

**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 : Cao Tuấn Nghĩa

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 XEM VIDEO
TRỰC TUYẾN**

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

**Sinh viên : Cao Tuấn Nghĩa
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: Cao Tuấn Nghĩa

Mã SV: 1412101091

Lớp: CT1801

Ngành: Công nghệ thông tin

Tên đề tài: Xây dựng ứng dụng Android xem video trực tuyến

NHIỆM VỤ ĐỀ TÀI

1. Nội dung và các yêu cầu cần giải quyết trong nhiệm vụ đề tài tốt nghiệp

- a. Nội dung:

- Tìm hiểu hệ điều hành android.
- Tìm hiểu môi trường lập trình Android Studio.
- Tìm hiểu kỹ thuật phát video trên internet thông qua đường dẫn.
- Tìm hiểu hosting, xây dựng kho lưu trữ video trực tuyến trên hosting.
- Xây dựng chương trình xem video trực tuyến chạy trên điện thoại Android

- b. Các yêu cầu cần giải quyết

- Sử dụng công cụ Android Studio để viết chương trình chạy trên điện thoại android.
- Tạo hosting miễn phí.
- Xây dựng kho lưu trữ video trên hosting.
- Xây dựng được chương trình có một số chức năng cơ bản xem video trực tuyến, đóng gói chương trình thành file *.apk cho phép cài đặt và chạy trên điện thoại android thật..
- 2. Các số liệu cần thiết để thiết kế, tính toán
- Sử dụng số liệu thực tế trên Internet.
- 3. Địa điểm thực tập

Trường Đại học Dân lập Hải Phòng.

CÁN BỘ HƯỚNG DẪN ĐỀ TÀI TỐT NGHIỆP

Người hướng dẫn thứ nhất:

Họ và tên : Phùng Anh Tuấn.

Học hàm, học vị : Thạc sỹ.

Cơ quan công tác : Trường Đại học Dân lập Hải Phòng.

Nội dung hướng dẫn:

- Tìm hiểu hệ điều hành android.
- Tìm hiểu môi trường lập trình Android Studio.
- Tìm hiểu kỹ thuật phát video trên internet thông qua đường dẫn.
- Tìm hiểu hosting, xây dựng kho lưu trữ video trực tuyến trên hosting.
- Xây dựng chương trình xem video trực tuyến chạy trên điện thoại Android.

Người hướng dẫn thứ hai:

Họ và tên: Học hàm, học vị

Cơ quan công tác: Nội dung hướng dẫn:

.....
.....

Đề tài tốt nghiệp được giao ngày 15 tháng 10 năm 2018

Yêu cầu phải hoàn thành trước ngày 07 tháng 01 năm 2019

Đã nhận nhiệm vụ: Đ.T.T.N

Sinh viên

Đã nhận nhiệm vụ: Đ.T.T.N

Cán bộ hướng dẫn Đ.T.T.N

Hải Phòng, ngàytháng.....năm 2019

Hiệu trưởng

GS.TS.NGƯT Trần Hữu Nghị

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN TỐT NGHIỆP

Họ và tên giảng viên: Phùng Anh Tuấn.

Đơn vị công tác: Trường Đại học Dân lập Hải Phòng.

Họ và tên sinh viên: Cao Tuấn Nghĩa.

Ngành: Công Nghệ Thông Tin.

Nội dung hướng dẫn:

Tìm hiểu hệ điều hành android.

- Tìm hiểu môi trường lập trình Android Studio.
- Tìm hiểu kỹ thuật phát video trên internet thông qua đường dẫn.
- Tìm hiểu hosting, xây dựng kho lưu trữ video trực tuyến trên hosting.
- Xây dựng chương trình xem video trực tuyến chạy trên điện thoại Android.

1. Tinh thần thái độ của sinh viên trong quá trình làm đề tài tốt nghiệp

- Có nghiên cứu tài liệu phục vụ cho nội dung đề án.
- Có khả năng làm việc độc lập.
- Chưa nghiêm túc tuân thủ lịch làm việc với cán bộ hướng dẫn.

2. Đánh giá chất lượng của đề án/khóa luận (so với nội dung yêu cầu đã đề ra trong nhiệm vụ Đ.T. T.N trên các mặt lý luận, thực tiễn, tính toán số liệu...)

- Đã thực hiện được các nội dung đề ra trong đề cương.
- Nội dung đề án có tính thực tế.
- Cần trình bày nội dung đề án logic hơn.

3. Ý kiến của giảng viên hướng dẫn tốt nghiệp

Đạt Không đạt

Điểm:.....

Hải Phòng, ngày tháng 01 năm 2019

Giảng viên hướng dẫn

(Ký và ghi rõ họ tên)

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập - Tự do - Hạnh phúc

PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN CHẤM PHẢN BIỆN

Họ và tên giảng viên:

Đơn vị công tác:

Họ và tên sinh viên: Ngành:

Đề tài tốt nghiệp:

.....

1. Phần nhận xét của giảng viên chấm phản biện

.....
.....
.....
.....
.....
.....
.....
.....
.....

2. Những mặt còn hạn chế

.....
.....
.....
.....
.....
.....
.....
.....

3. Ý kiến của giảng viên chấm phản biện

Được bảo vệ Không được bảo vệ Điểm:.....
vê

Hải Phòng, ngày tháng 01 năm 2019

Giảng viên chấm phản biện

(Ký và ghi rõ họ tên)

MỤC LỤC

LỜI CẢM ƠN	11
CHƯƠNG 1: GIỚI THIỆU HỆ ĐIỀU HÀNH ANDROID	12
1.1. Lịch sử ra đời.	12
1.2. Giao diện Android.	13
1.3. Nhân Linux.	14
1.4. Quản lý bộ nhớ.....	16
1.5. Lịch cập nhật.....	16
1.6. Cộng đồng mã nguồn mở.	17
1.7. Bảo mật và tính riêng tư.	18
1.8. Các phiên bản của Android.	19
CHƯƠNG 2: MÔI TRƯỜNG ANDROID STUDIO	23
2.1. Giới thiệu Android Studio.	23
2.2. Cài đặt môi trường Android Studio.....	23
2.2.1. Cấu hình tối thiểu cài đặt Android Studio.	23
2.2.2. Các bước cài đặt.	23
2.3. Cấu trúc dự án Android trong ANDROID STUDIO.....	30
2.3.1. Tạo mới một project.	31
2.3.2. Màn hình làm việc của Android Studio.	34
2.4. Thành phần giao diện trong Android Studio.....	36
2.4.1. Thành phần hiển thị.	36
2.4.2. Một số nhóm hiển thị cơ bản.....	37
2.5. Bắt và xử lý sự kiện trên giao diện.....	38
2.6. Vòng đời ứng dụng Android.....	39

2.7. Thành phần Intent	41
CHƯƠNG 3: CÁC KỸ THUẬT XÂY DỰNG ỨNG DỤNG XEM VIDEO TRỰC TUYÊN.....	43
3.1. Dịch vụ web và cơ sở dữ liệu trên internet.	43
3.2. Một số kỹ thuật lập trình cơ sở dữ liệu trên internet.....	50
3.2.1. Xây dựng Webservice.	50
3.2.2. Kỹ thuật kiểm tra cấp quyền kết nối internet thiết bị di động.	56
3.2.3. Kỹ thuật lấy dữ liệu từ CSDL trên internet về thiết bị di động.	57
3.2.4. Bắt và xử lý sự kiện click.	64
CHƯƠNG 4: CHƯƠNG TRÌNH THỰC NGHIỆM	75
4.1. Bài toán.	75
4.2. Mô hình chương trình.	75
4.3. Các bước xây dựng ứng dụng Android xem video trực tuyến.	76
4.4. Giao diện chương trình.	77
KẾT LUẬN.....	80
TÀI LIỆU THAM KHẢO.....	81

LỜI CẢM ƠN

Lời đầu tiên, em xin gửi tới các thầy cô khoa Công nghệ - Thông tin Trường Đại Học Dân Lập Hải Phòng lời chào trân trọng, lời chúc sức khỏe và lời cảm ơn sâu sắc. Thầy cô đã trang bị cho em những kiến thức cơ bản nhất để em có thể hoàn thành tốt đồ án: “Xây dựng ứng dụng android xem video trực tuyến”.

Đặc biệt em xin gửi lời cảm ơn chân thành nhất tới thầy ThS. Phùng Anh Tuấn đã quan tâm giúp đỡ, hướng dẫn em hoàn thành tốt đồ án này trong suốt thời gian qua.

Với điều kiện thời gian cũng như kinh nghiệm còn hạn chế của một sinh viên, đồ án này không thể tránh khỏi những thiếu sót. Em rất mong nhận được sự chỉ bảo, đóng góp ý kiến của thầy cô để em có thể bổ sung, phục vụ tốt cho công việc thực tế sau này.

Xin chân thành cảm ơn!

Sinh viên

Cao Tuấn Nghĩa

CHƯƠNG 1: GIỚI THIỆU HỆ ĐIỀU HÀNH ANDROID

1.1. Lịch sử ra đời.

Vào tháng 10/2003, trước khi thuật ngữ “điện thoại thông minh” được hầu hết công chúng sử dụng và vài năm trước khi Apple công bố iPhone đầu tiên và hệ điều hành iOS, công ty Android Inc được thành lập ở Palo Alto, California. Bốn người sáng lập là Rich Miner, Nick Sears, Chris White và Andy Rubin. Vào thời điểm thành lập, ông Rubin nói rằng Android Inc sẽ phát triển “thiết bị di động thông minh hơn về vị trí và sở thích của chủ sở hữu” [1].

Theo PC World, Rubin sau này đã tiết lộ trong một bài phát biểu vào năm 2013 tại Tokyo rằng hệ điều hành Android đã được ra mắt để cải thiện hệ điều hành của máy ảnh số. Rõ ràng, nhóm nghiên cứu tại Android đã không nghĩ ngay từ đầu về việc tạo ra một hệ điều hành có thể phục vụ như phần cốt lõi của một hệ thống máy tính di động hoàn chỉnh.

Nhưng sau đó, thị trường máy ảnh kỹ thuật số giảm sút, Android Inc đã quyết định chuyển sang hệ điều hành trên điện thoại di động. Như Rubin đã nói vào năm 2013, “Cùng một nền tảng, cùng hệ điều hành chúng tôi xây dựng cho máy ảnh, nó đã trở thành Android cho điện thoại di động.”

Năm 2005, chương lớn tiếp theo trong lịch sử của Android được thực hiện khi Google mua lại công ty gốc. Ông Andy Rubin và các thành viên sáng lập khác vẫn tiếp tục phát triển hệ điều hành dưới quyền chủ sở hữu mới của họ. Quyết định này được đưa ra để sử dụng Linux làm nền tảng cho hệ điều hành Android và điều đó cũng có nghĩa là Android sẽ được cung cấp miễn phí cho các nhà sản xuất điện thoại di động của bên thứ ba. Google và nhóm Android cảm thấy công ty có thể kiếm tiền với các dịch vụ khác sử dụng hệ điều hành, bao gồm cả ứng dụng.

Năm 2007, Apple công bố chiếc iPhone đầu tiên với thế giới và đã tự tay thiết lập một kỷ nguyên mới dành cho điện toán di động. Cùng thời điểm, Google vẫn đang làm việc hết công suất với Android và đảm bảo mọi thông tin đều tuyệt mật. Đến tháng 11 cùng năm, công ty bắt đầu từ từ hé lộ kế hoạch cạnh tranh “thẳng mặt” với Apple và

các nền tảng di động khác. Google đã tận dụng sự hình thành của một thứ có tên là Liên minh Thiết bị cầm tay Mở (Open Handset Alliance), vốn dĩ có sự góp mặt của nhiều nhà sản xuất điện thoại như HTC, Motorola, nhà sản xuất chip như Qualcomm và Texas Instruments, cũng như nhà mạng lớn như T-Mobile.

Hệ điều hành Android được chính thức ra mắt từ năm 2007 cùng với tuyên bố thành lập Liên minh thiết bị cầm tay mở. Tháng 9/2008, smartphone chạy Android đầu tiên đã trình làng - T-Mobile G1, có tên khác là HTC Dream. Vào thời gian đầu, rất nhiều tính năng cơ bản bị thiếu sót như: bàn phím ảo, cảm ứng đa điểm và tính năng mua ứng dụng vẫn chưa xuất hiện. Tuy nhiên, một số tính năng cũng như giao diện đặc sản của hệ điều hành này đã khởi nguồn từ chiếc G1 và trở thành những yếu tố không thể thiếu trên Android sau này.

1.2. Giao diện 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 [2]. 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 nhỡ 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.

1.3. Nhân Linux.

Android có một hạt nhân dựa trên nhân Linux phiên bản 2.6, kể từ Android 4.0 Ice Cream Sandwich (bánh ngọt kẹp kem) trở về sau, là phiên bản 3.x, với middleware, thư viện và API viết bằng C, còn phần mềm ứng dụng chạy trên một nền tảng ứng dụng gồm các thư viện tương thích với Java dựa trên Apache Harmony. Android sử dụng máy ảo Dalvik với một trình biên dịch động để chạy 'mã dex' (Dalvik Executable) của Dalvik, thường được biên dịch sang Java bytecode. Nền tảng phần cứng chính của Android là kiến trúc ARM. Người ta cũng hỗ trợ x86 thông qua dự án Android x86, và Google TV cũng sử dụng một phiên bản x86 đặc biệt của Android.

Nhân Linux dùng cho Android đã được Google thực hiện nhiều thay đổi về kiến trúc so với nhân Linux gốc. Android không có sẵn X Window System cũng không hỗ trợ các thư viện GNU chuẩn, nên việc chuyển các ứng dụng hoặc thư viện Linux có sẵn sang Android rất khó khăn. Các ứng dụng C đơn giản và SDL cũng được hỗ trợ bằng cách chèn những đoạn shim Java và sử dụng tương tự JNI, như khi người ta chuyển Jagged Alliance 2 sang Android.

Một số tính năng cũng được Google đóng góp ngược vào nhân Linux, đáng chú ý là tính năng quản lý nguồn điện có tên wakelock, nhưng bị những người lập trình chính cho nhân từ chối vì họ cảm thấy Google không có định sẽ tiếp tục bảo trì đoạn mã do họ viết. Google thông báo vào tháng 4 năm 2010 rằng họ sẽ thuê hai nhân viên để làm việc với cộng đồng nhân Linux, nhưng Greg Kroah-Hartman, người bảo trì nhân Linux hiện tại của nhánh ổn định, đã nói vào tháng 12 năm 2010 rằng ông ta lo ngại rằng Google không còn muốn đưa những thay đổi của mình vào Linux dòng chính nữa. Một số lập trình viên Android của Google tỏ ý rằng "nhóm Android thấy chán với quy trình đó," vì nhóm họ không có nhiều người và có nhiều việc khẩn cấp cần làm với Android hơn.

Vào tháng 12 năm 2011, nhắm tới việc đưa một số driver, bản vá và tính năng của Android ngược vào nhân Linux, bắt đầu từ Linux 3.3. Linux cũng đưa tính năng autosleep (tự nghỉ hoạt động) và wakelocks vào nhân 3.5, sau nhiều nỗ lực phối trộn trước đó. Tương tác thì vẫn vậy nhưng bản hiện thực trên Linux dòng chính cho phép hai chế độ nghỉ: bộ nhớ (dạng nghỉ truyền thống mà Android sử dụng), và đĩa (là ngủ đông trên máy tính để bàn). Việc trộn sẽ hoàn tất kể từ nhân 3.8, Google đã công khai kho mã nguồn trong đó có những đoạn thử nghiệm đưa Android về lại nhân 3.8.

Việc Android có được xem là một bản phân phối Linux hay không vẫn còn là vấn đề gây tranh cãi, tuy được Linux Foundation và Chris DiBona, trưởng nhóm mã nguồn mở Google, ủng hộ. Một số khác, như linux-magazine.com thì không đồng ý, do Android không hỗ trợ nhiều công cụ GNU, trong đó có glibc.

1.4. Quản lý bộ nhớ.

Vì các thiết bị Android chủ yếu chạy bằng pin, nên Android được thiết kế để quản lý bộ nhớ (RAM) để giảm tối đa tiêu thụ điện năng, trái với hệ điều hành máy tính để bàn luôn cho rằng máy tính sẽ có nguồn điện không giới hạn. Khi một ứng dụng Android không còn được sử dụng, hệ thống sẽ tự động ngưng nó trong bộ nhớ - trong khi ứng dụng về mặt kỹ thuật vẫn "mở", những ứng dụng này sẽ không tiêu thụ bất cứ tài nguyên nào (như năng lượng pin hay năng lượng xử lý) và nằm đó cho đến khi nó được cần đến. Cách làm như vậy có lợi kép là vừa làm tăng khả năng phản hồi nói chung của thiết bị Android, vì ứng dụng không nhất phải đóng rồi mở lại từ đầu, vừa đảm bảo các ứng dụng nền không làm tiêu hao năng lượng một cách không cần thiết.

Android quản lý các ứng dụng trong bộ nhớ một cách tự động: khi bộ nhớ thấp, hệ thống sẽ bắt đầu diệt ứng dụng và tiến trình không hoạt động được một thời gian, sắp theo thời điểm cuối mà chúng được sử dụng (tức là cũ nhất sẽ bị tắt trước). Tiến trình này được thiết kế ẩn đi với người dùng, để người dùng không cần phải quản lý bộ nhớ hoặc tự tay tắt các ứng dụng. Tuy nhiên, sự che giấu này của hệ thống quản lý bộ nhớ Android đã dẫn đến sự thịnh hành của các ứng dụng tắt chương trình của bên thứ ba trên cửa hàng Google Play; những ứng dụng kiểu như vậy được cho là có hại nhiều hơn có lợi.

1.5. Lịch cập nhật.

Google đưa ra các bản cập nhật lớn cho Android theo chu kỳ từ 6 đến 9 tháng, mà phần lớn thiết bị đều có thể nhận được qua sóng không dây. Bản cập nhật lớn mới nhất là Android 9.0 P.

So với các hệ điều hành cạnh tranh khác, như iOS, các bản cập nhật Android thường mất thời gian lâu hơn để đến với các thiết bị. Với những thiết bị không thuộc dòng Nexus và Pixel, các bản cập nhật thường đến sau vài tháng kể từ khi phiên bản được chính thức phát hành. Nguyên nhân của việc này một phần là do sự phong phú về phần cứng của các thiết bị Android, nên người ta phải mất thời gian điều chỉnh bản

cập nhật cho phù hợp, vì mã nguồn chính thức của Google chỉ chạy được trên những thiết bị Nexus chủ lực của họ. Chuyển Android sang những phần cứng cụ thể là một quy trình tốn thời gian và công sức của các nhà sản xuất thiết bị, những người luôn ưu tiên các thiết bị mới nhất và thường bỏ rơi các thiết bị cũ hơn. Do đó, những chiếc điện thoại thông minh thế hệ cũ thường không được cập nhật nếu nhà sản xuất quyết định rằng nó không đáng để bỏ thời gian, bất kể chiếc điện thoại đó có khả năng chạy bản cập nhật hay không. Vấn đề này còn trầm trọng hơn khi những nhà sản xuất điều chỉnh Android để đưa giao diện và ứng dụng của họ vào, những thứ này cũng sẽ phải làm lại cho mỗi bản cập nhật. Sự chậm trễ còn được đóng góp bởi nhà mạng, sau khi nhận được bản cập nhật từ nhà sản xuất, họ còn điều chỉnh thêm cho phù hợp với nhu cầu rồi thử nghiệm kỹ lưỡng trên hệ thống mạng của họ trước khi chuyển nó đến người dùng.

1.6. Cộng đồng mã nguồn mở.

Android có một cộng đồng các lập trình viên và những người đam mê rất năng động. Họ sử dụng mã nguồn Android để phát triển và phân phối những phiên bản chỉnh sửa của hệ điều hành. Các bản Android do cộng đồng phát triển thường đem những tính năng và cập nhật mới vào nhanh hơn các kênh chính thức của nhà sản xuất/nhà mạng, tuy không được kiểm thử kỹ lưỡng cũng như không có đảm bảo chất lượng; cung cấp sự hỗ trợ liên tục cho các thiết bị cũ không còn nhận được bản cập nhật chính thức; hoặc mang Android vào những thiết bị ban đầu chạy một hệ điều hành khác, như HP Touchpad. Các bản Android của cộng đồng thường được root sẵn và có những điều chỉnh không phù hợp với những người dùng không rành rẽ, như khả năng ép xung hoặc tăng/giảm áp bộ xử lý của thiết bị. CyanogenMod là firmware của cộng đồng được sử dụng phổ biến nhất, và hoạt động như một tổ chức của số đông khác.

Trước đây, nhà sản xuất thiết bị và nhà mạng tỏ ra thiếu thiện chí với việc phát triển firmware của bên thứ ba. Những nhà sản xuất còn thể hiện lo ngại rằng các thiết bị chạy phần mềm không chính thức sẽ hoạt động không tốt và dẫn đến tổn tiền hỗ

trợ. Hơn nữa, các firmware đã thay đổi như CyanogenMod đôi khi còn cung cấp những tính năng, như truyền tải mạng (tethering), mà người dùng bình thường phải trả tiền nhà mạng mới được sử dụng. Kết quả là nhiều thiết bị bắt đầu đặt ra hàng rào kỹ thuật như khóa bootloader hay hạn chế quyền truy cập root. Tuy nhiên, khi phần mềm do cộng đồng phát triển ngày càng trở nên phổ biến, và sau một thông cáo của Thư viện Quốc hội Hoa Kỳ cho phép "jailbreak" (vượt ngục) thiết bị di động, các nhà sản xuất và nhà mạng đã tỏ ra mềm mỏng hơn với các nhà phát triển thứ ba, thậm chí một số hãng như HTC, Motorola, Samsung và Sony, còn hỗ trợ và khuyến khích phát triển. Kết quả của việc này là dần dần nhu cầu tìm ra các hạn chế phần cứng để cài đặt được firmware không chính thức đã bớt đi do ngày càng nhiều thiết bị được phát hành với bootloader đã mở khóa sẵn hoặc có thể mở khóa, tương tự như điện thoại dòng Nexus, tuy rằng thông thường họ sẽ yêu cầu người dùng từ bỏ chế độ bảo hành nếu họ làm như vậy. Tuy nhiên, tuy được sự chấp thuận của nhà sản xuất, một số nhà mạng tại Mỹ vẫn bắt buộc điện thoại phải bị khóa.

1.7. Bảo mật và tính riêng tư.

Các ứng dụng Android chạy trong một "hộp cát", là một khu vực riêng rẽ với hệ thống và không được tiếp cận đến phần còn lại của tài nguyên hệ thống, trừ khi nó được người dùng trao quyền truy cập một cách công khai khi cài đặt. Trước khi cài đặt ứng dụng, Cửa hàng Play sẽ hiển thị tất cả các quyền mà ứng dụng đòi hỏi.

Hệ thống hộp cát và hỏi quyền làm giảm bớt ảnh hưởng của lỗi bảo mật hoặc lỗi chương trình có trong ứng dụng, nhưng sự bối rối của lập trình viên và tài liệu hướng dẫn còn hạn chế đã dẫn tới những ứng dụng hay đòi hỏi những quyền không cần thiết, do đó làm giảm đi hiệu quả của hệ thống này. Một số công ty bảo mật, như Lookout Mobile Security, AVG Technologies, và McAfee, đã phát hành những phần mềm diệt virus cho các thiết bị Android. Phần mềm này không có hiệu quả vì cơ chế hộp cát vẫn áp dụng vào các ứng dụng này, do vậy làm hạn chế khả năng quét sâu vào hệ thống để tìm nguy cơ.

Một nghiên cứu của một công ty bảo mật Trend Micro đã liệt kê tình trạng lạm dụng dịch vụ trả tiền là hình thức phần mềm ác ý phổ biến nhất trên Android, trong đó tin nhắn SMS sẽ bị gửi đi từ điện thoại bị nhiễm đến một số điện thoại trả tiền mà người dùng không hề hay biết. Loại phần mềm ác ý khác hiển thị quảng cáo không mong muốn và gây khó chịu trên thiết bị. Đe dọa bảo mật trên Android được cho là tăng rất nhanh theo cấp số mũ: tuy nhiên, các kỹ sư Google phản bác rằng hiểm họa từ phần mềm ác ý và virus đã bị thổi phồng bởi các công ty bảo mật đang lợi dụng sự sợ hãi để bán phần mềm ác ý thật sự cho người dùng.

Google hiện đang sử dụng bộ quét phần mềm ác ý Google Bouncer để theo dõi và quét các ứng dụng trên Cửa hàng Play. Nó sẽ đánh dấu các phần mềm bị nghi ngờ và cảnh báo người dùng về những vấn đề có thể xảy ra trước khi họ tải nó về máy. Android phiên bản 4.2 được phát hành vào năm 2012 cùng với các tính năng bảo mật được cải thiện, bao gồm một bộ quét phần mềm ác ý được cài sẵn trong hệ thống, hoạt động cùng với Google Play nhưng cũng có thể quét các ứng dụng được cài đặt từ nguồn thứ ba, và một hệ thống cảnh báo sẽ thông báo cho người dùng một ứng dụng cố gắng gửi tin nhắn vào số tính tiền, chặn tin nhắn đó lại và trừ khi người dùng công khai cho phép nó.

Điện thoại thông minh Android có khả năng báo cáo vị trí của điểm truy cập kết nối WI-FI, phát hiện ra việc di chuyển của người dùng điện thoại, để xây dựng những cơ sở dữ liệu có chức vị trí của hàng trăm triệu điểm truy cập. Những cơ sở này tạo nên một bản đồ điện tử để tìm vị trí điện thoại thông minh. Bản chất mã nguồn mở của Android cho phép những nhà thầu bảo mật lấy những thiết bị sẵn có rồi điều chỉnh để sử dụng ở mức độ bảo mật cao hơn.

1.8. Các phiên bản của Android.

Gần 10 năm kể từ khi tung ra phiên bản Android đầu tiên năm 2008, hệ điều hành di động đã vượt qua mốc 2 tỷ thiết bị [3].

Qua 10 năm phát triển, Google đã ghi được những bước tiến đáng kể, thể hiện ở con số cứ 10 điện thoại thông minh được bán trên toàn cầu thì có 9 chiếc là chạy hệ điều hành Android.

- **Phiên bản Android 1.0**

Android 1.0 (2008) lần đầu tiên phân phối ứng dụng thông qua Android Market với 35 ứng dụng ra mắt. Google Maps đã sử dụng GPS, Wi-Fi của điện thoại và Google Maps đã tích hợp sẵn trình duyệt Android.

- **Phiên bản Android 1.5 Cupcake.**

Android 1.5 Cupcake (2009) là bản cập nhật lớn đầu tiên của Android. Cupcake đã thêm các tiện ích cho màn hình chính, bàn phím ảo, quay video trong máy ảnh, chức năng sao chép và dán vào trình duyệt web.

- **Phiên bản Android 2.0 Eclair.**

Android 2.0 Eclair (2009) hỗ trợ tài khoản Google, cho phép người dùng tìm kiếm nội dung theo từ khóa trong các tin nhắn văn bản, đồng thời bổ sung hỗ trợ đa chạm và máy ảnh được cải tiến với đèn flash, zoom số.

- **Phiên bản Android 2.2 Froyo.**

Android 2.2 Froyo (2010) giới thiệu Flash Player 10.1, cho phép điện thoại phát video và phát trực tuyến âm thanh. Máy ảnh flash tương thích Bluetooth giúp người dùng có thể sử dụng điện thoại của mình làm điểm phát sóng Wi-Fi.

