

A

Project Report

On

----DOCA----

(Department of Computer Application)

[Submitted in partial fulfilment of the requirements for the degree of
Master of Computer Applications (MCA)]

Guided By

Name of the Guide (From Company/External)



**SRM INSTITUTE OF SCIENCE & TECHNOLOGY NCR CAMPUS
MODINAGAR**

(Deemed to be University U/S – 3 of UGC Act. 1956)(F. 9-9/98-U.3)

Year-2023

SUBMITTED TO:-

Mrs. Beena Sachan

Assistant Professor

MCA Department

SRMIST, NCR Campus, Modi Nagar

SUBMITTED BY:-

Student Name : Sourav Rana

Registration No. RA2132241030034

Dept. MCA

DECLARATION

I **Sourav Rana**, Registration no.**RA2132241030034**, MCA programme, SRM Institute of Science & Technology (Deemed to be University), NCR Campus, Modinagar-201204, Ghaziabad, batch of 2021-2023 do hereby solemnly declare that this Project work report is an original work of mine and this has not been submitted to any other institute/University towards any other degree/diploma.

(Student Name)

ACKNOWLEDGEMENT

I take this opportunity to express my deep sense of gratitude to all those who have contributed significantly by sharing their knowledge and experience in the completion of this project work.

My first word of gratitude is to Director Dr.S. Viswanathan, Dean Dr. D.K Sharma sir, Dean Campus Life & HOD-MCA Dr. Navin Ahlawat sir and Project Coordinator Dr. Rupak Sharma sir, for providing this kind of opportunity and guidance throughout the project.

My heartfelt thanks to My Project Guide Mrs. Beena Sachan without his continuation help the project would not have been materialized in the present form. I am thankful to him for providing me with necessary insights and valuable suggestions helped me at every step. We would like to take this opportunity to extend our warm thoughts to those who helped me in making this project a wonderful experience.

Last but not least; we would also like to thank our family and friends for their support and encouragement.

(Signature)

Student Name: Sourav Rana

Reg No.: RA2132241030034

Date: 26-04-2023

**SRM Institute of Science & Technology,
NCR Campus, Modinagar**

Date: 26-04-2023

TO WHOM SOEVER IT MAY CONCERN

This is to certify that Mr. Sourav Rana is a bonafide student of MCA-II Year of this institute for the session 2021-23 and he has prepared Project Report (MCA- IVth semester) titled DOCA for partial fulfilment of Master of Computer Application (MCA) affiliated to SRM Institute of Science & Technology, NCR Campus, Modinagar. He has worked to my supervision and her performance during the project has been satisfactory.

(Signature)

Project Guide Name:

Mrs. Beena Sachan

Assistant Professor

MCA Department

INDEX

- DECLARATION BY STUDENT
- ACKNOWLEDGEMENT
- CERTIFICATE
- PLAGRISM REPORT
- ABSTRACT.....
- What is problem facing now a days with your topic and technology?
- What improvement you are thinking?
- INTRODUCTION.....
- What is topic?
- What is history?
- Concern technology in market.....
- Difference with previous technology used with new one you are using.
- PROJECT MODULE DESCRIPTION.....
- MODULE WORKED ON IN THIS PROJECT.....
- SRS REPORT.....
- PROJECT DESIGN
- IMPLEMENTATION
- FUTURE OF THE WORK
- REFERENCES

ABSTRACT

Technology has changed our daily life routine as well as living style. So, student of school or colleges or university require application that supports smart phone to get all type of information related to examination, lecture notes, placement regarding question, notification, events, transportation etc. instead of calling system because almost all mobile users has smart phone now days. Each and every institute provides limited services to their users including students, parents, guardian and public. If provided services are more than ease of using is very difficult. That is why students interest towards using college or school or university is decreasing day by day. We designed an application to fulfil the requirement of students or parents, or employee based on present scenario of market using latest technology like java, android, GPS etc. to solve the students' problem.

- Study App is Android based application that helps students to interactively learn about courses allow experts to publish new study material on different subjects.
- Many Categories in this app like Update faculties, Photos, post anything in the notice.
- The real meaning of this app to know about your department like teachers and see the posts and you can study too in this app.

What Is the Definition of Mobile Learning?

In the past, students could only get an education in an actual classroom, but mobile learning allowed students to access classrooms via devices such as smartphones, tablets, and laptops. Students can access the learning process at any time and from anywhere by downloading assignments and uploading completed homework. Mobile learning is beginning to sweep the world, and education is growing at a rapid pace. While there are many advantages to mobile learning, there are also some disadvantages. But our application only have advantages.....

INTRODUCTION

There are several services required for the students in an application. Most of them are schedule of new session, time table of the class, schedule of examination, registration in new semester, examination form, new admission, study materials, placement materials common or company wise, list of company coming for placements, scope of technology in present and future, scope of courses, real time transportation status, real time placement status, real time performance status, real time attendance status, results, infrastructure like lab, workshop, gym, classroom, smart class, Wi-Fi, hostel, dispensary, bank, post office, library etc. But, mostly all school and colleges providing services of infrastructures, results, admission, placement but not up the mark, study materials

- Study App is Android based application that helps students to interactively learn about courses allow experts to publish new study material on different subjects.
- Many Categories in this app like Update faculties, Photos, post anything in the notice.
- The real meaning of this app to know about your department like teachers and see the posts and you can study too in this app.

Some Benefits of Educational Learning Apps

In the wake of the global pandemic, the education sector has seen a massive expansion of technology. With technology on its side, the entire industry is discovering new ways of doing things. It's not that technology has not been used in education before, but the use of educational applications has been limited. Using technology used to be a choice, but now it's a requirement. This has led to the adoption of educational software development via mobile applications, allowing companies, particularly in the education sector, to reach new heights. During compulsory distance learning, it was evident that there was a huge demand for technical tools and systems that allowed professors to communicate with their students, track their learning progress, and distribute their courses.

Personalized Learning And Mobile Educational Apps

When there are almost twenty to thirty students in the class, it will be very difficult for a teacher to understand where each student stands, in terms of complete knowledge. It is also sometimes very difficult for the students to keep with the pace in the class being conducted. The education apps come to the rescue in this regard. The apps can help the students get the personalized attention they need. It can help them additionally with up-to-date knowledge.

Enhancement Of Knowledge

The area of education is very dynamic and every day, some new information is being added. There are these education apps that provide updated information. For education, an app to teach the student about general knowledge. By using these educational apps, students can enhance their knowledge base and can increase their credibility. Even the teachers use these apps to provide updated knowledge to the students. This increases the quality of education which is being provided in the schools. The new way of learning has changed the way a student perceives information.

HISTORY

DOCA(Study app) help students to maintain a their notes or department related information. study App are targeted at student rather than study professionals. The most popular categories of downloaded Study apps are Notes, Video Lectures, and many other categories. It is the common understanding which every student must follow throughout the life for the knowledge of their department and the faculty and see the notices about the department. The real meaning of the DOCA involves the efficient and proper working of all the information about the department systems such as notices, Faculty, Gallery images, etc.

The study app makes it easy to learn about your study and start reaching your goals towards a intelligent and stress free life. Mobile devices have become common place in study settings, leading to rapid growth in the development of study software applications for these platforms. Thus, it is a challenge for developers to satisfy the different user needs which requires serious research in order to facilitate the effective usage of study apps Given that, in the near future, study apps must seriously address the need for quality content including the level of content detail and presentation, and users" special needs based on the user categories.

An educational institute is not open or available all the time. But unlike schools and institutes, the education mobile apps are available all the time. You do not have to worry about the schedules. You can use your desired mobile learning app anytime you want to study or learn.

Even if you have any queries or confusions related to the subject, you can get in touch with your tutors and teachers. You do not have to wait for the next day to arrive so that you can take expert help. A mobile education app is a solution for all.

CONCERN TECHNOLOGY IN MARKET

The study app makes it easy to learn about your notes and start reaching your goals. It provides all study data from smartphone and third-party apps. we already use, so that we can view all progress in one single place.

The idea behind study app is brilliant: it acts as one centralized hub for all your study data from various third-party study apps like read a lectures note, seeing the lecture videos and more. Instead of opening all these individual apps they feed you with each concept in one place i.e. our study app.

The study app in turn allows these third party study apps to access your study data from other apps so each individual app can get a better picture of your overall data. These apps can be useful because they make it easier to communicate problems with your teacher, often they allow you to record photos or documentation we can use at the time of the examination.

An XML file is an extensible markup language file, and it is used to structure data for storage and transport. In an XML file, there are both tags and text. The tags provide the structure to the data. The text in the file that you wish to store is surrounded by these tags, which adhere to specific syntax guidelines.

- XML uses human, not computer, language. XML is readable and understandable, even by novices, and no more difficult to code than HTML.
- XML is completely compatible with Java™ and 100% portable. Any application that can process XML can use your information, regardless of platform.
- XML is extendable.

Right-click the XML file you want to open, point to “Open With” on the context menu, and then click the “Notepad” option. Note: We're using Windows examples here, but the same holds true for other operating systems. Look for a good third-party text editor that is designed to support XML files.

In fact, proprietary data formats have become so complex that frequently one version of a complex application can't even read data from an earlier version of the same application. In XML, data and markup is stored as text that you yourself can configure. Just about every browser can open an XML file. In Chrome, just open a new tab and drag the XML file over. Alternatively, right click on the XML file and hover over "Open with" then click "Chrome". When you do, the file will open in a new tab XML files are encoded in plaintext, so you can open them in any text editor and be able to clearly read it. Right-click the XML file and select "Open With." This will display a list of programs to open the file in. Select "Notepad" (Windows) or "TextEdit" (Mac).

Java runs on the WORA (Writing Once, Run Anywhere). A Java code is compiled into an intermediate format (a.k.a. bytecode), which is then executed in the JVM (Java Virtual Machine). So, any system running a JVM can execute Java code. ... Object-oriented – Java is a purely objectoriented language. 'Java can be used to create complete applications that may run on a single computer or be distributed among servers and clients in a network. It can also be used to build a small application module or applet (a simply designed, small application) for use as part of a Web page. High-Level Languages and Computer Code To achieve its goal of "Write once, run anywhere," Java does both: a Java compiler converts Java code into machine-readable bytecode, then the Java Virtual Machine interprets the code for the computer it runs on. A Java developer is responsible for many duties throughout the development lifecycle of applications, from concept and design right through to testing. ... Conduct software analysis, programming, testing and 9 | P a g e debugging. Identifying production and non-production application issues. Transforming requirements into stipulations. While there are many developers with a host of experience working with the second most popular coding language, Java developer remains one of the most difficult jobs to fill.

Basic Java developer skills: -

- Java Server pages (JSP) and servlets.
- Web frameworks (e.g., Struts and Spring)
- Service-oriented architecture/web services (SOAP/REST)
- Web technologies like HTML, CSS, JavaScript, and jQuery.
- Markup languages like XML and JSON.
- Object-oriented programming (OOP) concepts and pattern

DIFFERENCE WITH PREVIOUS TECHNOLOGY USED WITH NEW ONE YOU ARE USING

In the last few decades, significant developments in technology greatly impacted every aspect of our lives. Living in the 21st century means we must be familiar with the ever-changing world of technology, and they certainly define how we deal with everyday situations. The field of modern education is no exception. The world's teachers and mentors must maintain a solid connection to the pace of technology to integrate the benefits of technological progression effectively.

Technology in education articles can tell how technologies impact on education system with time.

New technologies are slowly but surely completely changing education patterns. From classes that students can attend online to new forms of communication and media, new technologies facilitate the way professors interact with students and introduce new ways to teach that could completely transform the way classrooms will look in the future.

Nowadays, technology and education are interconnected. Technology in education, bringing new possibilities. The impact of technology on education has changed our way of taking education offline or online. Let's see how technology has changed education ?

One would be hard-pressed to find an educator who disagrees with such a standpoint. From laptops to tablets to **online courses** and beyond, technology has most definitely changed the face of education throughout time. Here are a few of the most significant ways the world of technology has collided with our education system.

Both teaching and learning have changed under the influence of the latest technological advancements in education. Chalkboards, printed textbooks, and libraries seem ancient in the era of tablets, smartphones, and e-books. Yet, here we have some of the most significant changes technology brought to education.

How Has Technology Changed Education?

Online classrooms

One of the most ground-breaking changes in the last recent years in education is online classrooms. **Online education** allows students to study from anywhere, anytime they want, and more and more reputable online institutions are making an appearance in the educational landscape.

Furthermore, the degrees dispensed through online schools have considerably changed over the years. Only a few degrees were attainable when universities first introduced online education. However, advanced courses such as Boston University's online criminal justice master's degree or advanced nursing studies are now available online. Just a few years ago, more and more institutions recognized online studies as legitimate these days.

It is now possible to complete a college degree without ever stepping into a classroom. In the past few years, online campuses have spawned with veracity. They are typically much more affordable than traditional universities, as students have no need to travel to class or pay for on-campus housing. In addition, modern education offers online colleges, which are much more adaptable to the lifestyle of a single and working parent.

Nowadays, online study materials are more interactive, user-friendly, and easy to read. These usually include videos, images, and links to relevant websites, which grab students' attention in the classroom. Moreover, since heavy textbooks are no longer obligatory, they can find everything they need on their tablets, laptops, or smartphones. There is also an environmental advantage is that schools are now using less paper to save trees.

Chalkboards are a thing of the past

Chalkboards have almost completely disappeared from the classroom. Markerboards are next! Nowadays, smartboards are replacing chalkboards. Smartboards are interactive and have a touch screen. In addition, teachers can host educational games with interactive interfaces and live results.

Most smart board models have a special pen tray that holds two specialized writing utensils. When students or teachers remove a pen from the tray, an electronic signal activates the pen. Now, they can use the stylus on the smartboard.

No more tiny notes floating around the classroom

No more tiny love notes are floating around the classrooms. Now the kids text each other. The average age of a child receiving a cell phone has recently reached 10-12 years. Only ten years prior, this was an outrageous prediction.

Teachers are faced with a whole new set of challenges when kids are equipped with tiny handheld computers. Distraction and lack of focus are at an entirely new level with that sort of distraction at a student's fingertips, and we all know that rules do not deter stubborn kids.

Note passing went from in-class to online

Passing notes in class became a part of history as soon as students got computer-literate. Nowadays, there are many websites for sharing HSC notes created by young people who want to help others learn. Students post their notes online, making them available to other students in their school and all over the country. These notes can be either purchased or downloaded free of charge. In this way, the quality of learning improves because students can collect different notes and take the most important parts out of them.

Web-based research instead of library-based

Searching for a particular piece of information in a regular library can be very frustrating since it is very time-consuming and exhausting. On the other hand, web-based research gives us vast amounts of data in no time – one click is enough to find what you need. In that way, the research process is sped up, thus leaving more time for learning itself.

Developments in technology definitely brought many changes in our everyday lives, including education. However, it is essential to know that the use of technology combined with traditional teaching methods gives the best results. In that way, students gain knowledge relevant for their future life, but they are also prepared to make the most of the latest technological improvements.

Communication is exponentially more effective

Communication is a million times faster and easier than it was ten years ago. Students, teachers, and parents have no excuse not to contact one another regularly. Email communications, mass text messages, automated phone calls, and much more create the ability never to lose touch.

The communication between teachers and students is no longer limited just to school hours. Instead, students can contact teachers simply by sending emails. This is actually a considerable

advantage of technological development because it saves time and allows students to stay in touch with their teachers and with each other. Also, teachers are able to share study materials online, using services like Dropbox, which enhances the process of studying even more.

eBooks

While many students report that they prefer traditional textbooks, with changing education technology, eBooks open a whole new world of possibilities for academics. While many people see eBooks as simply virtual forms of conventional books, eBooks are much more than that. For instance, they can insert multimedia into a PDF book to help explain certain concepts. Furthermore, it allows students to limit the number of books they have to carry and also allows them to access a wide variety of books in seconds.

Instead of hauling bulky books, students carry their tablets or portable laptops. In that way, the process of learning is advanced and simplified, because everything they need is stored in their gadgets. Also, online libraries are becoming popular because the notion of the book is changed and taken to the next level.

iPads and tablets create learning diversity

Technology has done some spectacular things in modern education, primarily through the development of the iPad and, similarly, tablets. As a result, students with all ranges of capabilities can mold their educational experience to their ability. In addition, developmentally disabled students have a wide range of new skills in the classroom through technology integration.

Improvement of simulations and models

Digital simulations allow teachers to simplify complicated concepts that are difficult to explain through traditional methods. For example, introducing concepts such as “how molecules react” or “the reactions when two particular chemicals are mixed” can be difficult outside a lab. However, virtual labs allow students to conduct experiments without the risks associated with lab work. It is also much more cost-efficient for universities.

Better assessment

A growing number of instructions have gone entirely digital when assessing their students’ work and how well they’ve integrated the principles they taught to them. In addition, it allows teachers to create tailor-made tests that students can take online to evaluate their knowledge base.

There is absolutely no telling what the face of education will look like in the next few years. Still, technology is sure to become more and more ubiquitous in classrooms and will probably change the lives of students and teachers for the better.

PROJECT MODULE DESCRIPTION

The Project divided into three modules students, subject faculty and administrator. The role of the module as follows: -

Students :- The Student select from various subjects available. The student takes study notes, projects, and test on a subject. The student can see the class schedules of subject, allow to interact with faculty related to their subjects.

Faculties :- Providing and build up best study material for the subjects enrolled and the admin can change the faculties and edit the info about the faculties like edit photo, name and post etc.

Administrator :- Publish materials submitted by faculty. Before publishing the material, it is compulsory to get review by admin. After going through its content either it gets approved or gets rejected. Modify all the activity by the admin in the user app like notice, images, e-book, faculties and after that the admin can delete anything in the activity.

Admin Module Features

- **Upload Notice**:- Edit any notice and upload notice regarding the department like holiday or event announcement and the test or practical announcement.
- **Upload Image**:- Upload any image regarding the category of the images like admin can upload the image in the user app images like Independence Day, convocation day, others day. All the users are easily seeing the images of the events.
- **Upload E-book** :- In this admin provide the study materials for the students to learn the notes for all the subjects.
- **Upload Faculties** :- In this admin can change or update or edit the faculties in the user app and all the user easily contact the subject wise teachers and admin provide the faculty name with their post and the image.
- **Delete Notice** :- In this the admin can delete the notice or images in the user app.

