Android là gì

Android là một hệ điều hành mã nguồn mở dựa trên nhân Linux, được phát triển bởi Google chủ yếu dành cho các thiết bị di động như điện thoại thông minh, máy tính bảng, đồng hồ thông minh, TV và thậm chí cả ô tô.

**🔹 Đặc điểm chính của Android**

✔ **Mã nguồn mở**: Được phát triển trên nền tảng Linux, cho phép các nhà sản xuất và lập trình viên tùy chỉnh.  
✔ **Được hỗ trợ bởi Google**: Liên tục được cập nhật và cải tiến qua các phiên bản mới.  
✔ **Hệ sinh thái rộng lớn**: Hỗ trợ hàng triệu ứng dụng trên Google Play Store.  
✔ **Đa thiết bị**: Có thể chạy trên nhiều loại phần cứng khác nhau từ điện thoại đến TV, ô tô (Android Auto) và thiết bị đeo tay (Wear OS).  
✔ **Ngôn ngữ lập trình phổ biến**: Hỗ trợ Java, Kotlin (chính thức), C++, Python và nhiều ngôn ngữ khác.

**🔹 Các thành phần chính của Android**

📌 **Kernel (Nhân Linux)**: Cung cấp các dịch vụ hệ thống cơ bản như quản lý bộ nhớ, tiến trình và thiết bị.  
📌 **Android Runtime (ART)**: Thay thế Dalvik VM trước đây, giúp tăng hiệu suất ứng dụng.  
📌 **Framework**: Cung cấp API giúp lập trình viên dễ dàng phát triển ứng dụng.  
📌 **Ứng dụng**: Hệ điều hành hỗ trợ các ứng dụng của bên thứ ba chạy trên Android.

**🔹 Một số phiên bản Android quan trọng**

📌 Android 4.4 (KitKat) - Cải tiến hiệu suất trên thiết bị cấu hình thấp.  
📌 Android 5.0 (Lollipop) - Giới thiệu Material Design.  
📌 Android 6.0 (Marshmallow) - Hỗ trợ quyền truy cập ứng dụng.  
📌 Android 9.0 (Pie) - AI tối ưu hóa pin.  
📌 Android 10 - Chế độ tối Dark Mode.  
📌 Android 12 - Thiết kế Material You.  
📌 Android 14 (mới nhất) - Cải tiến bảo mật và hiệu năng.

Bạn đang quan tâm đến khía cạnh nào của Android? Lập trình ứng dụng, hệ điều hành hay phần cứng?

Cấu trúc các thư mục trong project Android Studio

📂 MyAndroidProject/

├── 📂 .gradle/ # Thư mục hệ thống của Gradle

├── 📂 .idea/ # Cấu hình Android Studio

├── 📂 app/ # Thư mục chính của ứng dụng

│ ├── 📂 manifests/ # Chứa AndroidManifest.xml

│ ├── 📂 java/ # Chứa mã nguồn Java/Kotlin

│ ├── 📂 res/ # Chứa tài nguyên (layout, hình ảnh, v.v.)

│ └── 📜 build.gradle # Cấu hình build cho module app

├── 📂 gradle/ # Thư mục Gradle Wrapper

├── 📜 build.gradle # Cấu hình Gradle toàn bộ project

├── 📜 settings.gradle # Cấu hình module trong project

└── 📜 gradle.properties # Cấu hình Gradle bổ sung

AndroidManifest là gì

**AndroidManifest.xml** là một tệp quan trọng trong mọi ứng dụng Android, giúp hệ điều hành hiểu được:  
✅ Tên ứng dụng, biểu tượng, chủ đề  
✅ Các **Activity, Service, BroadcastReceiver**  
✅ Các quyền yêu cầu (ví dụ: Internet, Camera)  
✅ Cấu hình màn hình chính (MainActivity)

## 🎯 **Tóm tắt**

📌 **AndroidManifest.xml giúp Android hiểu cấu trúc ứng dụng**  
📌 **Chứa thông tin về Activity, Service, quyền truy cập, v.v.**  
📌 **Mọi ứng dụng Android đều phải có AndroidManifest.xml**