- **Phiên bản Android 2.3 Gingerbread.**

Android 2.3 Gingerbread (2011), Android được biết đến rộng rãi hơn nhờ tính năng giao tiếp trường gần (NFC), cho phép điện thoại thông minh kết nối với các thiết bị lân cận khác. Hệ điều hành này cũng cho phép gọi điện video bằng máy ảnh mặt trước và thêm trình quản lý tải xuống.

- **Phiên bản Android Honeycomb 3.0.**

Android Honeycomb 3.0 (2011) là bản cập nhật đầu tiên chỉ dành cho máy tính bảng, hỗ trợ đồ họa 3D, các tab trình duyệt cạnh nhau, trò chuyện video với Google Talk, chia sẻ kết nối Bluetooth và chế độ toàn màn hình trong thư viện ảnh.

- **Phiên bản Android Ice Cream Sandwich 4.0.**

Android Ice Cream Sandwich 4.0 (2011) sáp nhập các hệ điều hành điện thoại và máy tính bảng. ICS cũng thêm nhận dạng khuôn mặt để mở khóa điện thoại, phản hồi văn bản, tự động trả lời các cuộc gọi bị từ chối và hiệu ứng video trực tiếp trong camera.

- **Phiên bản Android Jelly Bean 4.1 – 4.3.1.**

Android Jelly Bean là tên được đặt cho 3 phiên bản chính của hệ điều hành Android mobile operating system developed by Google, trải qua các phiên bản từ 4.1 đến 4.3.1.

Android Jelly Bean 4.1 (2012) có hiệu suất nhanh hơn, mượt mà hơn nhờ "Project Butter", cho phép người dùng tương tác nhiều hơn, thông báo có thể mở rộng, trình duyệt Chrome mặc định... Hai phiên bản còn lại cũng có cùng tên Jelly Bean, được phát hành tương ứng vào tháng 10 năm 2012 và tháng 7 18 năm 2013, trong đó phiên bản 4.2 gồm tối ưu hóa, hỗ trợ nhiều người dùng cho máy tính bảng, widget cho màn hình khóa, tùy chỉnh nhanh, và screen saver, còn phiên bản 4.3 gồm các cải tiến và cập nhật nội bộ cho nền tảng Android.

- **Phiên bản Android 4.4 KitKat.**

Android 4.4 KitKat (2013) có thêm biểu tượng cảm xúc vào bàn phím Google, bộ nhớ nhỏ hơn để hỗ trợ điện thoại cấp thấp hơn, đồng thời cho phép người dùng in văn bản khi đang di chuyển với tính năng Google Cloud Print.

- **Phiên bản Android 5.0 Lollipop.**

Android 5.0 Lollipop (2014) có giao diện phẳng Material Design, thông báo xuất hiện trên màn hình khóa. Hệ điều hành cũng có chế độ ưu tiên, hỗ trợ đa người dùng, ghim màn hình...

- **Phiên bản Android 6.0 Marshmallow.**

Android 6.0 Marshmallow (2015) bắt đầu có chế độ Doze để tiết kiệm pin. Đồng thời bổ sung thêm hỗ trợ tích hợp cho đầu đọc vân tay, USB Type-C và chế độ 4K cho các ứng dụng.

- Phiên bản Android 7.0 Nougat.

Android 7.0 Nougat (2016) người dùng có thể xóa tất cả ứng dụng của mình bằng một lần nhấn, đồng thời điều chỉnh tông màu da của biểu tượng cảm xúc và hỗ trợ VR.

- Phiên bản Android 8.0 Oreo.

Android 8.0 Oreo (2017) hay Android Oreo bổ sung nhiều tác vụ trong ứng dụng ảnh. Ngoài ra còn mang đến trải nghiệm sao chép và dán tốt hơn, cải thiện bảo mật và quản lý pin tốt hơn.

- Phiên bản Android Pie 9.0.

Android Pie 9.0 (2018) tập trung nâng cấp phần mềm giúp điện thoại Android hoạt động nhanh hơn và tiết kiệm pin. Android Pie bổ sung công cụ trí tuệ nhân tạo (AI) vào các ứng dụng và lỗi tắt tiện lợi hơn cho người dùng.

CHƯƠNG 2: MÔI TRƯỜNG ANDROID STUDIO

2.1. Giới thiệu Android Studio.

Android Studio là môi trường phát triển tích hợp (IDE) chính thức dành cho phát triển nền tảng Android.[2]

Nó được ra mắt vào ngày 16 tháng 5 năm 2013 tại hội nghị Google I/O. Android Studio được phát hành miễn phí theo giấy phép Apache Licence 2.0.

Android Studio ở giai đoạn truy cập xem trước sớm bắt đầu từ phiên bản 0.1 vào tháng 5.2013, sau đó bước vào giai đoạn beta từ phiên bản 0.8 được phát hành vào tháng 6 năm 2014. Phiên bản ổn định đầu tiên được ra mắt vào tháng 12 năm 2014, bắt đầu từ phiên bản 1.0.

Dựa trên phần mềm IntelliJ IDEA của JetBrains, Android Studio được thiết kế đặc biệt để phát triển ứng dụng Android. Nó hỗ trợ các hệ điều hành Windows, Mac OS X, Linux và là IDE chính thức của Google để phát triển ứng dụng Android gốc để thay thế cho Android Development Tools (ADT) dựa trên Eclipse.

2.2. Cài đặt môi trường Android Studio.

2.2.1. Cấu hình tối thiểu cài đặt Android Studio.

- Microsoft® Windows® 10/8/7 (32 or 64-bit).
- 4 GB RAM. (Khuyến cáo là 8GB).
- Chip core I3 trở lên.
- 400 MB hard disk space + ít nhất 1GB cho Android SDK, emulator.
- Độ phân giải tối thiểu 1366 x 768.

2.2.2. Các bước cài đặt.

Bước 1: Cài đặt JAVA JDK.

- ❖ Tải và cài JDK (là bộ Java) theo đường dẫn

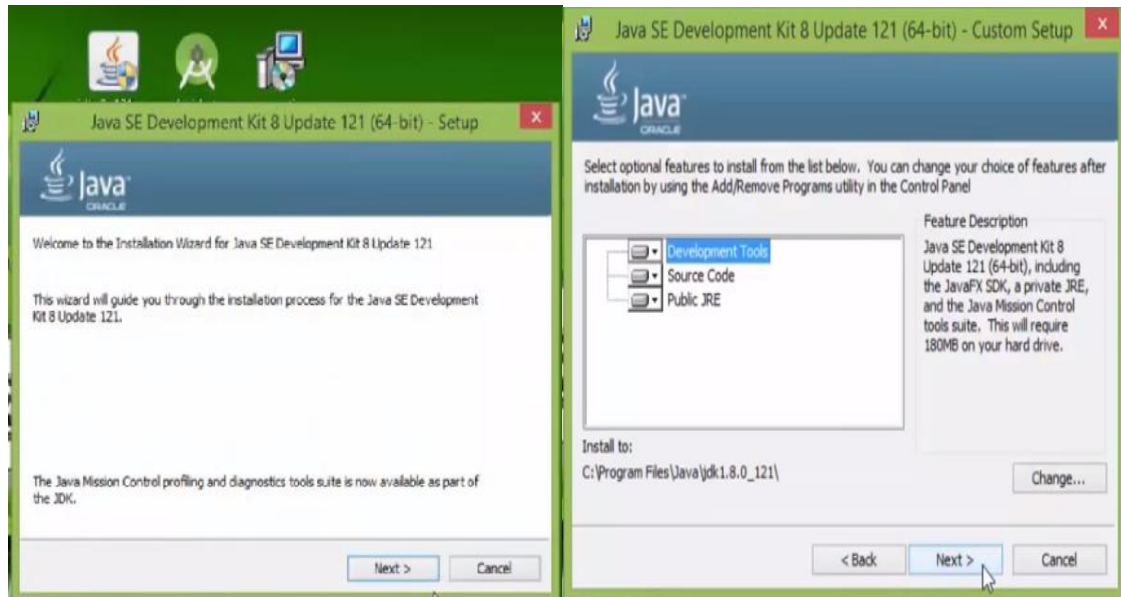
<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

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

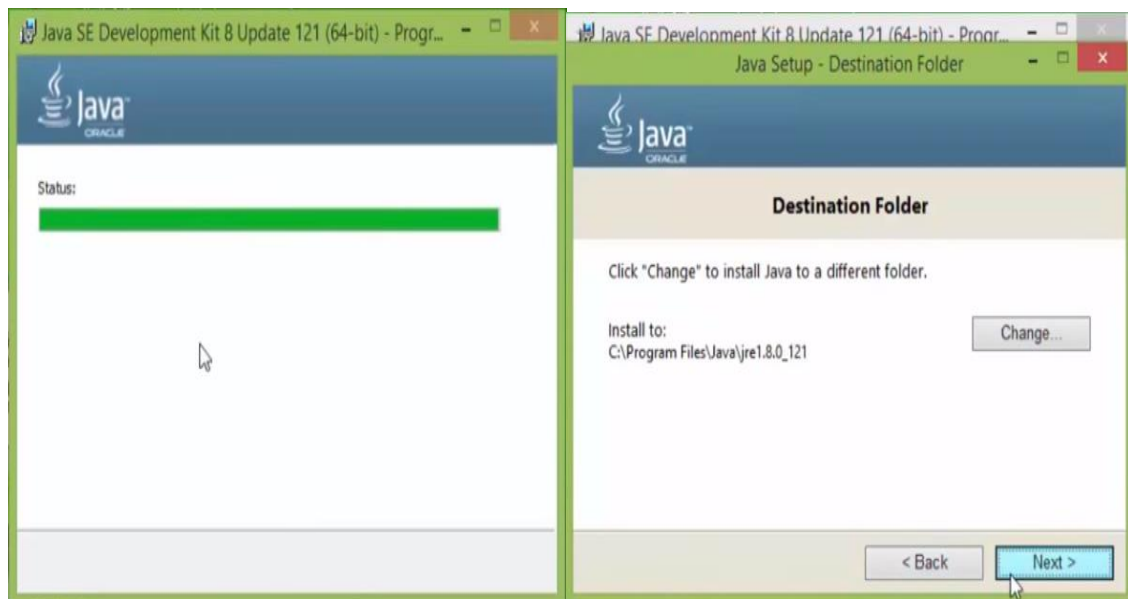
- ❖ Mở file cài đặt “jdk-*.exe” để tiến hành cài đặt.

❖ Cấu hình biến môi trường cho Java.

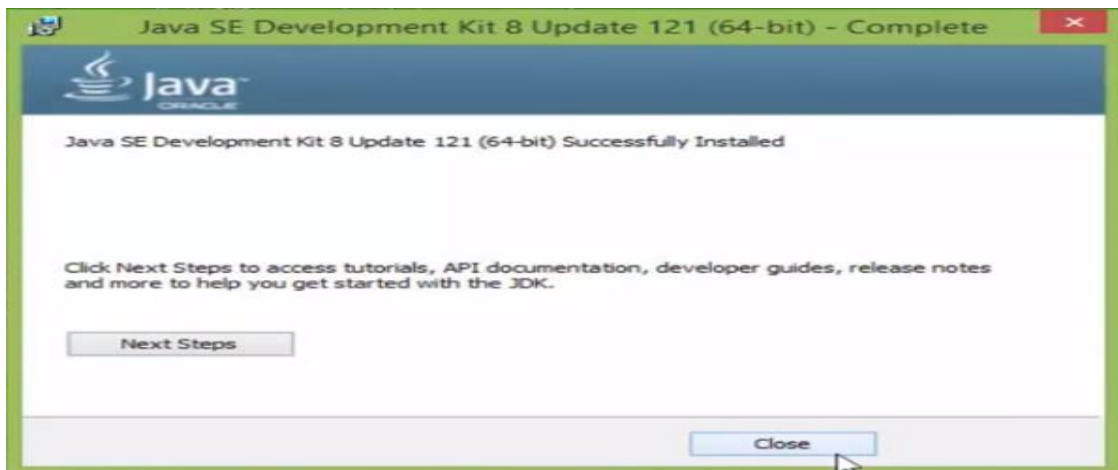
- Chọn next để tiếp tục cài đặt.



- Chờ hệ thống cài đặt và bấm next để tiếp tục.



- Chọn close để hoàn tất cài đặt.



Bước 2: Cài đặt Android Studio.

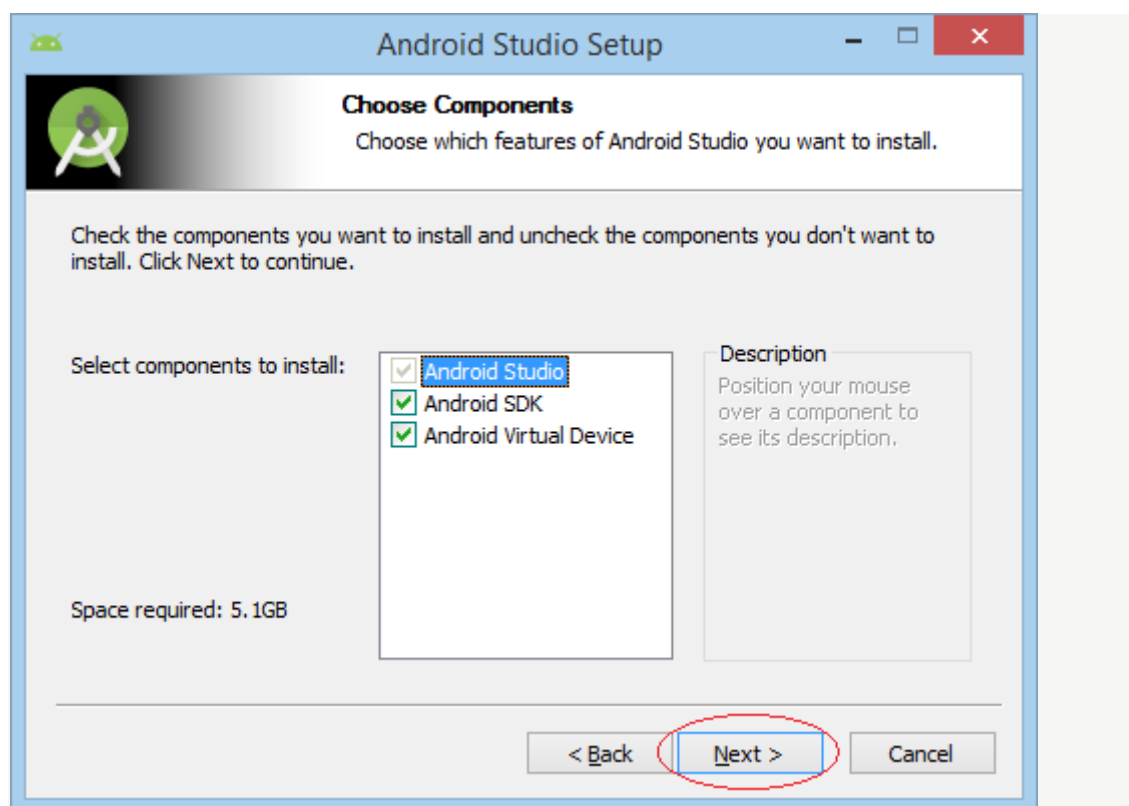
Tải và cài đặt theo đường dẫn: <https://developer.android.com/sdk/index.html>

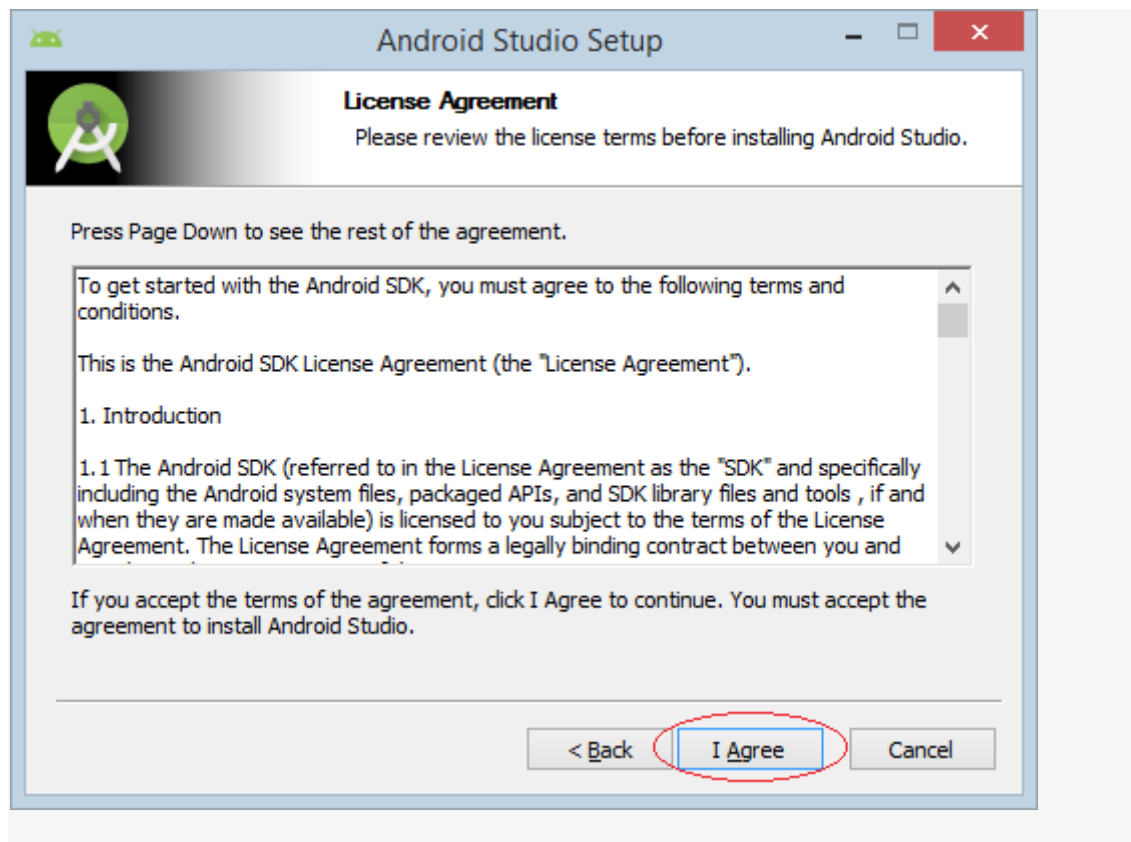
Tiền hành cài đặt:



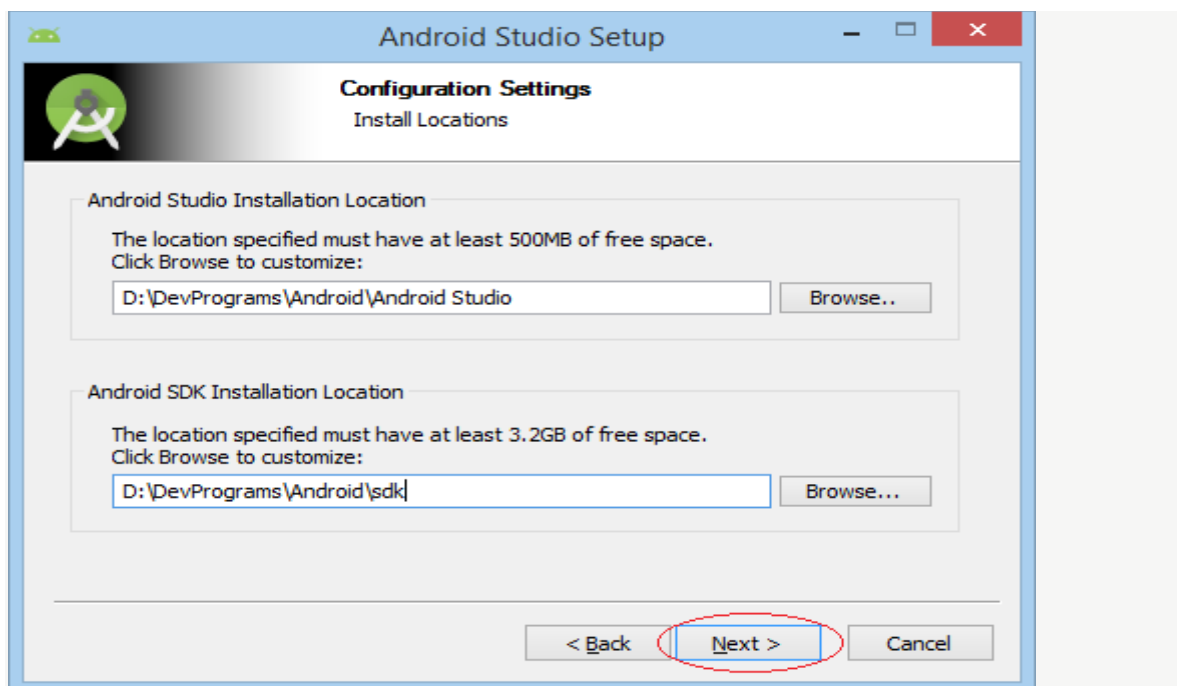
Android SDK (software development kit) là một tập hợp các công cụ được sử dụng để phát triển ứng dụng cho Android. Android SDK bao gồm:

- Các thư viện đòi hỏi
- Bộ dò lỗi (Debugger)
- Thiết bị giả lập (emulator)
- Các tài liệu liên quan cho Android API.
- Các đoạn code mẫu.
- Các hướng dẫn cho hệ điều hành Android.
- Android Virtual Device (AVD) là một thiết bị cấu hình, nó chạy với bộ giả lập Android (Android emulator). Nó làm việc với bộ giả lập để cung cấp một môi trường thiết bị ảo cụ thể, để cài đặt và chạy ứng dụng Android.

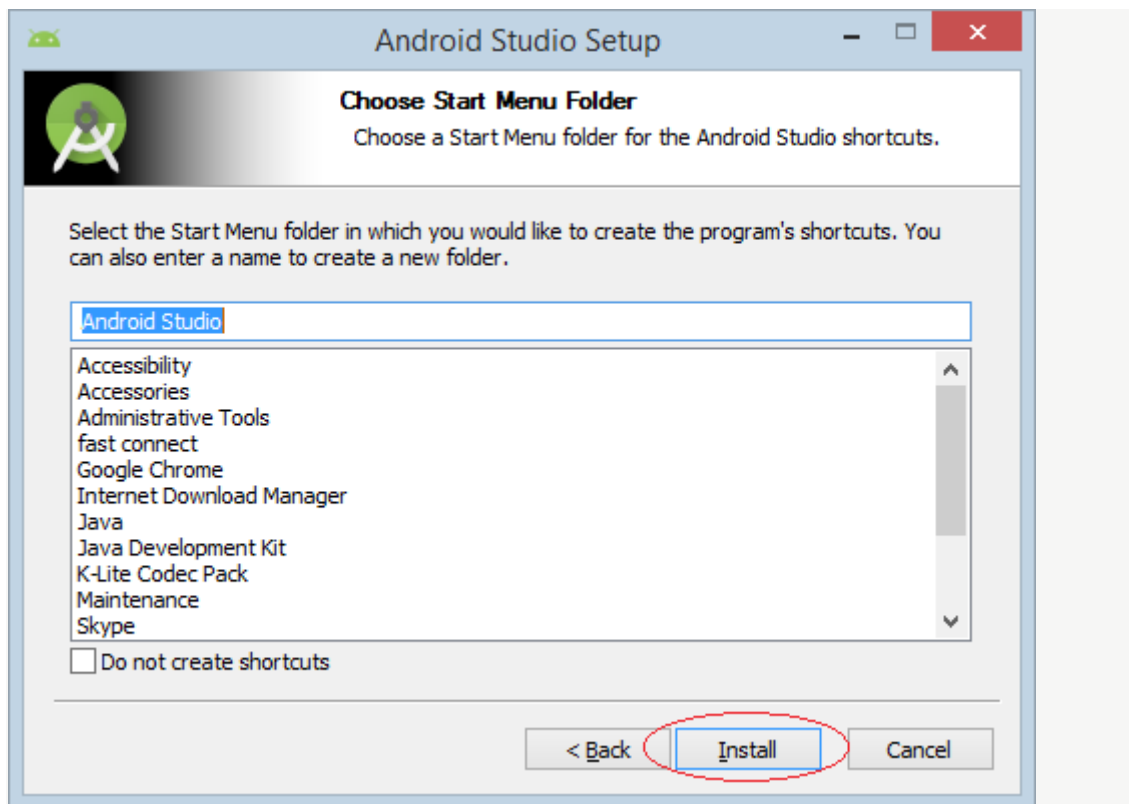




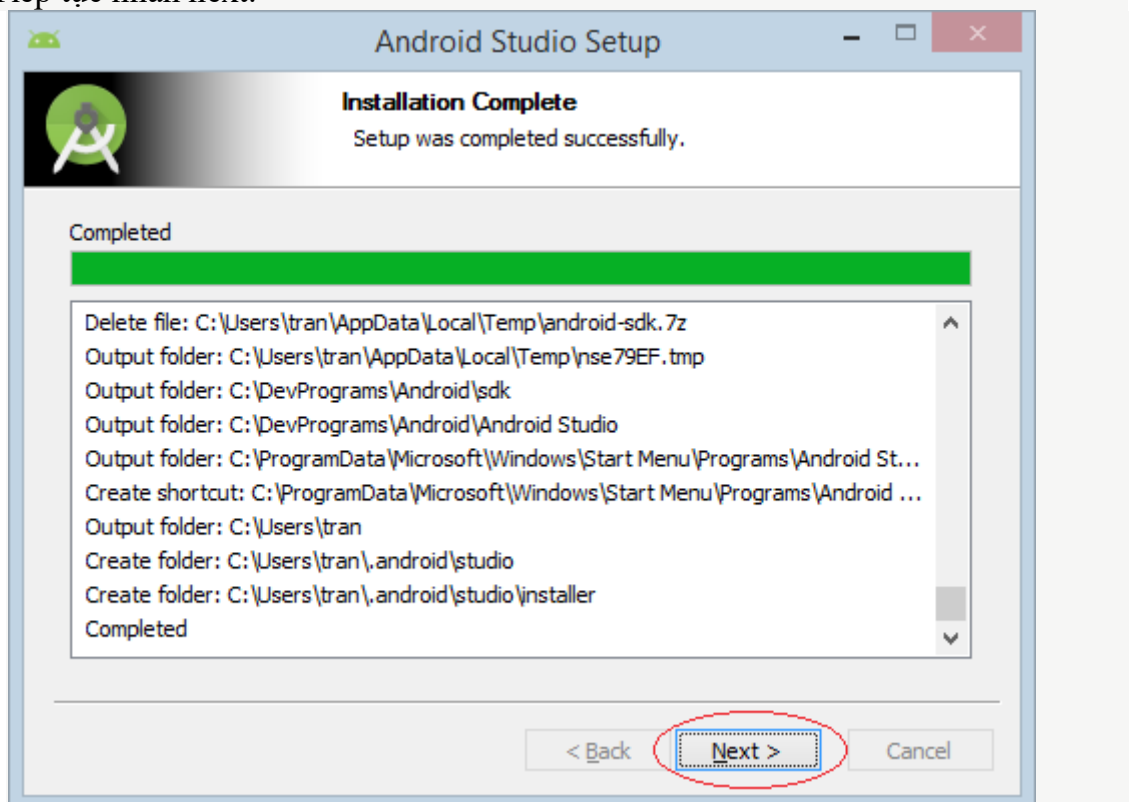
Chọn thư mục cài đặt sau đó chọn next:

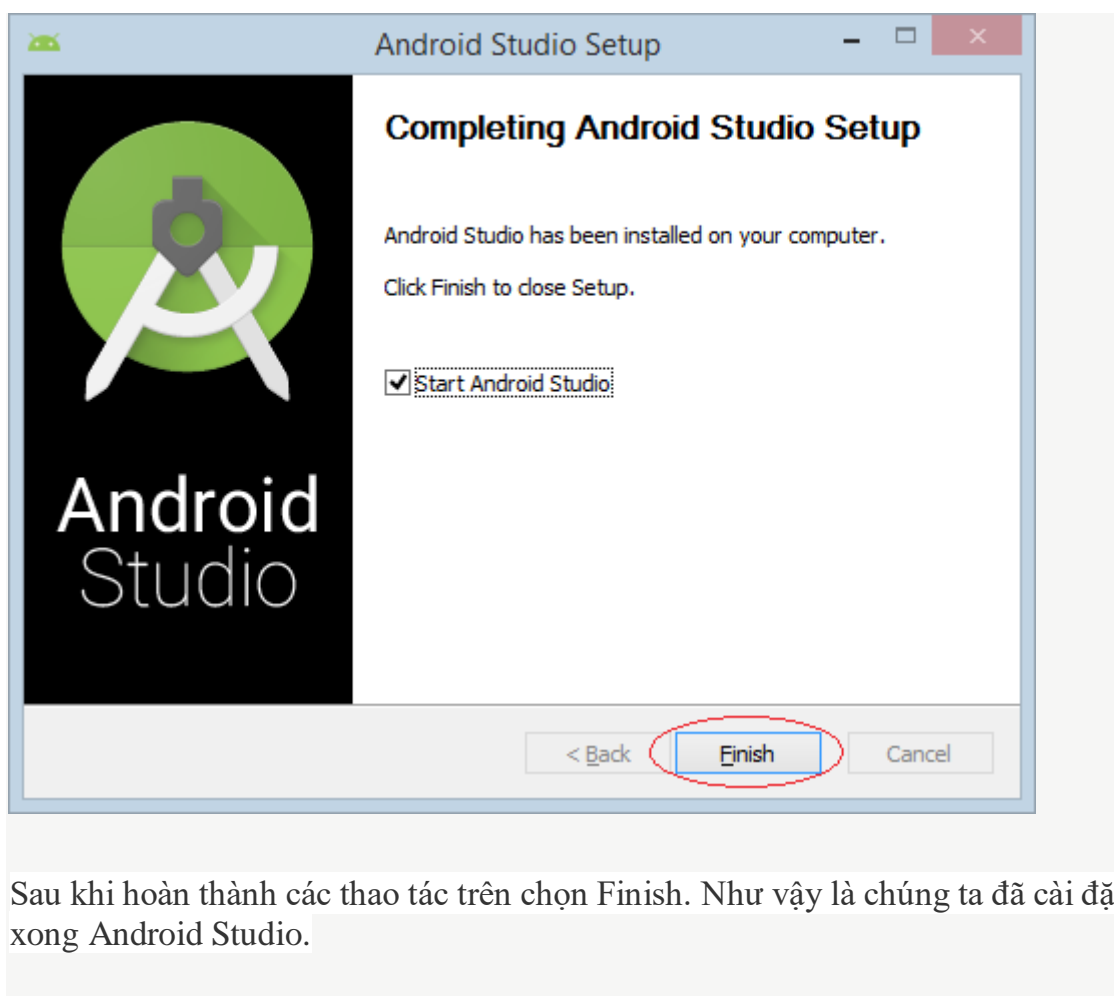


Tiếp tục chọn Install.