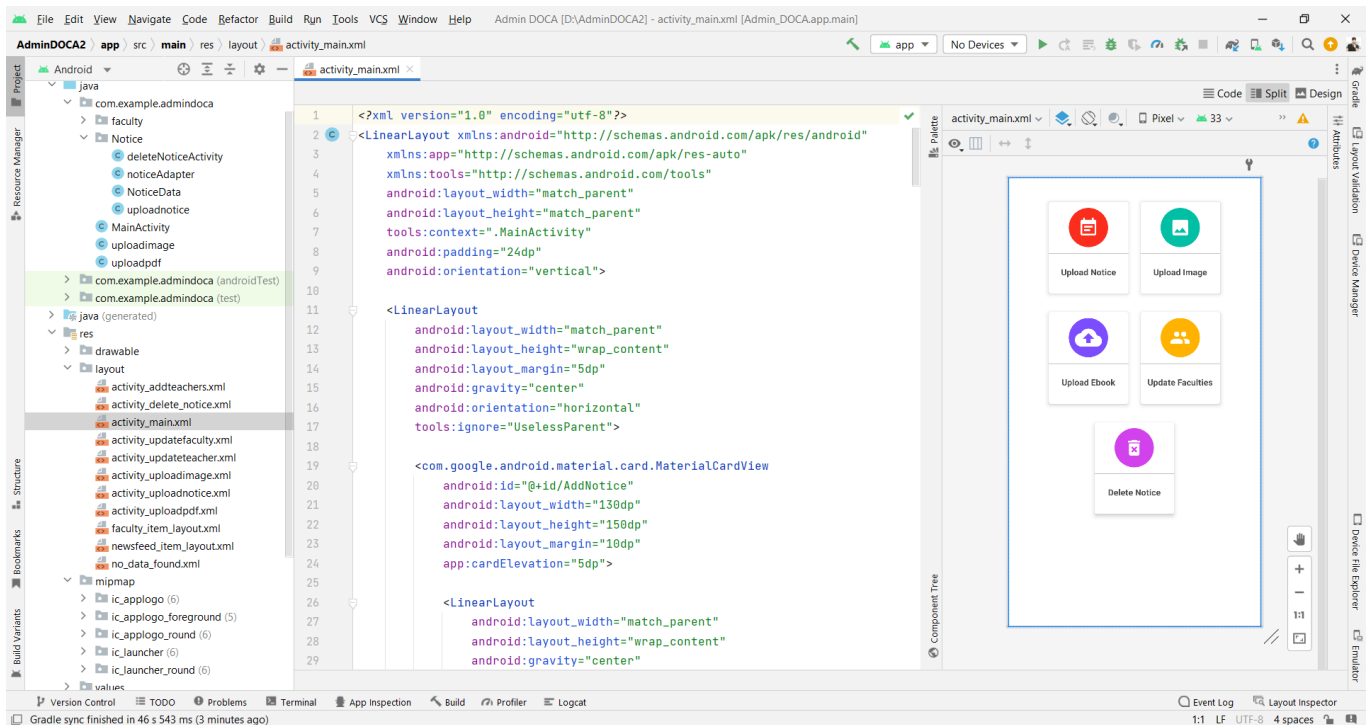
User App Features

- ▶ **Home Fragment** : In this we will show you our college information and photos and the contact details or map location.
- ▶ **Notice Fragment** : In this the admin update the notice after that users see the notice in the user app and check all the notifications about the course in the notice fragment.
- ▶ **Faculty Fragment** : In this the users see the all faculties with their profile photos and the info about the faculties.
- ▶ **Gallery Fragment** : In this the users show all the photos per parts like independence, convocation and other events photos are update in the gallery by the admin.
- ▶ **ERP Login** : In this context the users or students login their student college id with the id or passwords
- ▶ **Video lectures** : In these users take the video lectures from this button and the admin update all the subjects video lectures in this.
- ▶ **E-books** : In these users read all the pdf lectures from this button and the admin update all the subject's wise pdf lectures in this after that users are easily preparation for their exams.
- ▶ **Themes** : In these users change the theme of the application to her/him choice for the work . Two themes are available Dark & Light.
- ▶ **Website** : In this the users sign in and open the ERP page to check your progress in the class like test marks, attendance etc.
- ▶ **Share** : In this the users will share the app in your group and all the students to take the advantage of this app.
- ▶ **Rate us** : In this the users will rate our app in the play store in the star form.

MODULE WORKED ON IN THIS PROJECT

Admin Module Work

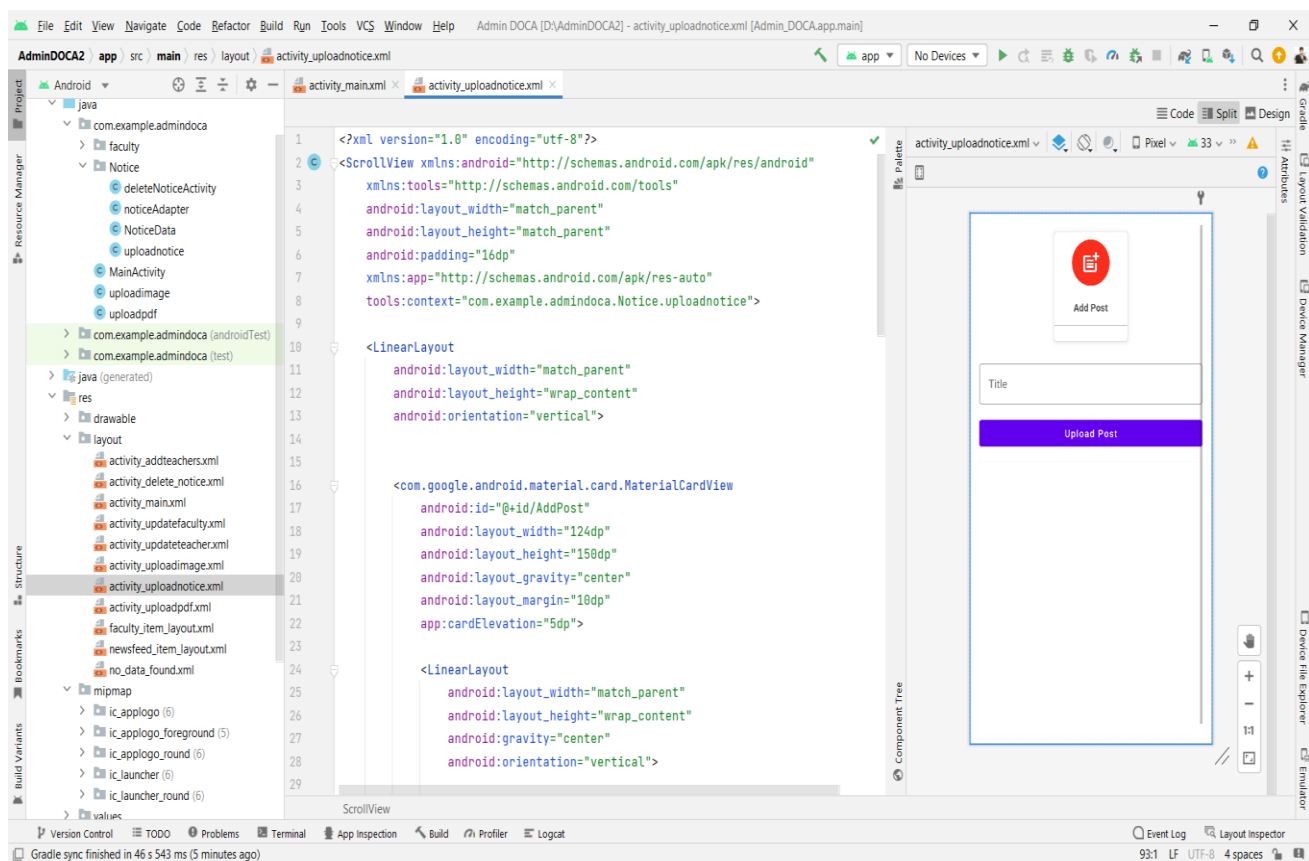
- This is a home screen of admin app who shows all the update functions like upload notice, image, pdf, faculty and deleting the notice.



Features Of Admin Module Work

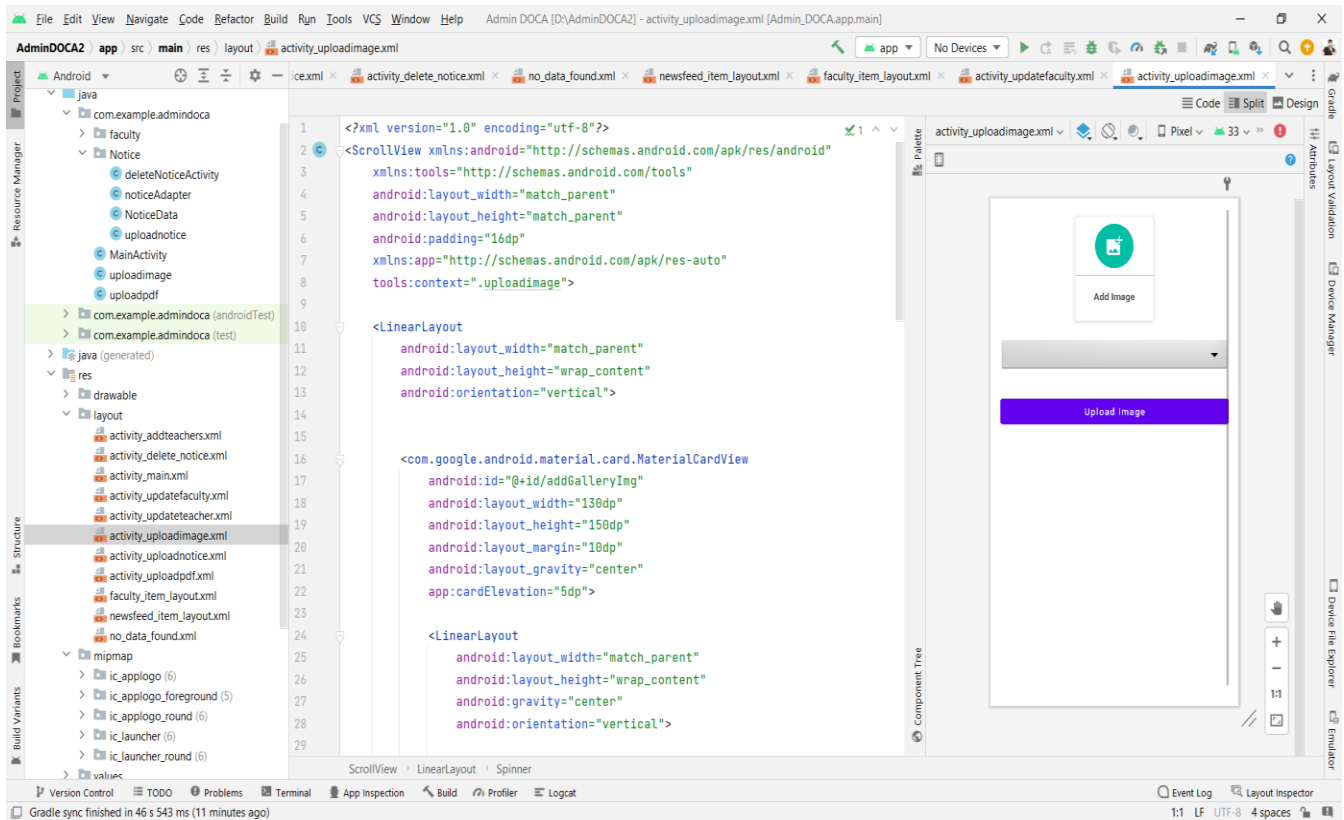
➤ Upload Notice :-

The Admin module has responsibility to edit any notice and upload new notice regarding the department like holiday or event announcement and the test examination or practical announcement. Any time upload notices to share with everyone regarding concert with time delay it can be removed from here.



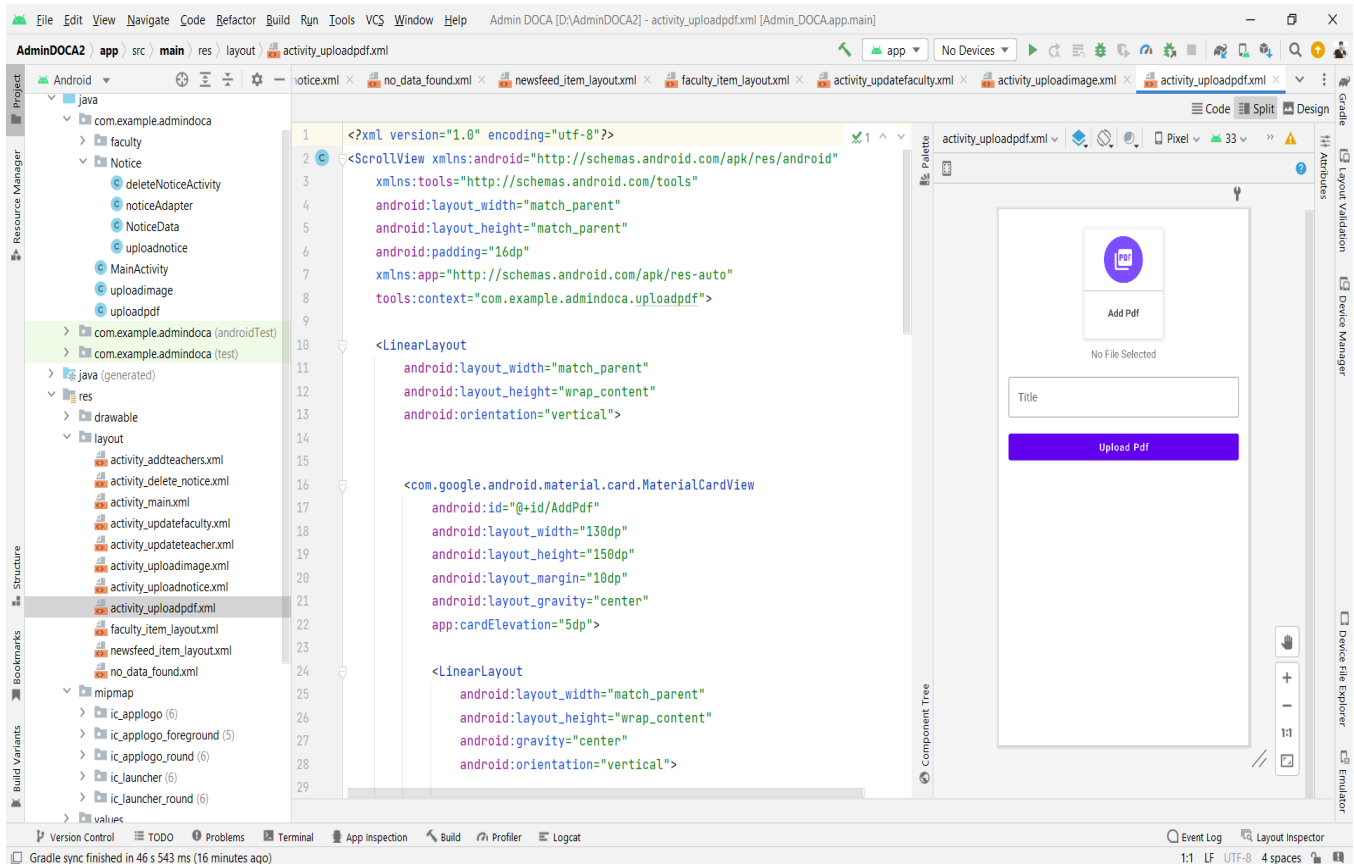
➤ Upload Image :-

Upload any image or post the ideas regarding the category of image like admin can upload the image in the user app and images like Independence Day, convocation day or other events photos can be upload. All the users are easily seeing the images of events.



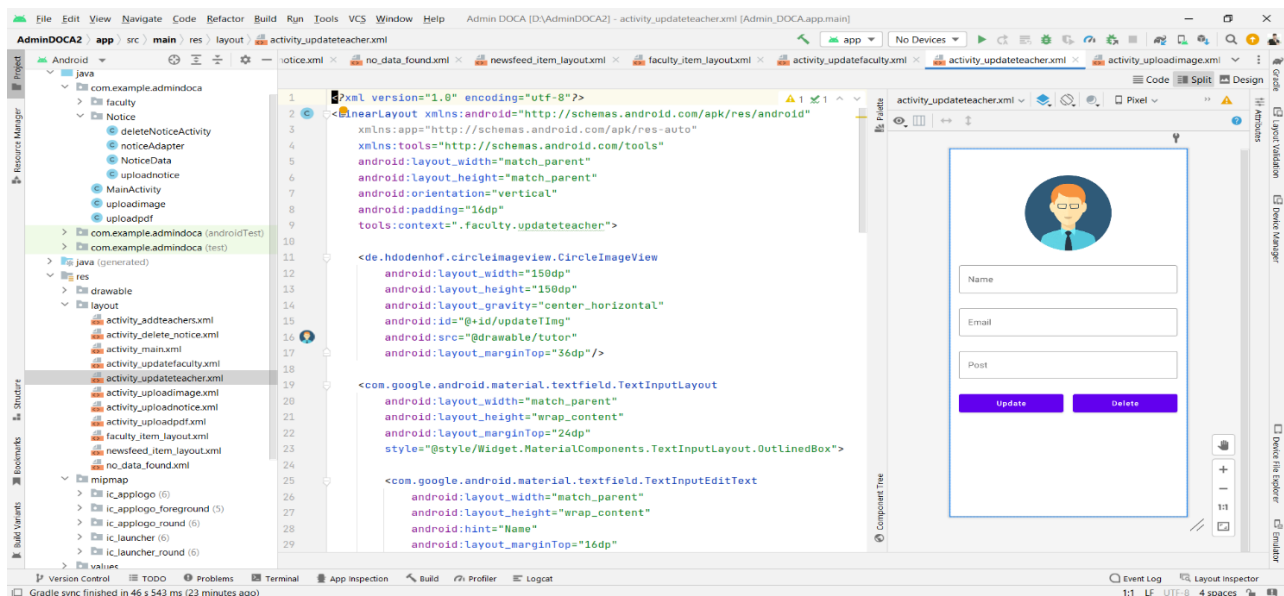
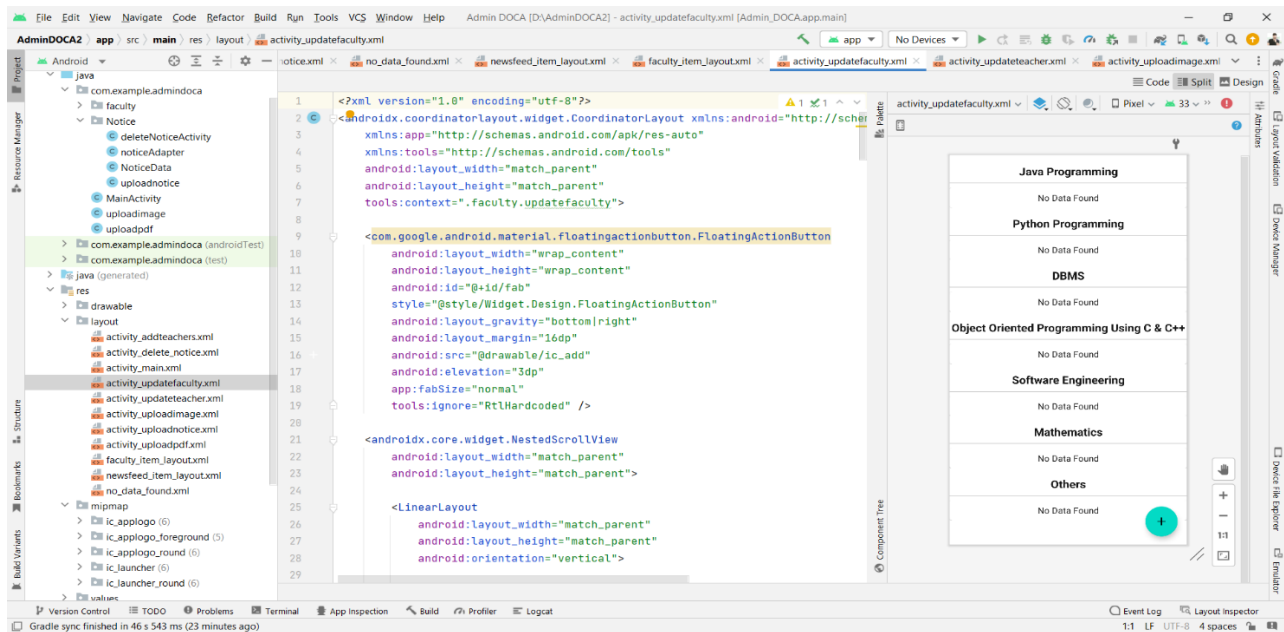
➤ Upload E-Books :-

In this the admin provide the study materials for the students to learn and prepare for the exams. User can view the pdf and download it. Subject wise eBooks are also located here according to the subjects.



