SOURCE CODE

ADMIN

ANDROID MANIFESTS

```
    <?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
  package="com.example.admincollegeapp">
  <uses-permission android:name="android.permission.INTERNET"/>
  <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>

  <application
    android:allowBackup="true"
    android:icon="@drawable/logoadmin"
    android:label="Vitech(Admin)"
    android:roundIcon="@drawable/logoadmin"
    android:supportsRtl="true"
    android:theme="@style/AppTheme">
    <activity android:name=".notice.DeleteNoticeActivity"></activity>
    <activity android:name=".faculty.UpdateTeacherActivity" />
    <activity android:name=".faculty.AddTeacher" />
    <activity android:name=".faculty.UpdateFaculty" />
    <activity android:name=".UploadPdfActivity" />
    <activity android:name=".UploadImage" />
    <activity android:name=".notice.UploadNotice" />
    <activity android:name=".MainActivity">
      <intent-filter>
        <action android:name="android.intent.action.MAIN" />

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

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

</manifest>
```

ADD FACULTY

```
   package com.example.admincollegeapp.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.admincollegeapp.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 csDepartment, itDepartment, electricalDepartment,
mechanicalDepartment;
    private LinearLayout csNoData, itNoData, electricalNoData,mechanicalNoData;
    private List<TeacherData> list1, list2, list3 , list4;
    private TeacherAdapter adapter;

    private DatabaseReference reference, dbRef;
```

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

    csDepartment = findViewById(R.id.csDepartment);
    itDepartment = findViewById(R.id.itDepartment);
    electricalDepartment = findViewById(R.id.electricalDepartment);
    mechanicalDepartment = findViewById(R.id.mechanicalDepartment);

    csNoData = findViewById(R.id.csNoData);
    itNoData = findViewById(R.id.itNoData);
    electricalNoData = findViewById(R.id.electricalNoData);
    mechanicalNoData = findViewById(R.id.mechanicalNoData);


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

    csDepartment();
    itDepartment();
    electricalDepartment();
    mechanicalDepartment();

    fab = findViewById(R.id.fab);

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

private void csDepartment() {
    dbRef = reference.child("Computer Science");
    dbRef.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                list1 = new ArrayList<>();
                if (!dataSnapshot.exists()){
                    csNoData.setVisibility(View.VISIBLE);
                    csDepartment.setVisibility(View.GONE);
                }else {
                    csNoData.setVisibility(View.GONE);
                    csDepartment.setVisibility(View.VISIBLE);
                    for (DataSnapshot snapshot: dataSnapshot.getChildren()){
                        TeacherData data = snapshot.getValue(TeacherData.class);
                        list1.add(data);
                    }
                    csDepartment.setHasFixedSize(true);
                    csDepartment.setLayoutManager(new
LinearLayoutManager(UpdateFaculty.this));
                    adapter = new TeacherAdapter(list1, UpdateFaculty.this, "Computer
Science");
                    csDepartment.setAdapter(adapter);
                }
            }

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


    private void itDepartment() {
        dbRef = reference.child("Information Technology");
        dbRef.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                list2 = new ArrayList<>();
                if (!dataSnapshot.exists()){
                    itNoData.setVisibility(View.VISIBLE);
                    itDepartment.setVisibility(View.GONE);
                }else {
                    itNoData.setVisibility(View.GONE);
                    itDepartment.setVisibility(View.VISIBLE);
                    for (DataSnapshot snapshot: dataSnapshot.getChildren()){
                        TeacherData data = snapshot.getValue(TeacherData.class);
                        list2.add(data);
                    }
                    itDepartment.setHasFixedSize(true);
                    itDepartment.setLayoutManager(new
LinearLayoutManager(UpdateFaculty.this));
                    adapter = new TeacherAdapter(list2, UpdateFaculty.this,"Information
Technology");
                    itDepartment.setAdapter(adapter);
                }
            }

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

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

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

    private void mechanicalDepartment() {
        dbRef = reference.child("Mechanical");
        dbRef.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                list4 = new ArrayList<>();
                if (!dataSnapshot.exists()){
                    mechanicalNoData.setVisibility(View.VISIBLE);
                    mechanicalDepartment.setVisibility(View.GONE);
                }else {
                    mechanicalNoData.setVisibility(View.GONE);
                    mechanicalDepartment.setVisibility(View.VISIBLE);
                    for (DataSnapshot snapshot: dataSnapshot.getChildren()){
                        TeacherData data = snapshot.getValue(TeacherData.class);
                        list4.add(data);
                    }
                    mechanicalDepartment.setHasFixedSize(true);
                    mechanicalDepartment.setLayoutManager(new
LinearLayoutManager(UpdateFaculty.this));
                    adapter = new TeacherAdapter(list4, UpdateFaculty.this, "Mechanical");
                    mechanicalDepartment.setAdapter(adapter);
                }
            }

