

## Janardhan Bhagat Shikshan Prasarak Sanstha’s

**CHANGU KANA THAKUR ARTS, COMMERCE AND SCIENCE**

## COLLEGE

**NEW PANVEL (Autonomous)**

## PROJECT ON

“SKYFLIGHTS BOOKING APP”

## DEVELOPED BY

**Ms. Vakhardar Siddhi Sanjay**

## UNDER THE GUIDANCE OF

**Prof. Ms. Vrinda Patil**

## ACADEMIC YEAR

### 2022-2023



**Department of Computer Science**

### CERTIFICATE

#### This is to certified that the project entitled

“SKYFLIGHTS BOOKING APP”

Is successfully completed by **Ms. Vakhardar Siddhi Sanjay,** Roll No: 54, Examination No:**BSCCS5062** under the guidance of **Prof. Ms. Vrinda Patil** ,during the academic period of 13th June 2022 to2nd Nov 2022 as per the syllabus, fulfillment for the completion of the BCS degree in the Computer Science. It is also to certify that this is original work of the candidate done during academic year 2022-2023.

**Place:** Panvel

**Date:**

#### Internal Examiner Principal

**External Examiner Head of Department**



# ACKNOWLEDGEMENT

It is indeed a matter of great pleasure and proud privilege to be able to present this project on “**SKYFLIGHTS BOOKING APP**”

I would also like to express my deep regards and the gratitude towards the principal **Prof. Dr. Sanjay Patil.**

I respect and thank Head of the department **Prof. Mrs. Pratibha Jadhav**, for providing me an opportunity to do the project work and giving me all the support and guidance. Also, I would like to tender our sincere thank to all the teachers for their co-operation.

The completion of the project work is a mile stone in student life and it execution is inevitable in the hands of guide. I am highly indebted the projects guide **Prof. Ms. Vrinda Patil** for his invaluable guidance and appreciation for giving form and substance to this report. It is due to his enduring efforts; patience and enthusiasm, which has given a sense of direction and purposefulness to this project and ultimately made it a Success.

I would Wish to thank the non - teaching staff and my friends who have helped me.

Really it is highly impossible to repay the dept of all the people who have me all the time in one way or the other directly or in directly helped me for performing the project.

# INDEX

|  |  |  |
| --- | --- | --- |
| **Sr.No.** | **Title** | **PageNo.** |
| **01** | **Introduction** | **05** |
| **02** | **Existing System** | **07** |
| **03** | Proposed System | **08** |
| **04** | **Requirement Specification** | **09** |
| **05** | **System Design** | **12** |
| **06** | **System Implementation** | **16** |
| **07** | **Results** | **28** |
| **08** | **Conclusion and future Scope** | **35** |
| **09** | **Plagiarism Report** | **38** |
| **10** | **References** | **40** |

**INTRODUCTION**

## Introduction:

* Today, with the digital revolution we can notice a sudden change in customer expectation. Thus, the traditional flight booking system is moving onto the digital platform. This, the same need for digitization has uplifted the demand for fully-featured smart phone apps. Consequently, you are noticing everything accessible more easily than ever.
* Besides, travel [ticket booking](https://www.emizentech.com/blog/event-ticket-booking-mobile-app-development.html) has become so handy today for the customers that traveling all over the world has become a convenient and fun way to handle everything so easily. Every second travel booking business person is trying hard to make their customer experience faster and smoother, and therefore the demand for a flight booking app has increased a lot. In fact, it’s becoming a mandatory facility for business owners to retain customers.
* Skyflight Airline Reservation System is software which is helpful for ticketing manager as well as the customers. In the later system all the activities were done manually. It was very time consuming and costly.
* Our Skyflight Airline Reservation System deals with the various activities related to the Flights.
* There are mainly 3 modules in this software:
* •Book a New Ticket
  + Cancel an existing Ticket
  + Get an existing Ticket
* Flights are booked through Flight Reservation Module in which all the details regarding customer and his flight are entered. A Booking id. is provided to every customer which is unique for each customer and with the help of which cancellation and checking the already existing flight can be done.

