

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG**



ISO 9001:2015

ĐỒ ÁN TỐT NGHIỆP

NGÀNH: CÔNG NGHỆ THÔNG TIN

Sinh viên : Đàm Duy Quang
Giảng viên hướng dẫn: ThS. Phùng Anh Tuấn

HẢI PHÒNG - 2019

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

XÂY DỰNG ỨNG DỤNG ANDROID XỬ LÝ TIN NHẮN
BÌNH CHỌN

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY
NGÀNH: CÔNG NGHỆ THÔNG TIN

Sinh viên : Đàm Duy Quang
Giảng viên hướng dẫn: ThS. Phùng Anh Tuấn

HẢI PHÒNG - 2019

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

NHIỆM VỤ ĐỀ TÀI TỐT NGHIỆP

Sinh viên: Đàm Duy Quang

Mã SV: 1412101020

Lớp: CT1802

Ngành: Công nghệ thông tin

Tên đề tài: Xây dựng ứng dụng Android xử lý tin nhắn bình chọn

MỤC LỤC

CHƯƠNG 1 : TỔNG QUAN HỆ ĐIỀU HÀNH ANDROID	3
1.1. Giới thiệu chung về hệ điều hành Android	3
1.2. Kiến trúc của hệ điều hành Android	4
1.2.1. Nhân Linux	5
1.2.2. Tầng thư viện và thực thi	5
1.2.3. Tầng khung ứng dụng	6
1.2.4. Tầng ứng dụng	7
1.3. Giao diện hệ điều hành Android	7
1.4. Quá trình phát triển phiên bản Android	8
1.4.1. Phiên bản Android 1.0	8
1.4.2. Phiên bản Android 1.5 : Cup Cake	9
1.4.3. Phiên bản Android 2.0 và Android 2.1 : Eclair.....	9
1.4.4. Phiên bản Android 2.2 : Froyo	9
1.4.5. Phiên bản Android 2.3 : Gingerbread	9
1.4.6. Phiên bản Android 3.x : Honeycomb	10
1.4.7. Phiên bản Android 4.0 : Ice Cream Sandwich	10
1.4.8. Phiên bản Android 4.1 : Jelly Bean	10
1.4.9. Phiên bản Android 4.2 : Jelly Bean	10
1.4.10. Phiên bản Android 4.3 : Jelly Bean	11
CHƯƠNG 2 : GIỚI THIỆU VỀ MÔI TRƯỜNG ANDROID STUDIO	12
2.1. Giới thiệu Android studio	12
2.2. Cài đặt môi trường lập trình Android	12
2.2.1. Cài đặt JAVA JDK	12
2.2.2. Cài đặt Android Studio	14
2.2.3. Cài đặt máy ảo Android Genymotion	15
2.3. Sơ lược về sử dụng Android Studio	19
2.3.1. Tạo mới một Project	19
2.3.2. Cấu trúc của một dự án trong Android Studio	22
2.4. Quản lý màn hình điện thoại	25
2.5. Chạy ứng dụng	25
CHƯƠNG 3 : KỸ THUẬT HỖ TRỢ XÂY DỰNG ỨNG DỤNG	29
3.1. Hosting	29
3.1.1. Khái niệm	29
3.1.2. Yêu cầu và tính năng cần thiết của Hosting	29
3.2. Website	29
3.2.1. Khái niệm	29
3.2.2. Vai trò	29
3.3. Cơ sở dữ liệu web	30
3.3.1. Khái niệm	30
3.3.2. Xây dựng cơ sở dữ liệu web	30

3.4. Dịch vụ web	33
3.4.1. Khái niệm	33
3.4.2. Chức năng	34
3.4.3. Kỹ thuật xây dựng dịch vụ web cho thiết bị di động	34
3.5. Bộ nghe sự kiện tin nhắn	36
3.5.1. Khái niệm	36
3.5.2. Chức năng	36
3.5.3. Kỹ thuật đăng ký bộ nghe sự kiện tin nhắn	37
3.6. Kỹ thuật xử lý tin nhắn	37
CHƯƠNG 4 : CHƯƠNG TRÌNH THỬ NGHIỆM	38
4.1. Phát biểu bài toán	38
4.2. Mô hình chương trình	38
4.3. Thiết kế dữ liệu	39
4.4. Một số giao diện chương trình	39
4.4.1. Giao diện chính của chương trình	39
4.4.2. Giao diện trả lời tin nhắn	40
4.4.3. Giao diện thông báo thêm dữ liệu thành công	40
4.4.4. Giao diện thông báo thêm dữ liệu thất bại	41
KẾT LUẬN	42
TÀI LIỆU THAM KHẢO	43

CHƯƠNG 1 : TỔNG QUAN HỆ ĐIỀU HÀNH ANDROID

1.1. Giới thiệu chung về hệ điều hành android

Android là một hệ điều hành di động dựa trên nền tảng linux phiên bản 2.6 dành cho các dòng điện thoại SmartPhone [1]. Đầu tiên được ra đời bởi công ty liên hợp Android, sau đó được Google mua lại và phát triển từ năm 2005 và trở thành một hệ điều hành di động mã nguồn mở, miễn phí, mạnh mẽ và được ưa chuộng cao trên thế giới. Hệ điều hành android một hệ điều hành rất mạnh mẽ, có bảo mật cao, hỗ trợ được nhiều công nghệ tiên tiến như 3G, GPS, EDGE, Wifi.. tương thích với nhiều phần cứng, hỗ trợ nhiều loại bộ nhập dữ liệu như keyboard, touch và trackball. Android là hệ điều hành di động nên có khả năng kết nối cao với các mạng không dây. Hỗ trợ công nghệ OpenGL nên có khả năng chơi các phương tiện media, hoạt hình cũng như trình diễn các khả năng đồ họa khác cực tốt, là tiền đề để phát triển các ứng dụng có giao diện phức tạp chẳng hạn như là các trò chơi.

Android liên tục được phát triển, mỗi bản cập nhật từ google là mỗi lần android được tối ưu hóa để hoạt động tốt hơn, nhanh và ổn định hơn, hỗ trợ thêm công nghệ mới. Chẳng hạn như theo một đánh giá thì android phiên bản 2.2 hoạt động nhanh hơn bản 2.1 tới 450%. Hiện nay, phiên bản mới nhất 2.3 phát hành ngày 6/12/2010 và đang tiếp tục được cập nhật. Năm 2008, hệ điều hành android đã chính thức mở toàn bộ mã nguồn, điều đó cho phép các hãng điện thoại có thể đem mã nguồn về tùy chỉnh, thiết kế lại sao cho phù hợp với mỗi mẫu mã điện thoại của họ và điều quan trọng nữa là hệ điều hành mở này hoàn toàn miễn phí, không phải trả tiền nên giúp họ tiết kiệm khá lớn chi phí phát triển hệ điều hành. Những điều đó là cực kỳ tốt không chỉ đối với các hãng sản xuất điện thoại nhỏ mà ngay cả với những hãng lớn như Samsung, HTC.... Với Google, vì android hoàn toàn miễn phí, Google không thu tiền từ những hãng sản xuất điện thoại, tuy không trực tiếp hưởng lợi từ android nhưng bù lại, những dịch vụ của hãng như Google Search, Google Maps,... nhờ có android mà có thể dễ dàng xâm nhập nhanh vào thị trường di động vì mỗi chiếc điện thoại được sản xuất ra đều được tích hợp hàng loạt dịch vụ của Google. Từ đó hãng có thể kiếm bội, chủ yếu là từ các nguồn quảng cáo trên các dịch vụ đó. Với các nhà phát triển ứng dụng (developers), việc hệ điều hành android được sử dụng phổ biến đồng nghĩa với việc họ có thể thoải mái phát triển ứng dụng trên nền android với sự tin tưởng là ứng dụng đó sẽ có thể chạy được ngay trên nhiều dòng điện thoại của các hãng khác nhau. Họ ít phải quan tâm là đang phát triển cho điện thoại nào, phiên bản bao nhiêu vì nền tảng android là chung cho nhiều dòng máy, máy ảo Java đã chịu trách nhiệm thực thi những ứng dụng phù hợp với mỗi dòng điện thoại mà nó đang chạy. Tất cả các chương trình ứng dụng được viết bằng ngôn ngữ Java kết hợp với XML nên có khả năng khả chuyên cao.

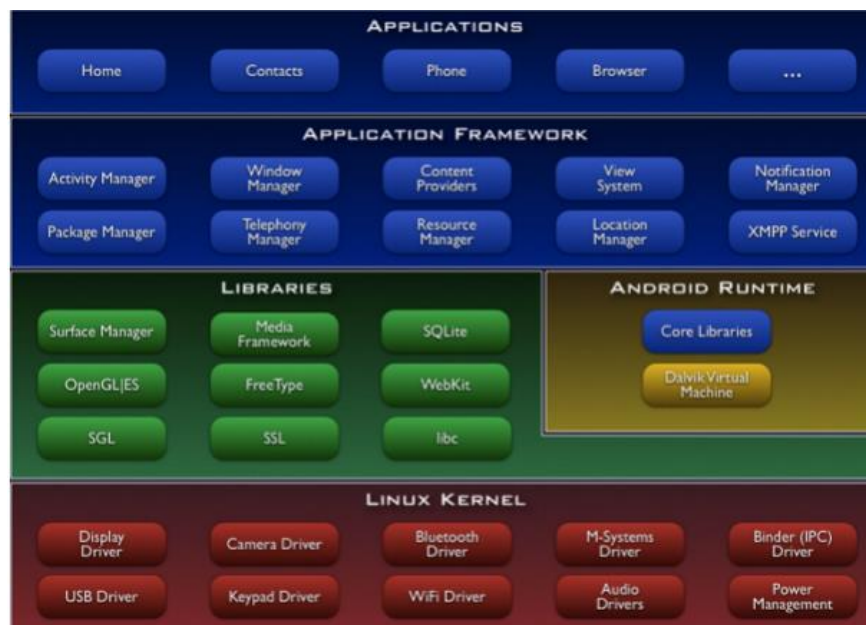
Một số hãng sản xuất điện thoại có sử dụng hệ điều hành android tiêu biểu :

- HTC với các dòng Desire HD, Evo 4G, DROID ERIS, Desire Z, Hero, Desire, Tattoo, Wildfire, Droid Incredible, Legend, Magic, Google Nexus One, Dream, Aria, Paradise
- LG với các dòng GT540 Optimus, Optimus Chic E720, Optimus One P500, GW620, Optimus Z, Optimus Q, KH5200 Andro-1, GW880, C710Aloha

- MOTOROLA với các dòng MILESTONE 2, BACKFLIP, Droid XTreme, MT710 ZHILING, MILESTONE, XT720 MOTOROI, A1680, XT800 ZHISHANG, DEFY, CHARM, XT806...
- SAMSUNG với các dòng máy I9000 Galaxy S, Galaxy Tab, Epic 4G, i5510, I5500 Galaxy 5, I7500 Galaxy, I5800 Galaxy 3, M110S Galaxy S, I6500U Galaxy, Galaxy Q, I5700 Galaxy Spica, I8520 Galaxy Beam, I909 Galaxy S
- SONY : XPERIA X10, XPERIA X10 mini, XPERIA X8
- ACER với các dòng máy beTouch T500, Liquid E, Stream, Liquid, beTouch E110, beTouch E130, beTouch E400, beTouch E120, Liquid Metal
- Ngoài ra còn nhiều hãng điện thoại vừa và nhỏ khác nữa cũng sử dụng hệ điều hành android trong sản phẩm của mình....

1.2. Kiến trúc của hệ điều hành Android

Hệ điều hành android có 4 tầng từ dưới lên trên là tầng hạt nhân Linux (Phiên bản 2.6), tầng Tầng Libraries & Android runtime , Tầng Application Framework và trên cùng là tầng Application.



Hình 1.2.1. Kiến trúc hệ điều hành Android

1.2.1. Nhân Linux

Hệ điều hành android được phát triển dựa trên hạt nhân linux, cụ thể là hạt nhân linux phiên bản 2.6, điều đó được thể hiện ở lớp dưới cùng này. Tất cả mọi hoạt động của điện thoại muốn thi hành được thì đều được thực hiện ở mức cấp thấp ở lớp này bao gồm quản lý bộ nhớ (memory management), giao tiếp với phần cứng (driver model), thực hiện bảo mật (security), quản lý tiến trình (process). Tuy được phát triển dựa vào nhân linux nhưng thực ra nhân linux đã được nâng cấp và sửa đổi rất nhiều để phù hợp với tính chất của những thiết bị cầm tay như hạn chế về bộ vi xử lý, dung lượng bộ nhớ, kích thước màn hình, nhu cầu kết nối mạng không dây...

Tầng này có các thành phần chủ yếu :

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

1.2.2. Tầng thư viện và thực thi

Phần này có 2 thành phần là phần thư viện và thực thi.

- Phần thư viện(Libraries). Phần này có nhiều thư viện được viết bằng C/C++ để các phần mềm có thể sử dụng, các thư viện đó được tập hợp thành một số nhóm như :

Thư viện hệ thống (System C library) : thư viện dựa trên chuẩn C, được sử dụng chỉ bởi hệ điều hành.

Thư viện Media (Media Libraries) : Có nhiều codec để hỗ trợ việc phát và ghi các loại định dạng âm thanh, hình ảnh, video thông dụng.

Thư viện web (LibWebCore) : Đây là thành phần để xem nội dung trên web, được sử dụng để xây dựng phần mềm duyệt web (Android Browse) cũng như để các ứng dụng khác có thể nhúng vào. Nó cực kỳ mạnh, hỗ trợ được nhiều công nghệ mạnh mẽ như HTML5, JavaScript, CSS, DOM, AJAX..

Thư viện SQLite : Hệ cơ sở dữ liệu để các ứng dụng có thể sử dụng.

- Phần thực thi

Phần này chứa các thư viện mà một chương trình viết bằng ngôn ngữ Java có thể hoạt động. Phần này có 2 bộ phận tương tự như mô hình chạy Java trên máy tính thường. Thứ nhất là các thư viện lõi (Core Library) , chứa các lớp như JAVA IO, Collections, File Access. Thứ hai là một máy ảo java (Dalvik Virtual Machine) . Mặc dù cũng được viết từ ngôn ngữ Java nhưng một ứng dụng Java của hệ điều hành android không được chạy bằng JRE của Sun (nay là Oracle) (JVM) mà là chạy bằng máy ảo Dalvik do Google phát triển.

1.2.3. Tầng khung ứng dụng

Tầng này xây dựng bộ công cụ - các phần tử ở mức cao để các lập trình viên có thể nhanh chóng xây dựng ứng dụng. Nó được viết bằng Java, có khả năng sử dụng chung để tiết kiệm tài nguyên. Đây là một nền tảng mở, điều đó có 2 điều lợi:

- Với các hãng sản xuất điện thoại : Có thể tùy biến để phù hợp với cấu hình điện thoại mà họ sản xuất cũng như để có nhiều mẫu mã, style hợp thị hiếu người dùng. Vì thế

nên tuy cùng chung nền tảng android mà điện thoại của Google có thể khác hẳn với Motorola, HTC, T-Mobile, Samsung...

