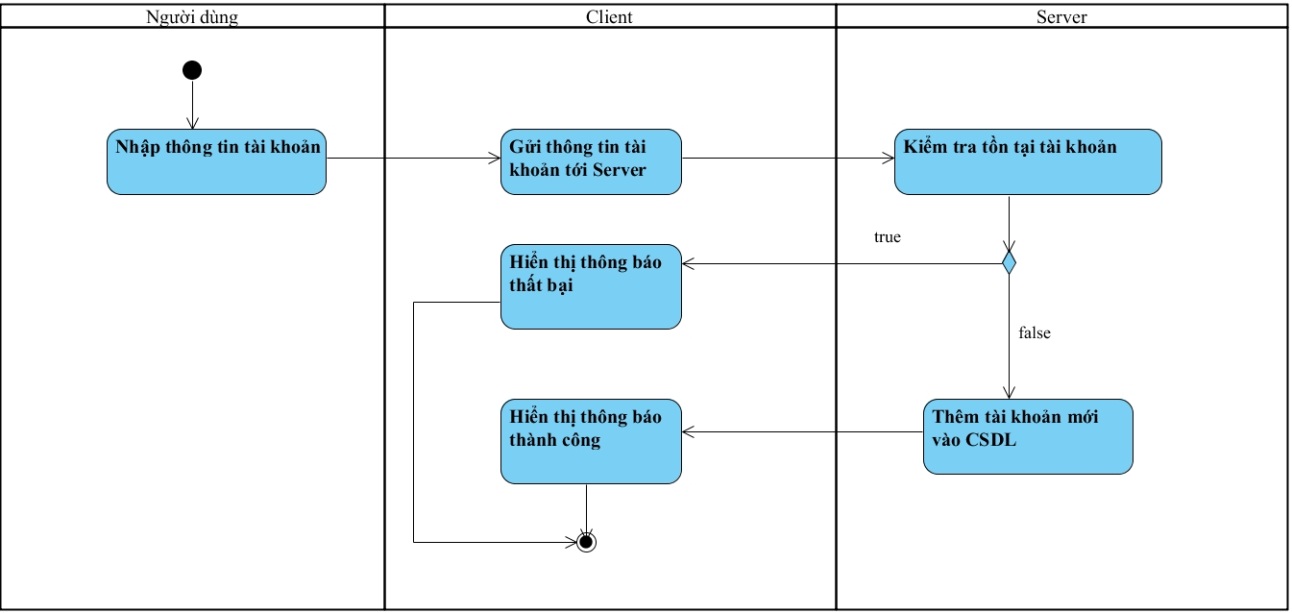


Hình .Biểu đồ hoạt động chức năng đăng nhập

Mô tả:

* Người dùng nhập username và password
* Client gửi thông tin đăng nhập lên Server để xử lý .
* Server tiến hành kiểm tra tài khoản đăng nhập và trả về kết quả cho client.
* Client phân tích trả lời từ server . Nếu đăng nhập thành công, hiển thị thông báo thành công. Nếu đăng nhập thất bại , hiển thị thông báo lỗi cho người dùng.