Gradle là gì

**Gradle** là một công cụ **tự động hóa build** (build automation) được sử dụng trong **Android Studio** để:  
✅ **Quản lý dependencies** (các thư viện)  
✅ **Biên dịch mã nguồn** Kotlin/Java/XML  
✅ **Tạo tệp APK/AAB** để cài đặt lên thiết bị Android  
✅ **Tự động hóa quá trình build** (kiểm tra, đóng gói, ký ứng dụng)

## 🎯 **Tóm lại**

✅ **Gradle giúp quản lý build và dependencies trong Android Studio**  
✅ **Có 2 tệp build.gradle.kts: một cho toàn project, một cho từng module**  
✅ **Gradle tự động tải thư viện, biên dịch & đóng gói ứng dụng Android**

Đơn vị đo trong Android: dp và sp

## 📏 **Đơn vị đo trong Android:** dp **và** sp

Trong Android, để thiết kế giao diện **đồng nhất trên mọi thiết bị**, chúng ta không dùng **px** (pixel) mà sử dụng **dp (density-independent pixels) và sp (scale-independent pixels)**.

## 1️⃣ dp **(Density-independent Pixels) - Đơn vị độc lập với mật độ màn hình**

📌 **Dùng để đo kích thước layout (width, height, margin, padding,...)**  
📌 **Giúp giao diện hiển thị đồng nhất trên các màn hình có mật độ DPI khác nhau**

### 🔹 ****Công thức chuyển đổi**** dp ****sang**** px

px = dp \* (dpi / 160)

## 2️⃣ sp **(Scale-independent Pixels) - Đơn vị độc lập với tỷ lệ phóng to văn bản**

📌 **Dùng để đo kích thước chữ (textSize)**  
📌 **Tự động thay đổi kích thước theo cài đặt phóng to chữ của người dùng**

### 🔹 ****Công thức chuyển đổi**** sp ****sang**** px

px = sp \* (dpi / 160) \* scaleFactor  
(**scaleFactor** thay đổi theo cài đặt kích thước font của người dùng)

Khai báo các giá trị Resource: dimen, color, font-family, image, drawable,..

## 1️⃣ dimen.xml **- Khai báo kích thước (**dp**,** sp**)**

📌 **Dùng để khai báo giá trị dp cho layout và sp cho chữ**  
📍 **Vị trí:** res/values/dimens.xml

## 2️⃣ colors.xml **- Khai báo màu sắc**

📌 **Dùng để định nghĩa màu sắc theo mã HEX hoặc dạng rgb, rgba**  
📍 **Vị trí:** res/values/colors.xml

## 3️⃣ font **- Khai báo font chữ (**font-family**)**

📌 **Dùng để khai báo font chữ trong thư mục res/font/**  
📍 **Vị trí:** res/font/roboto.xml

## 4️⃣ image **- Khai báo hình ảnh**

📌 **Tất cả ảnh đặt trong thư mục res/drawable/ hoặc res/mipmap/**  
📍 **Vị trí:** res/drawable/logo.png hoặc res/mipmap/ic\_launcher.png

## 5️⃣ drawable **- Khai báo background, shape, vector**

📌 **Dùng để tạo hình ảnh, nền, hoặc hiệu ứng động**  
📍 **Vị trí:** res/drawable/bg\_button.xml

## 🎯 **Tóm lại**

| **Loại Resource** | **Vị trí** | **Chức năng** |
| --- | --- | --- |
| **dimen.xml** | res/values/dimens.xml | Định nghĩa kích thước (dp, sp) |
| **colors.xml** | res/values/colors.xml | Định nghĩa màu sắc |
| **font-family** | res/font/ | Quản lý font chữ |
| **image** | res/drawable/ hoặc res/mipmap/ | Lưu trữ hình ảnh PNG, JPG, SVG |
| **drawable.xml** | res/drawable/ | Định nghĩa hình dạng (shape), hiệu ứng (gradient) |

Activity trong Android là gì

**Activity** là một **thành phần (component) chính trong Android**, đại diện cho **một màn hình giao diện người dùng**. Nó là nơi người dùng có thể tương tác với ứng dụng.

