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

**TÊN ĐỀ TÀI**

**XÂY DỰNG ỨNG DỤNG BÁO THỨC CHO**

**ĐIỆN THOẠI TRÊN**

**HỆ ĐIỀU HÀNH ANDROID**

Sinh viên thực hiện: **Đặng Xuân Hưng**

Lớp: CN-CNTT K57

Giáo viên hướng dẫn: ThS **Hoàng Anh Việt**

Hà Nội 05-2016

***Lời cảm ơn***

*Em xin cảm ơn thầy giáo – Thạc sỹ Hoàng Anh Việt đã chỉ bảo, giúp đỡ, tạo điều kiện cho em hoàn thành đồ án này.*

*Em xin cảm ơn các thầy cô trong Viện Công nghệ thông tin & truyền thông và trong toàn trường Đại học Bách Khoa Hà Nội đã giúp đỡ em trong suốt các năm theo học tại trường, dạy bảo những kiến thức để làm nền tảng cho em thực hiện Đồ án này.*

***Hà Nội, ngày 17 tháng 5 năm 2016***

***Sinh viên thực hiện:***

***Đặng Xuân Hưng***

Contents

[CHƯƠNG I: ĐẶT VẤN ĐỀ 6](#_Toc451338358)

[1.1 Khảo sát thực tế 6](#_Toc451338359)

[1.2 Nội dung đề tài 6](#_Toc451338360)

[1.3 Bố cục đồ án 6](#_Toc451338361)

[Chương II: CƠ SỞ LÝ THUYẾT VÀ ĐỊNH HƯỚNG GIẢI PHÁP 7](#_Toc451338362)

[2.1. Giới thiệu hệ điều hành Android 7](#_Toc451338363)

[2.2.Kiến trúc Android 9](#_Toc451338364)

[2.3. Phát triển ứng dụng trên Android 12](#_Toc451338365)

[2.3.1. Ngôn ngữ lập trình 12](#_Toc451338366)

[2.3.2. Vấn đề phân mảnh phiên bản Android trên thị trường: 12](#_Toc451338367)

[2.3.3. Vấn đề kích thước màn hình 14](#_Toc451338368)

[2.3.4.Môi trường lập trình cho Android 14](#_Toc451338369)

[2.3.5. Android Studio 14](#_Toc451338370)

[2.3.6. Các thành phần có bản của một project trên Android Studio 15](#_Toc451338371)

[CHƯƠNG III: XÂY DỰNG ỨNG DỤNG 18](#_Toc451338372)

[BÁO THỨC 18](#_Toc451338373)

[3.1 Phân tích các chức năng 18](#_Toc451338374)

[3.2. Biểu đồ Use case 23](#_Toc451338375)

[3.2.1. Use case hiển thị danh sách báo thức 23](#_Toc451338376)

[3.2.2. Use case thêm mới một báo thức 24](#_Toc451338377)

[3.2.3. Use case chỉnh sửa báo thức 24](#_Toc451338378)

[3.2.4. Use case xóa báo thức 25](#_Toc451338379)

[3.2.5. Use case tắt báo thức 25](#_Toc451338380)

[3.2.6. Use case đăng nhập 26](#_Toc451338381)

[3.3. Biểu đồ tuần tự 27](#_Toc451338382)

[3.3.1. Biểu đồ tuần tự chức năng hiển thị danh sách báo thức 27](#_Toc451338383)

[3.3.2. Biểu đồ tuần tự chức năng thêm mới báo thức 28](#_Toc451338384)

[3.3.3. Biểu đồ tuần tự chức năng chỉnh sửa báo thức 29](#_Toc451338385)

[3.3.4 Biểu đồ tuần tự chức năng xóa báo thức 30](#_Toc451338386)

[3.3.5. Biểu đồ tuần tự chức năng thay đổi trạng thái on/off báo báo thức 31](#_Toc451338387)

[3.3.6. Biểu đồ tuần tự chức năng nghe nhạc tắt báo thức 32](#_Toc451338388)

[3.3.7. Biểu đồ tuần tự chức năng tắt báo thức mặc định 33](#_Toc451338389)

[3.4 Biểu đồ hoạt động 34](#_Toc451338390)

[3.4.1 .Biểu đồ hoạt động cho chức năng xem danh sách báo thức 34](#_Toc451338391)

[3.4.2 Biểu đồ hoạt động cho chức năng thêm mới báo thức 35](#_Toc451338392)

[3.4.3 Biểu đồ hoạt động cho chức năng sửa báo thức 36](#_Toc451338393)

[3.4.4 Biểu đồ hoạt động cho chức năng xóa báo thức 37](#_Toc451338394)

[3.4.5. Biểu đồ hoạt động cho chức năng thay đổi trạng thái báo thức 38](#_Toc451338395)

[3.4.6 Biểu đồ hoạt động cho chức năng tắt báo thức 39](#_Toc451338396)

[3.5 Thiết kế CSDL 40](#_Toc451338397)

[CHƯƠNG IV: TRIỂN KHAI ỨNG DỤNG 43](#_Toc451338398)

[4.1 Môi trường triển khai 43](#_Toc451338399)

[4.2. Cách thức triển khai và sử dụng hệ thống 43](#_Toc451338400)

[4.3. Kịch bản deme 43](#_Toc451338401)

[CHƯƠNG V: KẾT LUẬN 47](#_Toc451338402)

[5.1 Các kết quả đạt được 47](#_Toc451338403)

[5.1.1.Ưu điểm 47](#_Toc451338404)

[5.1.2 Nhược điểm 47](#_Toc451338405)

[5.2 Hướng phát triển trong tương lai 47](#_Toc451338406)

[**TÀI LIỆU THAM KHẢO** 48](#_Toc451338407)

[**DANH MỤC CÁC HÌNH ẢNH** 48](#_Toc451338408)

[**DANH MỤC CÁC BẢNG** 49](#_Toc451338409)

# CHƯƠNG I: ĐẶT VẤN ĐỀ

## Khảo sát thực tế

Hiện nay, với sự phát triển về kinh tế, xã hội, gần như mỗi người từ học sinh, sinh viên, các bậc phụ huynh đến cả những người cao tuổi đều có thể sở hữu một chiếc Smart Phone. Vì vậy nhu cầu công việc, giải trí từ các ứng dụng trên di động là rất lớn và gần như là vô tận. Trong đó nhu cầu đặt báo thức, hẹn giờ cũng là khá phổ biến. Mỗi chiếc Smart Phone đều được trang bị sẵn một ứng dụng báo thức mặc định. Nhưng gần như nó quá nhàm chán, và đôi khi không đáp ứng đúng nhu cầu là báo thức của người dung. Tình trạng học sinh, sinh viên, hay người đi làm thức đêm để rồi sang hôm sau “ngủ nướng” là rất phổ biến. Một báo thức thông thường chỉ khiến họ tỉnh dậy và tắt báo thức bởi một thao tác chạm đơn giản rồi tiếp tục giấc ngủ. Đây là môt thói quen không hề tốt. Dấu hỏi đặt ra: Liệu có thể thay đổi báo thức nhàm chán bằng một báo thức thú vị, đồng thời thay đổi được thói quen không tốt đó của người dùng?

## Nội dung đề tài

**Tên đề tài**

Xây dựng ứng dụng báo thức thông minh trên cho di động trên hệ điều hành Android

**Nhiệm vụ của đề tài**

Đề tài nhằm xây dựng một ứng dụng báo thức có tính giải trí, và mang lại hiệu quả báo thức cho người dùng. Người dùng có thể thêm, sửa, xóa một báo thức như bình thường. Và khi chuông báo thức rung lên, người dùng sẽ được nghe những bản nhạc từ chính list nhạc trong điện thoại của mình. Nhiệm vụ của người dùng là phải trả lời đúng tên của 3 bản nhạc, nếu không những bản nhạc này sẽ phát mãi mãi. Tất nhiên giấc ngủ sẽ rất khó được tiếp tục.

Đây là ứng dụng trải nghiệm đầu tiên của em về hệ điều hành Android, do còn hạn chế về kiến thức nên ứng dụng sẽ không tránh khỏi những vấn đề về thiết kế, chức năng… Em mong được sự góp ý của thầy cô để hoàn thiện ứng dụng cũng như có kinh nghiệm cho những ứng dụng tiếp theo.

Em xin chân thành cảm ơn thầy Hoàng Anh Việt đã hướng dẫn, giúp đỡ em trong quá trình thực hiện đề tài này.