### 3.6.7.Biểu đồ hoạt động cho chức năng Register new account

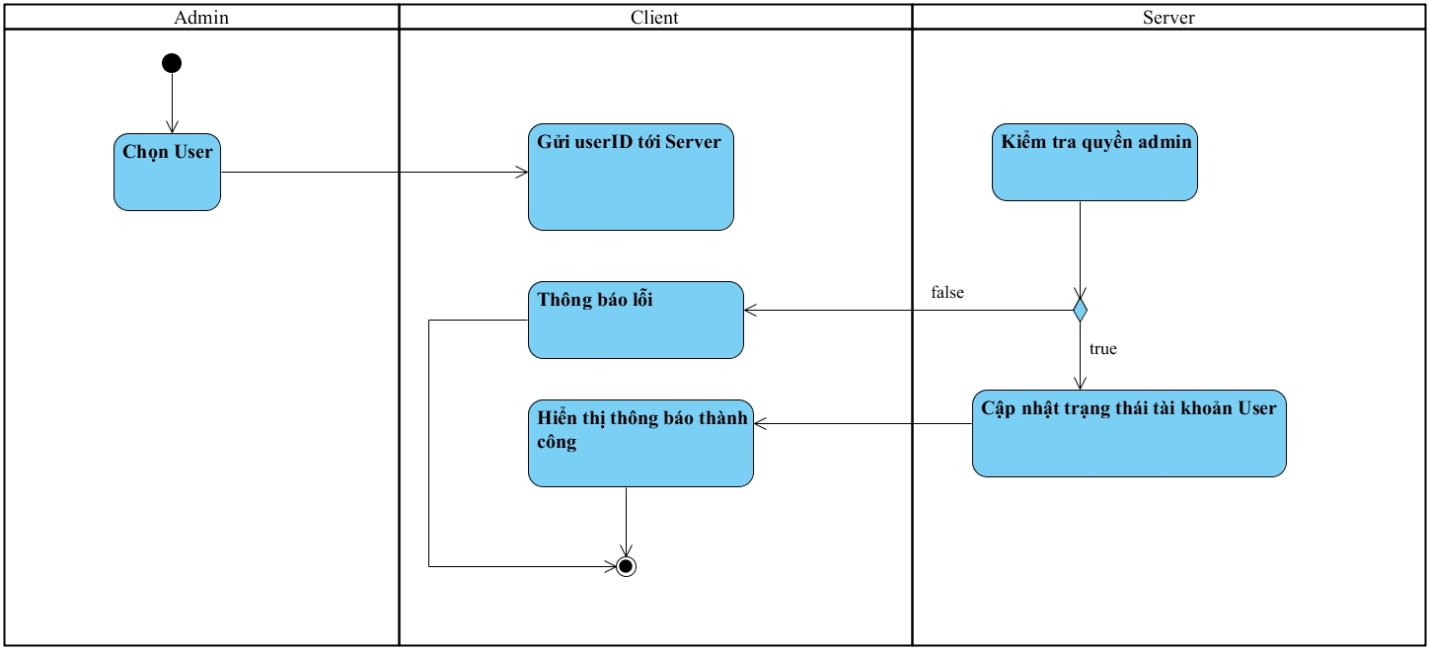


Hình .Biểu đồ hoạt động chức năng đăng ký tài khoản

Mô tả:

* Người dùng nhập thông tin tài khoản
* Client gửi thông tin tài khoản lên Server để xử lý .
* Server tiến hành kiểm tra đã tồn tại người dùng này chưa. Nếu chưa tồn tại thì tiến hành thêm mới vào CSDL.
* Client phân tích trả lời từ server . Nếu đăng ký thành công, hiển thị thông báo thành công. Nếu đăng ký thất bại , hiển thị thông báo lỗi cho người dùng.

### 3.6.8.Biểu đồ hoạt động cho chức năng Block account

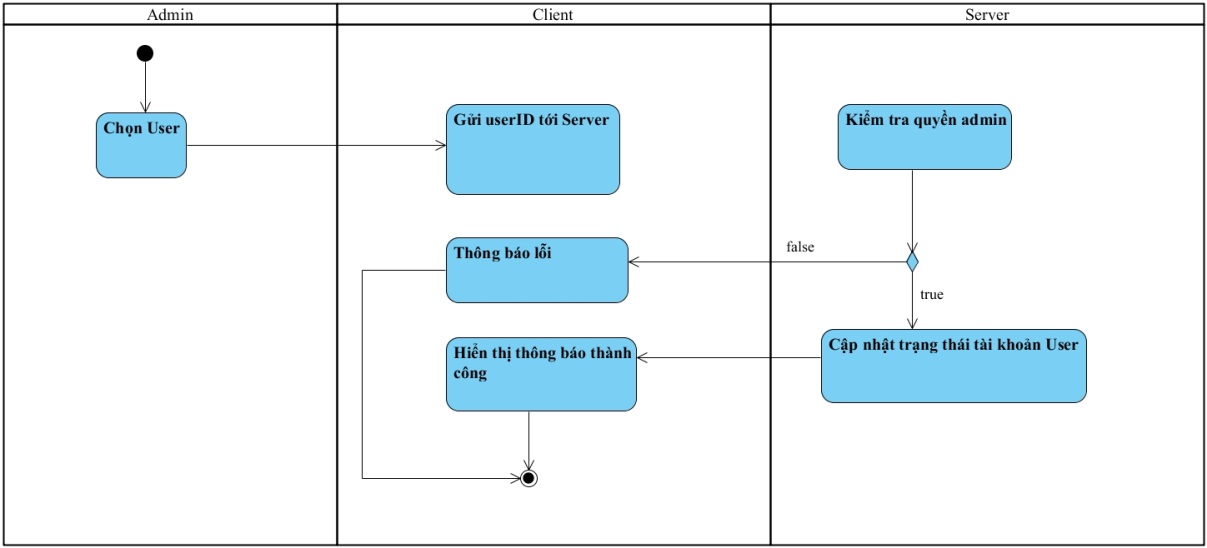


Hình .Biểu đồ hoạt động chức năng khóa tài khoản

Mô tả:

