

TRƯỜNG ĐẠI HỌC THỦY LỢI
KHOA CÔNG NGHỆ THÔNG TIN



GIÁO TRÌNH

THỰC HÀNH PHÁT TRIỂN ỨNG DỤNG CHO THIẾT BỊ DI ĐỘNG

Hà Nội, 2.2025

MỤC LỤC

CHƯƠNG 1. Làm quen.....	3
Bài 1) Tạo ứng dụng đầu tiên	3
1.1) Android Studio và Hello World	3
1.2) Giao diện người dùng tương tác đầu tiên	18
1.3) Trình chỉnh sửa bố cục	18
1.4) Văn bản và các chế độ cuộn	18
1.5) Tài nguyên có sẵn.....	18
Bài 2) Activities	18
2.1) Activity và Intent	18
2.2) Vòng đời của Activity và trạng thái	18
2.3) Intent ngầm định.....	18
Bài 3) Kiểm thử, gỡ lỗi và sử dụng thư viện hỗ trợ.....	18
3.1) Trình gỡ lỗi.....	18
3.2) Kiểm thử đơn vị.....	18
3.3) Thư viện hỗ trợ.....	18
CHƯƠNG 2. Trải nghiệm người dùng.....	18
Bài 1) Tương tác người dùng.....	18
1.1) Hình ảnh có thể chọn	18
1.2) Các điều khiển nhập liệu	19
1.3) Menu và bộ chọn	19
1.4) Điều hướng người dùng	19
1.5) RecyclerView	19
Bài 2) Trải nghiệm người dùng thú vị.....	19
2.1) Hình vẽ, định kiểu và chủ đề	19
2.2) Thẻ và màu sắc	19

2.3)	Bố cục thích ứng.....	19
Bài 3)	Kiểm thử giao diện người dùng.....	19
3.1)	Espresso cho việc kiểm tra UI	19
CHƯƠNG 3.	Làm việc trong nền	19
Bài 1)	Các tác vụ nền.....	19
1.1)	AsyncTask	19
1.2)	AsyncTask và AsyncTaskLoader	19
1.3)	Broadcast receivers	19
Bài 2)	Kích hoạt, lập lịch và tối ưu hóa nhiệm vụ nền.....	19
2.1)	Thông báo	19
2.2)	Trình quản lý cảnh báo	19
2.3)	JobScheduler.....	19
CHƯƠNG 4.	Lưu dữ liệu người dùng	19
Bài 1)	Tùy chọn và cài đặt.....	19
1.1)	Shared preferences.....	20
1.2)	Cài đặt ứng dụng.....	20
Bài 2)	Lưu trữ dữ liệu với Room	20
2.1)	Room, LiveData và ViewModel.....	20
2.2)	Room, LiveData và ViewModel.....	20
3.1)	Trình gowx lỗi	

CHƯƠNG 1. LÀM QUEN

Bài 1) Tạo ứng dụng đầu tiên

1.1) Android Studio và Hello World

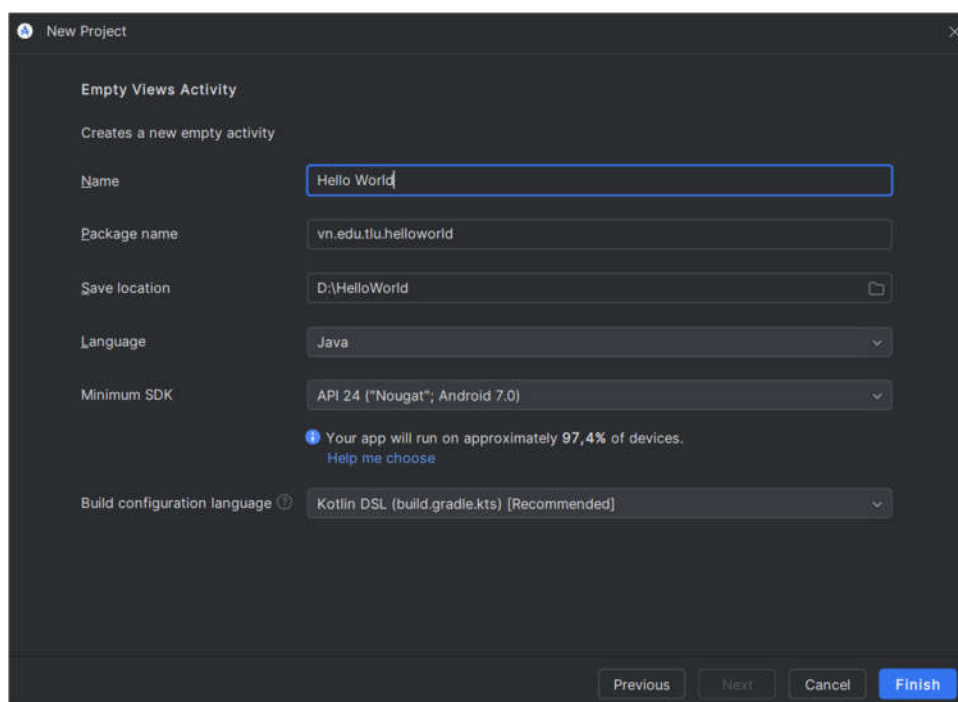
Giới thiệu

Trong bài thực hành này, bạn sẽ tìm hiểu cách cài đặt Android Studio, môi trường phát triển Android. Bạn cũng sẽ tạo và chạy ứng dụng Android đầu tiên của mình, Hello World, trên một trình giả lập và trên một thiết bị vật lý.

Những gì Bạn nên biết

Bạn nên có khả năng:

- Hiểu quy trình phát triển phần mềm tổng quát cho các ứng dụng lập trình hướng đối tượng sử dụng một IDE (môi trường phát triển tích hợp) như Android Studio.
- Chứng minh rằng bạn có ít nhất 1-3 năm kinh nghiệm trong lập trình hướng đối tượng, với một phần trong số đó tập trung vào ngôn ngữ lập trình Java. (Các bài thực hành này sẽ không giải thích về lập trình hướng đối tượng hoặc ngôn ngữ Java.



Những gì Bạn sẽ cần:

- Một máy tính chạy Windows hoặc Linux, hoặc một Mac chạy macOS. Xem trang tải xuống Android Studio để biết yêu cầu hệ thống cập nhật.
- Truy cập Internet hoặc một phương pháp thay thế để tải các cài đặt mới nhất của Android Studio và Java lên máy tính của bạn.