Tiếp tục nhấn next.





Bước 3: Cài đặt máy ảo GenyMotion.

Tải và cài theo đường dẫn: <https://www.genymotion.com/>

Yêu cầu:

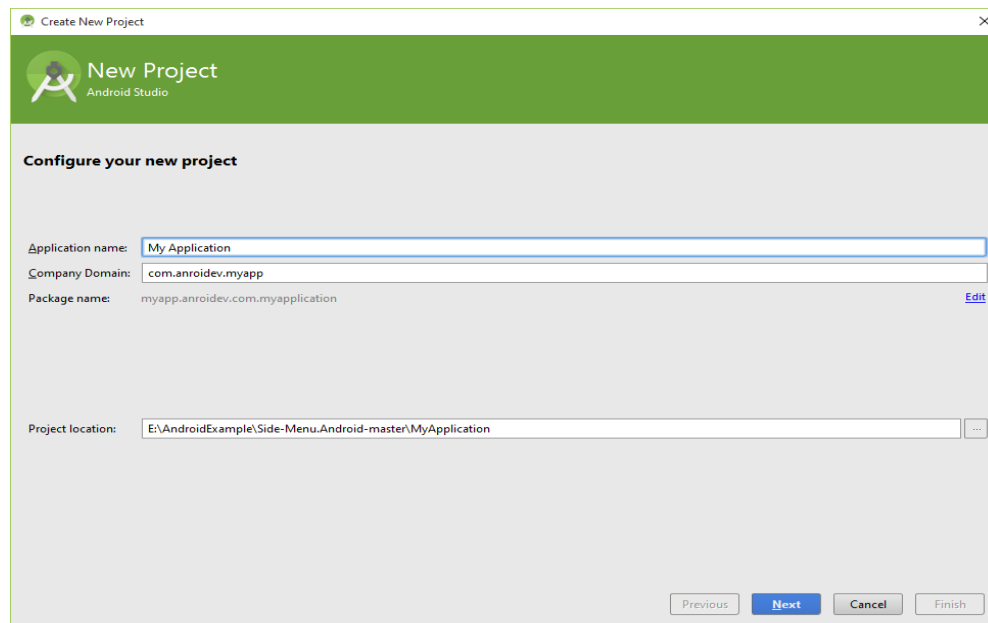
- Card đồ họa OpenGL 2.0.
- CPU hỗ trợ VT-x hoặc AMD-V và kích hoạt thiết lập BIOS.
- RAM: Tối thiểu 1 GB.
- Dung lượng trống của ổ cứng: Ít nhất là 2GB để cài đặt Genymotion và các máy ảo chạy Genymotion (đây chỉ là mức yêu cầu tối thiểu, bởi nếu sử dụng nhiều máy ảo cùng lúc và có nhiều ứng dụng, phần 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.

Tiến hành cài đặt.

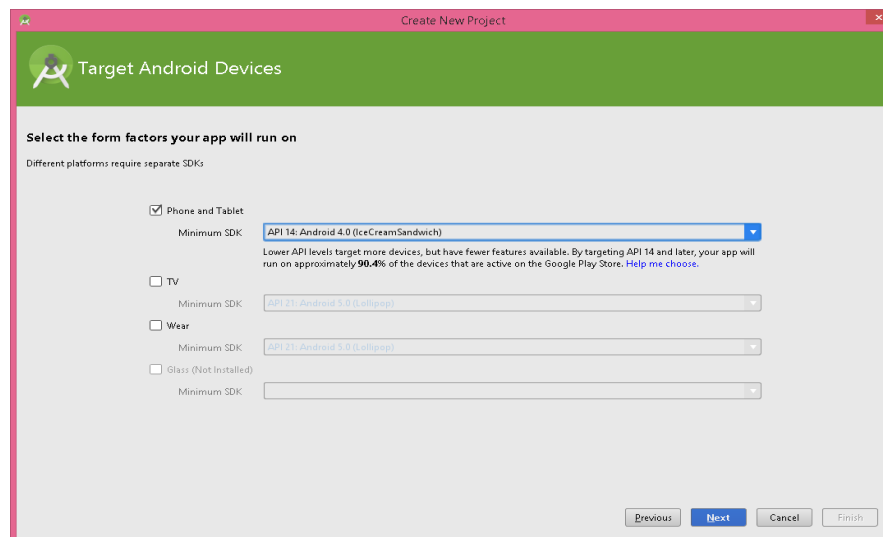
- Tìm tới vị trí lưu file vừa tải về đầu tiên, sau đó click đúp chuột trái vào file đó để bắt đầu tiến hành.
- Tùy theo ý muốn người dùng mà có thể Next ngay và cài đặt Genymotion ngay tại vị trí được gợi ý hoặc chọn Browse để tìm vị trí khác. Tiếp tục Next.
- Chọn mục Create a desktop shortcut để tạo biểu tượng cho phần mềm này trên máy tính, sau đó Next tiếp. Chọn vào Install để thực hiện cài đặt.
- 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.
- 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.
- Tiếp tục Next. Click Yes và tiếp theo là Install.
- Bỏ dấu tick ở Start Oracle VM VirtualBox và chọn vào Finish.

2.3. Cấu trúc dự án Android trong 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 Company Domain với Application name.
- Project location: Là nơi lưu trữ ứng dụng.
- Sau đó nhấp Next để tiếp tục.

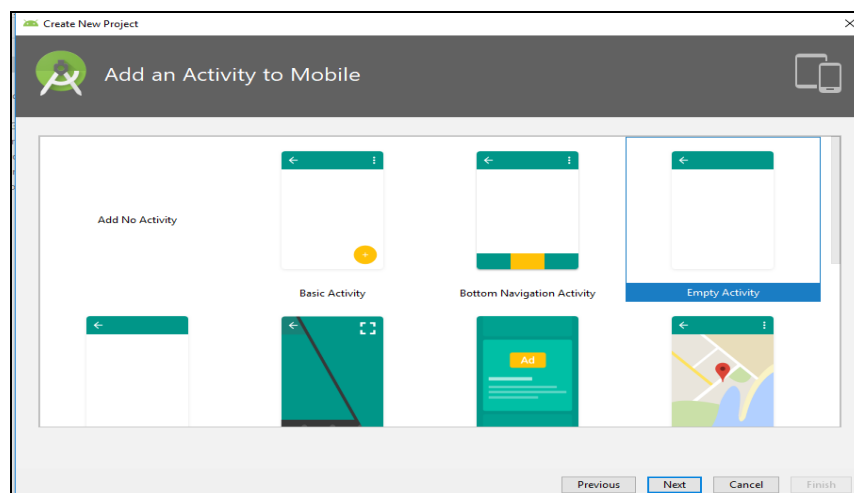


Ở hộp thoại trên cho phép ta lựa chọn là ứng dụng sẽ được viết cho những thiết bị nào (Phone and Tablet, TV, Wear).

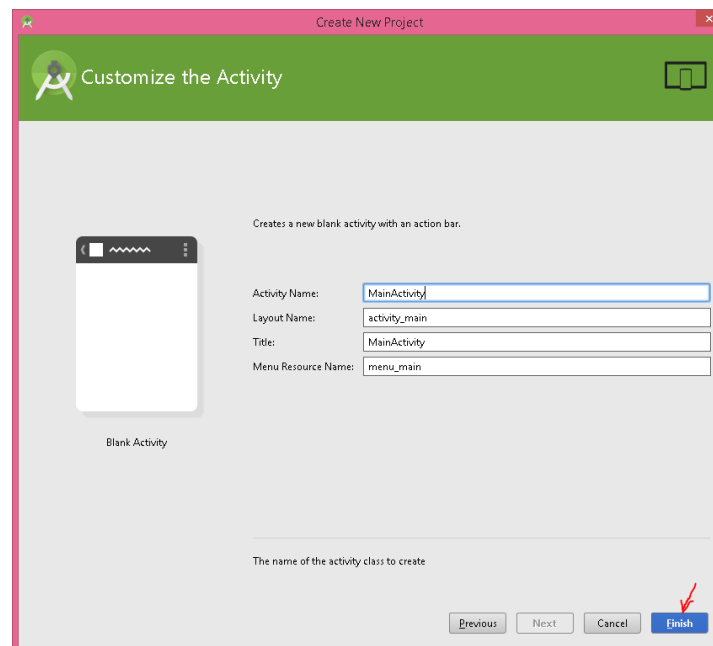
Ở mục Minimum SDK, quy định phiên bản android tối thiểu để chạy ứng dụng.

Hiện nay bản API14 Android 4.0 (IceCreamSandwich) vẫn đứng đầu về số lượng thiết bị sử dụng chiếm tới hơn 90%) nên thường lựa chọn.

Màn hình này hiển thị cho phép chọn loại Activity mặc định.

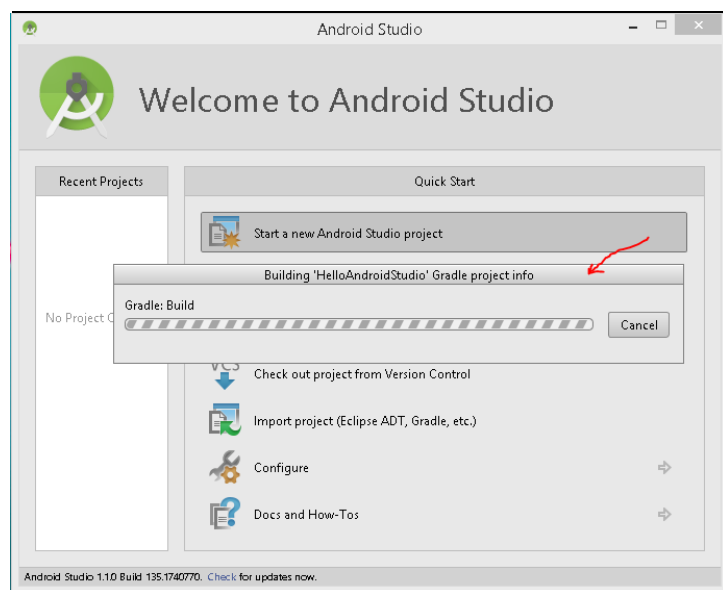


Chọn **Blank Activity** rồi bấm Next:

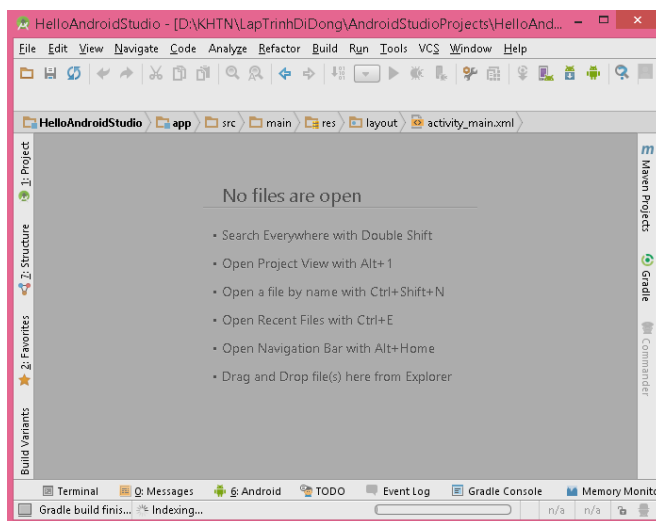


- Activity Name: Tên class Activity (java) để ta viết mã lệnh
- Layout Name: Tên file XML làm giao diện cho Activity Name.
- Title: Tiêu đề hiển thị khi kích hoạt Activity trên thiết bị.
- Menu Resource Name: Tên file xml để tạo menu cho phần mềm.

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



Khi build xong mặc định có màn hình dưới đây:

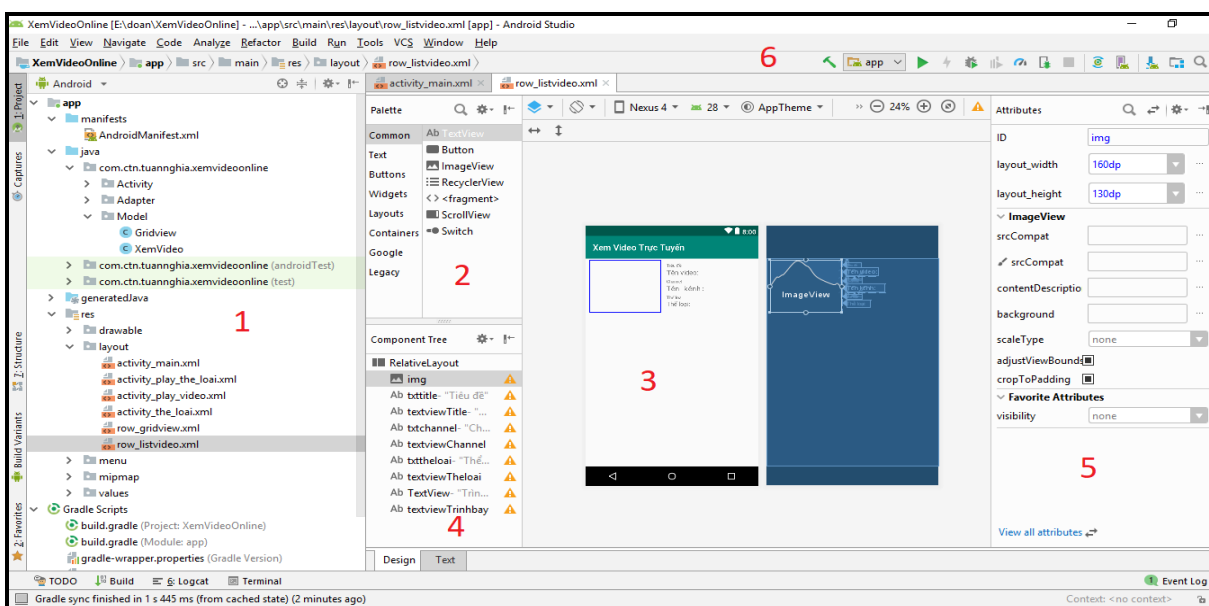


2.3.2. Màn hình làm việc của Android Studio.

Theo mặc định Android Studio hiển thị các files trong project theo góc nhìn Android. Góc nhìn này Android Studio sẽ tổ chức các files theo 3 module.

- Manifests: chứa file AndroidManifest.xml.
- Java: chứa các file mã nguồn Java.
- Res: chứa tất cả các file layout, xml, giao diện người dùng (UI), ảnh.

Mở Project mặc định activity_main.xml sẽ được chọn ta có màn hình sau chia thành 6 vùng:



Hình 2.3.2.1. Các vùng làm việc của Android Studio.

❖ **Vùng 1:** Là nơi cấu trúc hệ thống thông tin của Ứng dụng, Ta có thể thay đổi cấu trúc hiển thị (thường để mặc định là Android).

Ta có thể thấy AndroidManifest.xml nằm ở đây. File này vô cùng quan trọng trong việc cấu hình ứng dụng.

Các thư mục:

- drawable: chứa các file hình ảnh và xml trong ứng dụng.
- layout: chứa các giao diện màn hình được thiết kế dưới dạng xml.
- values: chứa các file lưu giá trị màu sắc, kích thước, chuỗi....

❖ **Vùng 2:** Là vùng khá quan trọng cho những bạn mới bắt đầu lập trình, nó là nơi hiển thị các View mà Android hỗ trợ, cho phép bạn kéo thả trực tiếp vào vùng 3 (Giao Diện Thiết Bị) để thiết kế.

Ở vùng số 2 này nó có 2 tab: Design và Text ở góc trái dưới cùng.

- Tab Design là tab mà ta đang nhìn và thao tác với nó (cho phép thiết kế giao diện bằng cách kéo thả).
- Tab Text là tab cho phép ta thiết kế giao diện bằng viết thẻ XML.

❖ **Vùng 3:** Là vùng giao diện thiết bị, cho phép các kéo thả View vào đây và đồng thời cho ta hiệu chỉnh View.

❖ **Vùng 4:** Khi màn hình có nhiều View thì vùng 4 này trở lên hữu ích, nó cho phép hiển thị giao diện theo dạng cấu trúc cây, nên ta dễ dàng quan sát và lựa View bị chồng lặp trên giao diện (vùng 3).

❖ **Vùng 5:** Vùng này rất quan trọng, đây là vùng cho phép thiết lập trạng thái hay thuộc tính cho các View trên giao diện.

❖ **Vùng 6:** Là vùng các chức năng quan trọng thường dùng trong Android Studio:

- Chạy ứng dụng và Debug ứng dụng.
- Quản lý máy ảo.
- Quản lý Android SDK Manager.
- Quản lý Android Device Manager.

2.4. Thành phần giao diện trong Android Studio.

2.4.1. Thành phần hiển thị.

Một thành phần hiển thị trên màn hình đó là một view. Trong Android có rất nhiều loại view, phù hợp nhiều nhu cầu. Bản thân lập trình viên cũng có thể tự tạo ra view của riêng mình bằng cách code các lớp Java *extends* từ lớp view [4].

Một số loại view tiêu biểu là:

- TextView: Hiển thị chữ. Có thể thay đổi nội dung chữ, màu sắc, kích cỡ, kiểu đậm/gạch chân....
- Button: Nút có thể thay đổi độ rộng, cao, màu nền nút, kiểu viền, đổ bóng,...
- Switch: Công tắc có 2 chế độ gạt qua gạt lại như chơi điện tử, nhưng mà chỉ trái-phải thôi.
- EditText: Là một ô nhập văn bản. Văn bản ở EditText có thể là 1 hay nhiều dòng, có thể cài đặt đầu vào chỉ được nhập số, nhập chữ hay nhập dạng mật khẩu.
- SeekBar: Thanh kéo sử dụng khi người dùng muốn thao tác gì đó về biên độ/độ lớn bằng tay.

Các View có những thuộc tính khác nhau, gọi là các attributes.

- Ví dụ như EditText thì có attributes là MaxLength (số ký tự tối đa) và MaxLines (số dòng tối đa). TextView thì có fontSize (cỡ chữ) và inputType (kiểu nhập: số / chữ / email...).
- Tuy nhiên các View đều có một số attributes chung như:
 - Layout_width: Độ dài chiều ngang của View. Có thể là con số cụ thể hoặc là wrap_content (Bao đủ nội dung bên trong thì thôi), hay match_parent (bằng với chiều ngang của View chứa nó).
 - Layout_height: Độ dài chiều dọc của View. Có thể là con số cụ thể hoặc là wrap_content (Bao đủ nội dung bên trong thì thôi), hay match_parent (bằng với chiều dọc của View chứa nó).
- ID: Cái này rất quan trọng, mỗi một View trong một layout đều được gắn với một định danh (id) này để gắn chức năng với code Java ở trong.

2.4.2. Một số nhóm hiển thị cơ bản.

LinearLayout.

Quá nổi tiếng, được sử dụng rất nhiều. Các View con cấp 1 trong Layout này sẽ được sắp xếp chỉ theo 1 hướng: Dọc hoặc ngang (Vertical / Horizontal), chỉ định bằng thuộc tính android:orientation.

Trong LinearLayout nhiều khả năng là bạn sẽ thấy có các thuộc tính sau hay được dùng:

- Android:gravity: Căn lề của nội dung của View (hướng nội, giống thuộc tính text-align trong CSS).
- Android:layout_gravity: Căn lề hướng ngoại, so với ViewGroup chứa nó, giống thuộc tính float trong CSS.
- Android:layout_weight: Tỷ lệ của view đó so với view mẹ. Mặc định view mẹ nếu không đặt thuộc tính weight_sum thì sẽ là 1 và view con sẽ có tỉ lệ nằm trong khoảng từ 0 đến 1 (chấp nhận số thập phân).

RelativeLayout.

Khác với LinearLayout, RelativeLayout và các Layout con của nó hiển thị bằng các quan hệ với các layout trong cùng một layout mẹ, hoặc với chính layout mẹ.

- Vị trí dựa trên quan hệ: layout_above, layout_below, layout_toLeftOf, layout_toRightOf.
- Vị trí dựa trên layout mẹ: android:layout_centerHorizontal, android:layout_centerVertical.
- Căn chỉnh dựa trên quan hệ: layout_alignTop, layout_alignBottom, layout_alignLeft, layout_alignRight, layout_alignBaseline.
- Căn chỉnh dựa trên layout mẹ: layout_alignParentTop, layout_alignParentBottom, layout_alignParentLeft, layout_alignParentRight.

FrameLayout.

FrameLayout là dạng layout rất đơn giản: Tất cả những view nằm trong đều được nhồi nhét phía trong nó. Và nếu bạn cần một layout sắp xếp theo trục tọa độ Z (giống z-index trong CSS) thì đây chính là lựa chọn phù hợp cho bạn.

ScrollView.

Một lớp mở rộng của `FrameLayout`, lớp `ScrollView` xử lý việc cuộn các đối tượng con của nó trên màn hình.

RecyclerView.

Lớp `RecyclerView` là một lớp con của `ViewGroup`, nó liên quan đến các lớp `ListView` và `GridView` và nó được cung cấp bởi Google thông qua thư viện hỗ trợ `RecyclerView` cho các phiên bản Android cũ hơn. Lớp `RecyclerView` đòi hỏi việc sử dụng các mẫu thiết kế view holder để tái sử dụng phần tử một cách có hiệu quả và nó hỗ trợ việc sử dụng một `LayoutManager`, một thành phần trang trí, và một phần tử động để làm cho thành phần này vô cùng linh hoạt và đơn giản.

CoordinatorLayout.

Được thêm gần đây vào thư viện hỗ trợ thiết kế, lớp `CoordinatorLayout` sử dụng một đối tượng `Behavior` để xác định cách các phần tử `View` con sẽ được sắp xếp và di chuyển khi người dùng tương tác với ứng dụng của.

2.5. Bắt và xử lý sự kiện trên giao diện.

Sự kiện là một cách hữu ích để thu thập dữ liệu về sự tương tác của người dùng với các thành phần tương tác của ứng dụng [5]. Giống như bấm vào một nút hoặc chạm vào màn hình cảm ứng, vv. Ta có thể nắm bắt những sự kiện trong chương trình và có những xử lý thích hợp theo yêu cầu. Có hai khái niệm liên quan đến quản lý sự kiện Android:

- Event Listeners là một interface. Event Listeners được sử dụng để đăng ký sự kiện cho các thành phần trong UI. (Đăng ký sự kiện).
- Event Handlers – Là phương thức xử lý khi phát sinh sự kiện. (Xử lý sự kiện).

Xử lý sự kiện trong android – Một số sự kiện thường gặp trong Android.

Event Handler	Event Listener Name
onClick()	OnClickListener() Đăng ký sự kiện khi người dùng hoặc click hoặc chạm (touche) hoặc focuse trên bất kỳ widget như button, text, image vv. Chúng ta sẽ sử dụng phương <code>onClick()</code> để xử lý sự kiện.

onLongClick()	OnLongClickListener() Đăng ký sự kiện khi người dùng hoặc click hoặc chạm (touche) hoặc focus trên bất kỳ widget như button, text, image vv. trong một hoặc nhiều giây. Chúng ta sẽ sử dụng phương onLongClick() để xử lý sự kiện.
onFocusChange()	OnFocusChangeListener() Sự kiện phát sinh khi widget mất focus.
onKey()	OnFocusChangeListener() Sự kiện phát sinh khi người dùng focus trên widget và nhấn (presse) hoặc thả (release) một phím trên thiết bị.
onTouch()	OnTouchListener() Sự kiện phát sinh khi người dùng nhấn phím, nhả phím, hoặc bất kỳ cử chỉ chuyển động trên màn hình.
onMenuItemClick()	OnMenuItemClickListener() Sự kiện phát sinh khi người dùng chọn một mục trong menu.
onCreateContextMenu()	OnCreateContextMenuListener() Sự kiện phát sinh khi người dùng chọn một mục trong menu ngữ cảnh (Context Menu)

2.6. Vòng đời ứng dụng Android.

Các Activity trong hệ thống được quản lý như 1 ngăn xếp activity (activity stack). Khi 1 activity mới bắt đầu nó được đặt lên đầu của ngăn xếp và trở thành Running Activity (activity đang chạy) [6], đồng thời activity trước đó sẽ nằm ngay phía dưới trong ngăn xếp đó, và sẽ không trở nên visible (nhìn thấy) cho đến khi activity ở trên thoát ra khỏi ngăn xếp.