* Admin chọn user muốn khóa tài khoản
* Client gửi thông tin user lên Server để xử lý .
* Server tiến hành kiểm tra người dùng có quyền admin hay không. Nếu có , tiến hành cập nhật trạng thái user .
* Client phân tích trả lời từ server . Nếu việc khóa tài khoản thành công, hiển thị thông báo thành công. Nếu thất bại , hiển thị thông báo người dùng hiện tại không có quyền admin.

### 3.6.9.Biểu đồ hoạt động cho chức năng Active account



Hình .Biểu đồ hoạt động chức năng mở khóa tài khoản

Mô tả:

* Admin chọn user muốn mở khóa tài khoản
* Client gửi thông tin user lên Server để xử lý .
* Server tiến hành kiểm tra người dùng có quyền admin hay không. Nếu có , tiến hành cập nhật trạng thái user .
* Client phân tích trả lời từ server . Nếu việc mở khóa tài khoản thành công, hiển thị thông báo thành công. Nếu thất bại , hiển thị thông báo người dùng hiện tại không có quyền admin.

### 3.6.10.Biểu đồ hoạt động cho chức năng Delete Comment



Hình .Biểu đồ hoạt động chức năng xóa bình luận

Mô tả:

* Admin chọn comment muốn xóa.
* Client gửi thông tin comment lên Server để xử lý .
* Server tiến hành kiểm tra người dùng có quyền admin hay không. Nếu có , xóa comment .
* Client phân tích trả lời từ server . Nếu việc xóa comment thành công, hiển thị thông báo thành công. Nếu thất bại , hiển thị thông báo người dùng hiện tại không có quyền admin.

## 3.7.Thiết kế CSDL

Cơ sở dữ liệu gồm có 9 bảng :

* Bảng story:lưu trữ truyện

|  |  |  |  |
| --- | --- | --- | --- |
| Name | DataType | Constraints | Documentation |
| id | integer(10) | PK | khóa chính |
| name | varchar(255) |  | Tên của truyện |
| description | varchar(255) |  | Mô tả về truyện |
| image | varchar(255) |  | Ảnh của truyện |
| is\_audio | tinyint(1) |  | Có phải audio hay không |
| type | tinyint(1) |  | Loại tài khoản |

Bảng .Bảng story

* Bảng category: lưu trữ thể loại truyện

|  |  |  |  |
| --- | --- | --- | --- |
| Name | DataType | Constraints | Documentation |
| id | integer(10) | PK | khóa chính |
| name | varchar(255) |  | Tên của thể loại |
| image | varchar(255) |  | Ảnh mô tả cho thể loại |

Bảng .Bảng category

* Bảng author: lưu trữ tác giả

|  |  |  |  |
| --- | --- | --- | --- |
| Name | DataType | Constraints | Documentation |
| id | integer(10) | PK | khóa chính |
| name | varchar(255) |  | Tên của tác giả |
| description | varchar(255) |  | Mô tả về tác giả |
| image | varchar(255) |  | Ảnh của tác giả |

Bảng .Bảng author

* Bảng chapter:lưu trữ về các chap của truyện

|  |  |  |  |
| --- | --- | --- | --- |
| Name | DataType | Constraints | Documentation |
| id | integer(10) | PK | khóa chính |
| name | varchar(255) |  | Tên của chapter |
| story\_id | integer(10) | FK(story.id) | story id |
| path | varchar(255) |  | Đường dẫn tới file chapter |

Bảng .Bảng chapter

* Bảng story\_author: liên kết bảng story và author

|  |  |  |  |
| --- | --- | --- | --- |
| Name | DataType | Constraints | Documentation |
| story\_id | integer(10) | FK(story.id) | story id |
| author\_id | integer(10) | FK(author.id) | author id |

Bảng .Bảng story\_author

* Bảng story\_category: liên kết bảng story và category

|  |  |  |  |
| --- | --- | --- | --- |
| Name | DataType | Constraints | Documentation |
| story\_id | integer(10) | FK(story.id) | story id |
| category\_id | integer(10) | FK(author.id) | category\_id |

