A

Mini-Project Report

on

PDF SCANNER APP

Submitted as partial fulfilment for the award of

BACHELOR OF TECHNOLOGY DEGREE

Session 2021-22

in

Computer Science & Engineering

Submitted By:

SHUBHAM SRIVASTAVA Roll. No: 114

SURYANSH SHARMA Roll.No:124

UNNATI AGARWAL Roll.No:126

Under the guidance of:

Ms. Tanu Shree

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GALGOTIAS COLLEGE OF ENGINEERING AND TECHNOLOGY, GREATER NOIDA

AFFILIATED TO
DR. A.P.J ABDUL KALAM TECHNICAL UNIVERSITY, LUCKNOW, UP

GALGOTIAS COLLEGE OF ENGINEERING & TECHNOLOGY

GREATER NOIDA, UTTER PRADESH, INDIA- 201306.

CERTIFICATE

This is to certify that the mini-project report entitled "**PDF SCANNER APP**" submitted by **Mr. SHUBHAM SRIVASTAVA (Roll.No:114), Mr. SURYANSH SHARMA (Roll.No:124), Ms. UNNATI AGARWAL (Roll.No:126),** to the Galgotias College of Engineering & Technology, Greater Noida, Utter Pradesh, affiliated to Dr. A.P.J. Abdul Kalam Technical University Lucknow, Uttar Pradesh in partial fulfillment for the award of Degree of Bachelor of Technology in Computer science & Engineering is a bonafide record of the project work carried out by them under my supervision during the year 2021-2022.

Ms. Tanu Shree
Assistant Professor
Deptt. of CSE

Dr. Vishnu Sharma
Professor and Head
Deptt. of CSE

ACKNOWLEDGEMENT

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend my sincere thanks to all of them.

We are highly indebted to **Mr. Sardesh Sharma** for his guidance and constant supervision. Also, we are highly thankful to him for providing necessary information regarding the project & also for his support in completing the project.

We are extremely indebted to **Dr. Vishnu Sharma**, HOD, Department of Computer Science and Engineering, GCET and **Ms. Tanu Shree**, Project Coordinator, Department of Computer Science and Engineering, GCET for their valuable suggestions and constant support throughout my project tenure. We would also like to express our sincere thanks to all faculty and staff members of Department of Computer Science and Engineering, GCET for their support in completing this project on time.

We also express gratitude towards our parents for their kind co-operation and encouragement which helped me in completion of this project. Our thanks and appreciations also go to our friends in developing the project and all the people who have willingly helped me out with their abilities.

SHUBHAM SRIVASTAVA
SURYANSH SHARMA
UNNATI AGARWAL

ABSTRACT

Android Development Has been developing at a very fast pace all over the world. The more the need of the applications, the more is the development. In this project we come with a good project work that will be useful to the whole society. as , worldis growing we need materials in compressed and digital form. So, in this we have created a pdf scanner app which will be helpful to store and collect data at one placeand can be efficiently used very well.

In this project we have given you the whole process of app making and its future projections for our coming world.

This app will be useful for all age peoples to store and modify data or compile large documents at one place for any further use. You don't need to carry a wholesome ofbooks or documents with you all the time. It is the time to utilize the technology in a more advantageous manner and be the one to save time and caliber.

Keywords: Android, Application, Gradle, Java, PDF, Scanner, Screen, Software, Technology, XML.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

A PDF Scanner app is a professional utility program that lets you to scan and organize your paper in PDF format. It provides a powerful ability to scan and repair your paper documents. A-PDF Scan Paper uses clear thumbnails and metadata to allow you to easily organize, filter, secure, send and retrieval your scanned documents.

1.1 MOTIVATION

As Covid 19 spreads throughout the world, all professional places like offices, colleges and schools etc are now running in online mode. Therefore, there increases the sudden increase in digital documents. Now everyone has to upload their paperwork in form of PDF on the institution webpage. Therefore, need for PDF scanner apps also increased throughout the world, which is where our mini project PDF SCANNER APP reduces that problem and helps everyone by scanning and converting clicked images into PDF format.

1.2 DESCRIPTION

The scanner app turns your device into a powerful portable scanner that recognizes text automatically (OCR) and allows you to save to multiple file formats including JPG, JPEG and PNG etc. Unlike other pdf scanners, this app will provide us with various different shades, features for our documentation pages, like not only original or black and white features will pop up, insteadwe will be getting gray shades, inverted, sepia, etc.

This app will help in fast documentation of our images whenever we want to create one.
It willhelp in easily saving and sharing our made PDFs.

The most intelligent scanner app. Scan anything — receipts, notes, documents, photos, businesscards, whiteboards — with text you can reuse from each PDF and photo scan.

1.3 DEFINITIONS

- Queen Scanner is the best scanner app that will turn your phone into a PDF scanner.
- Convert images to pdf in a simple tap.
- Turn Paper and Whiteboard to PDF
- Scan anything in a snap.
- With Adobe Scan, easily capture and convert documents, forms, business cards and whiteboards into high-quality Adobe PDFs. And with different capture modes along withautomatic boundary detection, you can get the best scans every time
- Intelligently cleaner and clearer scans.
- Using Adobe Sensei, the powerful AI behind many of our intelligent features, AdobeScan corrects image perspective and sharpens handwritten or printed text, while
- Removing elements you don't want, like glare and shadow.

1.4 CHAPTER OVERVIEWS

Chapter-2- Literature Review- This chapter will contain data regarding introduction of our app,theory behind the app making and pdf scanner. It will also contain detailed analysis of how we searched and collected ideas for the same. Then it will include the conclusion of a literature review of how we came to the conclusion of deciding our work.

Chapter-3- Problem Formulation- In Chapter 3 we discussed about the How you are arriving at the problem? And the problem statement and its depiction

Chapter-4- Methodology- This will include the introduction of our research design, research instrumentation or data collection. This will also inculcate information regarding our data analysis and ethics involved in our project which will provide no harm to the society. Thenfinally we came to a conclusion about how we did our research findings and analysis.

Chapter-5- Implementation- This part will include our proper material of our project which will include the screenshots of our app to give you an overview of how it will look thereafter. And it will also contain data flow diagrams (DFD).

Chapter-6- Project Relevance- It is the conclusion part, comparing the data analysis of the our app with the older apps.

Chapter-7- Conclusion and Future Projections-This part will be briefing us about a summary of findings and conclusions.

This will also brief us about the summary of contributions we made and our team. after that therewill be a future projection of our project on how it will be going to help our society in future andethics related to it.

CHAPTER 2: LITERATURE REVIEW

2.1 INTRODUCTION

PDF scanner is the app using which we can assemble our images or pdfs at one place.

This app turns your mobile into a portable scanner, which can be used for scanning handwrittennotes and printed documents. It automatically detects the edge of the page over a contrasting surface. After the page is detected, it compensates any perspective from the image, adjusting it to a 90 degree top view and saves it to a folder on the device.

It is also possible to launch the application from any other application that asks for a picture.

2.1.1 What is PDF?

Portable Document Format (PDF), standardized as ISO 32000, is a file format developed by Adobe in 1992 to present documents, including text formatting and images, in a manner independent of application software, hardware, and operating systems.Based on the PostScript language, each PDF file encapsulates a complete description of a fixed-layout flat document, including the text, fonts, vector graphics, raster images and other information needed to display.

PDF combines three technologies:

- A subset of the PostScript page description programming language, for generating thelayout and graphics.
- A font-embedding/replacement system to allow fonts to travel with the documents.
- A structured storage system to bundle these elements and any associated content into a single file, with data compression where appropriate.

2.1.2 What is an Android APP?

Android App is a software designed to run on an Android device or emulator. The term also refers to an APK file which stands for Android package. This file is a Zip archive containing appcode, resources, and meta information.

Android apps can be written in Kotlin, Java, and C++ and are run inside Virtual Machine. Theofficial development environment is Android Studio.

2.2 BACKGROUND INFORMATION

Queen Scanner – PDF Scanner phone version is perfectly compatible with tablets now! We willkeep updating the phone version only and have stopped updating the HD version. Please installthe Queen Scanner app to get the most updated features and experience.

The Queen Scanner - Simple Scanner is a PDF document scanner application that turns your phone into a portable scanner. you can scan documents, photos, receipts, reports, or just aboutanything. The scan will be saved to the device in image or PDF format.

2.3 DETAILED DESCRIPTION

We have used a popular Android developing app named ANDROID STUDIO for developing ourproject and executing and running it on our Mobile Phones.

- Mobile phone document, automatically remove the clutter background, generate high-definitionJPEG pictures or PDF files.
- A variety of image processing modes, you can manually adjust the image parameters, with amobile phone can be paper documents, quickly turned into a clear electronic draft.
- Scan color, grayscale, or black and white
- Can be used in office, school, home and any place you want
- Automatically detects page edges
- 5 levels of contrast for clear monochrome text
- Set the PDF page size (Letter, Legal, A4, etc.)