Upload Faculties :-

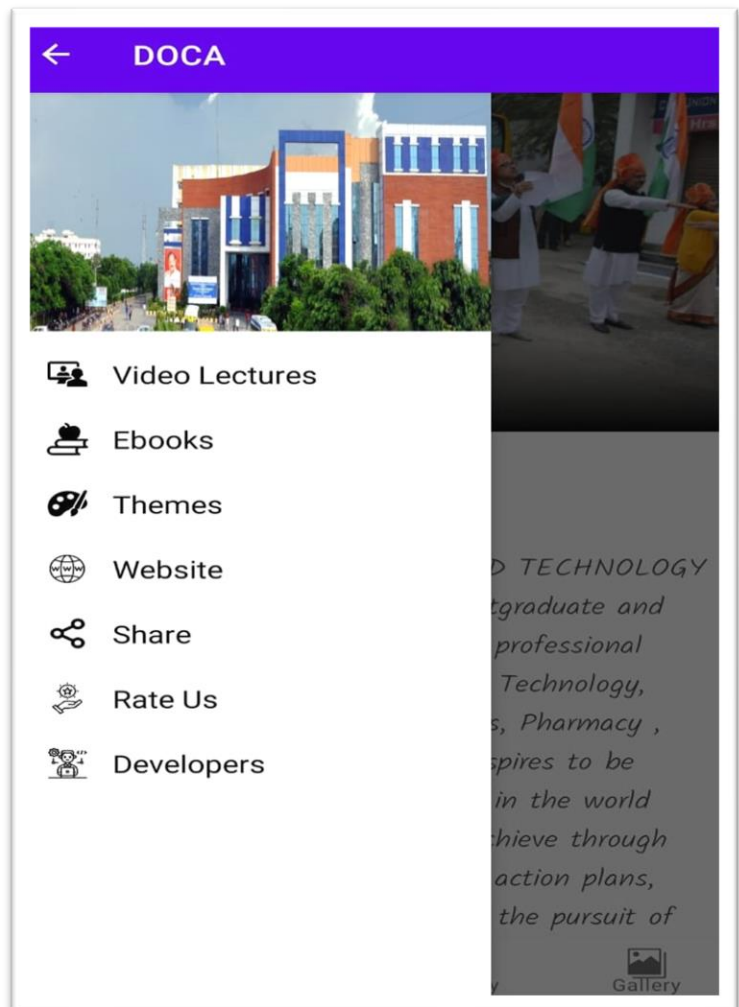
The Admin can change or update the faculties with their information like name, photo, post, mail id in user app and all the users easily contact the subject wise faculty and solve their problems.



In this the admin can update the faculty or delete the faculty means admin can change anything with the faculty.

User App Work

This is a home screen of admin app who shows all the buttons in navigation drawers and we are attach some applications screenshots. In this the side navigation have some functions and all are very important for the students to see the ERP portal and their profile easily and take the video lectures or the pdf files read and download .



Features Of User App Module Work

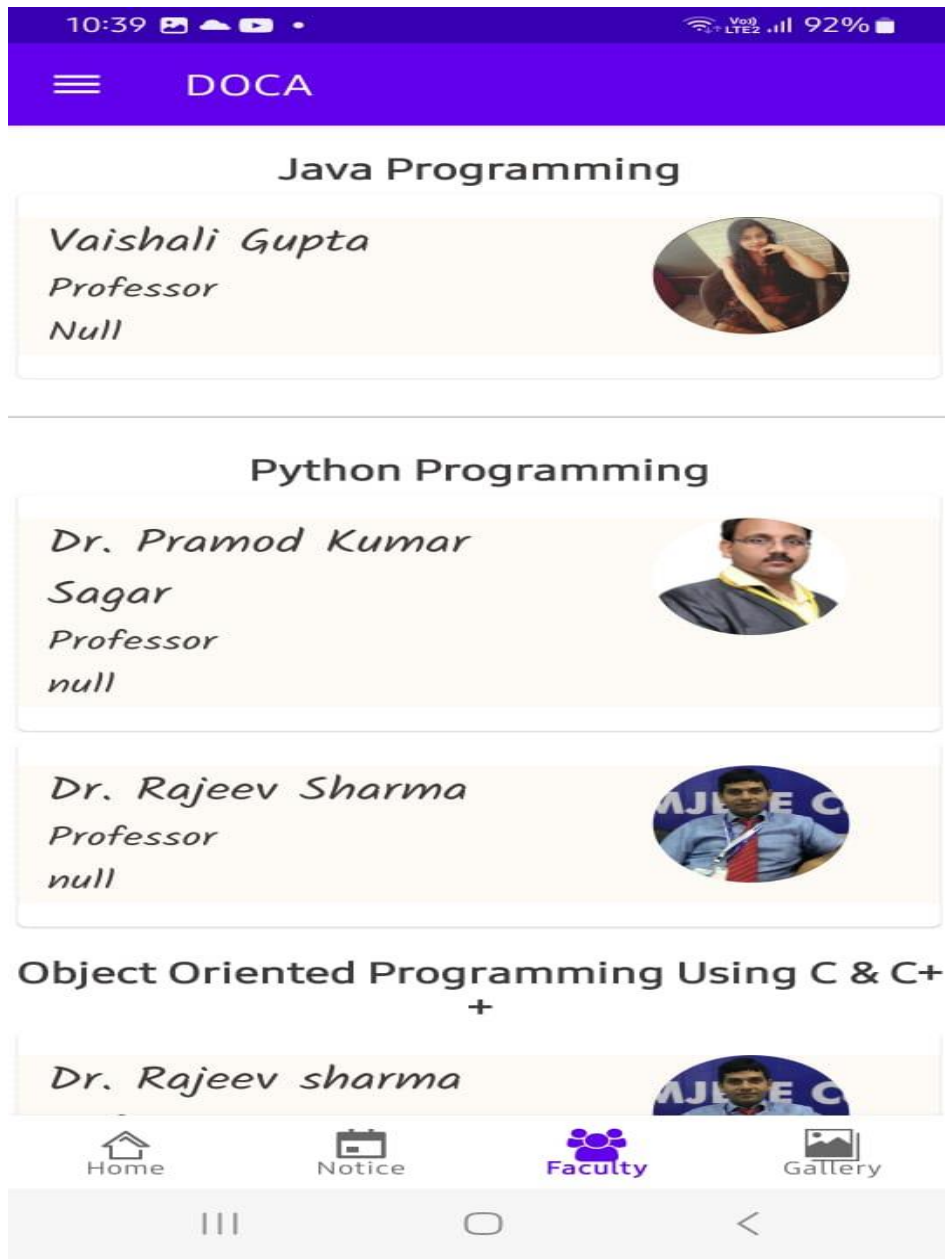
➤ Notice :-

In this fragment Users can see all the notices updates by the admin_like the holiday notification or examination etc.



➤ Faculty :-

In this fragment the users can see the faculty's name, post, photo and the email address for the contact and solving the student problems.



➤ **Gallery :-**

In this fragment the users can see the images of college events, festivals and the department photos.



SRS REPORT

Software & Hardware Requirements: -

Software Requirements:

Net beans :- NetBeans IDE is a free, open source, integrated development environment (IDE) that enables you to develop desktop, mobile and web applications. The IDE supports application development in various languages, including Java, HTML5, PHP and C++.

Top 5 Features of NetBeans 8 :-

- Tools for Java 8 Technologies. Especially code analysers. ...
- Tools for Java EE Developers. Here the focus is on out of. ...
- Tools for Maven. Maven is really the key feature of. ...
- Tools for JavaScript. Lots of love for AngularJS in. ...
- Tools for HTML5. Grunt and Karma out of the box!

Android studio :-

Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. A unified environment where you can develop for all Android devices. Apply Changes to push code and resource changes to your running app without restarting your app.

Top 10 Features of Android Studio that You can't Miss :-

1. Instant App Run

It is an advanced technology in which it cleverly understands the transmutations done in the applications and deliver it instantly without taking time to rebuild the apk and installations make. So, quick that you can see the changes in app immediately. This is done by launching the URL on Android application installing the native libraries with Android Instant apps. Instant App sometimes is known as " New Module Wizard".

2. Visual Layout Editor

Layout editor helps to build the layout quickly by adding different attributes either by hard-code or drag and drop. The preview of the codes can be seen easily on the visual editor screen and changes can be made accordingly by resizing it dynamically. This will make testing the application process more facile and more exhaustive.

3. Fast Emulator

Android has a great feature of Emulator which is exactly like the android phones to test how the application looks like in physical devices. It gives real-time experience to the Android applications. It allows you to test your applications faster and on different-different configuration devices like tablet, android phone etc. It helps you to make your application development life cycle shorter and more efficient.

4. Intelligence Code Editor

Android Studio provides you with the intelligent and quick code editor. This will help you and guide you with the accurate code. It helps you to complete code in advance and analyse your code in advance before building. Android studio has the special feature of the development of code by the drop-down list with suggesting the code you can integrate.

5. Addition of New Activity as a Code Template

Yes, Android also has the feature of templates built-in. If you know to build that accordingly that makes your task easier. it has both pros and cons, you don't find every template in Android Studio. It's an additional feature which helps the developer to build an application efficiently and effectively which provide effective solutions.

6. Help to Build Up App for All Devices

Android studio builds applications for every screen size, for wear and gear devices etc. It also can stimulate the various type of features which a hardware has like GPS location tracker, multi-touch.

7. Help to Connect with Firebase

Android Studio helps to give real-time experience with IOT based project development with dynamic upgrades in the application. Firebase connectivity help to create direct updates and provide databases connectivity. To build high-quality applications we must use Firebase connectivity it helps to build the scalable infrastructure for building the application. You can create chat applications by using firebase connectivity it helps you to do happy chat experience.

8. Support KOTLIN

Kotlin the official language can be considered for Android. It is a language without having any new restrictions and has various advantages in it. The great feature of Kotlin is it run without any disturbances in older android versions as well that means no issues of specific android versions. It runs fast and equivalent to Java. Java developers can easily hands-on Kotlin with no-issues because it based on automated Java only.

9. Colour Previews

Android studio helps to see the code XML part in a preview to know that how perfectly we are designing the application according to the need before launching the application. It provides powerful functionality and enhanced features of drag and drops or resizes the application. It contains drag and drop features but not support for every function, that's why be careful while doing that.

10. Maven Repository

In Android Studio, Maven integration of its repository can be done, within SDK manager support libraries of IDE is used. It's a kind of a repository which is a directory in which various jar files like project jars, Plugin are stored.

App inventor :- App Inventor lets you develop applications for Android phones using a web browser and either a connected phone or emulator. The App Inventor servers store your work and help you keep track of your projects. You can build just about any app imaginable with App Inventor: games, informational apps with user-generated data, personal convenience apps, apps to help people communicate, apps that make use of the phone's sensors, and even apps that talk to web services like Twitter.

Languages in App Inventor: -

Java MIT App Inventor is a web application integrated development environment originally provided by Google, and now maintained by the Massachusetts Institute of Technology (MIT).

App Inventor for Android.

MIT App Inventor	
Written in	Java, Kawa, Scheme
Operating system	Android
Available in	19 languages

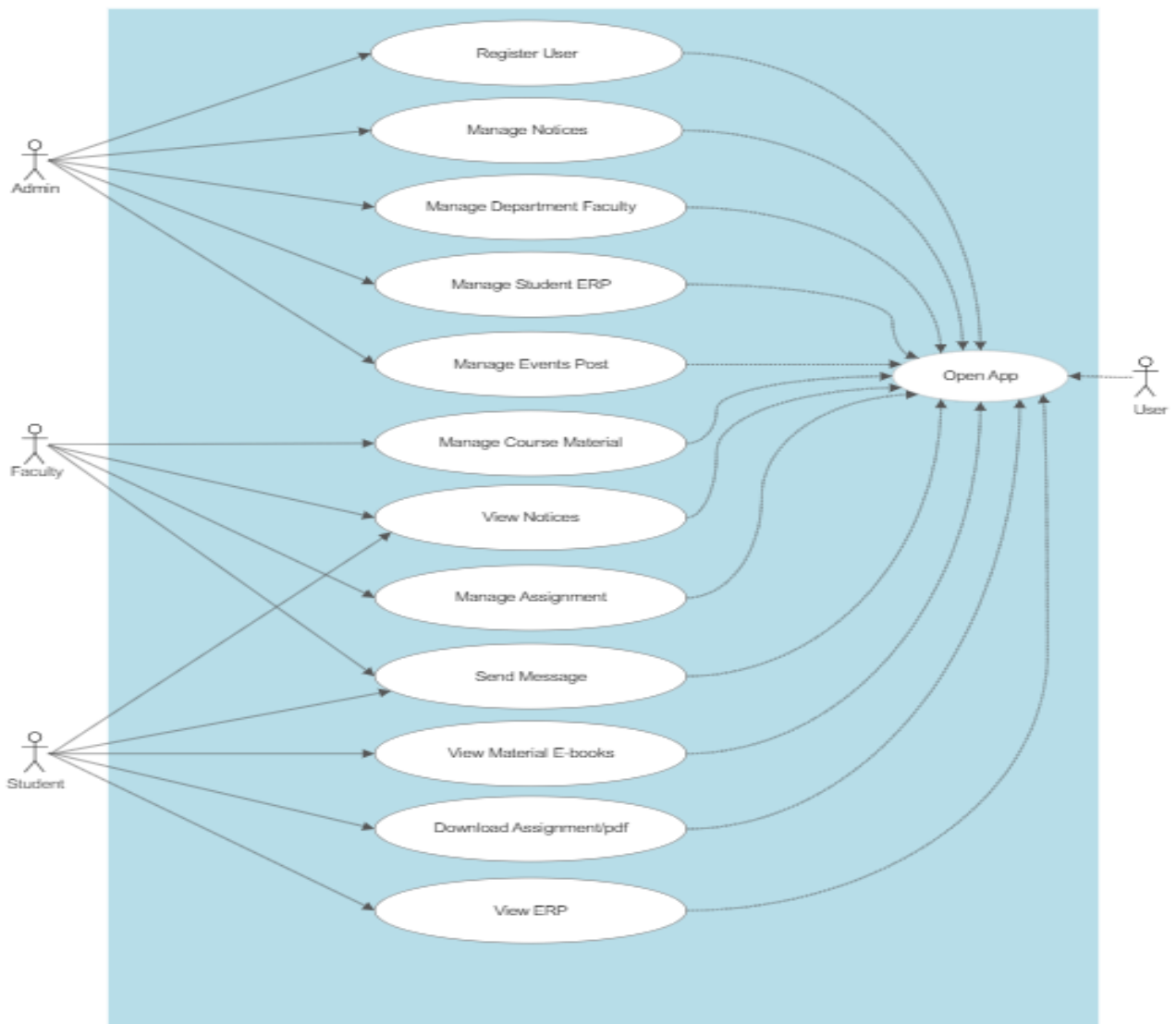
HARDWARE REQUIREMENTS:

- Operating system required- Windows 7 and above.
- 4GB RAM and above

Project Design

- Use Case Diagram

Use-case diagrams describe the high-level functions and scope of a system. These diagrams also identify the interactions between the system and its actors. The use cases and actors in use-case diagrams describe what the system does and how the actors use it, but not how the system operates internally.