### Existing System:

### Earlier no system existed to cater to the needs of ‘Secure Infrastructure Implementation System’. The current system developed is technically feasible. It is a 17 web based user interface for audit workflow at NIC-CSD. Thus, it provides an easy access to the users. The database’s purpose is to create, establish and maintain a workflow among various entities to facilitate all concerned users in their various capacities or roles.

### Permission to the users would be granted based on the roles specified. Therefore, it provides the technical guarantee of accuracy, reliability and security. The software and hardware requirements for the development of this project are not many and are already available in-house at NIC or are available as free as open source. The work for the project is done with the current equipment and existing software technology.

### 

## Proposed System:

* In the proposed system the customer can sit in the comfort of his home and check out the available products. He can check and compare each and every product in less time and with minimum effort. The product will be delivered at your home. Replacing a product or refunding can be done very easily. The customer can get authentic information about the products available in the market. The system allows the airline passenger to search for flights that are available between the two travel cities, namely the “Departure city” and “Arrival city” for a particular departure and arrival dates.
* After search the system display list of available flights and allows customer to choose a particular flight. Then the system checks for the availability of seats on the flight.
* It uses less Data (Internet Data) Resources that web booking.
* It is more secure as App Security is managed indirectly by Android Security.
* Systems while Data Base Security is managed by Google’s Firebase Fire store Security.

**REQUIREMENT**

**SPECIFICATION**

## 4.Requirement Specification:

## 4.1 Hardware Requirements:

|  |  |
| --- | --- |
| Operating system | Windows 10 |
| Processor | Intel i3 |
| System Type | 64 bit operating system |
| Installed Ram | 8 GB |

## 4.2 Software Requirements:

|  |  |
| --- | --- |
| Operating System | Microsoft’s Windows 8/10/11 (64-bit) |
| Programming Language | Kotlin |
| SDK (min) | Android 6.0 (API level 23) |
| Gradle (BUILD VERSION) | 7.2.2 |
| Software (IDE) | Android Studio |
| Dependencies | Firebase Fire store (24.0.0) |