## 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 báo thức**

**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 DataBaseManagement System nhỏ gọn và mạnh mẽ.

Android Runtime

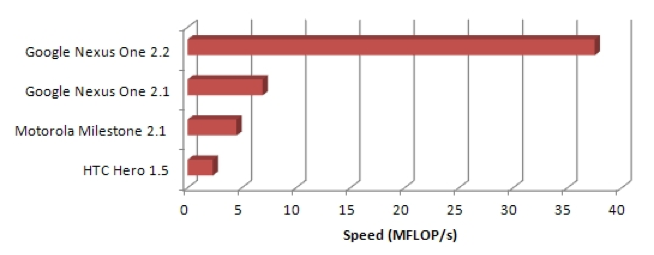
Hệ điều hành Android tích hợp sẳn một tập hợp các thư viện cốt lõi cung cấp hầu hết các chức năng có sẵn trong các thư viện lõi của ngôn ngữ lập trình Java. Mọi ứng dụng của Android chạy trên một tiến trình của riêng nó cùng với một thể hiện của máy ảo Dalvik. Máy ảo Dalvik thực tế là một biến thể của 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.

## 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:

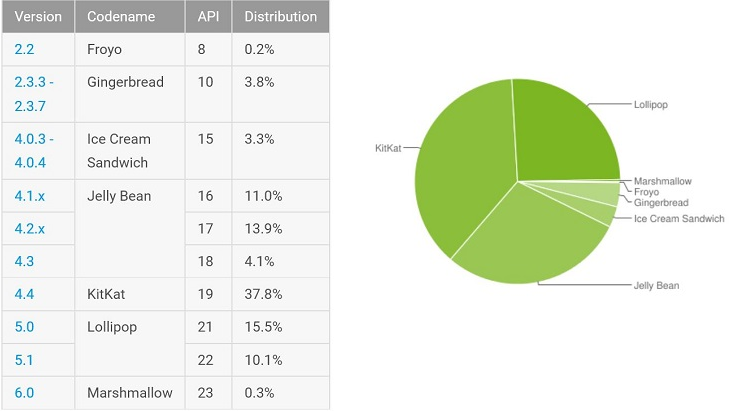
Do Android là hệ điều hành mã nguồn mở và miễn phí, bất kỳ một nhà sản xuất phần cứng nào cũng có thể sử dụng và tùy biến Android để cài đặt trên các thiết bị của mình. Điều này dẫn đến vấn để phân mảnh (fragmentation) khi trên thị trường tồn tạinhiềumẫu điện thoại Android, mỗi mẫu chạy một phiên bản 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 việc cập 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 thiết kế fileXML 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. Android Studio

Google cung cấp một công cụ phát triển ứng dụng Android trên Website chính thức dựa trên nền tảng IntelliJ IDEA gọi là Android Studio. Bộ công cụ Android Studio cung cấp những trình soạn thảo riêng biệt tương ứng với hầu hết các file cấu hình và Layout của ứng dụng Android với định dạng XML. Với những file Layout giao diện, Android Studio cho phép các lập trình viên dễ dàng chuyển đổi chế độ chỉnh sửa giữa trình biên soạn nội dung XML hoặc trình biên soạn dưới dạng giao diện (GUI).  
Ngoài ra , bộ phát triển Android Studio còn tích hợp bên trong những tiện ích hỗ trợ phát triển ứng dụng như sau:

Bộ xây dựng ứng dụng Gradle với nhiều cấu hình linh động.

Cho phép xây dựng ứng dụng tùy biến và tự động tạo ra file .apk tương thích với thông tin tùy biến.

Bộ code mẫu giúp chúng ta xây dựng các chức năng phổ biến của các ứng dụng.

Trình biên soạn Layout GUI cho ứng dụng Android phong phú và tiện lợi, cho phép người dùng dễ dàng tạo ra giao diện màn hình bằng cách kéo thả các Component mẫu có sẵn và chỉnh sửa giao diện themes (kích thướt, màu sắc, …) tùy ý.

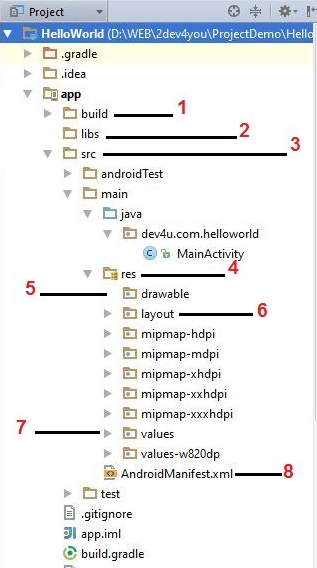
Tích hợp **lint** – Một ứng dụng tích hợp giúp các developer kiểm soát về hiệu suất (performance), tính khả dụng (usability), khả năng tương thích của các phiên bản API sử dụng, và những vấn đề tiềm ẩn bên trong có thể xảy ra lúc Runtime.

ProGuard (tiện ích tối ưu và mã hóa code khi build ứng dụng) và Android app-signing.

Bộ tích hợp hỗ trợ phát triển ứng dụng Android dễ dàng với các dịch vụ cùa nền tảng đám mây của Google

Mới đây , Google đã cho ra mắt phiên bản 2.0 , và ngay sau đó là 2.1 với các tính năng mới nổi bật là Instant Run, trình giả lập Android nhanh hơn, Cloud Test Lab và GPU Debugger.

### 2.3.6. Các thành phần có bản của một project trên Android Studio



Hình : Cấu trúc thư mục và file của một dự án phần mềm Android trên Android Studio

**build:**Thư mục này chứa các tập tin được tạo tự động như Aidl,Build configuration, R(R.JAVA) và thư mục outputs chứa file apk.

**libs:**Đây là một thư mục để thêm các thư viện để phát triển các ứng dụng Android.

**src:**Thư mục này chứa các file nguồn .java cho cho project của bạn. Mặc định, nó bao gồm file nguồn MainActivity.java.

**res:** Thư mục này chứa các thư mục con khác như drawable , layout, values….

**res/drawable:**Thư mục này chứa các hình ảnh hoặc các file xml được sử dụng trong ứng dụng, ví dụ 1 hình ảnh để làm background cho app…

**res/layout:**Đây là một thư mục chứa các tập tin định nghĩa giao diện người dùng trong ứng dụng của bạn.

**res/values:**Đây là một thư mục chứa các tập tin XML khác nhau chứa một tập hợp các nguồn lực, chẳng hạn như các chuỗi string trong  có trong ứng dụng và màu sắc .

**AndroidManifest.xml:**Đây là  file trong đó mô tả các đặc điểm cơ bản của ứng dụng và xác định từng thành phần của nó.

# CHƯƠNG III: XÂY DỰNG ỨNG DỤNG

# BÁO THỨC

## 3.1 Phân tích các chức năng

Ứng dụng thiết kế cho người dùng có thể cài đặt báo thức như những ứng dụng báo thức mặc định. Điểm khác biệt là ở chỗ cách tắt báo thức.

Cụ thể người dùng có thể:

* Xem danh sách các báo thức đã cài đặt
* Thêm báo thức mới.
* Sửa báo thức đã cài đặt
* Xóa báo thức
* Thay đổi trạng thái cho báo thức đã cài đặt : on/off
* Nghe nhạc và đoán tên khi muốn tắt báo thức.