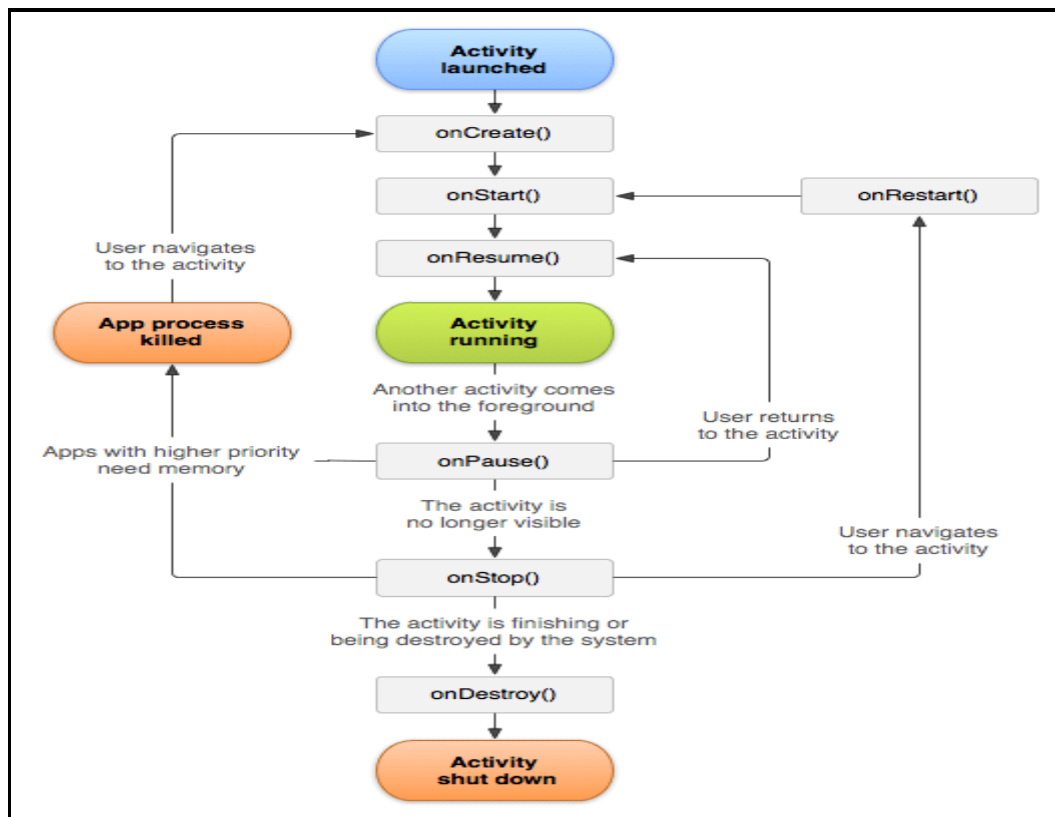
Một Activity gồm 4 trạng thái chính:

- Nếu activity ở phía trên của màn hình (hay ở trên cùng của ngăn xếp), thì nó đang ở trạng thái active (hoạt động) / running (đang chạy). Ví dụ khi ta cần gọi điện thì activity bấm số đó đang ở trạng thái active.
- Nếu activity không thể tương tác nhưng vẫn nhìn thấy (khi mà bị che bởi 1 activity khác nhưng người dùng vẫn có thể nhìn thấy nó ở phía sau) thì activity này đang ở trạng thái paused (tạm dừng). Khi ở trạng thái này activity có thể bị

xóa bỏ bởi hệ thống khi thiết bị thiếu bộ nhớ. Ví dụ khi có 1 activity khác dạng dialog hiện lên chỉ che đi 1 phần của activity hiện tại thì activity vào trạng thái paused.

- Nếu activity hoàn toàn bị che khuất bởi activity khác thì nó đang ở trạng thái stopped (đã dừng). Activity này vẫn giữ được tất cả trạng thái và thông tin, nhưng không còn hiển thị với người dùng và thường xuyên bị xóa bỏ bởi hệ thống khi thiếu bộ nhớ. Ví dụ khi ta tắt màn hình thì khi đó activity vào trạng thái stopped.
- Nếu activity ở trạng thái paused (tạm dừng) hay stopped (đã dừng), hệ thống có thể xóa bỏ activity đó khỏi bộ nhớ bằng cách yêu cầu nó tự kết thúc hoặc xóa bỏ tiến trình của nó. Khi activity đó hiển thị lại với người dùng thì sẽ được khởi tạo lại và khôi phục lại trạng thái trước đó.

Hình ảnh sau đây minh họa cho vòng đời của 1 Activity cùng với các trạng thái của nó:



Hình ảnh 2.6.1. Vòng đời của một Activity.

2.7. Thành phần Intent.

Trong Android, khả năng gửi tin nhắn đi xung quanh được thực hiện bởi đối tượng Intent [7]. Với sự trợ giúp của Intents, các thành phần của Android có thể yêu cầu chức năng từ các thành phần khác. Khi bạn mở ứng dụng Instagram trên điện thoại của bạn và sử dụng nó để chụp ảnh, bạn chỉ cần sử dụng một Intent.

Intent cũng giúp liên lạc giữa các phần của một ứng dụng dễ dàng. Di chuyển từ một màn hình (Activity) sang một màn hình khác được thực hiện thông qua Intents.

Bạn hãy nhìn nhận theo cách này : tất cả các thành phần (ứng dụng và màn hình) của thiết bị Android đều bị cô lập. Cách duy nhất chúng giao tiếp với nhau là thông qua Intents.

Như đã đề cập trước đó, bạn có thể sử dụng Intents để khởi chạy các thành phần khác : activities, services, và broadcast receivers.

Để khởi chạy một Activity, bạn sẽ sử dụng phương thức `startActivity(intent)`. Đoạn code sau mô tả cách khởi chạy một Activity từ Intent.

```
Intent numbersIntent = new Intent(MainActivity.this,
NumbersActivity.class);
```

```
startActivity(numbersIntent);
```

Đầu tiên, chúng ta tạo mới một đối tượng Intent và truyền vào nó tên class sẽ được khởi chạy NumbersActivity. Sau đó, chúng ta bắt đầu Activity sử dụng Intent đó.

Các loại Intent.

Android hỗ trợ hai loại Intent : minh bạch và ngầm. Khi một ứng dụng định nghĩa thành phần đích của nó trong một Intent, đó là một Intent minh bạch. Khi ứng dụng không đặt tên cho một thành phần đích, đó là một Intent ngầm.

❖ Intent minh bạch.

Đoạn code nêu trên chính là ví dụ của Intent minh bạch. Chúng ra hãy xem lại một lần nữa :


```
Intent numbersIntent = new Intent(MainActivity.this,
NumbersActivity.class);

startActivity(numbersIntent);
```

Ở đây, NumbersActivity là mục đích rõ ràng từ MainActivity. Điều này có nghĩa là NumbersActivity là thành phần được xác định sẽ được gọi bởi hệ thống Android. Điều quan trọng cần lưu ý (như trong ví dụ ở trên), các Intent minh bạch thường được sử dụng trong một ứng dụng, bởi vì nó cho phép nhà phát triển kiểm soát nhiều nhất về lớp nào sẽ được khởi chạy.

❖ Intent ngầm.

Đây là ví dụ về Intent ngầm :

```
Intent intent = new Intent(Intent.ACTION_VIEW,
Uri.parse("http://recruit.framgia.vn/"));

startActivity(intent);
```

Nếu bạn thực hiện đoạn code trên, ứng dụng của bạn sẽ khởi chạy một thành phần trình duyệt cho một URL nhất định thông qua Intent. Nhưng hệ thống Android xác định các thành phần có thể phản ứng với một Intent nhất định như thế nào?

Một thành phần có thể được đăng ký thông qua một Intent Filter cho một hành động cụ thể. Intent Filter có thể được đăng ký cho các thành phần tĩnh trong AndroidManifest.xml.

CHƯƠNG 3: CÁC KỸ THUẬT XÂY DỰNG ỨNG DỤNG XEM VIDEO TRỰC TUYẾN

3.1. Dịch vụ web và cơ sở dữ liệu trên internet.

Dịch vụ web (tiếng Anh: web service) là sự kết hợp các máy tính cá nhân với các thiết bị khác, các cơ sở dữ liệu và các mạng máy tính để tạo thành một cơ cấu tính toán ảo mà người sử dụng có thể làm việc thông qua các trình duyệt mạng.

Bản thân các dịch vụ này sẽ chạy trên các máy chủ trên nền Internet chứ không phải là các máy tính cá nhân, do vậy có thể chuyển các chức năng từ máy tính cá nhân lên Internet. Người sử dụng có thể làm việc với các dịch vụ thông qua bất kỳ loại máy nào có hỗ trợ web service và có truy cập Internet, kể cả các thiết bị cầm tay. Do đó các web service sẽ làm Internet biến đổi thành một nơi làm việc chứ không phải là một phương tiện để xem và tải nội dung.

Cơ sở dữ liệu (viết tắt CSDL; tiếng Anh là database) là một tập hợp những thông tin được tổ chức để dễ dàng trong việc tạo lập, cập nhập và khai thác thông tin. Cơ sở dữ liệu được duy trì dưới dạng một tập hợp các tập tin trong hệ điều hành hay được lưu trữ trong các hệ quản trị cơ sở dữ liệu.

a. Dịch vụ hosting.

Hosting là dịch vụ lưu trữ và chia sẻ dữ liệu trực tuyến, là không gian trên máy chủ có cài đặt các dịch vụ Internet như word wide web (www), truyền file (FTP), Mail,... ta có thể chứa nội dung trang web hay dữ liệu trên không gian đó.

Các thông số cần biết trong hosting:

Hệ điều hành (OS) của máy chủ : hiện tại có hai loại OS thông dụng là Linux và Windows.

- Hosting Linux: là Hosting chuyên hỗ trợ ngôn ngữ lập trình PHP, Joomla, các mã nguồn mở...
- Hosting Windows: Hosting Windows chuyên hỗ trợ về ngôn ngữ lập trình ASP, ASP.Net, HTML vì các Ngôn ngữ này, chạy chuyên trên Hosting Windows,

do vậy khi load Web sẽ hỗ trợ tốt hơn, Hosting Windows có hỗ trợ ngôn ngữ PHP, nhưng chủ yếu là hỗ trợ chính là ASP.

- Dung lượng: Bộ nhớ lưu trữ cho phép bạn tải file lên host.
- Băng thông: Bandwidth (băng thông) là thông số chỉ dung lượng thông tin tối đa mà website được lưu chuyển qua lại mỗi tháng.
- PHP: Phiên bản php hỗ trợ.
- Max file: Số lượng file tối đa có thể upload lên host.
- RAM: Bộ nhớ đệm
- Addon domain: Số lượng domain ta có thể trở tới hosting.
- Subdomain: Số lượng tên miền phụ có thể tạo ra cho mỗi tên miền.
- Park domain: Số lượng tên miền có thể parking.
- Email accounts: Số lượng email đi kèm với hosting.
- FTP accounts: Số lượng FTP account bạn có thể tạo và dùng nó upload dữ liệu lên hosting.

b. Đăng ký mua và sử dụng hosting.

Tùy vào mục đích sử dụng của mỗi cá nhân, tổ chức và doanh nghiệp có thể chọn các hình thức đăng ký mua, thuê hosting khác nhau tại các nhà cung cấp hosting tại Việt Nam như: PA Việt Nam, DIGSTAR, Viettell, FPT, Vhost,...

Hiện nay cũng có rất nhiều hosting miễn phí như: Hostinger.vn, 000webhost.com, Hostbuddy.com, freewebhostingarea.com,...

c. Các bước đăng ký hosting tại freewebhostingarea.com.

- Truy cập đường dẫn: https://maximus.freewha.com/cgi-bin/create_ini.cgi
- Điền tên miền theo ý bạn như hình 3.1.1 ví dụ bên dưới:

Hình 3.1.1. Đăng ký hosting.

- Điền đầy đủ thông tin và ấn CREATE để hoàn tất như hình 3.1.2 ví dụ dưới.

The screenshot shows the account creation page for Free Web Hosting Area .com. At the top, it says 'xemvideo1.xp3.biz is available on Maximus.freewha.com server.' Below this is the 'Account Information' section. The 'E-mail' field contains 'huucaoco.ct@gmail.com'. A note states 'You need a valid email address to confirm your account.' The 'Password' and 'Re-type password' fields are both filled with '*****'. A note below these fields says 'Password must have minimum 6 characters including letters and numbers. Do not use special characters or spaces.' There is a checked checkbox for 'I have read the Service Agreement and agree to its terms.' At the bottom, there is a 'CREATE' button.

Hình 3.1.2. Các thông tin cần điền.

- Khi tạo thành công bạn sẽ nhận được các thông tin cần thiết để quản trị tên miền như hình 3.1.3.

Be sure that your account has minimum 1hit per month. Inactive accounts are deleted at the end of month.

Account Manager:
 You're hosted on maximus.freewha.com server. Account Manager is located at <http://xemvideo1.xp3.biz/cpanel>.
Username: xemvideo1.xp3.biz
Password: tuannghia1
 If you have accounts created on more than one server, see <http://www.freewebhostingarea.com/members/>.
 Some scripts require **server path**. Yours is **/home/vhosts/xemvideo1.xp3.biz**.

Your Personal FTP Information:
 FTP Server/Host: xemvideo1.xp3.biz
 FTP Login/Username: xemvideo1.xp3.biz
 FTP PassWord: tuannghia1
 To upload your files go to our **Web based FTP client** (recommended), located at <http://xemvideo1.xp3.biz/ftp/>.
 Once logged in, you will be able to create a **new dir**, **new file**, to **upload a zip archive** and have it **unzipped** automatically, to **edit** your files, to select files and download them as .zip, to **copy**, **move**, **delete**, **rename**, **chmod** (change permissions), **search** anything under your account, ability to see and edit dot files like **.htaccess**. With a normal ftp program you will not be able to do this.
 For a complete list of features see <http://xemvideo1.xp3.biz/ftp/help.html>.

MySQL database Information:
 You will get a limited number of free databases per account. But you can create other accounts and get as many free databases as you want.
ATTENTION!!! MySQL databases must be activated @ Account Manager >> Hit Create DB button.
 MySQL HostName is **localhost** only. We don't allow external connections.
phpMyAdmin is located at <http://xemvideo1.xp3.biz/pma/>. Log in using **MySQL username**.
PhpMyAdmin (PMA) is a tool that will help you to administer your database. You don't have privileges to create a database with PMA because, as said before, databases must be created **@ Account Manager** using **Create DB** button. **Database details** (db name, username) **will be provided at Account Manager once you create the database.**

Other features at Account Manager
One click Autoinstaller for **PhpBB3**, **SMF**, **WordPress**, **Joomla**, **Drupal** (with Clean URLs preenabled), **Moodle**.
One click database backup & One click database import

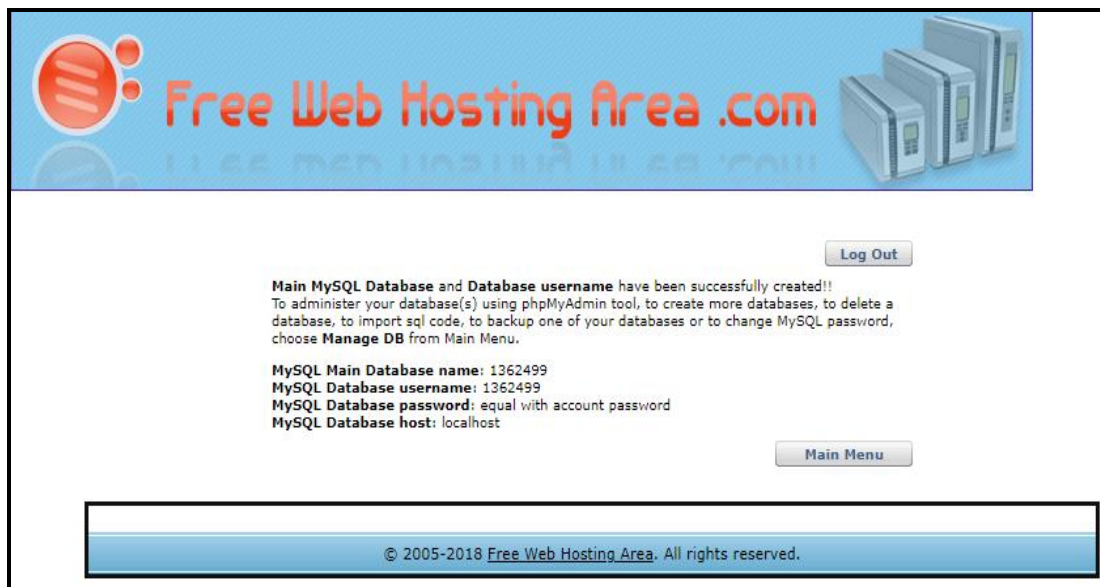
Hình 3.1.3. Các thông tin nhận được khi đăng ký xong.

- Truy cập đường dẫn <https://maximus.freewha.com/cgi-bin/amanager.cgi> đăng nhập tài khoản bạn vừa tạo để vào trang quản trị như hình 3.1.4.



Hình 3.1.4. Giao diện trang quản trị hosting.

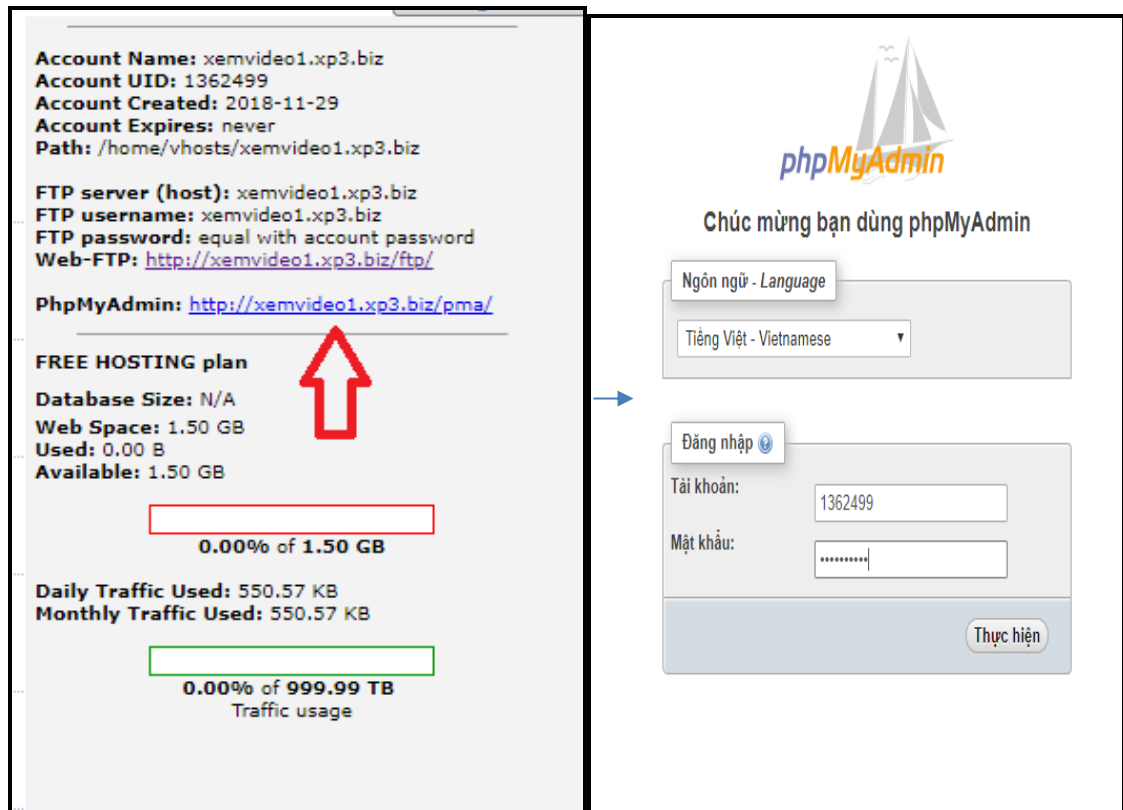
- Chọn Create DB để tạo cơ sở dữ liệu. Sau đó bạn sẽ nhận được các thông tin như hình 3.1.5 ví dụ dưới. Thông tin này để bạn có thể đăng nhập vào phpMyAdmin.



Hình 3.1.5. Các thông tin nhận khi tạo database.

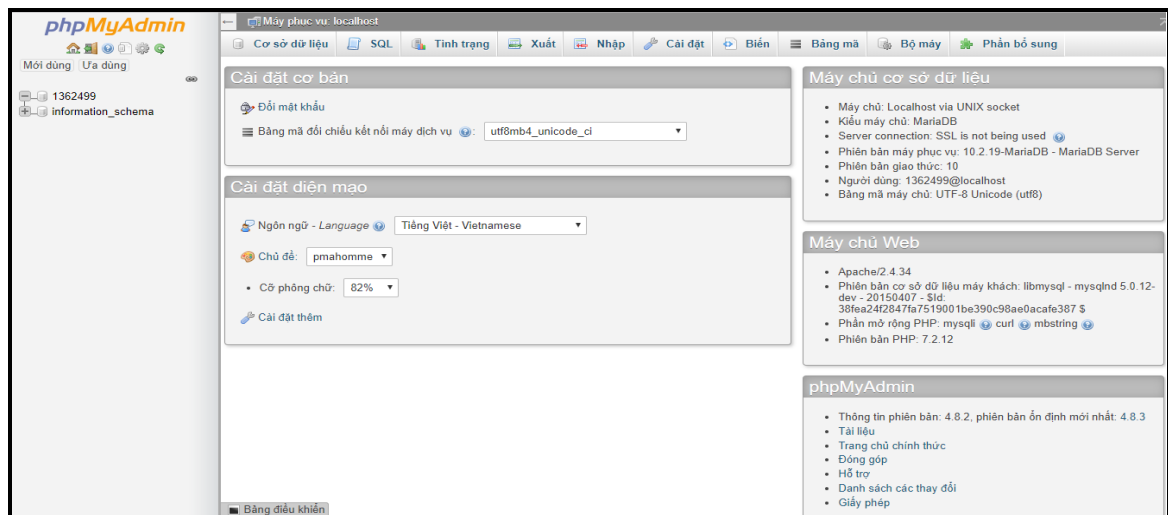
- Tiếp theo chọn vào đường dẫn đến cơ sở dữ liệu của bạn như hình 3.1.6.

- Bạn sẽ được chuyển sang trang đăng nhập phpMyAdmin. Bạn cần nhập đúng thông tin bạn vừa nhận được ở bên trên.



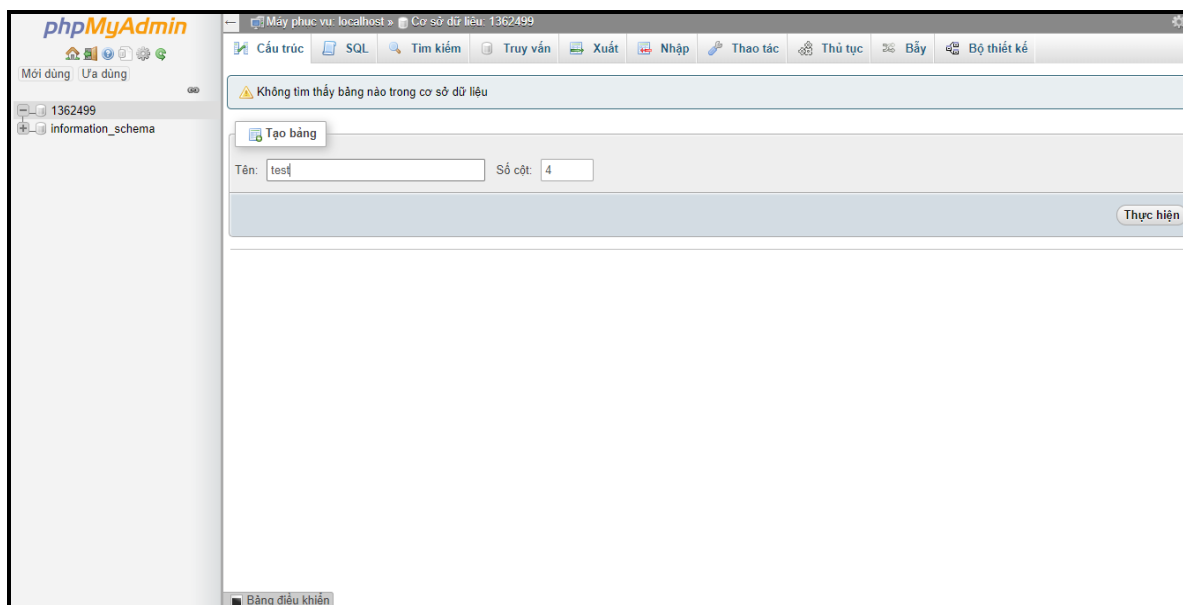
Hình 3.1.6. Đăng nhập phpMyAdmin.

- Sau khi bạn đăng nhập phpMyAdmin thành công. Sẽ chuyển sang có giao diện như hình 3.1.7 ví dụ dưới.



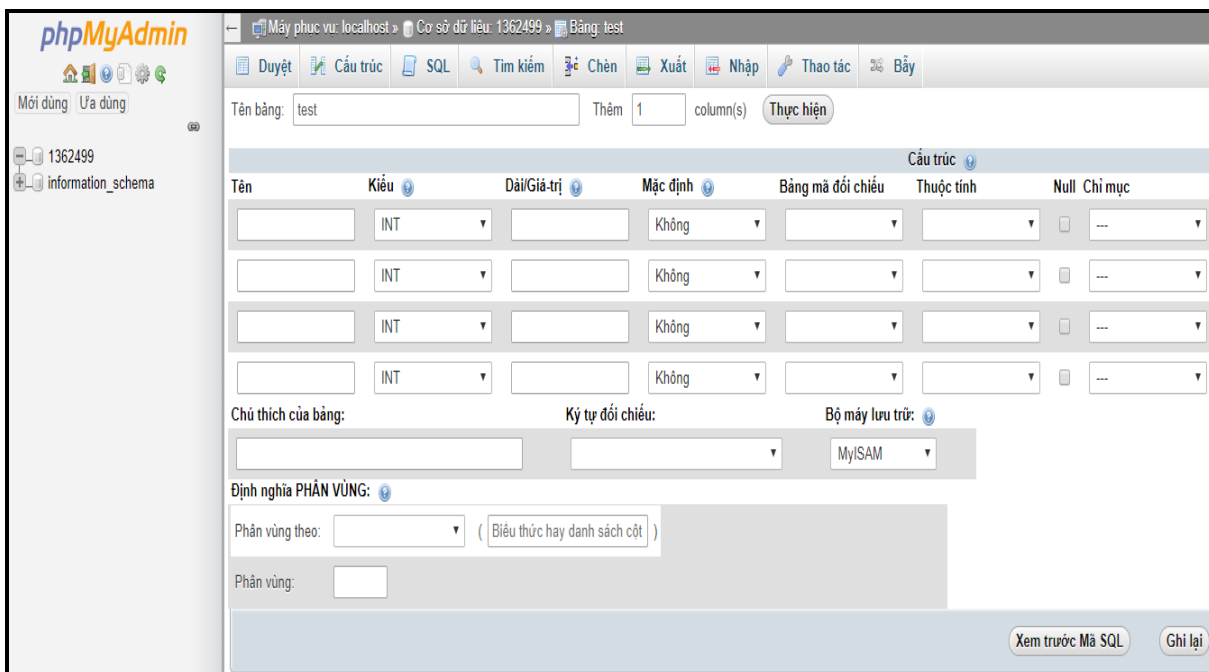
Hình 3.1.7. Giao diện đăng nhập thành công.