**SYSTEM**

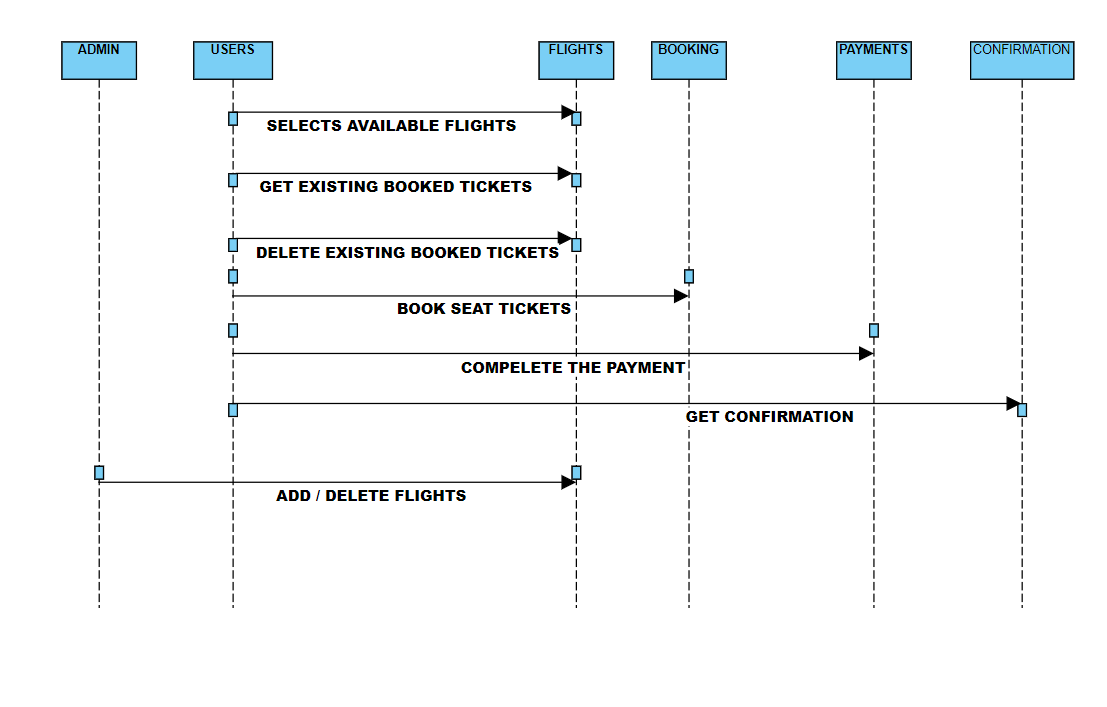
**DESIGN**

## System Design Details:

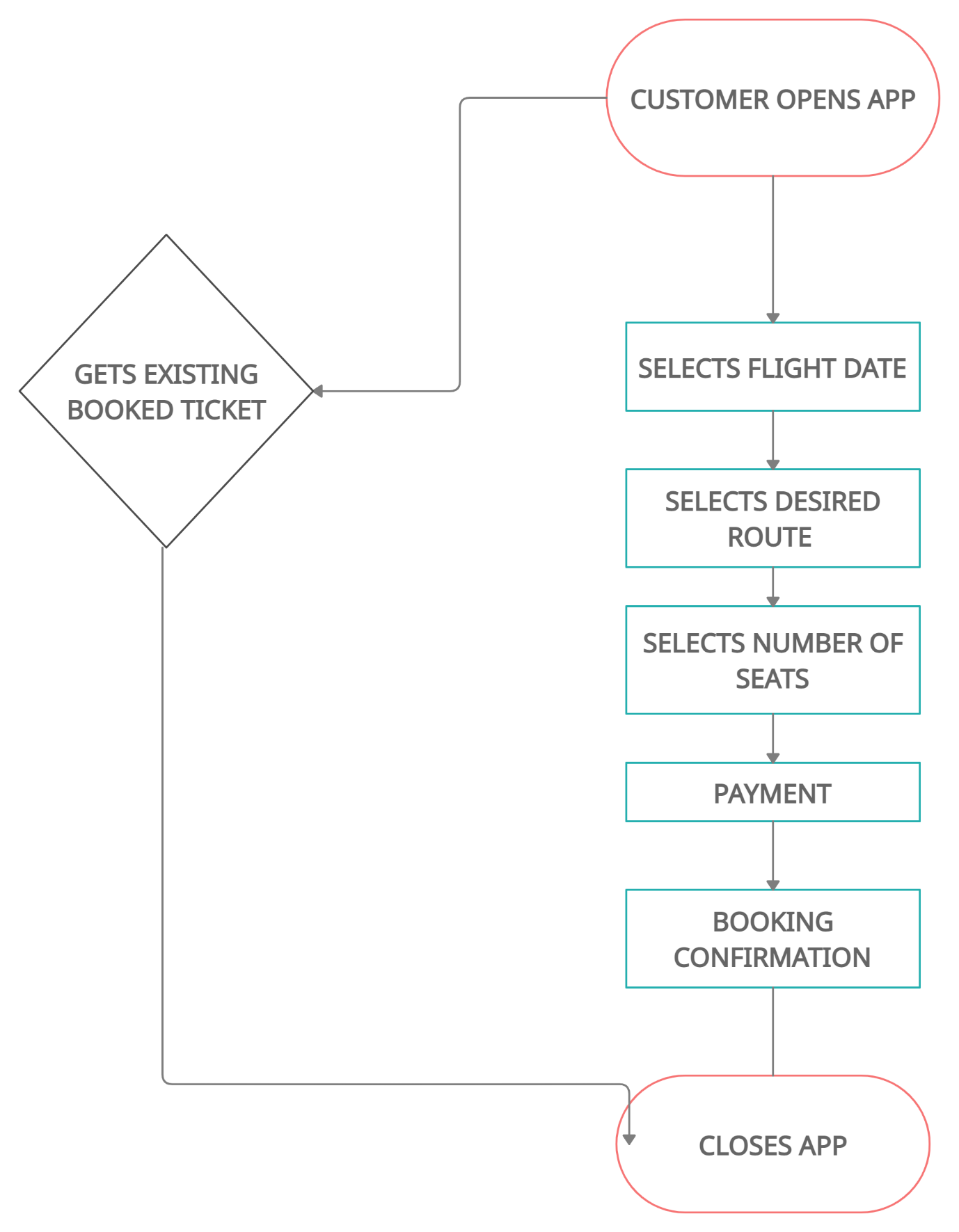
**Use Case Diagram:**

## 

**Sequence Diagram:**

****

**Activity Diagram:**



**SYSTEM**

**IMPLEMENTATION**

## System Implementation:

package com.awesome.skylightflights.MainHelpers;  
  