IMPLIMENTATION (SOURCE CODE)

➤ **ADMIN APP Source Code**

Home Screen. Java

```
package com.example.admindoca;

import androidx.appcompat.app.AppCompatActivity;
import androidx.cardview.widget.CardView;

import android.annotation.SuppressLint;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;

import com.example.admindoca.Notice.deleteNoticeActivity;
import com.example.admindoca.Notice.uploadnotice;
import com.example.admindoca.faculty.updatefaculty;

public class MainActivity extends AppCompatActivity implements View.OnClickListener {

    CardView UploadNotice, AddGal, AddPdf, Faculty, DeleteNotice;

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

        UploadNotice = findViewById(R.id.AddNotice);
        AddGal = findViewById(R.id.AddGal);
        AddPdf = findViewById(R.id.AddPdf);
        Faculty = findViewById(R.id.Faculty);
        DeleteNotice = findViewById(R.id.DeleteNotice);

        UploadNotice.setOnClickListener(this);
        AddGal.setOnClickListener(this);
        AddPdf.setOnClickListener(this);
        Faculty.setOnClickListener(this);
        DeleteNotice.setOnClickListener(this);
    }

    @SuppressLint("NonConstantResourceId")
    @Override
    public void onClick(View view) {
        Intent intent;
        switch (view.getId()) {
            case R.id.AddNotice:
                intent = new Intent(MainActivity.this, uploadnotice.class);
```

```

        startActivity(intent);
        break;

    case R.id.AddGal:
        intent = new Intent(MainActivity.this, uploadimage.class);
        startActivity(intent);
        break;

    case R.id.AddPdf:
        intent = new Intent(MainActivity.this, uploadpdf.class);
        startActivity(intent);
        break;

    case R.id.Faculty:
        intent = new Intent(MainActivity.this, updatefaculty.class);
        startActivity(intent);
        break;
    case R.id.DeleteNotice:
        intent = new Intent(MainActivity.this, deleteNoticeActivity.class);
        startActivity(intent);
        break;
    }
}
}

```

Home Screen. XML

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:padding="24dp"
    android:orientation="vertical">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="5dp"
        android:gravity="center"
        android:orientation="horizontal"
        tools:ignore="UselessParent">

        <com.google.android.material.card.MaterialCardView
            android:id="@+id/AddNotice"
            android:layout_width="130dp"
            android:layout_height="150dp"
            android:layout_margin="10dp"
            app:cardElevation="5dp">

            <LinearLayout
                android:layout_width="match_parent"

```



```

        android:layout_height="wrap_content"
        android:gravity="center"
        android:orientation="vertical">

        <ImageView
            android:layout_width="64dp"
            android:layout_height="64dp"
            android:background="@drawable/circle_red"
            android:padding="15dp"
            android:layout_marginTop="10dp"
            android:src="@drawable/ic_notice"
            tools:ignore="ContentDescription" />

        <View
            android:layout_width="match_parent"
            android:layout_height="1dp"
            android:background="@color/light_gray"
            android:layout_marginTop="10dp"
            />
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Upload Notice"
            android:textStyle="bold"
            android:padding="5dp"
            android:layout_marginTop="15dp"
            android:textColor="#4E4E4E"/>

    </LinearLayout>

</com.google.android.material.card.MaterialCardView>

<com.google.android.material.card.MaterialCardView
    android:id="@+id/AddGal"
    android:layout_width="130dp"
    android:layout_height="150dp"
    android:layout_margin="10dp"
    app:cardElevation="5dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:orientation="vertical">

        <ImageView
            android:layout_width="64dp"
            android:layout_height="64dp"
            android:background="@drawable/circle_green"
            android:padding="15dp"
            android:layout_marginTop="10dp"
            android:src="@drawable/ic_gallery"
            tools:ignore="ContentDescription" />

        <View
            android:layout_width="match_parent"

```

```

        android:layout_height="1dp"
        android:background="@color/light_gray"
        android:layout_marginTop="10dp"
    />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Upload Image"
        android:textStyle="bold"
        android:padding="5dp"
        android:layout_marginTop="15dp"
        android:textColor="#4E4E4E"
        tools:ignore="HardcodedText" />

</LinearLayout>

</com.google.android.material.card.MaterialCardView>

</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="5dp"
    android:gravity="center"
    android:orientation="horizontal"
    tools:ignore="UselessParent">

    <com.google.android.material.card.MaterialCardView
        android:id="@+id/AddPdf"
        android:layout_width="130dp"
        android:layout_height="150dp"
        android:layout_margin="10dp"
        app:cardElevation="5dp">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:orientation="vertical">

            <ImageView
                android:layout_width="64dp"
                android:layout_height="64dp"
                android:background="@drawable/circle_purple"
                android:padding="10dp"
                android:layout_marginTop="10dp"
                android:src="@drawable/ic_upload"
                tools:ignore="ContentDescription" />

            <View
                android:layout_width="match_parent"
                android:layout_height="1dp"
                android:background="@color/light_gray"
                android:layout_marginTop="10dp"
            />

```

```

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Upload Ebook"
            android:textStyle="bold"
            android:padding="5dp"
            android:layout_marginTop="15dp"
            android:textColor="#4E4E4E"
            tools:ignore="HardcodedText" />

    </LinearLayout>

</com.google.android.material.card.MaterialCardView>

<com.google.android.material.card.MaterialCardView
    android:id="@+id/Faculty"
    android:layout_width="130dp"
    android:layout_height="150dp"
    android:layout_margin="10dp"
    app:cardElevation="5dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:orientation="vertical">

        <ImageView
            android:layout_width="64dp"
            android:layout_height="64dp"
            android:background="@drawable/circle_yellow"
            android:padding="15dp"
            android:layout_marginTop="10dp"
            android:src="@drawable/ic_group"
            tools:ignore="ContentDescription" />

        <View
            android:layout_width="match_parent"
            android:layout_height="1dp"
            android:background="@color/light_gray"
            android:layout_marginTop="10dp"
            />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Update Faculties"
            android:textStyle="bold"
            android:padding="5dp"
            android:layout_marginTop="15dp"
            android:textColor="#4E4E4E"
            tools:ignore="HardcodedText" />

    </LinearLayout>

```

```

        </com.google.android.material.card.MaterialCardView>

</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="5dp"
    android:gravity="center"
    android:orientation="horizontal"
    tools:ignore="UselessParent">

    <com.google.android.material.card.MaterialCardView
        android:id="@+id/DeleteNotice"
        android:layout_width="130dp"
        android:layout_height="150dp"
        android:layout_margin="10dp"
        app:cardElevation="5dp">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:orientation="vertical">

            <ImageView
                android:layout_width="64dp"
                android:layout_height="64dp"
                android:background="@drawable/circle_pink"
                android:padding="15dp"
                android:layout_marginTop="10dp"
                android:src="@drawable/ic_delete"
                tools:ignore="ContentDescription" />

            <View
                android:layout_width="match_parent"
                android:layout_height="1dp"
                android:background="@color/light_gray"
                android:layout_marginTop="10dp"
                />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Delete Notice"
                android:textStyle="bold"
                android:padding="5dp"
                android:layout_marginTop="15dp"
                android:textColor="#4E4E4E"
                tools:ignore="HardcodedText" />

        </LinearLayout>
    </com.google.android.material.card.MaterialCardView>
</LinearLayout>
</LinearLayout>

```

Upload Notice. Java

```
package com.example.admindoca.Notice;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.cardview.widget.CardView;

import android.app.ProgressDialog;
import android.content.Intent;
import android.graphics.Bitmap;
import android.net.Uri;
import android.os.Bundle;
import android.provider.MediaStore;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Toast;

import com.example.admindoca.R;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.google.firebase.storage.UploadTask;

import java.io.ByteArrayOutputStream;
import java.io.IOException;
import java.text.SimpleDateFormat;
import java.util.Calendar;

public class uploadnotice extends AppCompatActivity {
    private CardView addPost;
    private ImageView noticeImageview;
    private EditText NoticeTitle;
    private Button UploadPostBtn;

    private final int REQ = 1;
    private Bitmap bitmap;
    private DatabaseReference reference, dbRef;
    private StorageReference storageReference;
    String downloadurl = "";
    private ProgressDialog pd;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_uploadnotice);
```

```

reference = FirebaseDatabase.getInstance().getReference();
storageReference = FirebaseStorage.getInstance().getReference();

pd = new ProgressDialog(this);

addPost = findViewById(R.id.AddPost);
noticeImageview = findViewById(R.id.PostView);
NoticeTitle = findViewById(R.id.NoticeTitle);
UploadPostBtn = findViewById(R.id.UploadPostBtn);

addPost.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View view) {
        openGallery();
    }
});

UploadPostBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (NoticeTitle.getText().toString().isEmpty()){
            NoticeTitle.setError("Empty");
            NoticeTitle.requestFocus();
        }else if(bitmap == null){
            uploadData();
        }else {
            uploadImage();
        }
    }
});
}

private void uploadImage() {
    pd.setMessage("Uploading...");
    pd.show();
    ByteArrayOutputStream baos = new ByteArrayOutputStream();
    bitmap.compress(Bitmap.CompressFormat.JPEG, 50, baos);
    byte[] finalimg = baos.toByteArray();
    final StorageReference filepath;
    filepath = storageReference.child("Notice").child(finalimg+"jpg");
    final UploadTask uploadTask = filepath.putBytes(finalimg);
    uploadTask.addOnCompleteListener(uploadnotice.this, new
    OnCompleteListener<UploadTask.TaskSnapshot>() {
        @Override
        public void onComplete(@NonNull Task<UploadTask.TaskSnapshot> task) {
            if (task.isSuccessful()){
                uploadTask.addOnSuccessListener(new
    OnSuccessListener<UploadTask.TaskSnapshot>() {
                    @Override
                    public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {
                        filepath.getDownloadUrl().addOnSuccessListener(new
    OnSuccessListener<Uri>() {
                            @Override
                            public void onSuccess(Uri uri) {

```

```

        downloadurl = String.valueOf(uri);
        uploadData();
    }
    });
}
});
}
}else {
    pd.dismiss();
    Toast.makeText(uploadnotice.this, "Something went wrong",
Toast.LENGTH_SHORT).show();
}

}

});

}

private void uploadData() {
    dbRef = reference.child("Notice");
    final String uniquekey = dbRef.push().getKey();

    String title = NoticeTitle.getText().toString();

    Calendar calendar = Calendar.getInstance();
    SimpleDateFormat currentDate = new SimpleDateFormat("dd-MM-yy");
    String date = currentDate.format(calendar.getTime());

    Calendar calfortime = Calendar.getInstance();
    SimpleDateFormat currentTime = new SimpleDateFormat("hh:mm a");
    String time = currentTime.format(calfortime.getTime());

    NoticeData noticeData = new NoticeData(title,downloadurl,date,time,uniquekey);

    dbRef.child(uniquekey).setValue(noticeData).addOnSuccessListener(new
OnSuccessListener<Void>() {
        @Override
        public void onSuccess(Void unused) {
            pd.dismiss();
            Toast.makeText(uploadnotice.this, "Notice Uploaded",
Toast.LENGTH_SHORT).show();
        }
    }).addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {
            pd.dismiss();
            Toast.makeText(uploadnotice.this, "Something went wrong",
Toast.LENGTH_SHORT).show();
        }
    });
}
}

```

```

    }

    private void openGallery() {
        Intent pickImage = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);

        startActivityForResult(pickImage, REQ);
    }

    @Override
    protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent
data) {
        super.onActivityResult(requestCode, resultCode, data);
        if (requestCode == REQ && resultCode == RESULT_OK) {

            Uri uri = data.getData();
            try {
                bitmap = MediaStore.Images.Media.getBitmap(getContentResolver(), uri);
            } catch (IOException e) {
                e.printStackTrace();
            }
            noticeImageview.setImageBitmap(bitmap);
        }
    }
}

```

➤ Noticedata

```

package com.example.admindoca.Notice;

public class NoticeData {
    private String title;
    private String image;
    private String date;
    private String time;
    private String key;

    public NoticeData() {

    }

    public NoticeData(String title, String image, String date, String time, String key) {
        this.title = title;
        this.image = image;
        this.date = date;
        this.time = time;
        this.key = key;
    }
}

```



```

public String getTitle() {
    return title;
}

public void setTitle(String title) {
    this.title = title;
}

public String getImage() {
    return image;
}

public void setImage(String image) {
    this.image = image;
}

public String getDate() {
    return date;
}

public void setDate(String date) {
    this.date = date;
}

public String getTime() {
    return time;
}

public void setTime(String time) {
    this.time = time;
}

public String getKey() {
    return key;
}

public void setKey(String key) {
    this.key = key;
}
}

```

➤ Notice adapter

```

package com.example.admindoca.Notice;

import android.annotation.SuppressLint;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import com.example.admindoca.R;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.database.DatabaseReference;

```

```

import com.google.firebase.database.FirebaseDatabase;
import com.squareup.picasso.Picasso;

import java.util.ArrayList;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

public class noticeAdapter extends RecyclerView.Adapter<noticeAdapter.noticeViewHolder>
{
    private Context context;
    private ArrayList<NoticeData>list;

    public noticeAdapter(Context context,ArrayList<NoticeData>list){
        this.context = context;
        this.list = list;
    }

    @NonNull
    @Override
    public noticeViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType)
    {
        View view =
        LayoutInflater.from(context).inflate(R.layout.newsfeed_item_layout,parent,false);

        return new noticeViewHolder(view);
    }

    @SuppressWarnings("SuspiciousIndentation")
    @Override
    public void onBindViewHolder(@NonNull noticeViewHolder holder,
    @SuppressWarnings("RecyclerView") int position) {

        NoticeData currentItem = list.get(position);

        holder.deleteNoticeTitle.setText(currentItem.getTitle());

        try {
            if (currentItem.getImage() != null)
                Picasso.get().load(currentItem.getImage()).into(holder.deleteNoticeImageView);
        } catch (Exception e) {
            e.printStackTrace();
        }
        holder.deleteNoticeBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

                AlertDialog.Builder builder = new AlertDialog.Builder(context);
                builder.setMessage("Are you sure want to delete this notice ?");
                builder.setCancelable(true);
                builder.setPositiveButton(
                    "Ok",
                    new DialogInterface.OnClickListener() {
                        @Override

```

```

        public void onClick(DialogInterface dialogInterface, int i) {
            DatabaseReference reference =
                FirebaseDatabase.getInstance().getReference().child("Notice");
            reference.child(currentItem.getKey()).removeValue()
                .addOnCompleteListener(new
                    OnCompleteListener<Void>() {
                        @Override
                        public void onComplete(@NonNull Task<Void>
                            task) {
                            Toast.makeText(context, "Deleted",
                                Toast.LENGTH_SHORT).show();
                        }
                    }).addOnFailureListener(new OnFailureListener() {
                        @Override
                        public void onFailure(@NonNull Exception e) {
                            Toast.makeText(context, "Something Went
Wrong", Toast.LENGTH_SHORT).show();
                        }
                    });
            notifyItemRemoved(position);
        }
    };
    builder.setNegativeButton(
        "Cancel",
        new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialogInterface, int i) {
                dialogInterface.cancel();
            }
        }
    );
    AlertDialog dialog = null;

    try {
        dialog = builder.create();
    } catch (Exception e) {
        e.printStackTrace();
    }
    if (dialog!=null)
        dialog.show();
    });
}

@Override
public int getItemCount() {
    return list.size();
}

public class noticeViewAdapter extends RecyclerView.ViewHolder {

    private Button deleteNoticeBtn;
    private TextView deleteNoticeTitle;
    private ImageView deleteNoticeImageView;

```

```

        public noticeViewAdapter(@NonNull View itemView) {
            super(itemView);
            deleteNoticeBtn = itemView.findViewById(R.id.deleteNoticeBtn);
            deleteNoticeTitle = itemView.findViewById(R.id.deleteNoticeTitle);
            deleteNoticeImageView = itemView.findViewById(R.id.deleteNoticeImageView);
        }
    }
}

```

➤ Delete Notice Activity

```

package com.example.admindoca.Notice;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.annotation.SuppressLint;
import android.os.Bundle;
import android.view.View;
import android.widget.ProgressBar;
import android.widget.Toast;

import com.example.admindoca.R;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;

public class deleteNoticeActivity extends AppCompatActivity {

    private RecyclerView deleteNoticeRecycler;
    private ProgressBar progressBar;
    private ArrayList<NoticeData> list;
    private noticeAdapter adapter;

    private DatabaseReference reference;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_delete_notice);

        deleteNoticeRecycler=findViewById(R.id.deleteNoticeRecycler);
    }
}

```

```

        progressBar=findViewById(R.id.progressBar);

        reference= FirebaseDatabase.getInstance().getReference().child("Notice");

        deleteNoticeRecycler.setLayoutManager(new LinearLayoutManager(this));
        deleteNoticeRecycler.setHasFixedSize(true);

        getNotice();
    }

    private void getNotice() {

        reference.addValueEventListener(new ValueEventListener() {
            @SuppressWarnings("NotifyDataSetChanged")
            @Override
            public void onDataChange(@NonNull DataSnapshot snapshot) {
                list=new ArrayList<>();

                for (DataSnapshot snapshot1 : snapshot.getChildren()){
                    NoticeData data = snapshot1.getValue(NoticeData.class);
                    list.add(data);
                }
                adapter=new noticeAdapter(deleteNoticeActivity.this,list);
                adapter.notifyDataSetChanged();
                progressBar.setVisibility(View.GONE);

                deleteNoticeRecycler.setAdapter(adapter);

            }

            @Override
            public void onCancelled(@NonNull DatabaseError error) {

                progressBar.setVisibility(View.GONE);

                Toast.makeText(deleteNoticeActivity.this, error.getMessage(),
                    Toast.LENGTH_SHORT).show();

            }

        });
    }
}

```

Upload Notice. xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    tools:context="com.example.admindoca.Notice.uploadnotice">

```

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">

    <com.google.android.material.card.MaterialCardView
        android:id="@+id/AddPost"
        android:layout_width="124dp"
        android:layout_height="150dp"
        android:layout_gravity="center"
        android:layout_margin="10dp"
        app:cardElevation="5dp">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:orientation="vertical">

            <ImageView
                android:layout_width="64dp"
                android:layout_height="64dp"
                android:layout_marginTop="10dp"
                android:background="@drawable/circle_red"
                android:padding="15dp"
                android:src="@drawable/ic_post"
                tools:ignore="ContentDescription" />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginTop="15dp"
                android:padding="5dp"
                android:text="Add Post"
                android:textColor="#4E4E4E"
                android:textStyle="bold"
                tools:ignore="HardcodedText" />

            <View
                android:layout_width="match_parent"
                android:layout_height="1dp"
                android:layout_marginTop="10dp"
                android:background="@color/light_gray" />

        </LinearLayout>

    </com.google.android.material.card.MaterialCardView>

    <com.google.android.material.textfield.TextInputLayout
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:layout_width="match_parent"
        android:layout_marginTop="16dp"
        android:layout_height="match_parent">

        <com.google.android.material.textfield.TextInputEditText
            android:id="@+id/NoticeTitle"

```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Title" />
    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.button.MaterialButton
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Upload Post"
        android:layout_marginTop="16dp"
        android:textAllCaps="false"
        android:id="@+id/UploadPostBtn"/>
    <com.google.android.material.card.MaterialCardView
        android:layout_width="match_parent"
        android:layout_marginTop="16dp"
        android:layout_height="400dp">
        <ImageView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:scaleType="centerCrop"
            android:id="@+id/PostView"/>

    </com.google.android.material.card.MaterialCardView>

</LinearLayout>

</ScrollView>

