package com.appdroid.tourstravelsproject.Admin;  
  
import androidx.annotation.NonNull;  
import androidx.annotation.Nullable;  
import androidx.appcompat.app.AppCompatActivity;  
  
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.util.Log;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.ImageView;  
import android.widget.Toast;  
  
import com.appdroid.tourstravelsproject.R;  
import com.bumptech.glide.Glide;  
import com.google.android.gms.tasks.OnFailureListener;  
import com.google.android.gms.tasks.OnSuccessListener;  
import com.google.firebase.Timestamp;  
import com.google.firebase.firestore.DocumentReference;  
import com.google.firebase.firestore.FirebaseFirestore;  
import com.google.firebase.firestore.Query;  
import com.google.firebase.storage.FirebaseStorage;  
import com.google.firebase.storage.OnProgressListener;  
import com.google.firebase.storage.StorageReference;  
import com.google.firebase.storage.UploadTask;  
import com.iceteck.silicompressorr.SiliCompressor;  
  
import java.io.ByteArrayOutputStream;  
import java.io.File;  
import java.io.IOException;  
import java.util.Date;  
import java.util.HashMap;  
  
public class AddPackage extends AppCompatActivity {  
  
 EditText packageName, description, edDays, edNights, destination, facility, mainPrice, discountPrice;  
 ImageView image1;  
 Button upload;  
  
 FirebaseFirestore db = FirebaseFirestore.*getInstance*();  
 private Uri filePathUri;  
 ProgressDialog progressDialog;  
 Bitmap thumBitmap = null;  
 byte[] thum\_byte = new byte[0];  
  
 private StorageReference storageReference;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_add\_package*);  
  
 progressDialog = new ProgressDialog(AddPackage.this);  
 progressDialog.setCanceledOnTouchOutside(false );  
  
 storageReference = FirebaseStorage.*getInstance*().getReference();  
  
 packageName = findViewById(R.id.*packageName*);  
 description = findViewById(R.id.*description*);  
 edDays = findViewById(R.id.*edDays*);  
 edNights = findViewById(R.id.*edNights*);  
 destination = findViewById(R.id.*destination*);  
 facility = findViewById(R.id.*facility*);  
 mainPrice = findViewById(R.id.*mainPrice*);  
 discountPrice = findViewById(R.id.*discountPrice*);  
 image1 = findViewById(R.id.*image1*);  
 upload = findViewById(R.id.*upload*);  
  
 image1.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent intent = new Intent();  
 intent.setType("image/\*");  
 intent.setAction(Intent.*ACTION\_GET\_CONTENT*);  
 startActivityForResult(intent,1);  
 }  
 });  
  
 upload.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 String name = packageName.getText().toString();  
 String info = description.getText().toString();  
 String day = edDays.getText().toString();  
 String night = edNights.getText().toString();  
 String location = destination.getText().toString();  
 String activities = facility.getText().toString();  
 String realPrice = mainPrice.getText().toString();  
 String discountedPrice = discountPrice.getText().toString();  
  
 if (!name.isEmpty() && !info.isEmpty() && !day.isEmpty() && !night.isEmpty() && !location.isEmpty() && !activities.isEmpty() && !realPrice.isEmpty() && !discountedPrice.isEmpty()){  
 if (filePathUri != null){  
  
 progressDialog.show();  
 final StorageReference path = storageReference.child("PackagePhoto").child(filePathUri.getLastPathSegment());  
 path.putBytes(thum\_byte).addOnProgressListener(new OnProgressListener<UploadTask.TaskSnapshot>() {  
 @Override  
 public void onProgress(@NonNull UploadTask.TaskSnapshot snapshot) {  
 double progress=(100\* snapshot.getBytesTransferred())/snapshot.getTotalByteCount();  
 int a= (int) progress;  
  
 progressDialog.setMessage("Uploading : "+a+"%");  
 }  
 }).addOnSuccessListener(new OnSuccessListener<UploadTask.TaskSnapshot>() {  
 @Override  
 public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {  
 progressDialog.show();  
 path.getDownloadUrl().addOnSuccessListener(new OnSuccessListener<Uri>() {  
 @Override  
 public void onSuccess(Uri uri) {  
 String downloadUri = String.*valueOf*(uri);  
 HashMap<String, Object> post = new HashMap<>();  
 post.put("packageName",name);  
 post.put("description",info);  
 post.put("days",day);  
 post.put("nights",night);  
 post.put("destination",location);  
 post.put("facility",activities);  
 post.put("mainPrice",realPrice);  
 post.put("discountPrice",discountedPrice);  
 post.put("image1",downloadUri);  
 post.put("date",new Timestamp(new Date()));  
  
 db.collection("TourPackage").add(post).addOnSuccessListener(new OnSuccessListener<DocumentReference>() {  
 @Override  
 public void onSuccess(DocumentReference documentReference) {  
 Toast.*makeText*(getApplicationContext(), "Upload Successful.", Toast.*LENGTH\_SHORT*).show();  
 Intent i = new Intent(AddPackage.this,ViewPackage.class);  
 i.setFlags(Intent.*FLAG\_ACTIVITY\_CLEAR\_TOP*);  
 startActivity(i);  
 finish();  
 progressDialog.dismiss();  
 }  
 }).addOnFailureListener(new OnFailureListener() {  
 @Override  
 public void onFailure(@NonNull Exception e) {  
 progressDialog.dismiss();  
 Toast.*makeText*(getApplicationContext(), "Upload Failed. ", Toast.*LENGTH\_SHORT*).show();  
 }  
 });  
 }  
 });  
 }  
 });  
  
 }else {  
 Toast.*makeText*(AddPackage.this, "Please select a image...", Toast.*LENGTH\_SHORT*).show();  
 }  
 }else{  
 Toast.*makeText*(AddPackage.this, "One of the field is empty...", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 });  
 }  
  
 @Override  
 public void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {  
 super.onActivityResult(requestCode, resultCode, data);  
  
 if (requestCode == 1){  
 if (resultCode == *RESULT\_OK*)  
 {  
  
 filePathUri = data.getData();  
 *//Log.d("TAG", "onCreateView: filePath "+filePathUri);* if (data != null) {  
 Log.*d*("TAG", "data: "+filePathUri);  
 *//File reduceSizeFile = new File(filePathUri.getPath());* try {  
 thumBitmap = MediaStore.Images.Media.*getBitmap*(getContentResolver(), data.getData());  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
  
 ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();  
 thumBitmap.compress(Bitmap.CompressFormat.*JPEG*,75,byteArrayOutputStream);  
 thum\_byte = byteArrayOutputStream.toByteArray();  
  
 Glide.*with*(image1.getContext()).load(filePathUri).into(image1);  
 }  
  
 }else if (resultCode == 1)  
 {  
 Toast.*makeText*(getApplicationContext(), "Failed...", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 }  
  
}