- Thumbnail or list view, sorted by date or title
- Simple scanners are optimized to run very fast.
- Quick search by document title
- General - A single application that works on your phone!

2.4 CONCLUSION

Basically, this project is about scanning the image and making pdfs out of it. It has many beautiful features which make our life easy and resolve our issues of assembling our images and pdfs.

Name and organize your scan to a folder, or share it in the following ways:

- Automatically upload JPG and PDF files to the phone memory
- Backup and restore synchronization files between multiple devices
- E-mail, print, Fax
- WIFI, connects directly to your computer
- Support importing PDF files into JPG.
- Support for adding tags for quick file search.
- Support OCR text recognition, export text.

CHAPTER 3: PROBLEM FORMULATION

3.1 INTRODUCTION

Problem formulation is the study and analysis of the problem for which the project was started. This chapter will be consisting of description of problem domain (which will give you an idea of why there is need for a PDF Scanner App), problem statement, the block diagram, the objectives of our project as if what is the aim of our project and how it is going to solve the list of problems listed in the problem statements.

3.2 PROBLEMS FACED

- The older apps were slow and did not provide the sufficient features for color changing, and to use different images formats.
- This app helps in making our work even faster and it will do automatic editing without going on each page for editing, henceforth will save time.
- It also has an option for manual editing for those who wish to do.
- Older applications contain lots of bugs which increases the malfunctioning of the app.

3.3 OBJECTIVES

The main objective of this project is to implement new PDF SCANNER APP, which will resolve all the problems faced by previous applications, by enlisting all those features in our new applications

The main objectives are as follows:

- To avoid unnecessary Waste of Memory.
- Provide efficient, automatic and smart image conversion.
- Longer running of our new application.

3.4 CONCLUSIONS

In this chapter we learnt about the need of this project and the problem it solves increasing the effectiveness of the application. We took a look on the major problem faced by the older applications in order to give the results, and we looked at some new features from our new application which will make it better from older applications in order for fulfilling the other needs of the society.

CHAPTER 4: METHODOLOGY

4.1 Proposed work and Specifications

4.1.1 Android Studio

Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems or as a subscription-based service in 2020. It is a replacement for the EclipseAndroid Development Tools (E-ADT) as the primary IDE for native Android application development.

Android Studio was announced on May 16, 2013 at the Google I/O conference. It was in the early access preview stage starting from version 0.1 in May 2013, then entered beta stage startingfrom version 0.8 which was released in June 2014.The first stable build was released in December 2014, starting from version 1.0.

4.1.2 JAVA

Java is a high-level, class-based, object-oriented programming language that is designed to haveas few implementation dependencies as possible. It is a general-purpose programming languageintended to let programmers write once, run anywhere (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM)regardless of the underlying computer architecture. The syntax of Java is similar to C and C++, but has fewer low-level facilities than either of them. The Java runtime provides dynamic capabilities (such as reflection and runtime code modification) that are typically not available intraditional compiled languages. As of 2019, Java was one of the most popular programming languages in use according to GitHub, particularly for client–server web applications, with a reported 9 million developers.

4.1.3 SQLite

SQLite is a relational database management system (RDBMS) contained in a C library. In contrast to many other database management systems, SQLite is not a client–server database engine. Rather, it is embedded into the end program.

SQLite generally follows PostgreSQL syntax. SQLite uses a dynamically and weakly typed SQL syntax that does not guarantee the domain integrity.This means that one can, for example, insert a string into a column defined as an integer. SQLite will attempt to convert data between formats where appropriate, the string "123" into an integer in this case, but does not guarantee such conversions and will store the data as-is if such a conversion is not possible.

SQLite is a popular choice as embedded database software for local/client storage in application software such as web browsers. It is arguably the most widely deployed database engine, as it is used today by several widespread browsers, operating systems, and embedded systems (such as mobile phones), among others. SQLite has bindings to many programming languages.

4.1.4 XML

Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. The World Wide Web Consortium's XML 1.0 Specification of 1998 and several other related specifications—all of them free open standards—define XML.

The design goals of XML emphasize simplicity, generality, and usability across the Internet. It is a textual data format with strong support via Unicode for different human languages. Although the design of XML focuses on documents, the language is widely used for the representation of arbitrary data structures such as those used in web services.

Several schema systems exist to aid in the definition of XML-based languages, while programmers have developed many applications programming interfaces (APIs) to aid the processing of XML data.

4.1.6 FIREBASE

Firebase is a platform developed by Google for creating mobile and web applications. It was originally an independent company founded in 2011. In 2014, Google acquired the platform and it is now their flagship offering for app development.

4.1.7 GRADLE

Gradle is a build automation tool for multi-language software development. It controls the development process in the tasks of compilation and packaging to testing, deployment, and publishing. Supported languages include Java (as well as Kotlin, Groovy, Scala), C /C++, and JavaScript. The other, if not the major function of Gradle is to collect statistical data about the usage of software libraries around the globe.

Gradle builds on the concepts of Apache Ant and Apache Maven, and introduces a Groovy- & Kotlin-based domain-specific language contrasted with the XML-based project configuration used by Maven. Gradle uses a directed acyclic graph to determine the order in which tasks can be run, through providing dependency management. Gradle runs on the JVM.

Gradle was designed for multi-project builds, which can grow to be large. It operates based on a series of build tasks that can run serially or in parallel. Incremental builds are supported by determining the parts of the build tree that are already up to date; any task dependent only on those parts does not need to be re-executed. It also supports caching of build components, potentially across a shared network using the Gradle Build Cache. It produces web-based build visualization called Gradle Build Scans. The software is extensible for new features and programming languages with a plugin subsystem.

Gradle is distributed as open-source software under the Apache License 2.0, and was first released in 2008.

CHAPTER 5: IMPLEMENTATION

5.1 SCREENSHOTS WITH WORKING

5.1.1 HOME SCREEN:

This will be the home screen of our project which will include the features like we can scan images using the camera, from the gallery. We can also scan documents using the documentscanner feature. We can also manage pdfs like sharing, saving or renaming, etc.

There is an option of setting also which provides privacy policy and terms and conditions of our app.

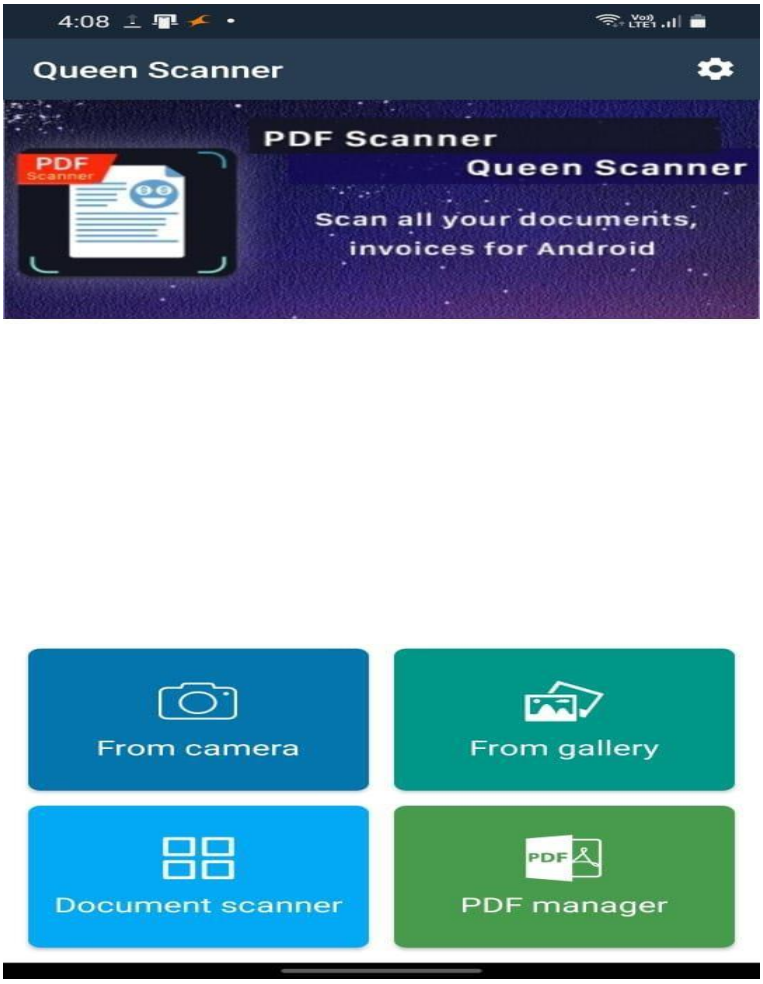

Fig 5.1 Home Screen

5.1.2 FROM CAMERA OPTION

This feature is actually showing the use of a camera feature using which we can take pictures andconvert them into pdfs. Moreover, there is a feature of retry, in case of any blurred picture improper picture we can also retry the capturing.