import android.util.Log;  
  
import javax.activation.DataHandler;  
import javax.activation.DataSource;  
import javax.mail.Message;  
import javax.mail.PasswordAuthentication;  
import javax.mail.Session;  
import javax.mail.Transport;  
import javax.mail.internet.InternetAddress;  
import javax.mail.internet.MimeMessage;  
import java.io.ByteArrayInputStream;  
import java.io.IOException;  
import java.io.InputStream;  
import java.io.OutputStream;  
import java.security.Security;  
import java.security.acl.LastOwnerException;  
import java.util.Properties;  
  
  
  
public class GMailSender extends javax.mail.Authenticator {  
 private String mailhost = "smtp.gmail.com";  
 private String user;  
 private String password;  
 private Session session;  
  
 static {  
 Security.addProvider(new JSSEProvider());  
 }  
  
 public GMailSender(String user, String password) {  
 this.user = user;  
 this.password = password;  
  
 Properties props = new Properties();  
 props.setProperty("mail.transport.protocol", "smtp");  
 props.setProperty("mail.host", mailhost);  
 props.put("mail.smtp.auth", "true");  
 props.put("mail.smtp.port", "465");  
 props.put("mail.smtp.socketFactory.port", "465");  
 props.put("mail.smtp.socketFactory.class",  
 "javax.net.ssl.SSLSocketFactory");  
 props.put("mail.smtp.socketFactory.fallback", "

package com.awesome.skylightflights;  
  
import java.io.Serializable;  
  
public class Bank implements Serializable {  
 public Bank() {  
 }  
  
 public Bank(String no1, String no2) {  
 this.no1 = no1;  
 this.no2 = no2;  
 }  
  
 private String no1, no2;  
  
 public String getNo1() {  
 return no1;  
 }  
  
 public void setNo1(String no1) {  
 this.no1 = no1;  
 }  
  
 public String getNo2() {  
 return no2;  
 }  
  
 public void setNo2(String no2) {  
 this.no2 = no2;  
 }  
  
 public String add(String price, String price2) {  
 return ((Integer.parseInt(price) + Integer.parseInt(price2)) + "");  
 }  
  
 public String sub() {  
 return ((Integer.parseInt(no1) - Integer.parseInt(no2)) + "");  
 }  
}

**Delete\_Flight.java**

package com.awesome.skylightflights;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.util.Log;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Toast;  
  
import com.google.android.gms.tasks.OnCompleteListener;  
import com.google.android.gms.tasks.OnSuccessListener;  
import com.google.android.gms.tasks.Task;  
import com.google.firebase.firestore.DocumentReference;  
import com.google.firebase.firestore.DocumentSnapshot;  
import com.google.firebase.firestore.FirebaseFirestore;  
  
import java.util.HashMap;  
import java.util.Map;  
  
public class Delete\_Flight extends AppCompatActivity {  
  
 private EditText d\_d,d\_m,d\_y,flight;  
 private Button delete;  
 FirebaseFirestore db = FirebaseFirestore.getInstance();  
 DocumentReference flightRef;  
 private DocumentReference bankRef;  
  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_delete\_\_flight);  
  
 d\_d = findViewById(R.id.d\_day);  
 d\_m = findViewById(R.id.name\_delete);  
 d\_y = findViewById(R.id.d\_year);  
 flight = findViewById(R.id.g\_booking\_id);  
  
 delete = findViewById(R.id.delete\_ticket);  
  