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

UPLOAD NOTICE

```
    package com.example.admincollegeapp.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.admincollegeapp.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 addImage;
    private ImageView noticeImageView;
    private EditText noticeTitle;
    private Button uploadNoticeBtn;

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

        pd = new ProgressDialog(this);

        addImage = findViewById(R.id.addImage);
        noticeImageView = findViewById(R.id.noticeImageView);
        noticeTitle = findViewById(R.id.noticeTitle);
        uploadNoticeBtn = findViewById(R.id.uploadNoticeBtn);

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

        uploadNoticeBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (noticeTitle.getText().toString().isEmpty()){
                    noticeTitle.setError("Emtpy");
                    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 calForDate = Calendar.getInstance();
        SimpleDateFormat currentDate = new SimpleDateFormat("dd-MM-yy");
        String date = currentDate.format(calForDate.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 aVoid) {
                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);
        }
    }
}
```

UPLOAD PHOTO

```
        package com.example.admincollegeapp;

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.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 imageCategory;
    private CardView selectImage;
    private Button uploadImage;
    private ImageView galleryImageView;

    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_upload_image);
        selectImage = findViewById(R.id.addGalleryImage);
        imageCategory = findViewById(R.id.image_category);
        uploadImage = findViewById(R.id.uploadImageBtn);
        galleryImageView = findViewById(R.id.galleryImageView);

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

        pd = new ProgressDialog(this);

        String [] items = new String[]{"Select Category","College Campus","Unifests","Other
Events"};
        imageCategory.setAdapter(new ArrayAdapter<String>(this,
android.R.layout.simple_spinner_dropdown_item,items));

        imageCategory.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener()
{
            @Override
            public void onItemSelected(AdapterView<?> parent, View view, int position, long id)
{
                category = imageCategory.getSelectedItem().toString();
            }

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

            }
        });

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

        uploadImage.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                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 PDF

```
     package com.example.admincollegeapp;
```

```
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.database.Cursor;
import android.graphics.Bitmap;
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 UploadPdfActivity extends AppCompatActivity {
```

```
private CardView addPdf;
private EditText pdfTitle;
private Button uploadPdfBtn;
private TextView pdfTexView;
private String pdfName, title;

private final int REQ = 1;
private Uri pdfData;
private DatabaseReference databaseReferencere;
private StorageReference storageReference;
String downloadUrl = "";
private ProgressDialog pd;

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

    databaseReferencere = 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);
    pdfTexView = findViewById(R.id.pdfTextView);

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

    uploadPdfBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                title = pdfTitle.getText().toString();
                if (title.isEmpty()){
                    pdfTitle.setError("Empty");
                    pdfTitle.requestFocus();
                }else if (pdfData == null) {
                    Toast.makeText(UploadPdfActivity.this, "Please select pdf",
Toast.LENGTH_SHORT).show();
                }else{
                    uploadPdf();
                }

            }
        });
    }

    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(UploadPdfActivity.this, "Something went wrong",
Toast.LENGTH_SHORT).show();

            }
        });
    }

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

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


databaseReferencere.child("pdf").child(uniqueKey).setValue(data).addOnCompleteListener(
new OnCompleteListener<Void>() {
            @Override
            public void onComplete(@NonNull Task<Void> task) {
                pd.dismiss();
                Toast.makeText(UploadPdfActivity.this, "Pdf uploaded Successfully",
Toast.LENGTH_SHORT).show();
                pdfTitle.setText("");

            }
        }).addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {
                pd.dismiss();
                Toast.makeText(UploadPdfActivity.this, "Failed to upload Pdf",
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 File"),REQ);
    }

    @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 = 
UploadPdfActivity.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();
            }
            pdfTexView.setText(pdfName);

        }
    }
}
```

USER APPLICATION

ANDROID MANIFESTS

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

    <application
        android:allowBackup="true"
        android:icon="@drawable/icon"
        android:label="@string/app_name"

        android:roundIcon="@drawable/icon"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".DeveloperActivity"
            android:parentActivityName=".MainActivity"></activity>
        <activity android:name=".FullImageView" />
        <activity android:name=".ebook.PdfViewerActivity" />
        <activity
            android:name=".ebook.EbookActivity"
            android:parentActivityName=".MainActivity" />
        <activity android:name=".MainActivity">
```

```
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

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

</manifest>
```