Những gì bạn sẽ học

- Cách cài đặt và sử dụng IDE Android Studio.
- Cách sử dụng quy trình phát triển để xây dựng ứng dụng Android.
- Cách tạo một dự án Android từ một mẫu.
- Cách thêm thông điệp ghi lại vào ứng dụng của bạn để phục vụ mục đích gỡ lỗi.

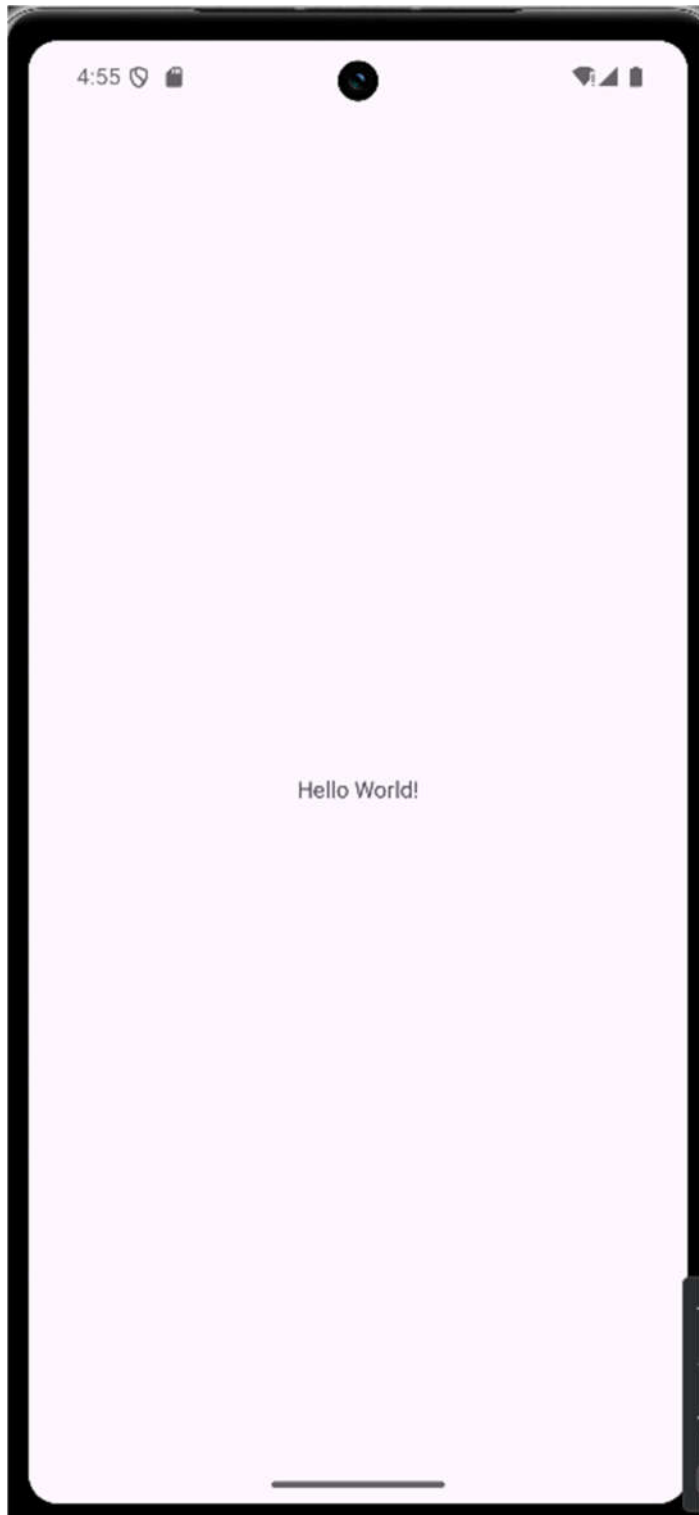
Những gì bạn sẽ làm

- Cài đặt môi trường phát triển **Android Studio**.
- Tạo một trình giả lập (thiết bị ảo) để chạy ứng dụng của bạn trên máy tính.
- Tạo và chạy ứng dụng **Hello World** trên các thiết bị ảo và vật lý.
- Khám phá cấu trúc dự án.
- Tạo và xem các thông điệp ghi lại từ ứng dụng của bạn.
- Khám phá tệp **AndroidManifest.xml**

Tổng quan về ứng dụng

Sau khi cài đặt thành công Android Studio, bạn sẽ tạo một dự án mới từ một mẫu cho Ứng dụng HelloWorld. Ứng dụng đơn giản này hiển thị chuỗi “Hello World” trên màn hình Android thiết bị ảo hoặc vật lý.

Đây là ứng dụng hoàn thiện sẽ trông như thế nào:



Nhiệm vụ 1: Cài đặt Android Studio

Android Studio cung cấp một môi trường phát triển tích hợp (IDE) hoàn chỉnh bao gồm trình soạn thảo mã nâng cao và một bộ mẫu ứng dụng. Ngoài ra, nó còn chứa các công cụ để phát triển, gỡ lỗi, kiểm tra và hiệu suất giúp phát triển ứng dụng nhanh hơn và dễ dàng hơn. Bạn có thể kiểm tra ứng dụng của bạn với nhiều trình mô phỏng được cấu hình sẵn hoặc trên thiết bị di động của riêng bạn, hãy xây dựng ứng dụng sản xuất và xuất bản trên cửa hàng Google Play.

