**Bảng Phân Chia Công Việc:**

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thành Viên | Công Việc | Tình Trạng |
| 1 | Bùi Đức Thắng – 2121050756 - C | - Tìm tài liệu và sửa báo cáo  -Làm báo cáo  chương 1  -Thiết kế trương trình  -Viết trương trình | 100%  100%  100%  100% |
| 2 | Nguyễn Văn Thiện - 2121051573 | -Tìm tài liệu và sửa báo cáo  -Làm báo cáo  chương 2  -Thiết kế trương trình  -Viết trương trình | 100%  100%  100%  100% |
| 3 | Nguyễn Xuân Phúc - 2121050762 | -Tìm tài liệu  -Làm báo cáo  chương 3  -Viết trương trình | 100%  100%  100% |
| 4 | Nguyễn Văn Khôi - 2221050473 | -Tìm tài liệu  -Làm báo cáo  chương 4  -Viết trương trình | 100%  100%  100% |

**Mục lục**

[DANH MỤC HÌNH VẼ 4](#_bookmark0)

[DANH MỤC BẢNG BIỂU 4](#_bookmark1)

[DANH MỤC CÁC THUẬT NGỮ 5](#_bookmark2)

[LỜI MỞ ĐẦU 6](#_bookmark3)

[CHƯƠNG I: TỔNG QUAN VỀ ĐỀ TÀI 7](#_bookmark4)

* 1. [Lý do chọn đề tài 7](#_bookmark5)
  2. [Mục tiêu của ứng dụng 7](#_bookmark7)
  3. [Phạm vi và đối tượng sử dụng 8](#_bookmark8)
  4. [Nội dung nghiên cứu 9](#_bookmark9)
  5. [Phương pháp nghiên cứu 10](#_bookmark10)
  6. [Kết luận và đề xuất hướng phát triển 11](#_bookmark11)

[CHƯƠNG II: CỞ SỞ LÝ THUYẾT 12](#_bookmark12)

* 1. [Tìm hiểu Android 12](#_bookmark13)
     1. [Đôi nét về Android 12](#_bookmark14)
  2. [Ngôn ngữ lập trình 12](#_bookmark15)
     1. [Java 12](#_bookmark16)

[2.2.2 Kotlin 15](#_bookmark18)

* 1. [Công nghệ phần mềm Android Studio 16](#_bookmark20)
     1. [Đôi nét về Android Studio 16](#_bookmark21)
     2. [Cài đặt môi trường Android Studio 17](#_bookmark23)

[CHƯƠNG III: CHIẾN LƯỢC THỰC HIỆN 21](#_bookmark29)

[CHƯƠNG IV: PHÂN TÍCH YÊU CẦU ỨNG DỤNG QUẢN LÝ FILE 23](#_bookmark30)

* 1. [Tương tác người dùng 23](#_bookmark31)
  2. [Thao tác của người dùng 23](#_bookmark32)
  3. [Nhu cầu từ người dùng 24](#_bookmark34)
  4. [Đối tượng hướng tới 25](#_bookmark35)
  5. [Thiết bị và nền tảng 25](#_bookmark36)

[CHƯƠNG V: THIẾT KẾ KIẾN TRÚC ỨNG DỤNG 26](#_bookmark37)

* 1. [Kiến trúc ứng dụng 26](#_bookmark38)
     1. [Giao diện người dùng (UI) 26](#_bookmark40)
     2. [Quản lý file 26](#_bookmark41)
     3. [Người dùng 26](#_bookmark42)
  2. [Thiết kế giao diện người dùng (UI) 27](#_bookmark43)
  3. [Giao diện sau khi chạy trên điện thoại Android 30](#_bookmark50)
  4. [Logic ứng dụng 32](#_bookmark51)
     1. [Quản lý tập tin và thư mục 32](#_bookmark52)
     2. [Tìm kiếm và sắp xếp 33](#_bookmark53)
     3. [Đồng bộ hóa dữ liệu 33](#_bookmark54)
     4. [Giao diện người dùng thân thiện 33](#_bookmark55)
     5. [Kiểm thử và đánh giá 33](#_bookmark56)
  5. [Những lưu ý cần thiết 33](#_bookmark57)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN TƯƠNG LAI 34](#_bookmark58)

[Kết luận 34](#_bookmark59)

[Hướng phát triển trong tương lai 34](#_bookmark60)

[TÀI LIỆU THAM KHẢO 37](#_bookmark61)

[PHỤ LỤC 38](#_bookmark62)

## DANH MỤC HÌNH VẼ

[Hình 1.1: Xu Hướng Phát Triển Của Người Dùng Di Động 7](#_bookmark6)

Hình 2.1: Hệ Điều Hành Android 12

[Hình 2.2: Ngôn Ngữ Lập Trình Java 12](#_bookmark17)

[Hình 2.3: Ngôn Ngữ Lập Trình Kotlin 15](#_bookmark19)

[Hình 2.4: Phần Mềm Android Studio 16](#_bookmark22)

[Hình 2.5: Mở Project hoặc Mở File 18](#_bookmark25)

[Hình 2.6: Khởi Tạo Giao Diện 19](#_bookmark26)

[Hình 2.7: Tạo Project Mới 19](#_bookmark27)

[Hình 2.8: Giao Diện Chính Của Android Studio 20](#_bookmark28)

Hình 3.1: Chiến Lược Thực Hiện 21

[Hình 4.1: File Dữ Liệu 24](#_bookmark33)

[Hình 5.1: Sơ Đồ Kiến Trúc Hệ Thống 26](#_bookmark39)

Hình 5.2: Giao Diện Người Dùng 27

[Hình 5.3: Giao Diện Bắt Đầu Khởi Tạo 27](#_bookmark44)

[Hình 5.4: Các Chức Năng Cơ Bản 28](#_bookmark45)

[Hình 5.5: Sửa Lỗi 28](#_bookmark46)

[Hình 5.6: Qúa Trình Chỉnh Sửa 29](#_bookmark47)

[Hình 5.7: Qúa Trình Chỉnh Sửa 29](#_bookmark48)

[Hình 5.8: Thêm Các Chức Năng Mới 30](#_bookmark49)

Hình 5.9: Các Chức Năng Chính 31

Hình 5.10: Các Chức Năng Cơ Bản 31

Hình 5.11: Giao Diện 32

Hình 5.12: Giao Diện Chính 32

## DANH MỤC BẢNG BIỂU

[Bảng 1.1: Cấu Hình Cài Đặt 17](#_bookmark24)

## DANH MỤC CÁC THUẬT NGỮ

|  |  |
| --- | --- |
| **Thuật Ngữ Viết Tắt** | **Giải Thích** |
| CNTT | Công Nghệ Thông Tin |
| File | Tệp tin hoặc dữ liệu lưu trữ trên máy tính hoặc thiết bị điện tử. |
| Copy | Sao chép, tạo một bản sao của dữ liệu hoặc tệp tin |
| Rename | Đổi tên, thay đổi tên của một tệp tin hoặc thư mục |
| Delete | Xóa, loại bỏ một tệp tin hoặc thư mục khỏi hệ thống |
| Cut | Cắt, di chuyển dữ liệu hoặc tệp tin từ một vị trí và chờ để dán ở vị trí khác |
| Select | Chọn, chọn một hoặc nhiều mục từ danh sách hoặc giao diện |
| AI | Trí tuệ nhân tạo |
| ML | Học máy |
| Notification | Thông báo, thông điệp hoặc cảnh báo |
| API | Giao diện lập trình ứng dụng |

## LỜI MỞ ĐẦU

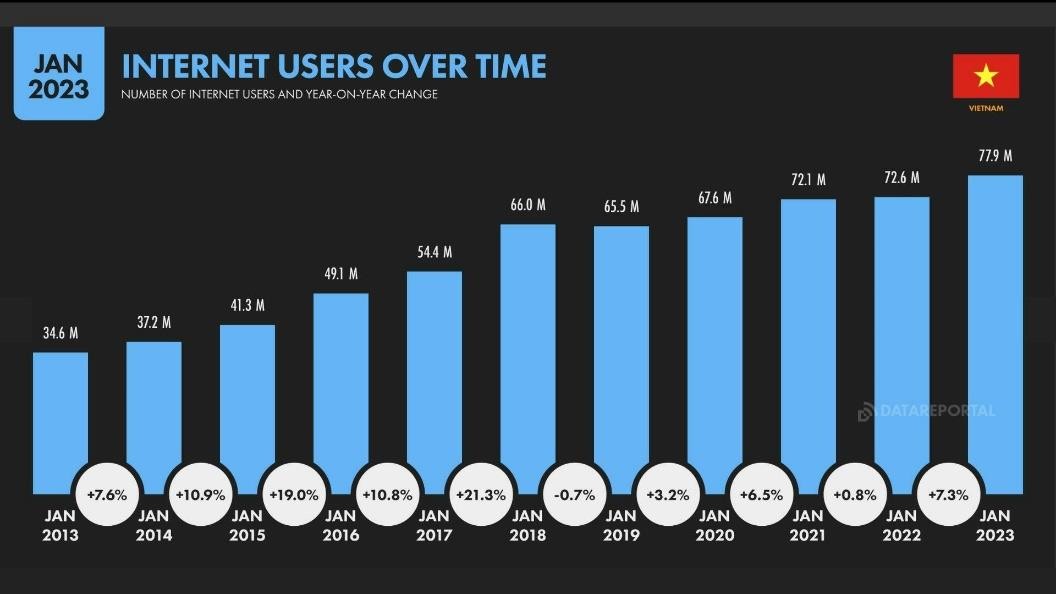
Thế giới ngày nay đã có nhiều phát triển mạnh mẽ về công nghệ thông tin (CNTT). Từ một tiềm năng thông tin đang trở thành một tài nguyên thực sự, trở thành một hàng hóa trong xã hội tạo ra sự thay đổi to lớn trong lực lượng sản xuất, cơ sở hạ tầng, cấu trúc kinh tế, tính chất lao động và cả cách thức quản lí trong các lĩnh vực của xã hội. Trong những năm gần đây nền CNTT nước ta đã có nhiều phát triển trong mọi lĩnh vực của cuộc sống cũng như trong lĩnh vực quản lí xã hội. Một trong những lĩnh vực mà máy tính được sử dụng nhiều nhất là các hệ thống thông tin quản lí nói chung. Sự phát triển của internet, internet đã đưa con người vào kỉ nguyên xa lộ thông tin trên mạng. Thông tin qua mạng internet con người có thể mua bán hàng hóa, trao đổi thông tin rộng rãi trên toàn cầu. Công nghệ đã thúc đẩy mạnh mẽ việc khai thác thông tin trên mạng với tính năng linh hoạt và dễ sử dụng nó.

Cùng với sự phát triển của ngành công nghệ thông tin thì máy tính càng trở nên gần gũi và thông dụng với cuộc sống của chúng ta. Khả năng hiệu quả hỗ trợ những côngviệc khó khan và phức tạp trong mọi lĩnh vực của cuộc sống nói chung và công tác quản lí trong các doanh nghiệp, cơ quan nói riêng như bảo mật thông tin, xem và chỉnh sửa thông tin một cách dễ dàng, hiệu quả và nhanh chóng. Đã có nhiều phần mềm được xấy dựng để phục vụ cho tính chất phực tạp trong mọi lĩnh vực của cuộc sống nói chung và công tác quản lý trong các doanh nghiệp và các cơ quan.

Trong suốt quá trình học tập, chúng em đã được các thầy cô giáo cung cấp và truyền đạt những kiến thức cần thiết trong lĩnh vưc công nghệ thông tin. Ngoài ra em còn được rèn luyện một tinh thần học tập, làm việc độc lập sáng tạo. Trong khuôn khổ một đề án môn học dưới sự chỉ bảo tận tình của thầy cô giáo, cùng sự tích lũy kiến thức của bản thân em đã xây dựng đề tài “**Xây dựng ứng dụng quản lý file trên nền tảng Android**”.