- Với lập trình viên : Cho phép lập trình viên có thể sử dụng các API ở tầng trên mà không cần phải hiểu rõ cấu trúc bên dưới, tạo điều kiện cho lập trình viên tự do sáng tạo bởi vì chỉ cần quan tâm đến nội dung mà ứng dụng họ làm việc. Một tập hợp API rất hữu ích được xây dựng sẵn như hệ thống định vị, các dịch vụ chạy nền, liên lạc giữa các ứng dụng, các thành phần giao diện cấp cao ...

Giới thiệu một số thành phần của phần này :

- Activity Manager : Quản lý các chu kỳ sống của một ứng dụng cũng như cung cấp công cụ điều khiển các Activity.
- Telephony Manager : Cung cấp công cụ để thực hiện việc liên lạc như gọi điện thoại.
- XMPP Service : Cung cấp công cụ để liên lạc trong thời gian thực.
- Location Manager : Cho phép xác định vị trí của điện thoại dựa vào hệ thống định vị toàn cầu GPS và Google Maps.
- Window Manager : Quản lý việc xây dựng và hiển thị các giao diện người dùng cũng như tổ chức quản lý các giao diện giữa các ứng dụng.
- Notification Manager : Quản lý việc hiển thị các thông báo (như báo có tin nhắn, có e-mail mới..)
- Resource Manager : Quản lý tài nguyên tĩnh của các ứng dụng bao gồm các file hình ảnh, âm thanh, layout, string. (Những thành phần không được viết bởi ngôn ngữ lập trình)

1.2.4. Tầng ứng dụng

Đây là lớp ứng dụng giao tiếp với người dùng, bao gồm các ứng dụng như : Các ứng dụng cơ bản, được cài đặt đi liền với hệ điều hành là gọi điện(phone), quản lý danh bạ(Contacts), duyệt web (Browser), nhắn tin (SMS), lịch làm việc (Calendar), đọc e-mail (Email-Client), bản đồ (Map), quay phim chụp ảnh (camera)... Các ứng dụng được cài thêm như các phần mềm chứng khoán (Stock), các trò chơi (Game), từ điển... Các chương trình có các đặc điểm là :

- Viết bằng Java, phần mở rộng là apk
- Khi mỗi ứng dụng được chạy, nó có một phiên bản Virtual Machine được dựng lên để phục vụ cho nó. Nó có thể là một Active Program : Chương trình có giao diện với người sử dụng hoặc là một background : chương trình chạy nền hay là dịch vụ.
- Android là hệ điều hành đa nhiệm, điều đó có nghĩa là trong cùng một thời điểm, có thể có nhiều chương trình cùng chạy một lúc, tuy nhiên, với mỗi ứng dụng thì có duy nhất một thực thể (instance) được phép chạy mà thôi. Điều đó có tác dụng hạn chế sự lạm dụng tài nguyên, giúp hệ thống hoạt động tốt hơn.
- Các ứng dụng được gán số ID của người sử dụng nhằm phân định quyền hạn khi sử dụng tài nguyên, cấu hình phần cứng và hệ thống.
- Android là một hệ điều hành có tính mở, khác với nhiều hệ điều hành di động khác, android cho phép một ứng dụng của bên thứ ba được phép chạy nền. Các ứng dụng đó chỉ có một hạn chế nhỏ đó là nó không được phép sử dụng quá 5~10% công suất CPU, điều đó nhằm để tránh độc quyền trong việc sử dụng CPU.
- Ứng dụng không có điểm vào cố định, không có phương thức main để bắt đầu.

1.3. Giao diện hệ điều hành Android

Giao diện người dùng của Android dựa trên nguyên tắc tác động trực tiếp, sử dụng cảm ứng chạm tương tự như những động tác ngoài đời thực như vuốt, chạm, kéo giãn và thu lại để xử lý các đối tượng trên màn hình. Sự phản ứng với tác động của người dùng diễn ra gần như ngay lập tức, nhằm tạo ra giao diện cảm ứng mượt mà, thường dùng tính năng rung của thiết bị để tạo phản hồi rung cho người dùng. Những thiết bị phần cứng bên trong như gia tốc kế, con quay hồi chuyển và cảm biến khoảng cách được một số ứng dụng sử dụng để phản hồi một số hành động khác của người dùng, ví dụ như điều chỉnh màn hình từ chế độ hiển thị dọc sang chế độ hiển thị ngang tùy theo vị trí của thiết bị, hoặc cho phép người dùng lái xe đua bằng xoay thiết bị, giống như đang điều khiển vô-lăng.

Các thiết bị Android sau khi khởi động sẽ hiển thị màn hình chính, điểm khởi đầu với các thông tin chính trên thiết bị, tương tự như khái niệm desktop (bàn làm việc) trên máy tính để bàn. Màn hình chính Android thường gồm nhiều biểu tượng (*icon*) và tiện ích (*widget*); biểu tượng ứng dụng sẽ mở ứng dụng tương ứng, còn tiện ích hiển thị những nội dung sống động, cập nhật tự động như dự báo thời tiết, hộp thư của người dùng, hoặc những mẫu tin thời sự ngay trên màn hình chính. Màn hình chính có thể gồm nhiều trang xem được bằng cách vuốt ra trước hoặc sau, mặc dù giao diện màn hình chính của Android có thể tùy chỉnh ở mức cao, cho phép người dùng tự do sắp đặt hình dáng cũng như hành vi của thiết bị theo sở thích. Những ứng dụng do các hãng thứ ba có trên Google Play và các kho ứng dụng khác còn cho phép người dùng thay đổi "chủ đề" của màn hình chính, thậm chí bắt chước hình dáng của hệ điều hành khác như Windows Phone chẳng hạn. Phần lớn những nhà sản xuất, và một số nhà mạng, thực hiện thay đổi hình dáng và hành vi của các thiết bị Android của họ để phân biệt với các hãng cạnh tranh.

Ở phía trên cùng màn hình là thanh trạng thái, hiển thị thông tin về thiết bị và tình trạng kết nối. Thanh trạng thái này có thể "kéo" xuống để xem màn hình thông báo gồm thông tin quan trọng hoặc cập nhật của các ứng dụng, như email hay tin nhắn SMS mới nhận, mà không làm gián đoạn hoặc khiến người dùng cảm thấy bất tiện. Trong các phiên bản đời đầu, người dùng có thể nhấn vào thông báo để mở ra ứng dụng tương ứng, về sau này các thông tin cập nhật được bổ sung thêm tính năng, như có khả năng lập tức gọi ngược lại khi có cuộc gọi nhớ mà không cần phải mở ứng dụng gọi điện ra. Thông báo sẽ luôn nằm đó cho đến khi người dùng đã đọc hoặc xóa nó đi.

1.4. Quá trình phát triển phiên bản Android

1.4.1. Phiên bản Android 1.0

Ngày ra mắt : 23/9/2008

Lần đầu xuất hiện, phiên bản Android đời đầu góp mặt trên chiếc điện thoại J1 của nhà mạng T-Mobile [2]. So với những hệ điều hành di động phổ biến thời điểm đó, Android 1.0 đem đến nhiều trải nghiệm mới mẻ độc đáo như màn hình Home Screen, thanh thông báo Notifications được kích hoạt theo phương thức vuốt dọc màn hình theo chiều từ trên xuống.

Ngoài ra, giới công nghệ còn hết sức ấn tượng với khả năng liên kết với Google Gmail của Android 1.0. Sản phẩm làm nên tên tuổi cho Android sau đó là HTC Cream, ra mắt 1 năm sau đó.



Hình 1.4.1.1 Phiên bản Android 1.0 ra đời năm 2008

1.4.2. Phiên bản Android 1.5 : Cup Cake

Ngày ra mắt: 27/4/2009

Rút kinh nghiệm từ Android 1.0, trên phiên bản Android 1.5 này, Google đã đem tới nhiều cải tiến đáng chú ý như việc cải tiến giao diện người sử dụng, hỗ trợ bàn phím ảo và dự đoán từ khi nhập văn bản. Một điều thú vị về Android 1.5 là kể từ phiên bản này, ban lãnh đạo Google bắt đầu đặt tên cho hệ điều hành theo các loại bánh kẹo, bắt đầu từ Cup Cake.

1.4.3. Phiên bản Android 2.0 và Android 2.1 : Eclair

Ngày ra mắt: 26/10/2009

Phiên bản 2.0 của hệ điều hành Android đem tới tiềm năng phát triển lớn hơn và bắt đầu góp mặt trên rất nhiều thiết bị di động "khủng", được nhiều nhà mạng danh tiếng của Mỹ phân phối, điển hình như chiếc Motorola Droid. Android 2.0 cho phép hỗ trợ nhiều tài khoản người dùng, bổ sung thêm bàn phím QWERTY ảo.

1.4.4. Phiên bản Android 2.2 : Froyo

Ngày ra mắt: 20/5/2010

Tới phiên bản 2.2 Froyo, gã khổng lồ phần mềm tiếp tục cập nhật một số tính năng điểm nhấn như Wi-Fi Hotspot, cho phép người dùng cài đặt trực tiếp ứng dụng vào thẻ nhớ ngoài của thiết bị. Điều đáng nhắc tới là từ Froyo, Google bắt đầu chú trọng tới việc xây dựng giao diện của Android để tăng tính thẩm mỹ.

1.4.5. Phiên bản Android 2.3: Gingerbread

Ngày ra mắt: 6/12/2010

Tính tới thời điểm phiên bản Gingerbread xuất hiện, hệ điều hành Android đã chứng minh được sức hút không thể cưỡng lại khi đã phủ sóng tới 54% số lượng thiết bị di động trên thị trường. Cũng trong năm này, Samsung bắt tay với Google để tung ra chiếc điện thoại Nexus S. Ưu thế của Android 2.3 nằm ở việc hệ điều hành này cho phép hỗ trợ camera trước, đi kèm với giao diện đẹp mắt và giúp tiết kiệm điện.



Hình 1.4.5.1. Giao diện hệ điều hành Gingerbread

1.4.6. Phiên bản Android 3.x : Honeycomb

Ngày ra mắt: 10/5/2011

Đây là phiên bản Android được phát triển chuyên để hỗ trợ các loại máy tính bảng, chiếc tablet đầu tiên được vinh dự cài đặt Android 3 là Motorola Xoom. Tuy không để lại nhiều dấu ấn quan trọng nhưng Honeycomb chính là nền móng để Google phát triển Android 4.0 thành công rực rỡ.

1.4.7. Phiên bản Android 4.0 : Ice Cream Sandwich

Ngày ra mắt: 16/12/2011

Bước vào cuối năm 2011, Google bắt đầu tung ra thị trường mẫu điện thoại Galaxy Nexus chạy hệ điều hành Android 4.0 Ice Cream Sandwich. Nhiều chuyên gia đánh giá đây là phiên bản chứa đựng nhiều sự thay đổi mạnh mẽ nhất qua các thế hệ Android. Trên bản 4.0 này, Google đã làm mới hoàn toàn hệ thống thông báo Notification sao cho tiện dụng và đẹp mắt hết sức có thể, đặc biệt là việc cho phép người dùng trượt ngón tay để xóa các thông báo. Công cụ bàn phím cũng được tinh chỉnh khả năng sửa lỗi tự động cũng như thao tác sao chép và dán nội dung. Đáng nói nhất là việc Google quyết định thống nhất một hệ điều hành duy nhất cho cả smartphone và tablet.

1.4.8. Phiên bản Android 4.1: Jelly Bean

Ngày ra mắt: 9/7/2012

Phiên bản 4.1 Jelly Bean ra mắt cùng với chiếc tablet Nexus 7. Không có nhiều sự thay đổi về mặt giao diện so với thời Android 4.0. Đáng kể nhất là việc Google đưa Google Now vào hoạt động nhằm tăng tính cạnh tranh với Siri của Apple. Điều quan trọng của lần cập nhật này là Project Butter với khả năng đem tới độ mượt mà đáng kể cho Android.

1.4.9. Phiên bản Android 4.2: Jelly Bean

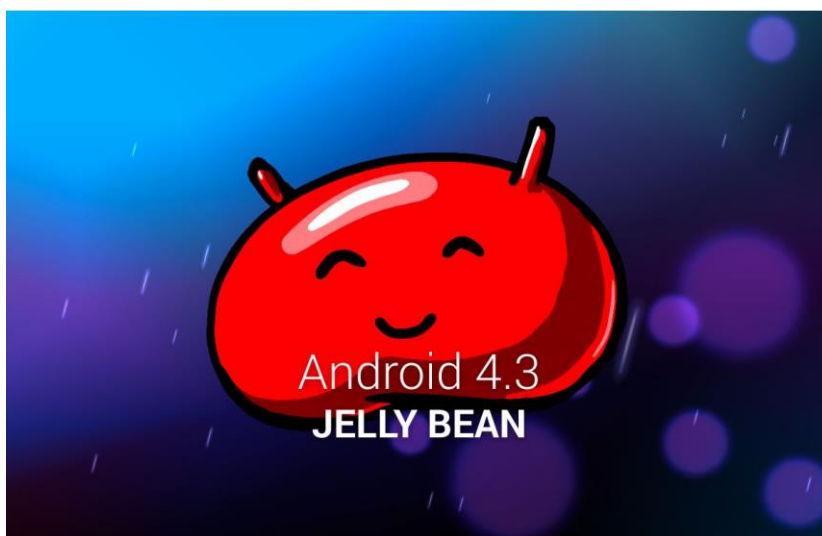
Ngày ra mắt: 13/11/2012

Tiếp tục đem cái tên Jelly Bean lên bản cập nhật 4.2, Google đem tới nhiều trải nghiệm mới mẻ như việc nhập liệu bằng nét vẽ, Miracast và chế độ chụp ảnh toàn cảnh Photo Sphere... Đến với Android 4.2 Jelly Bean, người dùng cũng có thể sử dụng nhiều tài khoản trên máy tính bảng nhằm tiện lợi hơn trong việc chia sẻ thông tin cho các thành viên ở nơi làm việc.

1.4.10. Phiên bản Android 4.3: Jelly Bean

Ngày ra mắt: 25/7/2013

Nhân dịp ra mắt sản phẩm Nexus 7, Google giới thiệu tới mọi người phiên bản Android 4.3 Jelly Bean cùng hàng loạt các tính năng mới như API OpenGL ES 3.0, Bluetooth smart và khả năng định vị bằng Wi-Fi. Đồng thời, hãng cũng hoàn thiện khả năng tương tác giữa nhiều tài khoản và cấp quyền cho cha mẹ để đề phòng trẻ em mua sắm app tốn hàng trăm USD.



Hình 1.4.10.1. Phiên bản Android 4.3

CHƯƠNG 2 : GIỚI THIỆU VỀ MÔI TRƯỜNG ANDROID STUDIO

2.1. Giới thiệu Android studio

Android Studio là một môi trường phát triển tích hợp (IDE) được google xây dựng và cung cấp miễn phí cho các nhà phát triển ứng dụng Android. Android studio dựa vào IntelliJ IDEA, là một IDE tốt cho nhất Java hiện nay. Do đó Android Studio sẽ là môi trường phát triển ứng dụng tốt nhất cho Android.

Một số nội dung tìm hiểu :

- Cách Download và cài đặt Android Studio.
- Sơ lược về sử dụng Android Studio.