Bảng .Bảng story\_category

* Bảng user: lưu trữ thông tin người dùng

|  |  |  |  |
| --- | --- | --- | --- |
| Name | DataType | Constraints | Documentation |
| id | integer(10) | PK | khóa chính |
| username | varchar(255) |  | Tên đăng nhập |
| password | varchar(255) |  | Mật khẩu |
| api\_key | varchar(32) |  | Key để thao tác với service |
| avatar | varchar(255) |  | Ảnh đại diện |
| description | varchar(255) |  | Mô tả về bản thân |

Bảng .Bảng user

* Bảng vote: lưu trữ vote của user cho story

|  |  |  |  |
| --- | --- | --- | --- |
| Name | DataType | Constraints | Documentation |
| story\_id | integer(10) | FK(story.id) | story id |
| user\_id | integer(10) | FK(user.id) | user id |

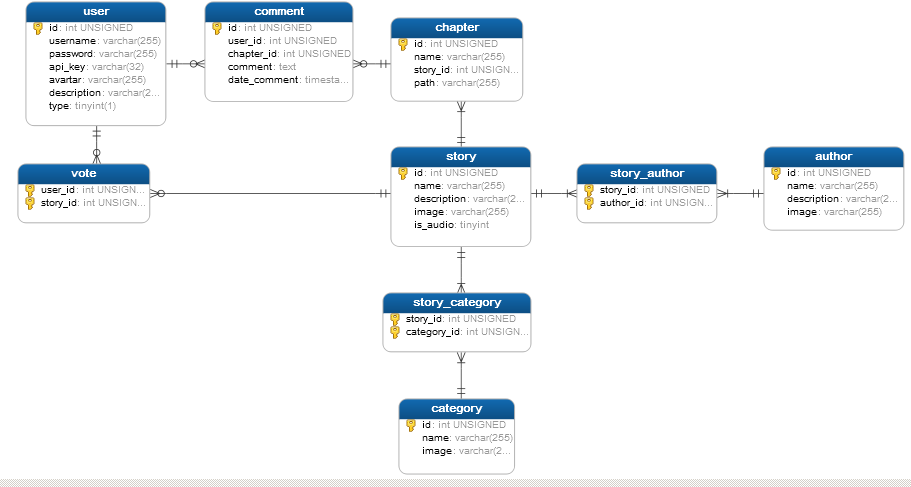
Bảng .Bảng Vote

* Bảng comment: lưu trữ comment của user về các chap

|  |  |  |  |
| --- | --- | --- | --- |
| Name | DataType | Constraints | Documentation |
| id | integer(10) | PK | khóa chính |
| user\_id | integer(10) | FK(user.id) | user id |
| chapter\_id | integer(10) | FK(chapter.id) | chapter id |
| comment | text |  | Nội dung comment |
| date\_comment | timestamp |  | Thời gian comment |

Bảng .Bảng comment

* Biểu đồ thiết kế CSDL



Hình .Biểu đồ thiết kế CSDL

# CHƯƠNG IV: TRIỂN KHAI ỨNG DỤNG

# 4.1.Môi trường triển khai

* Client được cài đặt trên Android 4.2
* Server được đặt tại localhost và chạy trên Server Apache
* Môi trường phát triển là trên máy tính cá nhân (Desktop hoặc Laptop) sử dụng hệ điều hành Windows có cài đặt máy ảo JAVA JRE version 7 và môi trường phát triển Android.Cấu hình tối thiểu để phát triển hệ thống là Desktop (hoặc Laptop) sử dụng chip intel Pentium 4 trở lên (hoặc tương dương),RAM từ 1 GB trở lên.
* Công cụ phát triển là NETBEAN 6.8 , Eclipse for JAVA developper đã cài Android SDK .

# 4.2. Cách thức triển khai và sử dụng hệ thống

* Triển khai Server chạy WebService : để triển khai được Web Service trên Server,hệ thống trên Server phải có
* Apache Server : đóng vai trò Web Server có chức năng xử lý Http Request
* Chrome REST Client Extension để kiểm tra các chức năng của Web Service đã tạo.
* Triển khai phía client

- Chạy ứng dụng Android bằng simulator cung cấp bởi Android SDK hoặc máy điện thoại thật .