 delete.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 int day,month,year,flight\_int;  
 try  
 {  
 day = Integer.parseInt(d\_d.getText().toString().trim());  
 month = Integer.parseInt(d\_m.getText().toString().trim());  
 year = Integer.parseInt(d\_y.getText().toString().trim());  
 Integer.parseInt(flight.getText().toString().trim());  
 flightRef = db.collection(day+"."+month+"."+year).document(flight.getText().toString().trim());  
 flightRef.get().addOnCompleteListener(new OnCompleteListener<DocumentSnapshot>() {  
 @Override  
 public void onComplete(@NonNull Task<DocumentSnapshot> task) {  
 if (task.isSuccessful()) {  
 DocumentSnapshot document = task.getResult();  
 if (document.exists())  
 {  
 int price = Integer.parseInt(document.getString("price"));  
 Map<String,Object> map = new HashMap<>();  
 map.put("status","Cancelled");  
 flightRef.update(map)  
 .addOnSuccessListener(new OnSuccessListener<Void>()  
 {  
 @Override  
 public void onSuccess(Void aVoid)  
 {  
 Toast.makeText(Delete\_Flight.this,"Flight Successfully Deleted.",Toast.LENGTH\_LONG).show();  
 // mail to be sent if booking is deleted.... This feature can be added but Right Now it will not affect the completeness of the program it after all just an extra feature  
 }  
 });  
 int c = 0;  
 for(int i=1;i<=20;i++)  
 if(!document.getString(i+"").equals("0"))  
 c++;  
 Log.d("Price",price+"");  
 int change\_price = c \* price;  
  
 final Bank bank = new Bank();  
 bank.setNo2(change\_price+"");  
 bankRef = db.collection("Bank").document("bank");  
 bankRef.get()  
 .addOnSuccessListener(new OnSuccessListener<DocumentSnapshot>() {  
 @Override  
 public void onSuccess(DocumentSnapshot documentSnapshot) {  
 bank.setNo1(documentSnapshot.getString("amount"));  
 String new\_amount = bank.sub();  
 Map<String, Object> map1 = new HashMap<>();  
 map1.put("amount", new\_amount);  
 bankRef.update(map1);  
 }  
 });  
 Intent intent = new Intent(Delete\_Flight.this,deleting\_flight\_process.class);//deleting\_flight\_process.class  
 }  
 else  
 {  
 Toast.makeText(Delete\_Flight.this,"No Such Flight Exist",Toast.LENGTH\_LONG).show();  
 }  
 } else {  
 Log.d("Failed Deleting Flight", "Failed with: ", task.getException());  
 }  
 }  
 });  
 }  
 catch(Exception e)  
 {  
 Toast.makeText(Delete\_Flight.this,"Enter Data Correctly",Toast.LENGTH\_LONG).show();  
 }  
 }  
 });  
 }  
}

**Delete\_Flight\_Ticket.java**

package com.awesome.skylightflights;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.util.Log;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Toast;  
  
import com.google.android.gms.tasks.OnCompleteListener;  
import com.google.android.gms.tasks.OnSuccessListener;  
import com.google.android.gms.tasks.Task;  
import com.google.firebase.firestore.DocumentReference;  
import com.google.firebase.firestore.DocumentSnapshot;  
import com.google.firebase.firestore.FirebaseFirestore;  
  
import java.util.HashMap;  
import java.util.Map;  
  
public class Delete\_Flight\_Ticket extends AppCompatActivity {  
  
 private EditText name,booking\_id;  
 private Button delete\_ticket;  
 FirebaseFirestore db = FirebaseFirestore.getInstance();  
 DocumentReference bookingRef;  
 DocumentReference emailRef;  
 DocumentReference bankRef;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_delete\_\_flight\_\_ticket);  
  
 name = findViewById(R.id.name\_delete);  
 booking\_id = findViewById(R.id.g\_booking\_id);  
  
 delete\_ticket = findViewById(R.id.delete\_ticket);  
  
