**TRƯỜNG ĐẠI HỌC HÀNG HẢI VIỆT NAM  
KHOA CÔNG NGHỆ THÔNG TIN**

**-----\*\*\*-----**

****

**BÁO CÁO BÀI TẬP LỚN**

**HỌC PHẦN “LẬP TRÌNH THIẾT BỊ DI ĐỘNG”**

***Đề tài:***

***TÌM HIỂU VỀ HỆ ĐIỀU HÀNH ANDROID***

***GVHD: Lê Bá Dũng***

***Sinh viên thực hiện: Đinh Thị Mây - Mã SV: 85718***

*Hải Phòng, tháng 2 năm 2023*

***TÌM HIỂU VỀ HỆ ĐIỀU HÀNH ANDROID***

# Android là gì?

* Android là một hệ điều hành mã nguồn mở được sử dụng trên các thiết bị di động, chẳng hạn như điện thoại di động và máy tính bảng.
* Mã nguồn mở là phần mềm có bộ source code mà bất kì ai cũng có thể dowload về, sửa đổi hoặc nâng cấp thêm cho nó những tính năng vượt trội khác.
* Ứng dụng Android thực thi trong quy trình riêng của nó và phiên bản riêng của máy ảo Dalvik( DVM) hoặc Android Run Time( ART).
* Ưu điểm:
* Kho ứng dụng đa dạng: hiện nay, trên CH Play đã có tổng cộng trên 3 triệu ứng dụng khác nhau. Mọi ứng dụng phục vụ công việc, học tập và giải trí đều có thể được tìm thấy ở trên đây.
* Mẫu mã đa dạng: có rất nhiều hãng sản xuất điện thoại đều sử dụng hệ điều hành Android cho thiết bị của hãng. Điều đó giúp mang đến nhiều mẫu điện thoại hấp dẫn khác nhau với đa dạng phân khúc giá, giúp mọi người dùng dễ dàng tiếp cận với thiết bị Android hơn.
* Có thể mở rộng bộ nhớ bằng thẻ nhớ: bên cạnh bộ nhớ trong của máy, ta có thể mở rộng bộ nhớ bằng việc gắn thêm thẻ SD dung lượng cao.
* Khả năng tùy biến cao: hệ điều hành Android hoạt động với bản chất mã nguồn mở. Do đó, bất kì ai cũng có thể lấy mã nguồn của hệ điều hành này. Vì vậy, các nhà sản xuất và các lập trình viên có thể tùy biến chỉnh sửa mang đến phiên bản tốt nhất dành cho điện thoại thông minh của hãng.
* Được nhiều người dùng ưa chuộng: giao diện của Android được đánh giá là dễ dàng sử dụng, không tốn nhiều thời gian để làm quen. Đồng thời nếu như gặp bất kì vấn đề nào trong khi sử dụng, bạn cũng sẽ được hỗ trợ vô cùng nhiệt tình từ cộng đồng Android.
* Nhược điểm của hệ điều hành Android:
* Nhiều ứng dụng chạy ngầm: khả năng tối ưu hóa bộ nhớ Ram của Android có phần kém hơn IOS. Điều đó khiến nhiều ứng dụng vẫn chạy ngầm dẫn đến máy bị chậm hoặc thậm chí bị đơ.
* Một số ứng dụng chưa được tối ưu hóa tốt: vì nhiều hãng sản xuất điện thoại khác nhau đề cùng chạy Android nên hệ điều hành chưa được tối ưu hóa tốt phù hợp với từng mẫu mã khác nhau. Điều đó khiến các ứng dụng có thể gặp lỗi, ví dụ như không hiển thị được toàn màn hình.
* Một số ứng dụng có chất lượng kém: một số ứng dụng trên CH Plau có chất lượng kém. Đi kèm với đó là rất nhiều quảng cáo gây khó chịu cho người dùng. Điều này ản hưởng đến tiến độ xử lí công việc cũng như thời gian giải trí của người dùng.
* Virus dễ dàng xâm nhập hơn: nếu không cài đặt sẵn phần mềm diệt virus thì một số ứng dụng tải xuống từ CH Play có thể khiến thiết bị của bạn bị nhiễm virus.

