A

PROJECT REPORT ON

**File Manager and**

**Duplicate Remover for**

**Android Devices**

**By**

**Hardik Bagada(CE009)(17CEUSG001)**

**Mahendrasinh Barad(CE011)(17CEUBG136)**

**Harshad Chovatiya(CE019)(17CEUOT078)**

**B.Tech CE Semester-VI**

**Subject: System Design Practice**

**Project Title: Swift File Manager**

**Guided By:**

Prof. Pandav K. Patel

Assistant Professor

Department of Computer Engineering



**Faculty of Technology**

**Department of Computer Engineering**

**Dharmsinh Desai University**



**Faculty of Technology**

**Department of Computer Engineering**

**Dharmsinh Desai University**

**CERTIFICATE**

This is to certify that the practical work/term work carried out in the subject of **System Design Practice** and recorded in this journal is the bonafide work of

**Hardik Bagada(CE009)(17CEUSG001)**

**Mahendrasinh Barad(CE011)(17CEUBG136)**

**Harshad Chovatiya(CE019)(17CEUOT078)**

Of B.Tech Semester **VI** in the Branch of **Computer Engineering** during the Academic year **2019-2020**.

Prof. Pandav K. Patel Dr. C.K. Bhensdadia,

Assistant Professor, Head of Department

Department of Computer Engineering, Of Computer Engineering,

Faculty of Technology, Faculty of Technology,

Dharmsinh Desai University, Nadiad Dharmsinh Desai University, Nadiad

**CONTENTS**

1. Front Page ………………………………………………….01
2. Certificate …………………………………………………. 02
3. Abstract …………………………………………………….04
4. Introduction ………………………………………………...05
5. Software Requirement Specification ………………………06
6. Design ……………………………………………………...09
   1. Use Case Diagram…………………………………...09
7. Implementation Details …………………………………….10
8. Screenshots ………………………………………………...12
9. Conclusion ………………………………………………....18
10. Limitation and Future Extension …………………………..19
11. Bibliography ……………………………………………….20

**Abstract**

It is very difficult to manage all the files and folders in an android device when the number of files and folders continuously increases day-to-day.Our application Swift File Manager helps users to manage all of his/her documents and keep them safe without loss of any data. An application also helps users to manage files based on their types.

Due to the long term uses of a device, storage memory of devices get fulled because of some duplicate files present in the device from which the user is unaware. Our application helps users to find all the duplicate files present in the device and also provides the functionality to delete them.

**Introduction**

Swift File Manager in an android application developed using Dart as a programming language. Users can easily store all of his files and documents and can also easily get them back when needed. The application also provides functionality to display files of any specific type. Through an application, Users can also find out a list of all duplicate files available on the system.

**Technology Used:**

* Flutter

**Platform Used:**

* Windows 10
* Ubuntu 18.04 (Bionic)

**Tools Used:**

* Visual Studio code
* Git as VCS(Version Control System)

**Software Requirements Specification**

Users of the system are any android end-users.

**End-User :**

**R.1 Manage Files**

R.1.1 Delete File

Input: Prompt to delete file

Description: Users are able to delete files in the file system.

R.1.2 View File Details

Input: Prompt to show details

Output: File properties display

Description: File Details like size, type and permissions are shown.

R.1.3 Share File

Input: Prompt to Share file

Output: Applications from which to share

Description: Share file to different applications from the file

manager.

R.1.4 Open File

Input: Open command

Output: Application from which file can be opened

R.1.5 Show Hidden Files

Description: Users can see all the hidden files in the system.

**R.2 Manage Folder**

R.2.1 Create Folder

Input: Prompt to create folder

Output: Created folder

R.2.2 Delete Folder

Input: Prompt to delete folder

Description: Users can delete the targeted folder.

R.2.3 Read Folder

Description: Users can open a folder from internal and external

storage and able to see all the files and folders

inside it.

**R.3 See Category wise Files**

Description: Users can easily find files by clicking on

Categories-images, audios, videos, and documents.

**R.4 Find Duplicate Files**

Description: Users can find out a list of all duplicate files

available on the system and users can also able to

delete duplicate files.

**Non-Functional Requirements:**

**Performance**

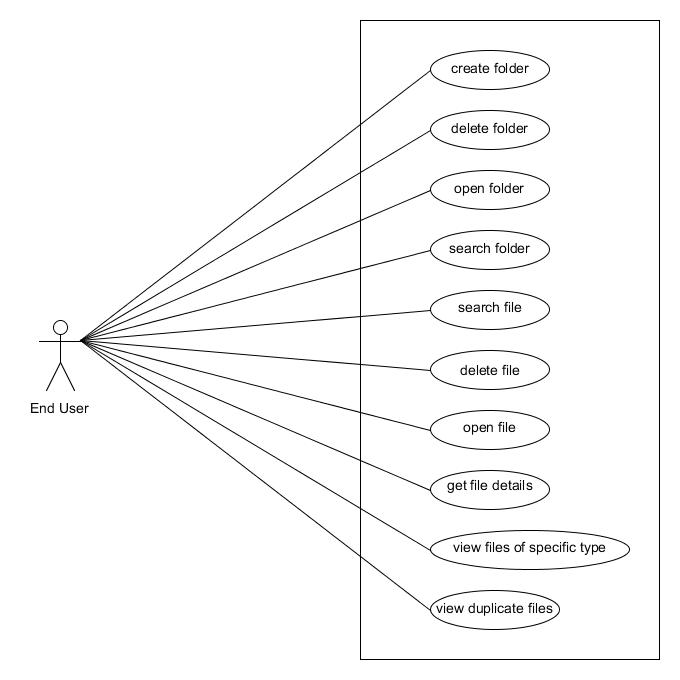
1)It responds to each and every query in less than 1 second

2)It operates with zero downtime.