Một số yêu cầu cấu hình lưu ý trước khi cài đặt:

- Microsoft® Windows® 8/7/Vista (32 or 64-bit)
- Tối thiểu 4 GB RAM.
- Còn trống 400 MB dung lượng ổ cứng + Ít nhất 1 G cho Android SDK, hình ảnh hệ thống giả lập và bộ nhớ cache.
- Độ phân giải tối thiểu 1366 x 768
- Java Development Kit.
- Lựa chọn thêm cho accelerated emulator: Intel® processor with support for Intel® VT-x, Intel® EM64T (Intel® 64), and Execute Disable (XD) Bit functionality.

2.2. Cài đặt môi trường lập trình Android

2.2.1. Cài đặt JAVA JDK

Bước 1: Tải file cài đặt từ đường dẫn

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

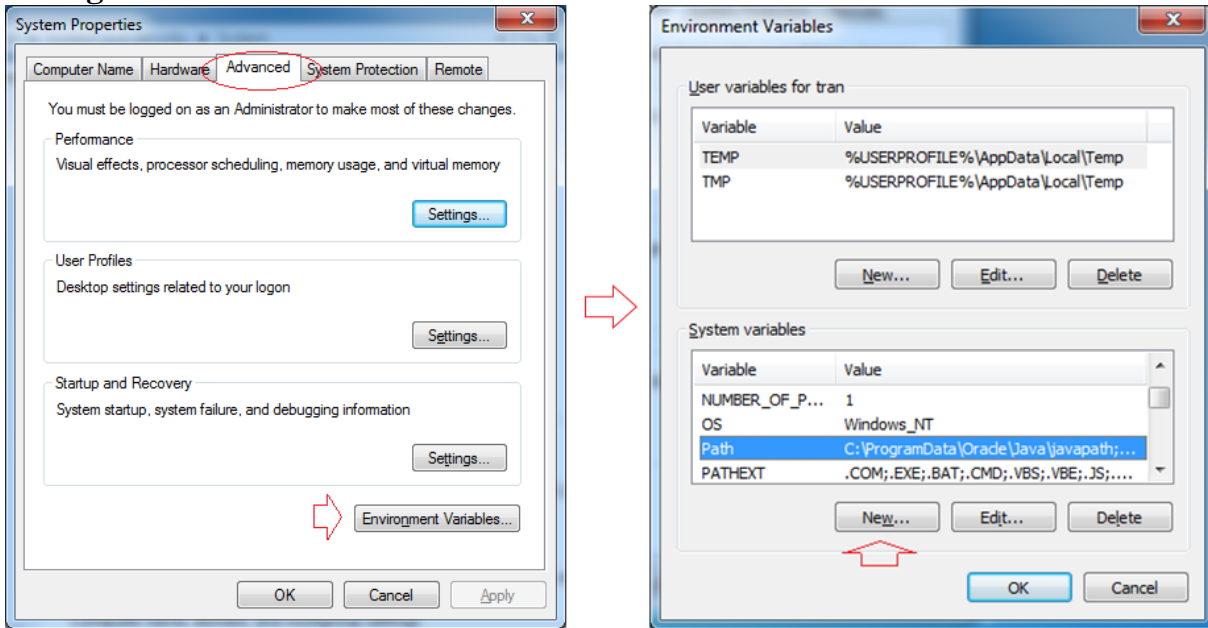
Lưu ý: Chọn phiên bản tương ứng với hệ điều hành đúng với máy đang sử dụng.

Bước 2: Mở file cài đặt “jdk-*.exe” để tiến hành cài đặt

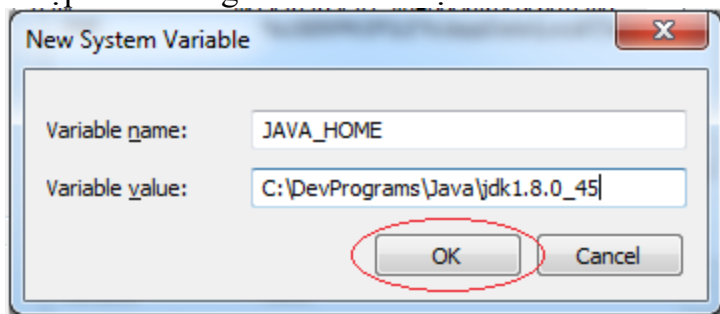
Bước 3: Cấu hình biến môi trường cho Java.

Việc này không bắt buộc, nhưng nếu trên máy tính của cài đặt nhiều phiên bản Java, thì việc cấu hình là cần thiết để xác định phiên bản java nào mặc định được sử dụng.

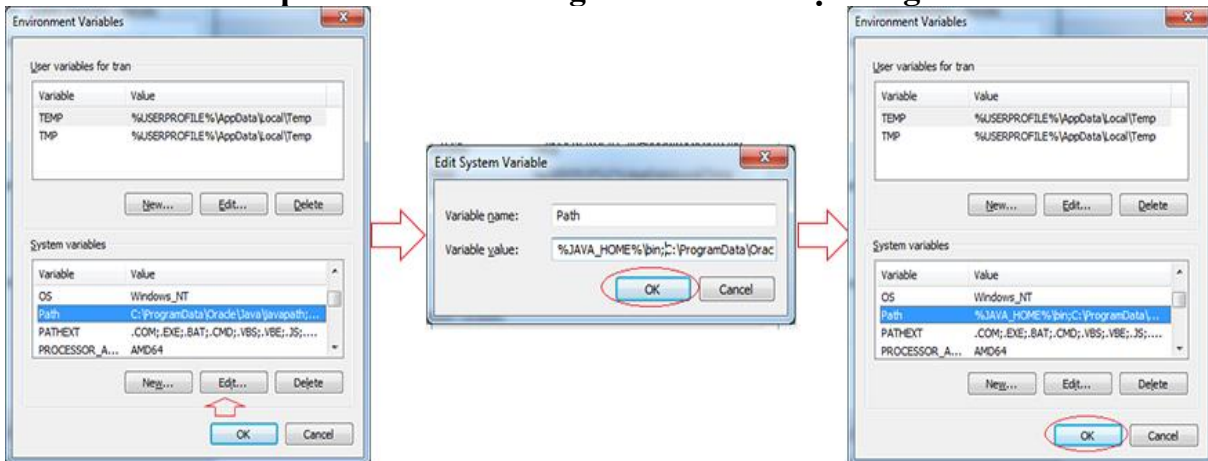
Trên Desktop, nhấn phải chuột vào Computer, chọn **Properties > Advanced system settings**.



Nhập vào đường dẫn tới thư mục JDK:



Tiếp theo đổi tên đường dẫn là đã cài đặt xong Java JDK:

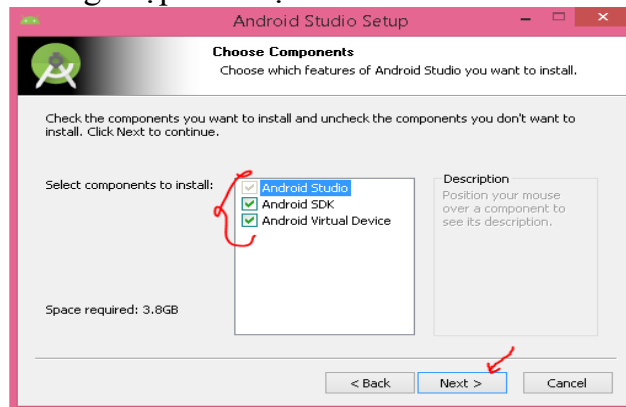


2.2.2. Cài đặt Android Studio

Download file cài đặt: <https://developer.android.com/studio/>



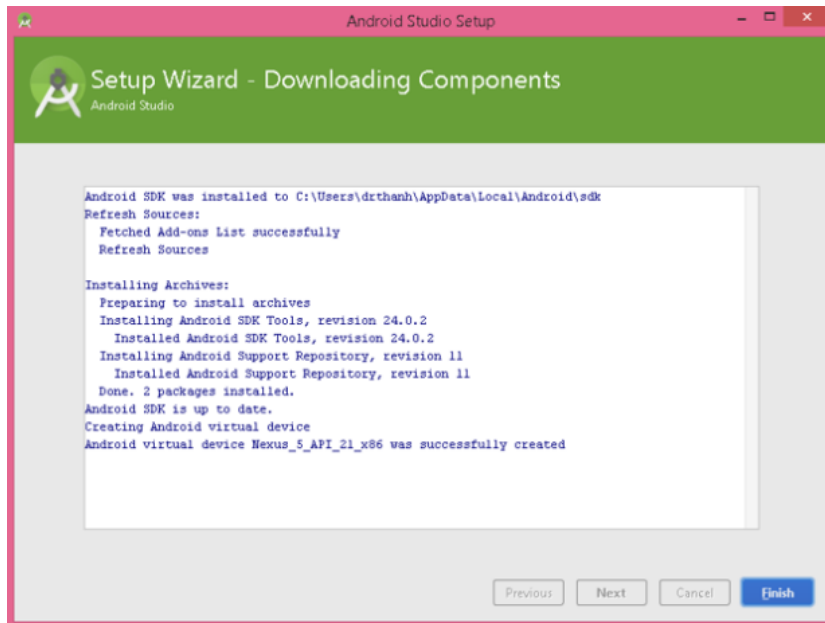
Chọn cả SDK và trình giả lập thiết bị android ảo



Cứ tiếp tục chọn next cho đến khi hoàn tất.



Đây là màn hình khởi động. Lần đầu tiên sẽ hơi lâu.



Lần đầu nó tiên chạy nó sẽ xuất hộp thoại để download vài thứ cần thiết (cứ chờ cho nó download xong).

2.2.3. Cài đặt máy ảo Android Genymotion

a. Yêu cầu

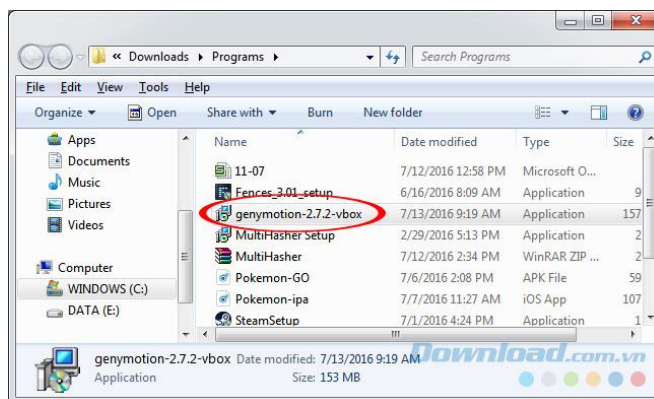
- Card đồ họa OpenGL 2.0.
- CPU hỗ trợ VT-x hoặc AMD-V và kích hoạt thiết lập BIOS.
- RAM: Tối thiểu 4 GB.
- Dung lượng trống của ổ cứng: Ít nhất là 2GB để cài đặt Genymotion và các máy ảo chạy Genymotion (đây chỉ là mức yêu cầu tối thiểu, bởi nếu sử dụng nhiều máy ảo cùng lúc và có nhiều ứng dụng, phần mềm cài đặt thì dung lượng trống có thể được yêu cầu nhiều hơn gấp 4 lần).
- Đảm bảo có kết nối Internet
- Độ phân giải màn hình: ít nhất 1366 x 768 pixel
- Oracle **VirtualBox 4.1** trở lên.
- Ngoài ra, người dùng cần có một tài khoản Genymotion để có thể sử dụng.

b. Tải file cài đặt

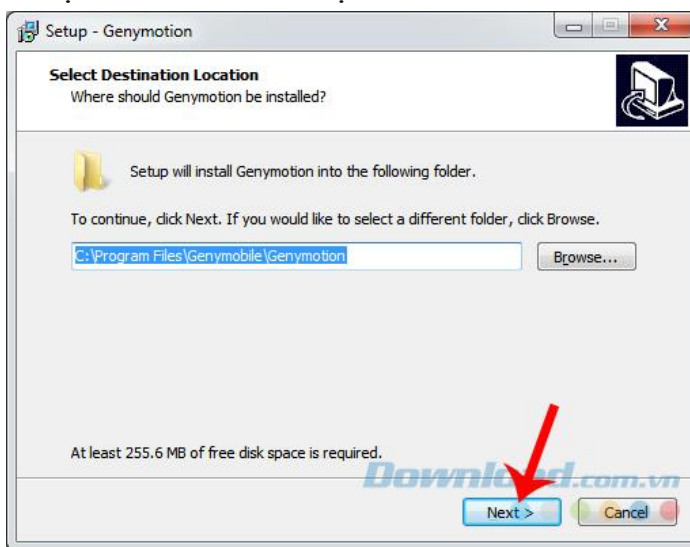
Download: <https://www.genymotion.com/fun-zone/>

c. Các bước cài đặt

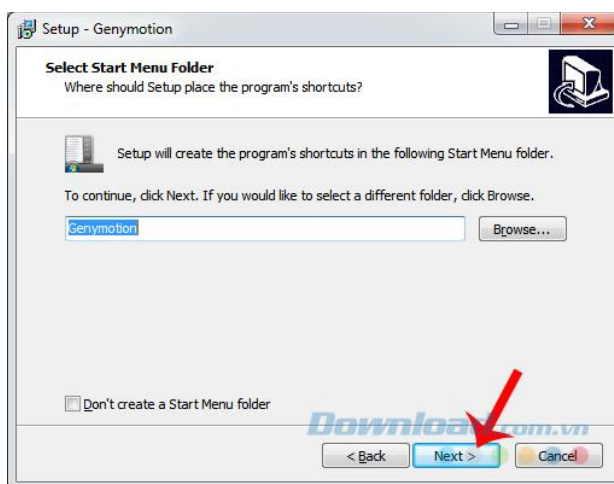
Bước 1: Tìm tới vị trí lưu file vừa tải về đầu tiên, sau đó click đúp chuột trái vào file đó để bắt đầu tiến hành.



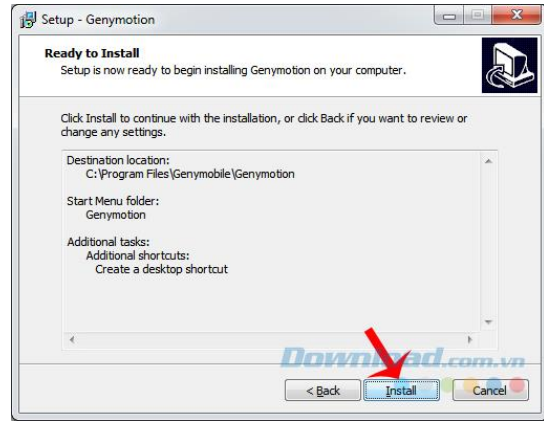
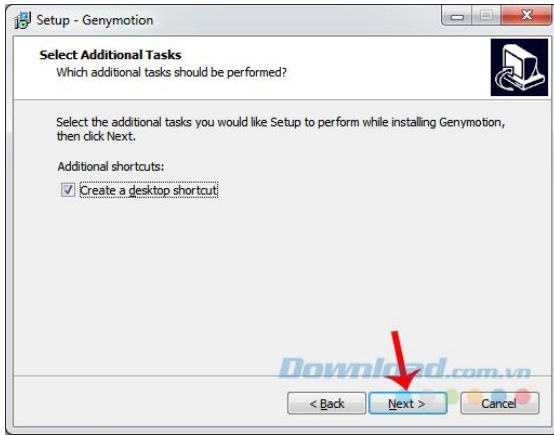
Bước 2: Tùy theo ý muốn người dùng mà có thể **Next** ngay và cài đặt Genymotion ngay tại vị trí được gợi ý hoặc chọn **Browse** để tìm vị trí khác.