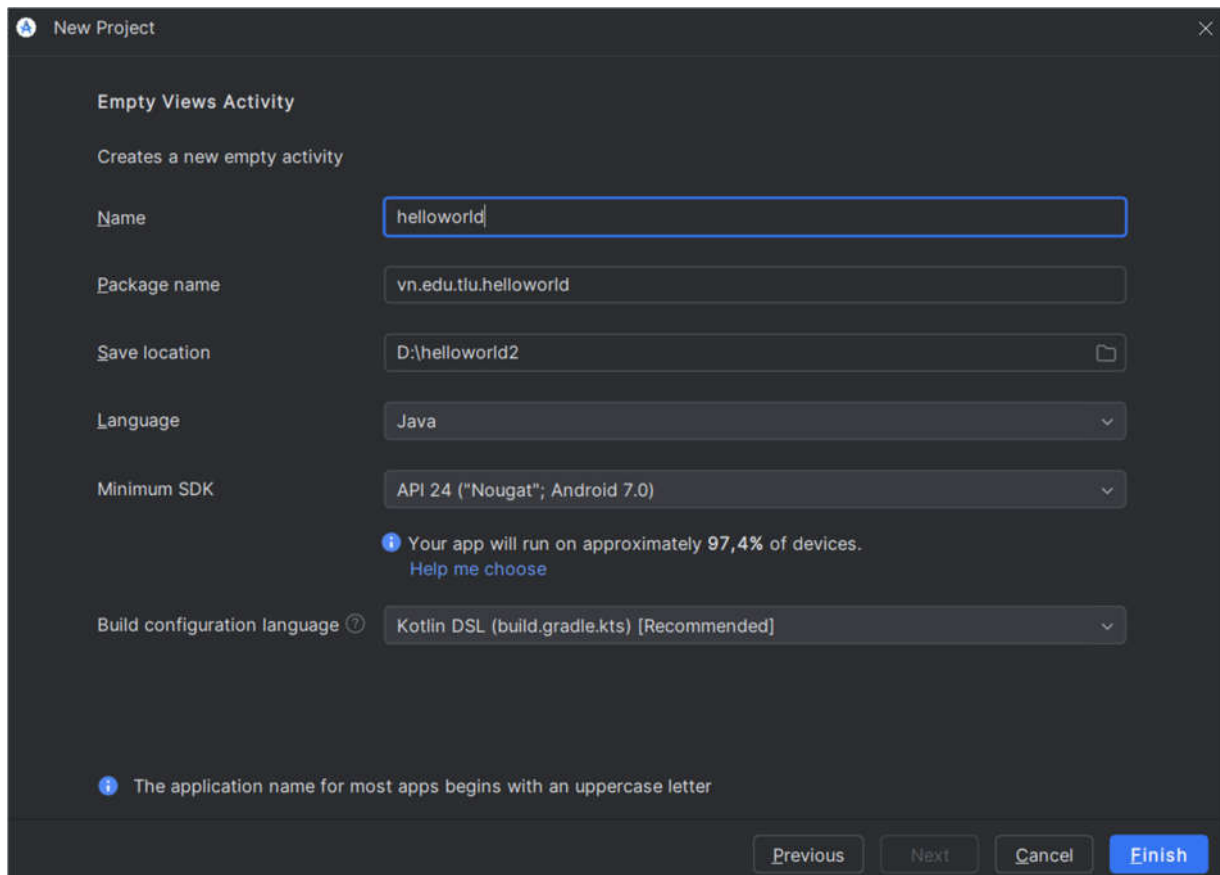
Android Studio có sẵn cho máy tính chạy Windows hoặc Linux và cho máy Mac chạy macOS. OpenJDK (Bộ công cụ phát triển Java) mới nhất đi kèm với Android Studio.

Để thiết lập và chạy Android Studio, trước tiên hãy kiểm tra các yêu cầu hệ thống để đảm bảo rằng hệ thống đáp ứng chúng. Việc cài đặt tương tự cho tất cả các nền tảng. Mọi khác biệt đều được ghi chú bên dưới

Task 2: Create the Hello World app

2.1 Tạo dự án ứng dụng

1. Mở Android Studio nếu nó chưa được mở.
2. Trong cửa sổ Chào mừng bạn đến với Android Studio chính, nhấp vào Bắt đầu Android Studio mới dự án.
3. Trong cửa sổ Tạo dự án Android, nhập Hello World cho tên Ứng dụng.



4. Xác minh rằng vị trí Dự án mặc định là nơi bạn muốn lưu trữ ứng dụng Hello World của mình và các dự án Android Studio khác hoặc thay đổi nó vào thư mục ưa thích của bạn.

5. Chấp nhận android.example.com mặc định cho Miền công ty hoặc tạo một tên miền duy nhất miền công ty. Nếu bạn không định xuất bản ứng dụng của mình, bạn có thể chấp nhận mặc định. Hãy nhận biết rằng việc thay đổi tên gói ứng dụng của bạn sau này là một công việc bổ sung.

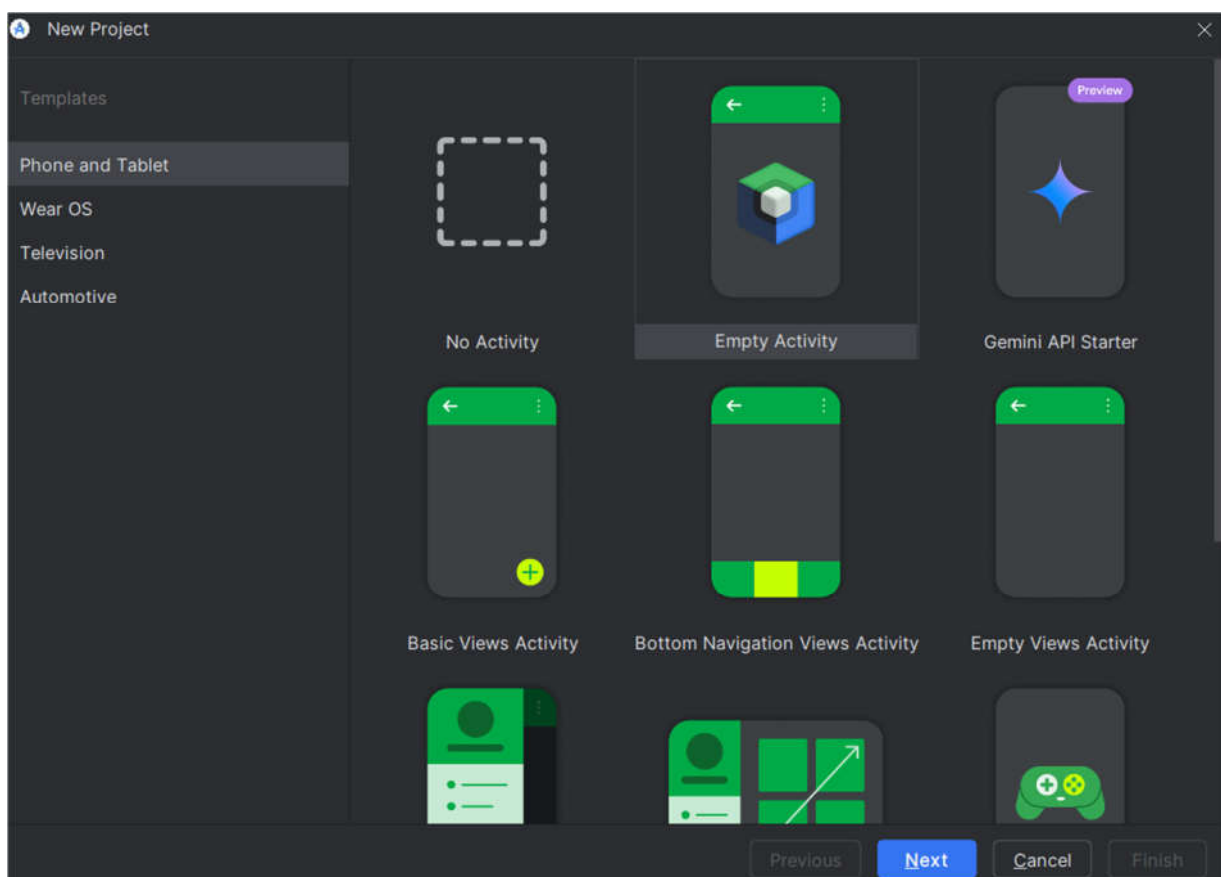
6. Bỏ chọn các tùy chọn Bao gồm hỗ trợ C++ và Bao gồm hỗ trợ Kotlin rồi nhấp vào Kế tiếp.