# CHƯƠNG I: TỔNG QUAN VỀ ĐỀ TÀI

## Lý do chọn đề tài

* + - Do nhu cầu ngày càng phát triển từ người dùng di động đã tạo ra một thị trường ứng dụng rộng lớn, đặc biệt là lĩnh vực quản lý file và các ứng dụng liên quan. Với lượng thông tin cá nhân và dữ liệu ngày càng gia tăng, việc tìm kiếm giải pháp quản lý thông tin hiệu quả là không thể tránh khỏi.

Hình 1.1: Xu Hướng Phát Triển Của Người Dùng Di Động

* + - Việc giải quyết thách thức này không chỉ đơn giản là đáp ứng nhu cầu, mà còn mang lại giá trị thực tế trong việc bảo vệ và quản lý an toàn dữ liệu cá nhân. Điều này không chỉ giúp tối ưu hóa không gian lưu trữ mà còn tăng cường tính an toàn và tiện lợi cho người dùng.
    - Thực hiện đề tài xây dựng ứng dụng quản lý file trên nền tảng Android không chỉ là một dự án, đề tài mà đó còn là một cơ hội để chúng em mở rộng kiến thức và kỹ năng của bản thân. Cũng như đáp ứng được những yêu cầu của người dùng hiện nay và tương lai

## Mục tiêu của ứng dụng

* + - Ứng dụng quản lý file đóng vai trò rất quan trọng trong việc quản lý và lưu trữ dữ liệu cá nhân. Việc nghiên cứu để xây dựng một ứng dụng quản lý file này sẽ tập trung vào quá trình lưu trữ, truy cập và chia sẻ dữ liệu. Nhóm nghiên cứu cũng tập trung vào việc đồng bộ hóa dữ liệu qua nhiều thiết bị, tương tác linh hoạt. Qua đó, mục tiêu là tạo ra một ứng dụng quản lý file không chỉ mạnh mẽ và hiệu quả mà còn đáp ứng đầy đủ các yêu cầu người dùng.
    - Giao diện dễ sử dụng:
      * Nghiên cứu về thiết kế giao diện người dùng để đảm bảo tính thân thiện, dễ sử dụng và tương tác hiệu quả.
      * Xác định các yếu tố cần thiết như thanh điều hướng, nút chức năng, và cách hiển thị thông tin để người dùng có thể dễ dàng tìm kiếm và quản lý file.
    - Tối ưu hóa quy trình làm việc:
      * Nghiên cứu về cách tối ưu hóa quy trình làm việc của người dùng khi tải lên, tải xuống, di chuyển, và xóa file.
      * Phát triển tính năng kéo và thả, gắn kết các lựa chọn nhanh, và cung cấp các công cụ tìm kiếm để tối ưu hóa thao tác.
    - Đồng bộ hóa dữ liệu:
      * Nghiên cứu về cách đồng bộ hóa dữ liệu một cách đơn giản
      * Phát triển cơ chế đồng bộ linh hoạt để đảm bảo rằng dữ liệu luôn được cập nhật và sẵn sàng sử dụng.
    - Kiểm thử và đánh giá chất lượng:
      * Tiến hành các bước kiểm thử chất lượng để đảm bảo ứng dụng hoạt động mượt mà, ổn định và đáp ứng đúng nhu cầu của người dùng.
    - Nghiên cứu phản hồi từ người dùng:
      * Thu thập ý kiến và phản hồi từ người dùng để cải thiện liên tục ứng dụng, điều chỉnh tính năng và đáp ứng đúng nhu cầu người dùng.
    - So sánh với các ứng dụng khác:
      * Nghiên cứu và so sánh với các ứng dụng báo thức khác trên thị trường để đánh giá và đặt ra những điểm mạnh, điểm yếu và nét độc đáo của

## Phạm vi và đối tượng sử dụng

* + - Phạm vi chức năng:
      * Xây dựng chức năng các chức năng cơ bản: Copy, Rename, Delete,…….
      * Tích hợp tính năng import dữ liệu từ hệ thống.
      * Xây dựng chức năng tạo thư mục
      * Mở/Chạy File
    - Phạm vi kỹ thuật:
      * Sử dụng Android Studio và ngôn ngữ lập trình Java để phát triển ứng dụng.
      * Tối ưu hóa hiệu suất và quản lý năng lượng của ứng dụng.
    - Phạm vi giao diện người dùng:
      * Phát triển giao diện người dùng thân thiện và dễ sử dụng.
      * Tạo trải nghiệm người dùng tích cực và linh hoạt.
    - Phạm vi kiểm thử và đánh giá:
      * Thực hiện kiểm thử chất lượng để đảm bảo tính ổn định và hiệu suất của ứng dụng.
      * Đánh giá sự hài lòng của người dùng thông qua các phản hồi và ý kiến.
    - Phạm vi so sánh và đối thoại:
      * So sánh ứng dụng với các sản phẩm tương tự trên thị trường.
      * Tìm hiểu ý kiến đánh giá từ người dùng và so sánh với các ứng dụng quản lý file khác.
    - Phạm vi thời gian:
      * Xác định một khung thời gian cụ thể cho quá trình phát triển, kiểm thử, và thu thập phản hồi.
      * Đảm bảo rằng mọi công việc được hoàn thành trong khoảng thời gian đề ra.
    - Phạm vi người dùng:
      * Định rõ đối tượng người dùng mục tiêu (ví dụ: người sử dụng điện thoại di động Android).
      * Phân tích nhu cầu và mong đợi của người dùng để tối ưu hóa trải nghiệm. Thay đổi đi và tôi ưu hơn và viết xây dưng ứng dụng quản lý file

## Nội dung nghiên cứu

* Nghiên cứu và phân tích các chức năng cơ bản của ứng dụng
  + Xác định nhu cầu cụ thể của đề tài đối với ứng dụng quản lý file, bao gồm các chức năng
  + Phân loại yêu cầu thành chức năng và phi chức năng để hiểu rõ về phạm vi và tính chất của ứng dụng.
  + Tìm hiểu về các vấn đề liên quan đến quản lý file như tìm kiếm, sắp xếp, và tương tác với tệp tin.
* Thiết kế hệ thống
  + Xây dựng kiến trúc tổng quan cho ứng dụng, bao gồm cấu trúc chung và mối quan hệ giữa các thành phần chính như server backend, client frontend, và cơ sở dữ liệu.
  + Tạo giao diện người dùng thân thiện, dễ sử dụng, và phản ánh đúng các chức năng cần thiết.
  + Bảo mật hệ thống bằng cách quản lý quyền truy cập và áp dụng các biện pháp mã hóa dữ liệu.
* Triển khai ứng dụng
  + Lựa chọn các công nghệ phù hợp, bao gồm ngôn ngữ lập trình, cơ sở dữ liệu, và framework.
  + Phát triển và triển khai mã nguồn và hệ thống lên môi trường thực tế.
  + Đảm bảo tính ổn định và hiệu suất của ứng dụng trong môi trường sản xuất.
* Thử nghiệm và đánh giá
  + Kiểm thử chức năng để đảm bảo rằng mọi chức năng của ứng dụng hoạt động đúng như mong đợi.
  + Đánh giá hiệu suất của ứng dụng dưới các điều kiện tải khác nhau.
  + Thu thập phản hồi từ người dùng để cải thiện trải nghiệm sử dụng và điều chỉnh các tính năng nếu cần thiết.

## Phương pháp nghiên cứu

+ Phân tích yêu cầu

* Phân tích và xác định rõ ràng những yêu cầu được đặt ra của đề tài này mong muốn đối với việc xây dựng ứng dụng quản lý file này .
* Chia thành hai loại yêu cầu chính là: yêu cầu chức năng và phi chức năng. Yêu cầu chức năng tập trung vào các tính năng cụ thể mà ứng dụng nên có, trong khi yêu cầu phi chức năng tập trung vào các khía cạnh như hiệu suất, bảo mật, và giao diện người dùng.

+ Thiết kế hệ thống

* Xây dựng một kiến trúc tổng quan cho ứng dụng, bao gồm việc xác định các thành phần chính của ứng dụng, cũng như mối quan hệ và tương tác giữa chúng. Điều này sẽ tạo ra một bức tranh tổng quan về cách hệ thống sẽ hoạt động.Tạo ra một giao diện người dùng thân thiện và dễ sử dụng, phản ánh đúng các chức năng cần thiết.
* Một phần quan trọng của việc xây dựng ứng dụng là thiết kế giao diện người dùng thân thiện và dễ sử dụng. Trong giai đoạn này, chúng ta sẽ tạo ra bản vẽ giao diện, bố cục, và phải đảm bảo rằng người dùng có thể tương tác với hệ thống một cách hiệu quả nhất có thể

+ Lựa chọn công nghệ

* + Xác định và lựa chọn các công nghệ phù hợp nhất đối với yêu cầu của ứng dụng. Điều này bao gồm quyết định về ngôn ngữ lập trình, cơ sở dữ liệu, framework, và các công nghệ khác liên quan đến triển khai và phát triển ứng dụng. Vì vậy sẽ yêu tiên sử dụng Android Studio và ngôn ngữ lập trình Java.
  + Triển khai mã nguồn và hệ thống lên môi trường thực tế. Quá trình này đòi hỏi sự chú ý đặc biệt đến việc đảm bảo tính ổn định và hiệu suất của ứng dụng khi hoạt động trong môi trường thực tế.

+ Thử nghiệm và đánh giá

* + Thực hiện các cuộc kiểm thử kỹ thuật để đảm bảo rằng mọi chức năng của ứng dụng hoạt động đúng như mong đợi. Các kịch bản thử nghiệm sẽ được xây dựng để kiểm tra từng chức năng cụ thể của hệ thống.
  + Tiến hành đánh giá hiệu suất của ứng dụng dưới các điều kiện tải khác nhau để đảm bảo rằng nó có thể đáp ứng đúng cách dưới áp lực sử dụng cao.

## Kết luận và đề xuất hướng phát triển

* Tổng hợp kết quả của quá trình xây dựng ứng dụng, so sánh chúng với yêu cầu ban đầu và mục tiêu đề ra trong chương tổng quan.
* Đánh giá mức độ thành công của ứng dụng dựa trên so sánh kết quả với mục tiêu ban đầu đã đặt ra trong giai đoạn tổng quan.
* Gợi ý các cải tiến và hướng phát triển tương lai của ứng dụng, có thể dựa trên phản hồi người dùng và xu hướng công nghệ mới.

# CHƯƠNG II: CỞ SỞ LÝ THUYẾT

## Tìm hiểu Android

* + 1. **Đôi nét về Android**

Android là hệ điều hành phát triển bởi Tổng công ty Android với sự hỗ trợ tài chính của Google từ năm 2003 và được Google chính thức mua lại vào năm 2005. Android là hệ điều hành dựa trên nền tảng Linux, chủ yếu dành cho các thiết bị có màn hình cảm ứng như: điện thoại, máy tính bảng…

Hiện nay Android không còn là hệ điều hành duy nhất trên thị trường nữa mà phải cạnh tranh với hệ điều hành iOS đến từ nhà Apple. Tuy nhiên, Android vẫn chiếm thị phần lớn trong ngành điện thoại.

## Ngôn ngữ lập trình

## Java

## Khái niệm



Hình 2.1: Hệ Điều Hành Android

* Khái niệm Java là một ngôn ngữ lập trình hướng đối tượng phổ biến, được sử dụng rộng rãi trong phát triển phần mềm, trang web, game, và ứng dụng di động. Ngôn ngữ này có nguồn gốc từ một dự án tên là Oak, được bắt đầu bởi James Gosling và đồng nghiệp tại Sun Microsystems vào năm 1991, với mục đích ban đầu là phát triển phần mềm cho các sản phẩm gia dụng.



Hình 2.2: Ngôn Ngữ Lập Trình Java

* Java chính thức được phát hành vào năm 1994 và sau đó, vào năm 2010, nó được Oracle mua lại từ Sun Microsystems. Một trong những đặc điểm nổi bật của Java là tiêu chí "Viết một lần, thực thi khắp nơi" (Write Once, Run Anywhere - WORA). Điều này có nghĩa là mã nguồn Java có thể chạy trên mọi nền tảng khác nhau thông qua một môi trường thực thi, với điều kiện là nền tảng đó hỗ trợ môi trường thực thi tương ứng.