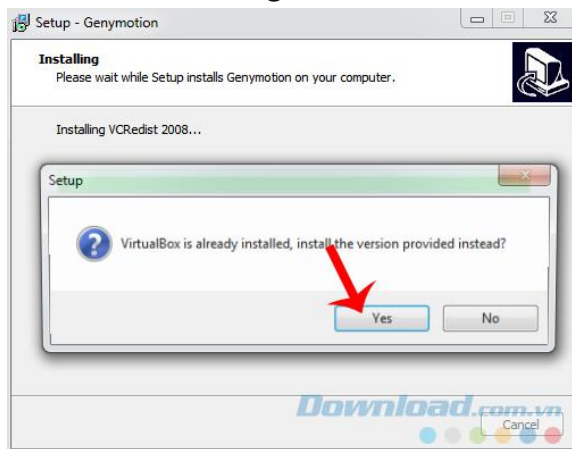
Bước 3: Tiếp tục ấn **Next**.



Bước 4: Chọn mục **Create a desktop shortcut** để tạo biểu tượng cho phần mềm này trên máy tính, sau đó **Next** tiếp.



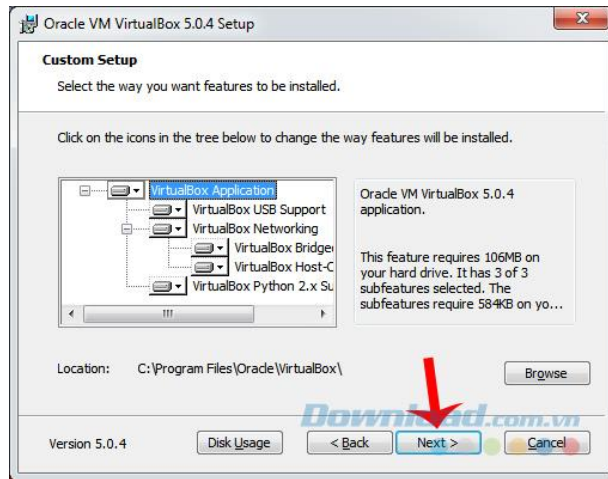
Bước 5: Rất có thể Genymotion sẽ tự động cài thêm một số phần mềm hỗ trợ, đồng thời hỏi bạn có muốn cài đặt VirtualBox hay không? Nếu máy tính của bạn đã có sẵn phần mềm VirtualBox mới nhất rồi thì thôi, còn nếu không, chọn **Yes** để cài đặt tiếp.



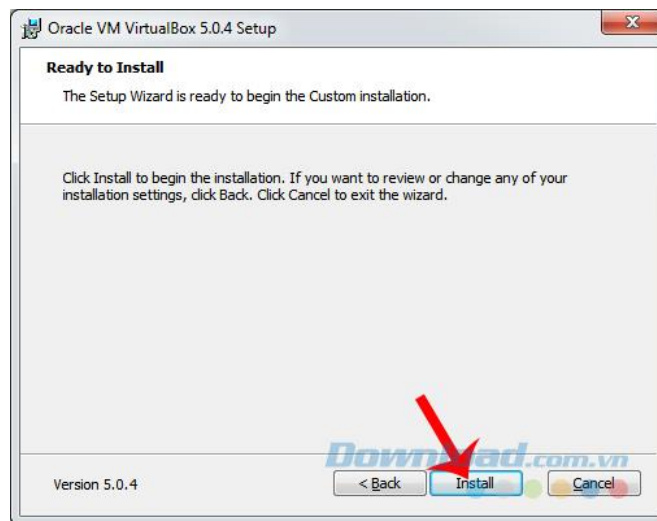
Bước 6: Tới đây, ngoài cửa sổ cài đặt Genymotion (sẽ tạm thời bị dừng lại), chúng ta sẽ thấy có thêm một giao diện khác để tiến hành cài đặt VirtualBox.



Bước 7: Next tới khi kết thúc.



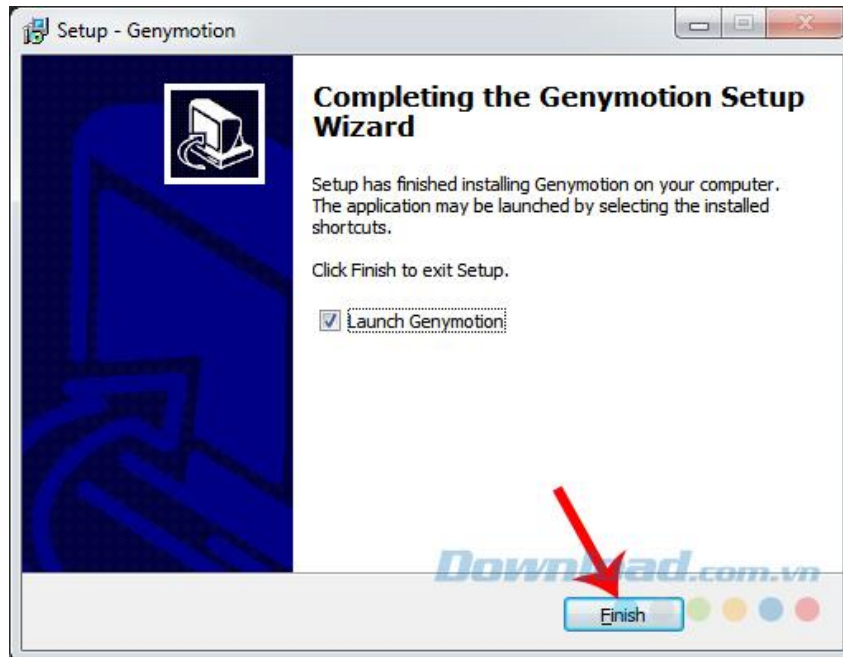
Bước 8: Cài đặt : Nhấn Install.



Bước 9: Bỏ dấu (✓) ở Start Oracle WM VirtualBox và chọn vào Finish.

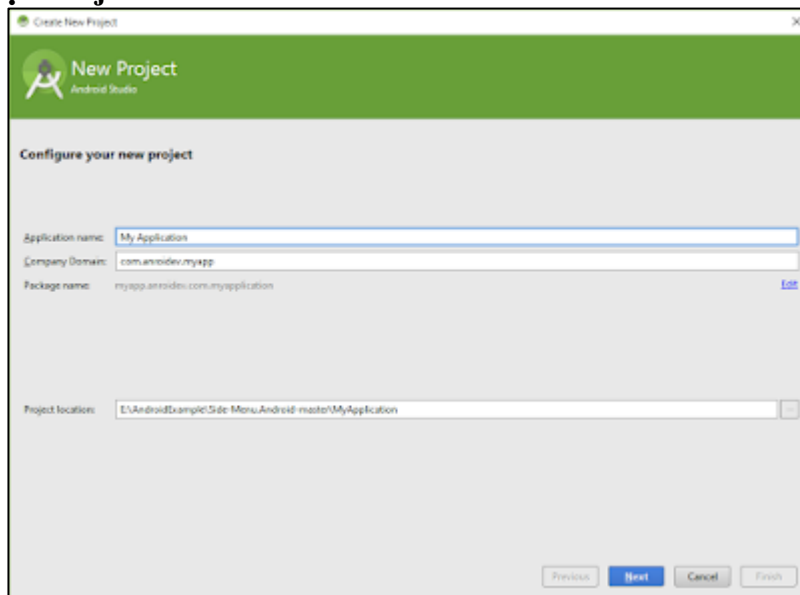


Bước 10: Lúc này cửa sổ cài đặt cuối cùng của Genymotion xuất hiện, Click vào **Finish** để kết thúc quá trình.

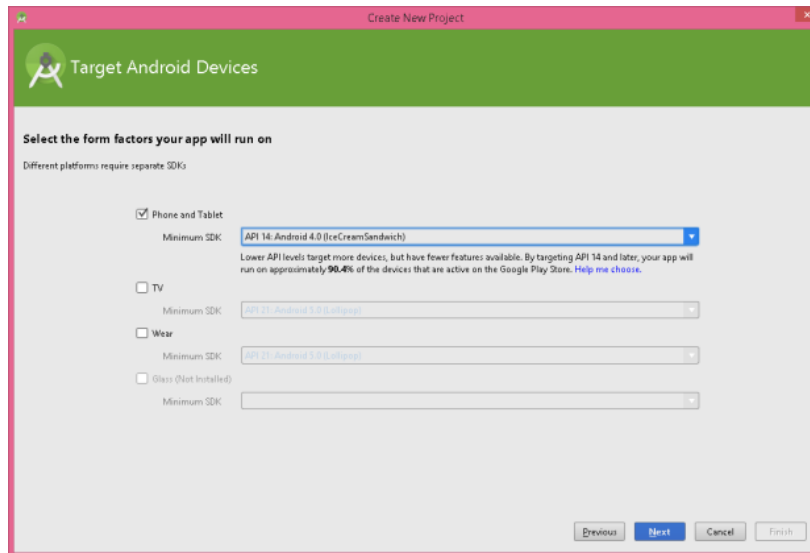


2.3. Sơ lược về sử dụng Android Studio

2.3.1. Tạo mới một Project

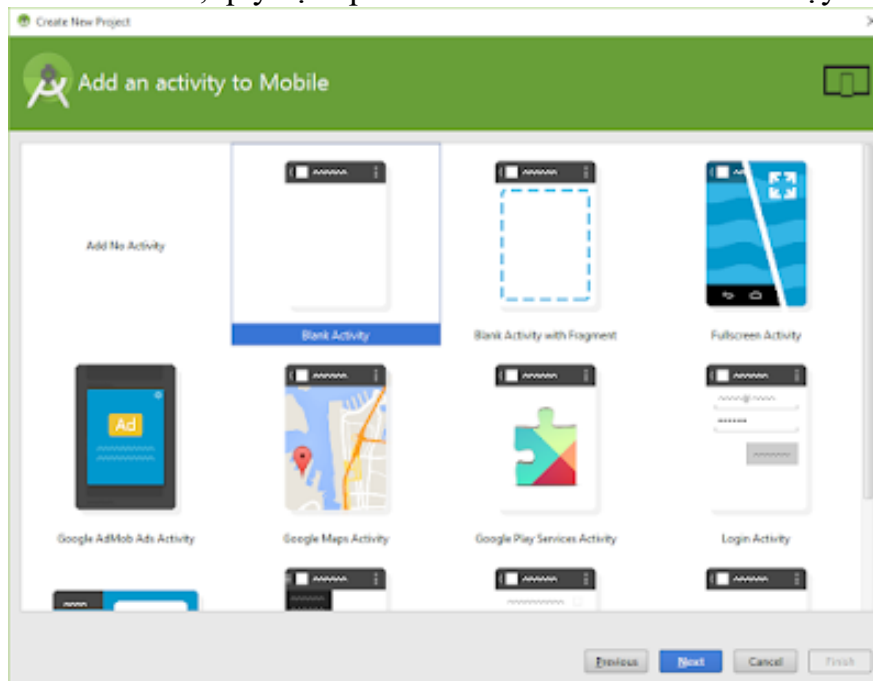


- **Application Name:** Tên ứng dụng muốn đặt
- **Company Domain:** Tên domain công ty, thường được dùng để kết hợp với tên Application để tạo thành Package (chú ý viết thường hết và có ít nhất 1 dấu chấm).
- **Package name:** Nó sẽ tự động nối ngược Company Domain với Application name.
- **Project location:** Là nơi lưu trữ ứng dụng.
Sau đó click chuột vào nút **Next** để tiếp tục.

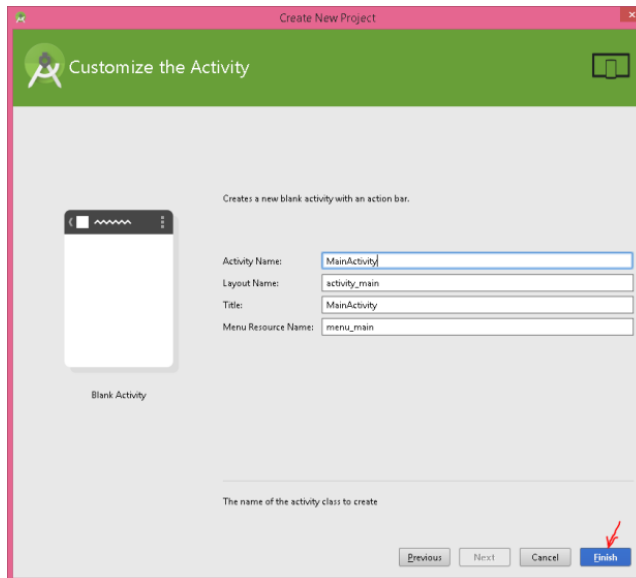


Ở hộp thoại trên cho phép bạn lựa chọn là ứng dụng của bạn sẽ được viết cho những thiết bị nào (Phone and Tablet, TV, Wear).

Ở mục Minimum SDK, quy định phiên bản android tối thiểu để chạy ứng dụng.



Màn hình này hiển thị cho phép chọn loại Activity mặc định.
Chọn **Blank Activity** rồi bấm Next:



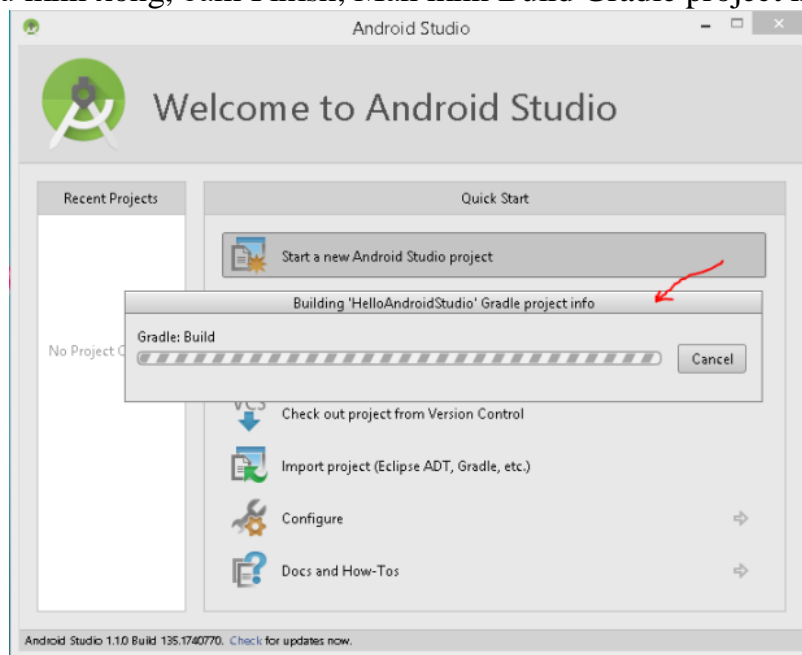
Activity Name: Tên class Activity (java) để ta viết mã lệnh

Layout Name: Tên file XML làm giao diện cho Activity Name.