**Chi tiết chức năng xem danh sách báo thức:**

|  |  |  |  |
| --- | --- | --- | --- |
| Thứ tự | Thao tác người dùng | Màn hình | Mô tả hệ thống |
| 1 | Mở ứng dụng |  | - Ứng dụng mở màn hình Main Activity.  - Hiển thị đồng hồ.  - Lấy danh sách báo thức từ cơ sở dữ liệu và hiển thị ra màn hình  - Với mỗi item hiển thị thời gian, title và một Switch Button thể hiện trạng thái on/off của báo thức. |

Bảng : Chi tiết chức năng xem danh sách báo thức

**Chi tiết chức năng thêm mới báo thức**

|  |  |  |  |
| --- | --- | --- | --- |
| Thứ tự | Thao tác người dùng | Màn hình | Mô tả hệ thống, Control |
| 1 | Click Add Button | MainActivity | -Ứng dụng mở màn hình AddAlarmActivity.  - Hiển thị thông tin cần cài đặt cho báo thức:  + TextView: hiển thị thời gian.  + EditText :.nhập Title  + TextView: hiển thị loại báo thức  ( mặc định hoặc chơi game)  + CheckBox: lựa chọn báo rung hoặc không.  + TextView: hiển thị âm báo thức lựa chọn. |
| 2 | Click TextView hiển thị thời gian | AddAlarmActivity | - Ứng dụng hiển thị TimeDialog cho người dùng cài đặt thời gian báo thức |
| 3 | Nhập Title cho báo thức vào EditText | AddAlarmActivity | - Hiển thị bàn phím cho người dùng nhập Title báo thức |
| 4 | Click TextView hiển thị loại báo thức | AddAlarmActivity | - Hiển thị AlertDiaglog cho người dùng lựa chọn loại báo thức: mặc định hoặc chơi game. |
| 5 | Check vào CheckBox chọn/bỏ chọn chế độ rung | AddAlarmActivity | - Xác nhận chế độ rung tương ứng trạng thái của checkbox. |
| 6 | Click TextView hiển thị âm báo | AddAlarmActivity | - Mở màn hình SelectRingActivity để người dùng chọn âm báo.  - Tại màn hình này, người dùng có thể lựa chọn âm báo có sẵn của ứng dụng, hoặc lấy từ danh sách bài hát trong điện thoại. |
| 7 | Chọn một âm báo thức | SelectRingActivity | - Trở về màn hình AddAlarmActivity và hiển thị tên âm báo đã chọn lên TextView âm báo. |
| 8 | Click Save Button | AddAlarmActivity | - Lưu thông tin báo thức vào cơ sở dữ liệu.  - Start Service SetAlarmService để cài đặt báo thức gần nhất cho điện thoại.  - Trở về màn hình MainActivity |
| 9 | Click Back Button | AddAlarmActivity | Trở về màn hình MainActivity |

Bảng : Chi tiết chức năng thêm mới báo thức

**Chi tiết chức năng chỉnh sửa báo thức**

|  |  |  |  |
| --- | --- | --- | --- |
| Thứ tự | Thao tác người dùng | Màn hình | Mô tả hệ thống, Control |
| 1 | Click chọn một item trên danh sách báo thức. | MainActivity | -Ứng dụng mở màn hình EddAlarmActivity.  - Hiển thị thông tin đã cài đặt cho báo thức:  + TextView: hiển thị thời gian.  + EditText :.nhập Title  + TextView: hiển thị loại báo thức  ( mặc định hoặc chơi game)  + CheckBox: báo rung hoặc không.  + TextView: hiển thị âm báo thức lựa chọn. |
| 2 | Click TextView hiển thị thời gian | EditAlarmActivity | - Ứng dụng hiển thị TimeDialog cho người dùng chỉnh sửa thời gian báo thức |
| 3 | Nhập Title cho báo thức vào EditText | EditAlarmActivity | - Hiển thị bàn phím cho người dùng chỉnh sửa Title báo thức |
| 4 | Click TextView hiển thị loại báo thức | EditAlarmActivity | - Hiển thị AlertDiaglog cho người dùng lựa chọn loại báo thức: mặc định hoặc chơi game. |
| 5 | Check vào CheckBox chọn/bỏ chọn chế độ rung | EditAlarmActivity | - Xác nhận chế độ rung tương ứng trạng thái của checkbox. |
| 6 | Click TextView hiển thị âm báo | EditAlarmActivity | - Mở màn hình SelectRingActivity để người dùng chọn âm báo.  - Tại màn hình này, người dùng có thể lựa chọn âm báo có sẵn của ứng dụng, hoặc lấy từ danh sách bài hát trong điện thoại. |
| 7 | Chọn một âm báo thức | SelectRingActivity | - Trở về màn hình EditAlarmActivity và hiển thị tên âm báo đã chọn lên TextView âm báo. |
| 8 | Click Save Button | AddAlarmActivity | - Update báo thức vào cơ sở dữ liệu.  - Start Service SetAlarmService để cài đặt báo thức gần nhất cho điện thoại.  - Trở về màn hình MainActivity |
| 9 | Click Back Button | AddAlarmActivity | Trở về màn hình MainActivity |

Bảng : Chi tiết chức năng sửa báo thức

**Chi tiết chức năng xóa báo thức:**

|  |  |  |  |
| --- | --- | --- | --- |
| Thứ tự | Thao tác người dùng | Màn hình | Mô tả hệ thống |
| 1 | Click chọn một item trên danh sách báo thức. | MainActivity | - Ứng dụng mở màn hình EditAlarmActivity  - Hiển thị đồng hồ thông tin báo thức đã cài đặt. |
| 2 | Click delete button | EditAlarmActivity | - Xóa báo thức trong cơ sở dữ liệu.  - Start Service SetAlarmService để cài đặt báo thức gần nhất cho điện thoại.  - Trở về màn hình MainActivity |

Bảng : Chi tiết chức năng xóa báo thức

**Chi tiết chức năng thay đổi trạng thái cho báo thức đã cài đặt: on/off**

|  |  |  |  |
| --- | --- | --- | --- |
| Thứ tự | Thao tác người dùng | Màn hình | Mô tả hệ thống |
| 1 | Vuốt Switch button tương ứng với mỗi báo thức trong danh sách báo thức để thay đổi trạng thái on/off. | MainActivity | - Update báo thức trong cơ sở dữ liệu.  - Start Service SetAlarmService để cài đặt báo thức gần nhất cho điện thoại. |

Bảng : Chi tiết chức năng thay đổi trạng thái on/off báo thức

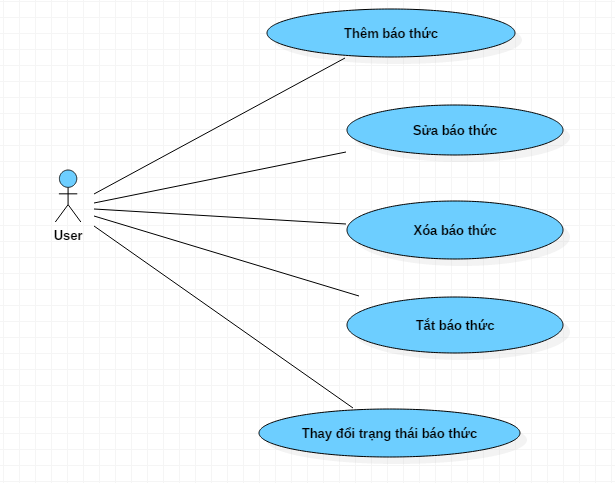
**Chi tiết chức năng nghe nhạc và đoán tên khi muốn tắt báo thức**

|  |  |  |  |
| --- | --- | --- | --- |
| Thứ tự | Thao tác người dùng | Màn hình | Mô tả hệ thống |
| 1 |  | StopAlarmActivity | - Khi đễn thời gian đã cài đặt của báo thức, hiển thị màn hình StopAlarmActivity.  - Start Service SetAlarmService để cài đặt báo thức mới gần nhất cho điện thoại.  - Đồng thời phát âm báo thức để báo hiệu cho người dùng. |
| 2 | Click Stop button | StopAlarmActivity | - Nếu báo thức là loại báo thức “Default” mặc định, ngừng phát âm báo và thoát khỏi màn hình này.  - Nếu báo thức là loại “PlayGame”, ngừng báo thức, mở màn hình AudioActiviy và phát một bản nhạc được lấy từ danh sách trong điện thoại, kèm theo hiển thị 4 phương án hiển thị tên bản nhạc cho nguời dùng lựa chọn. |
| 3 | Click 1 button chọn đáp án | AudioActivity | - Nếu đáp án chọn là đúng, sẽ được cộng 1 điểm, sai thì giữ nguyên.  - Cứ lần lượt phát từng bản nhạc và trả lời đến khi đạt đủ 3 điểm, thông báo hoàn thành, dừng phát bản nhạc và sau 5 giây sẽ thoát màn hình |

Bảng : Chi tiết chức năng tắt báo thức

## 3.2. 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.



Hình : Biểu đồ Use case của ứng dụng

### 3.2.1. Use case hiển thị danh sách báo thức

|  |  |
| --- | --- |
| **Use Case** | List Alarm |
| **Actor** | User |
| **Brief Description** | Hiển thị danh sách báo thức |
| **Main Flow** | 1. Mở ứng dụng 2. Hệ thống truy vấn CSDL và trả về kết quả danh sách báo thức 3. Nhận và hiển thị kết quả cho người dùng lên màn hình MainActivity |