7. Trên màn hình Thiết bị Android mục tiêu, nên chọn Điện thoại và Máy tính bảng. Đảm bảo rằng API 15: Android 4.0.3 IceCreamSandwich được đặt làm SDK tối thiểu; nếu không, hãy sử dụng menu bật lên để thiết lập nó. Đây là các cài đặt được sử dụng trong các ví dụ trong các bài học của khóa học này. Theo văn bản này, những cài đặt này giúp

ứng dụng Hello World của bạn tương thích với 97% thiết bị Android đang hoạt động trên Cửa hàng Google Play.

8. Bỏ chọn tùy chọn Bao gồm hỗ trợ Ứng dụng tức thì và tất cả các tùy chọn khác. Sau đó bấm vào Tiếp theo. Nếu như dự án của bạn yêu cầu các thành phần bổ sung cho SDK mục tiêu bạn đã chọn, Android Studio sẽ cài đặt chúng tự động.

9. Cửa sổ Thêm hoạt động xuất hiện. Hoạt động là một thứ duy nhất, tập trung mà người dùng có thể làm được. Nó là một thành phần quan trọng của bất kỳ ứng dụng Android nào. Một Hoạt động thường có bố cục được liên kết với nó để xác định cách các thành phần UI xuất hiện trên màn hình. Android Studio cung cấp Các mẫu hoạt động để giúp bạn bắt đầu. Đối với dự án Hello World, chọn Trống Hoạt động như hình bên dưới rồi nhấn Next.



10. Màn hình Định cấu hình Hoạt động xuất hiện (khác nhau tùy thuộc vào mẫu bạn

đã chọn ở bước trước). Theo mặc định, Hoạt động trống do mẫu cung cấp là được đặt tên là MainActivity. Bạn có thể thay đổi điều này nếu muốn, nhưng bài học này sử dụng MainActivity.

11. Đảm bảo rằng tùy chọn Tạo tệp bố cục đã được chọn. Tên bố cục theo mặc định là Activity_main. Bạn có thể thay đổi điều này nếu muốn, nhưng bài học này sử dụng Activity_main.

12. Đảm bảo rằng tùy chọn Tương thích ngược (Tương thích ứng dụng) được chọn. Cái này đảm bảo rằng ứng dụng của bạn sẽ tương thích ngược với các phiên bản Android trước đó.

13. Nhấp vào Hoàn tất.

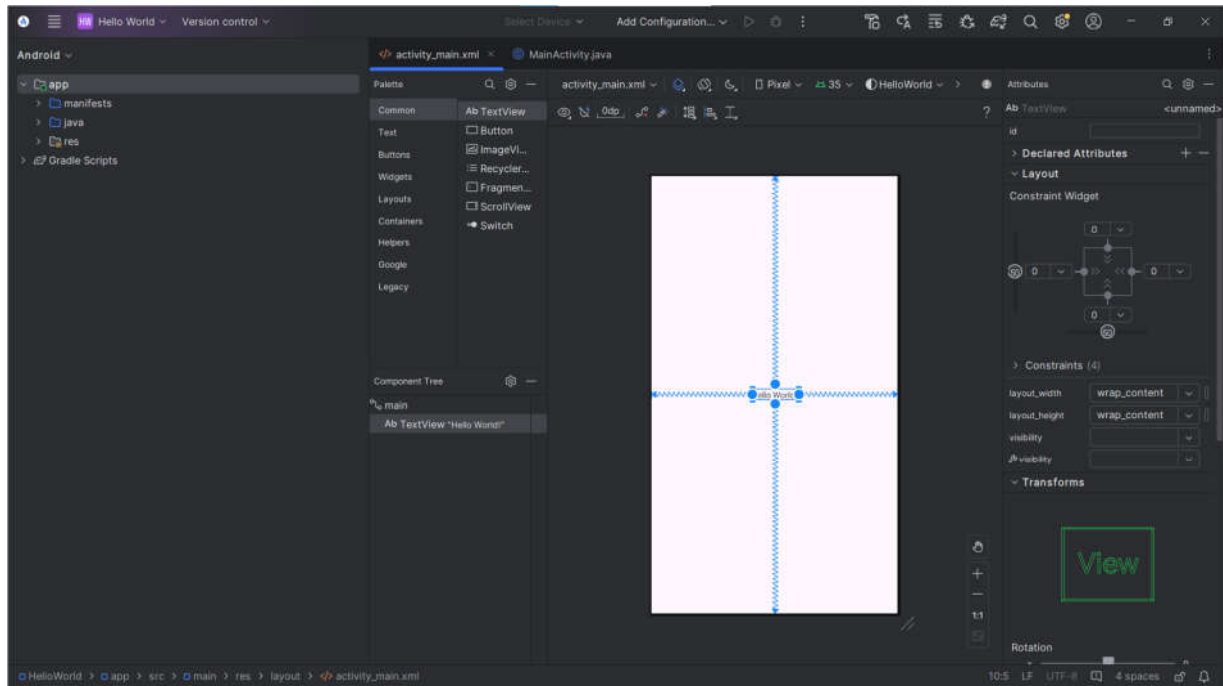
Android Studio tạo một thư mục cho các dự án của bạn và xây dựng dự án bằng Gradle (việc này có thể mất một chút thời gian).

Mẹo: Xem trang Định cấu hình nhà phát triển bản dựng của bạn để biết thông tin chi tiết.