# Các tính năng của kiến trúc Android?

* Kiến trúc Android đề cập đến các lớp khác nhau trong ngăn xếp Android. Nó bao gồm các hệ điều hành, phần mềm trung gian và các ứng dụng. Mỗi lớp trong kiến trúc Android cung cấp các dịch vụ khác nhau cho lớp ngay phía trên nó.
* Có 5 lớp trong Android:
* Linux Kernel: chịu trách nhiệm về trình điều khiển thiết bị, quản lí thiết bị, quản lí bộ nhớ, quản lí năng lượng và truy cập tài nguyên.
* Libraries: thư viện có trình duyệt web nguồn mở WebKit, thư viện nổi tiếng libc, thư viện để phát và ghi lại âm thanh và video, cơ sở dữ liệu SQLite để chia sẻ dữ liệu và lưu trữ ứng dụng, thư viện SSL để bảo mật internet…
* Android Runtime: có các thư viện cốt lõi cùng với DVM( máy ảo Dalvik) hoặc ART( thời gian chạy Android) dưới dạng thời gian chạy hữu ích để chạy ứng dụng Android. DVM được tối ưu hóa cho thiết bị di động. DVM cung cấp hiệu năng nhanh và tiêu tốn ít bộ nhớ hơn.
* Android Frameword: bao gồm các API Android như UI( giao diện người dùng), tài nguyên, nhà cung cấp nội dung( dữ liệu), vị trí, điện thoại và quy trình quản lí gói. Nó cung cấp các giao diện và các lớp để phát triển các ứng dụng Android.
* Android Applications: các ứng dụng gia đình, trò chơi, danh bạ, cài đặt, trình duyệt… sử dụng khung Android sẽ tận dụng thời gian chạy và thư viện của Android.
* Các tính năng nổi bật của Android:
* Giao diện người dùng mặc định của Android sử dụng các thao tác trực tiếp như chạm, vuốt và kéo để bắt đầu giao tiếp. Thiết bị cung cấp phản hồi xúc giác cho người dùng thông qua các cảnh báo như rung để phản hồi các hành động. Ví dụ: nếu người dùng nhấn vào nút điều hướng, thiết bị sẽ rung.
* Khi người dùng khởi động thiết bị, hệ điều hành Android sẽ hiển thị màn hình chính, đây là trung tâm điều hướng chính cho thiết bị Android, bao gồm các tiện ích và biểu tượng ứng dụng. Widget là màn hình thông tin tự động cập nhật nội dung như thời tiết hoặc tin tức. Màn hình chính có thể khác nhau tùy theo nhà sản xuất của thiết bị. Người dùng cũng có thể chọn các chủ đề khác nhau cho màn hình chính thông qua các ứng dụng của bên thứ ba trên Google Play.
* Thanh trạng thái ở phía trên cùng màn hình chính hiển thị các thông tin và trạng thái kết nối của thiết bị: mạng wifi đang kết nối, sóng 3G, cường độ sóng điện thoại. Người dùng có thể vuốt thanh trạng thái theo chiều đi xuống để xem màn hình thông báo.
* Hệ điều hành Android cũng sở hữu các tính năng giúp tiết kiệm pin. Hệ điều hành tạm dừng các ứng dụng không được sử dụng để tiết kiệm pin và mức sử dụng CPU. Android có các tính năng quản lý bộ nhớ giúp tự động đóng các trình xử lý không hoạt động lưu trong bộ nhớ.

# Danh sách các ngôn ngữ được sử dụng để xây dựng Android?

* Các ngôn ngữ lập trình phổ biến nhất có thể được sử dụng để phát triển ứng dụng trong Android là: java, kotlin, C#, python…
* Java: là điểm khởi đầu cho các nhà phát triển và được sử dụng bởi phần lớn những người làm việc trong lĩnh vực phát triển Android. Eclipse, NetBeans và IntelliJ IDE là những IDE phổ biến nhất được sử dụng để phát triển ứng dụng Android bằng java.
* Kotlin: Kotlin là một ngôn ngữ lập trình đa nền tảng tương đối mới, hiện đại, an toàn và hướng đối tượng sử dụng để phát triển ứng dụng Android. IDE được sử dụng với kotlin là Android studio, Eclipse IDE…
* C#: các nhà phát triển có thể xây dựng các ứng dụng di động IOS và Android gốc bằng cách sử dụng ngôn ngữ C#. Visual studio là công cụ tốt nhất để phát triển ứng dụng Android bằng C#.
* Python: là ngôn ngữ lập trình động và hướng đối tượng. Nó rất phổ biến trong học máy. Pydroid 3, Dcoder, trình chỉnh sửa mã spck là một số trình chỉnh sửa mã cho Python.
* Các ngôn ngữ khác có thể được sử dụng trong quá trình phát triển Android là C++, HTML 5.C4droid, CppDroid, AIDE… là các IDE dành cho C++. Acode, trình chỉnh sửa mã spck… là những ví dụ về IDE được sử dụng với HTML.