Bảng : Use case hiển thị danh sách báo thức

### 3.2.2. Use case thêm mới một báo thức

|  |  |
| --- | --- |
| **Use Case** | Add Alarm |
| **Actor** | User |
| **Brief Description** | Thêm mới một báo thức |
| **Main Flow** | 1. Click add button, mở màn hình AddAlarmActivity  2. Cài đặt các thuộc tính cho báo thức  3. Click save button  4. Hệ thống lưu báo thức vào cơ sở dữ liệu, start Service cài đặt báo thức gần nhất. Trở về màn hình MainActivity |
| **Alternative Flows** | 4.1. Nếu không thành công thông báo cho người dùng biết. |
| **Pre-condition** |  |

Bảng : Use case thêm mới báo thức

### 3.2.3. Use case chỉnh sửa báo thức

|  |  |
| --- | --- |
| **Use Case** | Edit Alarm |
| **Actor** | User |
| **Brief Description** | Chỉnh sửa báo thức |
| **Main Flow** | 1. Người dùng chọn 1 báo thức đã cài đặt, mở màn hình EditAlarmActivity.  2. Người dùng chỉnh sửa thông tin báo thức.  3. Click save button  4. Hệ thống update vào co sở dữ liệu, start service cài đặt báo thức gần nhất. Trở về màn hình MainActivity |
| **Alternative Flows** | 4.1. Nếu không thành công thông báo cho người dùng biết. |
| **Pre-condition** |  |

Bảng : Use case chỉnh sửa báo thức

### 3.2.4. Use case xóa báo thức

|  |  |
| --- | --- |
| **Use Case** | Delete Alarm |
| **Actor** | User |
| **Brief Description** | Xóa báo thức |
| **Main Flow** | 1. Người dùng chọn 1 báo thức đã cài đặt, mở màn hình EditAlarmActivity.  2. Click delete button.  3. Hệ thống kết nối cơ sở dữ liệu và xóa báo thức, start service cài đặt báo thức gần nhất. Trở về màn hình MainActivity |
| **Alternative Flows** | 3.1 Nếu không thành công, thông báo cho người dùng |
| **Pre-condition** |  |

Bảng : Use case xóa báo thức

### 3.2.5. Use case tắt báo thức

|  |  |
| --- | --- |
| **Use Case** | Stop Alarm |
| **Actor** | User |
| **Brief Description** | Tắt báo thức |
| **Main Flow** | 1. Đến thời gian báo thức, hệ thống tự động mở màn hình StopAlarm và phát âm báo thức báo hiệu cho người dùng. Đồng thời start Service để cài đặt báo thức mới gần nhất.  2. Người dùng click stop button. Nếu báo thức là mặc định thì ngừng báo thức. Nếu báo thức là “Play Game”, mở màn hình AudioActivity để người dùng chơi minigame.  3. Tại màn hình AudioActivity, hệ thống sẽ phát 1 từng bản nhạc bất kì trong điện thoại và hiển thị đáp án cho người dùng lựa chọn. Khi người dùng hoàn thành trò chơi, thông báo cho người dùng và thoát khỏi mà hình. |
| **Alternative Flows** |  |
| **Pre-condition** |  |

Bảng : Use case tắt báo thức

### 3.2.6. Use case đăng nhập

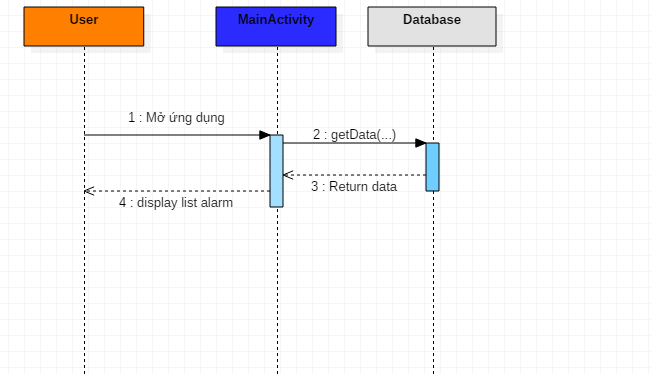
|  |  |
| --- | --- |
| **Use Case** | Change State Alarm |
| **Actor** | User |
| **Brief Description** | Thay đổi trạng thái báo thức (on/off) |
| **Main Flow** | 1. Người dùng vuốt thay đổi trạng thái on/off của từng báo thức trên danh sách báo thức. 2. Hệ thống update cơ sở dữ liệu, sau đó start Service cài đặt báo thức mới gần nhất cho ứng dụng. |
| **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 thay đổi trạng thái báo thức

## 3.3. 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.

### 3.3.1. Biểu đồ tuần tự chức năng hiển thị danh sách báo thức

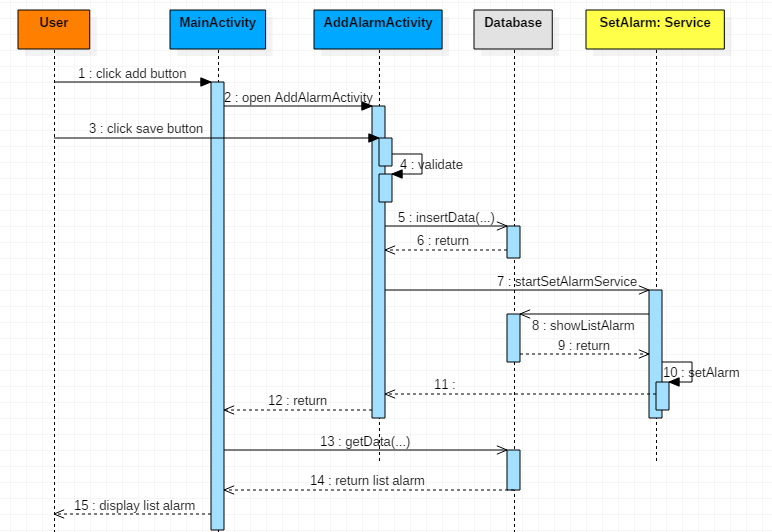


Hình : Biểu đồ tuần tự chức năng hiển thị danh sách báo thức

Các bước được thực hiện tuần tự như sau:

* 1.User vào ứng dụng
* 2-3-4: lấy tất cả các báo thức, trả kết quả, gán adapter vào Ryceclerview và hiển thị ra màn hình

### 3.3.2. Biểu đồ tuần tự chức năng thêm mới báo thức

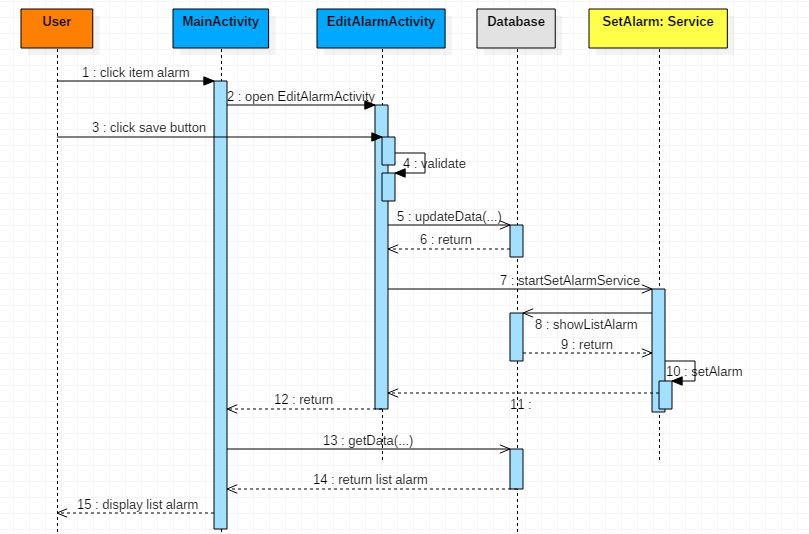


Hình : Biểu đồ tuần tự chức năng thêm mới báo thức

Các bước được thực hiện tuần tự như sau:

* 1-2-3 người dùng click add button, mở màn hình AddAlarmActivity, cài đặt thông tin báo thức và click save button.
* 4-5-6: hệ thống kiểm tra thông tin cài đặt báo thức và lưu vào cơ sở dữ liệu.
* 7-8-9-10-11: start Service thực hiện cài đặt váo thức gần nhất cho điện thoại.
* 12-13-14-15: trở về màn hình MainActivity, truy vấn cơ sở dữ liệu lấy danh sách báo thức và hiển thị ra màn hình