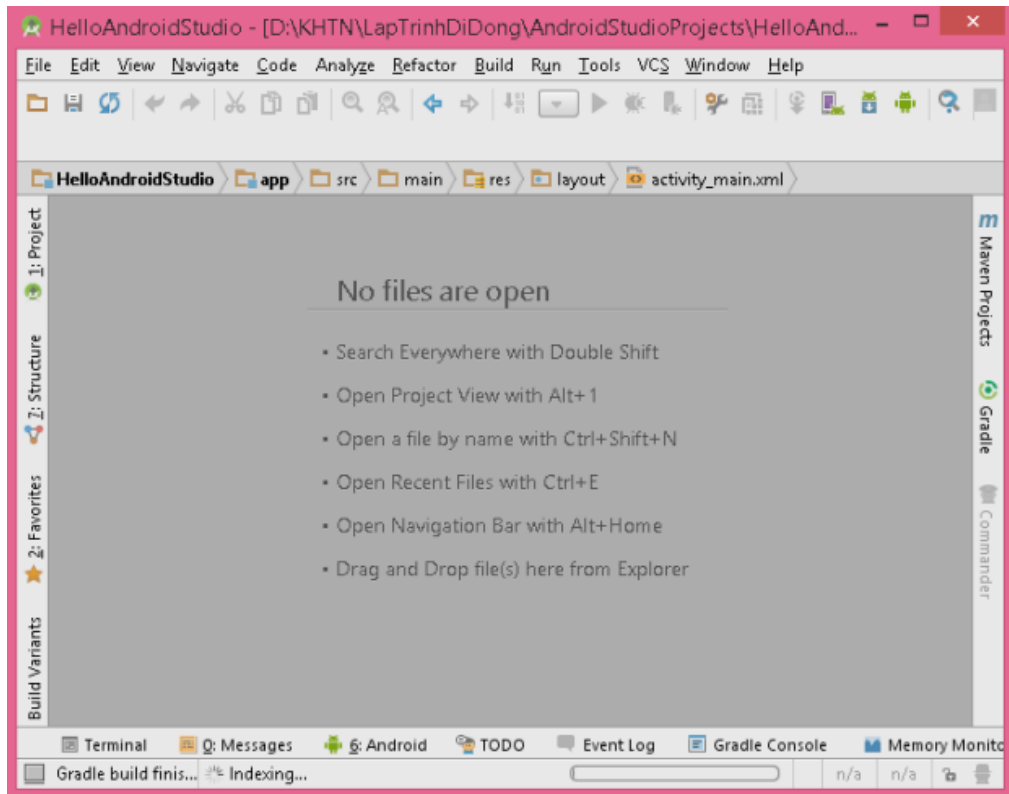
Title: Tiêu đề hiển thị khi kích hoạt Activity trên thiết bị.

Menu Resource Name: Tên file xml để tạo menu cho phần mềm.

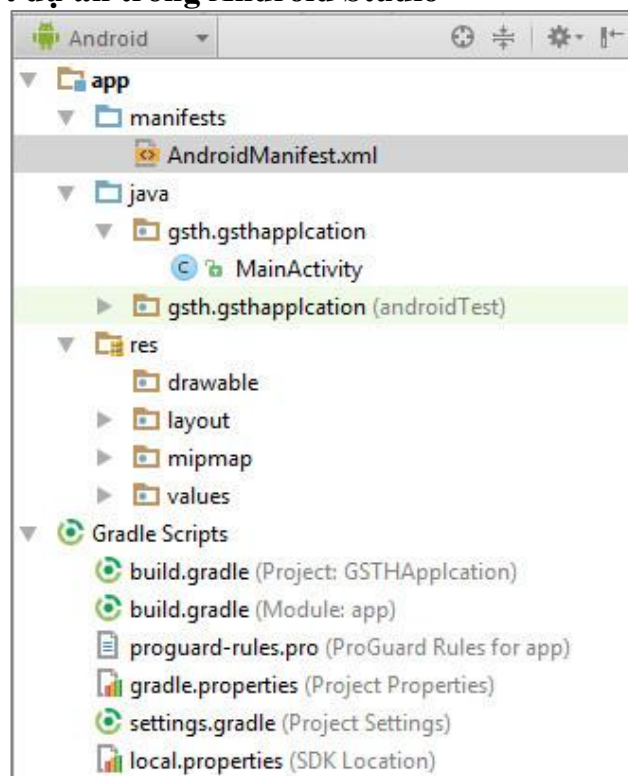
Sau khi cấu hình xong, bấm Finish, Màn hình Build Gradle project hiển thị:



Khi Android Studio tự động build dự án ta có màn hình dưới đây.



2.3.2. Cấu trúc của một dự án trong Android Studio



Hình 2.3.2.1. Cấu trúc của project

a. Thư mục app

Đây là thư mục chứa toàn bộ tài nguyên của dự án : cấu hình, file code,...

AndroidManifest.xml : Nơi khai báo các Activity trong ứng dụng.

Quy định Activity nào thực thi đầu tiên.

```
<intent-filter>
```

```
<action android:name="android.intent.action.MAIN" />
```



```
<category android:name="android.intent.category.LAUNCHER" />
```

```
</intent-filter>
```

Cấp quyền cho ứng dụng như quyền truy cập internet, quyền ghi nhớ,...

```
<uses-permission android:name="android.permission.INTERNET"/>
```

MainActivity : Là nơi viết code bằng java.

```
package gsth.gsthapplication;
```

```
import android.support.v7.app.AppCompatActivity;
```

```
import android.os.Bundle;
```

```
public class MainActivity extends AppCompatActivity {
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);
```

```
    }
```

```
}
```

Câu lệnh `setContentView(R.layout.activity_main);` dùng để chỉ định layout.

b. Thư mục res



Đây là thư mục chứa tài nguyên của Project, thư mục này chứa các file giao diện, hình ảnh,... mà ứng dụng cần đến khi chạy.

- Thư mục **drawable**: Đây là thư mục chứa các file hình ảnh của ứng dụng.
- Thư mục **layout**: Ở đây chứa các file giao diện của ứng dụng.
- Thư mục **menu**: Chứa các file giao diện của menu, các bạn dùng điện thoại Android thường thấy hầu hết các ứng dụng đều có thanh menu, thanh menu đó màu sắc ra sao,

hiển thị thế nào đều là do định dạng của các file giao diện chứa trong thư mục **menu** này.

- Thư mục **mipmap**: Nơi chứa icon của ứng dụng.
- Thư mục **values**: Đây là nơi quản lý các giá trị, ở trong này có các file **strings.xml**, **styles.xml** và thư mục **dimens.xml**.

c. Gradle Script

Quy định phiên bản SDK.

Khai báo thư viện sử dụng.

```
apply plugin: 'com.android.application'

android {

    compileSdkVersion 23

    buildToolsVersion "23.0.1"

    defaultConfig {

        applicationId "gsth.gsthapplication"

        minSdkVersion 19

        targetSdkVersion 23

        versionCode 1

        versionName "1.0"

    }

    buildTypes {

        release {

            minifyEnabled false

            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'

        }

    }

}

dependencies {

    compile fileTree(dir: 'libs', include: ['*.jar'])

    testCompile 'junit:junit:4.12'

    compile 'com.android.support:appcompat-v7:23.1.1'

}
```

2.4. Quản lý màn hình điện thoại

Một màn hình điện thoại được quản lý bằng 2 tệp :

- Tệp *.xml : Tạo giao diện (Edit text, Text view, Button,..)
- Tệp *.java : Nơi viết mã lệnh để bắt và xử lý các sự kiện trên giao diện.

Lý do cần phải ánh xạ : Android Studio không cho thao tác trực tiếp trong giao diện xml.

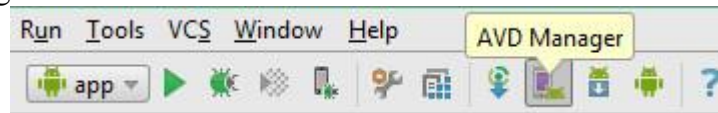
Vì thế nên chúng ta phải ánh xạ ở đối tượng điều khiển trong tệp *.java.

2.5. Chạy ứng dụng

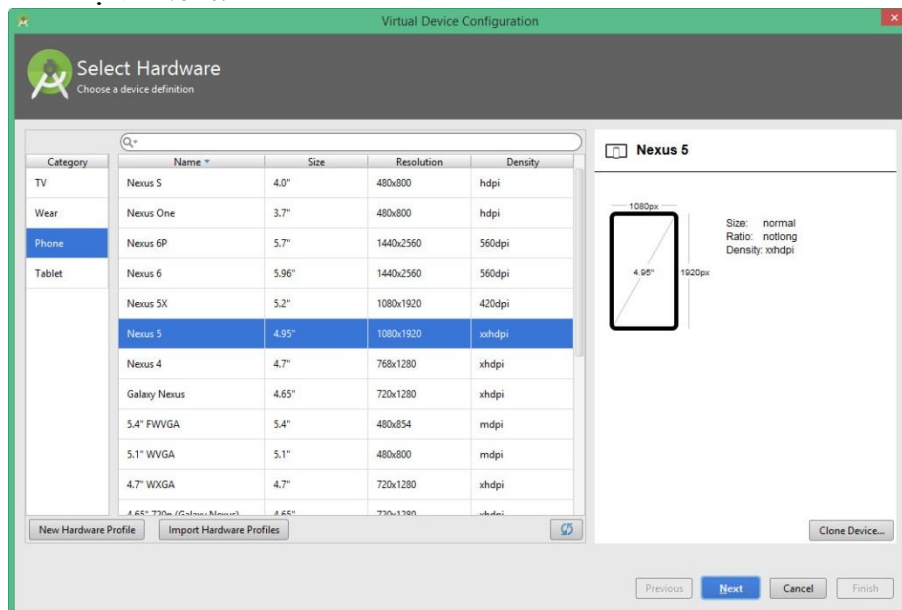
Một chương trình trước khi được đưa ra sử dụng phải trải qua việc kiểm thử. Đôi khi kiểm thử trên thiết bị thật gây khó khăn cho lập trình viên, tốn kém cho người sử dụng. Do đó, Android Studio đã hỗ trợ lập trình viên bằng cách hỗ trợ tạo ra các thiết bị ảo để kiểm thử chương trình.

Bước 1: Tạo thiết bị ảo (Emulator)

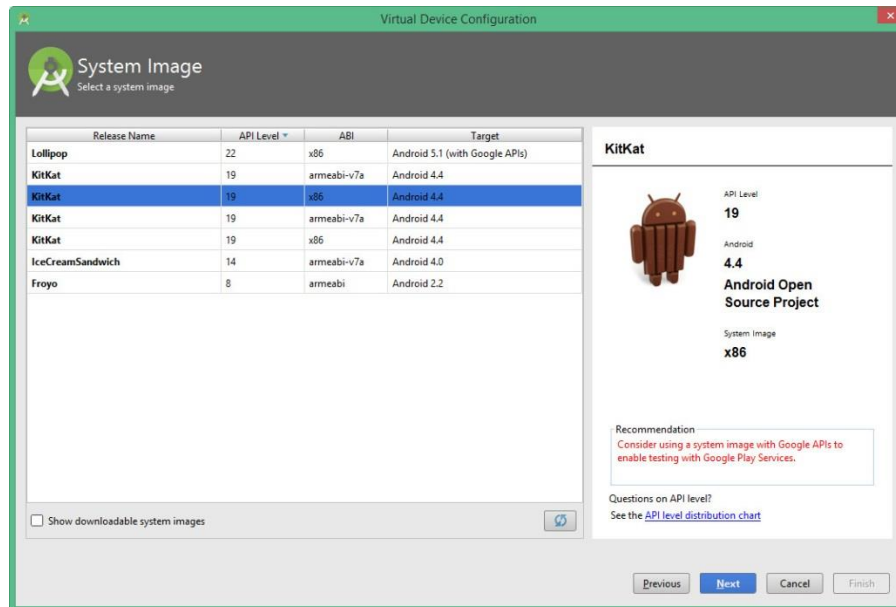
- Chọn AVD Manager



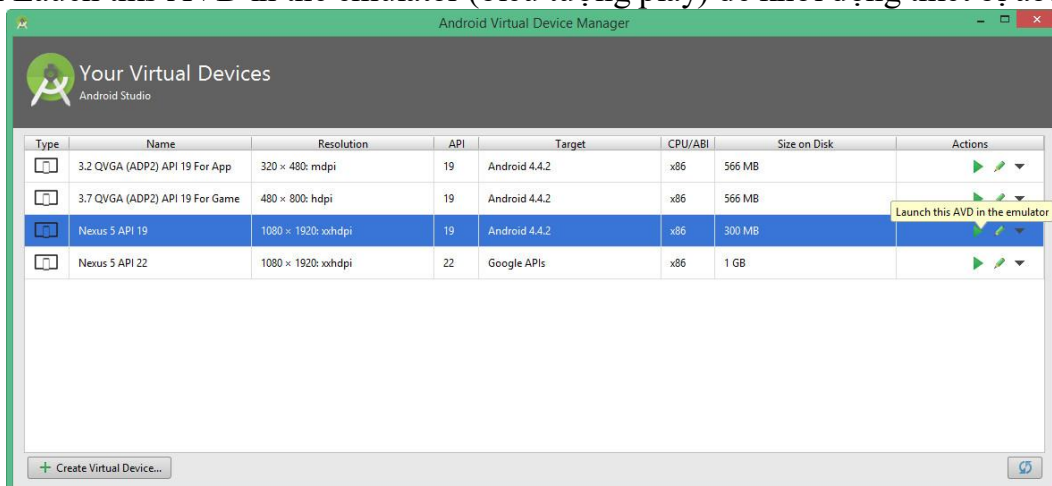
- Chọn Create Virtual Device...
- Lựa chọn thiết bị -> Next.



- Lựa chọn phiên bản Android -> Next -> Finish.



- Chọn Launch this AVD in the emulator (biểu tượng play) để khởi động thiết bị ảo.