# Hoạt động của Android?

* Hoạt động trong java là một hoạt động đại diện cho một màn hình duy nhất với giao diện người dùng giống như cửa sổ hoặc khung của java. Hoạt động Android là lớp con của lớp ContextThemeWrapper.
* Với ngôn ngữ C, C++ hoặc Java thì ta thấy rằng chương trình bắt đầu từ main() chức năng. Theo cách tương tự, hệ thống Android khởi chạy chương trình của mình bằng Activity bắt đầu với một cuộc gọi trên phương thức gọi lại onCreate(). Có một chuỗi các phương thức gọi lại để khởi động một hoạt động và một chuỗi các phương thức gọi lại chia nhỏ một hoạt động như được hiển thị trong biểu đồ vòng đời.

# Dịch vụ trong Android là gì?

* Dịch vụ là một thành phần ứng dụng tạo điều kiện cho ứng dụng chạy trong nền để thực hiện các hoạt động chạy dài mà không có sự tương tác của người dùng.
* Một dịch vụ có thể chạy liên tục trong nền ngay cả khi ứng dụng đã đóng hoặc ngay cả sau khi người dùng chuyển sang ứng dụng khác.

# Phân biệt hoạt động với dịch vụ?

* Các hoạt động có thể bị chấm dứt hoặc đóng bất cứ lúc nào người dùng muốn.
* Các dịch vụ được thiết kế để chạy trong nền và chúng có thể hoạt động độc lập. Hầu hết các dịch vụ đều chạy liên tục, bất kể có một số hoạt động nhất định hay không có hoạt động nào được thực hiện.

# Google Android SDK là gì? Những công cụ nào được đặt trong Android SDK?

* Google Android SDK là bộ công cụ được các nhà phát triển sử dụng để viết ứng dụng trên các thiết bị hỗ trợ Android.
* Các công cụ được đặt trong Android SDK được đưa ra dưới đây:
* Trình giả lập Android - Android Emulator là ứng dụng phần mềm giả lập thiết bị Android trên máy tính để bạn có thể chạy thử ứng dụng trên nhiều thiết bị và cấp API Android mà không cần có từng thiết bị vật lý.
* DDMS(Dịch vụ giám sát gỡ lỗi Dalvik) - Đây là công cụ gỡ lỗi từ bộ công cụ phát triển phần mềm Android (SDK) cung cấp các dịch vụ như tạo thông báo, giả mạo cuộc gọi, chụp ảnh màn hình, v.v.
* ADB(Android Debug Bridge) - Đây là một công cụ dòng lệnh được sử dụng để cho phép và kiểm soát giao tiếp với phiên bản giả lập.
* AAPT(Công cụ đóng gói tài sản Android) - Đây là một công cụ xây dựng cung cấp khả năng cho các nhà phát triển xem, tạo và cập nhật các kho lưu trữ tương thích với ZIP (zip, jar và apk).

# Công cụ của Bundle trong Android SDK là gì?

* Các gói được sử dụng để chuyển dữ liệu cần thiết giữa các hoạt động khác nhau của Android. Chúng giống như HashMap có thể lấy các loại dữ liệu tầm thường.

# Adapter trong Android là gì?

* Một Adapter là một đối tượng của một lớp cài đặt giao diện Adapter. Nó đóng vai trò như là 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.
* Trong Android, Adapter thường được sử dụng để liên kết dữ liệu với UI:
* BaseAdapter- Là lớp Adpater cơ sở cho các Adapter khác.
* ArrayAdapter- Một ListAdapter có thể quản lí một ListView chứa danh sách các phần tử có kiểu bất kì.
* Custom ArrayAdapter- Thường được sử dụng để hiển thị một danh sách tùy chỉnh.
* SimpleAdapter- là một Adapter đơn giản và dễ hiểu để ánh xạ dữ liệu vào những View được định nghĩa trong một tập tin XML.
* Custom SimpleAdapter- được sử dụng để hiển thị một danh sách tùy chỉnh để truy cập các mục con của List hoặc Grid.

# AAPT là gì?

* AAPT là viết tắt của Công cụ đóng gói tài sản Android. Nó là một công cụ xây dựng cung cấp khả năng cho các nhà phát triển xem, tạo và cập nhật các kho lưu trữ tương thích với ZIP (zip, jar và apk). Nó phân tích cú pháp, lập chỉ mục và biên dịch tài nguyên thành định dạng nhị phân được tối ưu hóa cho nền tảng Android.

# Điểm truy cập wifi di động là gì?

* Điểm truy cập Wi-Fi di động cho phép bạn chia sẻ kết nối internet di động của mình với các thiết bị không dây khác. Ví dụ: sử dụng điện thoại Android của bạn làm điểm truy cập Wi-Fi, bạn có thể sử dụng máy tính xách tay của mình để kết nối Internet bằng điểm truy cập đó.