## Đặc điểm của ngôn ngữ lập trình Java

**+** Tương tự C++, hướng đối tượng hoàn toàn

Trong quá trình tạo ra một ngôn ngữ mới phục vụ cho mục đích chạy được trên nhiều nền tảng, các kỹ sư của Sun MicroSystem muốn tạo ra một ngôn ngữ dễ học và quen thuộc với đa số người lập trình. Vì vậy họ đã sử dụng lại các cú pháp của C và C++.

Tuy nhiên, trong Java thao tác với con trỏ bị *lược bỏ* nhằm đảo bảo tính an toàn và dễ sử dụng hơn. *Các thao tác overload, goto hay các cấu trúc như struct và union cũng được loại bỏ khỏi Java.*

**+** Độc lập phần cứng và hệ điều hành

Một chương trình viết bằng ngôn ngữ Java có thể chạy tốt ở nhiều môi trường khác nhau. Gọi là khả năng “*cross-platform*”. Khả năng độc lập phần cứng và hệ điều hành được thể hiện ở 2 cấp độ là cấp độ mã nguồn và cấp độ nhị phân.

* **Ở cấp độ mã nguồn:** Kiểu dữ liệu trong Java nhất quán cho tất cả các hệ điều hành và phần cứng khác nhau. Java có riêng một bộ thư viện để hỗ trợ vấn đề này. Chương trình viết bằng ngôn ngữ Java có thể biên dịch trên nhiều loại máy khác nhau mà không gặp lỗi.
* **Ở cấp độ nhị phân:** Một mã biên dịch có thể chạy trên nhiều nền tảng khác nhau mà không cần dịch lại mã nguồn. Tuy nhiên cần có Java Virtual Machine để thông dịch đoạn mã này.

## + Ngôn ngữ thông dịch

Ngôn ngữ lập trình thường được chia ra làm 2 loại là:

* + **Thông dịch** (Interpreter) : Nó dịch từng lệnh rồi chạy từng lệnh, lần sau muốn chạy lại thì phải dịch lại.
  + **Biên dịch** (Compiler): Code sau khi được biên dịch sẽ tạo ra 1 file thường là .exe, và file .exe này có thể đem sử dụng lại không cần biên dịch nữa.

Ngôn ngữ lập trình Java thuộc loại ngôn ngữ thông dịch. Chính xác hơn, Java là loại ngôn ngữ vừa biên dịch vừa thông dịch. Cụ thể như sau

Khi viết mã, hệ thống tạo ra một tệp .java. Khi biên dịch mã nguồn của chương trình sẽ được biên dịch ra mã byte code. **Java Virtual Machine** sẽ thông dịch mã byte code này thành machine code (hay native code) khi nhận được yêu cầu chạy chương trình.

* + **Ưu điểm :** Phương pháp này giúp các đoạn mã viết bằng Java có thể chạy được trên nhiều nền tảng khác nhau. Với điều kiện là JVM có hỗ trợ chạy trên nền tảng này.
  + **Nhược điểm :** Cũng như các ngôn ngữ thông dịch khác, quá trình chạy các đoạn mã Java là chậm hơn các ngôn ngữ biên dịch khác.

## + Đa luồng

Java hỗ trợ lập trình đa tiến trình (multithread) để thực thi các công việc đồng thời. Đồng thời cũng cung cấp giải pháp đồng bộ giữa các tiến trình (giải pháp sử dụng priority…).

## + Tính an toàn và bảo mật

* **Tính an toàn**
  + Ngôn ngữ lập trình Java yêu cầu chặt chẽ về kiểu dữ liệu.
  + Dữ liệu phải được khai báo tường minh.
  + Không sử dụng con trỏ và các phép toán với con trỏ.
  + Java kiểm soát chặt chẽ việc truy nhập đến mảng, chuỗi. Không cho phép sử dụng các kỹ thuật tràn. Do đó các truy nhập sẽ không vượt quá kích thước của mảng hoặc chuỗi.
  + Quá trình cấp phát và giải phóng bộ nhớ được thực hiện tự động.
  + Cơ chế xử lý lỗi giúp việc xử lý và phục hồi lỗi dễ dàng hơn.

## Tính bảo mật

* + Java cung cấp một môi trường quản lý chương trình với nhiều mức khác nhau*.*
    - **Mức 1 :** Chỉ có thể truy xuất dữ liệu cũng như phương phức thông qua giao diện mà lớp cung cấp.
    - **Mức 2 :** Trình biên dịch kiểm soát các đoạn mã sao cho tuân thủ các quy tắc của ngôn ngữ lập trình Java trước khi thông dịch.
    - **Mức 3 :** Trình thông dịch sẽ kiểm tra mã byte code xem các đoạn mã này có đảm bảo được các quy định, quy tắc trước khi thực thi.
    - **Mức 4:** Java kiểm soát việc nạp các lớp vào bộ nhớ để giám sát việc vi phạm giới hạn truy xuất trước khi nạp vào hệ thống.

## Kotlin

* + - 1. **Đôi nét về Kotlin**

- Kotlin là một ngôn ngữ lập trình nguồn mở, kiểu tĩnh, hỗ trợ cả lập trình chức năng lẫn hướng đối tượng. Kotlin cung cấp cú pháp và khái niệm tương tự trong các ngôn ngữ khác, bao gồm cả C#, Java và cùng nhiều ngôn ngữ khác. Kotlin không phải là độc nhất – mà Kotlin lấy cảm hứng từ nhiều thập kỷ để phát triển ngôn ngữ.



Hình 2.3: Ngôn Ngữ Lập Trình Kotlin

* Kotlin do Kotlin Foundation quản lý, một nhóm do JetBrains và Google tạo ra với nhiệm vụ thúc đẩy và tiếp tục phát triển ngôn ngữ này. Google chính thức hỗ trợ Kotlin trong hoạt động phát triển Android.

## Đặc điểm

* + Về mặt ưu điểm:
    - Kotlin có thể khắc phục được các yếu điểm của Java
    - Code ngắn gọn và dễ hiểu
    - Kotlin không bị lỗi NullPointerException
    - Kotlin an toàn hơn Java
    - Kotlin có khả năng tương tác cao
    - Kotlin được hỗ trợ ưu tiên hỗ trong Android Studio và IDE
  + Về Mặt Hạn Chế:

+ Song song với các ưu điểm trên thì ngôn ngữ lập trình Kotlin còn tồn tại những mặt hạn chế như sau:

* + - Chưa có kiểu Aliases
    - Các mặc định class trong Kotlin là final.
    - Cộng đồng hỗ trợ hạn chế

## Công nghệ phần mềm Android Studio

## Đôi nét về Android Studio

Android Studio là môi trường phát triển tích hợp (IDE) chính thức của Google dành cho việc phát triển ứng dụng Android. Được xây dựng trên nền tảng IntelliJ IDEA, Android Studio mang đến một loạt các công cụ và tính năng mạnh mẽ giúp tăng năng suất trong quá trình phát triển ứng dụng Android.

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.



Hình 2.4: Phần Mềm Android Studio

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 và 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.

## Cài đặt môi trường Android Studio

* + 1. **Cấu hình cần thiết**

Khi bắt đầu làm đề tài với Android Studio, cần phải đảm bảo tuân thủ các yêu cầu cài đặt tối thiểu và đề xuất cho máy tính. Nó sẽ là một trong những yếu tố quan trọng trong việc khởi tạo một đề tài mới.

Cấu hình cần thiết khi cài đặt Android Studio

|  |  |  |
| --- | --- | --- |
| **Yêu Cầu** | **Tối Thiểu** | **Đề Xuất** |
| Hệ Điều Hành | Microsoft Windows 8 64 bit | Phiên Bản 64-Bits Mới Nhất |
| RAM | RAM 8GB | RAM 16 GB trở lên |
| CPU | Kiến trúc CPU x86\_64; Intel Core thế hệ 2 trở lên  hoặc CPU AMD có hỗ trợ Khung trình điều khiển ảo hoá của Windows. | Bộ xử lý Intel Core mới nhất |
| Dung Lượng | 8 GB (IDE, SDK Android và Trình mô phỏng) | Ổ thể rắn có dung  lượng từ 16 GB trở lên |
| Độ Phân Giải Màn Hình | 1280 x 800 | 1920 x 1080 |

Bảng 1.1: Cấu Hình Cài Đặt

* + 1. **Cài đặt Android Studio Bước 1:** Cài Đặt Java JDK

+ Tải và cài JDK theo hướng dẫn: <https://www.oracle.com/java/technologies/downloads/> Một số điều cần lưu ý khi cài đặt môi trường java:

* Mỗi phiên bản sẽ ứng với hệ điều hành của máy đang sử dụng
* Mở file cài đặt “jdk-\*.exe” để tiến hành cài đặt

**Bước 2:** Cài Đặt Android Studio

+ Tải và cài Android Studio:

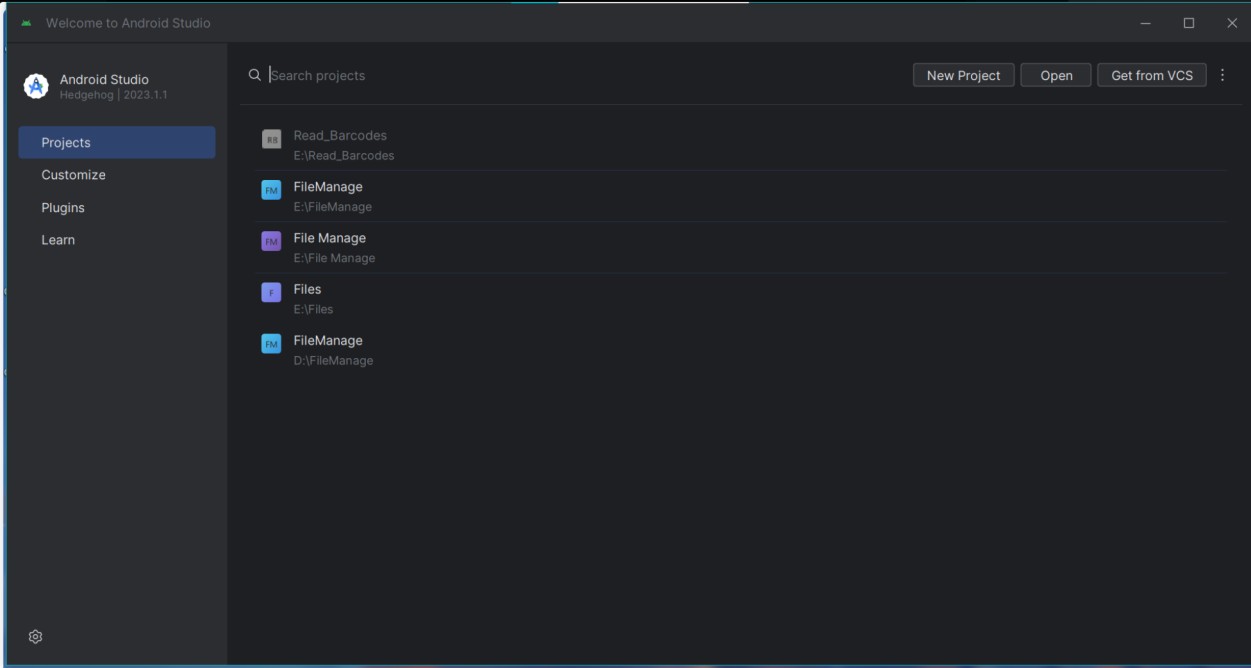
Link tải: <https://developer.android.com/studio?hl=vi>

Để cài đặt Android Studio trên Windows, hãy làm theo các bước sau:

* Nếu bạn đã tải một tệp .exe xuống (nên dùng), hãy nhấp đúp để mở tệp đó.
* Nếu bạn đã tải một tệp .zip xuống, thì hãy:
  + - 1. Giải nén .zip.
      2. Sao chép thư mục android-studio vào thư mục Program Files (Tệp chương trình).
      3. Mở thư mục android-studio > bin.
      4. Chạy studio64.exe (đối với máy 64 bit) hoặc studio.exe (đối với máy 32 bit).
      5. Làm theo Trình hướng dẫn thiết lập trong Android Studio và cài đặt mọi gói SDK được đề xuất.