### 3.3.3. Biểu đồ tuần tự chức năng chỉnh sửa báo thức

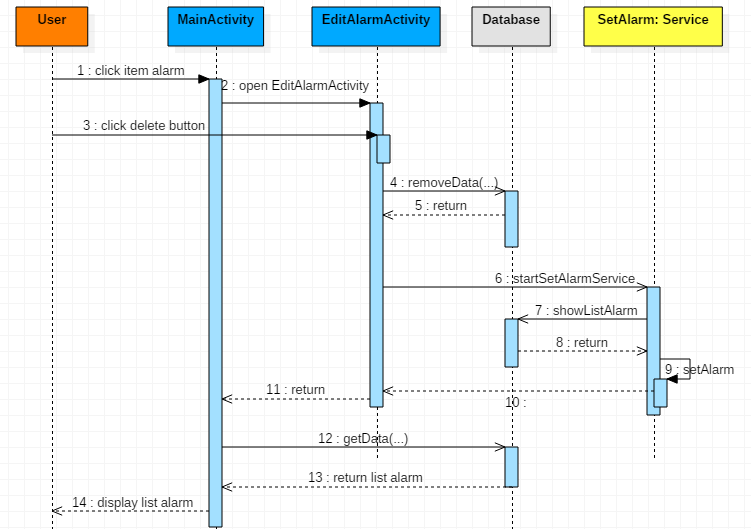


Hình : Biểu đồ tuần tự chức năng chỉnh sửa báo thức

Các bước được thực hiện tuần tự như sau:

* 1-2-3 người dùng click chọn một báo thức trong danh sách báo thức trên màn hình MainActicity, mở màn hình EditAlarmActivity, chỉnh sửa thông tin báo thức và click save button.
* 4-5-6: hệ thống kiểm tra thông tin cài đặt báo thức và cập nhật vào cơ sở dữ liệu.
* 7-8-9-10-11: start Service thực hiện cài đặt váo thức gần nhất cho điện thoại.
* 12-13-14-15: trở về màn hình MainActivity, truy vấn cơ sở dữ liệu lấy danh sách báo thức và hiển thị ra màn hình

### 3.3.4 Biểu đồ tuần tự chức năng xóa báo thức

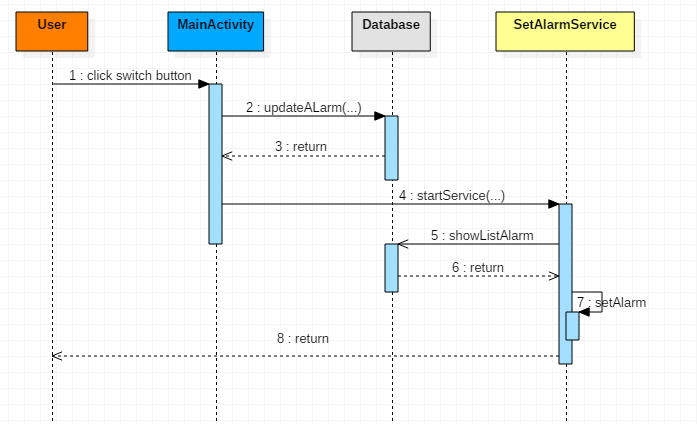


Hình : Biểu đồ tuần tự chức năng xóa báo thức

Các bước được thực hiện tuần tự như sau:

* 1-2-3 người dùng click chọn một báo thức trong danh sách báo thức trên màn hình MainActicity, mở màn hình EditAlarmActivity, click delete button.
* 4-5: Hệ thống xóa báo thức trong cơ sở dữ liệu
* 6-7-8-9-10: start Service thực hiện cài đặt váo thức gần nhất cho điện thoại.
* 11-12-13-14: trở về màn hình MainActivity, truy vấn cơ sở dữ liệu lấy danh sách báo thức và hiển thị ra màn hình

### 3.3.5. Biểu đồ tuần tự chức năng thay đổi trạng thái on/off báo báo thức

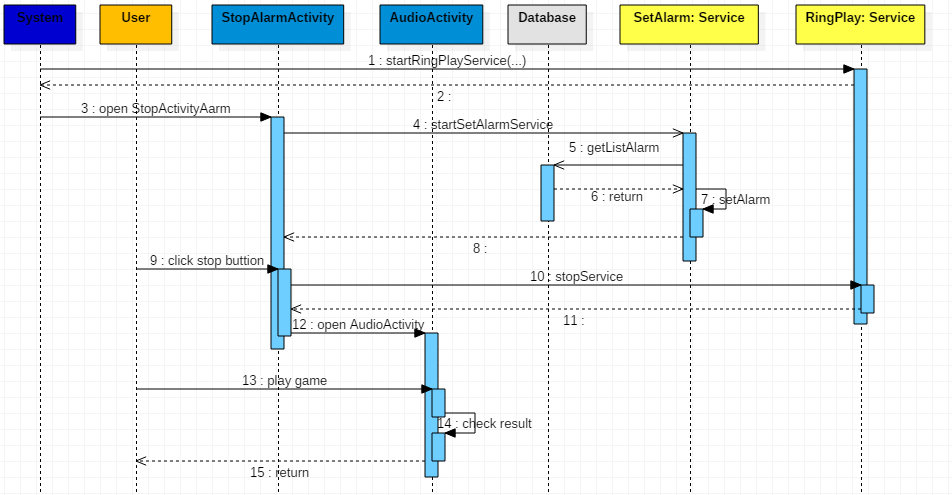


Hình : Biểu đồ tuần tự chức năng Comment

Các bước được thực hiện tuần tự như sau:

* 1. Người dùng click SwitchButton từng item báo thức để thay đổi trạng thái on/off
* 2-3: Hệ thống cập nhật vào cơ sở dữ liệu
* 4-5-6-7-8: start Service thực hiện cài đặt váo thức gần nhất cho điện thoại, báo kết quả cho người dùng

### 3.3.6. Biểu đồ tuần tự chức năng nghe nhạc tắt báo thức

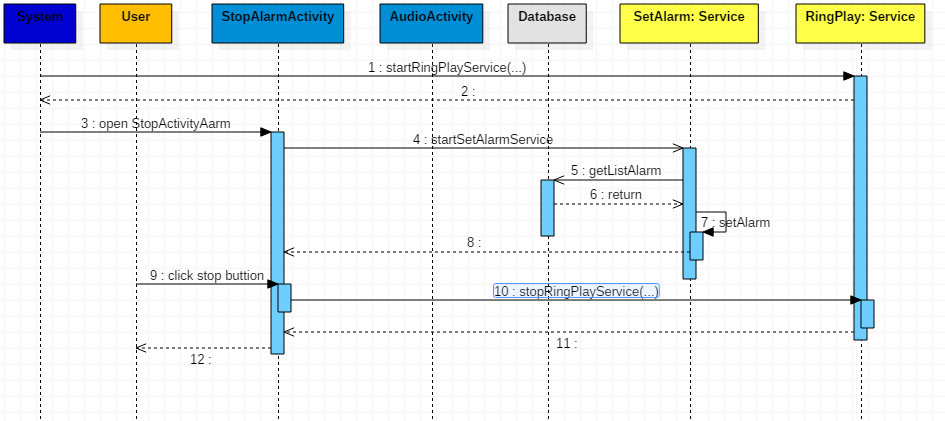


Hình : Biểu đồ tuần tự chức năng nghe nhạc tắt báo thức

Các bước được thực hiện tuần tự như sau:

* 1-2-3: Tới thời gian báo thức, hệ thống tự động gọi service phát âm báo thức, đồng thời mở màn hình StopAlarmActivity.
* 4-5-6-7-8: Tại màn hình này, tự động start Service cài đặt báo thức mới gần nhất cho điện thoại.
* 9-10-11-12: Người dùng click stop button. Hệ thống stop Service tắt âm báo thức, đồng thời mở màn hình AudioActivity.
* 13-14-15: Tại màn hình này, hệ thống phát lần lượt từng bản nhạc ngẫu nhiên lấy từ điện thoại và hiển thị các đáp án cho người dùng lựa chọn. Người dùng lựa chọn đáp án, hệ thống kiểm tra kết quả và thông báo cho người dùng.