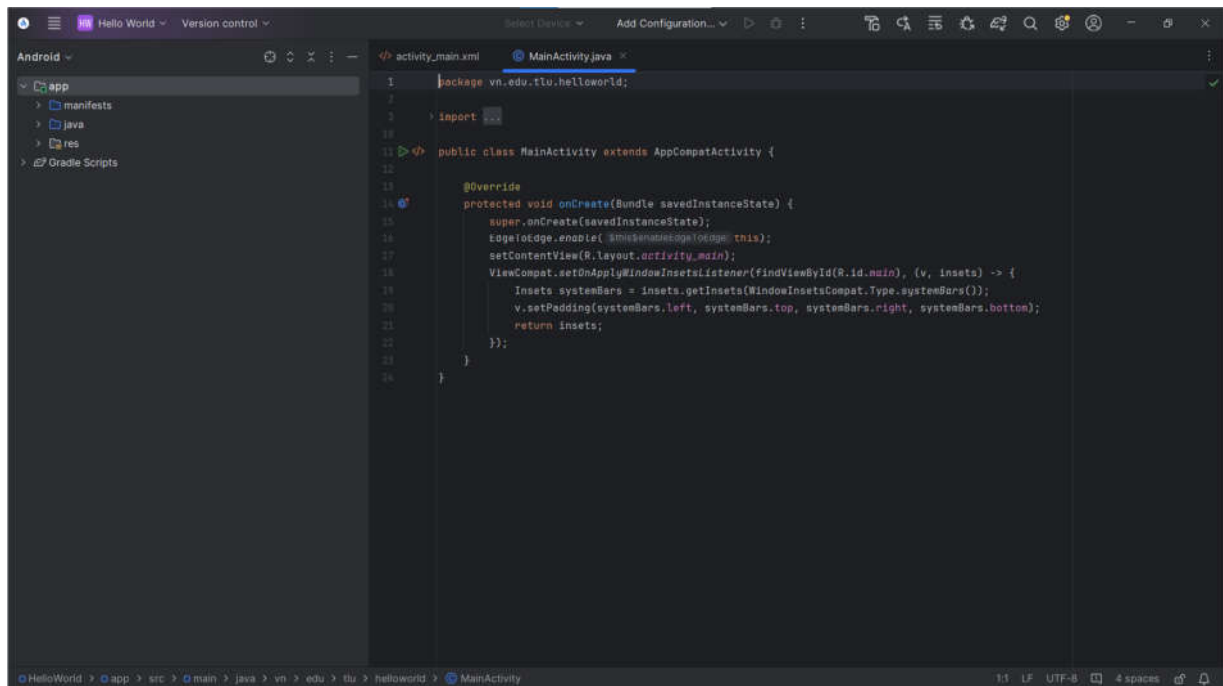
Bạn cũng có thể thấy thông báo "Mẹo trong ngày" kèm theo phím tắt và các mẹo hữu ích khác. Nhấp chuột Close để đóng tin nhắn.

Trình chỉnh sửa Android Studio xuất hiện. Thực hiện theo các bước sau:

1. Nhấp vào tab hoạt động_main.xml để xem trình chỉnh sửa bố cục.
2. Nhấp vào tab Thiết kế của trình chỉnh sửa bố cục, nếu chưa được chọn, để hiển thị đồ họa của bố cục như hình dưới đây.



3. Nhấp vào tab MainActivity.java để xem trình soạn thảo mã như hiển thị bên dưới

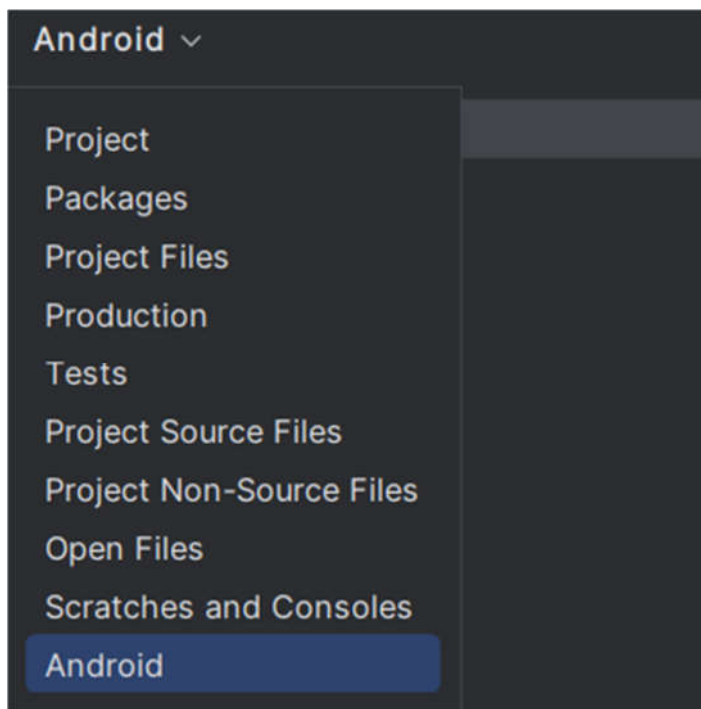


2.2 Explore the Project > Android pane

Trong phần thực hành này, bạn sẽ khám phá cách tổ chức dự án trong Android Studio.

1. Nếu chưa được chọn, hãy nhấp vào tab Dự án trong cột tab dọc ở phía bên trái của Cửa sổ Android Studio. Ngăn Dự án xuất hiện.

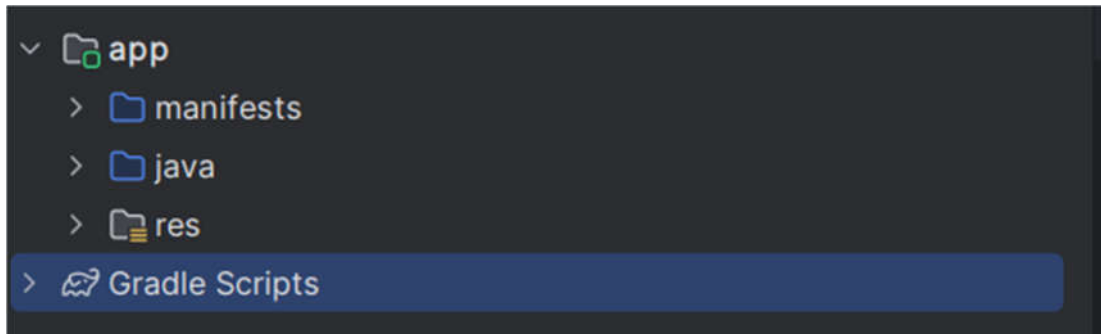
2. Để xem dự án theo hệ thống phân cấp dự án Android tiêu chuẩn, hãy chọn Android từ menu bật lên ở đầu ngăn Dự án, như hiển thị bên dưới.