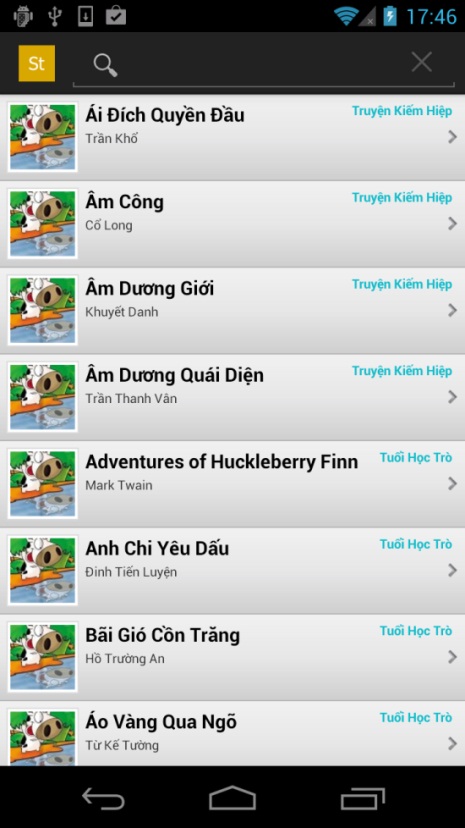
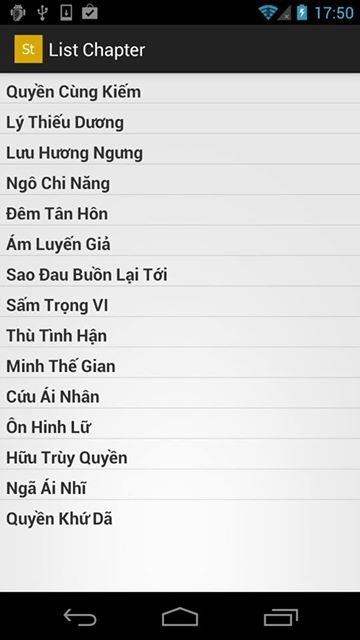
# 4.3. Kịch bản demo

**Kịch bản người dùng:**

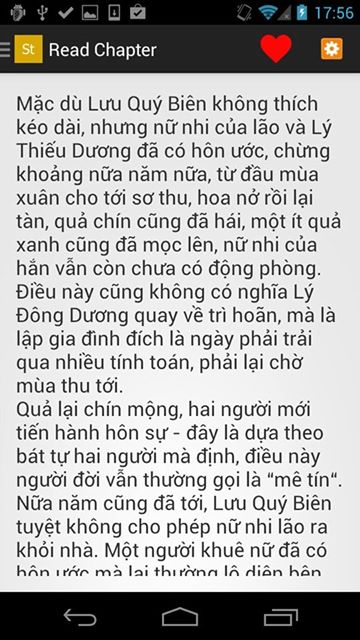
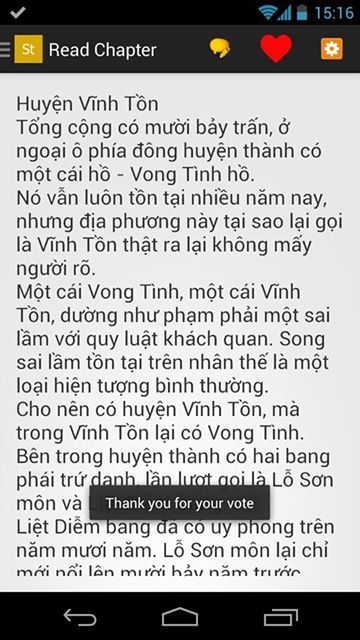
Với 2 người dùng đồng thời vào ứng dụng , một trên emulator và 1 trên máy điện thoại thật , chúng ta sẽ demo các chức năng:

* Xem danh sách truyện
* Xem danh sách chương của truyện
* Đọc truyện
* Bình luận
* Yêu thích truyện
* Đăng nhập
* Đăng ký tài khoản