HOME MODULE

```
      package com.example.vItECHcollege.ui.home;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;

import androidx.fragment.app.Fragment;

import com.example.viTechcollege.R;
import com.smarteist.autoimageslider.DefaultSliderView;
import com.smarteist.autoimageslider.IndicatorAnimations;
import com.smarteist.autoimageslider.SliderAnimations;
import com.smarteist.autoimageslider.SliderLayout;

public class HomeFragment extends Fragment {

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

        sliderLayout = view.findViewById(R.id.slider);
        sliderLayout.setIndicatorAnimation(IndicatorAnimations.FILL);

sliderLayout.setSliderTransformAnimation(SliderAnimations.SIMPLETRANSFORMATIO
N);
        sliderLayout.setScrollTimeInSec(1);

        setSliderViews();

        map = view.findViewById(R.id.map);
        map.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                openMap();
            }
        });

        return view;
    }

    private void openMap() {
        Uri uri = Uri.parse("geo:0, 0?q=Velammal Institute of Technology,panchetti");
        Intent intent = new Intent(Intent.ACTION_VIEW, uri);
        intent.setPackage("com.google.android.apps.maps");
        startActivity(intent);
    }

    private void setSliderViews() {
        for (int i = 0; i< 4; i++){
            DefaultSliderView sliderView = new DefaultSliderView(getContext());
            switch (i){
                case 0:
                    sliderView.setImageUrl("https://firebasestorage.googleapis.com/v0/b/ViTech-
college-app.appspot.com/o/photos%2Fgal1.jpg?alt=media&token=8c5ad8d9-3569-4f0c-
90c5-27ad60c18cb0");
                    break;
                case 1:
                    sliderView.setImageUrl("https://firebasestorage.googleapis.com/v0/b/ViTech-
college-app.appspot.com/o/photos%2Fgal2.jpg?alt=media&token=a8c9eeb0-76c4-4b37-
aa21-9a0b001886bb");
                    break;
                case 2:
                    sliderView.setImageUrl("https://firebasestorage.googleapis.com/v0/b/ViTech-
college-app.appspot.com/o/photos%2Fgal3.jpg?alt=media&token=5d3e19d9-65ed-459f-
b871-4054651505f3");
                    break;
                case 3:
                    sliderView.setImageUrl("https://firebasestorage.googleapis.com/v0/b/ViTech-
college-app.appspot.com/o/photos%2Fgal4.jpg?alt=media&token=224ea746-fc5c-40df-
867b-023aa5e14c3d");
                    break;
            }
            sliderView.setImageScaleType(ImageView.ScaleType.CENTER_CROP);

            sliderLayout.addSliderView(sliderView);
        }
    }
}
```

NOTICE MODULE

```
package com.example.Vitechcollege.ui.notice;

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

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

import com.bumptech.glide.Glide;
import com.example.ViTechcollege.FullImageView;
import com.example.ViTechcollege.R;

import java.util.ArrayList;

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);
    }

    @Override
    public void onBindViewHolder(@NonNull NoticeViewAdapter holder, final int position)
{

        final NoticeData currentItem = list.get(position);
    holder.deleteNoticeTitle.setText(currentItem.getTitle());
    holder.date.setText(currentItem.getDate());
    holder.time.setText(currentItem.getTime());



    try {
        if (currentItem.getImage() != null) {
            Glide.with(context).load(currentItem.getImage()).into(holder.deleteNoticeImage);
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
   holder.deleteNoticeImage.setOnClickListener(new View.OnClickListener() {
       @Override
       public void onClick(View v) {
           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 deleteNoticeTitle, date, time;
    private ImageView deleteNoticeImage;

    public NoticeViewAdapter(@NonNull View itemView) {
        super(itemView);
```

```
            deleteNoticeTitle = itemView.findViewById(R.id.deleteNoticeTitle);
            deleteNoticeImage = itemView.findViewById(R.id.deleteNoticeImage);
            date = itemView.findViewById(R.id.date);
            time = itemView.findViewById(R.id.time);
        }
    }
}
```