```

Newsfeed item layout

```

<?xml version="1.0" encoding="utf-8"?>
<com.google.android.material.card.MaterialCardView
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="16dp"
        android:orientation="vertical">

        <LinearLayout
            android:layout_width="match_parent"
            android:gravity="center_vertical"
            android:orientation="horizontal"
            android:layout_height="wrap_content">

            <de.hdodenhof.circleimageview.CircleImageView
                android:layout_width="36dp"

```

```

        android:layout_height="36dp"
        android:src="@drawable/tutor"/>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Admin DOCA App"
    android:layout_marginLeft="16dp"
    android:textSize="18dp"
    android:textColor="@color/textColor"/>

<com.google.android.material.button.MaterialButton
    android:id="@+id/deleteNoticeBtn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Delete"
    android:layout_marginLeft="-100dp"
    style="@style/Widget.MaterialComponents.Button.OutlinedButton"/>

</LinearLayout>

<TextView
    android:id="@+id/deleteNoticeTitle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:text="Department of Computer Application means keep Calm and Study
Online"
    android:textSize="15sp"/>

<ImageView
    android:id="@+id/deleteNoticeImageView"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:scaleType="fitXY"/>

</LinearLayout>
</com.google.android.material.card.MaterialCardView>

```

Upload Gallery Photo.java

```

package com.example.admindoca;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.cardview.widget.CardView;

```



```

import android.app.ProgressDialog;
import android.content.Intent;
import android.graphics.Bitmap;
import android.net.Uri;
import android.os.Bundle;
import android.provider.MediaStore;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.Spinner;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.google.firebase.storage.UploadTask;

import java.io.ByteArrayOutputStream;
import java.io.IOException;

public class uploadimage extends AppCompatActivity {

    private Spinner image_category;
    private CardView AddImage;
    private Button UploadImage;
    private ImageView GalleryImgView;

    private String category;
    private final int REQ=1;
    private Bitmap bitmap;
    ProgressDialog pd;

    private DatabaseReference reference;
    private StorageReference storageReference;
    String downloadurl;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_uploadimage);

        AddImage = findViewById(R.id.addGalleryImg);
        image_category = findViewById(R.id.image_category);
        UploadImage = findViewById(R.id.UploadImgBtn);
        GalleryImgView = findViewById(R.id.GalleryImgView);

        reference = FirebaseDatabase.getInstance().getReference().child("Gallery");
        storageReference = FirebaseStorage.getInstance().getReference().child("Gallery");

        pd = new ProgressDialog(this);

```

```

        String[] items = new String[] {"Select Category", "Department
Photos", "Festivals", "Other Events"};
        image_category.setAdapter(new ArrayAdapter<String>(this,
android.R.layout.simple_spinner_dropdown_item, items));

        image_category.setOnItemClickListener(new AdapterView.OnItemClickListener()
{
    @Override
    public void onItemClick(AdapterView<?> adapterView, View view, int i, long
1) {
        category = image_category.getSelectedItem().toString();
    }

    @Override
    public void onNothingSelected(AdapterView<?> adapterView) {

    }
});

AddImage.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        openGallery();
    }
});

UploadImage.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (bitmap == null) {
            Toast.makeText(uploadimage.this, "Please upload image",
Toast.LENGTH_SHORT).show();
        } else if (category.equals("Select Category")) {
            Toast.makeText(uploadimage.this, "Please select image category",
Toast.LENGTH_SHORT).show();
        } else {
            pd.setMessage("Uploading...");
            pd.show();

            Uploadimage();
        }
    }
});

}

private void Uploadimage() {
    ByteArrayOutputStream baos = new ByteArrayOutputStream();
    bitmap.compress(Bitmap.CompressFormat.JPEG, 50, baos);
    byte[] finalimg = baos.toByteArray();
    final StorageReference filepath;
    filepath = storageReference.child(finalimg+"jpg");
    final UploadTask uploadTask = filepath.putBytes(finalimg);
    uploadTask.addOnCompleteListener(uploadimage.this, new
OnCompleteListener<UploadTask.TaskSnapshot>() {

```

```

        @Override
        public void onComplete(@NonNull Task<UploadTask.TaskSnapshot> task) {
            if (task.isSuccessful()) {
                uploadTask.addOnSuccessListener(new
OnSuccessListener<UploadTask.TaskSnapshot>() {
                    @Override
                    public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {
                        filepath.getDownloadUrl().addOnSuccessListener(new
OnSuccessListener<Uri>() {
                            @Override
                            public void onSuccess(Uri uri) {
                                downloadurl = String.valueOf(uri);
                                uploadData();
                            }
                        });
                    }
                });
            } else {
                pd.dismiss();
                Toast.makeText(uploadimage.this, "Something went wrong",
Toast.LENGTH_SHORT).show();
            }
        }
    });
}

private void uploadData() {
    reference = reference.child(category);
    final String uniquekey = reference.push().getKey();
    reference.child(uniquekey).setValue(downloadurl).addOnSuccessListener(new
OnSuccessListener<Void>() {
        @Override
        public void onSuccess(Void avoid) {
            pd.dismiss();
            Toast.makeText(uploadimage.this, "Image Uploaded Successfully",
Toast.LENGTH_SHORT).show();
        }

    }).addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {
            pd.dismiss();
            Toast.makeText(uploadimage.this, "Something went wrong",
Toast.LENGTH_SHORT).show();
        }
    });
}

private void openGallery() {
    Intent pickImage = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);

    startActivityForResult(pickImage, REQ);
}

```

```

    }
    @Override
    protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
        super.onActivityResult(requestCode, resultCode, data);
        if (requestCode == REQ && resultCode == RESULT_OK) {

            Uri uri = data.getData();
            try {
                bitmap = MediaStore.Images.Media.getBitmap(getContentResolver(), uri);
            } catch (IOException e) {
                e.printStackTrace();
            }
            GalleryImageView.setImageBitmap(bitmap);
        }
    }
}

```

Upload Photo.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    tools:context=".uploadimage">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <com.google.android.material.card.MaterialCardView
            android:id="@+id/addGalleryImg"
            android:layout_width="130dp"
            android:layout_height="150dp"
            android:layout_margin="10dp"
            android:layout_gravity="center"
            app:cardElevation="5dp">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:gravity="center"
                android:orientation="vertical">

                <ImageView
                    android:layout_width="64dp"

```

```

        android:layout_height="64dp"
        android:background="@drawable/circle_green"
        android:padding="15dp"
        android:layout_marginTop="10dp"
        android:src="@drawable/ic_img"
        tools:ignore="ContentDescription" />

<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:background="@color/light_gray"
    android:layout_marginTop="10dp"
    />
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Add Image"
    android:textStyle="bold"
    android:padding="5dp"
    android:layout_marginTop="15dp"
    android:textColor="#4E4E4E"
    tools:ignore="HardcodedText" />
</LinearLayout>
</com.google.android.material.card.MaterialCardView>

<Spinner
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/image_category"
    android:spinnerMode="dropdown"
    android:background="@android:drawable/btn_dropdown"
    android:gravity="center"
    android:layout_marginVertical="16dp"/>

<com.google.android.material.button.MaterialButton
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Upload Image"
    android:layout_marginTop="16dp"
    android:textAllCaps="false"
    android:id="@+id/UploadImgBtn"/>
<com.google.android.material.card.MaterialCardView
    android:layout_width="match_parent"
    android:layout_marginTop="16dp"
    android:layout_height="400dp">
    <ImageView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:scaleType="centerCrop"
        android:id="@+id/GalleryImgView"/>

</com.google.android.material.card.MaterialCardView>

</LinearLayout>
</ScrollView>

```

Upload Pdf.java

```
package com.example.admindoca;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.cardview.widget.CardView;

import android.annotation.SuppressLint;
import android.app.ProgressDialog;
import android.content.Intent;
import android.database.Cursor;
import android.graphics.Bitmap;
import android.icu.text.CaseMap;
import android.net.Uri;
import android.os.Bundle;
import android.provider.MediaStore;
import android.provider.OpenableColumns;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.google.firebase.storage.UploadTask;

import java.io.File;
import java.io.IOException;
import java.util.HashMap;

public class uploadpdf extends AppCompatActivity {
    private CardView AddPdf;
    private EditText pdfTitle;
    private Button UploadPdfBtn;
    private TextView pdfTextview;
    private String pdfname,title;

    private final int REQ = 1;
    private Uri pdfData;
    private DatabaseReference Reference;
    private StorageReference storageReference;
    String downloadurl = "";
    private ProgressDialog pd;
    private String pdfName;
```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_uploadpdf);

    Reference = FirebaseDatabase.getInstance().getReference();
    storageReference = FirebaseStorage.getInstance().getReference();

    pd = new ProgressDialog(this);

    AddPdf = findViewById(R.id.AddPdf);
    pdfTitle = findViewById(R.id.pdfTitle);
    UploadPdfBtn = findViewById(R.id.UploadPdfBtn);
    pdfTextview = findViewById(R.id.pdfTextview);

    UploadPdfBtn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            title = pdfTitle.getText().toString();
            if (title.isEmpty()) {
                pdfTitle.setError("Empty");
                pdfTitle.requestFocus();
            } else if (pdfData == null) {
                Toast.makeText(uploadpdf.this, "Please Upload Pdf File",
Toast.LENGTH_SHORT).show();
            } else {
                uploadPdf();
            }
        }
    });

    AddPdf.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View view) {
            openGallery();
        }
    });
}

private void uploadPdf() {
    pd.setTitle("Please wait...");
    pd.setMessage("Uploading pdf");
    pd.show();
    StorageReference reference = storageReference.child("pdf/"+pdfName+"-"+
"+System.currentTimeMillis()+".pdf");
    reference.putFile(pdfData)
        .addOnSuccessListener(new OnSuccessListener<UploadTask.TaskSnapshot>() {
            @Override

```

```

        public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {
            Task<Uri> uriTask=taskSnapshot.getStorage().getDownloadUrl();
            while (!uriTask.isComplete());
            Uri uri = uriTask.getResult();
            uploadData(String.valueOf(uri));
        }
    }).addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {
            pd.dismiss();
            Toast.makeText(uploadpdf.this, "Something went wrong",
Toast.LENGTH_SHORT).show();
        }
    });
}

private void uploadData(String downloadurl) {
    String uniqueKey = Reference.child("pdf").push().getKey();

    HashMap data = new HashMap();
    data.put("pdfTitle", title);
    data.put("pdfUrl", downloadurl);

    Reference.child("pdf").child(uniqueKey).setValue(data).addOnCompleteListener(new
OnCompleteListener<Void>() {
        @Override
        public void onComplete(@NonNull Task<Void> task) {
            pd.dismiss();
            Toast.makeText(uploadpdf.this, "Pdf uploaded successfully",
Toast.LENGTH_SHORT).show();
            pdfTitle.setText("");
        }
    }).addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {
            pd.dismiss();
            Toast.makeText(uploadpdf.this, "Failed to upload Pdf file",
Toast.LENGTH_SHORT).show();
        }
    });
}

private void openGallery() {
    Intent intent = new Intent();
    intent.setType("application/pdf");
    intent.setAction(Intent.ACTION_GET_CONTENT);
    startActivityForResult(Intent.createChooser(intent, "Select pdf Files"), REQ);
}

@SuppressLint("Range")
@Override
protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent
data) {
    super.onActivityResult(requestCode, resultCode, data);

```



```

if (requestCode == REQ && resultCode == RESULT_OK){
    pdfData = data.getData();

    if (pdfData.toString().startsWith("content://")){
        Cursor cursor = null;
        try {
            cursor =
uploadpdf.this.getContentResolver().query(pdfData,null,null,null,null);
            if (cursor != null && cursor.moveToFirst()){
                pdfName =
cursor.getString(cursor.getColumnIndex(OpenableColumns.DISPLAY_NAME));
            }
        } catch (Exception e) {
            e.printStackTrace();
        }

    }else if (pdfData.toString().startsWith("file://")){
        pdfName = new File(pdfData.toString()).getName();
    }

    pdfTextView.setText(pdfName);
}
}
}

```

Upload Pdf.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    tools:context="com.example.admindoca.uploadpdf">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <com.google.android.material.card.MaterialCardView
            android:id="@+id/AddPdf"
            android:layout_width="130dp"
            android:layout_height="150dp"
            android:layout_margin="10dp"
            android:layout_gravity="center"
            app:cardElevation="5dp">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:gravity="center"

```

```

        android:orientation="vertical">

        <ImageView
            android:layout_width="64dp"
            android:layout_height="64dp"
            android:background="@drawable/circle_purple"
            android:padding="15dp"
            android:layout_marginTop="10dp"
            android:src="@drawable/ic_pdf"
            tools:ignore="ContentDescription" />

        <View
            android:layout_width="match_parent"
            android:layout_height="1dp"
            android:background="@color/light_gray"
            android:layout_marginTop="10dp"
            />
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Add Pdf"
            android:textStyle="bold"
            android:padding="5dp"
            android:layout_marginTop="15dp"
            android:textColor="#4E4E4E"
            tools:ignore="HardcodedText" />

    </LinearLayout>

</com.google.android.material.card.MaterialCardView>

<TextView
    android:id="@+id/pdfTextview"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="No File Selected"
    android:textSize="15sp"
    android:layout_gravity="center"/>

<com.google.android.material.textfield.TextInputLayout
    style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
    android:layout_width="match_parent"
    android:layout_marginTop="16dp"
    android:layout_height="match_parent">

    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/pdfTitle"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Title" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.button.MaterialButton
    android:layout_width="match_parent"
    android:layout_height="wrap_content"

```

```

        android:text="Upload Pdf"
        android:layout_marginTop="16dp"
        android:textAllCaps="false"
        android:id="@+id/UploadPdfBtn"/>

```

```

</LinearLayout>

```

```

</ScrollView>

```

Upload Faculty.java

```

package com.example.admindoca.faculty;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.LinearLayout;
import android.widget.Toast;

import com.example.admindoca.R;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;
import java.util.List;

public class updatefaculty extends AppCompatActivity {

    FloatingActionButton fab;
    private RecyclerView JP, PP, DBMS, Ops, SE, Math, Other;
    private LinearLayout
    JpNoData, PpNoData, DBNoData, OpsNoData, SENoData, MathNoData, OtherNoData;
    private List<teacherData>list1, list2, list3, list4, list5, list6, list7;

    private DatabaseReference reference, dbRef;
    private teacherAdapter adapter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_updatefaculty);
    }

```

```

JP = findViewById(R.id.JP);
PP = findViewById(R.id.PP);
DBMS = findViewById(R.id.DBMS);
Oops = findViewById(R.id.Oops);
SE = findViewById(R.id.SE);
Math = findViewById(R.id.Math);
Other = findViewById(R.id.Other);

JpNoData = findViewById(R.id.JpNoData);
PpNoData = findViewById(R.id.PpNoData);
DBNoData = findViewById(R.id.DBNoData);
OopsNoData = findViewById(R.id.OopsNoData);
SENoData = findViewById(R.id.SENoData);
MathNoData = findViewById(R.id.MathNoData);
OtherNoData = findViewById(R.id.OtherNoData);

reference = FirebaseDatabase.getInstance().getReference().child("teacher");

JP();
PP();
DBMS();
Oops();
SE();
Math();
Other();

fab = findViewById(R.id.fab);

fab.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        startActivity(new Intent(updatefaculty.this,
            Addteachers.class));
    }
});

private void Other() {
    dbRef = reference.child("Other Subjects");
    dbRef.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            list7 = new ArrayList<>();
            if (!snapshot.exists()) {
                OtherNoData.setVisibility(View.VISIBLE);
                Other.setVisibility(View.GONE);
            } else {
                OtherNoData.setVisibility(View.GONE);
                Other.setVisibility(View.VISIBLE);
                for (DataSnapshot dataSnapshot : snapshot.getChildren()) {
                    teacherData data = dataSnapshot.getValue(teacherData.class);

```

```

        list7.add(data);
    }
    Other.setHasFixedSize(true);
    Other.setLayoutManager(new LinearLayoutManager(updatefaculty.this));
    adapter = new teacherAdapter(list7,updatefaculty.this,"Other
Subjects");
    Other.setAdapter(adapter);

    }

    }

    @Override
    public void onCancelled(@NonNull DatabaseError error) {
        Toast.makeText(updatefaculty.this, error.getMessage(),
Toast.LENGTH_SHORT).show();
    }
    });
}

private void Math() {
    dbRef = reference.child("Mathematics");
    dbRef.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            list6 = new ArrayList<>();
            if (!snapshot.exists()){
                MathNoData.setVisibility(View.VISIBLE);
                Math.setVisibility(View.GONE);
            }else{
                MathNoData.setVisibility(View.GONE);
                Math.setVisibility(View.VISIBLE);
                for (DataSnapshot dataSnapshot:snapshot.getChildren()){
                    teacherData data = dataSnapshot.getValue(teacherData.class);
                    list6.add(data);
                }
                Math.setHasFixedSize(true);
                Math.setLayoutManager(new LinearLayoutManager(updatefaculty.this));
                adapter = new teacherAdapter(list6,updatefaculty.this,"Mathematics");
                Math.setAdapter(adapter);
            }
        }

        @Override
        public void onCancelled(@NonNull DatabaseError error) {
            Toast.makeText(updatefaculty.this, error.getMessage(),
Toast.LENGTH_SHORT).show();
        }
    });
}

private void SE() {

```

```

dbRef = reference.child("Software Engineering");
dbRef.addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot snapshot) {
        list5 = new ArrayList<>();
        if (!snapshot.exists()) {
            SENoData.setVisibility(View.VISIBLE);
            SE.setVisibility(View.GONE);
        } else {
            SENoData.setVisibility(View.GONE);
            SE.setVisibility(View.VISIBLE);
            for (DataSnapshot dataSnapshot: snapshot.getChildren()) {
                teacherData data = dataSnapshot.getValue(teacherData.class);
                list5.add(data);
            }
            SE.setHasFixedSize(true);
            SE.setLayoutManager(new LinearLayoutManager(updatefaculty.this));
            adapter = new teacherAdapter(list5, updatefaculty.this, "Software
Engineering");
            SE.setAdapter(adapter);
        }
    }

    @Override
    public void onCancelled(@NonNull DatabaseError error) {
        Toast.makeText(updatefaculty.this, error.getMessage(),
Toast.LENGTH_SHORT).show();
    }
});
}

private void Oops() {
    dbRef = reference.child("Object Oriented Programming Using C & C++");
    dbRef.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            list4 = new ArrayList<>();
            if (!snapshot.exists()) {
                OopsNoData.setVisibility(View.VISIBLE);
                Oops.setVisibility(View.GONE);
            } else {
                OopsNoData.setVisibility(View.GONE);
                Oops.setVisibility(View.VISIBLE);
                for (DataSnapshot dataSnapshot: snapshot.getChildren()) {
                    teacherData data = dataSnapshot.getValue(teacherData.class);
                    list4.add(data);
                }
                Oops.setHasFixedSize(true);
                Oops.setLayoutManager(new LinearLayoutManager(updatefaculty.this));
                adapter = new teacherAdapter(list4, updatefaculty.this, "Object
Oriented Programming Using C & C++");
                Oops.setAdapter(adapter);
            }
        }
    }
});
}

```