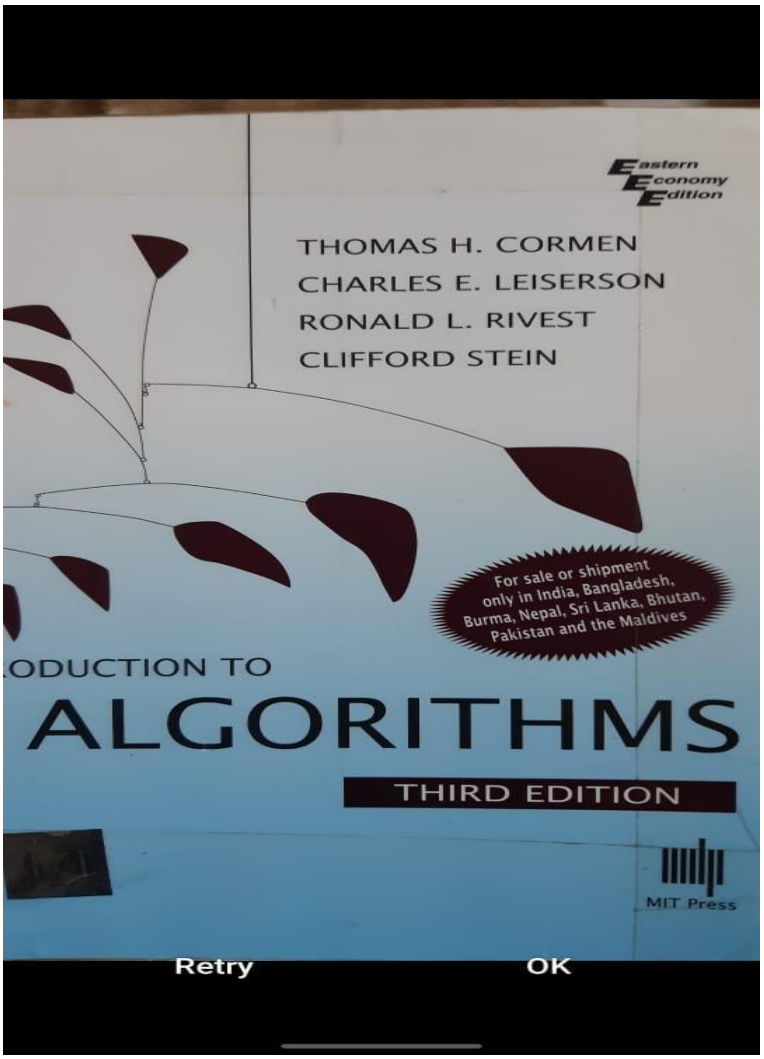

Fig 5.2 From Camera Option

5.1.3 FROM GALLERY OPTION

This image shows the feature of “from gallery”. We can select this feature and using this feature,images of the gallery can be chosen and inserted properly.

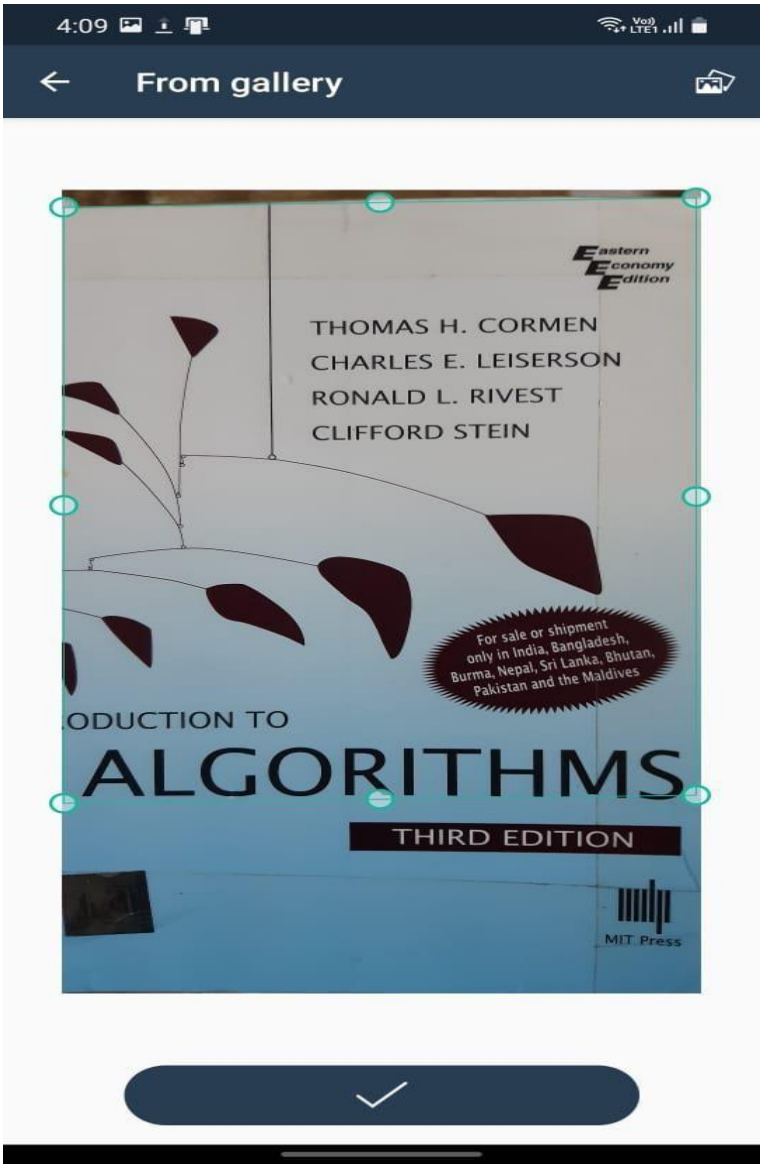

Fig 5.3 From Gallery Option

5.1.4 VARIOUS OPTION WHILE ADJUSTING IMAGES

1. This feature is again the colour feature of the page. Basically, this screenshot is showing the greyscale feature which makes our image to grayscale colour and can be saved.

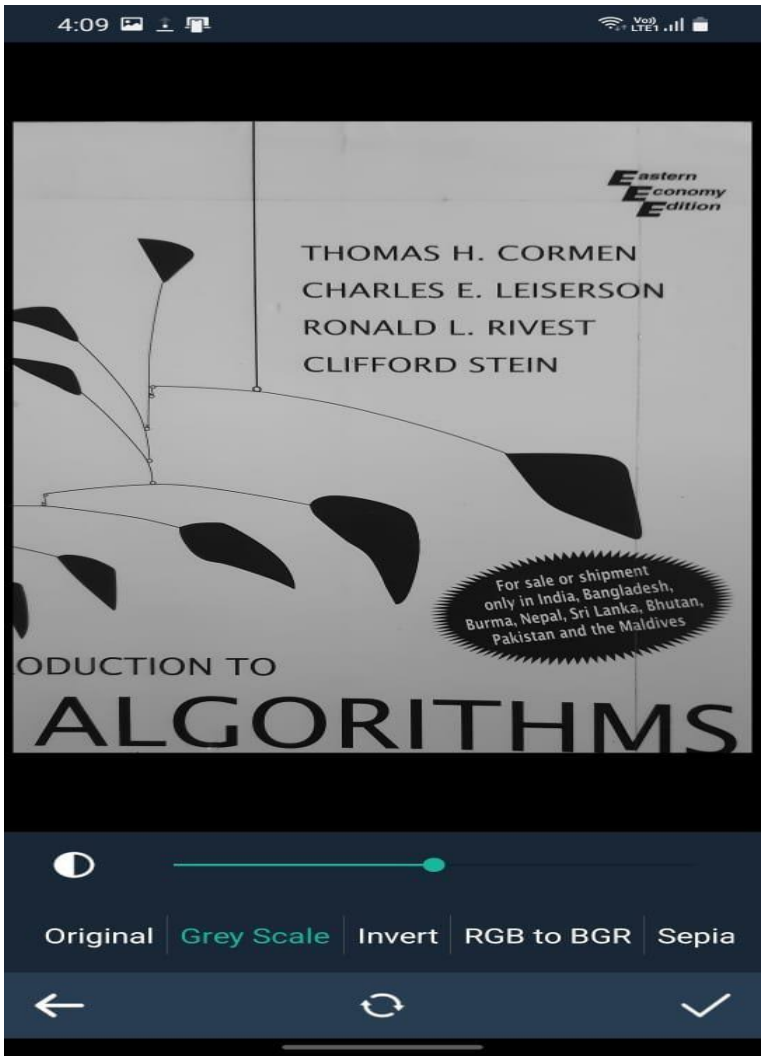

Fig 5.4 GreyScale

2. This is the feature using which we can invert the image in case the image taken is not in theportrait mode or not aligned properly.