3)Efficient Duplication and search results.

**Design**

**I) Use case Diagram :**

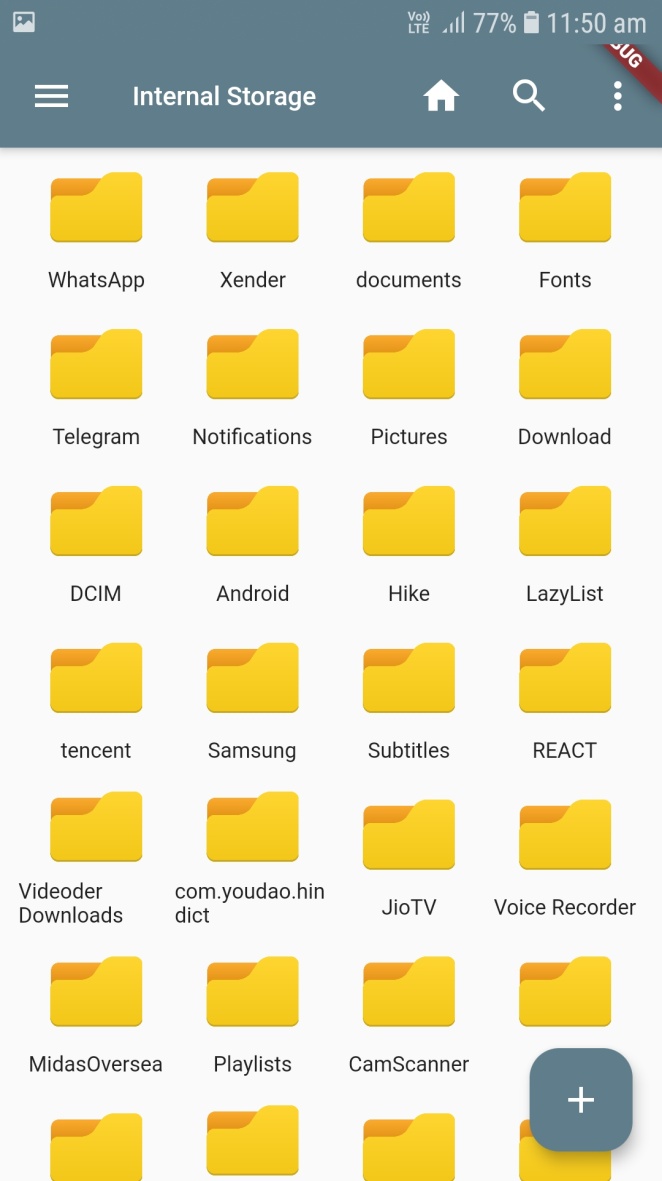
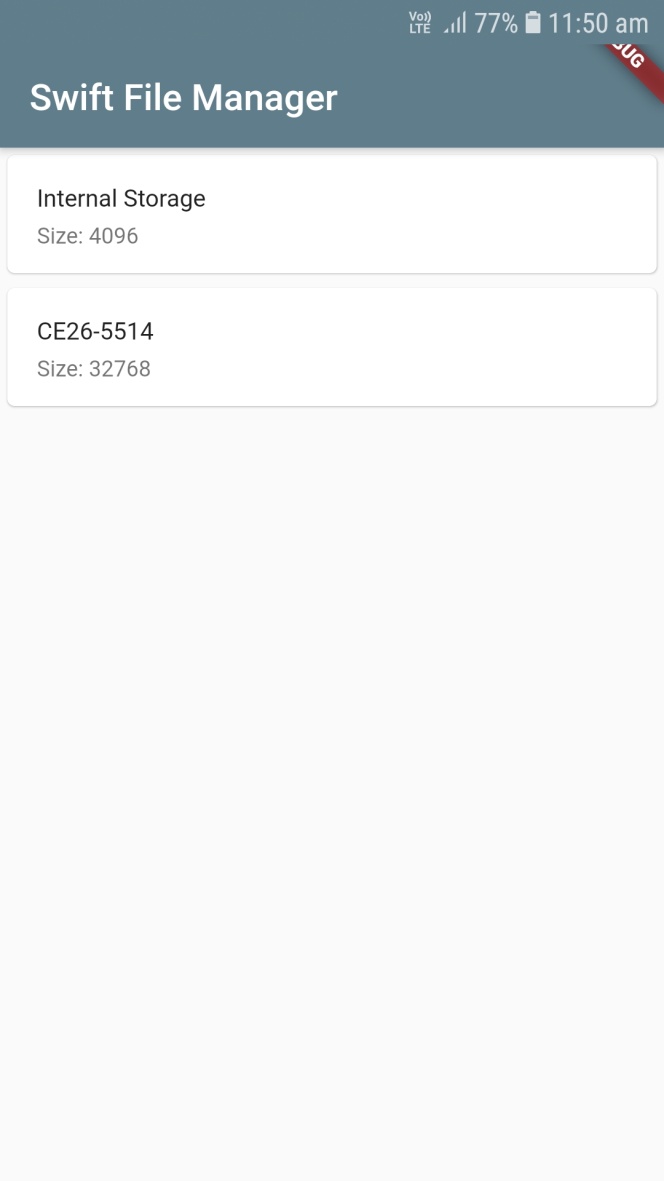