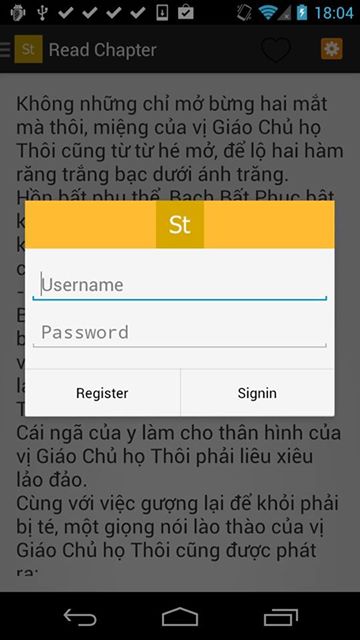
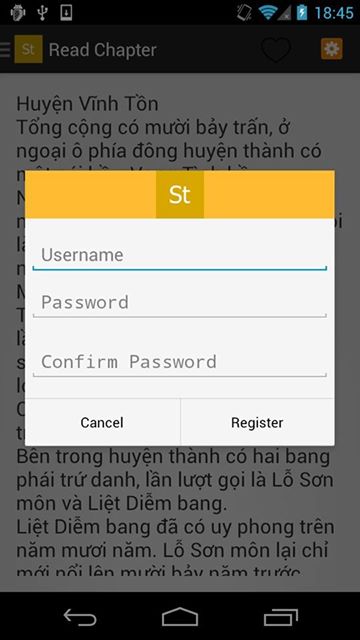
Một vài hình ảnh demo

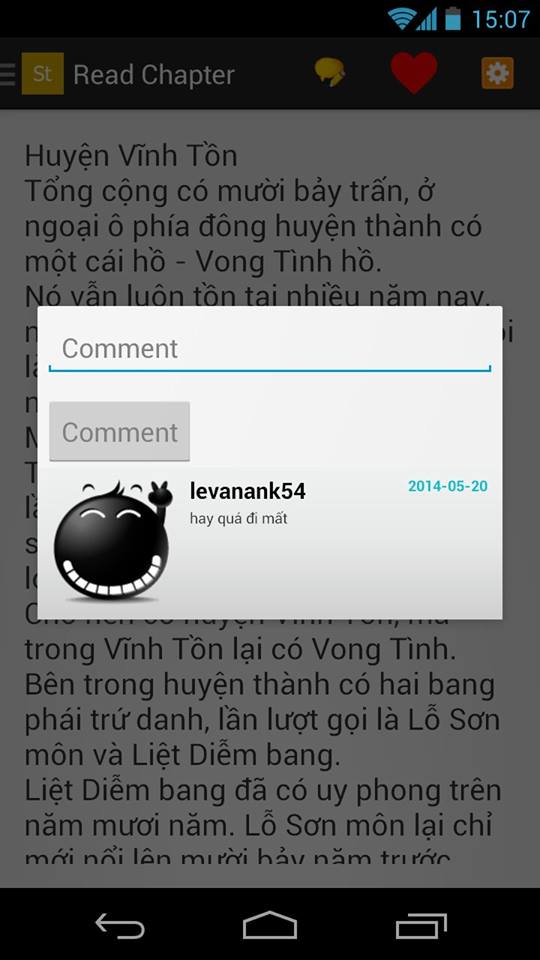
Hình .Demo chức năng show list story,list chapter

Hình .Demo chức năng đọc truyện và vote truyện

Hình .Demo chức năng đăng nhập và đăng ký

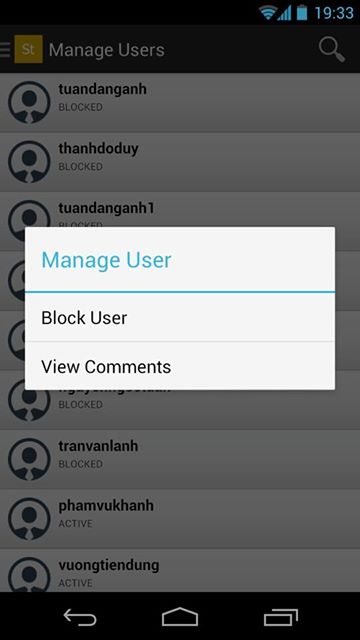
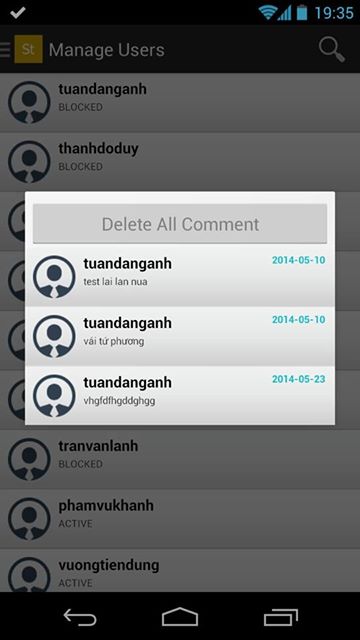
Hình .Demo chức năng nghe nhạc và bình luận

Nhận xét : tốc độ tải truyện, đọc truyện audio rất tốt . Đối với truyện dạng audio khi cả 2 người dùng đồng thời nghe cùng 1 truyện thì tốc độ tải gần như không thay đổi . Bitrate của truyện audio là 128kbps, khi thử nghiệm trên hosting miễn phí có tốc độ đường truyền 10 mbps thì các chức năng nghe,đọc truyện vẫn rất mượt, thời gian chờ ngắn.