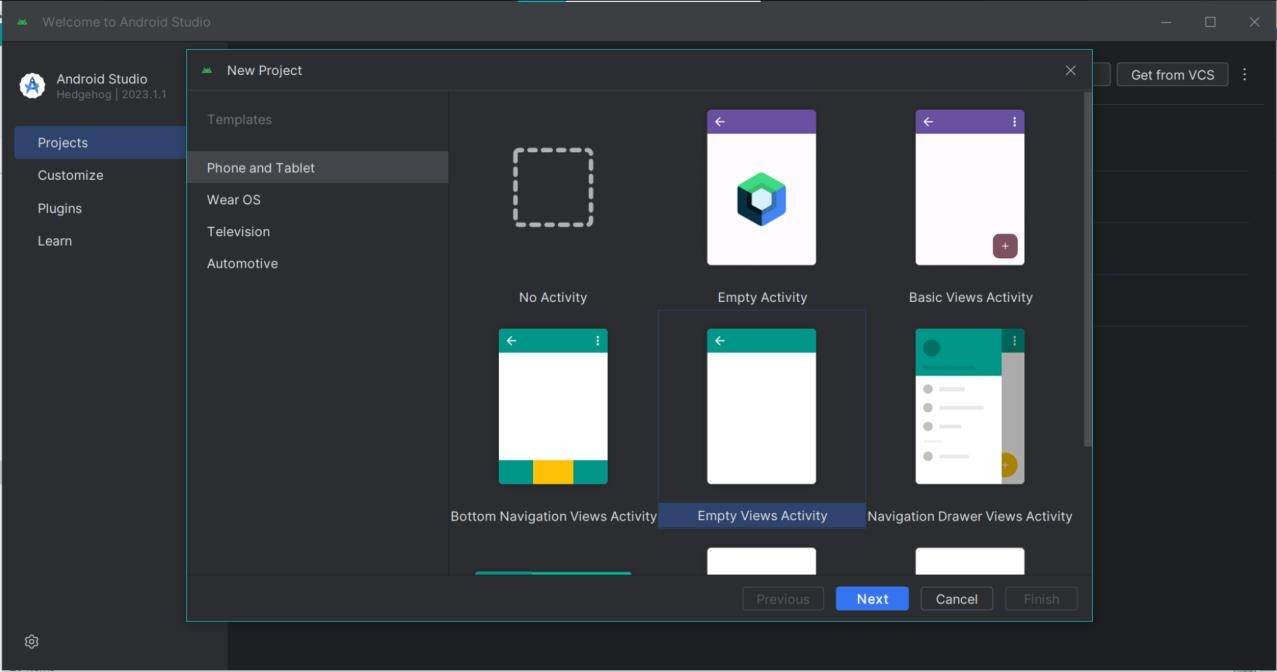
## Sử dụng Android Studio

*B1: Mở fodel, mở file*

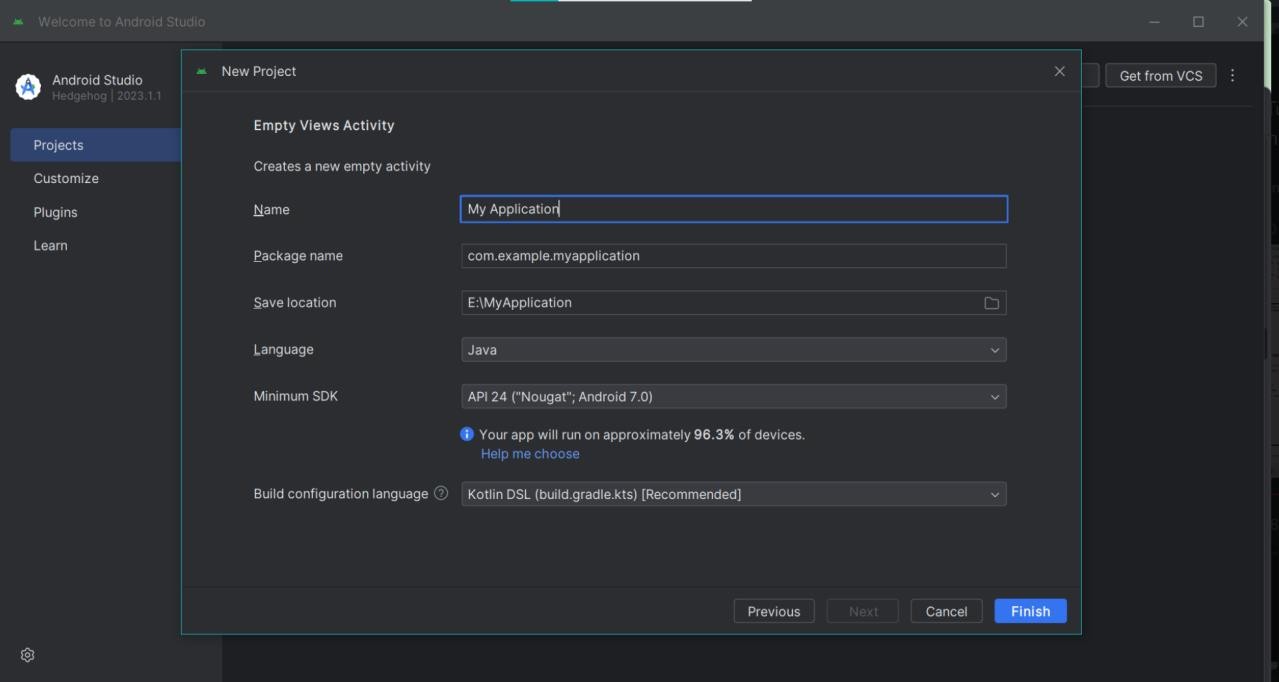


Hình 2.5: Mở Project hoặc Mở File

*B2: Tạo dự án mới*

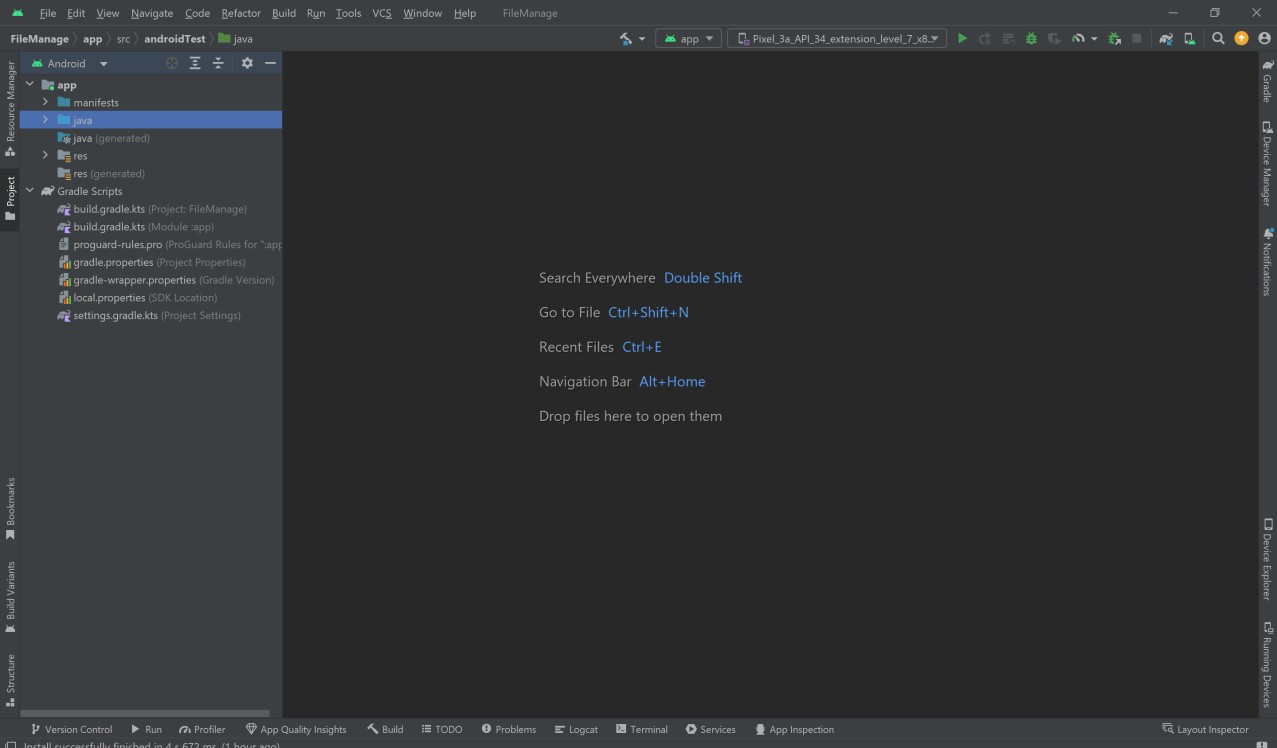


Hình 2.6: Khởi Tạo Giao Diện



Hình 2.7: Tạo Project Mới

*Giao diện của Android Studio*



Hình 2.8: Giao Diện Chính Của Android Studio

# CHƯƠNG III: KẾ HOẠCH THỰC HIỆN

## Lập kế hoạch làm việc theo từng giai đoạn của dự án

* Giai đoạn 1: Mục tiêu và xác định phạm vi

+ Mục tiêu chính:

* + Phát triển ứng dụng quản lý file nhằm cung cấp trải nghiệm người dùng thuận lợi và hiệu quả.
  + Tính năng cơ bản bao gồm xem, di chuyển, xóa, tạo mới và đổi tên file/thư mục.

+ Xác định phạm vi:

* + Hỗ trợ nhiều định dạng file phổ biến

như văn bản, hình ảnh, âm thanh, và video.

* + Tính năng tìm kiếm nhanh trong file và thư mục.

Hình 3.1: Chiến Lược Thực Hiện

* + Tích hợp chia sẻ file thông qua các kênh khác nhau như email hoặc messenger.
* Giai đoạn 2: Chuẩn bị và nghiên cứu

+ Chuẩn bị:

* + Xác định đội ngũ phát triển và phân công công việc.
  + Chuẩn bị môi trường làm việc với các công cụ phát triển như Android Studio.

+ Nghiên cứu:

* + Đọc tài liệu và học các thư viện phổ biến như Android FilePicker để xử lý file và thư mục trên thiết bị Android.
  + Nghiên cứu về giao diện người dùng hiệu quả để đảm bảo trải nghiệm người dùng tốt.
* Giai đoạn 3: Thiết kế và lập kế hoạch phát triển

+ Thiết kế giao diện người dùng và cơ sở dữ liệu:

* + Xây dựng bản vẽ giao diện người dùng sáng tạo và dễ sử dụng.
  + Thiết kế cơ sở dữ liệu để lưu trữ thông tin về file và thư mục.

+ Lập kế hoạch phát triển:

* + Phân tích yêu cầu và phân chia công việc thành các giai đoạn nhỏ hơn như phát triển giao diện, xử lý file, và tích hợp tính năng tìm kiếm.
* Giai đoạn 4: Phát triển ứng dụng

+ Xây dựng giao diện và chức năng cơ bản:

* + Phát triển giao diện người dùng dựa trên bản thiết kế đã được xác nhận.
  + Xây dựng các chức năng cơ bản như xem, di chuyển, xóa file/thư mục.

+ Tích hợp công nghệ:

* + Sử dụng thư viện Android FilePicker để thực hiện các thao tác quản lý file.
  + Tối ưu hóa trải nghiệm người dùng thông qua sự tích hợp linh hoạt và hiệu suất cao.
* Giai đoạn 5: Kiểm thử và tối ưu hóa

+ Kiểm thử chất lượng:

* + Thực hiện kiểm thử đơn vị để đảm bảo mỗi chức năng hoạt động đúng như kỳ vọng.
  + Kiểm thử tích hợp để đảm bảo tương tác chính xác giữa các thành phần của ứng dụng.

+ Tối ưu hóa hiệu suất và năng lượng:

* + Áp dụng các kỹ thuật tối ưu hóa mã nguồn và tối ưu hóa hình ảnh để giảm tiêu thụ năng lượng và tăng hiệu suất.
* Giai đoạn 6: Đánh giá và triển khai

+ So sánh và đánh giá:

* + So sánh ứng dụng với các ứng dụng quản lý file có sẵn trên thị trường.
  + Thu thập phản hồi từ nhóm người dùng thử nghiệm để điều chỉnh và cải thiện.