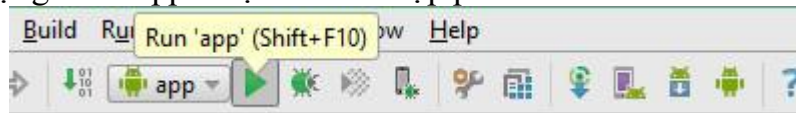


- Thiết bị ảo (emulator)

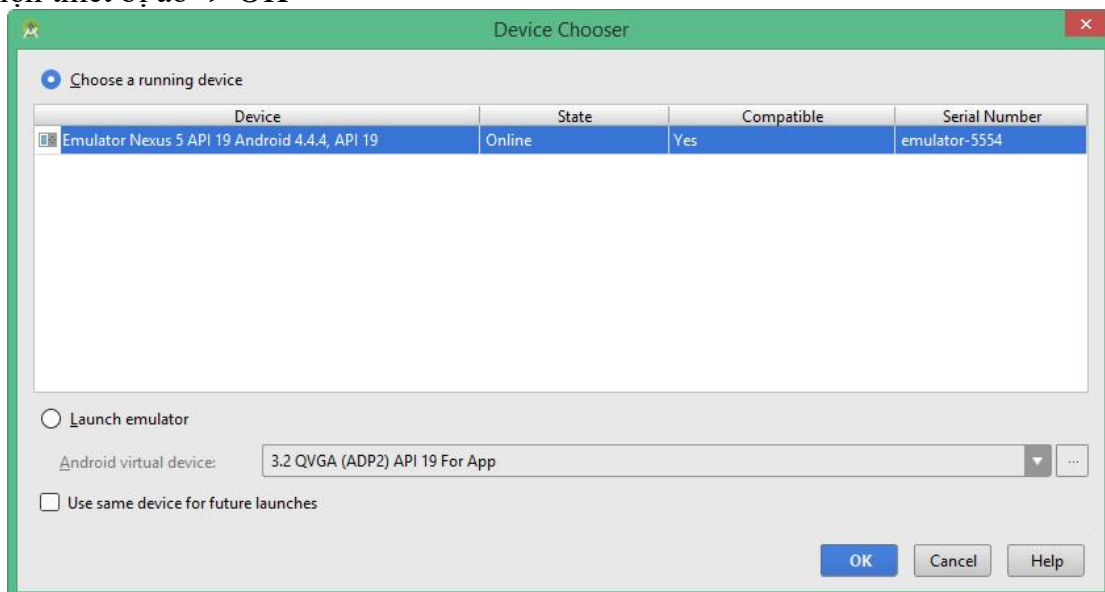


Bước 2: Chạy chương trình

- Chọn biểu tượng Run 'app' hoặc nhấn tổ hợp phím Shift + F10



- Chọn thiết bị ảo -> OK



- Ứng dụng đã được thực thi



CHƯƠNG 3 : KỸ THUẬT HỖ TRỢ XÂY DỰNG ỨNG DỤNG

3.1. Hosting

3.1.1. Khái niệm

Hosting là không gian lưu trữ trên máy chủ có cài đặt các dịch vụ Internet như world wide web (www), truyền file(FTP), Thư điện tử (Mail)[3]..., có thể chứa nội dung trang web hay dữ liệu trên không gian đó. Hosting đồng thời cũng là nơi diễn ra tất cả các hoạt động giao dịch, trao đổi thông tin giữa website với người sử dụng Internet và hỗ trợ các phần mềm Internet hoạt động. Doanh nghiệp có thể chọn thuê web hosting của nhà cung cấp dịch vụ (ISP) có dung lượng phù hợp với dung lượng website. Với bất kỳ hình thức nào (tự trang bị máy chủ hay thuê máy chủ) thì doanh nghiệp cũng nên có các hiểu biết cần thiết về Hosting và máy chủ Web.

3.1.2. Yêu cầu và tính năng cần thiết của Hosting?

- **Tốc độ:** Máy chủ chạy dịch vụ web phải có cấu hình đủ lớn để đảm bảo xử lý thông suốt, phục vụ cho số lượng lớn người truy cập. Phải có đường truyền kết nối tốc độ cao để đảm bảo không bị nghẽn mạch dữ liệu. Thông thường, nếu sử dụng Hosting nước ngoài, nên chọn máy chủ ở *Singapore* hoặc *Nhật* để có tốc độ nhanh nhất về Việt Nam.
- **Dung lượng:** Web Hosting phải có một dung lượng đủ lớn (tùy nhu cầu) để lưu giữ được đầy đủ các thông tin, dữ liệu, hình ảnh,... của website.
- **Băng thông:** Phải có bandwidth (băng thông) đủ lớn để phục vụ các hoạt động giao dịch, trao đổi thông tin giữa website với người dùng.
- **Hỗ trợ:** Lưu ý lựa chọn Hosting có chất lượng tốt, đội ngũ support nhanh chóng và hiệu quả khi có sự cố xảy ra.

3.2. Website

3.2.1. Khái niệm

Website là một tập hợp các trang web bao gồm hình ảnh, văn bản, video,... nằm trong một tên miền(domain name) hoặc tên miền phụ (submain) trên World Wide Web của Internet.

3.2.2. Vai trò

Website đóng vai trò là một văn phòng hay một cửa hàng trên mạng Internet – nơi giới thiệu thông tin về doanh nghiệp, sản phẩm hoặc dịch vụ do doanh nghiệp cung cấp... Có thể coi website chính là bộ mặt của doanh nghiệp, là nơi để đón tiếp và giao dịch với các khách hàng, đối tác trên Internet.

3.3. Cơ sở dữ liệu web

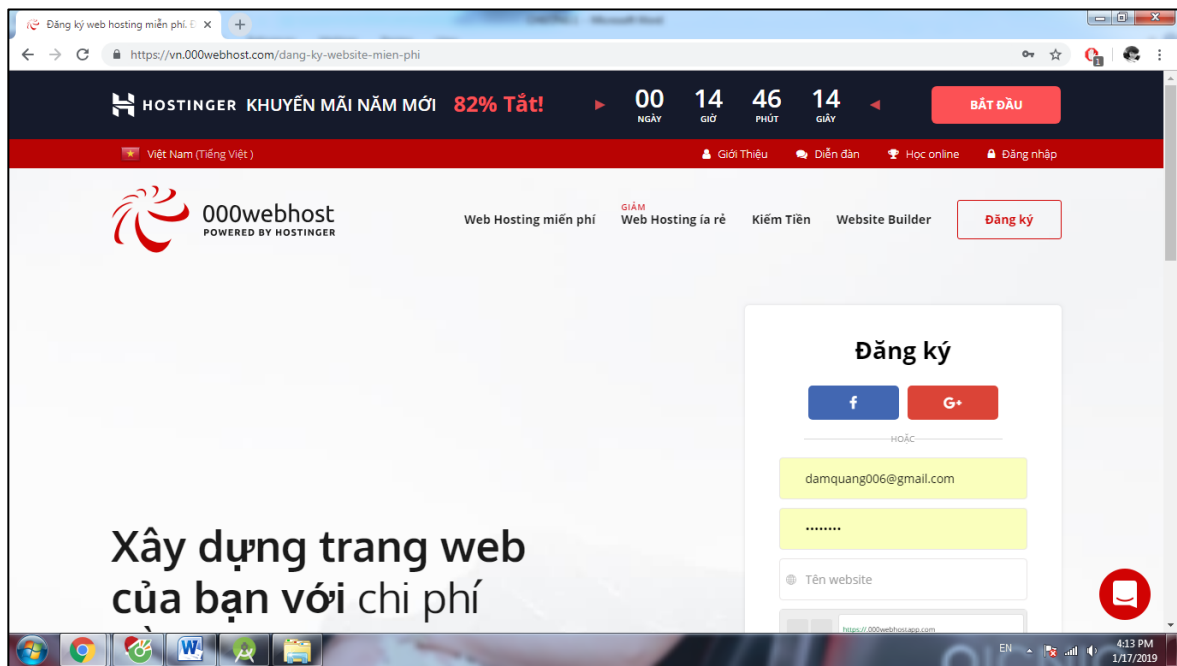
3.3.1. Khái niệm

Cơ sở dữ liệu web là cơ sở dữ liệu gắn với một trang web cụ thể. Là nền móng của một ứng dụng web, nó giúp máy tính có thể dễ dàng đọc thông tin, chỉnh sửa, thêm hoặc xóa dữ liệu.

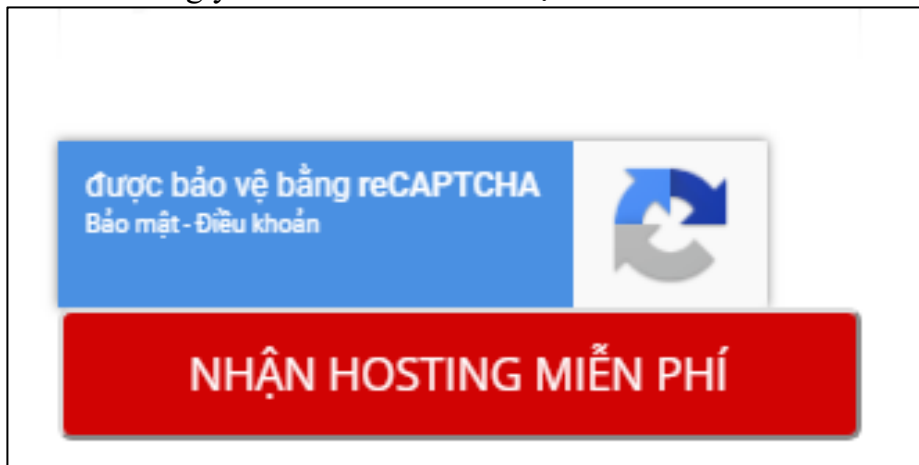
3.3.2. Xây dựng cơ sở dữ liệu web

a. Cách tạo một hosting free

Vào đường link đăng ký : <https://vn.000webhost.com/dang-ky-website-mien-phi>



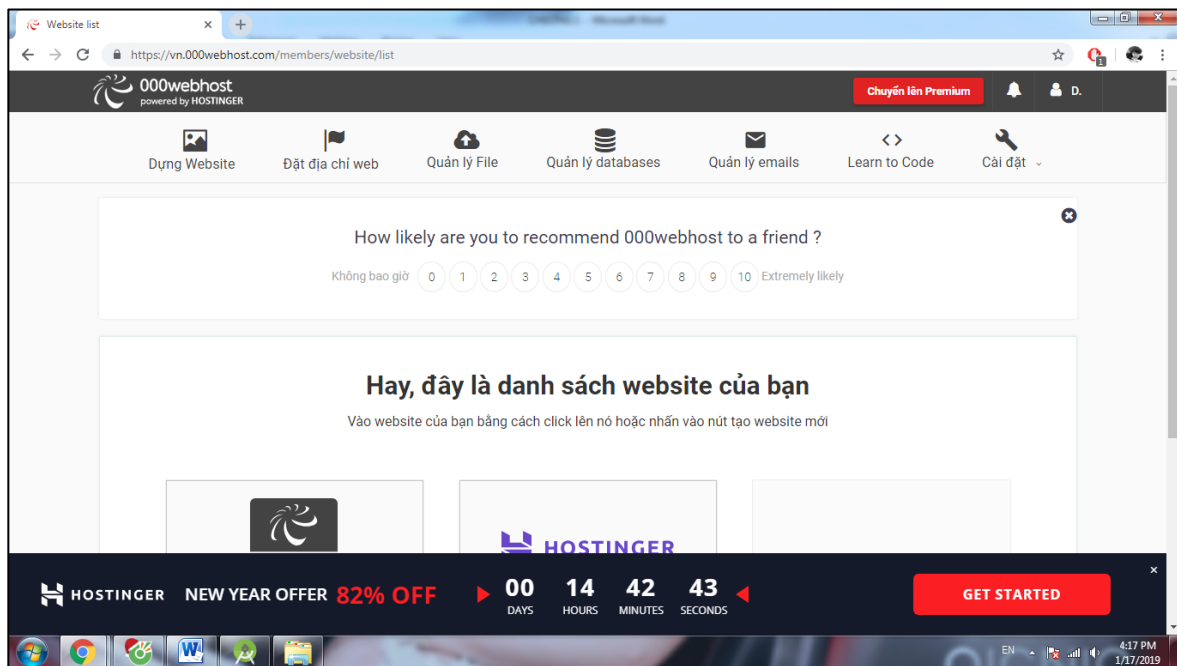
Điền theo các trường yêu cầu. Sau đó ấn **NHẬN HOSTING MIỄN PHÍ**.



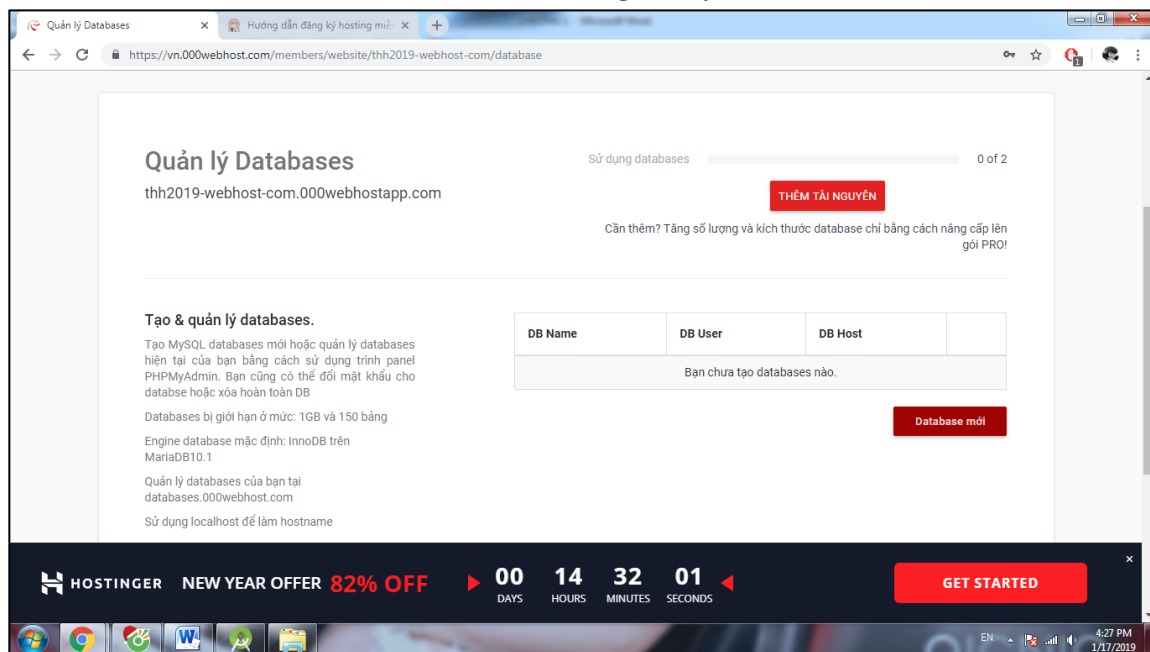
Sau đó Vào email để xác nhận đăng ký.