**Implementation Details**

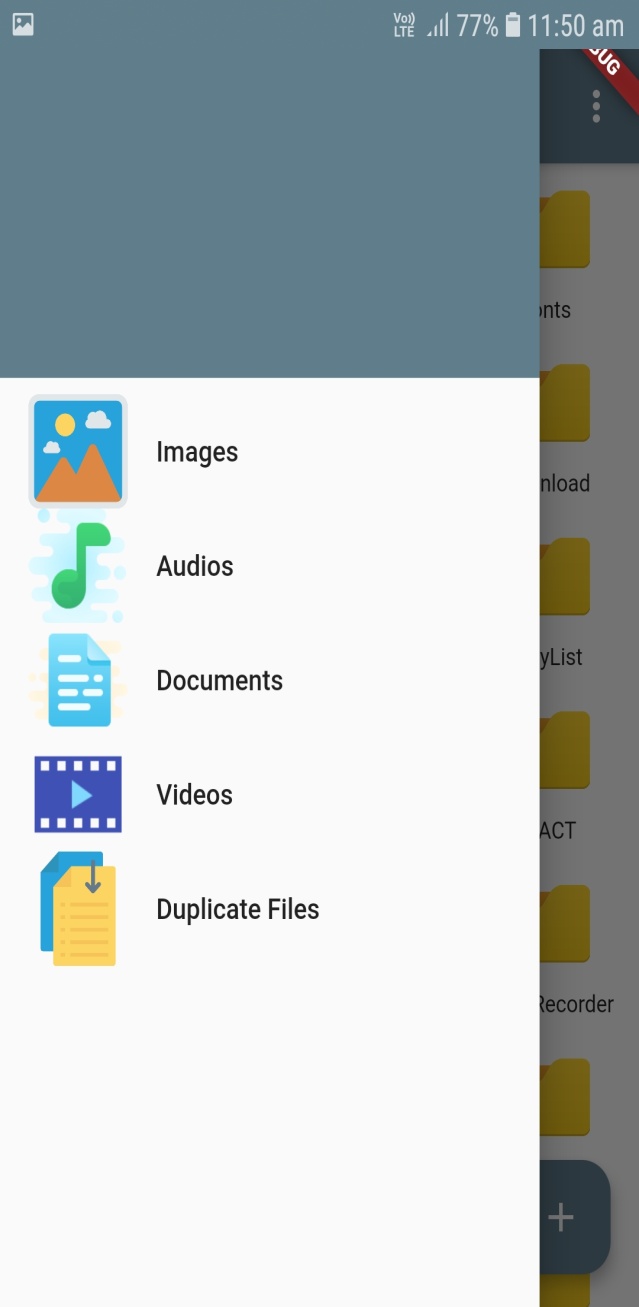
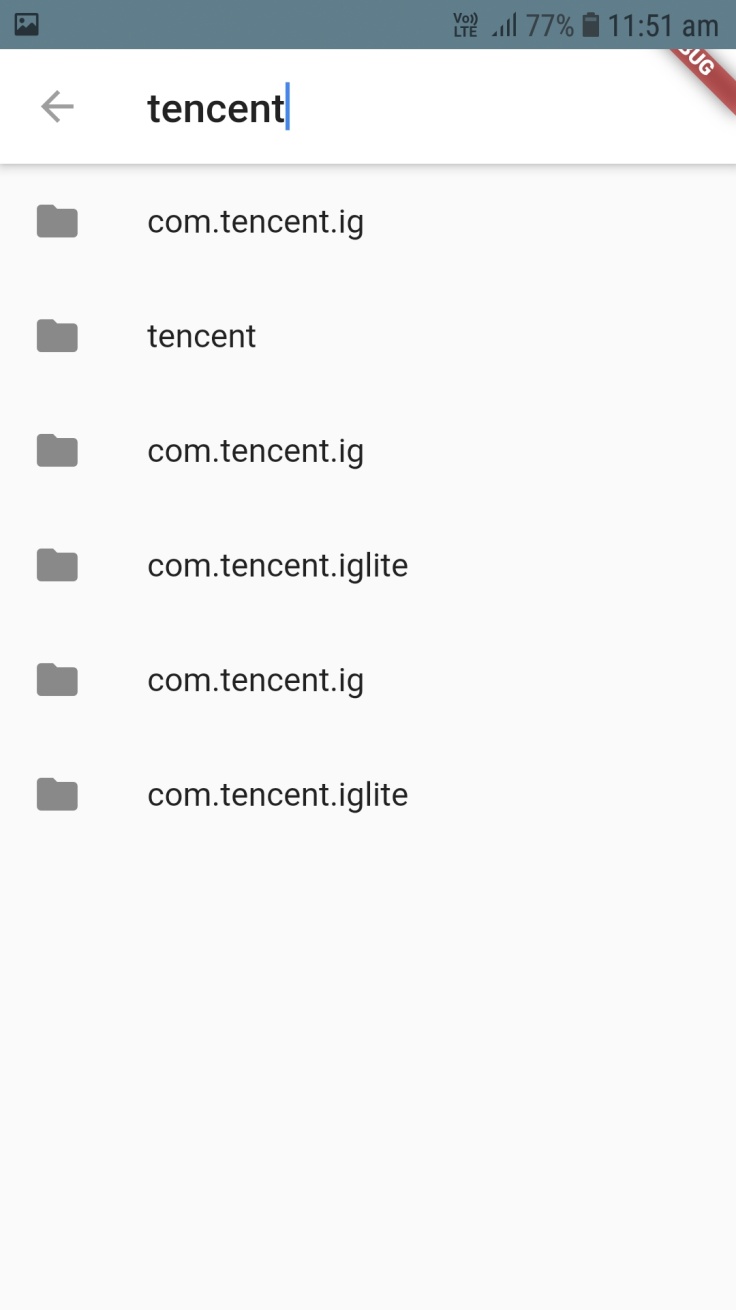
* Files/Folder Module
  + In flutter, everything is a widget so for file and folder displays used and FileTile view.
  + A *stateful* widget is dynamic: for example, it can change its appearance in response to events triggered by user interactions or when it receives data.
  + All modules are developed by extending the *stateful* widget.
* Search Module
  + Searching is done using getting all files and matching with the applied keyword.
  + It also counts all files in the system.
  + Searching takes 2 or 3 seconds.
* Category wise File module
  + Segregation of files done by the following steps:
    - Check whether it's a file or not.
    - If file then check if it matches the MIME(Multipurpose Internet Mail Extensions) of respected type or not.
    - Ex. for Image files - image/bmp, image/cis-cod, image/jpeg, image/tiff, image/gif, image/ief, image/png
    - If it matches mime type then add it to the list.
    - Then display using ListTile View and Card.
* Duplication Module:
  + To find duplicate files in the file system, first, find the checksum of file hash using the MD5 Algorithm.
  + MD5 has one important benefit above SHA1, SHA2, and upcoming SHA3: many platforms allow faster implementation of MD5 than other hashes. The performance in this use case varies highly according to the used hash function.
  + Calculating the hash will make the program run slow. It’s better to also **check the file size**.
  + Steps to find duplicate file:
    - Check if **file size** is equal
    - If step 1 passes, check the **file type**,
    - If step 2 passes, check the **hash at last.**
    - If step 3 passes, then add both files to the list.
  + The collision rate for this approach is 2^-89 in million files, which is very small.