+ Triển khai và quảng bá:

* + Chuẩn bị và triển khai ứng dụng lên cửa hàng ứng dụng (ví dụ: Google Play Store).
  + Phát triển chiến lược quảng bá để thu hút người dùng và tăng số lượng tải về.

## CHƯƠNG IV: PHÂN TÍCH YÊU CẦU ỨNG DỤNG QUẢN LÝ FILE

## Tương tác người dùng

* Dưới đây là một số hành vi phổ biến mà người dùng thường thực hiện khi sử dụng các ứng dụng quản lý file:

+ Tạo và xoá File/Folder:

* + Người dùng thường sẽ tạo mới các file hoặc thư mục để tổ chức thông tin.
  + Xoá file hoặc thư mục không cần thiết để giải phóng không gian lưu trữ.

+ Di chuyển và sao chép:

* + Người dùng thường di chuyển file hoặc thư mục từ một vị trí sang một vị trí khác để tổ chức dữ liệu.
  + Sao chép file có thể được sử dụng để sao lưu dữ liệu hoặc tạo bản sao làm việc.

+ Xem trước và mở file:

* + Người dùng thường muốn xem trước nội dung của file trước khi quyết định mở nó.
  + Mở file để chỉnh sửa, xem nội dung hoặc thực hiện các tác vụ khác

+ Chọn, Cut, Copy, Paste:

* + Chọn (Select): Cho phép người dùng chọn một hoặc nhiều file/thư mục cùng một lúc.
  + Cut (Cắt): Di chuyển dữ liệu đã chọn từ vị trí hiện tại đến vị trí mới.
  + Copy: Sao chép dữ liệu đã chọn và giữ nguyên ở vị trí hiện tại.
  + Paste: Đặt dữ liệu đã được cut hoặc copy vào vị trí mới.

+ Rename (Đổi tên):

* + Cho phép người dùng đổi tên file hoặc thư mục để phản ánh mục đích mới hoặc nội dung thay đổi.

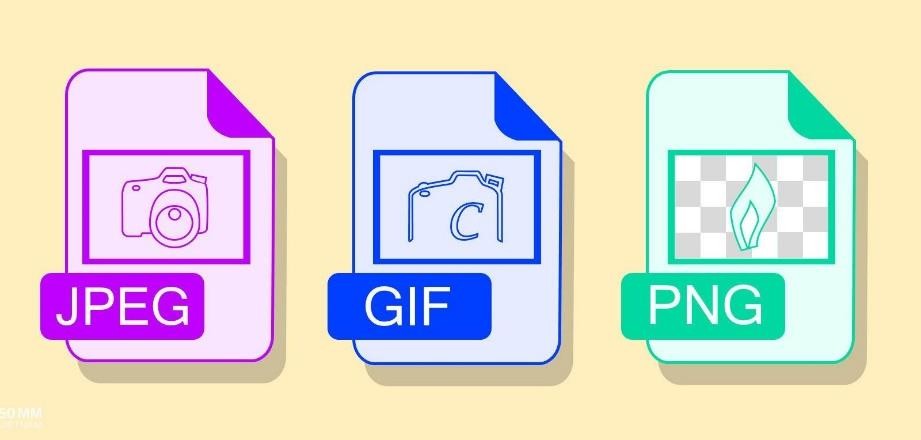
+ Delete (Xoá):

* + Loại bỏ file hoặc thư mục không còn cần thiết. Thường yêu cầu xác nhận để tránh xoá không chủ ý.

Các tính năng này giúp người dùng dễ dàng di chuyển, sao chép, xoá và đổi tên các file và thư mục, tạo ra quá trình quản lý dữ liệu một cách thuận lợi và hiệu quả.

## Thao tác của người dùng

* Import file: Thêm các file khác nhau (Video, Hình ảnh, audio, pdf, doc,…..) từ các nguồn khác nhau (ví dụ như từ SD Card, Storage hoặc những nguồn tương tự ) vào ứng dụng của mình để thực hiện các xử lý hoặc hiển thị nó trong ứng dụng quản lý file



Hình 4.1: File Dữ Liệu

* Tạo thư mục: Tạo ra một đơn vị lưu trữ dữ liệu để chứa và tổ chức các file hoặc thư mục con để nhìn thuận tiện cho việc sử dụng
* Mở file: Là quá trình truy cập và hiển thị nội dung của file trong không gian lưu trữ bằng việc sử dụng ứng dụng quản lý file
* Chạy file: Là quá trình thực hiện và xử lý nội dung của file được người dùng chọn trên ứng dụng, thường liên quan đến các hành động được định nghĩa bởi file đó.
* Thực hiện các thao tác cơ bản:
  + Chọn (Select): Lựa chọn một hoặc nhiều file hoặc thư mục từ danh sách hiển thị.
  + Cut (Cắt) và Copy (Sao Chép):
    - Cut: Di chuyển các file hoặc thư mục đã chọn từ vị trí hiện tại.
    - Copy: Tạo bản sao của các file hoặc thư mục đã chọn.
  + Dán (Paste): Đưa các file hoặc thư mục đã cut hoặc copy vào vị trí hiện tại hoặc đích đến mới.
  + Rename (Đổi Tên): Thay đổi tên của file hoặc thư mục đã chọn.
  + Delete (Xóa): Xóa các file hoặc thư mục đã chọn khỏi hệ thống.

## Nhu cầu từ người dùng

Dễ dàng tìm kiếm tệp tin:

* + - Chức năng tìm kiếm linh hoạt để nhanh chóng định vị và truy cập các file.
    - Giao diện người dùng thân thiện và hiệu quả để xem và quản lý nhiều loại file. Quản lý thư mục hiệu quả:
    - Khả năng tạo, xóa, di chuyển và đổi tên thư mục và file một cách dễ dàng.
    - Các tùy chọn sắp xếp và lọc để tổ chức dữ liệu một cách hiệu quả. Tương tác dễ dàng và tùy chỉnh:
    - Giao diện dễ sử dụng với các tùy chọn tương tác như kéo và thả.
    - Khả năng tùy chỉnh cài đặt và giao diện để đáp ứng nhu cầu cá nhân.

## Đối tượng hướng tới

Để đáp ứng đa dạng của người dùng và nhu cầu sử dụng đa dạng, ứng dụng này cung cấp trải nghiệm tùy chỉnh cho từng đối tượng sử dụng. Dưới đây là một số đối tượng sử dụng phổ biến:

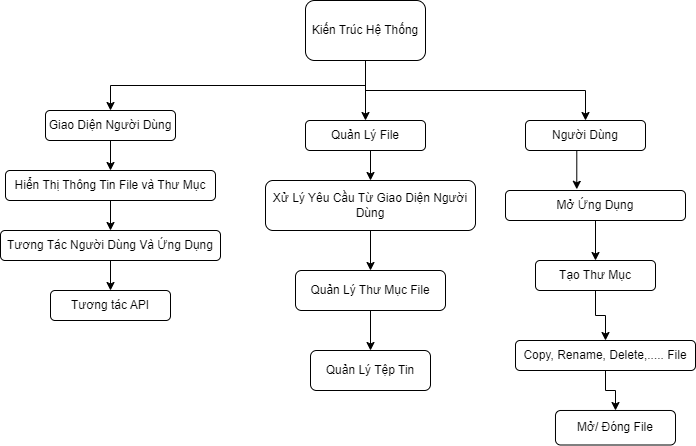
* + - Người dùng sử dụng ứng dụng trên thiết bị di động như điện thoại hoặc máy tính bảng
    - Những người sử dụng cá nhân, chủ yếu quản lý và tổ chức file cá nhân và dữ liệu trên thiết bị của họ.
    - Người dùng làm việc trong môi trường doanh nghiệp, có thể làm việc cộng tác và chia sẻ file với đồng nghiệp
    - Các nghệ sĩ, nhà thiết kế và những người sáng tạo khác cần tổ chức và lưu trữ tập tin nghệ thuật, hình ảnh, và dự án sáng tạo

## Thiết bị và nền tảng

Thông thường, người dùng sẽ sử dụng ứng dụng trên điện thoại di động, đòi hỏi giao diện phải tối ưu hóa cho trải nghiệm di động.

# CHƯƠNG V: THIẾT KẾ KIẾN TRÚC ỨNG DỤNG

## Kiến trúc ứng dụng



Hình 5.1: Sơ Đồ Kiến Trúc Hệ Thống

## Giao diện người dùng (UI)

Trách nhiệm: Hiển thị thông tin về file và thư mục, tương tác với người dùng.

Mô Tả: Giao diện người dùng là thành phần mà người dùng trực tiếp tương tác để quản lý và thao tác với file.

## Quản lý file

Trách nhiệm: Xử lý yêu cầu từ giao diện người dùng, quản lý thư mục, file, và tương tác với cơ sở dữ liệu.

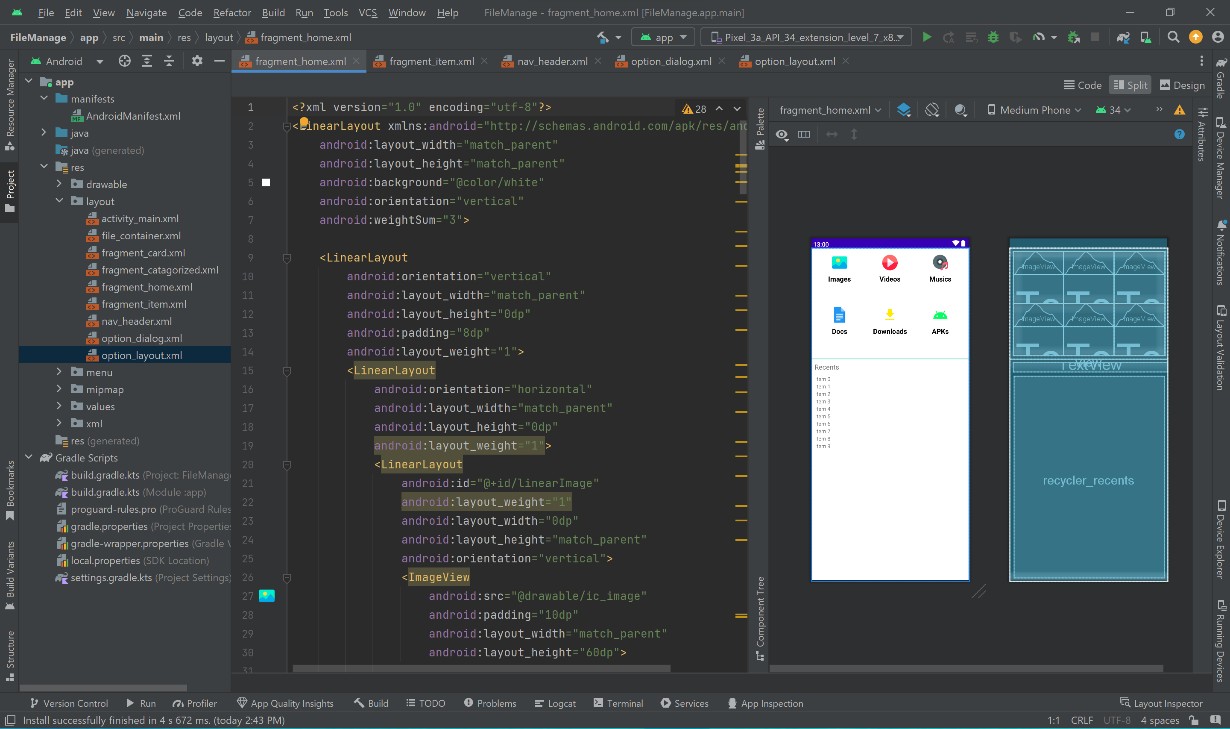
Mô Tả: Quản lý file là trung tâm xử lý logic của ứng dụng, thực hiện các chức năng quản lý file và thư mục.