# Android Debug Bridge( ADB) là gì?

* ADB là viết tắt của cụm từ *Android Debug Bridge*, là một chương trình dạng dòng lệnh( cmd) giúp giao tiếp giữa máy tính với các thiết bị chạy hệ điều hành *Android* như điện thoại, máy tính bảng…
* Hiểu một cách đơn giản, ADB chính là cầu nối, cho phép bạn truy cập vào Unix shell( *trình thông dịch lệnh*) để điều khiển thiết bị Android một cách tổng thể và chuyên sâu, không cần phải thao tác thủ công trên thiết bị.
* ADB sử dụng mô hình máy khách- máy chủ và trong hầu hết các trường hợp được sử dụng qua kết nối USB. Cũng có thể sử dụng nó thông qua wifi( ADB không dây). Ta không cần phải cài đặt gì trên thiết bị Android của mình, vì ADB đã được tích hợp vào hệ điều hành Android. Nó thường được truy cập thông qua giao diện dòng lệnh từ PC, nơi cài đặt *SDK* Android đầy đủ hoặc Mini ADB hoặc ADB Essential.

# DDMS là gì?

* DDMS( *Dalvik Debug Monitor Server*) là một công cụ sửa lỗi trong nền tảng Android.
* Dịch vụ giám sát gỡ lỗi Dalvik cho phép các nhà phát triển phát hiện ra các lỗi trong các ứng dụng chạy trên trình giả lập hoặc thiết bị Android thực tế. Ví dụ: bằng cách sử dụng tính năng LogCat của DDMS, nhà phát triển có thể xem các thông báo nhật kí liên quan đến trạng thái của ứng dụng và thiết bị. LogCat có thể xác định chính xác số dòng chính xác xảy ra lỗi.
* Một tính năng DDMS khác, được gọi là Điều khiển trình mô phỏng, cho phép các nhà phát triển mô phỏng trạng thái và hoạt động của điện thoại. Ví dụ, nó có thể mô phỏng các loại mạng khác nhau như GPRS, EDGE và UTMS, có thể có các đặc điểm mạng khác nhau như tốc độ và độ trễ.
* Bên cạnh các tính năng được đề cập trước đó, Dịch vụ giám sát gỡ lỗi Dalvik cũng cung cấp LogCat, xử lý và thông tin trạng thái vô tuyến cũng như cuộc gọi đến, SMS và giả mạo dữ liệu vị trí.
* Công cụ gỡ lỗi này có thể được tích hợp vào IDE Eclipse bằng cách thêm trình cắm thêm ADT( *Công cụ phát triển Android*). Nếu không, nó có thể được truy cập từ dòng lệnh và sẽ tự động kết nối với bất kỳ trình giả lập đang chạy nào.

# AIDL là gì? Loại dữ liệu nào được AIDL hỗ trợ?

* AIDL (*Android Interface Definition Language*)- ngôn ngữ định nghĩa giao diện: là cách cho phép ta có thể định nghĩa một cách mà cả client và server( một ứng dụng đóng vai trò là server cho các ứng dụng khác đóng vai trò là *client* có thể truy cập tới) có thể giao tiếp với nhau thông qua truyền thông liên tiến trình Interprocess communication( IPC). Thông thường, trong Android một process không thể trực tiếp truy cập vào bộ nhớ của một tiến trình khác. Vì vậy để có thể các tiến trình có thể giao tiếp với nhau, chúng cần phân tách các đối tượng thành dạng đối tượng nguyên thủy( *primitive object*) mà hệ thống có thể hiểu được.
* AIDL chỉ thích hợp cho các trường hợp cần xử lí multithreading. Nguyên lí hoạt động:
* AIDL làm việc giống như là thành phần trung gian giữa client và server application.
* AIDL sẽ cho phép tạo một interface để cả client và service có thể giao tiếp thông qua nó.
* Client app sẽ tào một vài actions cụ thể và truyền data/ request tới server app để xử lí. Sau đó server sẽ trả về kết quả thích hợp cho client.
* Có thể truyền primitive type data hoặc custom object giữa các app với nhau.
* Với custom objectthì ta phải phân rã nó thành các thành phần dữ liệu nguyên thủy để system có thể đọc và sau đó AIDL sẽ marshalling lại object đó.
* Các loại dữ liệu được hỗ trợ bởi AIDL như sau:
* String.
* List.
* Map.
* CharSequence.
* Java data types( int, long, char, boolean).