**Screenshots**

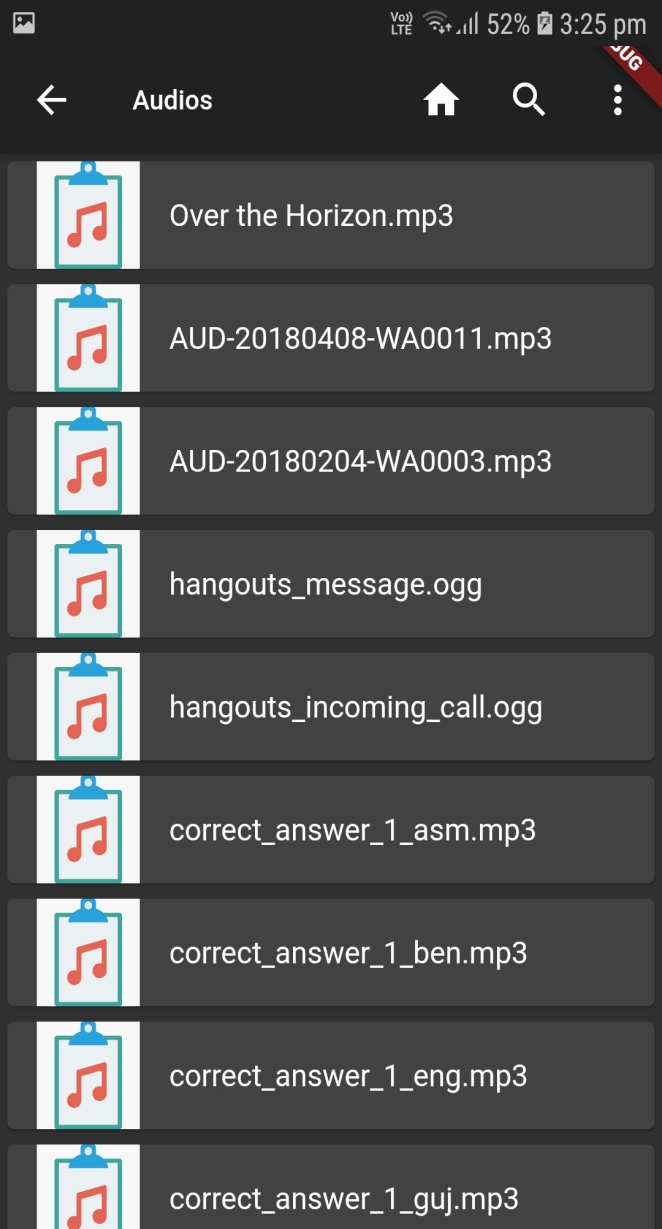
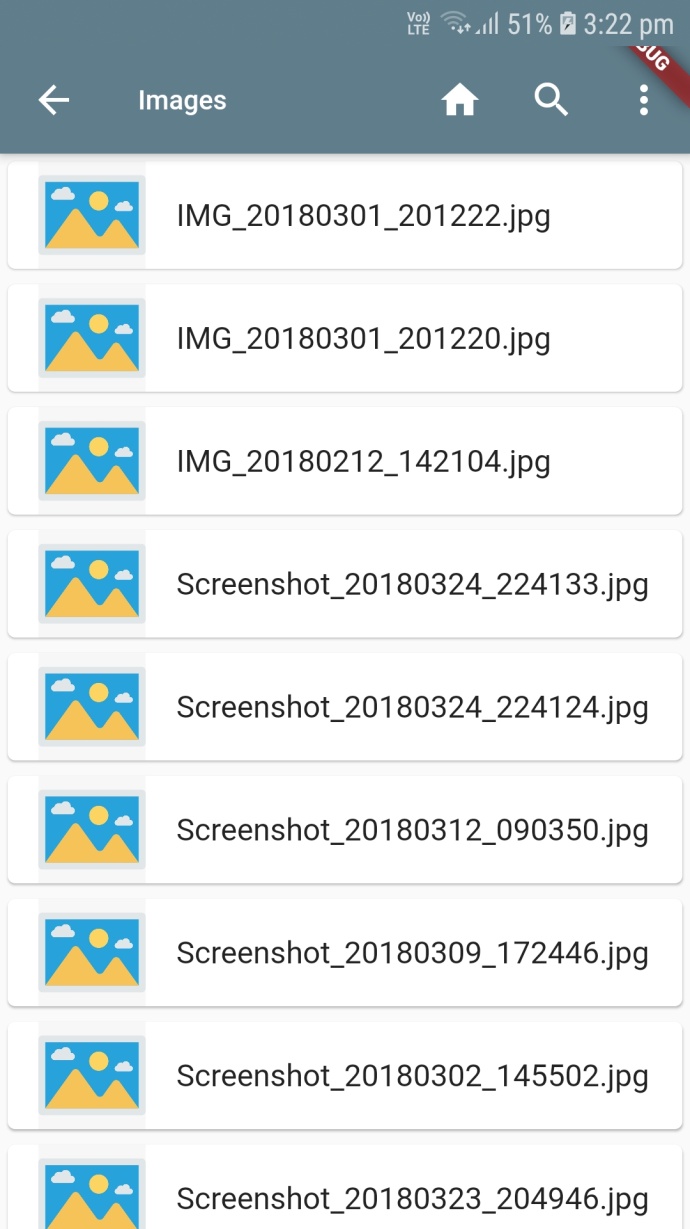
**Major Functions Screenshots:**