2.3 Khám phá thư mục Gradle Scripts

Hệ thống xây dựng Gradle trong Android Studio giúp dễ dàng đưa các tệp nhị phân bên ngoài hoặc thư viện khác vào mô-đun vào bản dựng của bạn dưới dạng phần phụ thuộc.

Khi bạn tạo một dự án ứng dụng lần đầu tiên, ngăn **Project > Android** sẽ xuất hiện cùng với Tập lệnh **Gradle Scripts** thư mục được mở rộng như hiển thị bên dưới.



Hãy làm theo các bước sau để khám phá hệ thống Gradle:

1. Nếu thư mục Tập lệnh Gradle không được mở rộng, hãy nhấp vào hình tam giác để mở rộng.

Thư mục này chứa tất cả các tệp cần thiết cho hệ thống xây dựng.

2. Tìm tệp build.gradle(Project: HelloWorld).

Đây là nơi bạn sẽ tìm thấy các tùy chọn cấu hình chung cho tất cả các mô-đun tạo nên dự án của bạn. Mỗi dự án Android Studio đều chứa một Gradle cấp cao nhất xây dựng tập tin. Trong hầu hết mọi trường hợp, bạn sẽ không cần thực hiện bất kỳ thay đổi nào đối với tệp này, nhưng nó vẫn hữu ích để hiểu nội dung của nó.

Theo mặc định, tệp xây dựng cấp cao nhất sử dụng khối buildscript để xác định Gradle kho lưu trữ và phần phụ thuộc chung cho tất cả các mô-đun trong dự án. Khi bạn phần phụ thuộc không phải là thư viện cục bộ hoặc cây tệp, Gradle sẽ tìm các tệp trong bất kỳ kho lưu trữ trực tuyến nào được chỉ định trong khối kho lưu trữ của tệp này. Theo mặc định, các dự án Android Studio mới khai báo JCenter và Google (bao gồm Google Maven kho lưu trữ) làm vị trí kho lưu trữ:

3. Tìm tệp build.gradle(Module:app).

Ngoài tệp build.gradle cấp dự án, mỗi mô-đun còn có tệp build.gradle của nó own, cho phép bạn định cấu hình cài đặt bản dựng cho từng mô-đun cụ thể (HelloWorld app chỉ có một mô-đun). Định cấu hình các cài đặt bản dựng này cho phép bạn cung cấp tùy chỉnh các tùy chọn đóng gói, chẳng hạn như kiểu xây dựng bổ sung và hương vị sản phẩm. Bạn cũng có thể ghi đè cài đặt trong tệp AndroidManifest.xml hoặc tệp build.gradle cấp cao nhất. Tệp này thường là tệp cần chỉnh sửa khi thay đổi cấu hình cấp ứng dụng, chẳng hạn như khai báo các phụ thuộc trong phần phụ thuộc. Bạn có thể khai báo một phụ thuộc thư viện bằng cách sử dụng một trong một số cấu hình

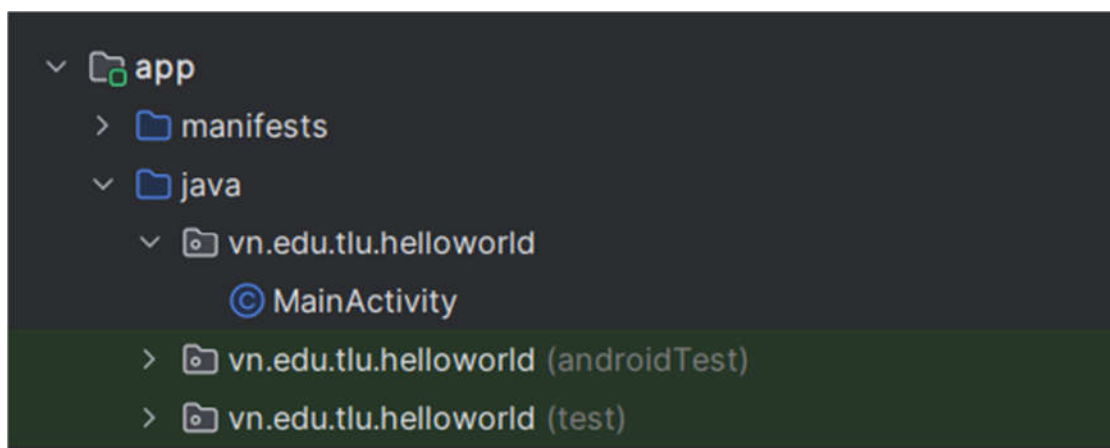
phụ thuộc khác nhau. Mỗi cấu hình phụ thuộc cung cấp cho Gradle những hướng dẫn khác nhau về cách sử dụng thư viện. Ví dụ, triển khai câu lệnh `fileTree(dir: 'libs', include: ['*.jar'])` thêm một sự phụ thuộc của tất cả các tệp “.jar” bên trong thư mục `libs`.

4. Nhấp vào hình tam giác để đóng Tập lệnh Gradle.

2.4 Khám phá ứng dụng và thư mục res

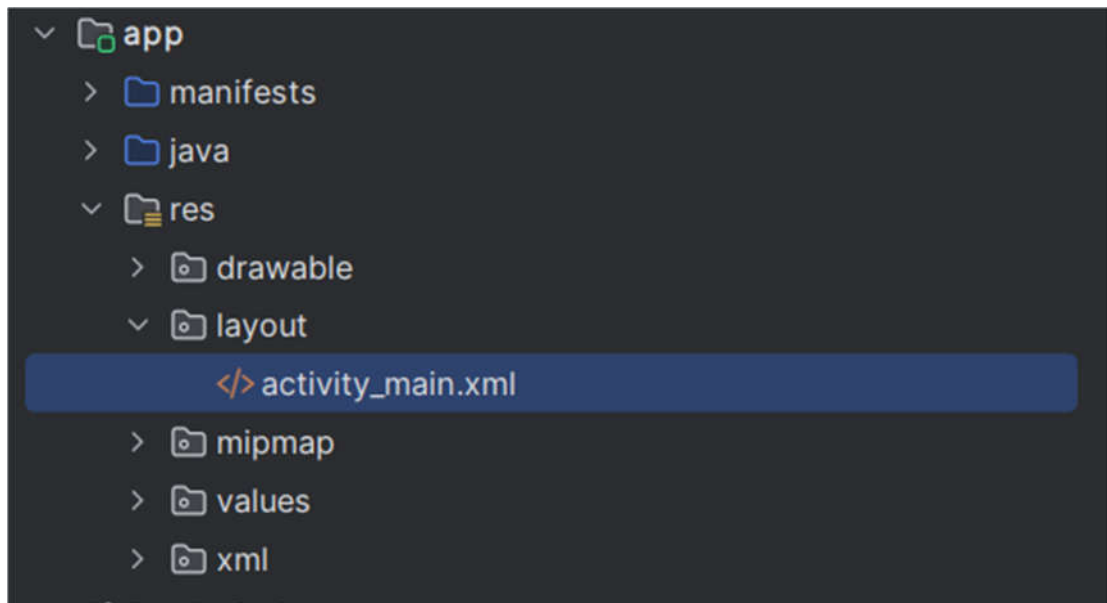
Tất cả mã và tài nguyên cho ứng dụng đều nằm trong thư mục ứng dụng và `res`.