## Người dùng

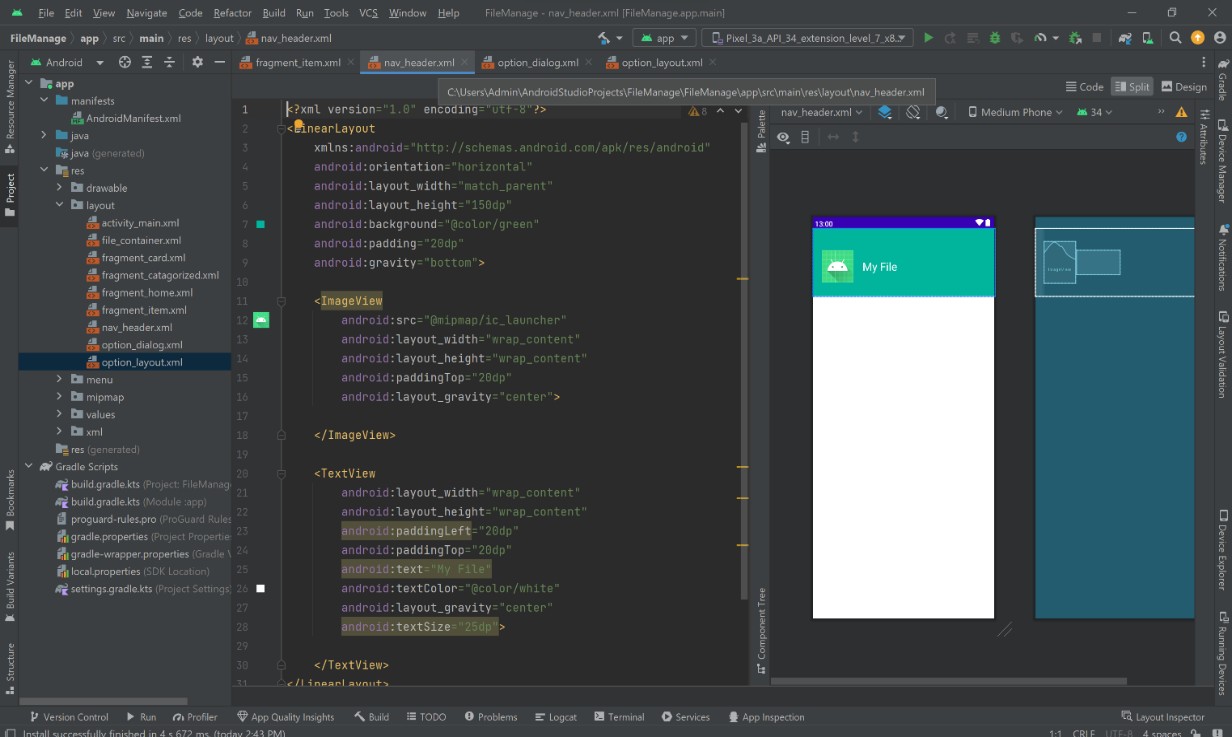
Trách nhiệm: Người dùng sẽ sử dụng các chức năng được cài sẵn trong ứng dụng.

Mô Tả: Các chức năng trong trong ứng dụng được khi được người dùng sử dụng sẽ liên kết chặt chẽ với nhau. Không có hiện tưởng xảy ra lỗi.

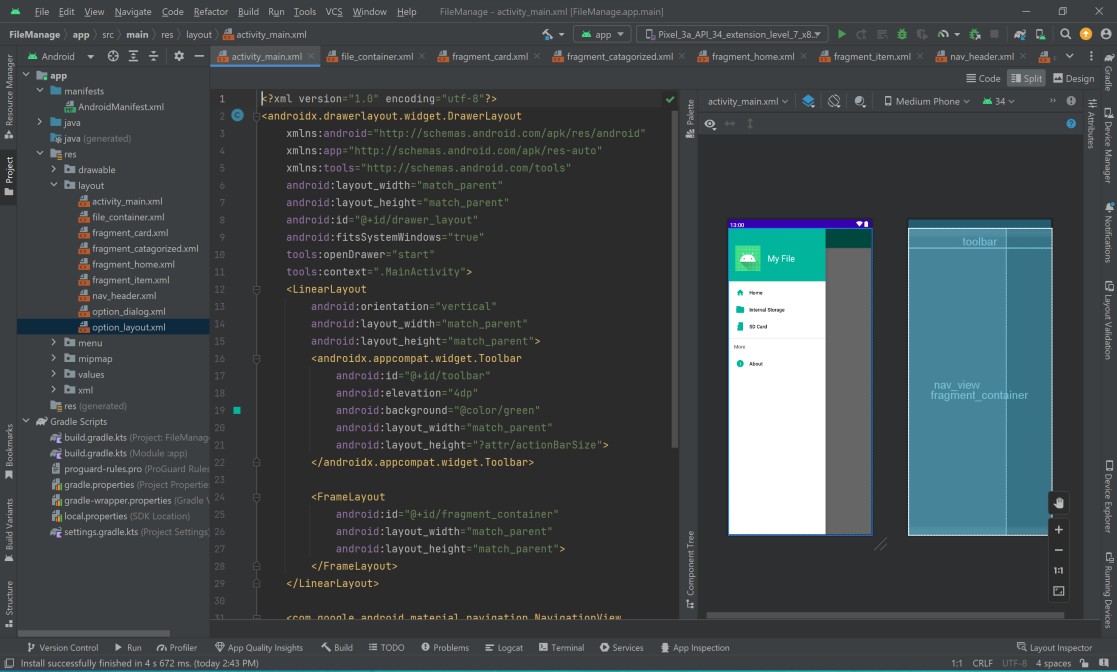
## Thiết kế giao diện người dùng (UI)