- Tiếp theo bạn cần tạo bảng: điền tên bảng bạn cần tạo vào số cột rồi chọn thực hiện như hình 3.1.8 ví dụ dưới.



Hình 3.1.8. Tạo bảng trong database.

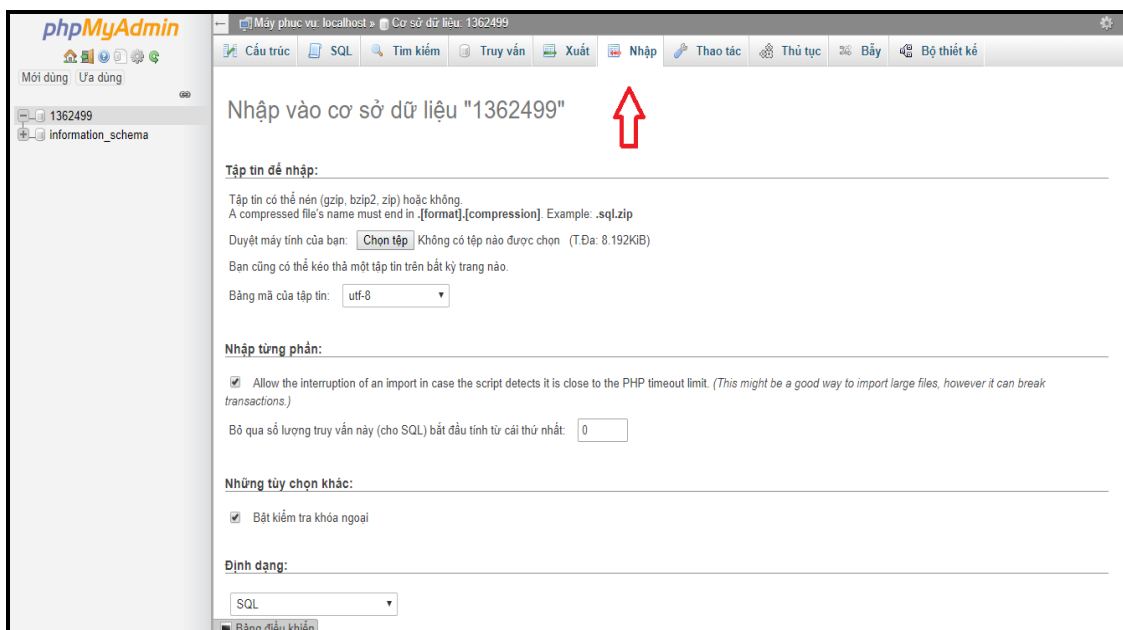
- Nhập thông số cho bảng dữ liệu: tên hàng tên cột số hàng số cột, kiểu dữ liệu của hàng cột trong bảng... bạn cũng có thể sửa tên bảng vừa tạo tại đây. Sau đó ấn ghi lại để hoàn tất quá trình tạo bảng mới database như hình 3.1.9 ví dụ dưới.



Hình 3.1.9. Nhập các thông tin để tạo bảng.

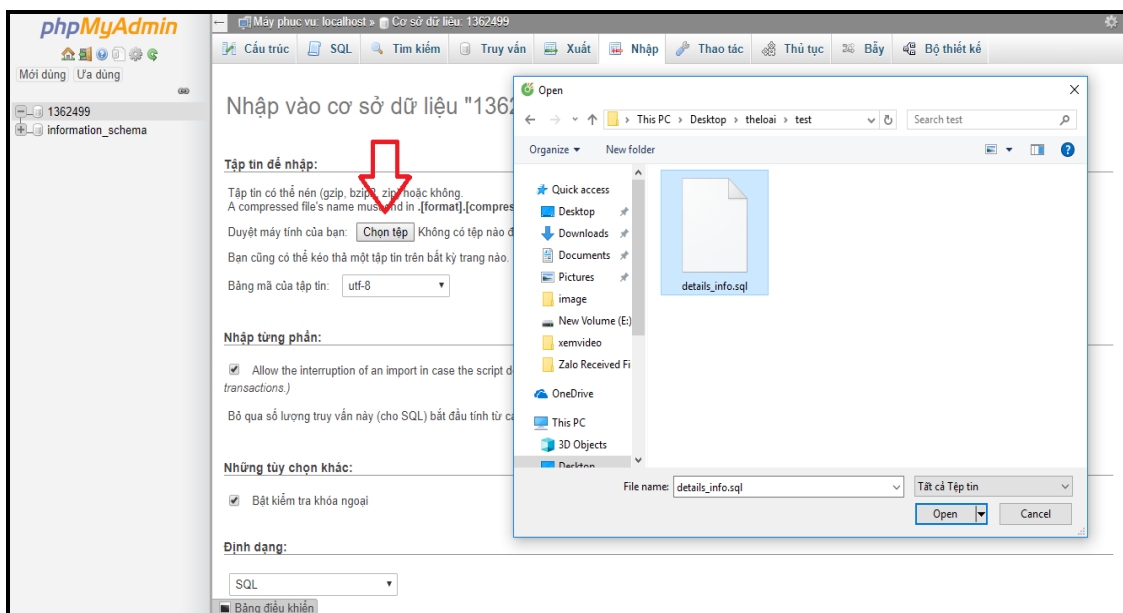
d. Tải cơ sở dữ liệu lên hosting.

- Chọn nhập để thực hiện chức năng tải file cơ sở dữ liệu bạn có sẵn như hình 3.1.10 ví dụ dưới.



Hình 3.1.10. Chọn nhập trên thanh công cụ.

- Sau đó click chọn tệp và bạn chỉ cần chọn đến vị trí file cơ sở dữ liệu của bạn. Chọn thực hiện để hoàn tất như hình 3.1.11 ví dụ dưới.



Hình 3.1.11. Chọn đường dẫn đến file cần tải lên database.

3.2. Một số kỹ thuật lập trình cơ sở dữ liệu trên internet.

3.2.1. Xây dựng Webservice.

Thực tế ứng dụng Android muốn trao đổi thông tin được với cơ sở dữ liệu trên Internet thì phải thông qua Webservice. Vì vậy việc xây dựng Webservice là rất quan trọng cho việc viết chương trình ứng dụng trên Android với cơ sở dữ liệu Internet. Muốn xây dựng được Webservice ta cần nắm được kỹ năng lập trình PHP và các kỹ thuật kết nối liên quan đến CSDL.

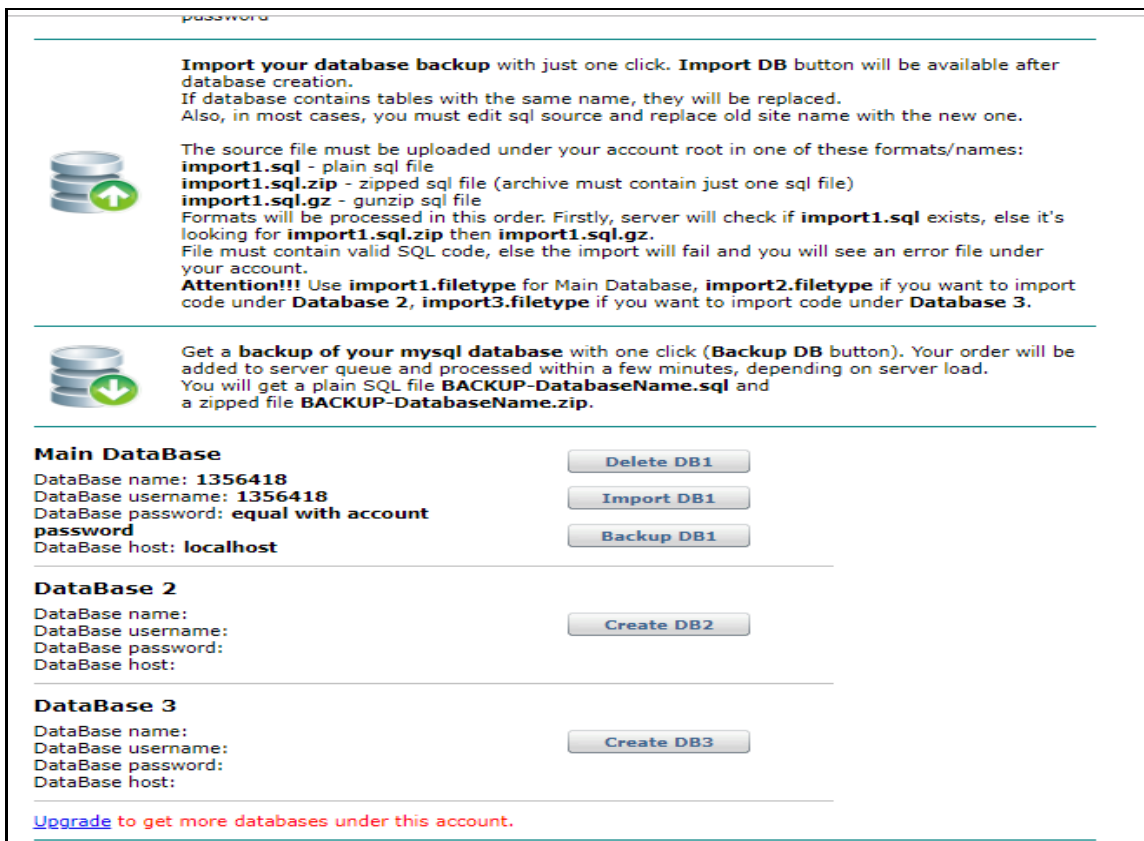
Định dạng dữ liệu (html, json, text, xml,...): Khi Client gửi một yêu cầu tới Webservice, nó thường được truyền tải dưới dạng dữ liệu mà máy tính hiểu được (XML hoặc JSON) và thông thường nhận về với hình thức tương tự từ máy chủ. Tuy nhiên Client cũng có thể chỉ định kiểu dữ liệu nhận về mà nó mong muốn (JSON, hoặc XML,...) các chỉ định này được gọi là các kiểu MIME, nó được gửi kèm trên phần HEADER của request.

Kiểu dữ liệu Json: tên đầy đủ JavaScript Object Notation [2] (gọi tắt là JSON) là một kiểu dữ liệu mở trong JavaScript. Kiểu dữ liệu này bao gồm chủ yếu là text, có thể đọc được theo dạng cặp "thuộc tính - giá trị". Nó mô tả một vật thể bằng cách bọc những vật thể con trong vật thể lớn hơn trong dấu ngoặc nhọn ({ }). JSON là một kiểu dữ liệu trung gian, chủ yếu được dùng để vận chuyển thông tin giữa các thành phần của một chương trình.

Tiến hành xây dựng Webservice:

Trước khi đi vào xây dựng: truy cập vào đường dẫn <https://www.sublimetext.com/> để tải và cài đặt Sublime Text. Sublime Text là một trình soạn thảo mã nguồn đa nền tảng độc quyền với giao diện lập trình ứng dụng Python (API) [2].

Chuẩn bị các thông tin đã nhận được từ việc tạo cơ sở dữ liệu ở phần trên như ví dụ hình 3.2.1.1:



Hình 3.2.1.1 Các thông tin đăng nhập database.

Bước 1: Tạo bảng trong CSDL để lưu trữ các thông tin cho Webservice.

Tùy theo nhu cầu của dự án để tạo bảng gồm các trường cho phù hợp.

Bước 2: Xây dựng chức năng kết nối CSDL với Android cho Webservice.

- Localhost: tương ứng với DataBase host (hình 3.2.1.1).
- 1356418: vị trí sau localhost, tương ứng với DataBase username (hình 3.2.1.1).
- Tuannghia12: tương ứng với DataBase password (hình 3.2.1.1).
- 1356418: vị trí cuối, tương ứng với DataBase name (hình 3.2.1.1).

```
<?php
// $db = mysqli_connect("localhost","root","","xemvideo");
$db = mysqli_connect("localhost","1356418","tuannghia12","1356418");
mysqli_query($db,"SET NAMES 'utf8'");

?>
```

Bước 3: Xây dựng chức năng thêm dữ liệu vào CSDL cho Webservice như ví dụ dưới đây từ hình 3.2.1.2 cho đến hình 3.2.1.6. Tạo một thư mục có tên images để lưu trữ toàn bộ ảnh.

```
<?php
require_once('index.php');
?>

<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<meta name="viewport" content="width=device-width,initial-scale=1.0"/>
<title>Thêm video</title>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" integrity="sha384-BViiSIFe" />
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css" />
<script type="text/javascript"></script>
<style type="text/css">
.thongbao{
    color:red;
    font-size: 15px;
}
</style>
</head>
<body>
<form method="post">
<div class="container" align="center" style="margin-top: 50px;">
<div class="get" >
<input type="text" class="form-control" style="width: 450px;" name="linkyoutube" value="<?php if(isset($_POST['linkyoutube'])&$_POST['linkyoutube']>
<br>
</div>
<button class="get btn btn-danger" style="width: 450px;" type="submit" name="getid">GET ID VIDEO</button>
</div>
</form>
```

Hình 3.2.1.2. Tạo giao diện cho chức năng cắt link video.

```
<?php

$upload_dir = 'images/';

if(isset($_POST['add'])){
//
$video_id = $_POST['video_id'];
$title = $_POST['title'];
$channelTitle = $_POST['channelTitle'];
$trinhbay = $_POST['trinhbay'];
$theloai = $_POST['theloai'];
// xy ly img
$imgName = $_FILES['myfile']['name'];
$imgTmp = $_FILES['myfile']['tmp_name'];
$imgSize = $_FILES['myfile']['size'];
if(empty($video_id)){
    $thongbao = 'ID video không được trống';
}elseif(empty($title)){
    $thongbao = 'Vui lòng nhập tiêu đề';
}elseif(empty($channelTitle)){
    $thongbao = 'Vui lòng nhập channel';
}
}elseif(empty($imgName)){
    $thongbao = 'Bạn chưa chọn hình ảnh cho video :)';
}else{
    //get image extension
    $imgExt = strtolower(pathinfo($imgName, PATHINFO_EXTENSION));
    //allow extension
    $allowExt = array('jpeg', 'jpg', 'png', 'gif');
    //random new name for photo
    $image = time().'_' . rand(1000,9999) . '.' . $imgExt;
    //check a valid image
    if(in_array($imgExt, $allowExt)){
```

Hình 3.2.1.3. Khai báo các biến.

Bước 4: Xây dựng chức năng truy vấn CSDL hiển thị toàn bộ thông tin video trong một giới hạn như hình 3.2.1.7 ví dụ dưới và thay đổi ngẫu nhiên theo ngày rồi chuyển sang dạng mã JSON.

```

<?php
require_once('index.php');
/**
 *
 */
class linkyoutube
{
    function linkyoutube($id, $video_id, $title, $channelTitle, $trinhbay, $theloai, $img)
    {
        # code...
        $this->id = $id;
        $this->video_id = $video_id;
        $this->title = $title;
        $this->channelTitle = $channelTitle;
        $this->trinhbay = $trinhbay;
        $this->theloai = $theloai;
        $this->img = $img;
    }
}
$query = "SELECT DISTINCT details_info.id, details_info.video_id, details_info.title, details_info.channelTitle, details_info.trinhbay,
details_info.theloai, details_info.img, theloai.id, theloai.tentheloai FROM `details_info`
INNER JOIN `theloai` ON details_info.theloai = theloai.id WHERE details_info.theloai = theloai.id ORDER BY RAND("date("Ymd").") LIMIT 50";
$data = mysqli_query($db, $query);
$mang = array();
while ($row0 = mysqli_fetch_assoc($data)) {
    # code...
    array_push($mang, new linkyoutube($row0['id'],$row0['video_id'],$row0['title'],$row0['channelTitle'],$row0['trinhbay'],
$row0['tentheloai'],$row0['img']));
}
echo json_encode($mang);
?>

```

Hình 3.2.1.7. Truy vấn và hiển thị toàn bộ thông tin video.

Bước 5: Xây dựng chức năng hiển thị toàn bộ thể loại trên một danh sách và hiển thị chi tiết từng thể loại rồi chuyển sang dạng mã JSON như các hình 3.2.1.8 và hình 3.2.1.9 ví dụ dưới.

```

<meta http-equiv="content-style-type" content="text/html; charset= utf-8" />
<?php
require_once('index.php');
/**
 *
 */
$sql = "SELECT * FROM `theloai`";
$query = mysqli_query($db,$sql);
$data = array();
$result = mysqli_query($db,$sql);
if (mysqli_num_rows($result) <>0) {
    # code...
    while ($row = mysqli_fetch_assoc($query)) {
        # code...
        $data[] = $row;
    }
    $n = mysqli_num_rows($result);
    echo "$n The loai<br>";
    foreach ($data as $value) {
        # code...
        print"<tr align='center'>";
        //print"<td>".$value['user_id']. "</td>";
        print"<td >".$value['id']. "</td><br>";
        print"<td>".$value['tentheloai']. "</td><br>";
        print"<td>".$value['img']. "</td><br>";
        print"<td><a href='getthongtin.php?id=".$value['id']."'>Thông tin</a></td><br>";
        print"</tr>";
    }
} else {echo "Khong co ket qua";}
?>

```

Hình 3.2.1.8. Hiển thị danh sách thể loại.

```

<?php
require("index.php");

class Theloai
{
    function Theloai($id, $video_id, $title, $channelTitle, $trinhbay, $theloai, $img)
    {
        # code...
        $this->id = $id;
        $this->video_id = $video_id;
        $this->title = $title;
        $this->channelTitle = $channelTitle;
        $this->trinhbay = $trinhbay;
        $this->theloai = $theloai;
        $this->img = $img;
    }
}

$arraylisttheloai = array();

$id = $_GET['id'];

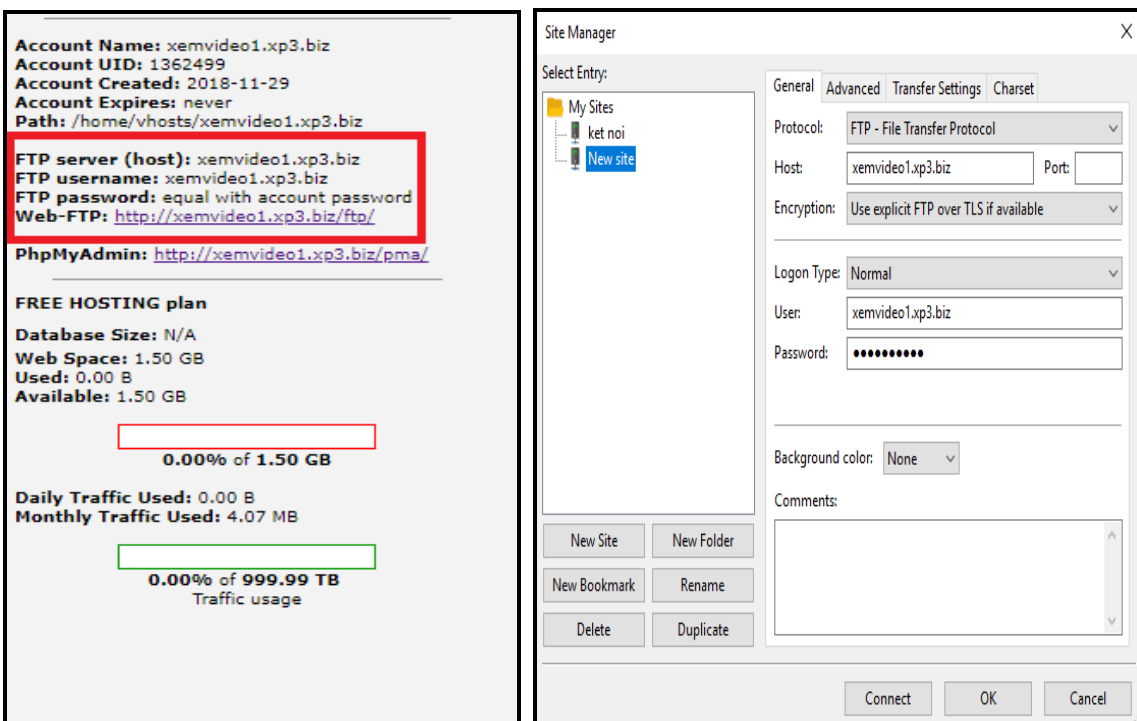
$query_theloai = "SELECT details_info.id, details_info.video_id, details_info.title, details_info.channelTitle,
details_info.trinhbay, details_info.theloai, details_info.img, theloai.id, theloai.tentheloai FROM `details_info`
INNER JOIN `theloai` ON details_info.theloai = theloai.id WHERE details_info.theloai = theloai.id && theloai = '$id'";
$data_theloai = mysqli_query($db, $query_theloai);
while ($row_theloai = mysqli_fetch_assoc($data_theloai)) {
    # code...
    array_push($arraylisttheloai, new Theloai($row_theloai['id'],$row_theloai['video_id'],$row_theloai['title'],
$row_theloai['channelTitle'],$row_theloai['trinhbay'],$row_theloai['tentheloai'],$row_theloai['img']));
}
echo json_encode($arraylisttheloai);

?>
    
```

Hình 3.2.1.9. Hiện thị video theo thể loại.

Bước 6: Tải và cài đặt FileZilla Client <https://filezilla-project.org/download.php>

Chuẩn bị thông tin và tạo kết nối thông qua FileZilla Client như hình 3.2.1.10 ví dụ dưới đây.

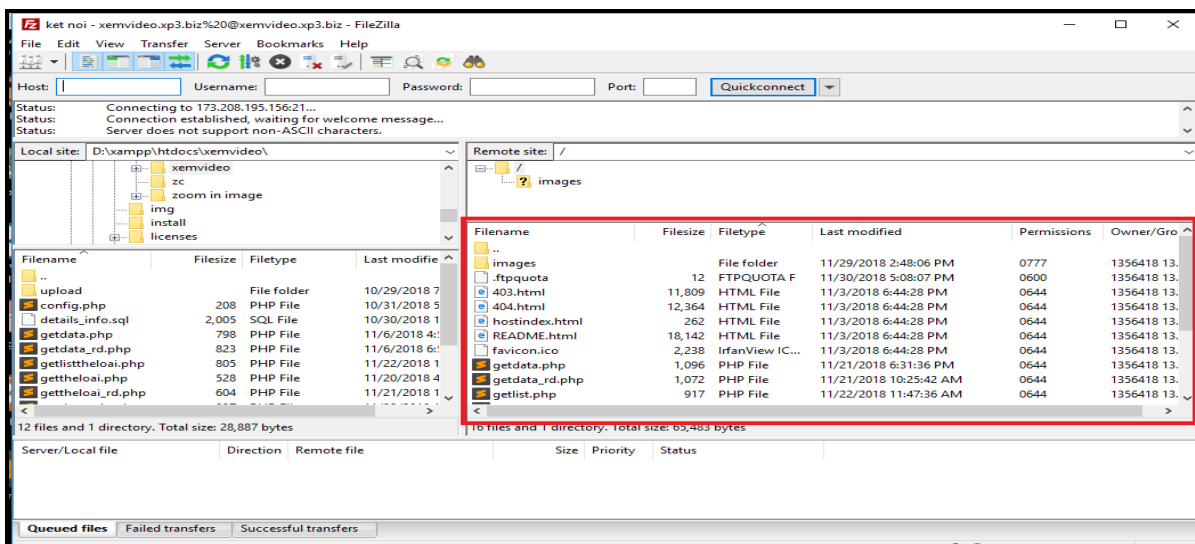


Hình 3.2.1.10. Kết nối FileZilla Client.

Sau khi tạo kết nối thành công, thực hiện kéo thả các file đã tạo bên trên vào khung bên phải như hình 3.2.1.11. Cuối cùng lấy đường dẫn chạy thử. Như vậy ta đã tạo dựng xong một Webservice.

VD: <http://xemvideo.xp3.biz/insert.php> (thêm mới dữ liệu vào CSDL).

<http://xemvideo.xp3.biz/getdata.php> (hiển thị toàn bộ dữ liệu trong giới hạn).



Hình 3.2.1.11. Kéo thả file vào khung bên phải.

3.2.2. Kỹ thuật kiểm tra cấp quyền kết nối internet thiết bị di động.

Trong các ứng dụng liên quan tới server, truy cập dữ liệu từ internet luôn đòi hỏi thiết bị luôn phải kết nối internet. Trong ứng dụng thực tế, ta cần phải thiết lập việc kiểm tra kết nối một cách liên tục. Và khi xử lý một vấn đề gì có liên quan tới kết nối, ta phải kiểm tra xem thiết bị có kết nối internet hay không.

Để kiểm tra kết nối internet thiết bị di động làm theo các bước:

- **Bước 1:** Cấp quyền truy cập internet trong AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
package="com.ctn.tuannghia.xemvideoonline">

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

- **Bước 2:** Xây dựng hàm kiểm tra kết nối trong MainActivity.java như hình 3.2.2.1 ví dụ dưới.

```
//check internet
public boolean isConnected(Context context) {
    //tạo đối tượng ConnectivityManager để kiểm tra thông tin mạng
    ConnectivityManager connectivityManager = (ConnectivityManager) context.getSystemService(Context.CONNECTIVITY_SERVICE);
    NetworkInfo netinfo = connectivityManager.getActiveNetworkInfo();

    if (netinfo != null && netinfo.isConnectedOrConnecting()) {
        //kiểm tra 2 loại kết nối mạng
        android.net.NetworkInfo wifi = connectivityManager.getNetworkInfo(ConnectivityManager.TYPE_WIFI);
        android.net.NetworkInfo mobile = connectivityManager.getNetworkInfo(ConnectivityManager.TYPE_MOBILE);

        if((mobile != null && mobile.isConnectedOrConnecting()) || (wifi != null && wifi.isConnectedOrConnecting())) return true;
    } else return false;
    } else
    return false;
}

//setup thông báo internet
public AlertDialog.Builder buildDialog(Context c) {

    AlertDialog.Builder builder = new AlertDialog.Builder(c);
    builder.setTitle("Không có kết nối Internet!");
    builder.setMessage("Vui lòng xem lại kết nối Internet. Nhấn OK để thoát!");

    builder.setPositiveButton( text: "Ok", (dialog, which) -> { finish(); });

    return builder;
}
}
```

Hình 3.2.2.1. Chức năng kiểm tra kết nối.

Cuối cùng sau đoạn mã : `setContentView(R.layout.activity_main);` trong MainActivity.java ta thêm đoạn mã như hình 3.2.2.2 dưới đây:

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    //
    if(!isConnected( context: MainActivity.this)) buildDialog( c: MainActivity.this).show();
    else {
        Toast.makeText( context: MainActivity.this, text: "Welcome", Toast.LENGTH_SHORT).show();
    }
}
```

Hình 3.2.2.2. Tạo lời chào khi có kết nối Internet.

3.2.3. Kỹ thuật lấy dữ liệu từ CSDL trên internet về thiết bị di động.

Trong các ứng dụng cần lưu trữ và hiển thị theo danh sách các thông tin đa phần ta dùng ListView. Để lấy được dữ liệu trên internet về thiết bị di động thì ta cần kết nối và lấy dữ liệu từ server thông qua Webservice có dữ liệu mã hóa dạng JSON. Để lấy và hiển thị dữ liệu ở cơ sở dữ liệu trên internet về thiết bị di động ta làm như sau:

Thêm 2 thư viện sau vào build.gradle (Module: app) và click Sync now.

implementation 'com.android.volley:volley:1.1.1'

implementation 'com.squareup.picasso:picasso:2.5.2'

- **Bước 1:** Để kiểm tra dữ liệu tất cả video trả về ta Toast thử. Xây dựng code trong MainActivity.java như hình 3.2.3.1 ví dụ dưới đây.