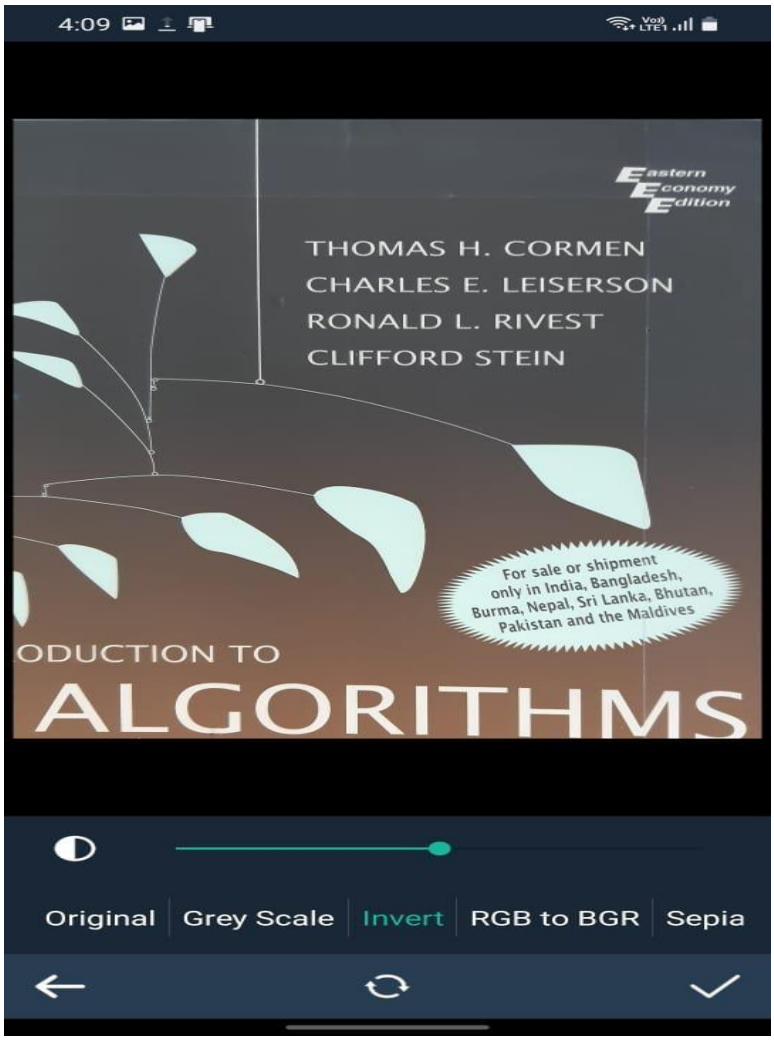

Fig 5.5 Invert

3. So this is the inverted image using the invert feature and after inverting there are other options too like we can use pinhole, kodachrome, technicolor and saturation etc.

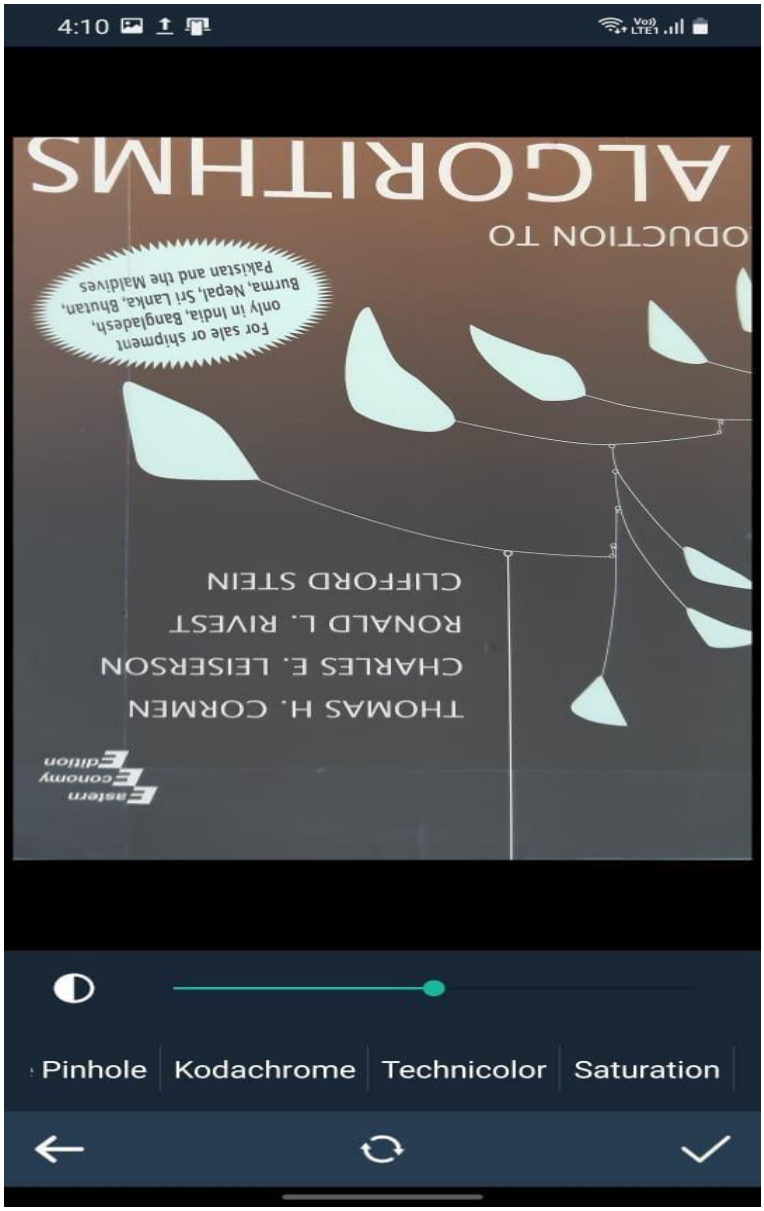

Fig 5.6 Flip

4. This is the screenshot of the screen where it is visible how our documents will be going to store properly.

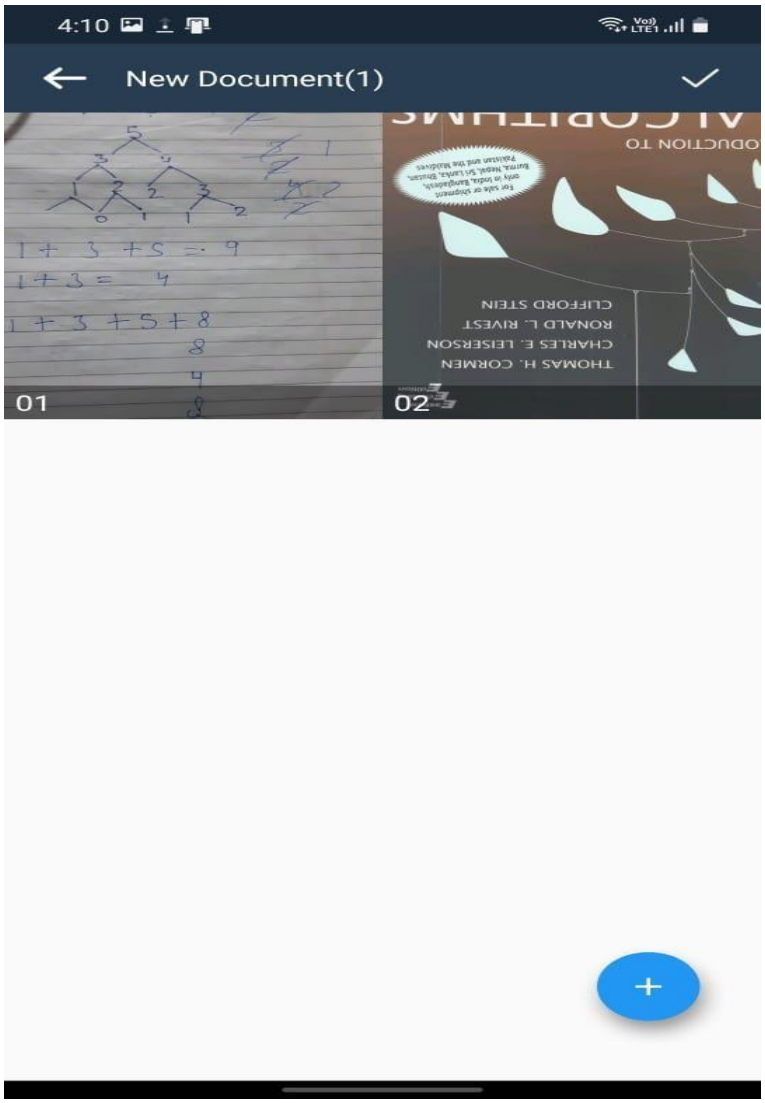

Fig 5.7 Overview

5.1.5 SAVE OPTION

1. This is the screenshot which is showing that next, if we want to create new documents, we havethe option of from gallery, from camera and no option chosen. Using these we can create new documents and save them to the scanner.