### 3.3.7. Biểu đồ tuần tự chức năng tắt báo thức mặc định



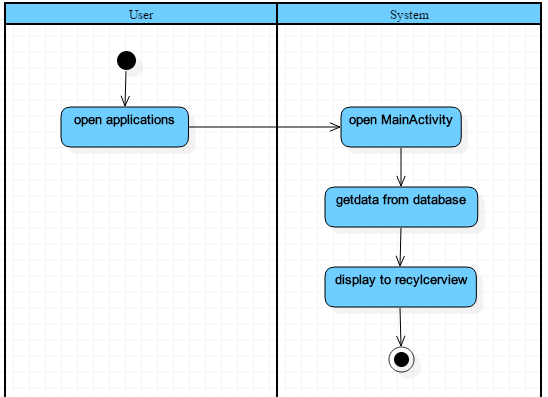
Hình : Biểu đồ tuần tự chức năng tắt báo thức mặc định

Các bước được thực hiện tuần tự như sau:

* 1-2-3: Tới thời gian báo thức, hệ thống tự động gọi service phát âm báo thức, đồng thời mở màn hình StopAlarmActivity.
* 4-5-6-7-8: Tại màn hình này, tự động start Service cài đặt báo thức mới gần nhất cho điện thoại.
* 9-10-11-12: Người dùng click stop button. Hệ thống stop Service tắt âm báo thức, rời khỏi màn hình.

## 3.4 Biểu đồ hoạt động

### 3.4.1 .Biểu đồ hoạt động cho chức năng xem danh sách báo thức

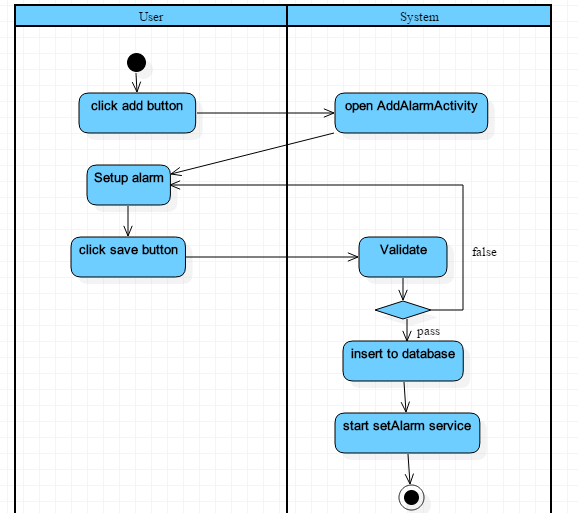


Hình : Biểu đồ hoạt động chức năng xem danh sách báo thức

Mô tả:

* Người dùng mở ứng dụng.
* Hệ thống truy vấn CSDL và trả về danh sách báo thức
* Hiển thị danh sách truyện cho người dùng.

### 3.4.2 Biểu đồ hoạt động cho chức năng thêm mới báo thức

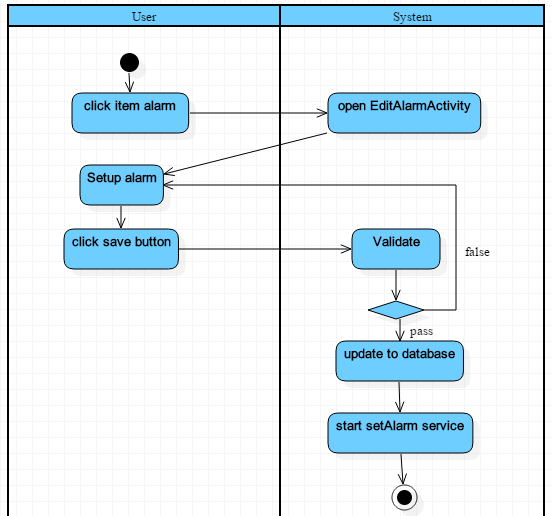


Hình : Biểu đồ hoạt động chức năng them mới báo thức

Mô tả:

* Người dùng click add button, mở màn hình AddAlarmActivity, cài đặt báo thức và click save button.
* Hệ thống insert data vào cơ sở dữ liệu, sau đó chạy service SetAlarmService để cài đặt báo thức gần nhất cho điện thoại.
* Trở về màn hình MainActivity

### 3.4.3 Biểu đồ hoạt động cho chức năng sửa báo thức

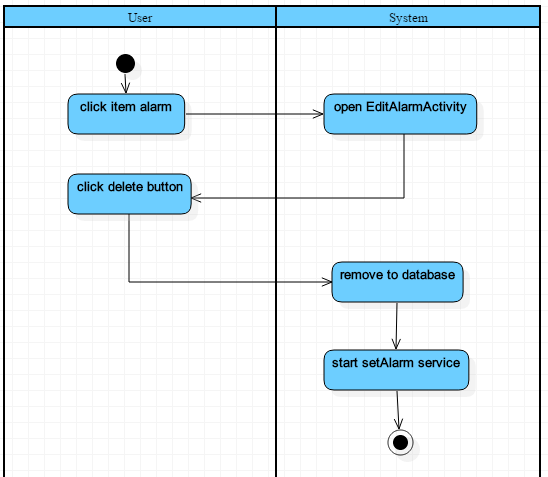


Hình : Biểu đồ hoạt động chức năng xem sửa báo thức

Mô tả:

* Người dùng chọn một bao thức, mở màn hình EditAlarmActivity, chỉnh sủa lại cài đặt báo thức và click save button.
* Hệ thống update vào cơ sở dữ liệu, sau đó chạy service SetAlarmService cài đặt báo thức gần nhất cho điện thoại.
* Trở về màn hình MainActivity

### 3.4.4 Biểu đồ hoạt động cho chức năng xóa báo thức

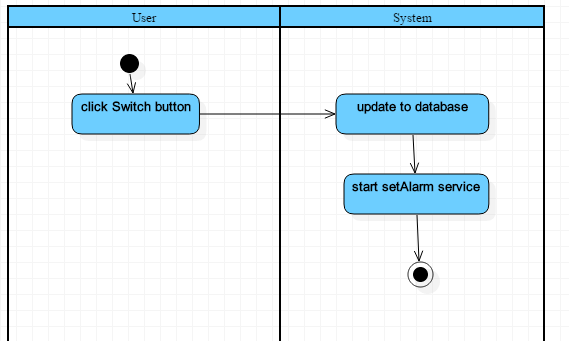


Hình : Biểu đồ hoạt động chức năng xóa báo thức

Mô tả:

* Người dùng chọn một báo thức, mở màn hình EditAlarmActivity và click delete button.
* Hệ thống xóa báo thức trong cơ sở dữ liệu, sau đó chạy service SetAlarmService cài đặt báo thức gần nhất cho điện thoại.
* Trở về màn hình MainActivity

### 3.4.5. Biểu đồ hoạt động cho chức năng thay đổi trạng thái báo thức

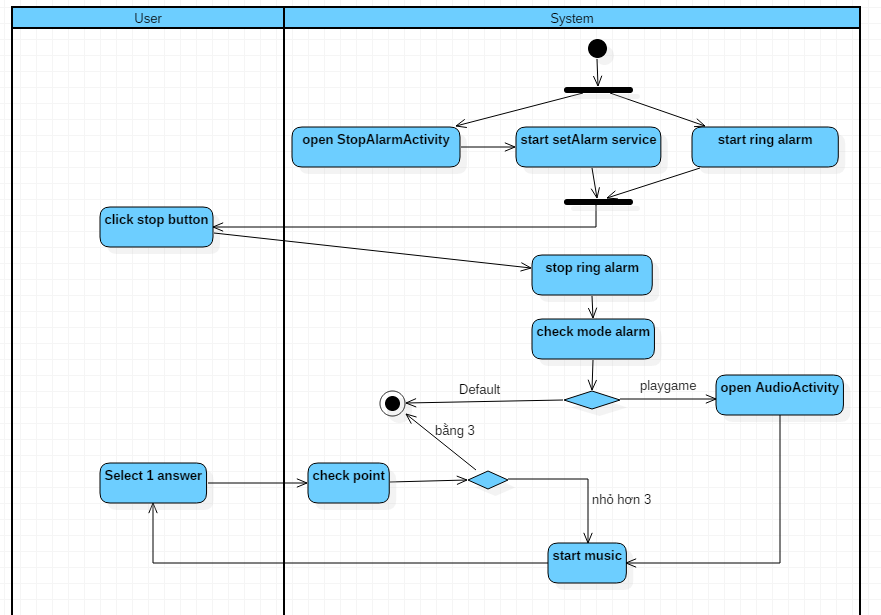


Hình : Biểu đồ hoạt động chức năng thay đổi trạng thái báo thức

Mô tả:

* Người dùng click Switch button ứng với mỗi báo thức để thay đổi trjang thái on/off.
* Ứng dụng update vào cơ sở dữ liệu, sau đó chạy service SetAlarmService để cài đặt báo thức gần nhất cho điện thoại

### 3.4.6 Biểu đồ hoạt động cho chức năng tắt báo thức



Hình : Biểu đồ hoạt động chức năng tắt báo thức

Mô tả:

* Khi đến thời gian báo thức, ứng dụng mở âm báo thức, đồng thời mở màn hình StopAlarmActivity, sau đó chạy service SetAlarmService để cài đặt báo thức mới gần nhất cho điện thoại.
* Người dùng click stop button.
* Nếu báo thức là loại “Default”, hệ thống sẽ tắt âm báo và finish. Nếu báo thức là loại “PlayGame”, tắt âm báo thức, mở tiếp màn hình AudioActivity để người dùng chơi mini game.
* Ứng dụng lần lượt phát từng bản nhạc trong danh sách nhạc của điển thoại, người dùng chọn đáp án tên bản nhạc. Nếu đúng được cộng 1 điểm, sai lại tiếp tục nghe bản nhạc khác. Đến khi nào đạt đủ 3 điểm, nhạc sẽ tắt, và việc tắt báo thức hoàn thành.