****

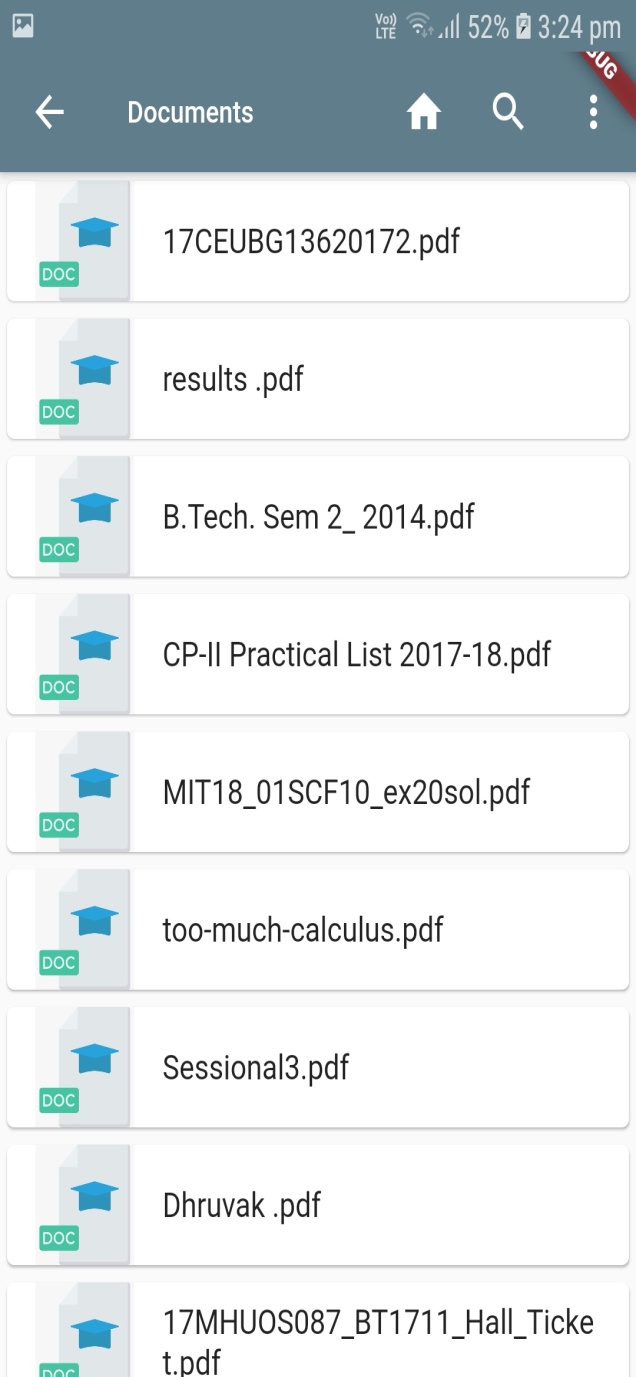
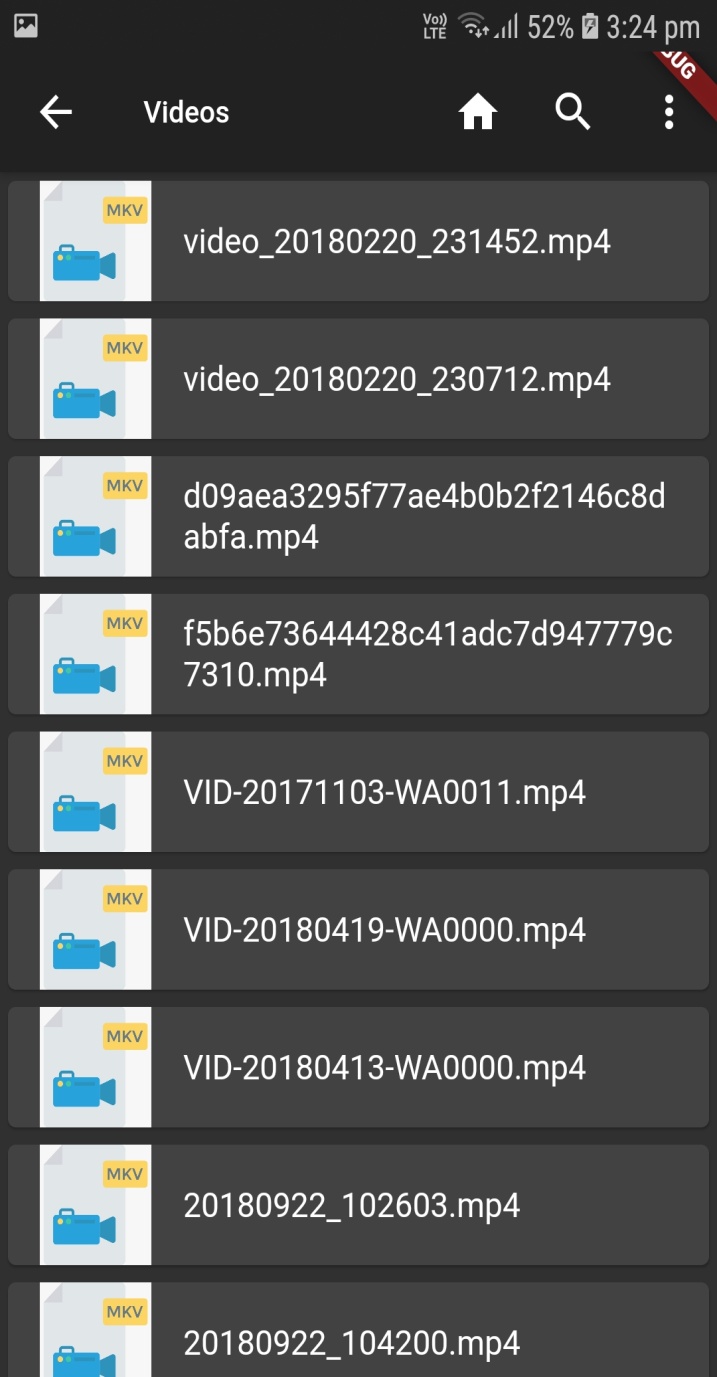
I) Home Screen II) Internal Storage Screen