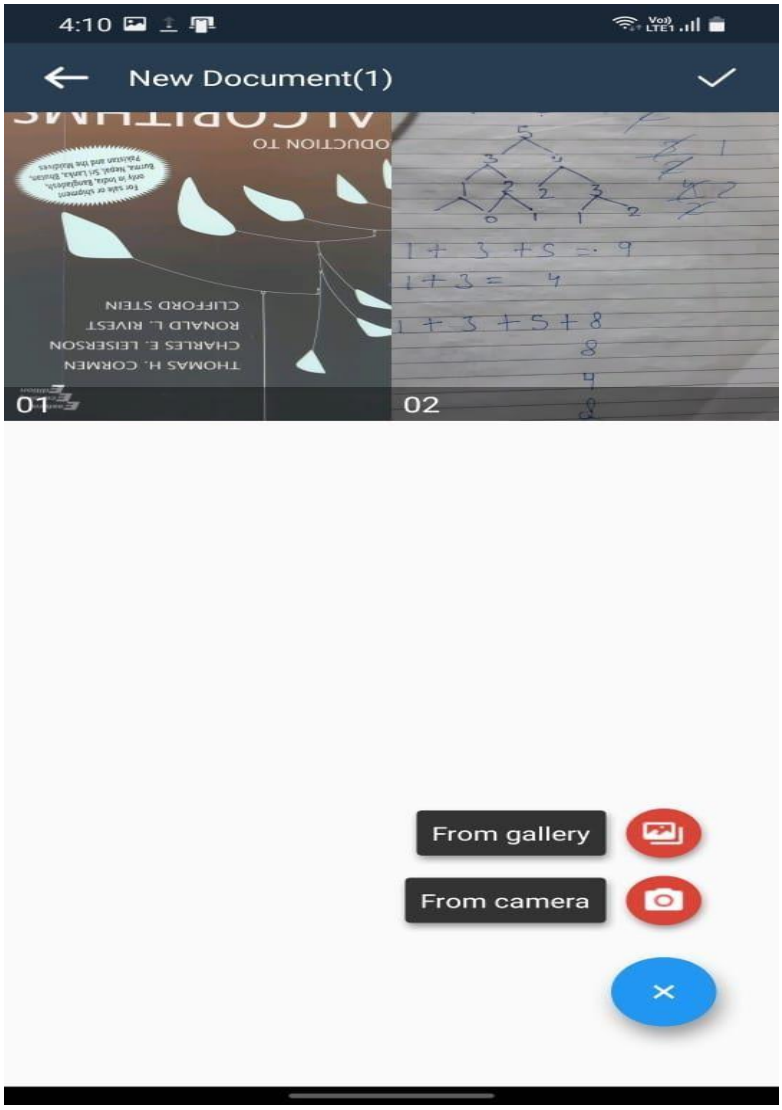

Fig 5.8 Save Option

2. This is the screenshot showing how we are going to select our images from various options in our phone. These will be the options visible to us while we need to select the image. Like we canselect from drive, from gallery, from recent folder, etc.

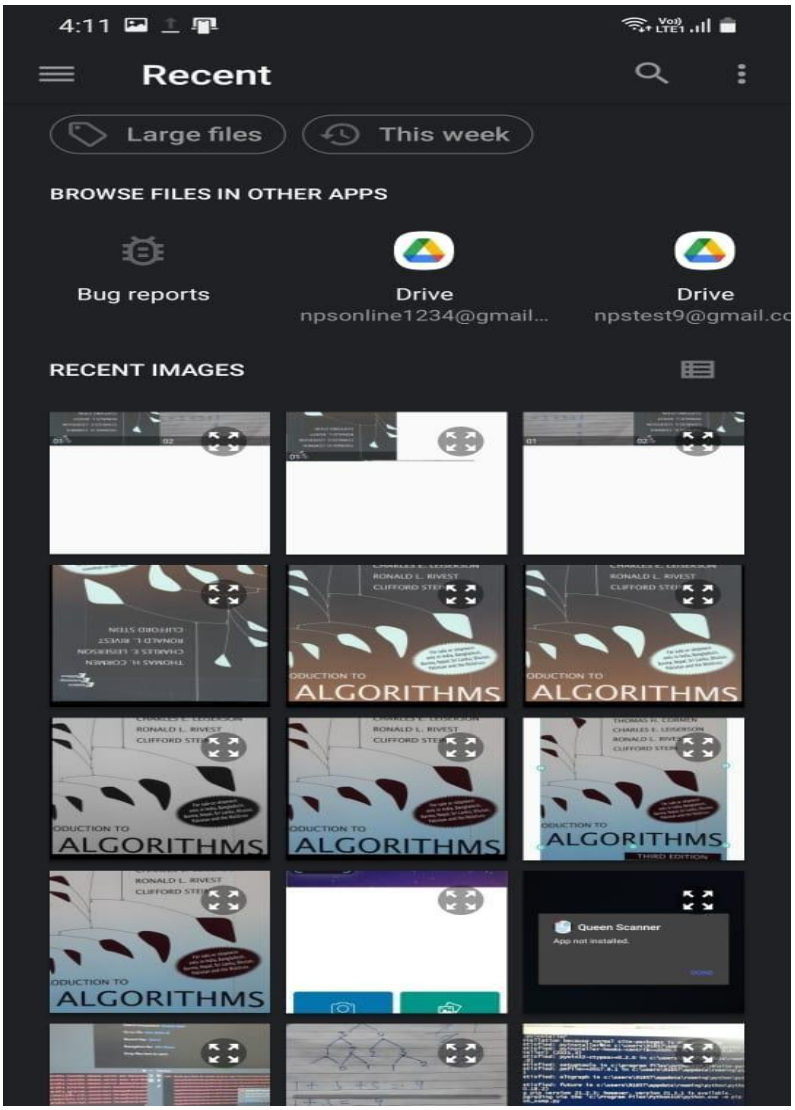

Fig 5.9 Gallery Overview

5.1.6 REVIEW OPTION

We can also change the format of saving our documents. Like we can save it row wise instead column wise. Here you can find your all documents in a proper date and time order.

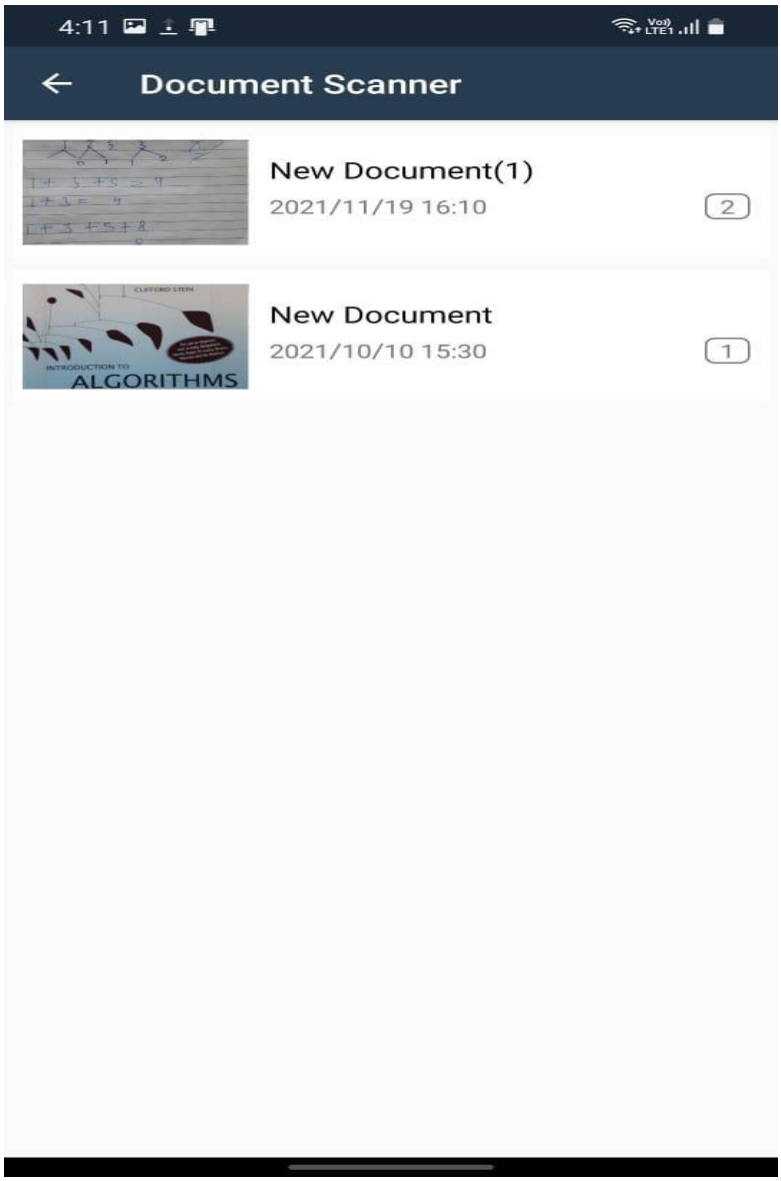

Fig 5.10 Review Option

5.1.7 SETTINGS

This is the screenshot showing the settings feature where we are getting the option of rating,sharing and feedback.