b.Cách sử dụng

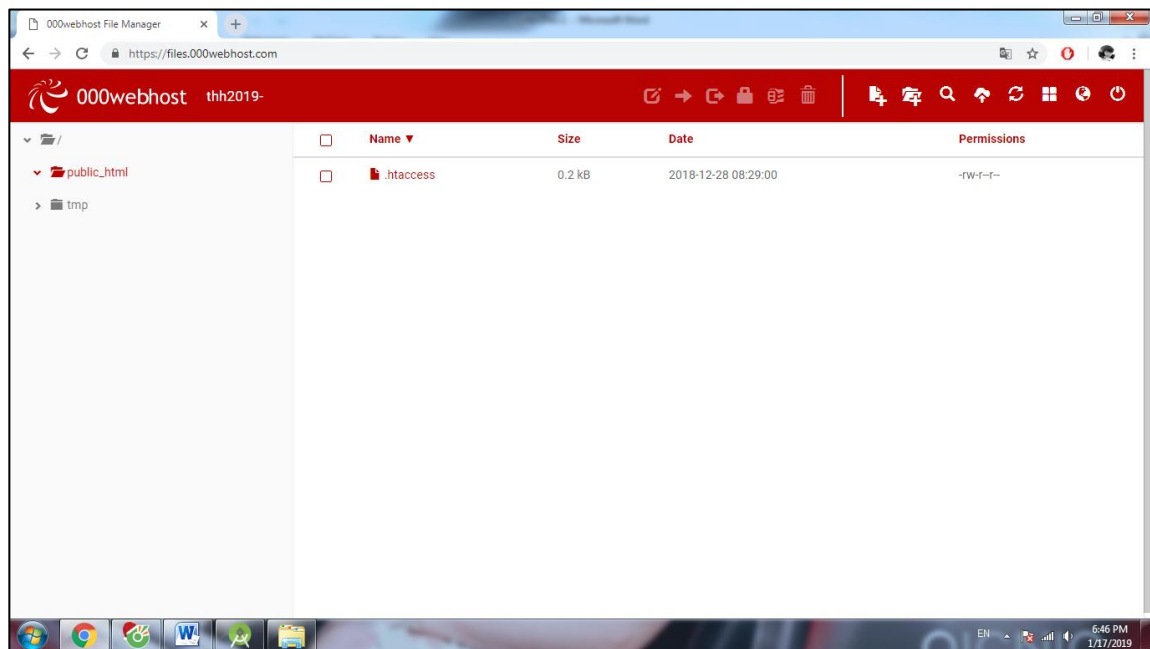
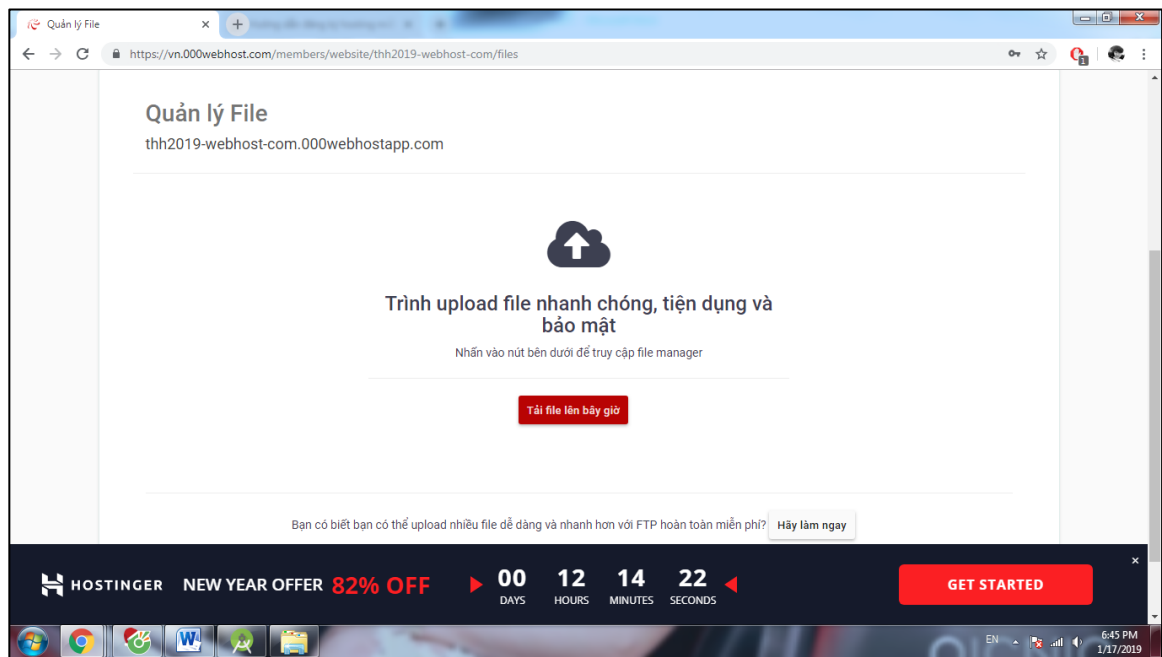
Sau khi đăng ký xong. Chúng ta bắt đầu đăng nhập.



Quản lý database : Truy cập vào đây để quản lý, tạo, xóa và chỉnh sửa các cơ sở dữ liệu MySQL, bạn chỉ có thể tạo tối đa 2 database trên gói này.

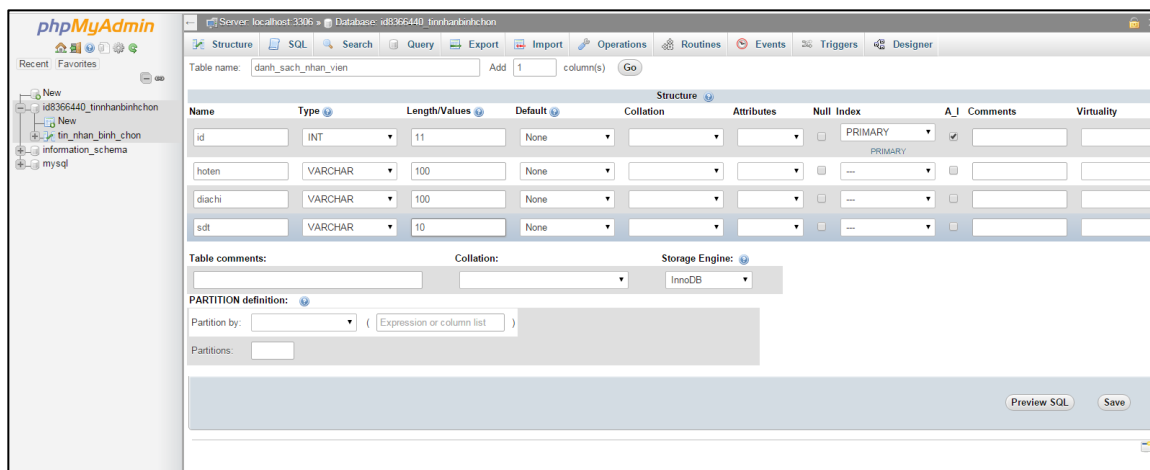


Quản lý File : Mục này cho phép bạn truy cập, xem, chỉnh sửa và tải các tập tin lên hosting. Bấm vào nút **Tải file lên bây giờ** để truy cập trình quản lý tập tin.

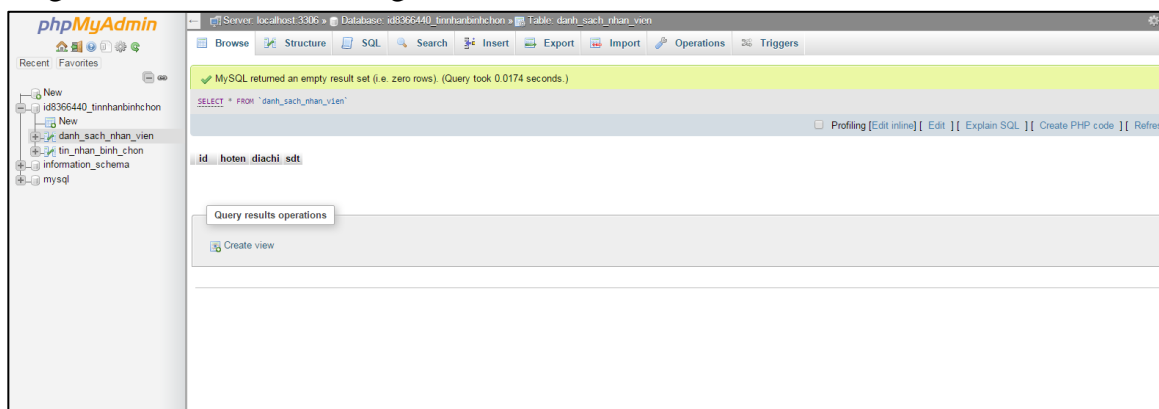


Tại đây, **Upload** những file cần sử dụng.

c. Tạo database



Tạo bảng danh_sach_nhan_vien gồm 4 cột : id, hoten, diachi, sdt.



Sau đó sẽ nhận được bảng như trên.

3.4. Dịch vụ web

3.4.1. Khái niệm

Web Service Là một hệ thống phần mềm được thiết kế để hỗ trợ khả năng tương tác giữa các ứng dụng trên máy tính khác nhau thông qua mạng Internet , giao diện chung và sự gắn kết của nó được mô tả bằng XML.

Web Service độc lập về ngôn ngữ và độc lập về nền tảng bởi vì nó tách biệt đặc tả ra khỏi cài đặt.

Web Service dựa trên kiến trúc phân tán trong đó không có bất kì dịch vụ xử lý trung tâm nào và tất cả dạng truyền thông đều xử dụng các giao thức chuẩn. Các giao thức không có bất kì ý nghĩa ngầm định nào bên trong mà phải được mô tả rõ ràng.

Web service sử dụng XML, một ngôn ngữ độc lập trong việc biểu diễn dữ liệu, làm ngôn ngữ trao đổi thông tin. Do đó, các mô hình web service có khả năng tích hợp phần mềm và đa kết nối khi kết hợp với nhau.

Các chuẩn Webservice mới được hỗ trợ các tính năng như hỗ trợ giao dịch, bảo mật,...

3.4.2. Chức năng

Web Service giúp nhận thông tin truyền lên từ ứng dụng được liên kết sẵn sau đó chèn dữ liệu lên database. Hoặc lấy thông tin từ database rồi truyền về ứng dụng.

3.4.3. Kỹ thuật xây dựng dịch vụ web cho thiết bị di động

a. Xây dựng Webservice bằng ngôn ngữ PHP

Đầu tiên ta viết một file PHP để liên kết với database đã tạo sẵn.

```

1 <?php
2     $host='localhost';
3     $username='id8366440_duyquang';
4     $password='duyquang';
5 >?>

```

Khai báo tên host, user name, password của database.

```

$link=mysqli_connect($host,$username,$password);
if(!$link)
{
    die('Could not connect to Server'.mysqli_errno($link));
    mysqli_close($link);
}
//else echo 'Connect to Server successfully!'.<br/>';

$databasename='id8366440_tinnhanbinhchon';
$test=mysqli_select_db($link,$databasename);
if(!$test)
{
    die('Could not connect to Database!'.mysqli_errno($link));
}
//else echo 'Connect to Database successfully!'.<br/>';

mysqli_set_charset($link,'utf8');
?>

```

Sau đó liên kết với database.

b. Viết Webservice để nhận thông tin đẩy lên từ ứng dụng sau đó tự động thêm vào database

```

<?php
require 'CSDL.php';

$jsonthongtin = $_POST['jsontt'];
$thongtin = json_decode($jsonthongtin, true);
foreach ($thongtin as $value) {
    $MSCT = $value['masocuochi'];
    $MSCS = $value['masocasi'];
    $Songuoi = $value['songuoibinhchon'];
    $phoneNumber = $value['sodienthoai'];
}

/*$MSCT = "TTH1";
$MSCS = 700;
$Songuoi = 300; */

$query = "INSERT INTO tin_nhan_binh_chon VALUES(null, '$MSCT', '$MSCS', '$Songuoi', $phoneNumber)";

if(mysqli_query($link, $query))
{
    //thanh cong
    echo "1";
}
else{
    //that bai
    echo "0";
}
?>

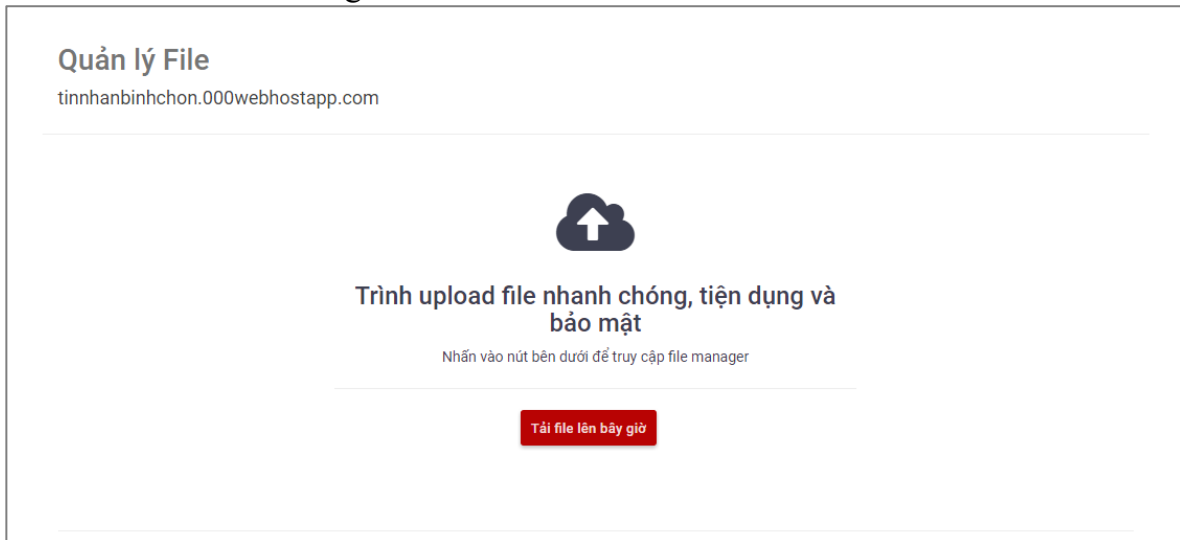
```

Khai báo các biến theo ứng dụng và database đã tạo.

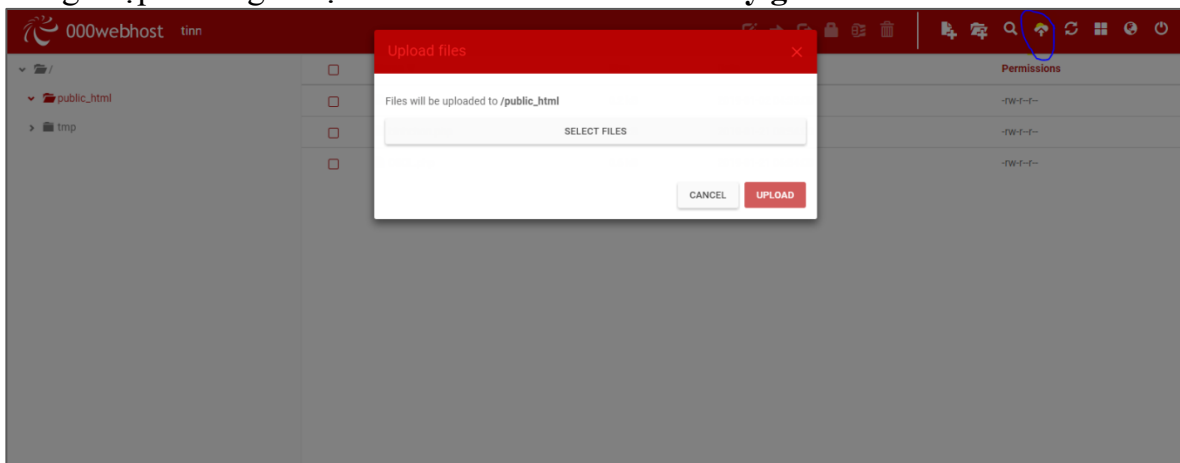
QC20-B19

Dùng lệnh POST để đẩy thông tin từ ứng dụng lên webservice.

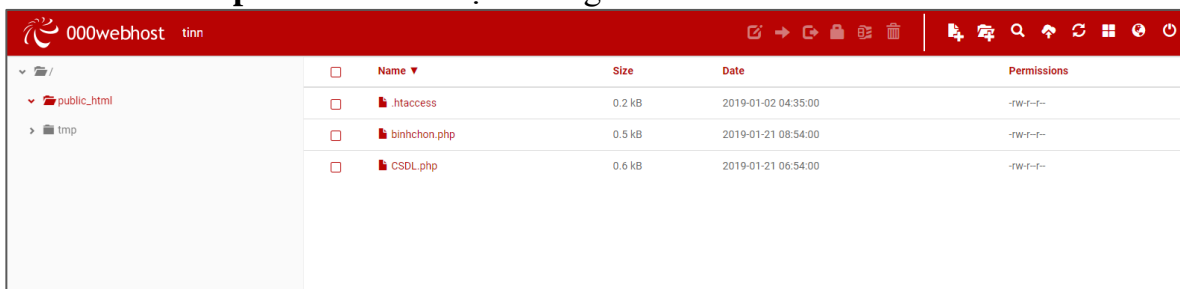
c. Tải file PHP lên hosting



Đăng nhập hosting đã tạo sẵn. Ấn vào **Tải file lên bây giờ**



Sau đó ấn vào **Upload files** để chọn đường dẫn tải 2 file PHP vừa khai báo.