```

        }

    }

    @Override
    public void onCancelled(@NonNull DatabaseError error) {
        Toast.makeText(updatefaculty.this, error.getMessage(),
Toast.LENGTH_SHORT).show();
    }
});

}

private void DBMS() {
    dbRef = reference.child("DBMS");
    dbRef.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            list3 = new ArrayList<>();
            if (!snapshot.exists()) {
                DBNoData.setVisibility(View.VISIBLE);
                DBMS.setVisibility(View.GONE);
            } else {
                DBNoData.setVisibility(View.GONE);
                DBMS.setVisibility(View.VISIBLE);
                for (DataSnapshot dataSnapshot: snapshot.getChildren()) {
                    teacherData data = dataSnapshot.getValue(teacherData.class);
                    list3.add(data);
                }
                DBMS.setHasFixedSize(true);
                DBMS.setLayoutManager(new LinearLayoutManager(updatefaculty.this));
                adapter = new teacherAdapter(list3, updatefaculty.this, "DBMS");
                DBMS.setAdapter(adapter);
            }
        }
    }

    @Override
    public void onCancelled(@NonNull DatabaseError error) {
        Toast.makeText(updatefaculty.this, error.getMessage(),
Toast.LENGTH_SHORT).show();
    }
});

}

private void PP() {
    dbRef = reference.child("Python Programming");
    dbRef.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            list2 = new ArrayList<>();

```

```

        if (!snapshot.exists()) {
            PpNoData.setVisibility(View.VISIBLE);
            PP.setVisibility(View.GONE);
        } else {

            PpNoData.setVisibility(View.GONE);
            PP.setVisibility(View.VISIBLE);
            for (DataSnapshot dataSnapshot: snapshot.getChildren()) {
                teacherData data = dataSnapshot.getValue(teacherData.class);
                list2.add(data);
            }
            PP.setHasFixedSize(true);
            PP.setLayoutManager(new LinearLayoutManager(updatefaculty.this));
            adapter = new teacherAdapter(list2, updatefaculty.this, "Python
Programming");
            PP.setAdapter(adapter);

        }

    }

    @Override
    public void onCancelled(@NonNull DatabaseError error) {
        Toast.makeText(updatefaculty.this, error.getMessage(),
Toast.LENGTH_SHORT).show();
    }
});
}

private void JP() {
    dbRef = reference.child("Java Programming");
    dbRef.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            list1 = new ArrayList<>();
            if (!snapshot.exists()) {
                JpNoData.setVisibility(View.VISIBLE);
                JP.setVisibility(View.GONE);
            } else {

                JpNoData.setVisibility(View.GONE);
                JP.setVisibility(View.VISIBLE);
                for (DataSnapshot dataSnapshot: snapshot.getChildren()) {
                    teacherData data = dataSnapshot.getValue(teacherData.class);
                    list1.add(data);
                }
                JP.setHasFixedSize(true);
                JP.setLayoutManager(new LinearLayoutManager(updatefaculty.this));
                adapter = new teacherAdapter(list1, updatefaculty.this, "Java
Programming");
                JP.setAdapter(adapter);

            }

        }

    }
}

```



```

        @Override
        public void onCancelled(@NonNull DatabaseError error) {
            Toast.makeText(updatefaculty.this, error.getMessage(),
                Toast.LENGTH_SHORT).show();
        }
    });
}
}

```

Update Faculty.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".faculty.updatefaculty">

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/fab"
        style="@style/Widget.Design.FloatingActionButton"
        android:layout_gravity="bottom|right"
        android:layout_margin="16dp"
        android:src="@drawable/ic_add"
        android:elevation="3dp"
        app:fabSize="normal"
        tools:ignore="RtlHardcoded" />

    <androidx.core.widget.NestedScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="vertical">

            <TextView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:text="Java Programming"
                android:gravity="center"
                android:textSize="20sp"
                android:textColor="@color/textColor"
                android:textStyle="bold"
                android:layout_marginTop="16dp"/>

            <View
                android:layout_width="match_parent"
                android:layout_height="1dp"

```

```

        android:layout_marginVertical="8dp"
        android:background="#c1c1c1"/>

<androidx.recyclerview.widget.RecyclerView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/JP"
    android:visibility="gone"
    android:nestedScrollingEnabled="false"/>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:id="@+id/JpNoData">

    <include layout="@layout/no_data_found"/>

</LinearLayout>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Python Programming"
    android:gravity="center"
    android:textSize="20sp"
    android:textColor="@color/textColor"
    android:textStyle="bold"
    android:layout_marginTop="16dp"/>

<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:layout_marginVertical="8dp"
    android:background="#c1c1c1"/>

<androidx.recyclerview.widget.RecyclerView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/PP"
    android:visibility="gone"
    android:nestedScrollingEnabled="false"/>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:id="@+id/PpNoData">

    <include layout="@layout/no_data_found"/>

</LinearLayout>
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="DBMS"
    android:gravity="center"
    android:textSize="20sp"

```

```

        android:textColor="@color/textColor"
        android:textStyle="bold"
        android:layout_marginTop="16dp"/>

<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:layout_marginVertical="8dp"
    android:background="#c1c1c1"/>

<androidx.recyclerview.widget.RecyclerView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/DBMS"
    android:visibility="gone"
    android:nestedScrollingEnabled="false"/>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:id="@+id/DBNoData">

    <include layout="@layout/no_data_found"/>

</LinearLayout>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Object Oriented Programming Using C & C++"
    android:gravity="center"
    android:textSize="20sp"
    android:textColor="@color/textColor"
    android:textStyle="bold"
    android:layout_marginTop="16dp"/>

<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:layout_marginVertical="8dp"
    android:background="#c1c1c1"/>

<androidx.recyclerview.widget.RecyclerView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/Oops"
    android:visibility="gone"
    android:nestedScrollingEnabled="false"/>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:id="@+id/OopsNoData">

    <include layout="@layout/no_data_found"/>

```

```

</LinearLayout>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Software Engineering"
    android:gravity="center"
    android:textSize="20sp"
    android:textColor="@color/textColor"
    android:textStyle="bold"
    android:layout_marginTop="16dp"/>

<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:layout_marginVertical="8dp"
    android:background="#c1c1c1"/>

<androidx.recyclerview.widget.RecyclerView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/SE"
    android:visibility="gone"
    android:nestedScrollingEnabled="false"/>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:id="@+id/SENoData">

    <include layout="@layout/no_data_found"/>

</LinearLayout>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Mathematics"
    android:gravity="center"
    android:textSize="20sp"
    android:textColor="@color/textColor"
    android:textStyle="bold"
    android:layout_marginTop="16dp"/>

<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:layout_marginVertical="8dp"
    android:background="#c1c1c1"/>

<androidx.recyclerview.widget.RecyclerView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/Math"
    android:visibility="gone"
    android:nestedScrollingEnabled="false"/>

```

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:id="@+id/MathNoData">

    <include layout="@layout/no_data_found"/>

</LinearLayout>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Others"
    android:gravity="center"
    android:textSize="20sp"
    android:textColor="@color/textColor"
    android:textStyle="bold"
    android:layout_marginTop="16dp"/>

<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:layout_marginVertical="8dp"
    android:background="#c1c1c1"/>

<androidx.recyclerview.widget.RecyclerView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/Other"
    android:visibility="gone"
    android:nestedScrollingEnabled="false"/>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:id="@+id/OtherNoData">

    <include layout="@layout/no_data_found"/>

</LinearLayout>

</LinearLayout>

</androidx.core.widget.NestedScrollView>

</androidx.coordinatorlayout.widget.CoordinatorLayout>

```

User App Source Code

Home Screen. Java

```
package com.example.doca;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;
import androidx.navigation.NavController;
import androidx.navigation.Navigation;
import androidx.navigation.ui.NavigationUI;

import android.annotation.SuppressLint;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.SharedPreferences;
import android.net.Uri;
import android.os.Bundle;
import android.view.MenuItem;
import android.widget.Toast;

import com.example.doca.pdf.EbookActivity;
import com.google.android.material.bottomnavigation.BottomNavigationView;
import com.google.android.material.dialog.MaterialAlertDialogBuilder;
import com.google.android.material.navigation.NavigationView;

import java.util.Objects;

import static androidx.appcompat.app.AppCompatActivity.MODE_NIGHT_FOLLOW_SYSTEM;

public class MainActivity extends AppCompatActivity implements
NavigationView.OnNavigationItemSelectedListener {

    private BottomNavigationView bottomNavigationView;
    private NavController navController;

    private DrawerLayout drawerLayout;
    private ActionBarDrawerToggle toggle;
    private NavigationView navigationView;

    private SharedPreferences sharedPreferences;
    private SharedPreferences.Editor editor;
    private int checkedItem;
    private String selected;
```

```

private final String CHECKEDITEM = "checked_item";

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

    sharedPreferences = this.getSharedPreferences("themes", Context.MODE_PRIVATE);
    editor=sharedPreferences.edit();

    switch (getCheckedItem()){
        case 0:
            AppCompatDelegate.setDefaultNightMode(MODE_NIGHT_FOLLOW_SYSTEM);
            break;

        case 1:
            AppCompatDelegate.setDefaultNightMode(AppCompatDelegate.MODE_NIGHT_YES);
            break;

        case 2:
            AppCompatDelegate.setDefaultNightMode(AppCompatDelegate.MODE_NIGHT_NO);
            break;
    }

    bottomNavigationView = findViewById(R.id.bottomNavigationView);
    navController = Navigation.findNavController(this,R.id.frame_layout);

    drawerLayout = findViewById(R.id.drawerLayout);
    navigationView = findViewById(R.id.navigation_view);

    toggle = new
ActionBarDrawerToggle(this, drawerLayout,R.string.start,R.string.close);

    drawerLayout.addDrawerListener(toggle);
    toggle.syncState();

    Objects.requireNonNull(getSupportActionBar()).setDisplayHomeAsUpEnabled(true);
    navigationView.setNavigationItemSelectedListener(this);

    NavigationUI.setupWithNavController(bottomNavigationView, navController);

}

@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {
    if (toggle.onOptionsItemSelected(item)) {
        return true;
    }
    return true;
}

```

```

@Override
public boolean onNavigationItemSelected(@NonNull MenuItem item) {
    Intent intent;

    switch (item.getItemId()) {

        case R.id.navigation_video:
            Intent intent2 = new Intent(Intent.ACTION_VIEW);
            intent2.setData(Uri.parse("https://www.codewithharry.com/"));
            startActivity(intent2);

            break;

        case R.id.navigation_rate:

            Uri uri =
Uri.parse("https://play.google.com/store/apps/details?id="+getApplicationContext().getPackageName());
            Intent i = new Intent(Intent.ACTION_VIEW);
            try {
                startActivity(i);
            } catch (Exception e) {
                Toast.makeText(this, "Unable to open\n"+e.getMessage(),
Toast.LENGTH_SHORT).show();
            }
            break;

        case R.id.navigation_ebook:

            intent = new Intent(MainActivity.this, EbookActivity.class);
            startActivity(intent);
            break;

        case R.id.navigation_website:
            Intent intent3 = new Intent(Intent.ACTION_VIEW);
            intent3.setData(Uri.parse("https://evarsity.srmist.edu.in/srmsip/"));
            startActivity(intent3);
            break;

        case R.id.navigation_share:

            try {
                Intent intent1 = new Intent(Intent.ACTION_SEND);
                intent1.setType("text/plain");
                intent1.putExtra(Intent.EXTRA_SUBJECT, "DOCA");

intent1.putExtra(Intent.EXTRA_TEXT, "https://play.google.com/store/apps/details?id="+getAp
plicationContext().getPackageName());
                startActivity(Intent.createChooser(intent1, "Share with"));
            } catch (Exception e) {
                Toast.makeText(this, "Unable to share this app.",
Toast.LENGTH_SHORT).show();
            }
            break;
    }
}

```



```

        case R.id.navigation_color:

            showDialog();

            break;
    }

    return true;
}

private void showDialog() {

    String [] themes = this.getResources().getStringArray(R.array.theme);

    MaterialAlertDialogBuilder builder= new MaterialAlertDialogBuilder(this);
    builder.setTitle("Select Theme");
    builder.setSingleChoiceItems(R.array.theme, getCheckedItem(), new
DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialogInterface, int i) {
            selected = themes[i];
            checkedItem = i;

        }
    });

    builder.setPositiveButton("Ok", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialogInterface, int i) {
            if (selected==null){
                selected = themes[i];
                checkedItem = i;

            }
            switch (selected){
                case "System Default":
                    AppCompatActivity.setDefaultNightMode(MODE_NIGHT_FOLLOW_SYSTEM);
                    break;

                case "Dark":
AppCompatActivity.setDefaultNightMode(AppCompatActivity.MODE_NIGHT_YES);
                    break;

                case "Light":
AppCompatActivity.setDefaultNightMode(AppCompatActivity.MODE_NIGHT_NO);
                    break;
            }
            setCheckedItem(checkedItem);
        }
    });

    builder.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialogInterface, int i) {

```

```

        dialogInterface.dismiss();
    }
});
AlertDialog dialog = builder.create();
dialog.show();

}

private int getCheckedItem() {
    return sharedPreferences.getInt(CHECKEDITEM, 0);
}

private void setCheckedItem(int i) {
    editor.putInt(CHECKEDITEM, i);
    editor.apply();
}

@Override
@SuppressLint("SuspiciousIndentation")
public void onBackPressed() {
    if (drawerLayout.isDrawerOpen(GravityCompat.START)) {
        drawerLayout.closeDrawer(GravityCompat.START);
    } else
        super.onBackPressed();
}
}

```

Home Screen. Xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.drawerlayout.widget.DrawerLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/drawerLayout"
    xmlns:app="http://schemas.android.com/apk/res-auto">

    <androidx.constraintlayout.widget.ConstraintLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        tools:context=".MainActivity">

        <com.google.android.material.bottomnavigation.BottomNavigationView
            android:id="@+id/bottomNavigationView"
            android:layout_width="match_parent"
            android:layout_height="50dp"

            app:labelVisibilityMode="labeled"
            app:layout_constraintBottom_toBottomOf="parent"

```

```

        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:menu="@menu/bottom_nav" />

<fragment
    android:id="@+id/frame_layout"
    android:name="androidx.navigation.fragment.NavHostFragment"
    android:layout_width="match_parent"
    android:layout_height="0dp"
    app:defaultNavHost="true"
    app:layout_constraintBottom_toTopOf="@+id/bottomNavigationView"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.0"
    app:navGraph="@navigation/mobile_navigation" />

</androidx.constraintlayout.widget.ConstraintLayout>

<com.google.android.material.navigation.NavigationView
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:id="@+id/navigation_view"
    app:menu="@menu/navigation_drawer"
    android:fitsSystemWindows="true"
    app:headerLayout="@layout/drawer_header"
    app:itemTextAppearance="?android:attr/textAppearanceMedium"
    android:layout_gravity="start"/>

</androidx.drawerlayout.widget.DrawerLayout>

```

Notice fragment. Java

```

package com.example.doca.ui.notice;

import android.annotation.SuppressLint;
import android.os.Bundle;

import androidx.annotation.NonNull;
import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ProgressBar;
import android.widget.Toast;

import com.example.doca.R;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;

```

```

import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;

public class noticeFragment extends Fragment {

    private RecyclerView deleteNoticeRecycler;
    private ProgressBar progressBar;
    private ArrayList<NoticeData> list;
    private noticeAdapter adapter;

    private DatabaseReference reference;

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        View view = inflater.inflate(R.layout.fragment_notice, container, false);

        deleteNoticeRecycler = view.findViewById(R.id.deleteNoticeRecycler);

        progressBar = view.findViewById(R.id.progressBar);

        reference = FirebaseDatabase.getInstance().getReference().child("Notice");

        deleteNoticeRecycler.setLayoutManager(new LinearLayoutManager(getContext()));
        deleteNoticeRecycler.setHasFixedSize(true);

        getNotice();

        return view;
    }

    private void getNotice() {

        reference.addValueEventListener(new ValueEventListener() {
            @SuppressWarnings("NotifyDataSetChanged")
            @Override
            public void onDataChange(@NonNull DataSnapshot snapshot) {
                list=new ArrayList<>();

                for (DataSnapshot snapshot1 : snapshot.getChildren()){
                    NoticeData data = snapshot1.getValue(NoticeData.class);
                    list.add(0,data);
                }
                adapter=new noticeAdapter(getContext(),list);
                adapter.notifyDataSetChanged();
                progressBar.setVisibility(View.GONE);

                deleteNoticeRecycler.setAdapter(adapter);

            }
        });
    }

```

```

        @Override
        public void onCancelled(@NonNull DatabaseError error) {

            progressBar.setVisibility(View.GONE);

            Toast.makeText(getApplicationContext(), error.getMessage(),
Toast.LENGTH_SHORT).show();

        }
    });
}
}

```

➤ Notice Data.java

```

package com.example.doca.ui.notice;

public class NoticeData {
    private String title, image, date, time, key;

    public NoticeData(){

    }

    public NoticeData(String title, String image, String date, String time, String key) {
        this.title = title;
        this.image = image;
        this.date = date;
        this.time = time;
        this.key = key;
    }

    public String getTitle() {
        return title;
    }

    public void setTitle(String title) {
        this.title = title;
    }

    public String getImage() {
        return image;
    }

    public void setImage(String image) {
        this.image = image;
    }

    public String getDate() {
        return date;
    }

    public void setDate(String date) {

```

```

        this.date = date;
    }

    public String getTime() {
        return time;
    }

    public void setTime(String time) {
        this.time = time;
    }

    public String getKey() {
        return key;
    }

    public void setKey(String key) {
        this.key = key;
    }
}

```

➤ Notice Adapter. Java

```

package com.example.doca.ui.notice;

import android.annotation.SuppressLint;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import com.bumptech.glide.Glide;
import com.example.doca.FullImageView;
import com.example.doca.R;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.Task;

import java.util.ArrayList;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

public class noticeAdapter extends RecyclerView.Adapter<noticeAdapter.noticeViewAdapter>
{
    private Context context;
    private ArrayList<NoticeData> list;
}