Rating means how much you like the app, sharing the documents and giving the feedback.

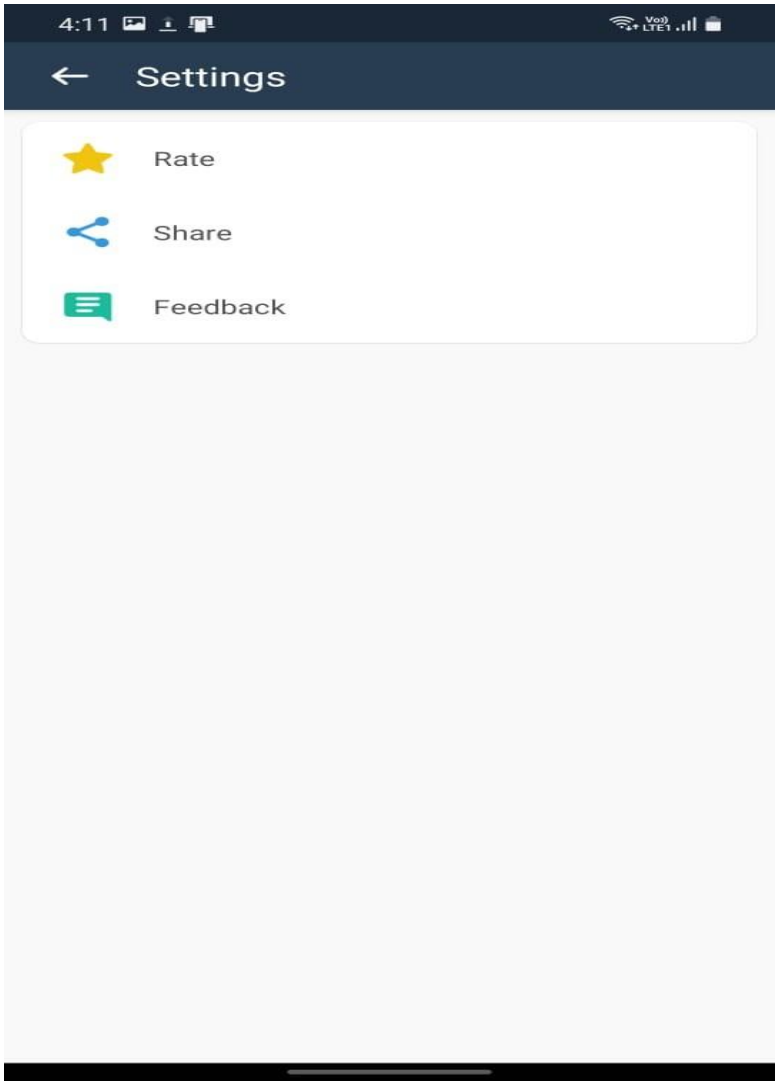

Fig 5.11 Settings

5.2 Data Flow Diagram (DFD)

5.2.1 Level 0

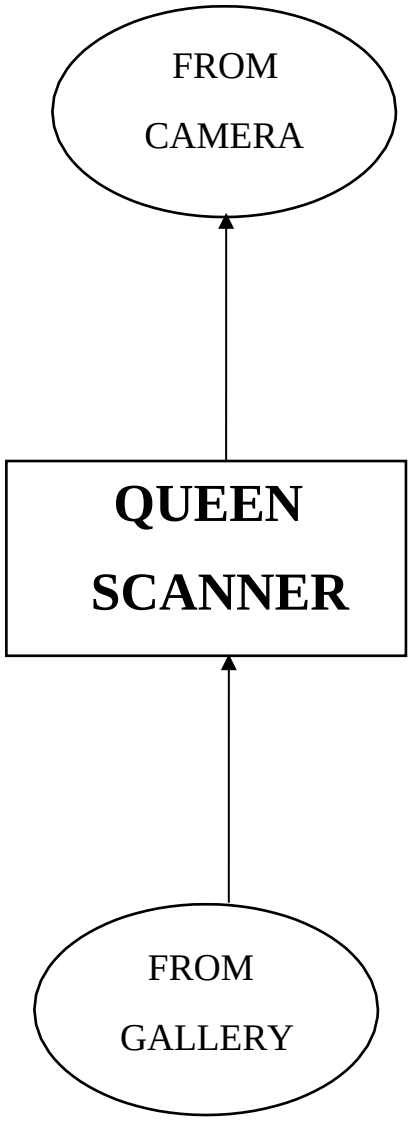

Fig 5.12 Level 0 DFD

5.2.2 Level 1

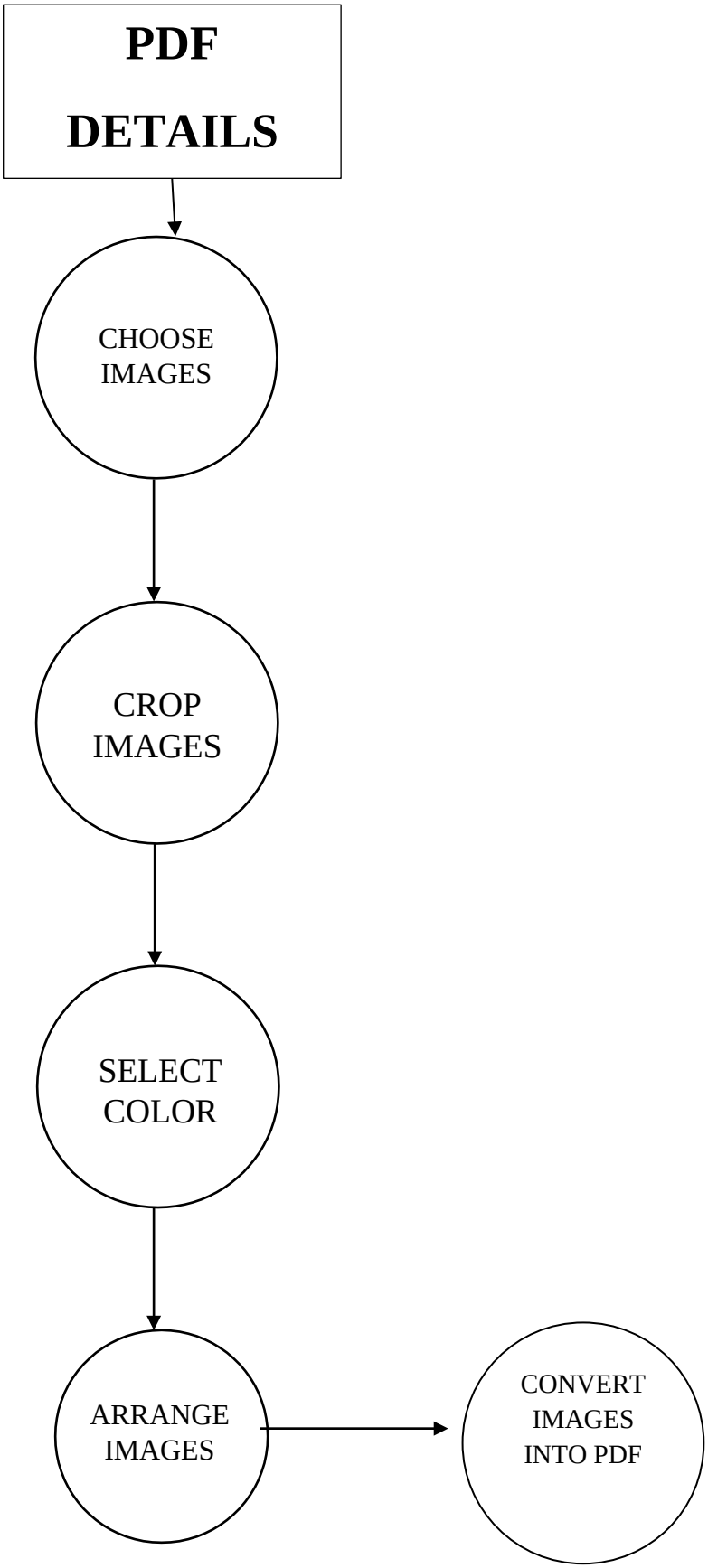

Fig 5.13 Level 1 DFD

- The application is developed by implementing the concept of inheritance, modularity which inturn reduces the complexity involved in maintaining it. The administrator should have a soundtechnical knowledge about maintaining the application and further enhancements will be undertaken by the developer.
- The application is portable which ensures its adaptability for use on different mobile deviceswith different Android versions and standards.
- The factors guarantee the application availability includes proper termination and correct inputdetails.
- Hence, we may conclude that the application system being developed helps a great deal inmodifying the PDF scanning.