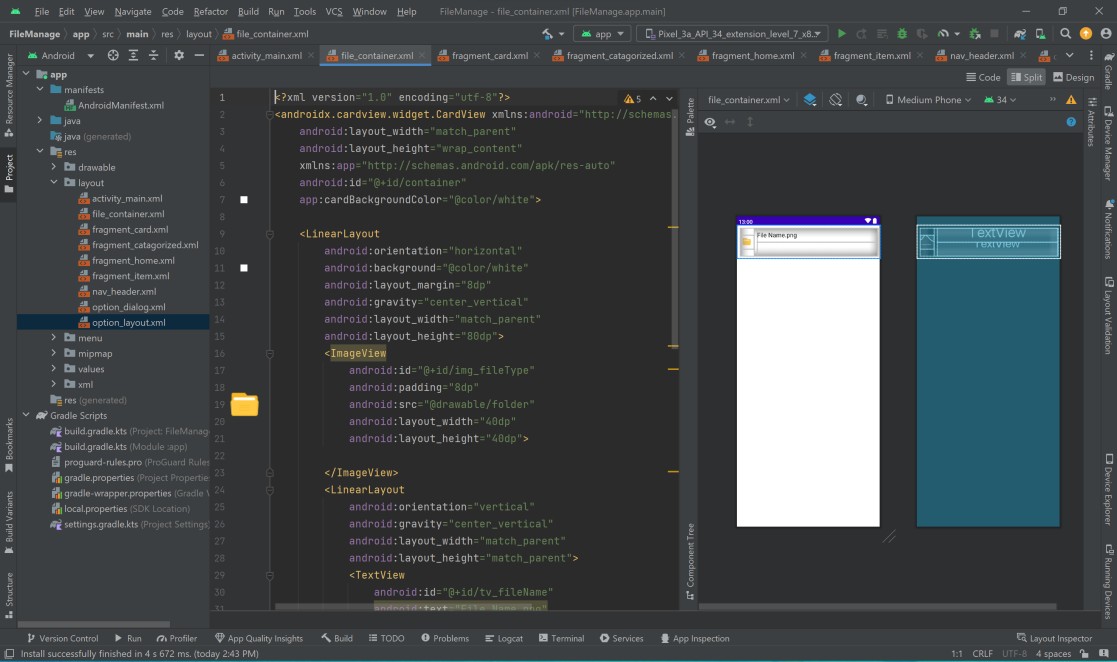
Hình 5.2: Giao Diện Người Dùng



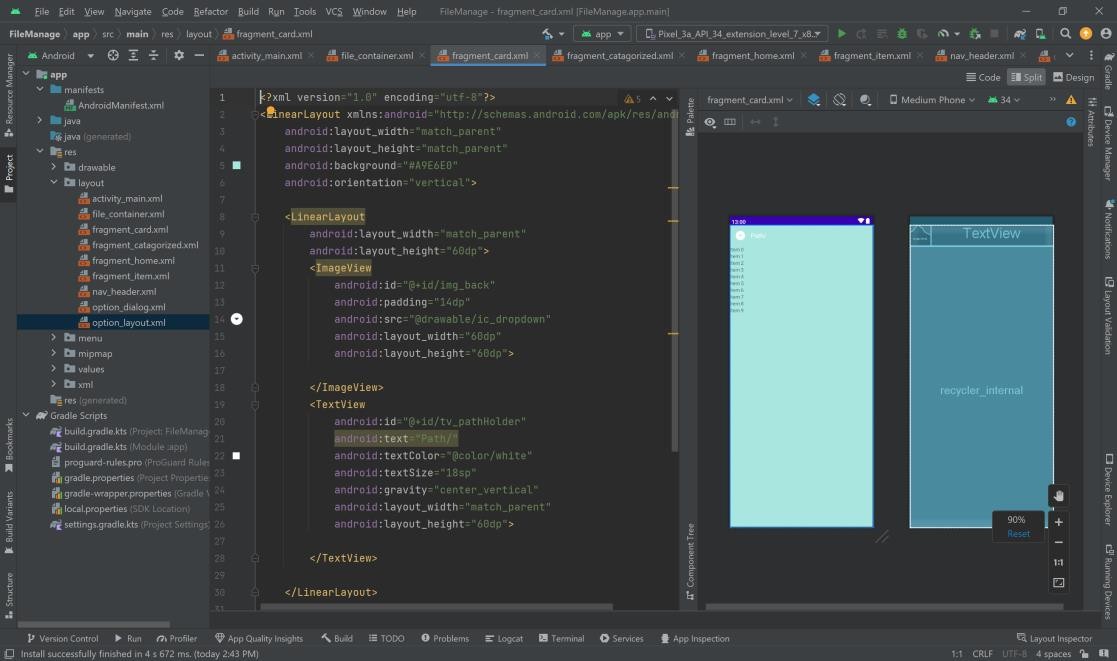
Hình 5.3: Giao Diện Bắt Đầu Khởi Tạo



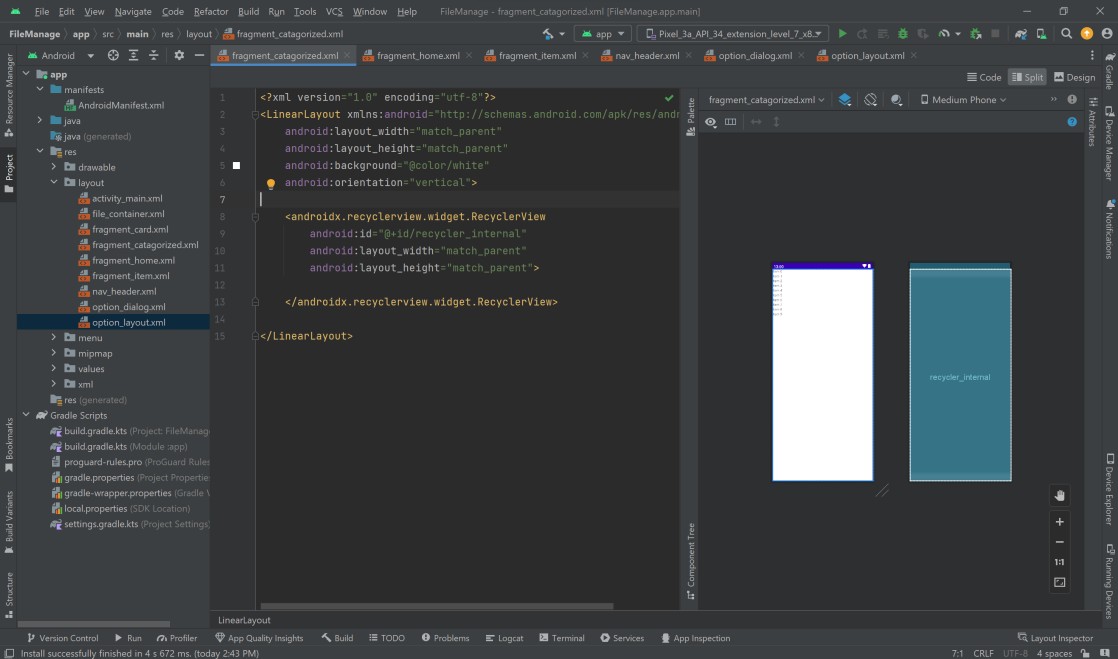
Hình 5.4: Các Chức Năng Cơ Bản



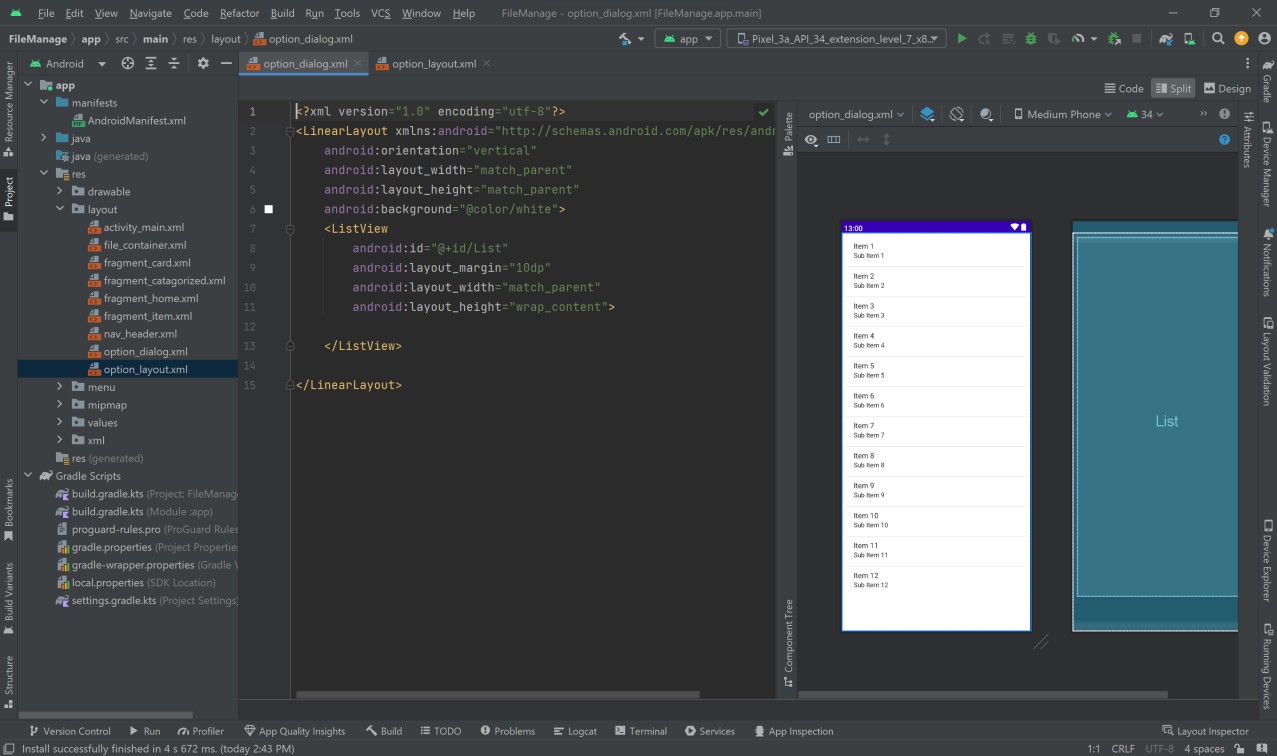
Hình 5.5: Sửa Lỗi



Hình 5.6: Qúa Trình Chỉnh Sửa



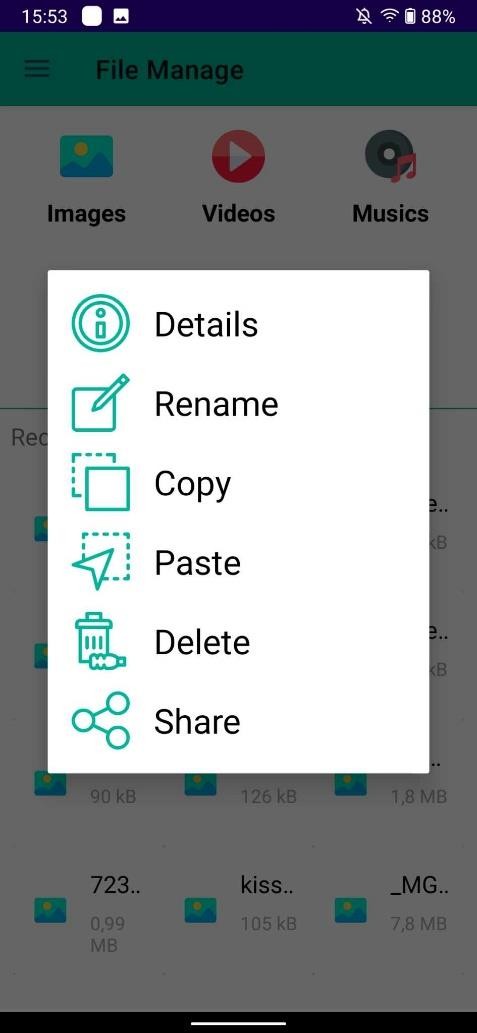
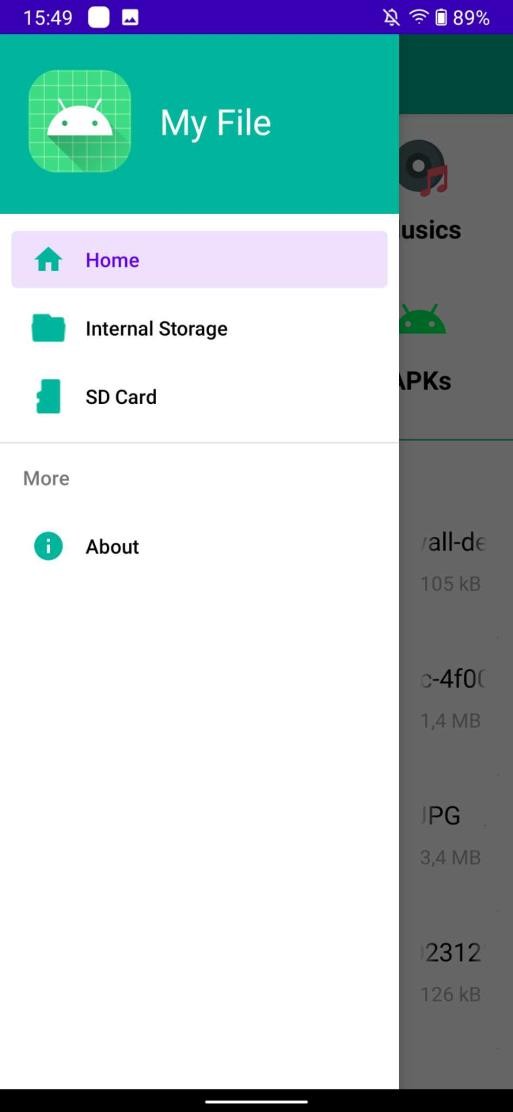
Hình 5.7: Qúa Trình Chỉnh Sửa



Hình 5.8: Thêm Các Chức Năng Mới

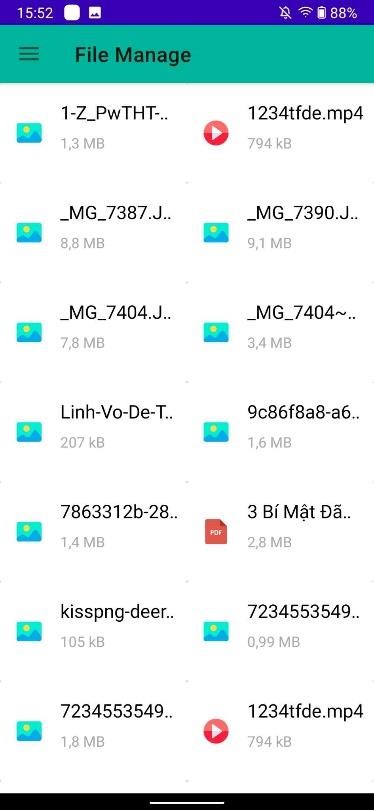
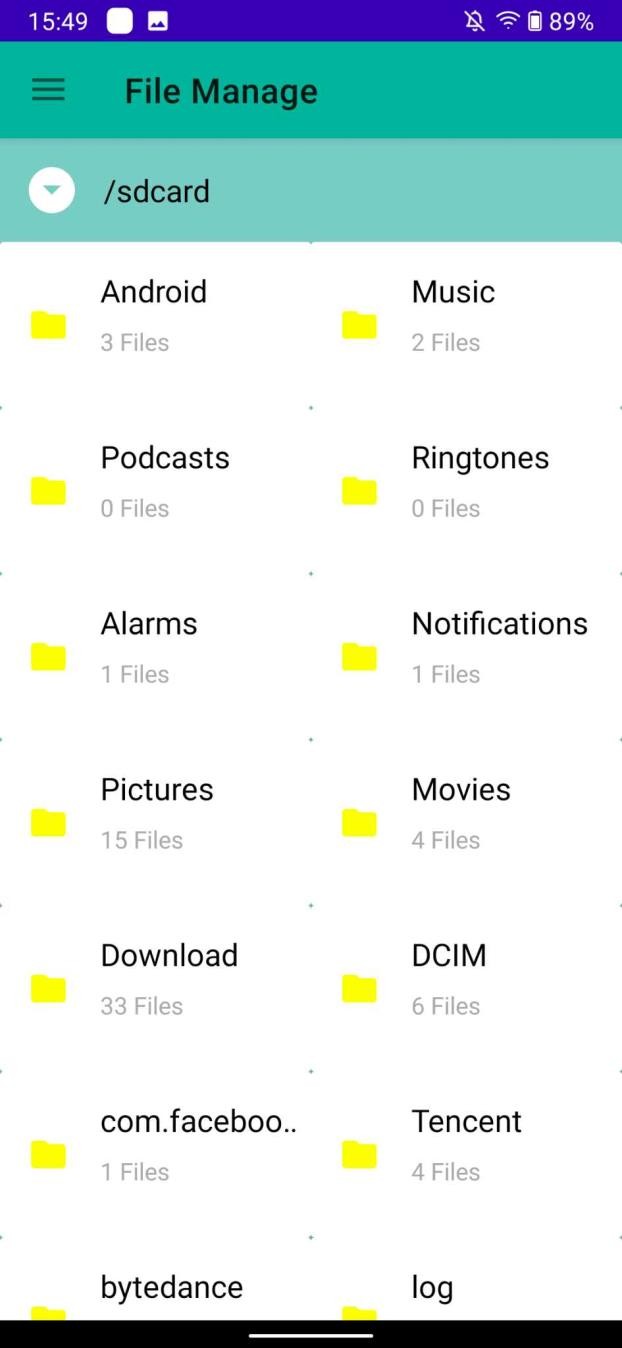
## Giao diện sau khi chạy trên điện thoại Android

Sau những lần thử nghiệm cũng như chỉnh sửa các phần liên quan đến đề tài. Thì sau đây là một số hình ảnh về ứng dụng quản lý file lần này.

Hình 5.9: Các Chức Năng Chính Hình 5.10: Các Chức Năng Cơ Bản

## Logic ứng dụng



Hình 5.11: Giao Diện Chính Hình 5.12: Giao Diện

## Quản lý tập tin và thư mục

* Tạo, Xóa, và Đổi Tên:
  + Tính năng tạo mới tập tin và thư mục phải đảm bảo tính linh hoạt, cho phép người dùng tạo và tổ chức dữ liệu theo cách họ muốn.
  + Tính năng xóa tập tin và thư mục cần phải có các bước xác nhận để tránh mất mát dữ liệu không mong muốn.
  + Đổi tên tập tin và thư mục phải được thực hiện một cách đơn giản và nhanh chóng.