```

//url tat ca video
String urlGetData = "http://xemvideo.xp3.biz/getdata.php";

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    GetData(urlGetData);
}

// get tat ca video
private void GetData(String url) {
    RequestQueue requestQueue = Volley.newRequestQueue( context: this);
    JSONArrayRequest jsonArrayRequest = new JSONArrayRequest(Request.Method.GET, url, jsonArrayRequest: null, new Response.Listener() {
        @Override
        public void onResponse(JSONArray response) {
            Toast.makeText( context: MainActivity.this, response.toString(), Toast.LENGTH_SHORT).show();
        }
    },
        new Response.ErrorListener() {
            @Override
            public void onErrorResponse(VolleyError error) {}
        }
    );
    requestQueue.add(jsonArrayRequest);
}

```

Hình 3.2.3.1. Đọc thử dữ liệu tất cả video trả về.

- **Bước 2:** Sau khi Toast thành công. Tạo class có tên XemVideo. Tạo các thuộc tính tương ứng với bảng trong CSDL như hình 3.2.3.2 ví dụ dưới.

```

public class XemVideo {
    private int ID;
    private String Video_id;
    private String Title;
    private String ChannelTitle;
    private String Trinhbay;
    private String Theloai;
    private String Img;

    public XemVideo(int id, String video_id, String title, String channelTitle, String trinhbay, String theloai, String img) {
        ID = id;
        Video_id = video_id;
        Title = title;
        ChannelTitle = channelTitle;
        Trinhbay = trinhbay;
        Theloai = theloai;
        Img = img;
    }

    public String getVideo_id() { return Video_id; }

    public void setVideo_id(String video_id) { Video_id = video_id; }

    public String getTitle() { return Title; }

    public void setTitle(String title) { Title = title; }

    public String getChannelTitle() { return ChannelTitle; }

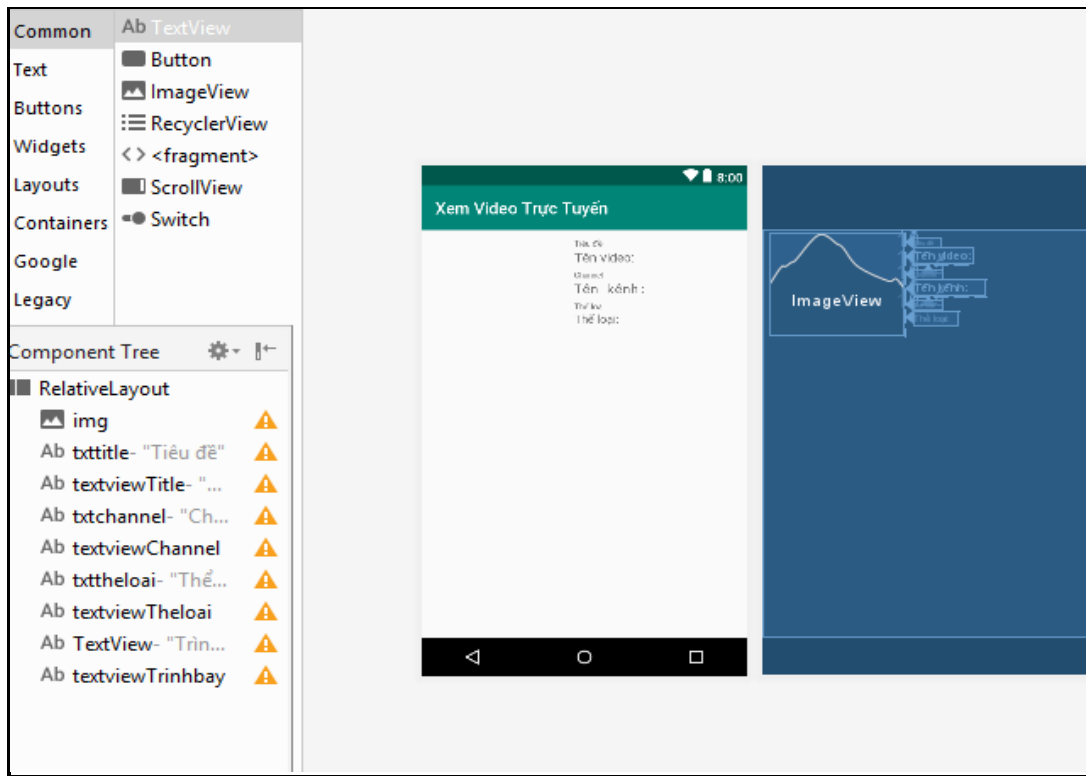
    public void setChannelTitle(String channelTitle) { ChannelTitle = channelTitle; }

    public String getTrinhbay() { return Trinhbay; }
}

```

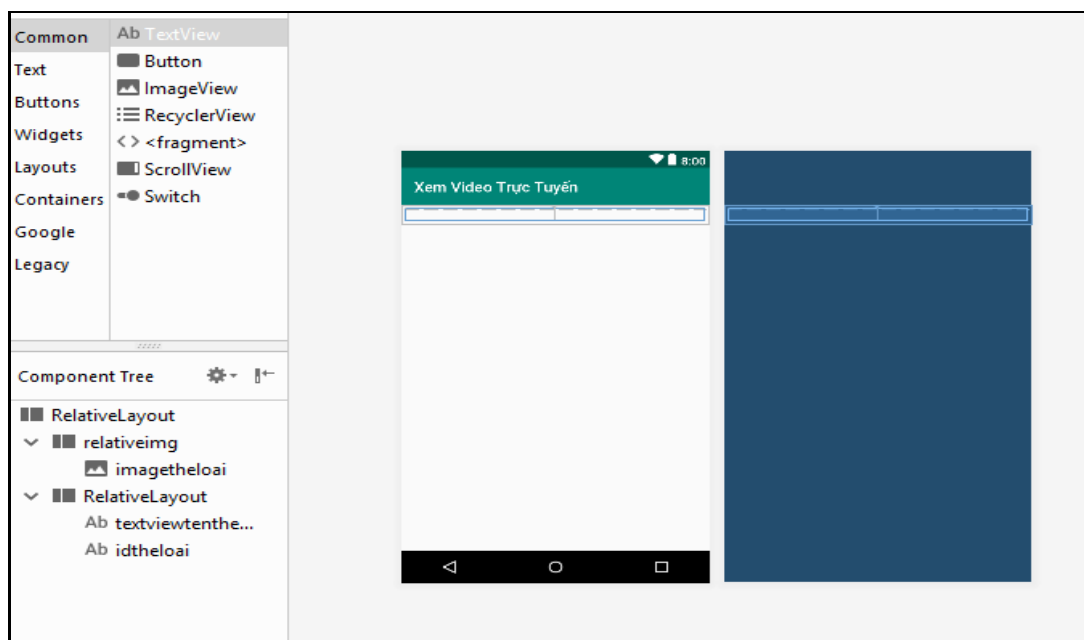
Hình 3.2.3.2. Tạo class XemVideo.

- **Bước 3:** Tạo giao diện file row_listview.xml tương ứng cho 1 video như hình 3.2.3.3 ví dụ dưới đây.



Hình 3.2.3.3. Tạo giao diện hiển thị video.

Tạo giao diện file row_gridview.xml tương ứng cho 1 thể loại như hình 3.2.3.4 ví dụ dưới.



Hình 3.2.3.4. Tạo giao diện hiển thị thể loại.

- **Bước 4:** Tạo class XemvideoAdapter như hình 3.2.3.5 ví dụ dưới.

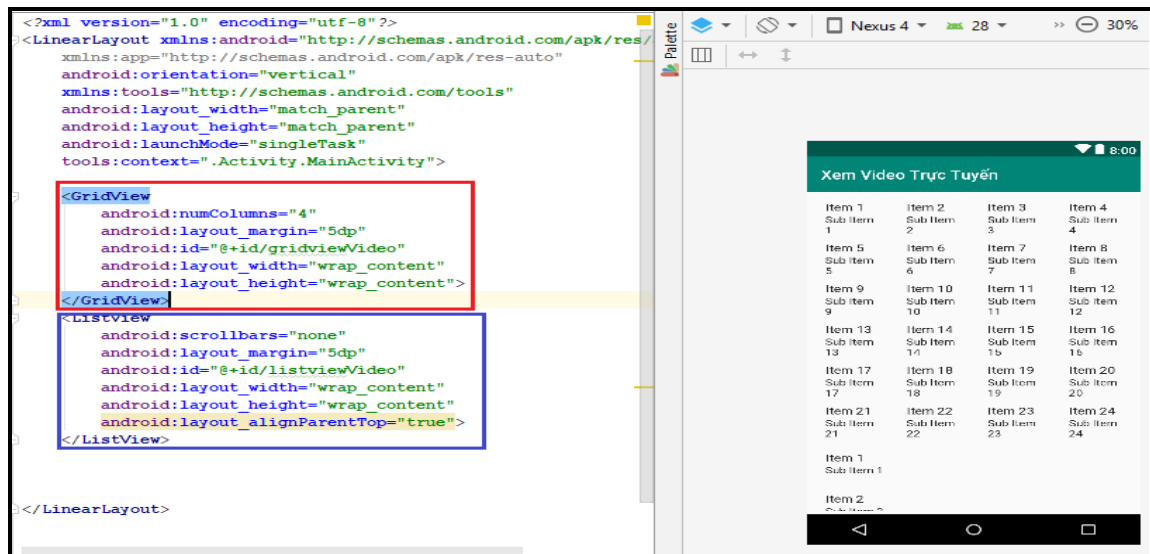
Một Adapter (bộ chuyển đổi) đóng vai trò như một liên kết giữa một tập hợp dữ liệu và một Adapter View, một đối tượng của một lớp thừa kế lớp trừu tượng AdapterView. Tập hợp dữ liệu có thể là bất cứ điều gì mà trình bày dữ liệu một cách có cấu trúc. Mảng, các đối tượng List và các đối tượng Cursor thường sử dụng bộ dữ liệu.

Một Adapter có trách nhiệm lấy dữ liệu từ bộ dữ liệu và tạo ra các đối tượng View dựa trên dữ liệu đó. Các đối tượng View được tạo ra sau đó được sử dụng để gắn lên bất kỳ Adapter View mà ràng buộc với Adapter.

```
public class XemvideoAdapter extends BaseAdapter {  
  
    private Context context;  
    private int layout;  
    private List<XemVideo> xemVideoList;  
    private ArrayList<XemVideo> arrayList;  
  
    public XemvideoAdapter(Context context, int layout, List<XemVideo> xemVideoList) {  
        this.context = context;  
        this.layout = layout;  
        this.xemVideoList = xemVideoList;  
        this.arrayList = new ArrayList<XemVideo>();  
        this.arrayList.addAll(xemVideoList);  
    }  
  
    @Override  
    public int getCount() { return xemVideoList.size(); }  
  
    @Override  
    public Object getItem(int i) { return null; }  
  
    @Override  
    public long getItemId(int i) { return 0; }  
  
    private class ViewHodler{  
        ImageView img;  
        TextView txtTitle;  
        TextView txtChannelTitle;  
        TextView txtTrinhbay;  
        TextView txtTheloi;  
    }  
  
    @Override  
    public View getView(int i, View view, ViewGroup viewGroup) {  
        ViewHodler holder;  
        if (view == null){  
            //anh xa  
            holder = new ViewHodler();  
            LayoutInflater inflater = (LayoutInflater) context.getSystemService(context.LAYOUT_INFLATER_SERVICE);  
            view = inflater.inflate(layout, root = null);  
            holder.img = (ImageView) view.findViewById(R.id.img);  
            holder.txtTitle = (TextView) view.findViewById(R.id.textviewTitle);  
            holder.txtChannelTitle = (TextView) view.findViewById(R.id.textviewChannel);  
            holder.txtTrinhbay = (TextView) view.findViewById(R.id.textviewTrinhbay);  
            holder.txtTheloi = (TextView) view.findViewById(R.id.textviewTheloi);  
            view.setTag(holder);  
        }  
        else{  
            holder = (ViewHodler) view.getTag();  
        }  
  
        final XemVideo xemVideo = xemVideoList.get(i);  
        //gan du lieu  
        holder.txtTitle.setText(xemVideo.getTitle());  
        holder.txtChannelTitle.setText(xemVideo.getChannelTitle());  
        holder.txtTrinhbay.setText(xemVideo.getTrinhbay());  
        holder.txtTheloi.setText(xemVideo.getTheloi());  
        Picasso.with(context).load( path: "http://xemvideo.xp3.biz/images/" + xemVideo.getImg()).into(holder.img);  
        ////  
  
        return view;  
    }  
}
```

Hình 3.2.3.5. Tạo class XemvideoAdapter.

- **Bước 5:** Tạo giao diện file activity_main.xml như hình 3.2.3.6 ví dụ dưới.
 - GridView dùng để hiển thị danh sách các thể loại.
 - ListView dùng để hiển thị danh sách các video.



Hình 3.2.3.6. Tạo giao diện cho trang chủ.

- **Bước 6:** Xây dựng code hoàn chỉnh trong MainActivity.java để hiển thị thông tin video và thể loại ra gridview, listview như các hình từ hình 3.2.3.7 đến 3.2.3.9 ví dụ dưới.

```

//url tat ca video
String urlGetData = "http://xemvideo.xp3.biz/getdata.php";
//url tat ca the loai
String urlGetTheLoai = "http://xemvideo.xp3.biz/gettheloai.php";

ListView lvvideo;
GridView grtheloai;
ArrayList<Gridview> arrayTheLoai;
Gridview_Adapter adapter1;
ArrayList<XemVideo> arrayXemVideo;
XemvideoAdapter adapter;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    //anh xa
    lvvideo = (ListView) findViewById(R.id.listviewVideo);
    arrayXemVideo = new ArrayList<>();
    grtheloai = (GridView) findViewById(R.id.gridviewVideo);
    arrayTheLoai = new ArrayList<>();

    GetData(urlGetData);
    GetTheLoai(urlGetTheLoai);
}

// get tat ca the loai
private void GetTheLoai(String urlGetTheLoai) {

```

Hình 3.2.3.7. Khai báo biến và ánh xạ.

```

// get tat ca the loai
private void GetTheLoai(String urlGetTheLoai) {
    RequestQueue requestQueue = Volley.newRequestQueue( context: this);
    JSONArrayRequest jsonArrayRequest = new JSONArrayRequest(Request.Method.GET, urlGetTheLoai, jsonRequest: null, onResp
        for (int i=0; i<response.length(); i++){
            try {
                JSONObject object = response.getJSONObject(i);
                arrayTheLoai.add(new Gridview(
                    object.getString( name: "id"),
                    object.getString( name: "tentheLoai"),
                    object.getString( name: "img")
                ));
            } catch (JSONException e) {
                e.printStackTrace();
            }
        }
        //adapter.notifyDataSetChanged();
        adapter1 = new Gridview_Adapter( context: MainActivity.this, R.layout.row_gridview, arrayTheLoai);
        grtheLoai.setAdapter(adapter1);
    },
    new Response.ErrorListener() {
        @Override
        public void onErrorResponse(VolleyError error) {

        }
    }
);
requestQueue.add(jsonArrayRequest);
}

```

Hình 3.2.3.8. Tạo hàm GettheLoai lấy dữ liệu thể loại.

```

// get tat ca video
private void GetData(String url) {
    RequestQueue requestQueue = Volley.newRequestQueue( context: this);
    JSONArrayRequest jsonArrayRequest = new JSONArrayRequest(Request.Method.GET, url, jsonRequest: null, onResponse(respon
        for (int i=0; i<response.length(); i++){
            try {
                JSONObject object = response.getJSONObject(i);
                arrayXemVideo.add(new XemVideo(
                    object.getInt( name: "id"),
                    object.getString( name: "video_id"),
                    object.getString( name: "title"),
                    object.getString( name: "channelTitle"),
                    object.getString( name: "trinhbay"),
                    object.getString( name: "theLoai"),
                    object.getString( name: "img")
                ));
            } catch (JSONException e) {
                e.printStackTrace();
            }
        }
        //adapter.notifyDataSetChanged();
        adapter = new XemvideoAdapter( context: MainActivity.this, R.layout.row_listvideo, arrayXemVideo);
        lvvideo.setAdapter(adapter);
    },
    new Response.ErrorListener() {
        @Override
        public void onErrorResponse(VolleyError error) {}
    }
);
requestQueue.add(jsonArrayRequest);
}

```

Hình 3.2.3.9. Tạo hàm GetData lấy toàn bộ dữ liệu video.

3.2.4. Bắt và xử lý sự kiện click.

- **Bước 1:** Xây dựng hàm event() trong file MainActivity.java như hình 3.2.4.1 ví dụ dưới. Bắt và xử lý sự kiện khi click vào mỗi item video. Tạo 2 Activity mới: PlayVideoActivity và TheLoaiActivity.

```

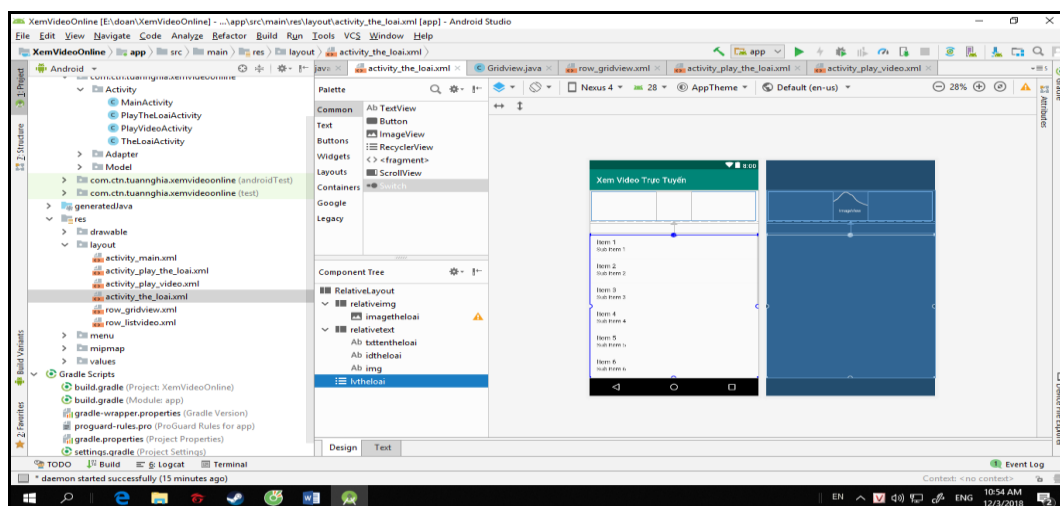
event();
}

//su kiện click item video
private void event(){
// su kiện click xem video
lvvideo.setOnItemClickListener(new AdapterView.OnItemClickListener() {
@Override
public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {
Intent intent = new Intent( packageContext: MainActivity.this, PlayVideoActivity.class);
intent.putExtra( name: "idvideo",arrayXemVideo.get(i).getVideo_id());
intent.putExtra( name: "title", arrayXemVideo.get(i).getTitle());
intent.putExtra( name: "channel", arrayXemVideo.get(i).getChannelTitle());
intent.putExtra( name: "trinhbay", arrayXemVideo.get(i).getTrinhbay());
intent.putExtra( name: "theloai", arrayXemVideo.get(i).getTheLoai());
startActivity(intent);
}
});
//su kiện click xem the loai
grtheloai.setOnItemClickListener(new AdapterView.OnItemClickListener() {
@Override
public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
Intent intent = new Intent( packageContext: MainActivity.this, TheLoaiActivity.class);
intent.putExtra( name: "idtheloai",arrayTheLoai.get(position).getID());
intent.putExtra( name: "tentheLoai", arrayTheLoai.get(position).getTentheLoai());
intent.putExtra( name: "img", arrayTheLoai.get(position).getImg());
startActivity(intent);
}
});
}
}

```

Hình 3.2.4.1. Xây dựng hàm event.

- **Bước 2:** Thiết kế giao diện file activity_the_loai.xml như hình 3.2.4.2 ví dụ dưới. Để hiển thị chi tiết 1 thể loại bất kỳ khi click sau khi đã bắt sự kiện.



Hình 3.2.4.2. Tạo giao diện hiển thị các video thể loại.

- **Bước 3:** Tạo hàm getIntent() để xử lý sự kiện click item thẻ loại trong file TheLoaiActivity.java như hình 3.2.4.3 ví dụ dưới.

```

ListView lvtheloai;
TextView txttenthelloai, imgtl;
ImageView imgtheloai;

String tenthelloai="";
String img="";
String id="";
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_the_loai);

    anhxa();
    getIntent();
}

private void getIntent() {
    //get id, tenthelloai, images
    final Intent intent = getIntent();
    id = intent.getStringExtra( name: "idtheloai");
    txttenthelloai.setText(tenthelloai = intent.getStringExtra( name: "tenthelloai"));
    imgtl.setText(img = intent.getStringExtra( name: "img"));
    Picasso.with(TheLoaiActivity.this).load( path: "http://xemvideo.xp3.biz/images/"+img).into(TheLoaiActivity.this.imgthel
}

private void anhxa() {
    lvtheloai = (ListView) findViewById(R.id.lvtheloai);
    txttenthelloai = (TextView) findViewById(R.id.txttenthelloai);
    imgtl = (TextView) findViewById(R.id.img);
    imgtheloai = (ImageView) findViewById(R.id.imgtheloai);
}

```

Hình 3.2.4.3. Xây dựng hàm getIntent.

- **Bước 4:** Tại file TheLoaiActivity.java tạo hàm init() event() GetData(). Để hiển thị danh sách các video có thẻ loại tương ứng và bắt sự kiện mỗi item video. Tạo file PlayTheLoaiActivity để phát video theo thẻ loại như các hình từ hình 3.2.4.4 đến hình 3.2.4.6 ví dụ dưới.

```

public class TheLoaiActivity extends AppCompatActivity {
    // khai báo biến
    ArrayList<XemVideo> arrayXemVideo;
    XemVideoAdapter adapter;

    ListView lvtheloai;
    TextView txttenthelloai, imgtl;
    ImageView imgtheloai;

    String tenthelloai="";
    String img="";
    String id="";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_the_loai);

        anhxa();
        init();
        getIntent();
        event();
        String urlGetData ="http://xemvideo.xp3.biz/getthongtin.php?id="+id;
        GetData(urlGetData);
    }

    private void getIntent() {
        //get id, tenthelloai, images
        final Intent intent = getIntent();
        id = intent.getStringExtra( name: "idtheloai");
        txttenthelloai.setText(tenthelloai = intent.getStringExtra( name: "tenthelloai"));
        imgtl.setText(img = intent.getStringExtra( name: "img"));
    }
}

```

Hình 3.2.4.4. Khai báo biến và gọi hàm.


```

        Picasso.with(TheLoaiActivity.this).load( path: "http://xemvideo.xp3.biz/images/"+img).into(TheLoaiActivity.this.imgthe
    }

    private void event() {
        //su kiện click item video
        lvtheloai.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view, int position, long l) {
                Intent intent = new Intent( packageContext: TheLoaiActivity.this, PlayTheLoaiActivity.class);
                intent.putExtra( name: "idvideo", arrayXemVideo.get(position).getVideo_id());
                intent.putExtra( name: "title", arrayXemVideo.get(position).getTitle());
                intent.putExtra( name: "channel", arrayXemVideo.get(position).getChannelTitle());
                intent.putExtra( name: "trinhbay", arrayXemVideo.get(position).getTrinhbay());
                intent.putExtra( name: "theloai", arrayXemVideo.get(position).getTheLoai());
                intent.putExtra( name: "idtheloai", id);
                startActivity(intent);
            }
        });
    }

    private void init() {
        arrayXemVideo = new ArrayList<>();
        adapter = new XemvideoAdapter( context: TheLoaiActivity.this, R.layout.row_listvideo, arrayXemVideo);
        lvtheloai.setAdapter(adapter);
    }

    private void anhxa() {
        lvtheloai = (ListView) findViewById(R.id.lvtheloai);
        txttentheLoai = (TextView) findViewById(R.id.txttentheLoai);
        imgt1 = (TextView) findViewById(R.id.img);
        imgtheloai = (ImageView) findViewById(R.id.imagetheLoai);
    }
}

```

Hình 3.2.4.5. Tạo hàm event, init, anhxa.