CHAPTER 6: PROJECT RELEVANCE

6.1 CONTRIBUTION OF PROJECT TO THE SOCIETY

QueenScanner is an app for scanning documents, images, using the camera of the mobile device.

QueenScanner is tiny scanner app (only ~10 MB download) to make your phone become a amazing scanner (or called by other names as CamScanner, Tiny Scanner, PDF Scanner, etc...) toscan photos, PDFs, receipts, reports, or just about anything in everywhere. Queen Scanner uses the best algorithms to detect document edges, enhance texts, graphics, and be easy to manage or share in PDF, JPEG formats.

It is like incorporating a scanner with extensive functionality to the mobile to use it at any timeand place, file, edit, make lists and share the files obtained in PDF format.

Queen Scanner is like an easy-to-use and fast access toolbox

QueenScanner features:

- **Mobile Scanner**

  Turn your device to a scanner, scan everything as photos, receipts…

- **Scan Automatically**

  Detects document edges automatically.

- **Optimize scan quality**

  Enhanced texts and graphics look clear and sharp.

- **Speed and quality option**

  Scan with higher quality means that decoding speed is slower (3 options : slow, medium, fast).

- **PDF settings**

  Set page sizes for PDF (Letter, Legal, A3, A4, A5 and more).

- **Documents Organization**

  Name and organize documents in folders and easy to merge, move, copy, or delete.

- **Quick search**

  Easy to find documents with search tools, or sort by modified time, creation time, name.

- **Apply effect**

  4 modes for scan documents: Magic, Original, Grayscale, Black & White.
  5 levels for adjusting contrast, text bold.

- **User interface**

  2 modes to view: Grid View or List View.

- **Share PDF/JPEG files**

  Easy share doc in PDF, JPEG formats via social, email,...

This is how this app will help society. This app will be serving the purpose of making our workeasy and simple by making the pdfs and holding our work at one place.

6.2. ETHICS AND CONCLUSION

- After so much research work and findings we came to the conclusion that our project will be noharm to society and it follows all the norms and guidelines of computer software systems.
- Ethics is the study of value concepts such as 'good,' 'bad,' 'right,' 'wrong,' 'ought', applied to actions in relation to group norms and rules. Therefore, it deals with many issues fundamental topractical decision-making (Veatch, 1977). Computer software systems lie at the heart of modern decision making, including data/information storage and manipulation, data availability, and 'alternatives' formulation and selection.
- In fact, the very use of computer systems can often frame the types of questions that can be asked as well as their possible answers. This is particularly evident when we incorporate software systems into our knowledge management methods (Schmoldt and Rauscher, 1994), as they then play an essential role in institutional memory. The ubiquity of software systems in all aspects of public and private institutions means that the environment that they create needs to be critically

examined as they are developed and deployed. Two major ethical questions must be addressed with regard to software systems.

- Firstly, can these systems represent the different codes of ethics of the groups affected by software-mediated decisions? Secondly, what ethical considerations should guide the design anddevelopment of the software itself?
- So, for the first question, we didn't use any unethical codes which may harm any society or community. For the second part of the question, it will not be harming the society and will bebeneficial to all and help in all possible ways.

CHAPTER 7: CONCLUSION AND FUTURE PROJECTIONS

7.1 CONCLUSION

This project report gives the detailed study of the "QUEEN SCANNER " a PDF scanner app. The construction, working, implementation of the project is given throughout this report. App meets expectations appropriately to scan and convert images into PDF format. With the use of android studio, JAVA, XML, the application is properly created and is been functional. The features and options of the applications are properly shown in the report via various screenshots describing every feature of our app.

The results of our project work support our hypothesis that the Queen Scanner App would solve the problems created by older applications to scanner PDF. The final conclusion drawn from our project work is that the application is very efficient and it can be used in ever smart phones.

7.2 FUTURE PROJECTIONS

This application is very flexible to modify itself with the needs of users and customers in future.Yet the application is capable of enhancement.

Here we describe some features, which can be later introduced in the application for enhancing it:

- In Future, features for cloud upload of pdf along with system storage will be madeavailable.
- Currently, this app is for android only. In future, it will be available in both IOS andAndroid platforms.
- Currently, this app may have some bugs which can be later rectified once it gets installed in many phones and everyone starts using it
- Currently, this app is exclusively to convert images into PDF but in future more tools toresize PDF, compress PDF etc can also be added in our app.
- Currently, this app only generates PDF, whereas in future it can convert PDF into other file formats like docx, ppt etc.

REFERENCES

1. https://www.researchgate.net/publication/323015324_A_Survey_on_Java_Programming_Language_and_Methods_of_Improvisation [seen on 27.10.21 17:30]

2. Wang P, Bangert J and Kern C. 2021. "If It's Not Secure, It Should Not Compile: Preventing DOM-Based XSS in Large-Scale Web Development with API Hardening 2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)." 10.1109/ICSE43902.2021.00123. **978-1-6654-0296-5**, (1360-1372)
3. https://books.google.co.in/books?hl=en&lr=&id=9ukkDQAAQBAJ&oi=fnd&pg=PR1&dq=android+studio&ots=BWheQesdMI&sig=n0GgDRiAJKMe8FkfqjQPGr5Olh4&redir_esc=y#v=onepage&q=android%20studio&f=false .[seen on 12-10-21 12:00]
4. https://books.google.co.in/books?hl=en&lr=&id=VwFwDQAAQBAJ&oi=fnd&pg=PP1&dq=android+studio&ots=JHQ-3sVFeG&sig=TpeOEijlYHYYMUVh0WD4RX8gjqQ&redir_esc=y#v=onepage&q=android%20studio&f=false [seen on 18.09.21 16:10]
5. Jensen S.H., Møller A., Thiemann P. (2009) Type Analysis for JavaScript. In: Palsberg J., Su Z. (eds) Static Analysis. SAS 2009. Lecture Notes in Computer Science, vol 5673. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-03237-0_17
6. Zhang M and Meng W. "Detecting and understanding JavaScript global identifier conflicts on the web Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering, (38-49)
7. Åkerblom B and Wrigstad T. 2015. "Measuring polymorphism in programs." ACM SIGPLAN Notices, **51**:2, (114-128), Online publication date: 11-May-2016.
8. Kristensen E and Møller A. "Reasonably-most-general clients for JavaScript library analysis Proceedings of the 41st International Conference on Software Engineering, (83-93)"
9. The hypermodel benchmark. - Anderson, Berre, et al. – 1990
10. "Android SQLite Essentials Book: Develop Android applications with one of the most widely used database engines, SQLite." By Sunny Kumar Aditya Vikash Kumar Karn

CONTRIBUTION OF EACH MEMBER

1. SHUBHAM SRIVASTAVA:- Managed the back-end, managed the database of app.
2. UNNATI AGARWAL:- Managed the Graphic User Design (GUI) of the app, created and placed different buttons and options of the app.
3. SURYANSH SHARMA:- Managed the Front end of the app, designed the app on Android Studio and implemented various other modules.