Ưu điểm của Progressive Download là người dùng có thể nghe lại audio nhiều lần mà không cần tải lại nội dung , bên cạnh đó cũng có nhược điểm là nếu người dùng chỉ nghe 1 phần nhỏ audio thì vẫn tải cho tới khi hết nội dung audio và không có cơ chế chuyển đổi bitrate khi đường truyền người dùng yếu .

**Kịch bản của admin:** admin đăng nhập vào hệ thống và tiến hành khóa,mở khóa tài khoản, cũng như xóa comment của người dùng.

Một vài hình ảnh demo

Hình .Demo chức năng khóa tài khoản và xóa bình luận

Nhận xét: Việc khóa, mở khóa , xóa bình luận của người dùng diễn ra nhanh chóng, độ trễ thấp .

# CHƯƠNG V: KẾT LUẬN

# 5.1.Các kết quả đạt được

## 5.1.1.Ưu điểm

Đồ án đã hoàn thành các mục tiêu đề ra. Các vấn đề mà đồ án này đã đạt được như sau:

* Hiểu được cơ chế làm việc của WebService
* Xây dựng cơ bản hoàn chỉnh ứng dụng Story, đáp ứng đầy đủ các chức năng đã đề ra ban đầu.
* Hệ thống WebService chạy ổn định với tốc độ lấy dữ liệu trong thời gian cho phép.
* Client hiển thị thông nhanh và chính xác ngay khi dữ liệu được lấy về.
* Giao diện chương trình trực quan, đơn giản, thể hiện đầy đủ thông tin về truyện.
* Phần mềm đã được test trên các kích thước màn hình khác nhau như HVGA, QVGA …
* Chạy tốt với các bản android 4.0.4, 4.1 và mới hơn.

## 5.1.2.Nhược điểm

* Chưa có chức năng upload ảnh người dùng , đọc offline.
* Hệ thống chưa có modul quản lý để cập nhật truyện.
* Chưa hệ thống số lượt view của mỗi truyện.
* Chưa có chia sẻ truyện qua mạng xã hội .
* Hệ thống gợi ý truyện dựa trên sở thích .

# 5.2.Hướng phát triển trong tương lai

Hệ thống đã đạt được các yêu cầu đề ra nhưng vẫn còn những hạn chế vì thế hướng phát triển tiếp theo cần cải thiện khắc phục các hạn chế trên:

* Hoàn thiện chức năng upload ảnh người dùng , đọc offline.
* Xây dựng hệ thống modul quản lý để cập nhật truyện.
* Cập nhật lượt view của mỗi truyện .
* Chia sẻ truyện qua mạng xã hội .
* Xây dựng chức năng thông báo cho người dùng biết khi có truyện mới: có thể dùng Google Cloud Messaging (GCM) để thông báo Client biết khi có sự thay đổi thông tin ở Server.
* Xây dựng chức năng gợi ý truyện, kết bạn .

**TÀI LIỆU THAM KHẢO**

1. google.com, [http://developer.android.com](http://developer.android.com/), <http://www.wikipedia.org/> , [http://vietandroid.com](http://vietandroid.com/)
2. <http://en.wikipedia.org/wiki/Agile_software_development>
3. Scrum Guide - Ken Schwaber and Jeff Sutherland
4. Profesional Android Development
5. Beginning Android
6. [Androidhive.info](http://www.androidhive.info/)

**MỤC LỤC HÌNH VẼ, BẢNG BIỂU**

[Hình 1.Mô hình kiến trúc nền tảng Android 9](#_Toc388406243)

[Hình 2.Mô hình hợp tác giữa máy ảo Dalvik và Navite code 11](#_Toc388406244)

[Hình 3.Biểu đồ so sánh tốc độ của máy ảo Java giữa các phiên bản Android (Càng lớn càng tốt) 13](#_Toc388406245)