## 3.5 Thiết kế CSDL

Cơ sở dữ liệu gồm có 2 bảng:

* Bảng Alarm: lưu trữ danh sách báo thức

|  |  |  |  |
| --- | --- | --- | --- |
| Name | DataType | Constraints | Documentation |
| id\_alarm | Integer | PK | khóa chính |
| name\_alarm | varchar(255) |  | Tiêu đề báo thức |
| ring\_alarm | varchar(255) |  | Âm báo thức |
| hour | Integer | NOT NULL | Giờ báo thức |
| minute | Integer | NOT NULL | Phút báo thức |
| arr\_day | varchar(255) | NOT NULL | Danh sách ngày báo thức |
| state | varchar(255) |  | Trạng thái báo thức(on/off) |
| mode | varchar(255) |  | Kiểu báo thức |
| type | varchar(255) |  | Loại âm báo (mặc định / từ điện thoại) |
| vibrate | varchar(255) |  | Báo rung |

Bảng : Bảng Alarm trong cơ sở dữ liệu

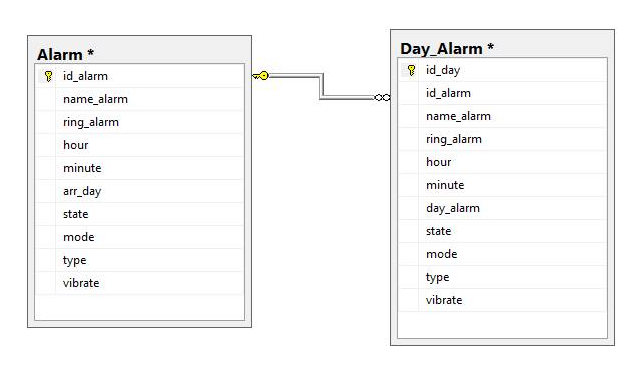
* Bảng Day\_Alarm: ứng với mỗi ngày của 1 báo thức trong bảng Alarm sẽ có 1 bảng Day\_Alarm, bảng này cũng lưu đầy đủ thông tin của báo thức.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | DataType | Constraints | Documentation |
| id\_day | Integer | PK | Khóa chính |
| id\_alarm | Integer | FOREIGN KEY | Khóa ngoại |
| name\_alarm | varchar(255) | NOT NULL | Tiêu đề báo thức |
| ring\_alarm | varchar(255) | NOT NULL | Âm báo thức |
| hour | Integer | NOT NULL | Giờ báo thức |
| minute | Integer | NOT NULL | Phút báo thức |
| day\_alarm | varchar(255) | NOT NULL | Ngày báo thức |
| state | varchar(255) |  | Trạng thái báo thức(on/off) |
| mode | varchar(255) |  | Kiểu báo thức |
| type | varchar(255) |  | Loại âm báo (mặc định / từ điện thoại) |
| vibrate | varchar(255) |  | Báo rung |

Bảng : Bảng Day\_Alarm trong cơ sở dữ liệu

Ứng với mỗi bản ghi của bảng Alarm có thể có nhiều bản ghi trong bảng Day\_Alarm (tối đa là 7 bản ghi tương ứng 7 ngày trong tuần). Mỗi một ngày trong dánh sách các ngày của một bản ghi ở bảng Alarm sẽ có một bản ghi tương ứng trong bảng Day\_Alarm.

* Biểu đồ thiết kế CSDL



Hình : Biểu đồ thiết kế CSDL

# CHƯƠNG IV: TRIỂN KHAI ỨNG DỤNG

## 4.1 Môi trường triển khai

* Ứng được cài đặt trên Android 5.0
* Môi trường phát triển là trên máy tính cá nhân (Desktop hoặc Laptop) sử dụng hệ điều hành Windows có cài đặt máy ảo Genymotion, công cụ Android Studio.
* Công cụ phát triển là Android Studio đã cài Android SDK.

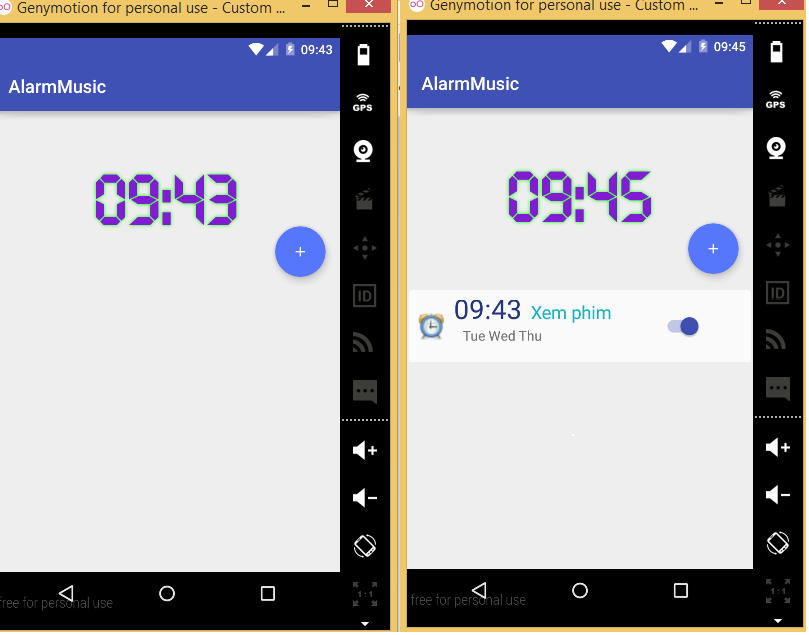
## 4.2. Cách thức triển khai và sử dụng hệ thống

- Chạy ứng dụng Android bằng máy ảo Genymotion hoặc máy điện thoại thật .