1. Mở rộng thư mục ứng dụng, thư mục `java` và thư mục `com.example.android.helloworld` thành xem tệp `MainActivity`. Bấm đúp vào tệp sẽ mở nó trong trình chỉnh sửa mã.



Thư mục `java` bao gồm các tệp lớp Java trong ba thư mục con, như thể hiện trong hình trên. Thư mục `com.example.hello.helloworld` (hoặc tên miền bạn đã chỉ định) chứa tất cả các tệp cho một gói ứng dụng. Hai thư mục còn lại được sử dụng để kiểm tra và được mô tả trong một bài học khác. Đối với ứng dụng Hello World, chỉ có một gói duy nhất và nó chứa `MainActivity.java`. Tên của Hoạt động (màn hình) đầu tiên mà người dùng nhìn thấy cũng khởi tạo tài nguyên trên toàn ứng dụng, thường được gọi là `MainActivity` (phần mở rộng tệp là bị bỏ qua trong khung `Project > Android`).

2. Mở rộng thư mục `res` và thư mục bố cục, rồi bấm đúp vào tệp `activity_main.xml` để mở nó trong trình chỉnh sửa bố cục.



Thư mục res chứa các tài nguyên như bố cục, chuỗi và hình ảnh. Một hoạt động thường là được liên kết với bố cục của các chế độ xem giao diện người dùng được xác định dưới dạng tệp XML. Tập tin này thường được đặt tên theo tên của nó Activity.

2.5 Khám phá thư mục tệp kê khai

Thư mục bảng kê khai chứa các tệp cung cấp thông tin cần thiết về ứng dụng của bạn cho Hệ thống Android mà hệ thống phải có trước khi có thể chạy bất kỳ mã nào của ứng dụng.

1. Mở rộng thư mục bảng kê khai.
2. Mở tệp AndroidManifest.xml.

Tệp AndroidManifest.xml mô tả tất cả các thành phần của ứng dụng Android. Tất cả các thành phần của một ứng dụng, chẳng hạn như mỗi Hoạt động, phải được khai báo trong tệp XML này. Trong các bài học khóa học khác, bạn sẽ sửa đổi tệp này để thêm các tính năng và quyền tính năng. Để biết phần giới thiệu, hãy xem Tổng quan về tệp kê khai ứng dụng.

Nhiệm vụ 3: Sử dụng thiết bị ảo (giả lập)

Trong nhiệm vụ này, bạn sẽ sử dụng trình quản lý Thiết bị ảo Android (AVD) để tạo một thiết bị ảo (cũng được gọi là trình mô phỏng) mô phỏng cấu hình cho một loại thiết bị Android cụ thể và sử dụng thiết bị ảo đó để chạy ứng dụng. Lưu ý rằng Trình mô phỏng

Android có các yêu cầu bổ sung ngoài các yêu cầu hệ thống cơ bản dành cho Android Studio.

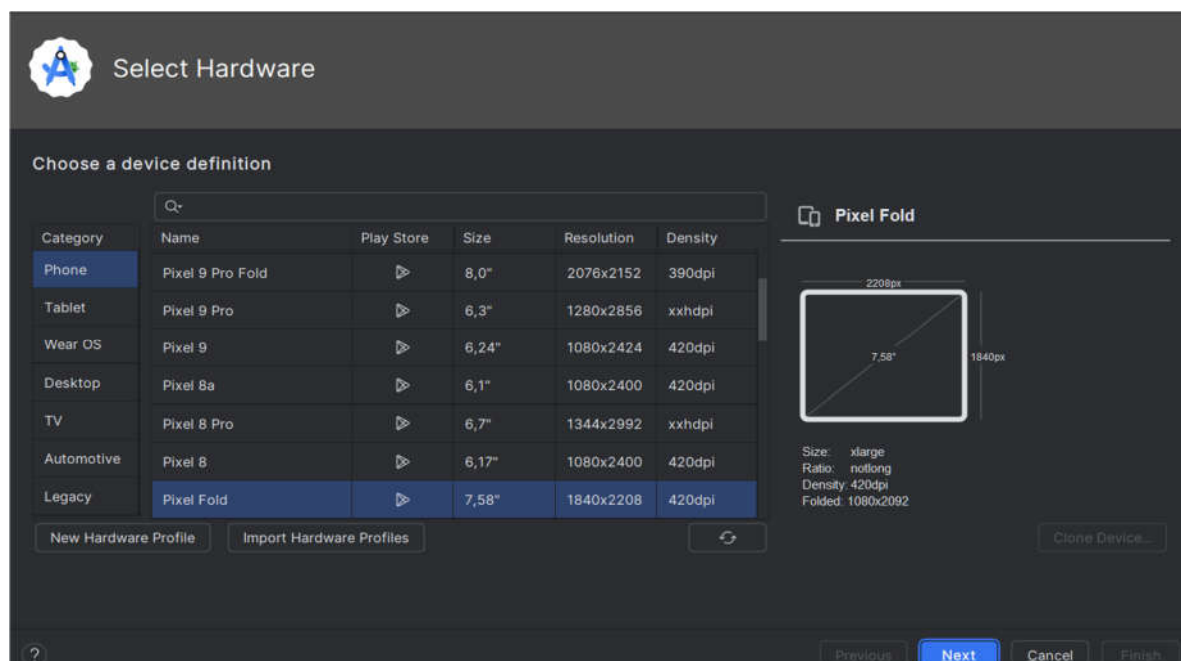
Sử dụng Trình quản lý AVD, bạn xác định các đặc điểm phần cứng của thiết bị, cấp độ API, bộ lưu trữ, skin và các thuộc tính khác và lưu nó dưới dạng thiết bị ảo. Với thiết bị ảo, bạn có thể thử nghiệm ứng dụng trên cấu hình thiết bị khác nhau (chẳng hạn như máy tính bảng và điện thoại) với các cấp độ API khác nhau mà không cần phải sử dụng các thiết bị vật lý.