[Hình 4. So sánh giao diện nguyên bản của Android và giao diện Sense của riêng hãng HTC thiết kế 14](#_Toc388406246)

[Hình 5.Thống kê số lượng điện thoại sử dụng các phiên bản Android 14](#_Toc388406247)

[Hình 6.Cấu trúc thư mục và file của một dự án phần mềm Android trên Eclipse 16](#_Toc388406248)

[Hình 7.Biểu đồ phân cấp chức năng 23](#_Toc388406249)

[Hình 8.Biểu đồ Use case của ứng dụng 24](#_Toc388406250)

[Hình 9.Biểu đồ tuần tự chức năng Show List Story 27](#_Toc388406251)

[Hình 10.Biểu đồ tuần tự chức năng Show List Chapter 28](#_Toc388406252)

[Hình 11.Biểu đồ tuần tự chức năng Read Chapter(text) 29](#_Toc388406253)

[Hình 12.Biểu đồ tuần tự chức năng Read Chapter(audio) 29](#_Toc388406254)

[Hình 13.Biểu đồ tuần tự chức năng Comment 30](#_Toc388406255)

[Hình 14.Biểu đồ tuần tự chức năng Vote Story 31](#_Toc388406256)

[Hình 15.Biểu đồ tuần tự chức năng Login 32](#_Toc388406257)

[Hình 16.Biểu đồ tuần tự chức năng Register account 33](#_Toc388406258)

[Hình 17.Biểu đồ lớp phía Client 34](#_Toc388406259)

[Hình 18.Biểu đồ lớp phía Server 35](#_Toc388406260)

[Hình 19.Biểu đồ hoạt động chức năng xem List Story 36](#_Toc388406261)

[Hình 20.Biểu đồ hoạt động chức năng xem List Chapter 37](#_Toc388406262)

[Hình 21.Biểu đồ hoạt động chức năng xem đọc truyện text 38](#_Toc388406263)

[Hình 22.Biểu đồ hoạt động chức năng xem đọc truyện audio 39](#_Toc388406264)

[Hình 23.Biểu đồ hoạt động chức năng bình luận 39](#_Toc388406265)

[Hình 24.Biểu đồ hoạt động chức năng vote truyện yêu thích 40](#_Toc388406266)

[Hình 25.Biểu đồ hoạt động chức năng đăng nhập 41](#_Toc388406267)

[Hình 26.Biểu đồ hoạt động chức năng đăng ký tài khoản 42](#_Toc388406268)

[Hình 27.Biểu đồ thiết kế CSDL 46](#_Toc388406269)

[Hình 28.Demo chức năng show list story,list chapter 48](#_Toc388406270)

[Hình 29.Demo chức năng đọc truyện và vote truyện 48](#_Toc388406271)

[Hình 30.Demo chức năng đăng nhập và đăng ký 49](#_Toc388406272)

[Hình 31.Demo chức năng nghe nhạc và bình luận 49](#_Toc388406273)

[Bảng 1.Use case hiển thị danh sách truyện 27](#_Toc388644166)

[Bảng 2.Use case hiển thị danh sách các chương của truyện 27](#_Toc388644167)

[Bảng 3.Use case đọc truyện 28](#_Toc388644168)

[Bảng 4.Use case bình luận truyện 28](#_Toc388644169)

[Bảng 5.Use case vote truyện 28](#_Toc388644170)

[Bảng 6.Use case đăng nhập 29](#_Toc388644171)

[Bảng 7.Use case đăng ký tài khoản 29](#_Toc388644172)

[Bảng 8.Use case khóa tài khoản 30](#_Toc388644173)

[Bảng 9.Use case mở khóa tài khoản 30](#_Toc388644174)

[Bảng 10.Use case xóa bình luận 31](#_Toc388644175)

[Bảng 11.Bảng story 50](#_Toc388644176)

[Bảng 12.Bảng category 51](#_Toc388644177)

[Bảng 13.Bảng author 51](#_Toc388644178)

[Bảng 14.Bảng chapter 51](#_Toc388644179)

[Bảng 15.Bảng story\_author 51](#_Toc388644180)

[Bảng 16.Bảng story\_category 52](#_Toc388644181)

[Bảng 17.Bảng user 52](#_Toc388644182)

[Bảng 18.Bảng Vote 52](#_Toc388644183)

[Bảng 19.Bảng comment 53](#_Toc388644184)