 delete\_ticket.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 int id;  
 try  
 {  
 id = Integer.parseInt(booking\_id.getText().toString().trim());  
 bookingRef = db.collection(booking\_id.getText().toString().trim()).document(name.getText().toString().trim());  
 emailRef = db.collection(booking\_id.getText().toString().trim()).document("email");  
 bankRef = db.collection("Bank").document("bank");  
 final Passenger passenger = new Passenger();  
 bookingRef.get().addOnCompleteListener(new OnCompleteListener<DocumentSnapshot>()  
 {  
 @Override  
 public void onComplete(@NonNull Task<DocumentSnapshot> task) {  
 if (task.isSuccessful()) {  
 final DocumentSnapshot document = task.getResult();  
 if (document.exists())  
 {  
 if(document.getString("status").equals("Cancelled"))  
 Toast.makeText(Delete\_Flight\_Ticket.this,"Ticket Already Cancelled",Toast.LENGTH\_LONG).show();  
 else  
 {  
 final Bank bank = new Bank();  
 bank.setNo2(document.getString("price"));  
 Map<String, Object> map = new HashMap<>();  
 map.put("status", "Cancelled");  
 bookingRef.update(map)  
 .addOnSuccessListener(new OnSuccessListener<Void>() {  
 @Override  
 public void onSuccess(Void aVoid) {  
 Toast.makeText(Delete\_Flight\_Ticket.this, "Ticket Successfully Cancelled.", Toast.LENGTH\_LONG).show();  
 // Intent intent = new Intent(Delete\_Flight\_Ticket.this, Menu.class);  
 // startActivity(intent);  
 }  
 });  
// bankRef.get()  
// .addOnSuccessListener(new OnSuccessListener<DocumentSnapshot>() {  
// @Override  
// public void onSuccess(DocumentSnapshot documentSnapshot) {  
// bank.setNo1(documentSnapshot.getString("amount"));  
// String new\_amount = bank.sub();  
// Map<String, Object> map1 = new HashMap<>();  
// map1.put("amount", new\_amount);  
// bankRef.update(map1);  
// }  
// });  
// bookingRef.get()  
// .addOnSuccessListener(new OnSuccessListener<DocumentSnapshot>() {  
// @Override  
// public void onSuccess(DocumentSnapshot documentSnapshot) {  
// String date = ""+documentSnapshot.getLong("day") + "." + documentSnapshot.getLong("month") + "." + documentSnapshot.getLong("year");  
// String flight = "" + documentSnapshot.getString("flight");  
// String seat = document.getString("seat");  
// Map<String,Object> map2 = new HashMap<>();  
// map2.put(seat,"0");  
// db.collection(date).document(flight).update(map2);  
// }  
// });  
 final email emails = new email();  
 bookingRef.get()  
 .addOnSuccessListener(new OnSuccessListener<DocumentSnapshot>()  
 {  
 @Override  
 public void onSuccess(DocumentSnapshot documentSnapshot) {  
 emails.setBody("Dear " + documentSnapshot.getString("name") + ",\n" + "Your flight ticket for flight number :- "  
 + documentSnapshot.getString("flight") + "\n Booking Id : " + documentSnapshot.getString("bookingId")  
 + "\n From : " + documentSnapshot.getString("from") + "\n To : "  
 + documentSnapshot.getString("to") + "\n Seat : " + documentSnapshot.getString("seat") + "\non " + documentSnapshot.getLong("day")  
 + "-" + documentSnapshot.getLong("month") + "-" + documentSnapshot.getLong("year") + " at "  
 + documentSnapshot.getString("dep\_time").substring(0, 2) + ":"  
 + documentSnapshot.getString("dep\_time").substring(2) + " has been successfully cancelled. " +  
 "Your balance will be credited back to your account within 5 working days." +  
 "\nThanks for using Skylights Flights App.\n\nRegards,\nSkylight Airlines.");  
 emailRef.get()  
 .addOnSuccessListener(new OnSuccessListener<DocumentSnapshot>() {  
 @Override  
 public void onSuccess(DocumentSnapshot documentSnapshot) {  
 emails.setEmail(documentSnapshot.getString("email"));  
 Log.d("check",emails.getBody() + emails.getEmail());  
 Intent intent = new Intent(Delete\_Flight\_Ticket.this,review\_details.class);  
 intent.putExtra("emails",emails);  
 startActivity(intent);  
 }  
 });  
  