```

```

public noticeAdapter(Context context, ArrayList<NoticeData> list) {
    this.context = context;
    this.list = list;

}

@NonNull
@Override
public noticeViewAdapter onCreateViewHolder(@NonNull ViewGroup parent, int viewType)
{
    View view = LayoutInflater.from(context).inflate(R.layout.newsfeed_item_layout,
parent, false);

    return new noticeViewAdapter(view);
}

@SuppressLint("SuspiciousIndentation")
@Override
public void onBindViewHolder(@NonNull noticeViewAdapter holder,
@SuppressLint("RecyclerView") int position) {

    NoticeData currentItem = list.get(position);

    holder.newsTitle.setText(currentItem.getTitle());
    holder.time.setText(currentItem.getTime());
    holder.date.setText(currentItem.getDate());

    try {
        if (currentItem.getImage() != null)
            Glide.with(context).load(currentItem.getImage()).into(holder.newsImg);
    } catch (Exception e) {
        e.printStackTrace();
    }

    holder.newsImg.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent intent = new Intent(context, FullImageView.class);
            intent.putExtra("image", currentItem.getImage());
            context.startActivity(intent);
        }
    });

}

@Override
public int getItemCount() {
    return list.size();
}

public class noticeViewAdapter extends RecyclerView.ViewHolder {

    private TextView newsTitle, date, time;
    private ImageView newsImg;

```

```

public noticeViewAdapter(@NonNull View itemView) {
    super(itemView);

    newsTitle = itemView.findViewById(R.id.newsTitle);
    newsImg = itemView.findViewById(R.id.newsImg);
    date = itemView.findViewById(R.id.date);
    time = itemView.findViewById(R.id.time);

}
}
}

```

Notice fragment.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

    tools:context=".ui.notice.noticeFragment">

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/deleteNoticeRecycler"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>

    <ProgressBar
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:id="@+id/progressBar"/>

</RelativeLayout>

```

Notice Layout.java

```

<?xml version="1.0" encoding="utf-8"?>
<com.google.android.material.card.MaterialCardView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="16dp"

```



```

android:background="#36FAECD5"
android:layout_margin="4dp"
android:orientation="vertical">

<LinearLayout
    android:layout_width="match_parent"
    android:gravity="center_vertical"
    android:orientation="horizontal"
    android:layout_height="wrap_content">

    <de.hdodenhof.circleimageview.CircleImageView
        android:layout_width="36dp"
        android:layout_height="36dp"
        android:src="@drawable/logo"/>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="16dp"

        android:orientation="vertical">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text=" SRM IST"
            android:textSize="18dp"
            android:textColor="@color/textColor"/>

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal">

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Date :"
                android:fontFamily="@string/ligth"
                android:textSize="@dimen/text_size01"
                android:textColor="@color/textColor"/>

            <TextView
                android:id="@+id/date"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text=" 21 March 2023"
                android:fontFamily="@string/ligth"
                android:textSize="@dimen/text_size01"
                android:textColor="@color/textColor"/>

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Time :"
                android:layout_marginLeft="@dimen/text_margin"
                android:fontFamily="@string/ligth"

```

```

        android:textSize="@dimen/text_size01"
        android:textColor="@color/textColor"/>
<TextView
    android:id="@+id/time"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="08:03"
    android:fontFamily="@string/ligth"
    android:textSize="@dimen/text_size01"
    android:textColor="@color/textColor"/>

</LinearLayout>

</LinearLayout>

</LinearLayout>

<TextView
    android:id="@+id/newsTitle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:layout_marginTop="@dimen/text_margin"
    android:text="Department of Computer Application means keep Calm and Study
Online"
    android:textSize="15sp"/>

<ImageView
    android:id="@+id/newsImg"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:scaleType="fitXY"/>

</LinearLayout>
</com.google.android.material.card.MaterialCardView>

```

Gallery Fragment.java

```

package com.example.doca.ui.Gallery;

import android.os.Bundle;

import androidx.annotation.NonNull;
import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.GridLayoutManager;

```

```

import androidx.recyclerview.widget.RecyclerView;

import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Toast;

import com.example.doca.R;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Objects;

public class galleryFragment extends Fragment {

    RecyclerView departmentRecycler;
    RecyclerView festivalRecycler;
    RecyclerView otherRecycler;
    GalleryAdapter adapter;
    DatabaseReference reference;

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        View view = inflater.inflate(R.layout.fragment_gallery, container, false);

        departmentRecycler = view.findViewById(R.id.departmentRecycler);
        festivalRecycler = view.findViewById(R.id.festivalRecycler);
        otherRecycler = view.findViewById(R.id.otherRecycler);

        reference = FirebaseDatabase.getInstance().getReference().child("Gallery");

        getDepartmentImage();
        getFestivalImage();
        getOtherImage();

        return view;
    }

    private void getOtherImage() {
        reference.child("Other Events").addValueEventListener(new ValueEventListener() {

            @Override
            public void onDataChange(@NonNull DataSnapshot snapshot) {
                List<String> imageList = new ArrayList<>();
                for (DataSnapshot snapshot1 : snapshot.getChildren()) {

```

```

        imageUrl.add(snapshot1.getValue().toString());
    }
    adapter = new GalleryAdapter(getContext(), imageUrl);
    otherRecycler.setLayoutManager(new GridLayoutManager(getContext(), 3));
    otherRecycler.setAdapter(adapter);
}

@Override
public void onCancelled(@NonNull DatabaseError error) {

}

});

}

private void getFestivalImage() {
    reference.child("Festivals").addValueEventListener(new ValueEventListener() {

        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            List<String> imageUrl = new ArrayList<>();
            for (DataSnapshot snapshot1 : snapshot.getChildren()) {
                imageUrl.add(snapshot1.getValue().toString());
            }
            adapter = new GalleryAdapter(getContext(), imageUrl);
            festivalRecycler.setLayoutManager(new GridLayoutManager(getContext(),
3));
            festivalRecycler.setAdapter(adapter);
        }

        @Override
        public void onCancelled(@NonNull DatabaseError error) {

        }

    });
}

private void getDepartmentImage() {
    reference.child("Department Photos").addValueEventListener(new
ValueEventListener() {

        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            List<String> imageUrl = new ArrayList<>();
            for (DataSnapshot snapshot1 : snapshot.getChildren()) {
                imageUrl.add(snapshot1.getValue().toString());
            }
        }
    }
}

```

```

        adapter = new GalleryAdapter(getContext(), imageList);
        departmentRecycler.setLayoutManager(new GridLayoutManager(getContext(),
3));

        departmentRecycler.setAdapter(adapter);

    }

    @Override
    public void onCancelled(@NonNull DatabaseError error) {

    }

    });
}
}

```

Gallery Fragment. Xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.core.widget.NestedScrollView
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

    tools:context=".ui.Gallery.galleryFragment">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:padding="@dimen/text_margin"
        android:orientation="vertical">

        <com.google.android.material.card.MaterialCardView
            android:layout_width="match_parent"
            android:layout_marginVertical="8dp"
            android:layout_height="280dp">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="vertical">

                <TextView
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_gravity="center_horizontal"
                    android:layout_marginTop="@dimen/text_margin"
                    android:fontFamily="@string/bold"
                    android:text="Department Photos"
                    android:textColor="@color/textColor"
                    android:textSize="@dimen/title_size" />

                <View

```

```

        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:layout_marginVertical="4dp"
        android:background="@color/grey" />

<androidx.recyclerview.widget.RecyclerView
    android:id="@+id/departmentRecycler"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />

</LinearLayout>

</com.google.android.material.card.MaterialCardView>

<com.google.android.material.card.MaterialCardView
    android:layout_width="match_parent"
    android:layout_marginVertical="8dp"
    android:layout_height="280dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Festivals"
            android:layout_marginTop="@dimen/text_margin"
            android:textSize="@dimen/title_size"
            android:layout_gravity="center_horizontal"
            android:textColor="@color/textColor"
            android:fontFamily="@string/bold"/>

        <View
            android:layout_width="match_parent"
            android:layout_height="1dp"
            android:layout_marginVertical="4dp"
            android:background="@color/grey"/>

        <androidx.recyclerview.widget.RecyclerView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/festivalRecycler"/>

    </LinearLayout>

</com.google.android.material.card.MaterialCardView>

<com.google.android.material.card.MaterialCardView

```

```

        android:layout_width="match_parent"
        android:layout_marginVertical="8dp"
        android:layout_height="280dp">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Other Events"
                android:layout_marginTop="@dimen/text_margin"
                android:textSize="@dimen/title_size"
                android:layout_gravity="center_horizontal"
                android:textColor="@color/textColor"
                android:fontFamily="@string/bold"/>

            <View
                android:layout_width="match_parent"
                android:layout_height="1dp"
                android:layout_marginVertical="4dp"
                android:background="@color/grey"/>

            <androidx.recyclerview.widget.RecyclerView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:id="@+id/otherRecycler"/>

        </LinearLayout>

    </com.google.android.material.card.MaterialCardView>

</LinearLayout>

</androidx.core.widget.NestedScrollView>

```

Faculty Fragment. Java

```

package com.example.doca.ui.faculty;

import android.content.Intent;
import android.os.Bundle;

import androidx.annotation.NonNull;

```

```

import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.LinearLayout;
import android.widget.Toast;

import com.example.doca.R;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;
import java.util.List;

public class facultyFragment extends Fragment {

    private RecyclerView JP, PP, DBMS, Oops, SE, Math, Other;
    private LinearLayout
JpNoData, PpNoData, DBNoData, OopsNoData, SENOData, MathNoData, OtherNoData;
    private List<teacherData> list1, list2, list3, list4, list5, list6, list7;

    private DatabaseReference reference, dbRef;
    private teacherAdapter adapter;

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        View view = inflater.inflate(R.layout.fragment_faculty, container, false);

        JP = view.findViewById(R.id.JP);
        PP = view.findViewById(R.id.PP);
        DBMS = view.findViewById(R.id.DBMS);
        Oops = view.findViewById(R.id.Oops);
        SE = view.findViewById(R.id.SE);
        Math = view.findViewById(R.id.Math);
        Other = view.findViewById(R.id.Other);

        JpNoData = view.findViewById(R.id.JpNoData);
        PpNoData = view.findViewById(R.id.PpNoData);
        DBNoData = view.findViewById(R.id.DBNoData);
        OopsNoData = view.findViewById(R.id.OopsNoData);
        SENOData = view.findViewById(R.id.SENOData);
        MathNoData = view.findViewById(R.id.MathNoData);
        OtherNoData = view.findViewById(R.id.OtherNoData);

        reference = FirebaseDatabase.getInstance().getReference().child("teacher");

```



```

JP();
PP();
DBMS();
Oops();
SE();
Math();
Other();

return view;
}

private void Other() {
    dbRef = reference.child("Other Subjects");
    dbRef.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            list7 = new ArrayList<>();
            if (!snapshot.exists()) {
                OtherNoData.setVisibility(View.VISIBLE);
                Other.setVisibility(View.GONE);
            } else {
                OtherNoData.setVisibility(View.GONE);
                Other.setVisibility(View.VISIBLE);
                for (DataSnapshot dataSnapshot: snapshot.getChildren()) {
                    teacherData data = dataSnapshot.getValue(teacherData.class);
                    list7.add(data);
                }
                Other.setHasFixedSize(true);
                Other.setLayoutManager(new LinearLayoutManager(getContext()));
                adapter = new teacherAdapter(list7, getContext());
                Other.setAdapter(adapter);
            }
        }

        @Override
        public void onCancelled(@NonNull DatabaseError error) {
            Toast.makeText(getContext(), error.getMessage(),
Toast.LENGTH_SHORT).show();
        }
    });
}

private void Math() {
    dbRef = reference.child("Mathematics");
    dbRef.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            list6 = new ArrayList<>();
            if (!snapshot.exists()) {
                MathNoData.setVisibility(View.VISIBLE);

```

```

        Math.setVisibility(View.GONE);
    }else{

        MathNoData.setVisibility(View.GONE);
        Math.setVisibility(View.VISIBLE);
        for (DataSnapshot dataSnapshot:snapshot.getChildren()){
            teacherData data = dataSnapshot.getValue(teacherData.class);
            list6.add(data);
        }
        Math.setHasFixedSize(true);
        Math.setLayoutManager(new LinearLayoutManager(getContext()));
        adapter = new teacherAdapter(list6,getContext());
        Math.setAdapter(adapter);

    }

}

@Override
public void onCancelled(@NonNull DatabaseError error) {
    Toast.makeText(getContext(), error.getMessage(),
Toast.LENGTH_SHORT).show();
}

});
}

private void SE() {
    dbRef = reference.child("Software Engineering");
    dbRef.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            list5 = new ArrayList<>();
            if (!snapshot.exists()){
                SENoData.setVisibility(View.VISIBLE);
                SE.setVisibility(View.GONE);
            }else{

                SENoData.setVisibility(View.GONE);
                SE.setVisibility(View.VISIBLE);
                for (DataSnapshot dataSnapshot:snapshot.getChildren()){
                    teacherData data = dataSnapshot.getValue(teacherData.class);
                    list5.add(data);
                }
                SE.setHasFixedSize(true);
                SE.setLayoutManager(new LinearLayoutManager(getContext()));
                adapter = new teacherAdapter(list5,getContext());
                SE.setAdapter(adapter);

            }

        }

    }

    @Override
    public void onCancelled(@NonNull DatabaseError error) {
        Toast.makeText(getContext(), error.getMessage(),
Toast.LENGTH_SHORT).show();
    }
}

```

```

    }
    });
}

private void Oops() {
    dbRef = reference.child("Object Oriented Programming Using C & C++");
    dbRef.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            list4 = new ArrayList<>();
            if (!snapshot.exists()){
                OopsNoData.setVisibility(View.VISIBLE);
                Oops.setVisibility(View.GONE);
            }else{

                OopsNoData.setVisibility(View.GONE);
                Oops.setVisibility(View.VISIBLE);
                for (DataSnapshot dataSnapshot:snapshot.getChildren()){
                    teacherData data = dataSnapshot.getValue(teacherData.class);
                    list4.add(data);
                }
                Oops.setHasFixedSize(true);
                Oops.setLayoutManager(new LinearLayoutManager(getContext()));
                adapter = new teacherAdapter(list4,getContext());
                Oops.setAdapter(adapter);

            }

        }

        @Override
        public void onCancelled(@NonNull DatabaseError error) {
            Toast.makeText(getContext(), error.getMessage(),
Toast.LENGTH_SHORT).show();
        }
    });
}

private void DBMS() {
    dbRef = reference.child("DBMS");
    dbRef.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            list3 = new ArrayList<>();
            if (!snapshot.exists()){
                DBNoData.setVisibility(View.VISIBLE);
                DBMS.setVisibility(View.GONE);
            }else{

                DBNoData.setVisibility(View.GONE);
                DBMS.setVisibility(View.VISIBLE);
                for (DataSnapshot dataSnapshot:snapshot.getChildren()){
                    teacherData data = dataSnapshot.getValue(teacherData.class);
                    list3.add(data);
                }
            }
        }
    });
}

```

```

        }
        DBMS.setHasFixedSize(true);
        DBMS.setLayoutManager(new LinearLayoutManager(getContext()));
        adapter = new teacherAdapter(list3, getContext());
        DBMS.setAdapter(adapter);
    }

}

@Override
public void onCancelled(@NonNull DatabaseError error) {
    Toast.makeText(getContext(), error.getMessage(),
Toast.LENGTH_SHORT).show();
}
});
}

private void PP() {
    dbRef = reference.child("Python Programming");
    dbRef.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            list2 = new ArrayList<>();
            if (!snapshot.exists()) {
                PpNoData.setVisibility(View.VISIBLE);
                PP.setVisibility(View.GONE);
            } else {
                PpNoData.setVisibility(View.GONE);
                PP.setVisibility(View.VISIBLE);
                for (DataSnapshot dataSnapshot: snapshot.getChildren()) {
                    teacherData data = dataSnapshot.getValue(teacherData.class);
                    list2.add(data);
                }
                PP.setHasFixedSize(true);
                PP.setLayoutManager(new LinearLayoutManager(getContext()));
                adapter = new teacherAdapter(list2, getContext());
                PP.setAdapter(adapter);
            }
        }

    }

}

@Override
public void onCancelled(@NonNull DatabaseError error) {
    Toast.makeText(getContext(), error.getMessage(),
Toast.LENGTH_SHORT).show();
}
});
}

```

```

private void JP() {
    dbRef = reference.child("Java Programming");
    dbRef.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            list1 = new ArrayList<>();
            if (!snapshot.exists()) {
                JpNoData.setVisibility(View.VISIBLE);
                JP.setVisibility(View.GONE);
            } else {
                JpNoData.setVisibility(View.GONE);
                JP.setVisibility(View.VISIBLE);
                for (DataSnapshot dataSnapshot: snapshot.getChildren()) {
                    teacherData data = dataSnapshot.getValue(teacherData.class);
                    list1.add(data);
                }
                JP.setHasFixedSize(true);
                JP.setLayoutManager(new LinearLayoutManager(getContext()));
                adapter = new teacherAdapter(list1, getContext());
                JP.setAdapter(adapter);
            }
        }

        @Override
        public void onCancelled(@NonNull DatabaseError error) {
            Toast.makeText(getContext(), error.getMessage(),
Toast.LENGTH_SHORT).show();
        }
    });
}
}

```

➤ Faculty data

➤ `package com.example.doca.ui.faculty;`

```

public class teacherData {
    private String name, email, post, image, key ;

    public teacherData() {
    }

    public teacherData(String name, String email, String post, String image, String
key) {
        this.name = name;
        this.email = email;
        this.post = post;
        this.image = image;
        this.key = key;
    }

    public String getName() {
        return name;
    }
}

```

```

    }

    public void setName(String name) {
        this.name = name;
    }

    public String getEmail() {
        return email;
    }

    public void setEmail(String email) {
        this.email = email;
    }

    public String getPost() {
        return post;
    }

    public void setPost(String post) {
        this.post = post;
    }

    public String getImage() {
        return image;
    }

    public void setImage(String image) {
        this.image = image;
    }

    public String getKey() {
        return key;
    }

    public void setKey(String key) {
        this.key = key;
    }
}

```

➤ Faculty Adapter

```

package com.example.doca.ui.faculty;

import android.content.Context;
import android.content.Intent;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;

import com.bumptech.glide.Glide;
import com.example.doca.R;