```

// get data video
private void GetData(String url) {
    RequestQueue requestQueue = Volley.newRequestQueue( context: this);
    JSONArrayRequest jsonArrayRequest = new JSONArrayRequest(Request.Method.GET, url, jsonRequest null, (response) ->
        for (int i=0; i<response.length(); i++){
            try {
                JSONObject object = response.getJSONObject(i);
                arrayXemVideo.add(new XemVideo(
                    object.getInt( name: "id"),
                    object.getString( name: "video_id"),
                    object.getString( name: "title"),
                    object.getString( name: "channelTitle"),
                    object.getString( name: "trinhbay"),
                    object.getString( name: "theloai"),
                    object.getString( name: "img")
                ));
            } catch (JSONException e) {
                e.printStackTrace();
            }
        }
        //adapter.notifyDataSetChanged();
        adapter = new XemvideoAdapter( context: TheLoaiActivity.this, R.layout.row_listvideo, arrayXemVideo);
        lvtheloai.setAdapter(adapter);
    },
    new Response.ErrorListener() {
        @Override
        public void onErrorResponse(VolleyError error) {
        }
    }
);
requestQueue.add(jsonArrayRequest);
}

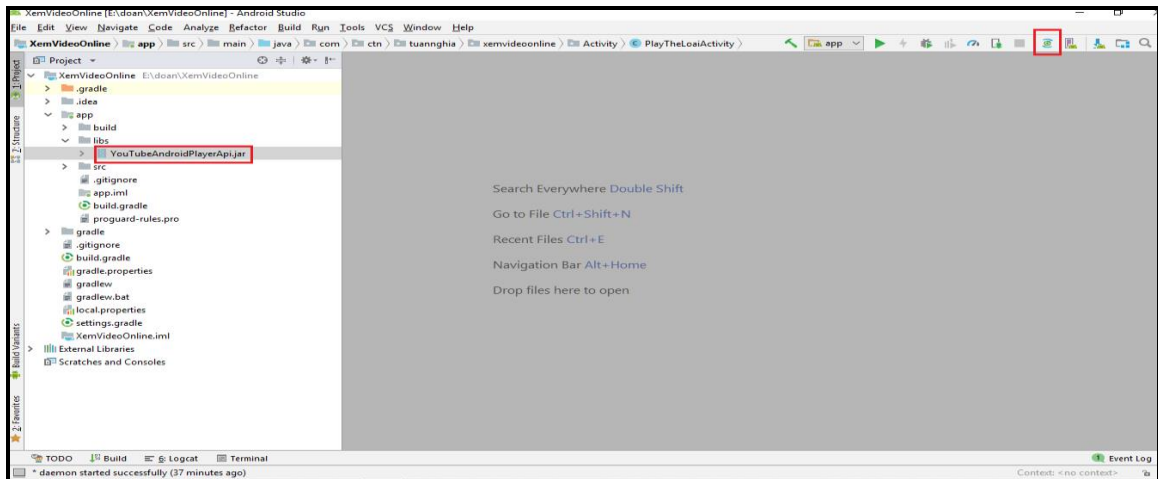
```

Hình 3.2.4.6. Tạo hàm GetData lấy toàn bộ dữ liệu video.

- **Bước 5:** Xử lý sự kiện click item mỗi video. Để xem được video YouTube trên máy ảo Genymotion cần cài đặt:

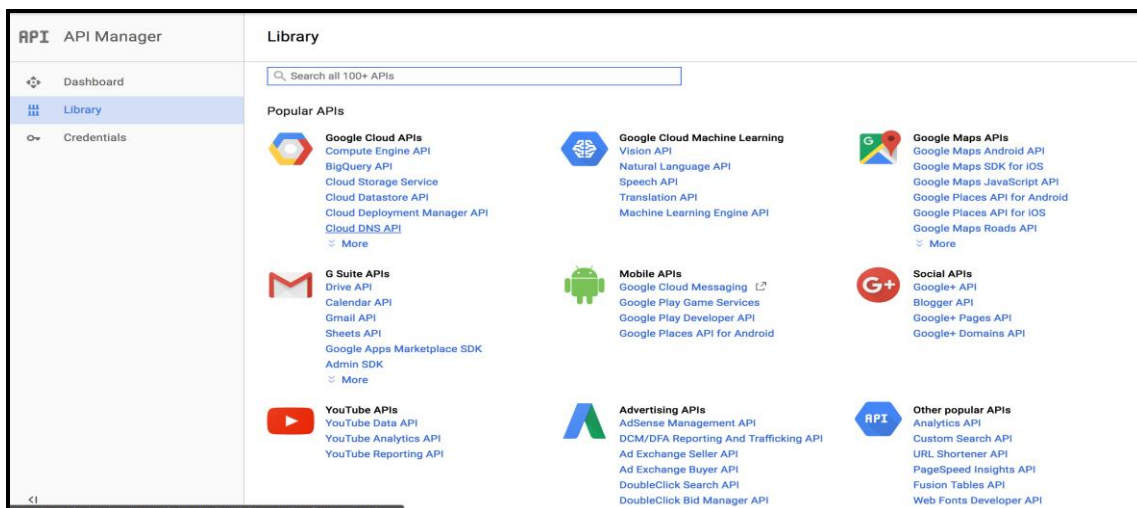
- Tải thư viện

<https://developers.google.com/youtube/android/player/downloads/> giải nén và copy file YouTubeAndroidPlayerApi.jar như hình dưới. Sau đó nhấn Sync now như hình 3.2.4.7 ví dụ dưới.

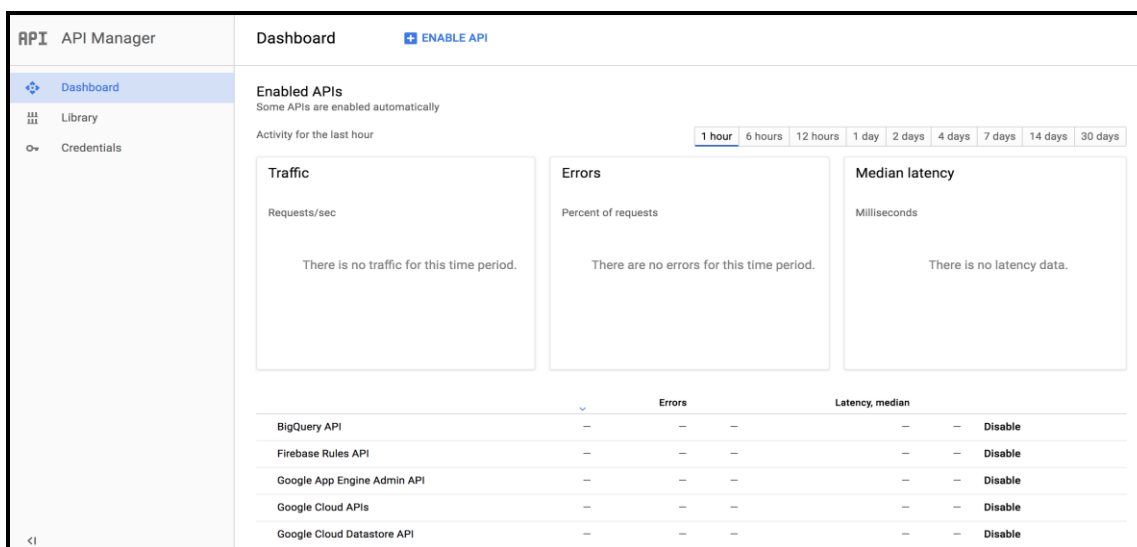


Hình 3.2.4.7. Sao chép file YouTubeAndroidPlayerApi.jar

- Đăng ký YouTube API. Để có thể sử dụng các dịch vụ của Google thì điều cần thiết đầu tiên là tạo 1 account Google. Sau khi đã có tài khoản của Google, truy cập <https://console.developers.google.com/apis?project=play-video-online-220314>
- Với API dữ liệu YouTube, bạn có thể thêm nhiều tính năng YouTube vào ứng dụng của mình. Sử dụng API để tải lên Video, quản lý danh sách phát và đăng ký, cập nhập cài đặt kênh và hơn thế nữa.
- Sử dụng API để tìm kiếm video phù hợp với các thuật ngữ tìm kiếm, chủ đề, địa điểm, ngày xuất bản cụ thể và nhiều hơn nữa. search.list Phương pháp API cũng hỗ trợ tìm kiếm danh sách phát và kênh
- Để có thể sử dụng các dịch vụ của Google, ta cần phải tạo một project để có thể sử dụng tại <https://console.developers.google.com/projectcreate>

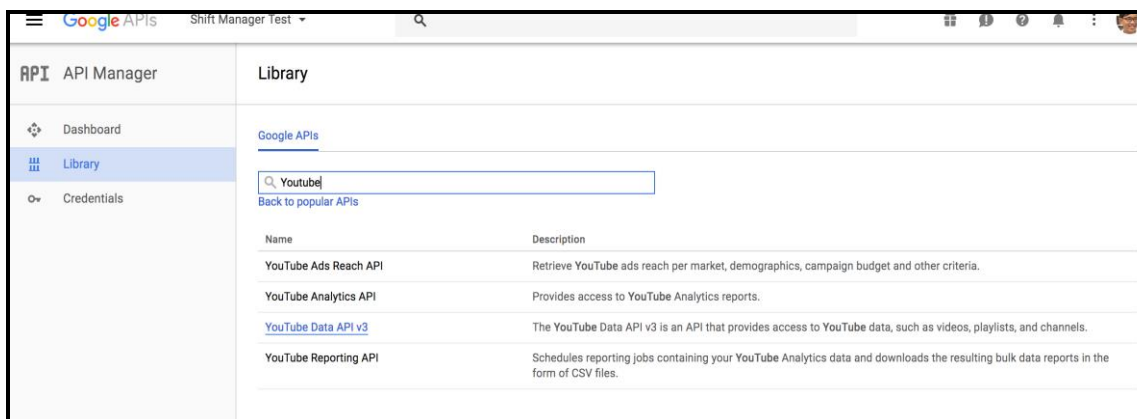


Hình 3.2.4.8. Giao diện Library API manager.



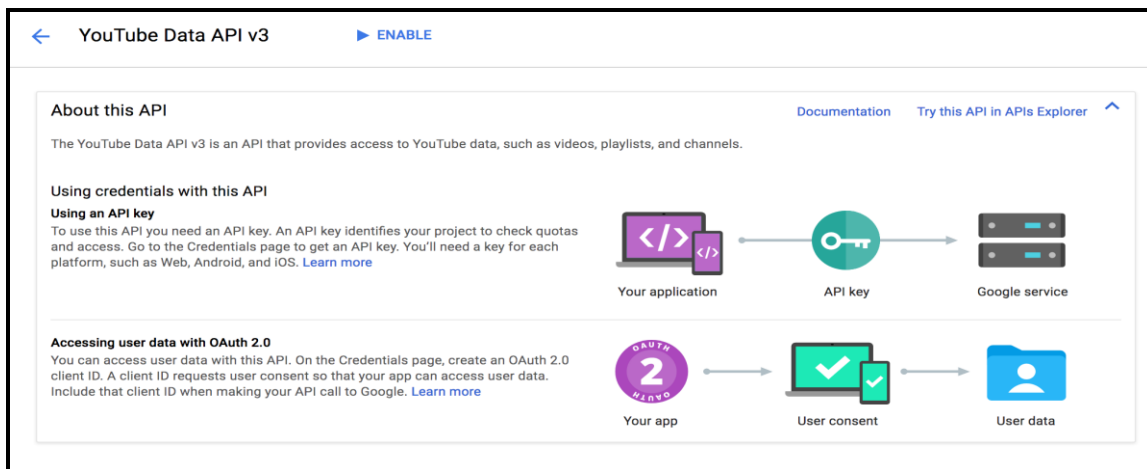
Hình 3.2.4.9. Giao diện Dashboard API manager.

- Trở về trang quản lý API, ta gõ vào thanh search cụm từ Youtube, sau đó chọn Youtube Data API v3 như hình 3.2.4.10.



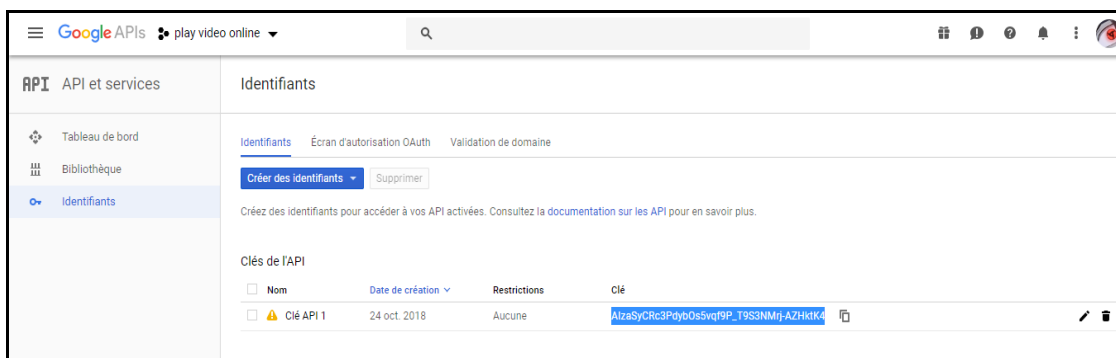
Hình 3.2.4.10. Tìm kiếm dịch vụ tại trang quản lý.

- Sau khi chọn API, ta bấm *Enable* để kích hoạt API như hình 3.2.4.11.



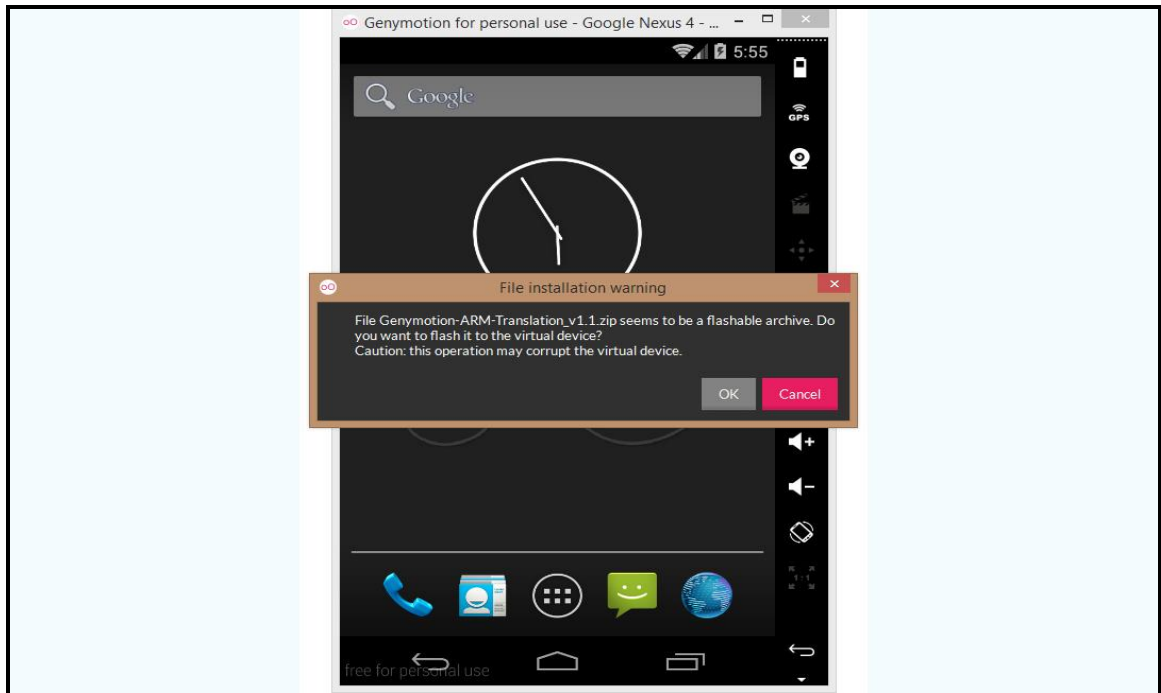
Hình 3.2.4.11. Kích hoạt dịch vụ Youtube Data API v3.

- Khởi tạo API Key: Sau khi kích hoạt thành công API, chúng ta vào mục **Credentials** -> **Create Credentials** -> **API Key**. Sau khi khởi tạo thành công, chúng ta đã có thể sử dụng key vừa tạo như hình 3.2.3.12 ví dụ dưới.



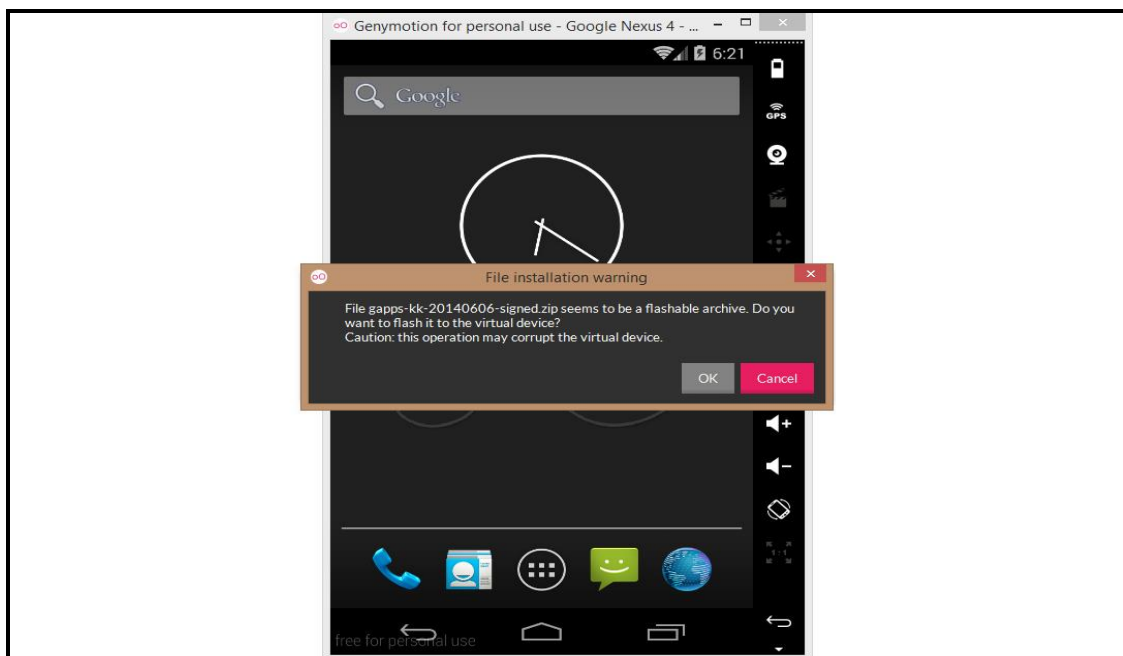
Hình 3.2.4.12. Khởi tạo API Key.

- Để phát thử video YouTube trên máy ảo ta cần xác thực dịch vụ Google. Truy cập <https://androidfilehost.com/?fid=23252070760974384> tải file rồi kéo và thả tệp vào thiết bị ảo Genymotion đã mở, sẽ có hộp thoại xuất hiện như hình 3.2.4.13. Nhấn ok để hoàn tất. Sau đó tắt máy ảo và khởi động lại[8].



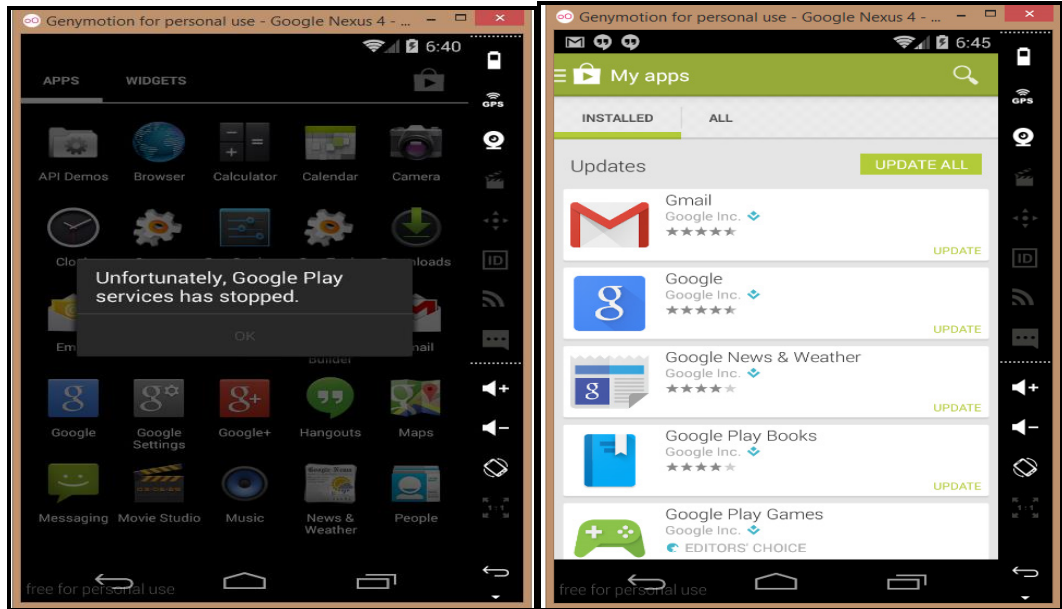
Hình 3.2.4.13. Kéo thẻ file zip vào máy ảo.

Tiếp theo tải file zip theo phiên bản Android của thiết bị ảo: Android 5.0.x , Android 4.4.x , Android 4.3.x , Android 4.2.x , Android 4.1.x , Android 4.0.x , Android 2.3.3 [10]. Sau đó kéo thả vào thiết bị ảo và thực hiện như bước trên.



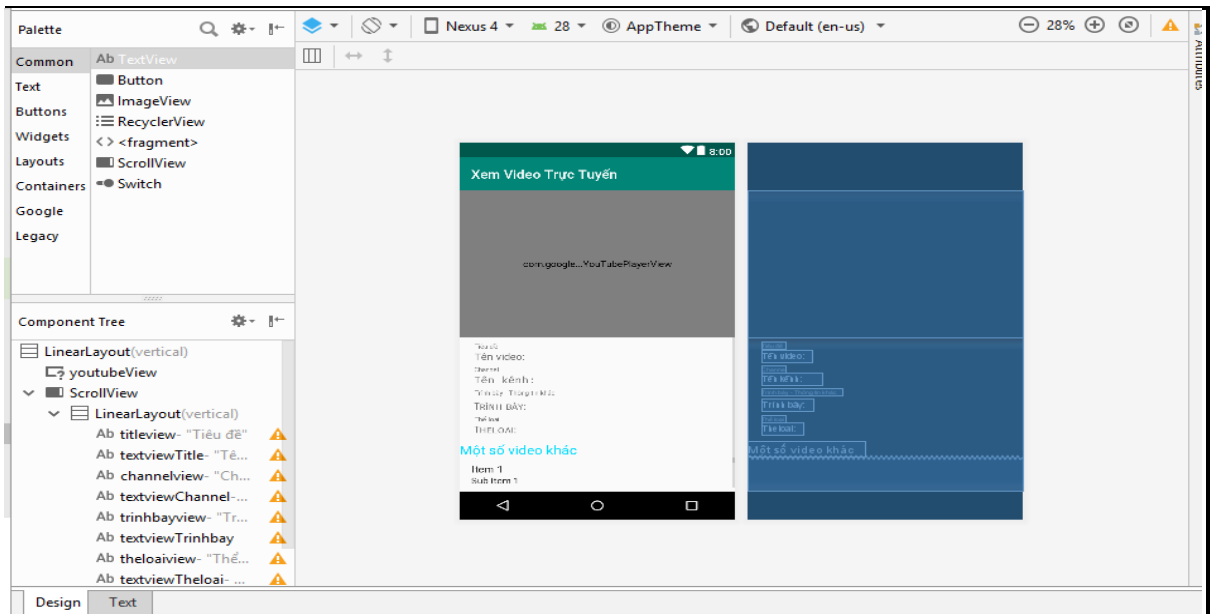
Hình 3.2.4.14. Kéo thẻ file zip theo phiên bản Android.

Sau khi thiết bị ảo được khởi động, sẽ thấy rằng các dịch vụ của Google Play đã dừng sự cố sẽ tiếp tục xuất hiện. Chỉ cần đăng nhập vào Google Play Store và cập nhật tất cả các ứng dụng đã cài đặt để hoàn thành như hình 3.2.4.15 ví dụ dưới.



Hình 3.2.4.15. Đăng nhập và cập nhật các ứng dụng.

Sau hoàn tất cả các bước trên. Để phát video bất kỳ khi click. Tạo giao diện activity_play_video.xml như hình 3.2.4.16 ví dụ dưới đây.



Hình 3.2.4.16. Tạo giao diện phát video bất kỳ.

Hoàn thành code trong file PlayVideoActivity.java như các hình ví dụ dưới để hoàn thành chức năng phát video như các hình từ hình 3.2.4.17 đến hình 3.2.4.21 ví dụ dưới.

```
public class PlayVideoActivity extends YouTubeBaseActivity implements YouTubePlayer.OnInitializedListener {

    // khai báo biến
    String urlGetData ="http://xemvideo.xp3.biz/getdata_rd.php";

    ListView lvviewplay;
    ArrayList<XemVideo> arrayXemVideo;
    XemvideoAdapter adapter;

    TextView txtTitle, txtChannel, txttrinhbay, txtTheloai;

    String id = "", title="", channel="", trinhbay="", theloai="";
    private YouTubePlayerView youtubePlayerView;

    int REQUEST_VIDEO = 123;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_play_video);

        anhxa();
        init();
        getIntent();
        event();
        GetData(urlGetData);
    }
}
```

Hình 3.2.4.17. Khai báo biến và gọi hàm.

```
private void event() {
    //su kien click item video
    lvviewplay.setOnItemClickListener((adapterView, view, i, l) -> {
        Intent intent = new Intent( packageContext: PlayVideoActivity.this, PlayVideoActivity.class);
        intent.putExtra( name: "idvideo", arrayXemVideo.get(i).getVideo_id());
        intent.putExtra( name: "title", arrayXemVideo.get(i).getTitle());
        intent.putExtra( name: "channel", arrayXemVideo.get(i).getChannelTitle());
        intent.putExtra( name: "trinhbay", arrayXemVideo.get(i).getTrinhbay());
        intent.putExtra( name: "theloai", arrayXemVideo.get(i).getTheloai());
        startActivity(intent);
    });
}

private void getIntent() {
    final Intent intent = getIntent();
    id = intent.getStringExtra( name: "idvideo");

    txtTitle.setText(title = intent.getStringExtra( name: "title"));
    txtChannel.setText(channel = intent.getStringExtra( name: "channel"));
    txttrinhbay.setText(trinhbay = intent.getStringExtra( name: "trinhbay"));
    txtTheloai.setText(theloai = intent.getStringExtra( name: "theloai"));
}
}
```

Hình 3.2.4.18. Các hàm event và getIntent.

```

private void init() {
    arrayXemVideo = new ArrayList<>();

    adapter = new XemvideoAdapter( context: PlayVideoActivity.this, R.layout.row_listvideo, arrayXemVideo);
    lvviewplay.setAdapter(adapter);
    youtubePlayerView.initialize(MainActivity.API_KEY, onInitializedListener: this);
}

private void anhxa() {
    lvviewplay = (ListView) findViewById(R.id.lvviewplay);
    txtTitle = (TextView) findViewById(R.id.textviewTitle);
    txtChannel = (TextView) findViewById(R.id.textviewChannel);
    txttrinhbay = (TextView) findViewById(R.id.textviewTrinhbay);
    txtTheLoai = (TextView) findViewById(R.id.textviewTheLoai);
    youtubePlayerView = (YouTubePlayerView) findViewById(R.id.youtubeView);
}

```

Hình 3.2.4.19. Các hàm init và anhxa.

```

private void GetData(String url) {
    RequestQueue requestQueue = Volley.newRequestQueue( context: this);
    JSONArrayRequest jsonArrayRequest = new JSONArrayRequest( Request.Method.GET, url, jsonRequest: null, (response) -> {
        for (int i = 0; i < response.length(); i++) {
            try {
                JSONObject object = response.getJSONObject(i);
                arrayXemVideo.add(new XemVideo(
                    object.getInt( name: "id"),
                    object.getString( name: "video_id"),
                    object.getString( name: "title"),
                    object.getString( name: "channelTitle"),
                    object.getString( name: "trinhbay"),
                    object.getString( name: "theloai"),
                    object.getString( name: "img")
                ));
            } catch (JSONException e) {
                e.printStackTrace();
            }
        }
        adapter.notifyDataSetChanged();
    },
    new Response.ErrorListener() {
        @Override
        public void onErrorResponse(VolleyError error) {
        }
    }
);
requestQueue.add(jsonArrayRequest);
}

```

Hình 3.2.4.20. Hàm GetData lấy dữ liệu toàn bộ video.

```

@Override
public void onInitializationSuccess(YouTubePlayer.Provider provider, YouTubePlayer youtubePlayer, boolean b) {
    youtubePlayer.loadVideo(id);
}

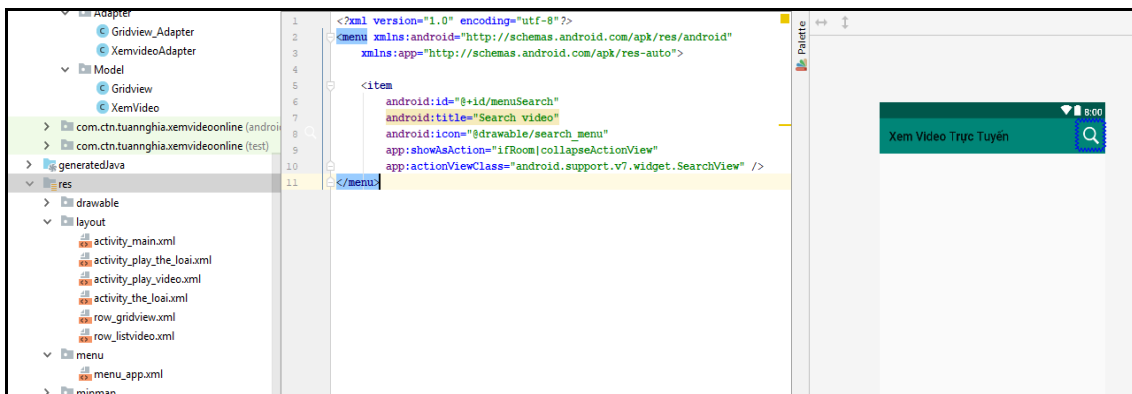
@Override
public void onInitializationFailure(YouTubePlayer.Provider provider, YouTubeInitializationResult youtubeInitializationResult) {
    if (youtubeInitializationResult.isUserRecoverableError()) {
        youtubeInitializationResult.getErrorDialog( activity: PlayVideoActivity.this, REQUEST_VIDEO);
    } else {
        Toast.makeText( context: this, text: "Error!!", Toast.LENGTH_SHORT).show();
    }
}

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    if (requestCode == REQUEST_VIDEO) {
        youtubePlayerView.initialize(MainActivity.API_KEY, onInitializedListener: PlayVideoActivity.this);
    }
    super.onActivityResult(requestCode, resultCode, data);
}

```

Hình 3.2.4.21. Tạo hàm initialization successful để phát video.

- **Bước 6:** Nhúng SearchView vào ActionBar để xây dựng chức năng tìm kiếm video. Tạo 1 Directory có tên menu trong res có giao diện như hình 3.2.4.22 ví dụ dưới.



Hình 3.2.4.22. Nhúng SearchView vào ActionBar.

- **Bước 7:** Khởi tạo menu search tại file MainActivity.java. Xây dựng chức năng tìm kiếm trong file XemvideoAdapter.java như các hình từ hình 3.2.4.23 đến hình 3.2.4.24 ví dụ dưới.