## Tìm kiếm và sắp xếp

* Tìm Kiếm Linh Hoạt:
  + Tính năng tìm kiếm phải hỗ trợ tìm kiếm theo tên, loại, kích thước và thời gian sửa đổi.
  + Sắp Xếp Hiệu Quả:
  + Các tiêu chí sắp xếp như abc, thời gian, và kích thước phải được cung cấp để người dùng có thể dễ dàng sắp xếp dữ liệu của họ theo ý muốn.

## Đồng bộ hóa dữ liệu

* Quản lý Đồng Bộ:
  + Đảm bảo dữ liệu được đồng bộ hóa đúng cách giữa các thiết bị, bằng cách sử dụng các cơ chế đồng bộ linh hoạt và hiệu quả.
  + Cung cấp cơ chế xử lý xung đột để tránh tình trạng mất dữ liệu hoặc trùng lặp.

## Giao diện người dùng thân thiện

* Thiết Kế Giao Diện Linh Hoạt:
  + Giao diện người dùng phải thân thiện, linh hoạt và tương thích trên nhiều loại thiết bị, từ điện thoại đến máy tính bảng và máy tính cá nhân.
  + Sử dụng các yếu tố như thanh điều hướng, nút chức năng và cách hiển thị thông tin một cách hợp lý để tối ưu hóa trải nghiệm người dùng.

## Kiểm thử và đánh giá

* Kiểm Thử Chất Lượng:
  + Bao gồm kiểm thử tích hợp, kiểm thử đơn vị, và kiểm thử hệ thống để đảm bảo ứng dụng hoạt động mượt mà và không gặp lỗi.

## Những lưu ý cần thiết

Giao diện người dùng thân thiện: Tạo một giao diện người dùng thân thiện và dễ sử dụng, giúp người dùng dễ dàng đặt và quản lý báo thức.

Tích hợp Notification: Nếu cần, tích hợp thông báo để hiển thị trạng thái của ứng dụng nếu ứng dụng không chạy hoặc bị lỗi

Kiểm thử: Thực hiện kiểm thử kỹ lưỡng để đảm bảo rằng mọi chức năng hoạt động đúng và không gặp lỗi.

Đa nhiệm và tiết kiệm pin: Đảm bảo ứng dụng hoạt động hiệu quả và tiết kiệm pin.

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN TƯƠNG LAI

## Kết luận

Khi nhìn lại quá trình thực hiện dự án, chúng tôi nhận ra rằng việc xây dựng ứng dụng quản lý file không chỉ là về việc lập trình và phát triển tính năng, mà còn là về việc hiểu sâu về nhu cầu và mong đợi của người dùng. Mục tiêu của chúng tôi là không chỉ tạo ra một công cụ mạnh mẽ, mà còn làm cho trải nghiệm người dùng trở nên tối ưu nhất có thể.

Trong quá trình phát triển, chúng tôi đã đối mặt với nhiều thách thức, đặc biệt là trong việc thiết kế giao diện người dùng. Việc làm cho giao diện thân thiện, dễ sử dụng và đồng thời đáp ứng được đầy đủ yêu cầu đã đòi hỏi sự sáng tạo và kiên nhẫn. Các phiên bản thử nghiệm và ý kiến phản hồi từ người dùng đã đóng vai trò lớn trong việc điều chỉnh và hoàn thiện giao diện của chúng tôi.

Về mặt ưu điểm và hạn Chế: Một điểm mạnh của dự án là quy trình làm việc của người dùng được tối ưu hóa, từ việc tải lên, tải xuống, đến di chuyển và xóa file. Tính năng đồng bộ hóa dữ liệu cũng đạt được mức độ linh hoạt mà chúng tôi mong đợi. Tuy nhiên, chúng tôi nhận thức rằng vẫn còn nhiều không gian để cải thiện và mở rộng tính năng để đáp ứng đầy đủ nhu cầu đa dạng của người dùng.

Dự án này không chỉ là một kết thúc mà là sự bắt đầu cho những phát triển tương lai. Chúng tôi đặt ra kế hoạch để liên tục cập nhật ứng dụng dựa trên phản hồi của người dùng và xu hướng công nghệ mới. Việc so sánh với các ứng dụng khác trên thị trường đã mở ra cơ hội để chúng tôi học hỏi và áp dụng những điểm mạnh vào sản phẩm của mình.

Quá trình nghiên cứu và phát triển không chỉ mang lại kiến thức chuyên sâu về lập trình và quản lý dự án mà còn là cơ hội để chúng tôi hiểu rõ hơn về sự quan trọng của sự linh hoạt và tương tác trong thiết kế ứng dụng. Được nghe ý kiến và phản hồi từ người dùng đã là một trải nghiệm học hỏi vô cùng quý giá, giúp chúng tôi hiểu rõ hơn về những yếu tố quyết định sự thành công của một sản phẩm.

Tổng quan, dự án này không chỉ mang lại thành quả về mặt kỹ thuật mà còn giúp chúng tôi phát triển kỹ năng tư duy phê phán, sáng tạo và làm việc nhóm. Chúng tôi nhận thức rằng công việc này chỉ là bước đầu tiên, và chúng tôi sẽ tiếp tục nỗ lực để đưa ứng dụng của mình đến với người dùng một cách tốt nhất, đồng thời liên tục học hỏi và cập nhật theo sự phát triển của công nghệ và yêu cầu người dùng.

## Hướng phát triển trong tương lai

Ứng dụng quản lý file trên Android ngày nay không chỉ đơn giản là công cụ sắp xếp dữ liệu mà đã là một ứng dụng không thể thiếu trong việc quản lý dữ liệu cá nhân hiện nay. Để đối mặt với sự phức tạp ngày càng tăng, chúng ta cần phải tích hợp những công nghệ mới để tăng

hiệu suất cũng như cải thiện thêm trải nghiệm của người dùng đối với ứng dụng quản lý file. Điều này mang lại ứng dụng không chỉ là công cụ hữu ích mà còn là sự đồng hành linh hoạt và thông minh trong quản lý dữ liệu cá nhân.

Dưới đây sẽ là những hướng phát triển để xây dựng những ứng dụng quản lý file tương lai, mang lại trải nghiệm người dùng độc đáo và tiện ích không ngừng.

+ Tích hợp AI và Machine Learning (ML)

* Sử dụng hệ thống AI để có thể tự động phân loại và gắn nhãn cho các tệp dựa trên nội dung. *Ví dụ, nhận biết văn bản, hình ảnh, âm thanh để tạo ra các nhãn mô tả chính xác.*
* Sử dụng ML (Machine Learning) để dự đoán hành động tiếp theo của người dùng dựa trên lịch sử sử dụng. *Ví dụ, đề xuất tệp liên quan khi người dùng đang tìm kiếm hoặc làm việc với một tệp cụ thể.*

+ Nâng cao khả năng truy cập đám mây

* Quản lý file liên tục đồng bộ hóa với các dịch vụ lưu trữ đám mây như Google Drive, Dropbox, OneDrive, để có thể đảm bảo sự nhất quán giữa các thiết bị và truy cập mọi nơi..

+ Bảo mật tăng cường mạnh mẽ hơn

* Sử dụng mã hóa mạnh mẽ để bảo vệ dữ liệu khi chúng được lưu trữ và truyền tải. Phải bao gồm mã hóa mức tệp và mã hóa kết nối.
* Tích hợp các cơ chế quản lý quyền truy cập tiên tiến để kiểm soát ai có quyền truy cập và chỉnh sửa các tệp tin cụ thể.

+ Giao diện người dùng hiện đại

* Cung cấp chức năng tìm kiếm nâng cao với các bộ lọc và tùy chọn tìm kiếm trong nội dung tệp để nhanh chóng định vị các tệp cần thiết.
* Sử dụng biểu đồ, đồ thị và hình ảnh để trực quan hóa cấu trúc thư mục và lịch sử sửa đổi tệp.

+ Quản lý tốt hơn lịch sử sửa đổi

* Cung cấp một giao diện hiển thị lịch sử sửa đổi chi tiết với thông tin về người sửa đổi, thời gian và nội dung thay đổi.

+ Hỗ trợ đa nhiệm

* Hỗ trợ nhiều tác vụ đa nhiệm trên cùng một tệp, cho phép người dùng thực hiện nhiều công việc trên cùng một không gian làm việc.

+ Hệ sinh thái mở rộng và tích hợp

* Hỗ trợ các tiêu chuẩn mở rộng để tích hợp với các ứng dụng và dịch vụ khác, giúp mở rộng chức năng theo nhu cầu người dùng.
* Cung cấp API mở để cho phép các nhà phát triển tích hợp quản lý file vào các ứng dụng và hệ thống của họ.

+ Quản lý dữ liệu lớn

* Tối ưu hóa quá trình xử lý để đảm bảo hiệu suất cao khi làm việc với dữ liệu lớn, bao gồm tìm kiếm và truy xuất.
* Hỗ trợ nhiều định dạng dữ liệu khác nhau và cung cấp công cụ chuyển đổi để làm việc dễ dàng với các loại tệp tin đa dạng.

## TÀI LIỆU THAM KHẢO

**Đối với sách:**

1. Android Programming for Beginners

## Đối với website:

1. Android là gì?

[https://vi.wikipedia.org/wiki/Android\_(h%E1%BB%87\_%C4%91i%E1%BB%81u\_h%](https://vi.wikipedia.org/wiki/Android_(h%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh)) [C3%A0nh)](https://vi.wikipedia.org/wiki/Android_(h%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh)) truy cập ngày 02/11/2024

1. Kotlin là gì?

[https://vi.wikipedia.org/wiki/Kotlin\_(ng%C3%B4n\_ng%E1%BB%AF\_l%E1%BA%AD](https://vi.wikipedia.org/wiki/Kotlin_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)) [p\_tr%C3%ACnh)](https://vi.wikipedia.org/wiki/Kotlin_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)) truy cập ngày 02/11/2024

1. Java là gì?

[https://vi.wikipedia.org/wiki/Java\_(ng%C3%B4n\_ng%E1%BB%AF\_l%E1%BA%ADp](https://vi.wikipedia.org/wiki/Java_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh))

[\_tr%C3%ACnh)](https://vi.wikipedia.org/wiki/Java_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)) truy cập ngày 02/11/2024

1. Xây dựng ứng dụng quản lý file android.

[https://123docz.net/document/4044314-xay-dung-ung-dung-quan-ly-file-manager-cho-](https://123docz.net/document/4044314-xay-dung-ung-dung-quan-ly-file-manager-cho-thiet-bi-su-dung-he-dieu-hanh-android.htm) [thiet-bi-su-dung-he-dieu-hanh-android.htm](https://123docz.net/document/4044314-xay-dung-ung-dung-quan-ly-file-manager-cho-thiet-bi-su-dung-he-dieu-hanh-android.htm) truy cập ngày 02/11/2024

1. Xử lý sự kiện (Event Handling) trong Android. <https://vietjack.com/android/xu_ly_su_kien_trong_android.jsp#google_vignette> truy cập ngày 02/11/2024
2. Android Tutorial.

[Android Tutorial - GeeksforGeeks](https://www.geeksforgeeks.org/android-tutorial/)

1. Lập Trình Android – Java

[https://xuanthulab.net/cai-dat-android-studio-may-ao-android-tao-du-an-ung-dung-dau-](https://xuanthulab.net/cai-dat-android-studio-may-ao-android-tao-du-an-ung-dung-dau-tien.html) [tien.html](https://xuanthulab.net/cai-dat-android-studio-may-ao-android-tao-du-an-ung-dung-dau-tien.html) truy cập ngày 02/11/2024

1. Youtube

[https://www.youtube.com/playlist?list=PL9HYomN4XwEmeKW6CIXMIzQ-](https://www.youtube.com/playlist?list=PL9HYomN4XwEmeKW6CIXMIzQ-_vHCXqvNV)

[\_vHCXqvNV](https://www.youtube.com/playlist?list=PL9HYomN4XwEmeKW6CIXMIzQ-_vHCXqvNV) truy cập ngày 07/11/2024

## PHỤ LỤC

[1] Sourcode – File Thư Mục