### 📌 ****Đặc điểm của Activity****

✅ Mỗi ứng dụng có thể có nhiều Activity.  
✅ Một Activity thường chứa **giao diện (UI)** gồm các View như Button, TextView, RecyclerView,...  
✅ Một ứng dụng có **một Activity chính** (MainActivity) được khởi chạy đầu tiên khi ứng dụng mở.  
✅ Các Activity có thể liên kết với nhau bằng **Intent**.

## 🔄 **Vòng đời của Activity**

Activity trong Android có vòng đời được quản lý bởi hệ thống, gồm các trạng thái sau:

### 🏷 ****1. Trạng thái và phương thức vòng đời chính****

| **Trạng thái** | **Phương thức** | **Mô tả** |
| --- | --- | --- |
| **Khởi tạo** | onCreate() | Được gọi khi Activity được tạo. |
| **Hiển thị** | onStart() | Activity xuất hiện trên màn hình. |
| **Đang chạy** | onResume() | Người dùng có thể tương tác với Activity. |
| **Bị dừng tạm thời** | onPause() | Activity bị che bởi một Activity khác. |
| **Bị ẩn hoàn toàn** | onStop() | Activity không còn hiển thị nhưng chưa bị hủy. |
| **Bị hủy** | onDestroy() | Activity bị hủy để giải phóng bộ nhớ. |

📍 **Sơ đồ vòng đời Activity**  
📌 **Trình tự thực thi:**

scss

Sao chépChỉnh sửa

onCreate() → onStart() → onResume() → (Hoạt động) → onPause() → onStop() → onDestroy()

💡 **Lưu ý:**

* Nếu Activity bị che bởi một Activity khác → gọi onPause()
* Nếu người dùng quay lại Activity → gọi onResume()
* Nếu Activity bị hủy hoàn toàn → gọi onDestroy()

🔄 **Chuyển đổi giữa các Activity bằng Intent**

Các view cơ bản trong Android

+ ViewGroup: LinearLayout, RelativeLayout, ConstraintLayout, ...

+ View: TextView, EditText, ImageView, View, ...

**🏗 1. ViewGroup – Nhóm chứa các View**

📌 **ViewGroup** là các thành phần dùng để **chứa và sắp xếp** các **View con** bên trong nó.  
📌 Một số **ViewGroup phổ biến**:

| **Loại ViewGroup** | **Mô tả** |
| --- | --- |
| LinearLayout | Sắp xếp các View theo hàng ngang (horizontal) hoặc dọc (vertical). |
| RelativeLayout | Sắp xếp các View dựa trên vị trí của các View khác. |
| ConstraintLayout | Cho phép thiết kế linh hoạt, tối ưu hiệu suất. |
| FrameLayout | Chỉ chứa một View, phù hợp cho giao diện đơn giản. |
| ScrollView | Cho phép cuộn nội dung nếu vượt quá kích thước màn hình. |
| RecyclerView | Dùng để hiển thị danh sách dữ liệu lớn với hiệu suất cao. |

**🖼 2. View – Các Thành Phần Cơ Bản của UI**

📌 **View** là thành phần hiển thị trực tiếp trên giao diện.  
📌 Một số **View phổ biến**:

| **Loại View** | **Mô tả** |
| --- | --- |
| TextView | Hiển thị văn bản. |
| EditText | Ô nhập dữ liệu văn bản. |
| Button | Nút bấm. |
| ImageView | Hiển thị hình ảnh. |
| CheckBox | Hộp kiểm tra (check/uncheck). |
| RadioButton | Nút chọn một trong nhiều lựa chọn. |
| Switch | Công tắc bật/tắt. |
| SeekBar | Thanh kéo giá trị. |
| ProgressBar | Hiển thị tiến trình. |