FACULTY MODULE

```
    package com.example.Vitechcollege.ui.faculty;

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.ViTechcollege.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 csDepartment, itDepartment, electricalDepartment,
mechanicalDepartment;
    private LinearLayout csNoData, itNoData, electricalNoData,mechanicalNoData;
    private List<TeacherData> list1, list2, list3 , list4;
    private TeacherAdapter adapter;

    private DatabaseReference reference, dbRef;

    @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);



        csDepartment = view.findViewById(R.id.csDepartment);
        itDepartment = view.findViewById(R.id.itDepartment);
        electricalDepartment = view.findViewById(R.id.electricalDepartment);
        mechanicalDepartment = view.findViewById(R.id.mechanicalDepartment);

        csNoData = view.findViewById(R.id.csNoData);
        itNoData = view.findViewById(R.id.itNoData);
        electricalNoData = view.findViewById(R.id.electricalNoData);
        mechanicalNoData = view.findViewById(R.id.mechanicalNoData);



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

        csDepartment();
        itDepartment();
        electricalDepartment();
        mechanicalDepartment();

        return view;
    }

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

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


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

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

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

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

    private void mechanicalDepartment() {
        dbRef = reference.child("Mechanical");
        dbRef.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                list4 = new ArrayList<>();
                if (!dataSnapshot.exists()){
                    mechanicalNoData.setVisibility(View.VISIBLE);
                    mechanicalDepartment.setVisibility(View.GONE);
                }else {
                    mechanicalNoData.setVisibility(View.GONE);
                    mechanicalDepartment.setVisibility(View.VISIBLE);
                    for (DataSnapshot snapshot: dataSnapshot.getChildren()){
                        TeacherData data = snapshot.getValue(TeacherData.class);
                        list4.add(data);
                    }
                    mechanicalDepartment.setHasFixedSize(true);
                    mechanicalDepartment.setLayoutManager(new
LinearLayoutManager(getContext()));
                    adapter = new TeacherAdapter(list4, getContext());
                    mechanicalDepartment.setAdapter(adapter);
                }
            }

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

GALLERY MODULE

```
    package com.example.ViTechcollege.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 com.example.ViTechcollege.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 GalleryFragment extends Fragment {

    RecyclerView campusRecycler, unifestRecycler  , 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);
        campusRecycler = view.findViewById(R.id.campusRecycler);
        unifestRecycler = view.findViewById(R.id.unifestRecycler);
        otherRecycler = view.findViewById(R.id.otherRecycler);

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

        getCampusImage();

        getUnifestImage();

        getOtherImage();

        return view;
    }
    private void getOtherImage() {
        reference.child("Other Events").addValueEventListener(new ValueEventListener()
{

            List<String> imageList = new ArrayList<>();

            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                for (DataSnapshot snapshot: dataSnapshot.getChildren()){
                    String data = (String) snapshot.getValue();
                    imageList.add(data);
                }

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

            }

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

            }
        });
    }

    private void getCampusImage() {
        reference.child("College Campus").addValueEventListener(new
ValueEventListener() {

            List<String> imageList = new ArrayList<>();

            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                for (DataSnapshot snapshot: dataSnapshot.getChildren()){
                    String data = (String) snapshot.getValue();
                    imageList.add(data);
                }

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

            }

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

            }
        });
    }

    private void getUnifestImage() {
        reference.child("Unifests").addValueEventListener(new ValueEventListener() {

            List<String> imageList = new ArrayList<>();

            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                for (DataSnapshot snapshot: dataSnapshot.getChildren()){
                    String data = (String) snapshot.getValue();
                    imageList.add(data);
                }

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

            }

            @Override
```

```
            public void onCancelled(@NonNull DatabaseError databaseError) {

            }
        });
    }
}
```

EBOOK MODULE

```
 package com.example.ViTechcollege.ui.about;

import android.os.Bundle;

import androidx.fragment.app.Fragment;
import androidx.viewpager.widget.ViewPager;

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

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

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


public class AboutFragment extends Fragment {

    private ViewPager viewPager;
    private BranchAdapter adapter;
    private List<BranchModel>list;

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
```

```
        View view = inflater.inflate(R.layout.fragment_about, container, false);
        list = new ArrayList<>();


        adapter = new BranchAdapter(getContext(), list);

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

        viewPager.setAdapter(adapter);

        ImageView imageView = view.findViewById(R.id.college_image);

        Glide.with(getContext())
                .load("https://firebasestorage.googleapis.com/v0/b/ViTech-college-
app.appspot.com/o/photos%2Fgal1.jpg?alt=media&token=8c5ad8d9-3569-4f0c-90c5-
27ad60c18cb0")
                .into(imageView);

        return view;

    }
}
```