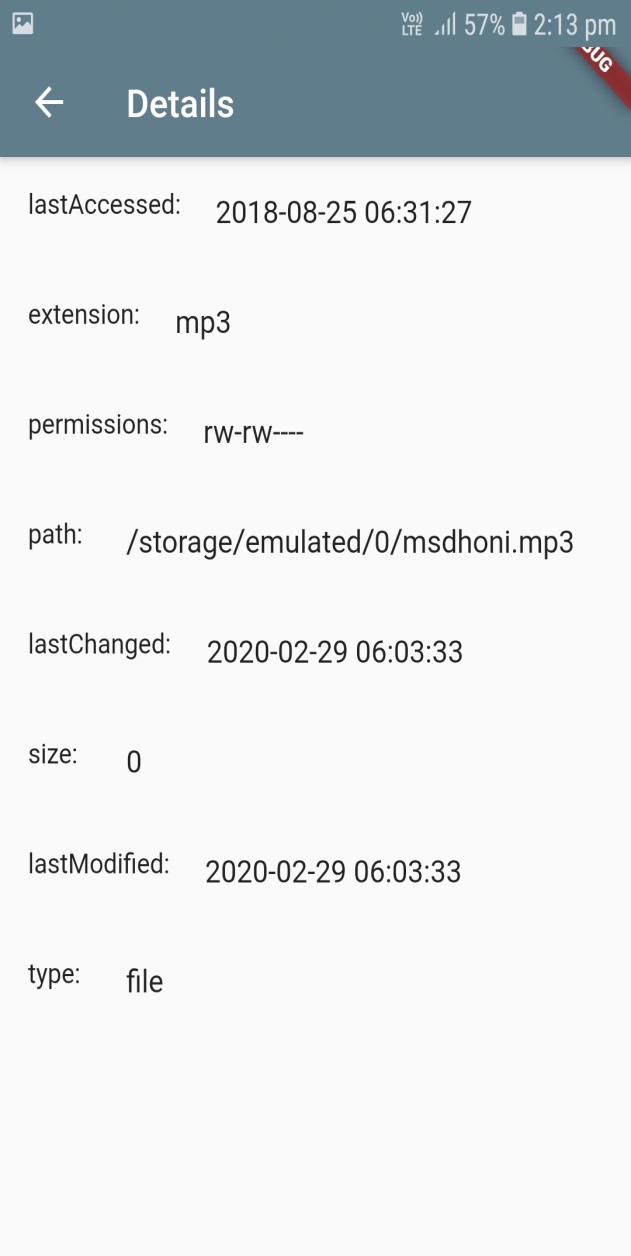
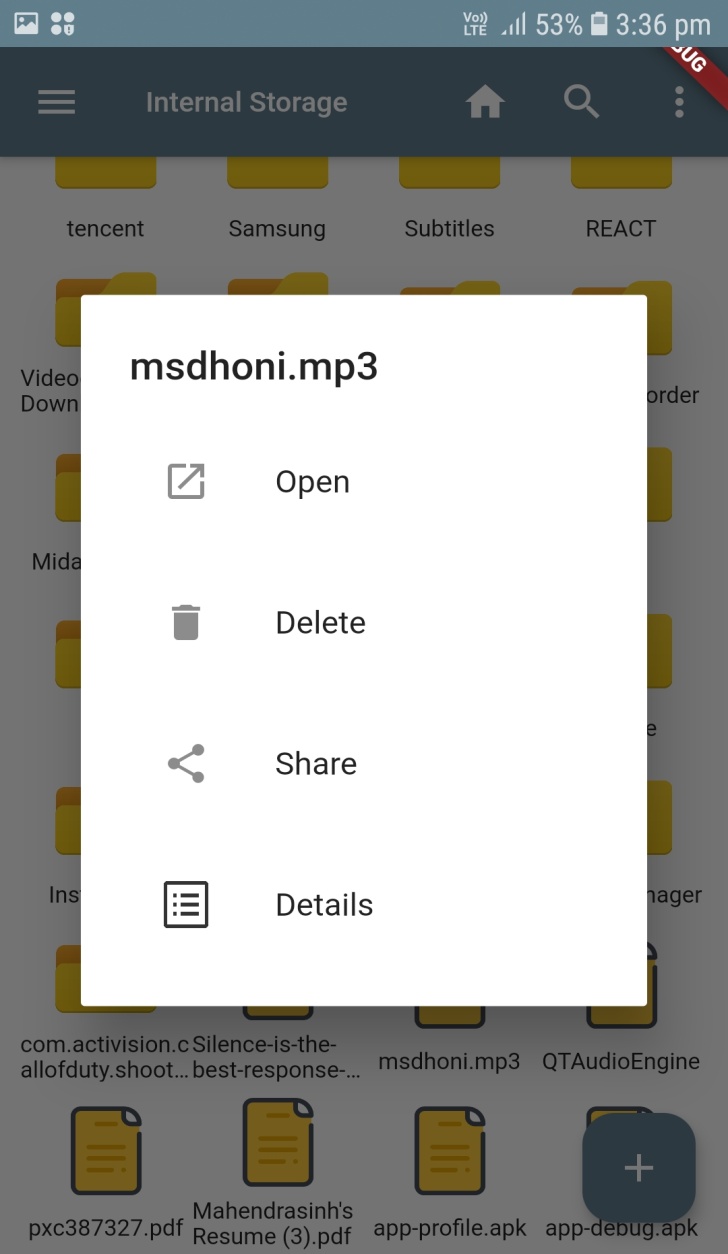
III) Search Screen IV) Drawer Screen