```
// khởi tạo menu searchview
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu_app, menu);

    SearchView searchView = (SearchView) menu.findItem(R.id.menuSearch).getActionView();

    searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {
        @Override
        public boolean onQueryTextSubmit(String query) { return false; }

        @Override
        public boolean onQueryTextChange(String newText) {
            adapter.filter(newText.trim());
            return false;
        }
    });
    return super.onCreateOptionsMenu(menu);
}
```

Hình 3.2.4.23. Khởi tạo menu search.

```
// filter chức năng searchview
public void filter(String charText){
    charText = charText.toLowerCase(Locale.getDefault());
    xemVideoList.clear();
    if (charText.length() == 0){
        xemVideoList.addAll(arrayList);
    }else{
        for (XemVideo video: arrayList){
            if (video.getTitle().toLowerCase(Locale.getDefault()).contains(charText)) {
                xemVideoList.add(video);
            }
            if (video.getTheLoai().toLowerCase(Locale.getDefault()).contains(charText)) {
                xemVideoList.add(video);
            }
            if (video.getChannelTitle().toLowerCase(Locale.getDefault()).contains(charText)) {
                xemVideoList.add(video);
            }
        }
    }
    notifyDataSetChanged();
}
```

Hình 3.2.4.24. Tạo hàm filter thực hiện chức năng tìm kiếm video.

CHƯƠNG 4: CHƯƠNG TRÌNH THỰC NGHIỆM

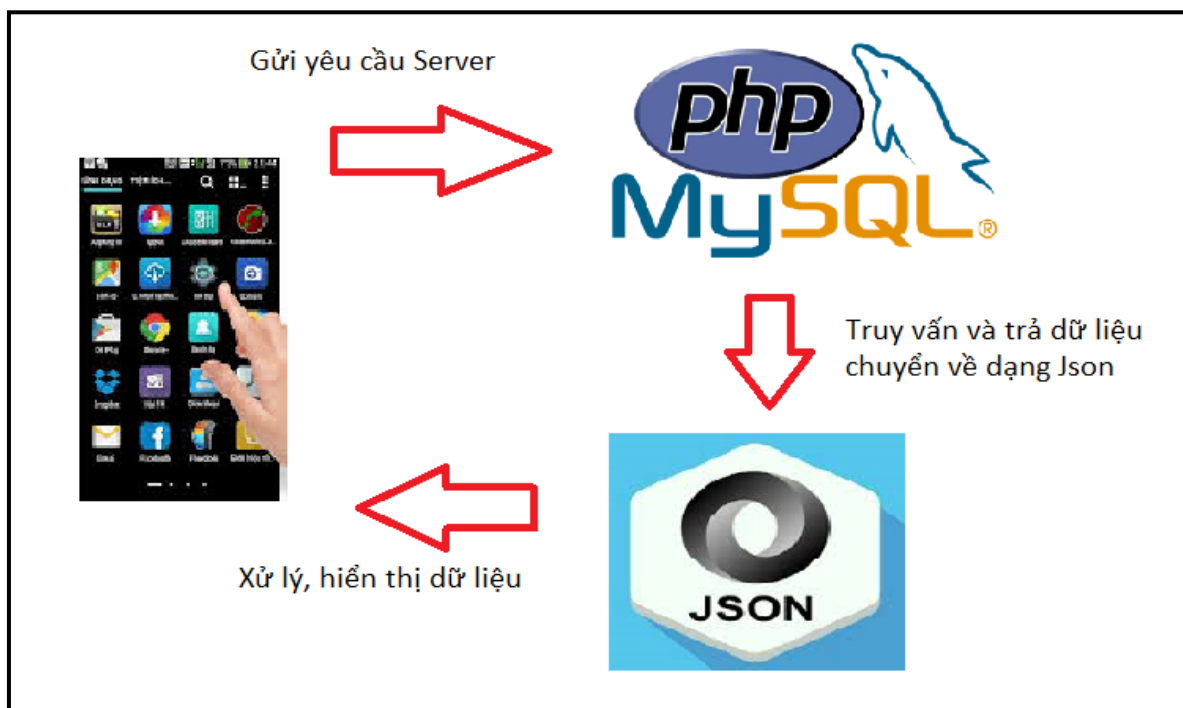
4.1. Bài toán.

Trên cơ sở nhu cầu của người sử dụng ngày càng tăng và mọi điều kiện về cơ sở vật chất đều đã có đầy đủ thì một hệ thống cho phép tìm kiếm, cập nhập, xem video trực tuyến mà không cần phải tải toàn bộ về thiết bị trước khi xem có lẽ sẽ trở lên phổ biến và quan trọng trong cộng đồng người sử dụng mạng 3G, Wifi trên hệ điều hành Android.

Vậy bài toán của chúng ta ở đây sẽ là xây dựng một hệ thống cho phép tìm kiếm, cập nhập, xem video trực tuyến trên nền tảng hệ điều hành mã nguồn mở Android. Đây là bài toán đầu tiên cho sự kết hợp giữa truyền dữ liệu cao trên công nghệ 3G, Wifi và hệ điều hành mã nguồn mở Android, nó sẽ là nền tảng cho sự phát triển các ứng dụng hệ thống thông tin trực tuyến đòi hỏi truyền dữ liệu tốc độ cao như học trực tuyến, truyền hình, giải trí trực tuyến,...

Trên thế giới có một hệ thống rất nổi tiếng, đó là hệ thống YouTube, được cung cấp qua trang web <https://www.youtube.com/> . Nó cũng cho phép xem video dưới dạng streaming trên cả thiết bị di động và trên máy tính.

4.2. Mô hình chương trình.



Hình 4.2.1 Mô hình chuyển hóa dữ liệu về thiết bị di động.

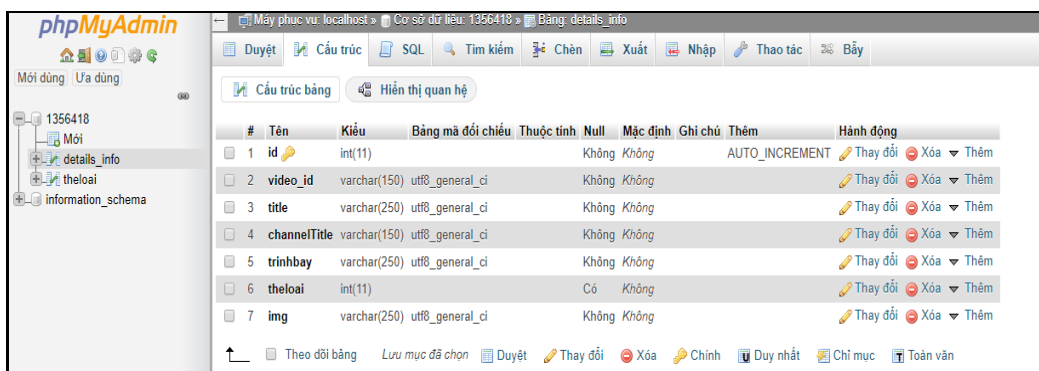
4.3. Các bước xây dựng ứng dụng Android xem video trực tuyến.

Sau khi đã tìm hiểu các kỹ thuật để xây dựng ứng dụng Android xem video trực tuyến ở chương 3 chúng ta đi vào xây dựng chương trình thực nghiệm gồm các bước:

Bước 1: Tạo cơ sở dữ liệu.

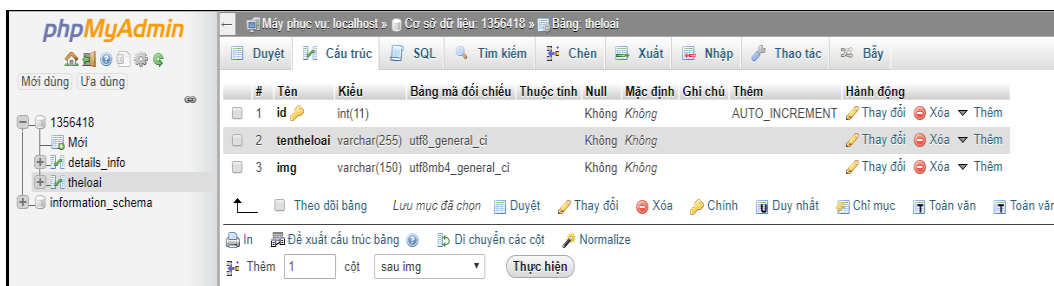
- Cơ sở dữ liệu gồm 2 bảng gồm các cột như hình 4.3.1 và hình 4.3.2 ví dụ dưới đây:

Bảng details_info: Lưu trữ toàn bộ thông tin cho các video. Khi quản trị viên thực hiện thêm mới, chỉnh sửa.



Hình 4.3.1. Bảng lưu trữ thông tin video.

Bảng theloai: Lưu trữ thông tin cho các thể loại. Khi quản trị viên thêm mới, chỉnh sửa.



Hình 4.3.2. Bảng lưu trữ thông tin thể loại.

Bước 2: Xây dựng Webservice như đã hướng dẫn tại mục 3.2.1 trong chương 3.

Sau khi xây dựng xong Webservice chúng ta có thể thực hiện thêm mới dữ liệu vào bảng. Thêm mới thành công sẽ CSDL như hình ví dụ 4.3.3.

+ Tùy chọn										
	id	video_id	title	channelTitle	trìnhbay	theloai	img			
<input type="checkbox"/>	Sửa	Chép	Xóa bỏ	30	k7x7Q3-nU6o	Lập trình Android A-Z: Cài đặt mới	Trung Tâm Đào Tạo Tin Học Khoa Phạm	Giảng viên	1	1541409857_9479.jpg
<input type="checkbox"/>	Sửa	Chép	Xóa bỏ	31	P60kcSaeFmg	Lập trình Android A-Z – Bài 1: Tạo project	Trung Tâm Đào Tạo Tin Học Khoa Phạm	Giảng viên	1	1541409928_8482.jpg
<input type="checkbox"/>	Sửa	Chép	Xóa bỏ	32	-C-7zYgci8	Lập trình Android A-Z – Bài 2: Giới thiệu Androi...	Trung Tâm Đào Tạo Tin Học Khoa Phạm	Giảng viên	1	1541411824_6789.jpg
<input type="checkbox"/>	Sửa	Chép	Xóa bỏ	33	wFnDXhnRFAM	Lập trình Android A-Z – Bài 3: Ôn tập Java - phần...	Trung Tâm Đào Tạo Tin Học Khoa Phạm	Giảng viên	1	1541411918_3510.jpg
<input type="checkbox"/>	Sửa	Chép	Xóa bỏ	34	zNrKU9SZUvA	Lập trình Android A-Z – Bài 4: Ôn tập Java - phần...	Trung Tâm Đào Tạo Tin Học Khoa Phạm	Giảng viên	1	1541411970_5493.jpg
<input type="checkbox"/>	Sửa	Chép	Xóa bỏ	35	DATqsv18ojg	Lập trình Android A-Z – Bài 5: Ôn tập Java - ph...	Trung Tâm Đào Tạo Tin Học Khoa Phạm	Giảng viên	1	1541412027_6200.jpg
<input type="checkbox"/>	Sửa	Chép	Xóa bỏ	36	3HIO4j4_VCU	Lập trình Android A-Z – Bài 6: Ôn tập Java - ph...	Trung Tâm Đào Tạo Tin Học Khoa Phạm	Giảng viên	1	1541412082_9820.jpg
<input type="checkbox"/>	Sửa	Chép	Xóa bỏ	37	4gLcrtg1fAo	Lập trình Android A-Z – Bài 7: Ôn tập Java – phâ...	Trung Tâm Đào Tạo Tin Học Khoa Phạm	Giảng viên	1	1541412120_3036.jpg
<input type="checkbox"/>	Sửa	Chép	Xóa bỏ	38	x4teN2o1FoU	Lập trình Android A-Z – Bài 8: Khái niệm View và ...	Trung Tâm Đào Tạo Tin Học Khoa Phạm	Giảng viên	1	1541412159_7914.jpg
<input type="checkbox"/>	Sửa	Chép	Xóa bỏ	39	YwXtYN3kxSA	Lập trình Android A-Z – Bài 9: Linear Layout	Trung Tâm Đào Tạo Tin Học Khoa Phạm	Giảng viên	1	1541412202_5900.jpg
<input type="checkbox"/>	Sửa	Chép	Xóa bỏ	40	R8Wd1ppqzpiI	Lập trình Android A-Z – Bài 10: Linear Layout với...	Trung Tâm Đào Tạo Tin Học Khoa Phạm	Giảng viên	1	1541412249_7420.jpg
<input type="checkbox"/>	Sửa	Chép	Xóa bỏ	41	rWMm_GDMlRE	Hướng Dẫn Luyện Nghe Tiếng Anh Cho Người Mới Bắt Đ...	Learn English Online	Cô Vũ Mai Phương	1	1541412717_4354.jpg
<input type="checkbox"/>	Sửa	Chép	Xóa bỏ	42	sRNHz_1_sN8	Luyện nghe Tiếng Anh cho người mới bắt đầu Bài 1...	Learn English Online	Cô Vũ Mai Phương	1	1541412823_8315.jpg
<input type="checkbox"/>	Sửa	Chép	Xóa bỏ	43	69s1NW16pLo	Luyện nghe Tiếng Anh cho người mới bắt đầu Bài 1...	Learn English Online	Cô Vũ Mai Phương	1	1541412871_9450.jpg
<input type="checkbox"/>	Sửa	Chép	Xóa bỏ	44	nKIDkajVLrs	[Pre Toieic] Bài 1 - Các thành phần	Learn English Online	Cô Vũ Mai Phương	1	1541412970_2131.jpg

Hình 4.3.3. Một số dữ liệu đã được thêm.

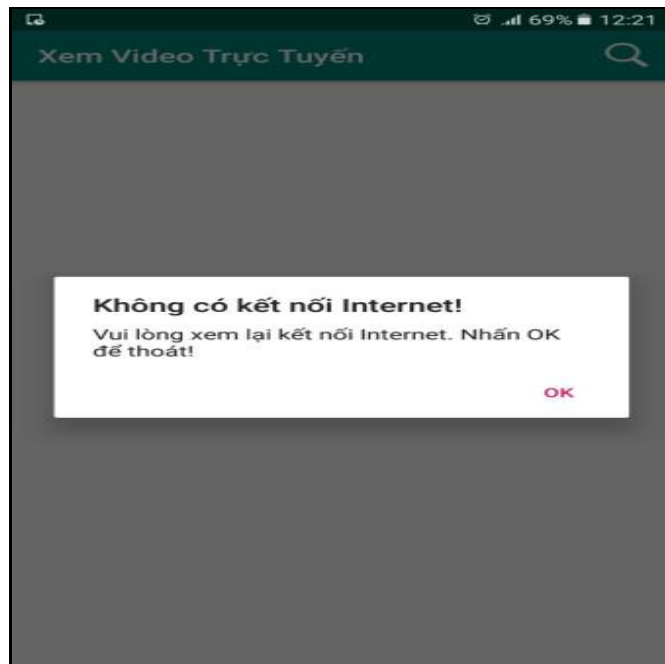
Bước 3: Thực hiện kiểm tra cấp quyền kết nối Internet trên thiết bị di động như đã hướng dẫn tại mục 3.2.2 trong chương 3.

Bước 4: Tạo giao diện và lấy dữ liệu từ CSDL trên Internet về thiết bị di động như đã hướng dẫn tại mục 3.2.3 trong chương 3.

Bước 5: Bắt và xử lý sự kiện click như đã hướng dẫn tại mục 3.2.4 trong chương 3.

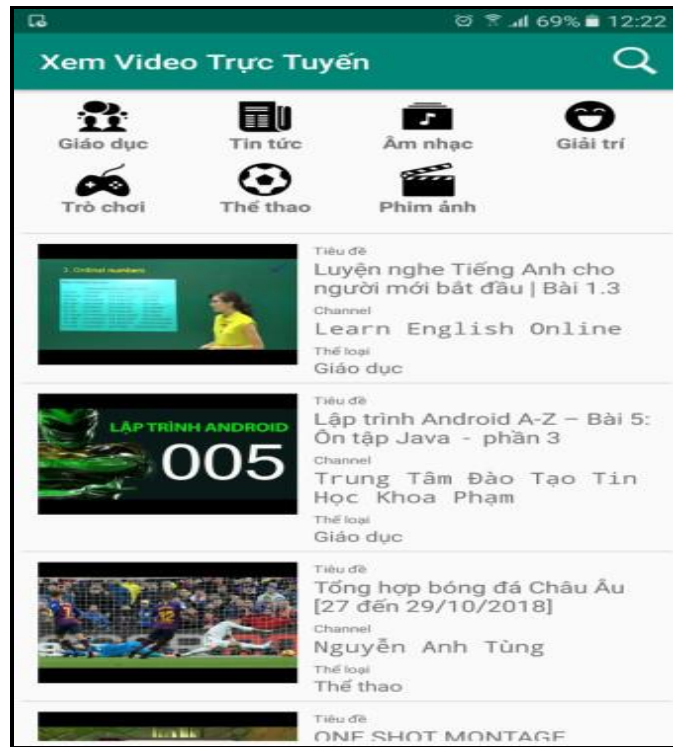
4.4. Giao diện chương trình.

- Giao diện ứng dụng khi thực hiện kiểm tra kết nối Internet như hình 4.4.1.



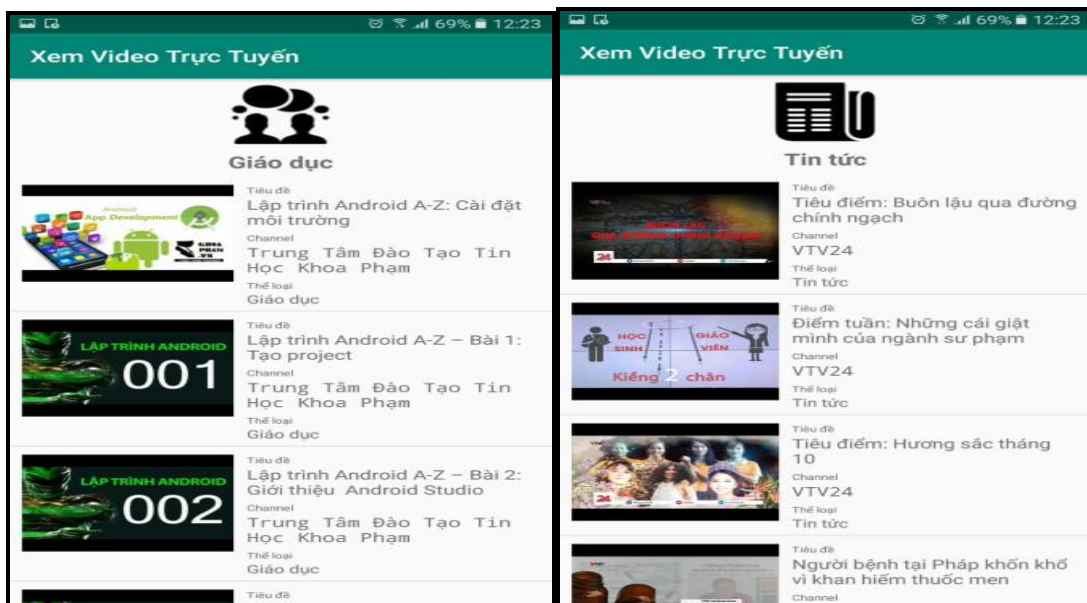
Hình 4.4.1. Giao diện khi không có kết nối Internet.

- Sau khi kiểm tra nếu kết nối Internet thành công. Sẽ xuất hiện giao diện ứng dụng khi ở màn hình trang chủ như hình 4.4.2.



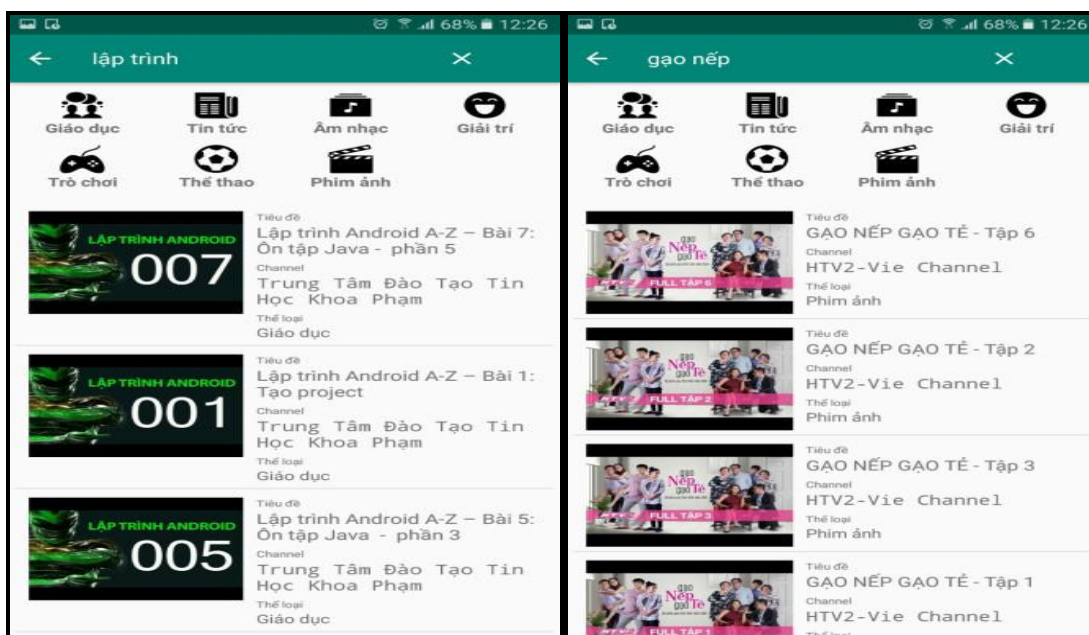
Hình 4.4.2. Giao diện màn hình trang chủ.

- Khi người dùng muốn chọn lọc các video theo thể loại có sẵn. Click vào biểu tượng một thể loại bất kỳ tại trang chủ sẽ xuất hiện giao diện như hình 4.4.3.



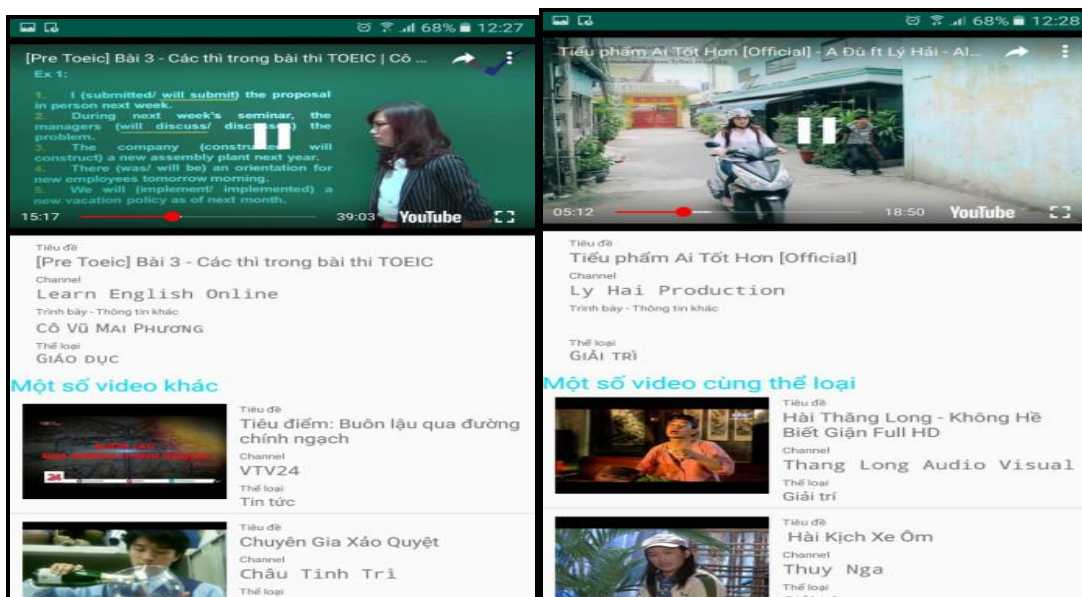
Hình 4.4.3. Giao diện khi click vào một thể loại.

- Người dùng muốn chọn lọc các video theo tiêu đề hoặc thể loại. Click thanh tìm kiếm và nhập nội dung cần tìm. Sẽ xuất hiện giao diện như hình 4.4.4.



Hình 4.4.4. Giao diện khi tìm kiếm video.

- Khi người dùng click vào một video bất kỳ. Video đó sẽ được phát và hiển thị các thông tin của video, có các chức năng cơ bản như tua video, thanh đổi chất lượng video, dừng và phát video, phóng to màn hình. Ngay bên dưới sẽ có một số video khác để người dùng lựa chọn trong khi vẫn xem được video như hình 4.4.5.



Hình 4.4.5. Giao diện khi phát một video bất kỳ.

KẾT LUẬN

Trong khoảng thời gian làm đồ án em đã học hỏi được rất nhiều điều từ nghiên cứu cho đến triển khai làm đồ án tốt nghiệp. Đồ án đã đạt được những yêu cầu: tìm hiểu lịch sử và kiến trúc hệ điều hành Android, tìm hiểu môi trường lập trình Android Studio, biết được các quy trình xây dựng một ứng dụng Android, tìm hiểu hosting, biết cách xây dựng Webservice, các kỹ thuật lấy dữ liệu từ Webservice cho thiết bị di động, xây dựng kho lưu trữ video trực tuyến trên hosting, tìm hiểu kỹ thuật phát video trên internet thông qua đường dẫn, các bước phát triển một ứng dụng trên hệ điều hành Android.

Bước đầu xây dựng thành công một ứng dụng Android xem video trực tuyến. Tuy nhiên, do thời gian và trình độ cũng như kiến thức còn hạn chế nên chương trình vẫn còn thiếu sót một số chức năng như: phát lại video xem trước đó, tải video về thiết bị di động,.... Em mong sẽ nhận được những góp ý của thầy cô và bạn bè để chương trình của em hoàn thiện và có tính ứng dụng nhiều hơn nữa.

Trong thời gian tới, em sẽ phát triển ứng dụng thêm nhiều chức năng mới để người dùng sử dụng một cách tiện lợi hơn nữa.

TÀI LIỆU THAM KHẢO

- [1] <https://ictnews.vn/kinh-doanh/ho-so>
- [2] <https://vi.wikipedia.org>
- [3] <https://vov.vn/cong-nghe/diem-lai-cac-phien-ban-he-dieu-hanh-android-trong-10-nam-qua-798192.vov>
- [4] <https://www.howkteam.vn/course/khoa-hoc-lap-trinh-android-co-ban/cac-thanh-phan-giao-dien-ui-co-ban-72>
- [5] <http://giasutinhoc.vn/lap-trinh-android-co-ban/xu-ly-su-kien-trong-android-bai-3/>
- [6] <https://viblo.asia/p/vong-doi-cua-activity-trong-ung-dung-android-NbmebzLMeYO>
- [7] <https://viblo.asia/p/su-dung-intent-trong-ung-dung-android-3Q75wyBGZWb>
- [8] <https://inthecheesefactory.com/blog/how-to-install-google-services-on-genymotion/en>