ANNEXURE

```
package com.gpaddy.hungdh.camscanner; import android.Manifest;

import java.io.File;

import java.io.IOException;


import static com.joshuabutton.queenscanner.PresenterScanner.FOLDER_NAME;

public class MainActivity extends AppCompatActivity implements View.OnClickListener {
private CardView cvCamera, cvFromGallery, cvGallery, cvPDF;

private static final int REQUEST_STORAGE = 212;

private static final int REQUEST_CAMERA = 1;

private static final int REQUEST_CAMERA_PERMISSION = 2; private String
pathCamera = null;

private AdsTask; private LinearLayout llAds;


@Override

protected void onCreate(@Nullable Bundle savedInstanceState) {
super.onCreate(savedInstanceState); setContentView(R.layout.activity_main);

adsTask = new AdsTask(this);

MainApplication application = (MainApplication) getApplication();


if (ActivityCompat.checkSelfPermission(MainActivity.this,
Manifest.permission.WRITE_EXTERNAL_STORAGE) ==
PackageManager.PERMISSION_GRANTED) {

initView(); initAds();

//        initBannerAds();

} else {

ActivityCompat.requestPermissions(MainActivity.this, new
String[]{Manifest.permission.WRITE_EXTERNAL_STORAGE},
REQUEST_STORAGE);

}


}
```

```
private void initAds() {
//        MobileAds.initialize(this, getResources().getString(R.string.admod_app_id));
//
//        mAdView = findViewById(R.id.adView);

//        AdRequest = new AdRequest.Builder().build();
//        mAdView.loadAd(adRequest);
//
//        mInterstitialAd = new InterstitialAd(this);
//        mInterstitialAd.setAdUnitId(getResources().getString(R.string.ads_full_id));
//        mInterstitialAd.loadAd(new AdRequest.Builder().build());
//        mInterstitialAd.setAdListener(new AdListener() {
//        @Override
//        public void onAdClosed() {
//        // Load the next interstitial.
//        mInterstitialAd.loadAd(new AdRequest.Builder().build());
//        }
//        });
//        adsTask.loadInterstitialAdsFacebook();
}

@Override
protected void onRestart() { super.onRestart(); adsTask.showInterstitialAds();
}

private void initView() {
llAds = findViewById(R.id.ll_ads);
```

```
cvCamera = (CardView) findViewById(R.id.cvCamera); cvFromGallery = (CardView)
findViewById(R.id.cvFromGallery); cvGallery = (CardView)
findViewById(R.id.cvGallery);

cvPDF = (CardView) findViewById(R.id.cvPDF);



cvCamera.setOnClickListener(this); cvFromGallery.setOnClickListener(this);
cvGallery.setOnClickListener(this); cvPDF.setOnClickListener(this);



adsTask.loadBannerAdsFacebook(llAds);

}



@Override

public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions,
@NonNull int[] grantResults) {

super.onRequestPermissionsResult(requestCode, permissions, grantResults); if
(requestCode == REQUEST_STORAGE) {



if (ActivityCompat.checkSelfPermission(MainActivity.this,
Manifest.permission.WRITE_EXTERNAL_STORAGE) ==
PackageManager.PERMISSION_GRANTED) {

initView();

//        initBannerAds();

} else {

ActivityCompat.requestPermissions(MainActivity.this, new
String[]{Manifest.permission.WRITE_EXTERNAL_STORAGE},
REQUEST_STORAGE);

}

}

if (requestCode == REQUEST_CAMERA_PERMISSION) { if
(ActivityCompat.checkSelfPermission(MainActivity.this,

Manifest.permission.CAMERA) == PackageManager.PERMISSION_GRANTED) {
callCamera();

} else {
```

```
ActivityCompat.requestPermissions(MainActivity.this, new
String[]{Manifest.permission.CAMERA}, REQUEST_CAMERA_PERMISSION);
}
}

}

private void initBannerAds() {
//        final AdView mAdView = (AdView) findViewById(R.id.adView);
//
//        if (mAdView != null) {
//        mAdView.setAdListener(new AdListener() {
//        @Override
//        public void onAdFailedToLoad(int i) {
//        super.onAdFailedToLoad(i);
//        mAdView.setVisibility(View.GONE);
//        }
//
//        @Override
//        public void onAdLoaded() {
//        super.onAdLoaded();
//        mAdView.setVisibility(View.VISIBLE);
//        }
//        });
//
//        AdRequest = new AdRequest.Builder().build();
//        mAdView.loadAd(adRequest);
//        }

}
```

```
@Override
public void onClick(View view) { switch (view.getId()) {
case R.id.cvCamera:
/*Intent intent = new Intent(MainActivity.this, OpenNoteScannerActivity.class);
intent.setAction(Intent.ACTION_MAIN);
startActivity(intent);*/
if (ActivityCompat.checkSelfPermission(MainActivity.this,
Manifest.permission.CAMERA) == PackageManager.PERMISSION_GRANTED) {
callCamera();
} else {
ActivityCompat.requestPermissions(MainActivity.this, new
String[]{Manifest.permission.CAMERA}, REQUEST_CAMERA_PERMISSION);
}
break;
case R.id.cvFromGallery:
getSharedPreferences("BVH", MODE_PRIVATE).edit().putInt("type", 1).commit();
SimpleDocumentScannerActivity.startScanner(MainActivity.this, "", "");
//        Intent iG = new Intent(MainActivity.this, SimpleDocumentScannerActivity.class);
//        iG.putExtra(SimpleDocumentScannerActivity.KEY_DOCUMENT, "");
//        startActivity(iG);
break;
case R.id.cvGallery:
startActivity(new Intent(MainActivity.this, DocsActivity.class)); break;
case R.id.cvPDF:
startActivity(new Intent(MainActivity.this, MyPDFActivity.class)); break;
}
}

private void callCamera() { SharedPreferences mSharedPref =
PreferenceManager.getDefaultSharedPreferences(this);
```

```
String folderName = mSharedPref.getString("storage_folder", FOLDER_NAME); File
folder = new File(getExternalFilesDir(Environment.DIRECTORY_PICTURES)

+ "/" + folderName); if (!folder.exists()) {

folder.mkdirs();

}

pathCamera = folder.getAbsolutePath() + "/.TEMP_CAMERA.xxx";


getSharedPreferences("BVH", MODE_PRIVATE).edit().putString("path",
pathCamera).commit();

getSharedPreferences("BVH", MODE_PRIVATE).edit().putInt("type", 1).commit(); File f
= new File(pathCamera);

try {

f.createNewFile();

} catch (IOException e) { e.printStackTrace();

}









f);


Uri outputFileUri;

if (Build.VERSION.SDK_INT < 24)

outputFileUri = Uri.fromFile(f); else {

outputFileUri = FileProvider.getUriForFile(this, getPackageName() + ".provider",


}
```

```
Intent takePictureIntent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
takePictureIntent.putExtra(MediaStore.EXTRA_OUTPUT, outputFileUri);


if (takePictureIntent.resolveActivity(getPackageManager()) != null) {
startActivityForResult(takePictureIntent, REQUEST_CAMERA);

}

}


@Override

protected void onActivityResult(int requestCode, int resultCode, Intent data) {
super.onActivityResult(requestCode, resultCode, data);

int i = 0; i++;

if (requestCode == REQUEST_CAMERA && resultCode == RESULT_OK) { pathCamera
= getSharedPreferences("BVH", MODE_PRIVATE).getString("path",

pathCamera);

File f = new File(pathCamera); if (f.exists()) {

SimpleDocumentScannerActivity.startScanner(MainActivity.this, pathCamera,

"");

//        Intent iC = new Intent(MainActivity.this, SimpleDocumentScannerActivity.class);

//        iC.putExtra(SimpleDocumentScannerActivity.KEY_DOCUMENT, pathCamera);

//        startActivity(iC);

} else {

Toast.makeText(this, getString(R.string.file_not_found), Toast.LENGTH_SHORT).show();

}


}

}


@Override

public boolean onCreateOptionsMenu(Menu menu) { MenuInflater inflater =
getMenuInflater(); inflater.inflate(R.menu.menu_main, menu);
```

```
return true;

}


@Override

public boolean onOptionsItemSelected(MenuItem item) { if (item.getItemId() ==
R.id.action_settings) {

startActivity(new Intent(MainActivity.this, AccountActivity.class)); return true;

}

if (item.getItemId() == R.id.action_settings0) { callSettings();

return true;

}

if (item.getItemId() == R.id.action_rate_app) { rateApp();

return true;

}

if (item.getItemId() == R.id.action_share_app) { shareApp();

return true;

}

if (item.getItemId() == R.id.action_more_app) { moreApp();

return true;

}

return super.onOptionsItemSelected(item);

}


private void callSettings() {

startActivity(new Intent(MainActivity.this, MySettingsActivity.class));

}


private void rateApp() {

final String appPackageName = getPackageName(); // getPackageName() from Context or
Activity object
```

```
try {


startActivity(new Intent(Intent.ACTION_VIEW, Uri.parse("market://details?id=" +
appPackageName)));

} catch (android.content.ActivityNotFoundException anfe) { startActivity(new
Intent(Intent.ACTION_VIEW,

Uri.parse("https://play.google.com/store/apps/details?id=" + appPackageName)));

}

}


private void shareApp() {

final String appPackageName = getPackageName();

String myUrl = "https://play.google.com/store/apps/details?id=" + appPackageName;


Intent sendIntent = new Intent(); sendIntent.setAction(Intent.ACTION_SEND);
sendIntent.putExtra(Intent.EXTRA_TEXT, myUrl); sendIntent.setType("text/plain");
startActivity(Intent.createChooser(sendIntent, getString(R.string.share)));

}


private void moreApp() {

final String appPackageName = "Office+Utilities"; // getPackageName() from Context or
Activity object

try {

startActivity(new Intent(Intent.ACTION_VIEW, Uri.parse("market://developer?id=" +
appPackageName)));

} catch (ActivityNotFoundException anfe) { startActivity(new
Intent(Intent.ACTION_VIEW,

Uri.parse("https://play.google.com/store/apps/developer?id=" + appPackageName)));

}

}

}
```