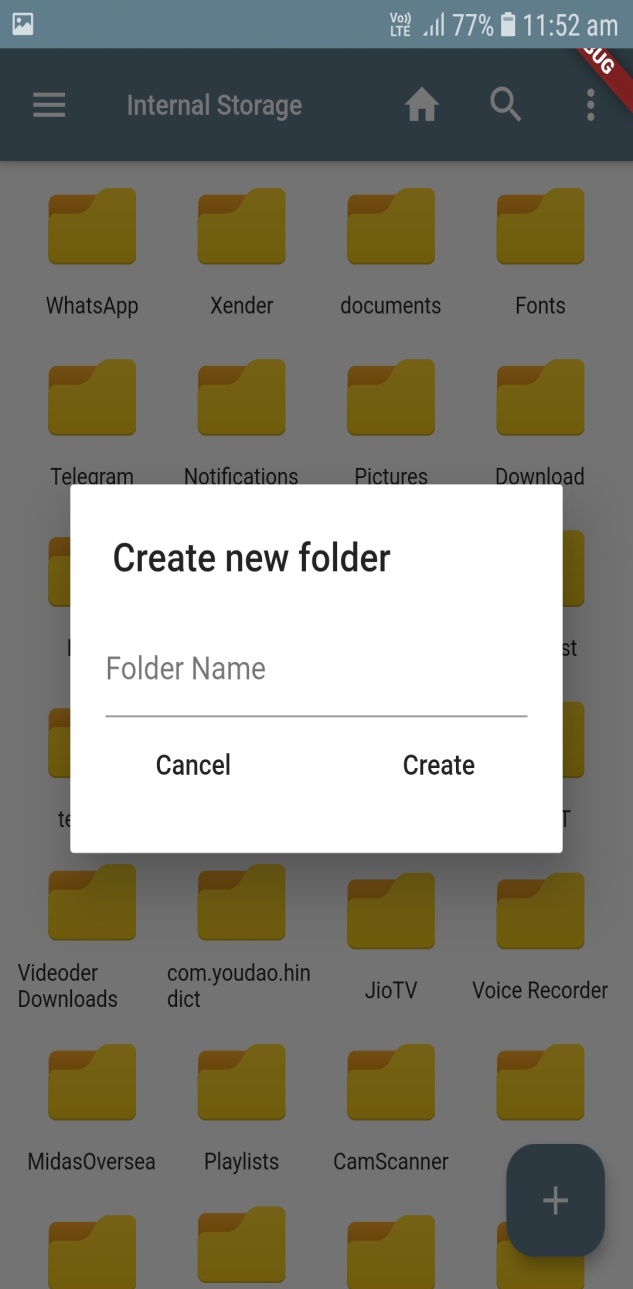
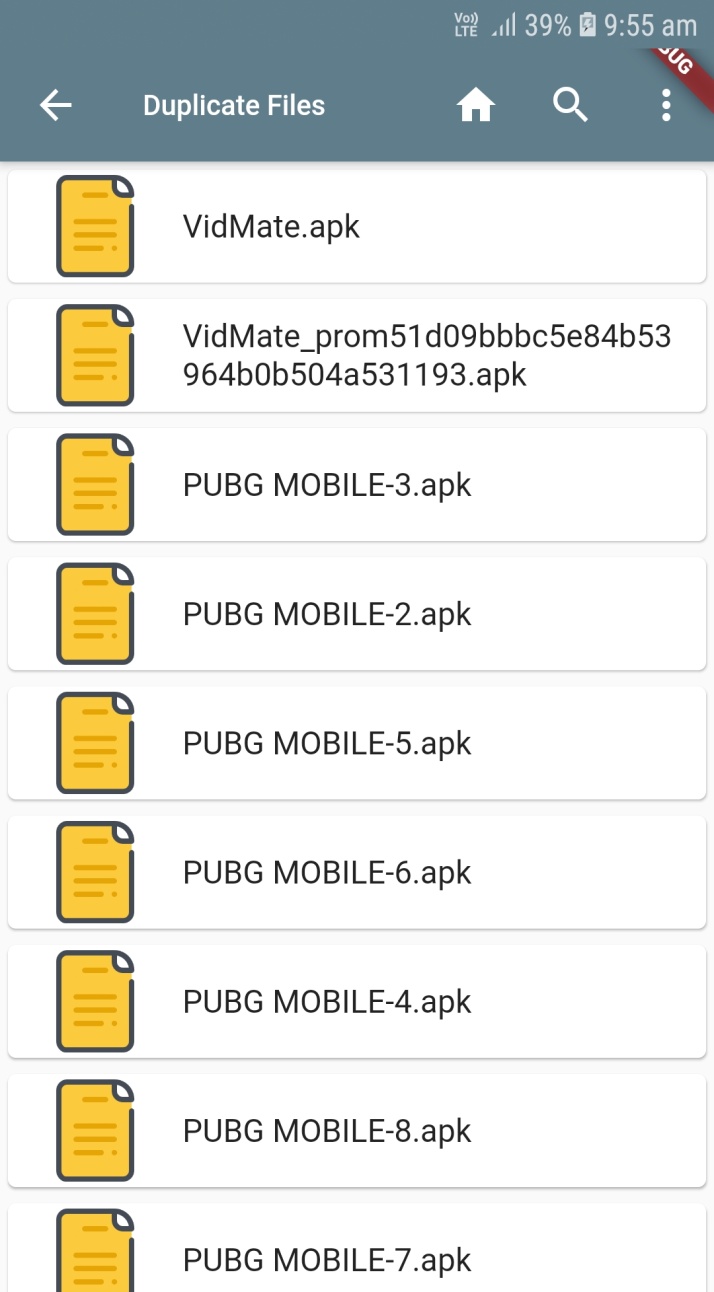
V) Images Screen VI) Audios Screen



VII)Videos Screen VIII) Documents Screen



IX) File Details Pop-up X) File Details Screen



XI) Duplicate Files Screen XII) Creation of Folder

**Conclusion**

Users have all the rights to use the basic functionality of the file manager. Users can easily get all the files of specific types and can easily navigate among folders. Users can also find all the duplicate files available in the system and also be able to delete them to make more space on the device.

**Limitation and Future Extension**

* Android application has lacked in terms of security aspects.
* Limitation
  + Duplication algorithm taking up to 2mins for the system containing more than 10000 files.
  + Move operation not implemented.
* Future Extension
  + Better User Interface
  + Efficient Searching
  + Cache deletion for better optimization
  + Login with third party

**BIBLIOGRAPHY**

For the successful implementation of this project, we referred to many websites and video tutorials. We developed an android application on visual studio code. We use Github as a project management tool. Mostly we referred to online material for the syntax of Dart.

Reference Websites:

* <https://dart.dev/>
* <https://flutter.dev/>
* <https://stackoverflow.com/>
* <https://medium.com/>