```

```

import java.util.List;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

public class teacherAdapter extends
RecyclerView.Adapter<teacherAdapter.teacherViewHolder> {

    private List<teacherData> list;
    private Context context;
    private String category;

    public teacherAdapter(List<teacherData> list, Context context) {
        this.list = list;
        this.context = context;
    }

    @NonNull
    @Override
    public teacherViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType)
    {
        View view = LayoutInflater.from(context).inflate(R.layout.faculty_item_layout,
parent, false);

        return new teacherViewHolder(view);
    }

    @Override
    public void onBindViewHolder(@NonNull teacherViewHolder holder, int position) {
        teacherData item = list.get(position);
        holder.name.setText(item.getName());
        holder.email.setText(item.getEmail());
        holder.post.setText(item.getPost());
        try {
Glide.with(context).load(item.getImage()).placeholder(R.drawable.avtar).into(holder.imageView);
        } catch (Exception e) {
            e.printStackTrace();
        }

    }

    @Override
    public int getItemCount() {
        return list.size();
    }

    public class teacherViewHolder extends RecyclerView.ViewHolder {

        private TextView name, post, email;

        private ImageView imageView;
    }
}

```

```

    public teacherViewAdapter(@NonNull View itemView) {
        super(itemView);
        name = itemView.findViewById(R.id.teacherName);
        post = itemView.findViewById(R.id.teacherPost);
        email = itemView.findViewById(R.id.teacherEmail);
        imageView = itemView.findViewById(R.id.teacherImg);
    }
}

```

Faculty Fragment. Xml

```

<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".ui.faculty.facultyFragment">

    <androidx.core.widget.NestedScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="vertical">

            <TextView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:text="Java Programming"
                android:gravity="center"
                android:textSize="20sp"
                android:textColor="@color/textColor"
                android:textStyle="bold"
                android:layout_marginTop="16dp"/>

            <androidx.recyclerview.widget.RecyclerView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:id="@+id/JP"
                android:visibility="gone"
                android:nestedScrollingEnabled="false"/>

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="vertical"
                android:id="@+id/JpNoData"

```



```

        android:visibility="gone">

        <include layout="@layout/no_data_found"/>

</LinearLayout>

<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:layout_marginVertical="22dp"
    android:background="#c1c1c1" />

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="0dp"
    android:gravity="center"
    android:text="Python Programming"
    android:textColor="@color/textColor"
    android:textSize="20sp"
    android:textStyle="bold" />

<androidx.recyclerview.widget.RecyclerView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/PP"
    android:visibility="gone"
    android:nestedScrollingEnabled="false"/>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:visibility="gone"

    android:id="@+id/PpNoData">

    <include layout="@layout/no_data_found"/>

    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:layout_marginVertical="22dp"
        android:background="#c1c1c1" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="0dp"
        android:gravity="center"
        android:text="DBMS"
        android:textColor="@color/textColor"
        android:textSize="20sp"
        android:textStyle="bold" />

</LinearLayout>

<androidx.recyclerview.widget.RecyclerView

```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/DBMS"
        android:visibility="gone"
        android:nestedScrollingEnabled="false"/>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:visibility="gone"

    android:id="@+id/DBNoData">

    <include layout="@layout/no_data_found"/>

    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:layout_marginVertical="8dp"
        android:background="#c1c1c1" />

</LinearLayout>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Object Oriented Programming Using C & C++"
    android:gravity="center"
    android:textSize="20sp"
    android:textColor="@color/textColor"
    android:textStyle="bold"
    android:layout_marginTop="14dp"/>

<androidx.recyclerview.widget.RecyclerView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/Oops"
    android:visibility="gone"
    android:nestedScrollingEnabled="false"/>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:visibility="gone"

    android:id="@+id/OopsNoData">

    <include layout="@layout/no_data_found"/>

    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:layout_marginVertical="8dp"
        android:background="#c1c1c1" />

</LinearLayout>

```

```

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Software Engineering"
    android:gravity="center"
    android:textSize="20sp"
    android:textColor="@color/textColor"
    android:textStyle="bold"
    android:layout_marginTop="10dp"/>

<androidx.recyclerview.widget.RecyclerView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/SE"
    android:visibility="gone"
    android:nestedScrollingEnabled="false"/>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:visibility="gone"

    android:id="@+id/SENoData">

    <include layout="@layout/no_data_found"/>

    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:layout_marginVertical="8dp"
        android:background="#c1c1c1" />

</LinearLayout>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Mathematics"
    android:gravity="center"
    android:textSize="20sp"
    android:textColor="@color/textColor"
    android:textStyle="bold"
    android:layout_marginTop="10dp"/>

<androidx.recyclerview.widget.RecyclerView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/Math"
    android:visibility="gone"
    android:nestedScrollingEnabled="false"/>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:visibility="gone"

```

```

        android:id="@+id/MathNoData">

        <include layout="@layout/no_data_found"/>

        <View
            android:layout_width="match_parent"
            android:layout_height="1dp"
            android:layout_marginVertical="8dp"
            android:background="#c1c1c1" />

    </LinearLayout>

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Others"
        android:gravity="center"
        android:textSize="20sp"
        android:textColor="@color/textColor"
        android:textStyle="bold"
        android:layout_marginTop="10dp"/>

    <androidx.recyclerview.widget.RecyclerView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/Other"
        android:visibility="gone"
        android:nestedScrollingEnabled="false"/>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:visibility="gone"
        android:id="@+id/OtherNoData">

        <include layout="@layout/no_data_found" />

        <View
            android:layout_width="match_parent"
            android:layout_height="1dp"
            android:layout_marginVertical="8dp"
            android:background="#c1c1c1" />

    </LinearLayout>

</LinearLayout>

</androidx.core.widget.NestedScrollView>

```

```
</FrameLayout>
```

➤ Faculty item layout.xml

```
<?xml version="1.0" encoding="utf-8"?>
<com.google.android.material.card.MaterialCardView
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="4dp"
    tools:ignore="MissingDefaultResource">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginVertical="@dimen/text_margin"
        android:background="#36FAECD5"

        android:orientation="vertical">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:weightSum="2">

            <LinearLayout
                android:layout_width="0dp"
                android:layout_height="wrap_content"
                android:paddingLeft="12dp"
                android:layout_weight="1.5"
                android:orientation="vertical"
                android:gravity="center_vertical">

                <TextView
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:id="@+id/teacherName"
                    android:text="Sourav Rana"
                    android:textColor="@color/textColor"
                    android:textSize="@dimen/title_size"
                    android:fontFamily="@string/regular"/>

                <TextView
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:id="@+id/teacherPost"
                    android:text="Professor "
                    android:textColor="@color/textColor"
                    android:textSize="@dimen/text_size"
                    android:fontFamily="@string/regular"/>>
            </LinearLayout>
        </LinearLayout>
    </LinearLayout>
</com.google.android.material.card.MaterialCardView>
```

```

        <TextView
            android:id="@+id/teacherEmail"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:fontFamily="@string/regular"
            android:text="souravrana13622@gmail.com"
            android:textColor="@color/textColor"
            android:textSize="@dimen/text_size" />/>

    </LinearLayout>

    <de.hdodenhof.circleimageview.CircleImageView
        android:id="@+id/teacherImg"
        android:layout_width="80dp"
        android:layout_height="80dp"
        android:src="@drawable/avtar"
        android:layout_weight="0.5"
        android:layout_gravity="center_horizontal"/>

</LinearLayout>

</LinearLayout>

</com.google.android.material.card.MaterialCardView>

```

Pdf Activity.java

```

package com.example.doca.pdf;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.os.Bundle;
import android.view.View;
import android.widget.LinearLayout;
import android.widget.Toast;

import com.example.doca.R;
import com.facebook.shimmer.ShimmerFrameLayout;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;
import java.util.List;

```

```

public class EbookActivity extends AppCompatActivity {

    RecyclerView ebookRecycler;
    DatabaseReference reference;
    List<EbookData> list;
    EbookAdapter adapter;

    ShimmerFrameLayout shimmerFrameLayout;
    LinearLayout shimmer_layout;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_ebook);

        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
        getSupportActionBar().setTitle("pdf");

        ebookRecycler = findViewById(R.id.ebookRecycler);
        reference = FirebaseDatabase.getInstance().getReference().child("pdf");

        shimmerFrameLayout = findViewById(R.id.shimmer_view_container);
        shimmer_layout = findViewById(R.id.shimmer_layout);

        getData();
    }

    private void getData() {
        reference.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot snapshot) {
                List<EbookData>list = new ArrayList<>();
                for (DataSnapshot snapshot1 : snapshot.getChildren()){
                    EbookData data = snapshot1.getValue(EbookData.class);
                    list.add(data);
                }
                adapter = new EbookAdapter(EbookActivity.this,list);
                ebookRecycler.setLayoutManager(new
LinearLayoutManager(EbookActivity.this));
                ebookRecycler.setAdapter(adapter);
                shimmerFrameLayout.stopShimmer();
                shimmer_layout.setVisibility(View.GONE);
            }

            @Override
            public void onCancelled(@NonNull DatabaseError error) {
                Toast.makeText(EbookActivity.this,error.getMessage(),
Toast.LENGTH_SHORT).show();
            }
        });
    }

    @Override
    protected void onPause() {
        shimmerFrameLayout.stopShimmer();
    }
}

```

```

        super.onPause();
    }

    @Override
    protected void onResume() {
        shimmerFrameLayout.startShimmer();
        super.onResume();
    }
}

```

➤ Ebook Adapter.java

```

package com.example.doca.pdf;

import android.annotation.SuppressLint;
import android.content.Context;
import android.content.Intent;
import android.net.Uri;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

import com.example.doca.R;

import java.util.List;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

public class EbookAdapter extends RecyclerView.Adapter<EbookAdapter.EbookViewHolder> {

    private final Context context;
    private final List<EbookData> list;

    public EbookAdapter(Context context, List<EbookData> list) {
        this.context = context;
        this.list = list;
    }

    @NonNull
    @Override
    public EbookViewHolder onCreateViewHolder(@NonNull ViewGroup parent, final int
viewType) {

        View view =
LayoutInflater.from(context).inflate(R.layout.ebook_item_layout, parent, false);
        return new EbookViewHolder(view);

    }

    @Override
    public void onBindViewHolder(@NonNull EbookViewHolder holder,
@SuppressLint("RecyclerView") final int position) {

```



```

holder.ebookTextview.setText(list.get(position).getPdfTitle());
holder.ebookDownload.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View view) {
        Intent intent = new Intent(Intent.ACTION_VIEW);
        Uri data = Uri.parse("PdfUrl");
        intent.setData(Uri.parse(list.get(position).getPdfUrl()));
        context.startActivity(intent);
    }
});

holder.itemView.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view){
        Intent intent = new Intent(context, PdfViewer.class);
        intent.putExtra("pdfUrl",list.get(position).getPdfUrl());
        intent.putExtra("pdfTitle",list.get(position).getPdfTitle());

        context.startActivity(intent);
    }
});

}

@Override
public int getItemCount() {
    return list.size();
}

public static class EbookViewHolder extends RecyclerView.ViewHolder {

    TextView ebookTextview;
    ImageView ebookDownload;

    public EbookViewHolder(@NonNull View itemView) {
        super(itemView);

        ebookDownload = itemView.findViewById(R.id.ebookDownload);
        ebookTextview = itemView.findViewById(R.id.ebookTextview);

    }
}
}

```

➤ Ebook Data.java

```
package com.example.doca.pdf;
```

```

public class EbookData {
    private String pdfTitle, pdfUrl;

    public EbookData() {
    }

    public EbookData(String pdfTitle, String pdfUrl) {
        this.pdfTitle = pdfTitle;
        this.pdfUrl = pdfUrl;
    }

    public String getPdfTitle() {
        return pdfTitle;
    }

    public void setPdfTitle(String pdfTitle) {
        this.pdfTitle = pdfTitle;
    }

    public String getPdfUrl() {
        return pdfUrl;
    }

    public void setPdfUrl(String pdfUrl) {
        this.pdfUrl = pdfUrl;
    }
}

```

➤ Pdf Viewer.java

```

package com.example.doca.pdf;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.graphics.drawable.ColorDrawable;
import android.os.AsyncTask;
import android.os.Bundle;
import android.view.MenuItem;

import com.example.doca.R;
import com.github.barteksc.pdfviewer.PDFView;

import java.io.BufferedInputStream;
import java.io.IOException;
import java.io.InputStream;
import java.net.HttpURLConnection;
import java.net.URL;

import javax.net.ssl.HttpURLConnection;

public class PdfViewer extends AppCompatActivity {
    String url;
}

```

```

private PDFView pdfView1;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_pdf_viewer);

    url = getIntent().getStringExtra("pdfUrl");

    pdfView1 = findViewById(R.id.pdfView1);

    new PdfDownload().execute(url);
}

class PdfDownload extends AsyncTask<String, Void, InputStream>{

    @Override
    protected InputStream doInBackground(String... strings) {
        InputStream inputStream = null;

        try {
            URL url = new URL(strings[0]);
            HttpURLConnection urlConnection = (HttpURLConnection)
url.openConnection();

            if(urlConnection.getResponseCode() == 200){
                inputStream = new
BufferedInputStream(urlConnection.getInputStream());
            }

        } catch (IOException e) {
            e.printStackTrace();
            return null;
        }
        return inputStream;
    }

    @Override
    protected void onPostExecute(InputStream inputStream) {
        pdfView1.fromStream(inputStream).load();
    }
}
}

```

Pdf Activity.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".pdf.EbookActivity">

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/ebookRecycler"
        android:layout_width="0dp"
        android:layout_height="0dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="1.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.0" />

    <com.facebook.shimmer.ShimmerFrameLayout
        android:id="@+id/shimmer_view_container"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent">

        <LinearLayout
            android:id="@+id/shimmer_layout"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">

            <include layout="@layout/shimmer_pdf"/>

        </LinearLayout>

    </com.facebook.shimmer.ShimmerFrameLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

➤ Pdf Viewer.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".pdf.PdfViewer">

    <com.github.barteksc.pdfviewer.PDFView
        android:id="@+id/pdfView1"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

➤ Pdf item layout.xml

```
➤ <?xml version="1.0" encoding="utf-8"?>
    <com.google.android.material.card.MaterialCardView
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:elevation="5dp"
    android:layout_height="wrap_content">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_margin="@dimen/text_margin"
            android:gravity="center"
            android:orientation="horizontal">

            <ImageView
                android:id="@+id/ebookName"
                android:layout_width="wrap_content"
                android:layout_height="27dp"
                android:src="@drawable/ic_pdf" />

            <TextView
                android:id="@+id/ebookTextview"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:text="All Material is here"
                android:textSize="@dimen/text_size"
                android:layout_marginLeft="@dimen/text_margin"
                android:textColor="@color/black"
                android:fontFamily="@string/regular"/>

            <ImageView
```

```

        android:id="@+id/ebookDownload"
        android:layout_width="30dp"
        android:layout_height="30dp"
        android:layout_marginLeft="-30dp"
        android:src="@drawable/ic_download" />

    </LinearLayout>

</com.google.android.material.card.MaterialCardView>

```

AndroidManifest.xml

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    package="com.example.doca">

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.DOCA"
        tools:targetApi="31">
        <activity
            android:name=".FullImageView"
            android:exported="false">
            <meta-data
                android:name="android.app.lib_name"
                android:value="" />
        </activity>
        <activity
            android:name=".pdf.PdfViewer"
            android:exported="false">
            <meta-data
                android:name="android.app.lib_name"
                android:value="" />
        </activity>
        <activity
            android:name=".pdf.EbookActivity"
            android:parentActivityName=".MainActivity"
            />

        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

```

```

        </intent-filter>

        <meta-data
            android:name="android.app.lib_name"
            android:value="" />
    </activity>
</application>

<uses-permission android:name="android.permission.INTERNET" />

</manifest>

```

Build.gradle

```

plugins {
    id 'com.android.application'
    id 'com.google.gms.google-services'
}

android {
    namespace 'com.example.doca'
    compileSdk 33

    defaultConfig {
        applicationId "com.example.doca"
        minSdk 21
        targetSdk 33
        versionCode 1
        versionName "1.0"

        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
    }

    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'),
'proguard-rules.pro'
        }
    }
    compileOptions {
        sourceCompatibility JavaVersion.VERSION_1_8
        targetCompatibility JavaVersion.VERSION_1_8
    }
}

dependencies {
    implementation fileTree (dir : "libs",include:["*.jar"])
    implementation 'androidx.appcompat:appcompat:1.6.1'
    implementation 'com.google.android.material:material:1.8.0'
    implementation 'androidx.constraintlayout:constraintlayout:2.1.4'
    implementation 'androidx.navigation:navigation-fragment:2.5.3'
}

```

```
implementation'androidx.navigation:navigation-ui:2.5.3'  
implementation'com.google.android.material:material:1.8.0'  
testImplementation 'junit:junit:4.13.2'  
androidTestImplementation 'androidx.test.ext:junit:1.1.5'  
androidTestImplementation 'androidx.test.espresso:espresso-core:3.5.1'  
implementation 'com.github.smarteist:autoimageslider:1.2.0'  
implementation 'com.github.bumptech.glide:glide:4.15.1'  
implementation 'com.google.firebase:firebase-analytics:21.2.1'  
implementation 'de.hdodenhof:circleimageview:3.0.1'  
implementation 'com.google.firebase:firebase-database:20.1.0'  
implementation 'com.github.barteksc:android-pdf-viewer:3.1.0-beta.1'  
implementation 'com.github.barteksc:android-pdf-viewer:2.8.2'  
implementation 'com.github.chrisbanes:PhotoView:2.3.0'  
implementation 'com.facebook.shimmer:shimmer:0.5.0'  
implementation 'com.github.bumptech.glide:glide:4.0.0'
```

```
}
```


FUTURE OF THE WORK

Many educational companies offer school management apps and other mobile applications for a variety of purposes, making it easy for an organization to keep students updated about school activities. Students have adopted innovative teaching methods via mobile applications. These mobile applications help students solve challenges and study more effectively. Mobile applications are the future of education and are paving the way for its growth

Additional Educational Benefits Of Mobile Applications

There are a few benefits listed below:

- They give the opportunity for discussion.
- The use of mobile devices sparks students' interest in learning, especially those who lacked passion during a traditional presentation.
- They allow for the sharing of information, and for learning without an internet connection.
- They can sort information to facilitate search based on certain criteria.
- They allow for practical work with different types of technologies—students have access to resources, not only about the topic being studied, but also additional knowledge, as well as the possibility to use different types of technologies, allowing them to obtain the appropriate level of education.
- They offer an overview of the different forms of resources (visual and audio educational materials).
- They offer the possibility of development in line with the times.

References

- *GitHub*
- *Google*
- *You tube*