Đây là kết quả nhận được.

3.5. Bộ nghe sự kiện tin nhắn

3.5.1. Khái niệm

Bộ nghe sự kiện tin nhắn (Broadcast Receiver) là một trong 4 thành phần quan trọng trong Android, với mục đích là lắng nghe các sự kiện, trạng thái của hệ thống phát ra thông qua Intent nhờ đó mà các lập trình viên có thể xử lý được các sự kiện hệ thống ở bên trong ứng dụng của mình.

Broadcast Receiver có thể hoạt động được cả khi ứng dụng bị tắt đi, nghĩa là ở background chính vì vậy nó thường được sử dụng với service.

3.5.2. Chức năng

Broadcast Receiver cho phép ứng dụng đăng ký nhận tin nhắn từ hệ điều hành Android.

Khi có tin nhắn gửi đến. Hệ điều hành Android sẽ nhận được đầu tiên. Muốn tin nhắn được gửi đến ứng dụng để xử lý ta phải đăng ký bộ nghe tin nhắn với hệ điều hành. Khi đó, nếu có tin nhắn gửi tới, hệ điều hành sẽ tự động gửi tin nhắn đến ứng dụng của người dùng để xử lý.

Đặc biệt có thể nhận tin nhắn đến ứng dụng ngay cả khi ứng dụng không hoạt động.

3.5.3. Kỹ thuật đăng ký bộ nghe sự kiện tin nhắn

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.ddq.duyquang.xulytinhan">
    <uses-permission android:name="android.permission.SEND_SMS" />
    <uses-permission android:name="android.permission.RECEIVE_SMS" />
    <uses-permission android:name="android.permission.WRITE_SMS" />
    <uses-permission android:name="android.permission.READ_SMS" />
    <uses-permission android:name="android.permission.INTERNET" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="xu ly tin nhan"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <receiver android:name="MyBroadCast" >
            <intent-filter>
                <action android:name="android.provider.Telephony.SMS_RECEIVED" />
            </intent-filter>
        </receiver>
    </application>
</manifest>
```

Ta khai báo trong Manifest [4].

- Dòng lệnh 5 → 8 là đăng ký permission cho việc Nhận tin nhắn, gửi tin nhắn, ghi tin nhắn, đọc tin nhắn.
- Dòng lệnh 24 → 27 là đăng ký Broadcast Receiver trong Manifest, giúp chương trình hoạt động như service chạy nền.

3.6. Kỹ thuật xử lý tin nhắn

Trong khuôn khổ đề án này. Một tin nhắn gửi từ điện thoại khách tới điện thoại chủ được quy ước làm 2 loại : Tin nhắn thường và tin nhắn bình chọn.

- Tin nhắn thường : Là các tin nhắn ở dạng văn bản không có cú pháp đặc biệt. Đối với tin nhắn thường. Chương trình sẽ tự động nhận dạng và không có bất kỳ phản hồi nào cả. Nếu phản hồi sẽ gây tốn kém kinh phí phía ban tổ chức.
- Tin nhắn bình chọn : Là các tin nhắn văn bản có nội dung được soạn thảo theo cú pháp đặc biệt. Nếu ứng dụng nhận diện được là tin nhắn bình chọn. Ứng dụng sẽ tách và lấy thông tin gửi lên Webservice.

CHƯƠNG 4 : CHƯƠNG TRÌNH THỬ NGHIỆM

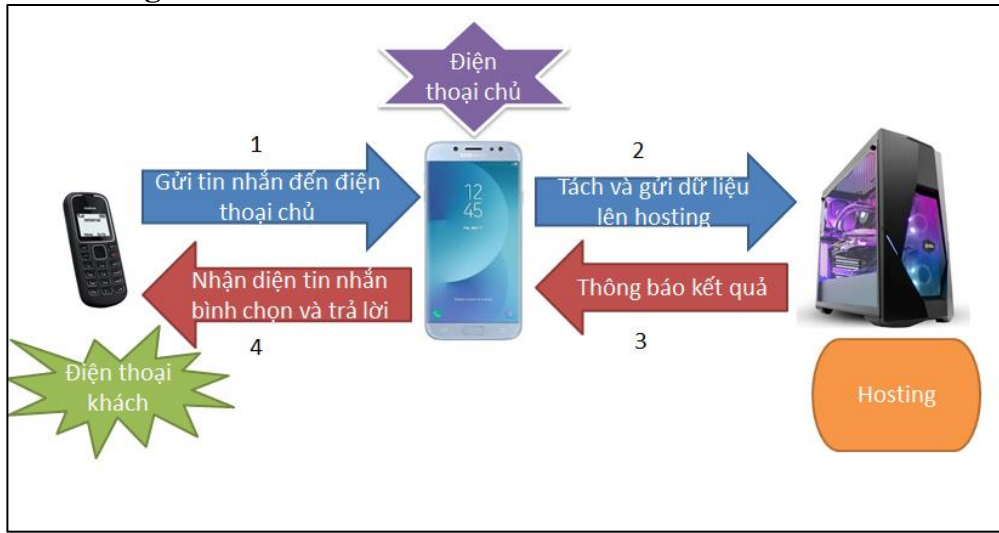
4.1. Phát biểu bài toán

Tại một cuộc thi Tiếng Hát Hay trên sóng truyền hình. Khi MC giới thiệu Ca sĩ “Nguyễn Văn A” thì một bài hát nào đó. Sau khi Ca Sĩ A thi xong thì chúng ta thường thấy MC nói rằng : “ Đừng quên bình chọn cho Ca Sĩ Nguyễn Văn A với mã số ...

Nếu chương trình tổ chức bình chọn theo hình thức phát phiếu. Thì chỉ những người có mặt dự chương trình mới có thể bình chọn. Như thế thì quy mô chương trình sẽ rất nhỏ. Tính khách quan cũng chưa cao và quan trọng nhất là sẽ không có kinh phí.

Bài toán đưa ra ở đây : Làm thế nào để cho tất cả mọi người xem chương trình ở khắp nơi đều có thể tham gia bình chọn. Từ đó xây dựng chương trình có thể xử lý được vấn đề đó...

4.2. Mô hình chương trình



Hình 4.2.1. Mô hình chương trình.

Hoạt động của chương trình :

1. Người bình chọn bắt đầu nhấn tin bình chọn vào Điện thoại chủ (Đầu số quy ước của chương trình) theo cú pháp được MC hướng dẫn.
2. Điện thoại chủ xác nhận tin nhắn có phải là tin nhắn bình chọn hay không ? Nếu có sẽ tự động trả lời lại : Đã nhận đc tin nhắn bình chọn.
3. Sau khi nhận đc tin nhắn bình chọn. Chương trình được cài đặt ở điện thoại chủ sẽ tách các thông tin của tin nhắn bình chọn ra làm 3 trường : THH(Mã cuộc thi), MSCS, Số người. Sau đó gửi thông tin lên Hosting.
4. Webservice nếu nhận được thông tin từ chương trình ở điện thoại chủ gửi lên sẽ tự động gửi thông tin lên Hosting chứa database tạo sẵn. Sau đó sẽ trả lời lại chương trình cài ở trên điện thoại chủ là đã thêm thông tin vào database được hay thất bại.

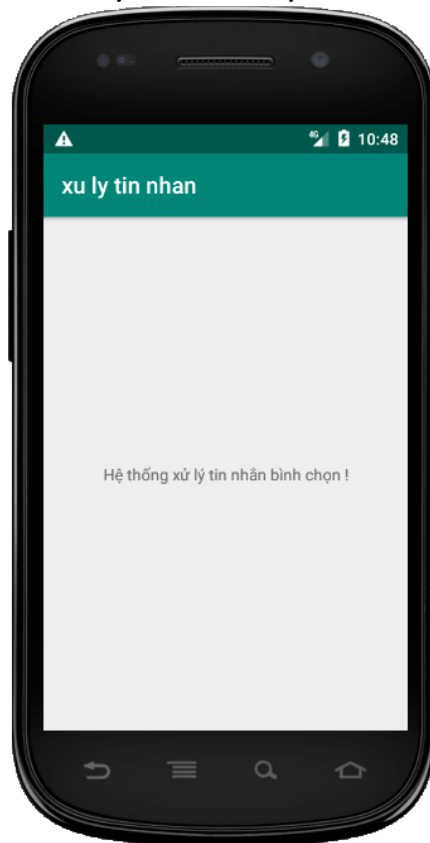
4.3. Thiết kế dữ liệu

STT	Tên trường	Kiểu trường	Độ rộng	Ghi chú
1	id	Auto Number	...	Định danh
2	MSCT	Varchar	10	Mã số cuộc thi
3	MSCS	Int	11	Mã số ca sĩ
4	Songnoi	Int	11	Số người dự đoán cùng kết quả
5	phoneNumber	Varchar	20	Số điện thoại gửi tin nhắn bình chọn

4.4. Một số giao diện chương trình

4.4.1. Giao diện chính của chương trình

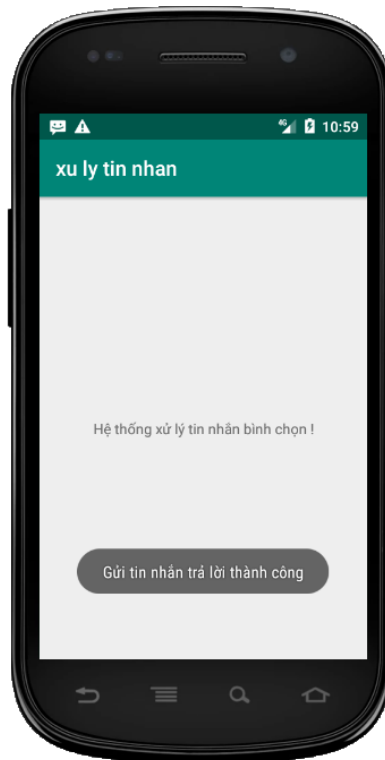
- Khi ứng dụng được khởi động. Giao diện sẽ hiển thị như hình dưới.



Hình 4.4.1. Giao diện chính của chương trình.

4.4.2. Giao diện trả lời tin nhắn

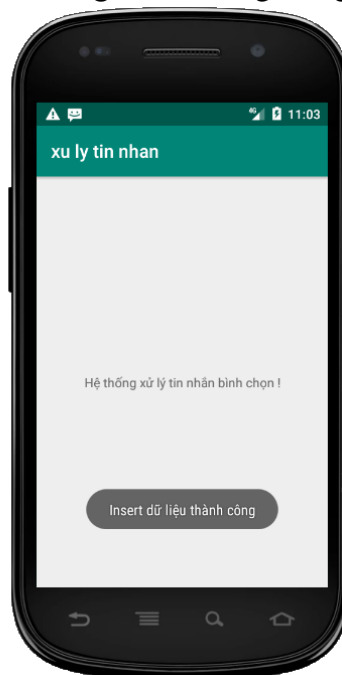
- Khi nhận được tin nhắn đúng theo **CÚ PHÁP** đã quy ước. Chương trình sẽ tự động trả lời và hiển thị ra màn hình : **Gửi tin nhắn trả lời thành công**



Hình 4.4.2.1. Giao diện trả lời tin nhắn tự động.

4.4.3. Giao diện thông báo thêm dữ liệu thành công

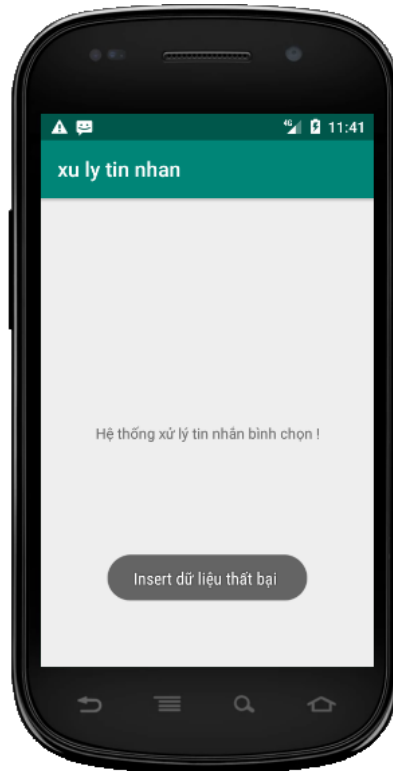
- Khi webservice nhận được dữ liệu gửi lên từ ứng dụng. Sau đó webservice sẽ chèn vào database. Nếu thành công sẽ gửi lại thông báo về ứng dụng : **Insert dữ liệu thành công**.



Hình 4.4.3.1. Giao diện của chương trình khi thêm dữ liệu thành công.

4.4.4. Giao diện thông báo thêm dữ liệu thất bại

- Khi webservice nhận được dữ liệu gửi lên từ ứng dụng. Sau đó webservice sẽ chèn vào database. Nếu thất bại sẽ gửi lại thông báo về ứng dụng : **Insert dữ liệu thất bại.**



Hình 4.4.4.1. Giao diện thông báo thêm dữ liệu thất bại.

KẾT LUẬN

Sau một thời gian tìm hiểu đề tài : “Xây dựng ứng dụng Android xử lý tin nhắn bình chọn” em đã thực hiện được nội dung đề tài theo yêu cầu đặt ra. Với mục đích là tìm hiểu về hệ điều hành Android, môi trường lập trình Android Studio và tìm hiểu các kỹ thuật để xây dựng một ứng dụng hoàn chỉnh trên thiết bị thật. Thông qua việc tìm hiểu và nắm bắt lý thuyết về hệ điều hành Android, bước đầu đã xây dựng thành công chương trình thực nghiệm với các chức năng cơ bản.

Nội dung của đồ án tốt nghiệp của em đã đạt được một số kết quả sau :

- Tạo được cơ sở dữ liệu web trên Hosting.
- Xây dựng được Webservice để đẩy dữ liệu từ ứng dụng lên cơ sở dữ liệu web.
- Bước đầu xây dựng thành công ứng dụng Android xử lý tin nhắn với các chức năng cơ bản.

Tuy nhiên, do thời gian và khả năng có hạn nên em chưa đi sâu tìm hiểu về ứng dụng, về giao diện ứng dụng vẫn còn sơ sài và không được trau chuốt.

Trong thời gian tới em sẽ phát triển ứng dụng này thêm nữa. Làm giao diện thân thiện với người sử dụng hơn. Tạo ra một ứng dụng hoàn thiện giúp các đơn vị sử dụng có thể mở rộng quy mô của các chương trình có tính chất bình chọn. Góp phần làm giảm bớt chi phí của đơn vị tổ chức,...

TÀI LIỆU THAM KHẢO

- [1]. <http://hd24h.edu.vn/tin-hoc/tong-quan-ve-android>
- [2]. <https://fptshop.com.vn/tin-tuc/danh-gia/lich-su-phat-trien-cua-android-he-dieu-hanh-di-dong-pho-bien-nhat-the-gioi-55019>
- [3]. <https://vinahost.vn/host-la-gi.html>
- [4]. <https://duythanhcse.wordpress.com/2014/02/26/bai-47-xu-ly-tin-nhan-sms-bang-broadcastreceiver/>