3.1 Tạo thiết bị ảo Android (AVD)

Để chạy trình mô phỏng trên máy tính, bạn phải tạo cấu hình mô tả thiết bị ảo.

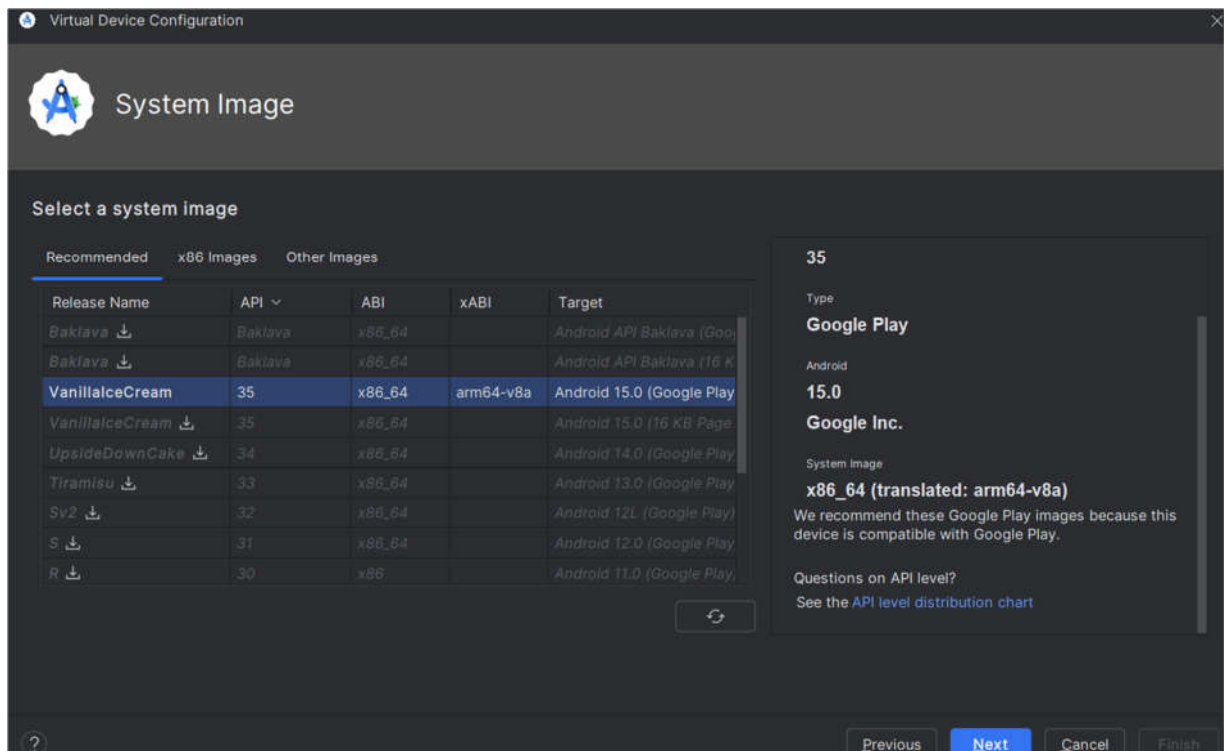
1. Trong Android Studio, chọn Tools > Device Manager, hoặc nhấp vào biểu tượng Trình quản lý AVD trên thanh công cụ. Màn hình Thiết bị ảo của bạn xuất hiện. Nếu bạn đã tạo thiết bị ảo, màn hình hiển thị chúng (như trong hình bên dưới); nếu không bạn sẽ thấy một danh sách trống.

2. Nhấp vào +Tạo thiết bị ảo. Cửa sổ Chọn phần cứng xuất hiện hiển thị danh sách thiết bị phần cứng được cấu hình sẵn. Đối với mỗi thiết bị, bảng cung cấp một cột cho kích thước hiển thị đường chéo (Size), độ phân giải màn hình tính bằng pixel (Resolution) và mật độ pixel (Tỉ trọng).



3. Chọn một thiết bị như Nexus 5x hoặc Pixel XL và nhấp vào Tiếp theo. Màn hình System image xuất hiện.

4. Nhấp vào tab Đề xuất nếu nó chưa được chọn và chọn phiên bản hệ thống Android để chạy trên thiết bị ảo (chẳng hạn như Oreo).



1.2) Giao diện người dùng tương tác đầu tiên

1.3) Trình chỉnh sửa bố cục

1.4) Văn bản và các chế độ cuộn

1.5) Tài nguyên có sẵn

Bài 2) Activities

2.1) Activity và Intent

2.2) Vòng đời của Activity và trạng thái

2.3) Intent ngầm định

Bài 3) Kiểm thử, gỡ lỗi và sử dụng thư viện hỗ trợ

3.1) Trình gỡ lỗi

3.2) Kiểm thử đơn vị

3.3) Thư viện hỗ trợ

CHƯƠNG 2. TRẢI NGHIỆM NGƯỜI DÙNG

Bài 1) Tương tác người dùng

1.1) Hình ảnh có thể chọn

1.2) Các điều khiển nhập liệu

1.3) Menu và bộ chọn

1.4) Điều hướng người dùng

1.5) RecyclerView

Bài 2) Trải nghiệm người dùng thú vị

2.1) Hình vẽ, định kiểu và chủ đề

2.2) Thẻ và màu sắc

2.3) Bố cục thích ứng

Bài 3) Kiểm thử giao diện người dùng

3.1) Espresso cho việc kiểm tra UI

CHƯƠNG 3. LÀM VIỆC TRONG NỀN

Bài 1) Các tác vụ nền

1.1) AsyncTask

1.2) AsyncTask và AsyncTaskLoader

1.3) Broadcast receivers

Bài 2) Kích hoạt, lập lịch và tối ưu hóa nhiệm vụ nền

2.1) Thông báo

2.2) Trình quản lý cảnh báo

2.3) JobScheduler

CHƯƠNG 4. LƯU DỮ LIỆU NGƯỜI DÙNG

Bài 1) Tùy chọn và cài đặt

1.1) Shared preferences

1.2) Cài đặt ứng dụng

Bài 2) Lưu trữ dữ liệu với Room

2.1) Room, LiveData và ViewModel

2.2) Room, LiveData và ViewModel