## 4.3. Kịch bản deme

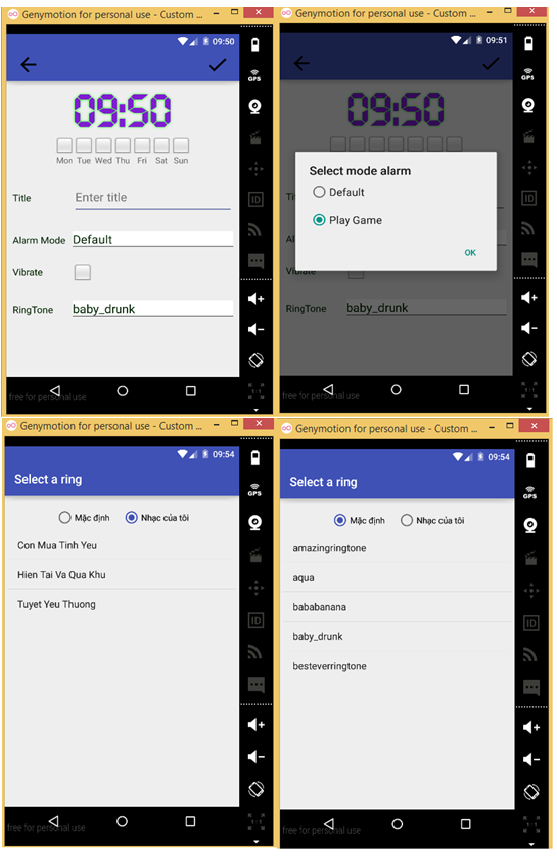
**Kịch bản người dung**

Người dụng mở ứng dụng xem danh sách báo thức



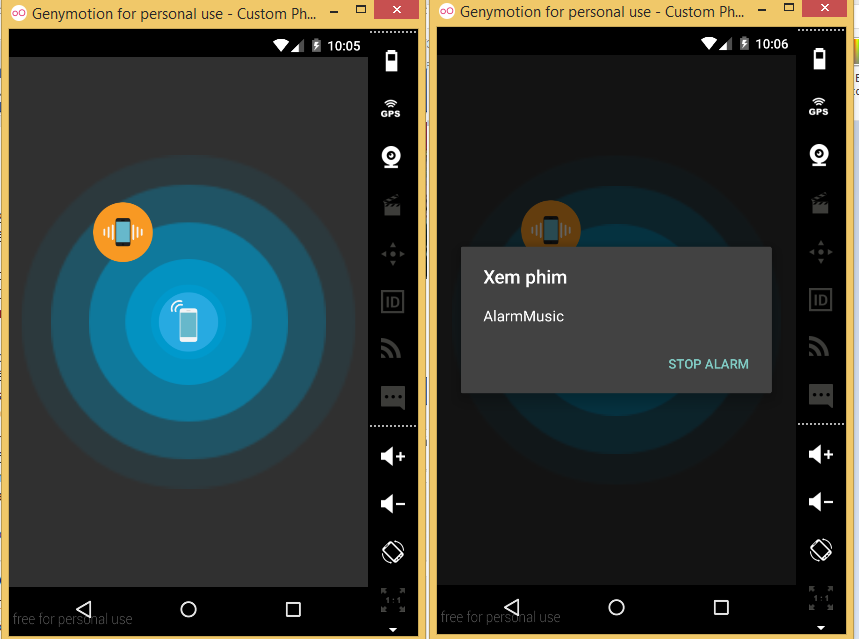
Hình : Demo chức năng hiển thị danh sách báo thức

Người dùng thêm mới hoặc chỉnh sửa báo thức



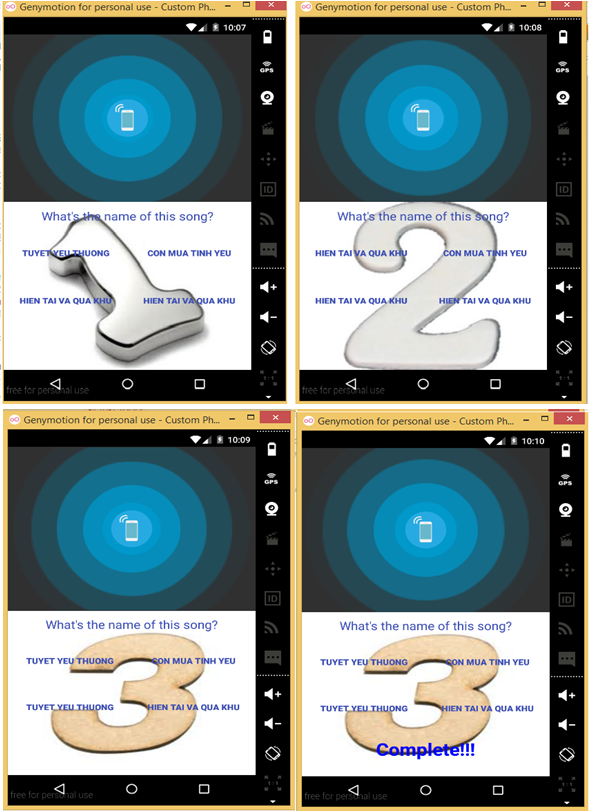
Hình 22: Demo chức năng thêm mới hoặc chỉnh sửa báo thức

Phát báo thức:



Hình : Demo chức năng phát âm báo thức

Người dùng nghe nhạc và đoán bài tắt báo thức



Hình : Demo chức năng nghe nhạc và tắt báo thứ

# 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 cách hoạt động của các thành phần cơ bản trong android, đặc biệt là Broadcast Receivers và Service để cài đặt báo thức.
* Xây dựng cơ bản hoàn chỉnh ứng dụng, đáp ứng đầy đủ các chức năng đã đề ra ban đầu.
* Giao diện đơn giản, dễ sử dụng. Phần báo thức khá thú vị.
* Chạy tốt với các bản từ android 4.0 và mới hơn.

### 5.1.2 Nhược điểm

* Giao diện còn đơn giản, chưa được sắc sảo.
* Chức năng còn đơn giản.

## 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:

* Thêm một số cách tắt báo thức độc đáo, mới lạ hơn
* Giao diện sinh động, tinh tế hơn để tạo ấn tượng với người dùng.

**TÀI LIỆU THAM KHẢO**

1. google.com, github.com.
2. <https://duythanhcse.wordprs.com>
3. https://khoapham.vn

**DANH MỤC CÁC HÌNH ẢNH**

[Hình 1: Mô hình kiến trúc nền tảng Android 9](#_Toc451291953)

[Hình 2: Biểu đồ so sánh tốc độ của máy ảo Java giữa các phiên bản Android (Càng lớn càng tốt) 12](#_Toc451291954)

[Hình 3: So sánh giao diện nguyên bản của Android và giao diện Sense của riêng hãng HTC thiết kế 13](#_Toc451291955)

[Hình 4: Thống kê số lượng điện thoại sử dụng các phiên bản Android 13](#_Toc451291956)

[Hình 5: Cấu trúc thư mục và file của một dự án phần mềm Android trên Android Studio 16](#_Toc451291957)

[Hình 6: Biểu đồ Use case của ứng dụng 23](#_Toc451291958)

[Hình 7: Biểu đồ tuần tự chức năng hiển thị danh sách báo thức 27](#_Toc451291959)

[Hình 8: Biểu đồ tuần tự chức năng thêm mới báo thức 28](#_Toc451291960)

[Hình 9: Biểu đồ tuần tự chức năng chỉnh sửa báo thức 29](#_Toc451291961)

[Hình 10: Biểu đồ tuần tự chức năng xóa báo thức 30](#_Toc451291962)

[Hình 11: Biểu đồ tuần tự chức năng Comment 31](#_Toc451291963)

[Hình 12: Biểu đồ tuần tự chức năng nghe nhạc tắt báo thức 32](#_Toc451291964)

[Hình 13: Biểu đồ tuần tự chức năng tắt báo thức mặc định 33](#_Toc451291965)

[Hình 14: Biểu đồ hoạt động chức năng xem danh sách báo thức 34](#_Toc451291966)

[Hình 15: Biểu đồ hoạt động chức năng them mới báo thức 35](#_Toc451291967)

[Hình 16: Biểu đồ hoạt động chức năng xem sửa báo thức 36](#_Toc451291968)

[Hình 17: Biểu đồ hoạt động chức năng xóa báo thức 37](#_Toc451291969)

[Hình 18: Biểu đồ hoạt động chức năng thay đổi trạng thái báo thức 38](#_Toc451291970)

[Hình 19: Biểu đồ hoạt động chức năng tắt báo thức 39](#_Toc451291971)

[Hình 20: Biểu đồ thiết kế CSDL 42](#_Toc451291972)

[Hình 21: Demo chức năng hiển thị danh sách báo thức 43](#_Toc451291973)

[Hình 22: Demo chức năng thêm mới hoặc chỉnh sửa báo thức 44](#_Toc451291974)

[Hình 23: Demo chức năng phát âm báo thức 45](#_Toc451291975)

[Hình 24: Demo chức năng nghe nhạc và tắt báo thức 46](#_Toc451291976)

**DANH MỤC CÁC BẢNG**

[Bảng 1: Chi tiết chức năng xem danh sách báo thức 18](#_Toc451292788)

[Bảng 2: Chi tiết chức năng thêm mới báo thức 20](#_Toc451292789)

[Bảng 3: Chi tiết chức năng sửa báo thức 21](#_Toc451292790)

[Bảng 4: Chi tiết chức năng xóa báo thức 21](#_Toc451292791)

[Bảng 5: Chi tiết chức năng thay đổi trạng thái on/off báo thức 21](#_Toc451292792)

[Bảng 6: Chi tiết chức năng tắt báo thức 22](#_Toc451292793)

[Bảng 7: Use case hiển thị danh sách báo thức 23](#_Toc451292794)

[Bảng 8: Use case thêm mới báo thức 24](#_Toc451292795)

[Bảng 9: Use case chỉnh sửa báo thức 24](#_Toc451292796)

[Bảng 10: Use case xóa báo thức 25](#_Toc451292797)

[Bảng 11: Use case tắt báo thức 26](#_Toc451292798)

[Bảng 12: Use case thay đổi trạng thái báo thức 26](#_Toc451292799)

[Bảng 13: Bảng Alarm trong cơ sở dữ liệu 40](#_Toc451292800)

[Bảng 14: Bảng Day\_Alarm trong cơ sở dữ liệu 41](#_Toc451292801)