 }  
 });  
 }  
 }  
 else  
 {  
 Toast.makeText(Delete\_Flight\_Ticket.this,"No Such Booking Exist",Toast.LENGTH\_LONG).show();  
 }  
 } else {  
 Log.d("Failed Deleting Booking", "Failed with: ", task.getException());  
 }  
 }  
 });  
  
 }  
 catch(Exception e)  
 {  
 Toast.makeText(Delete\_Flight\_Ticket.this,"Enter Data Correctly",Toast.LENGTH\_LONG).show();  
 }  
 }  
 });  
 }  
  
}

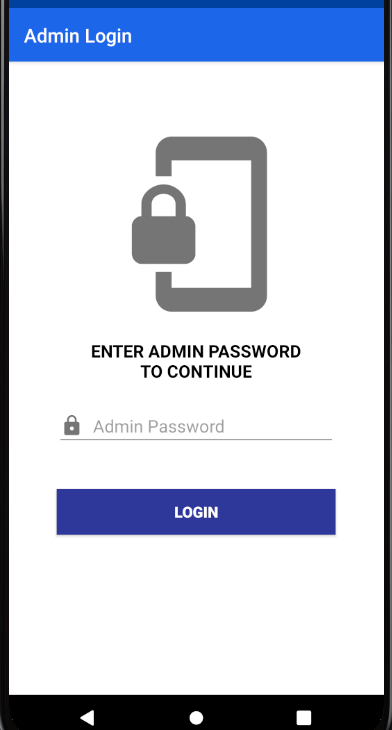
package com.awesome.skylightflights;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.app.ProgressDialog;  
import android.content.Intent;  
import android.os.Bundle;  
import android.util.Log;  
  
import com.awesome.skylightflights.MainHelpers.GMailSender;  
import com.google.android.gms.tasks.OnSuccessListener;  
import com.google.firebase.firestore.DocumentReference;  
import com.google.firebase.firestore.DocumentSnapshot;  
import com.google.firebase.firestore.FirebaseFirestore;  
  
import org.w3c.dom.Document;  
  
import java.util.HashMap;  
import java.util.Map;

**RESULT**

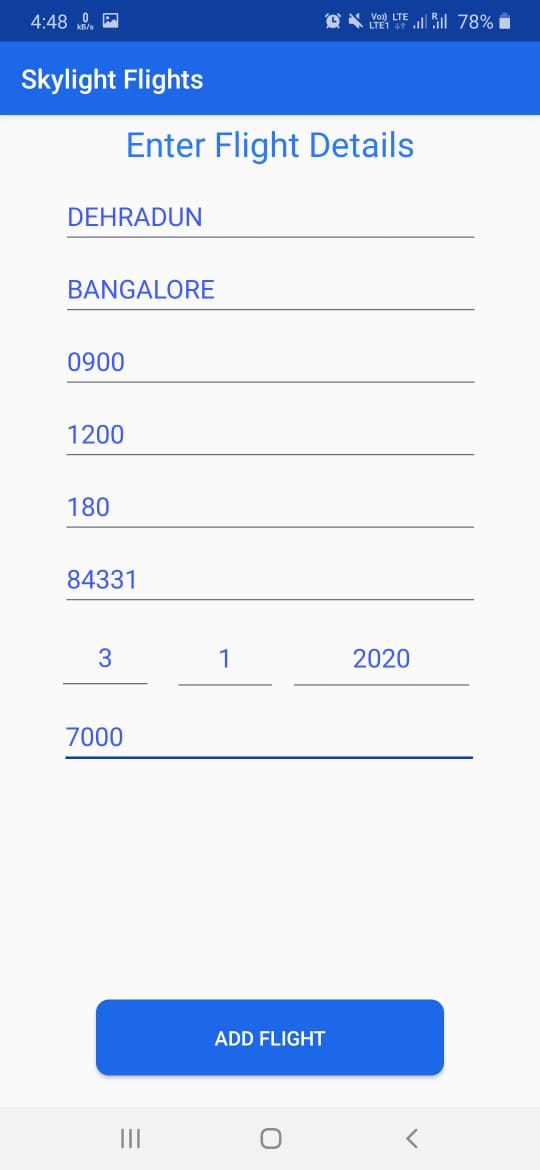
## Start Screen

## 

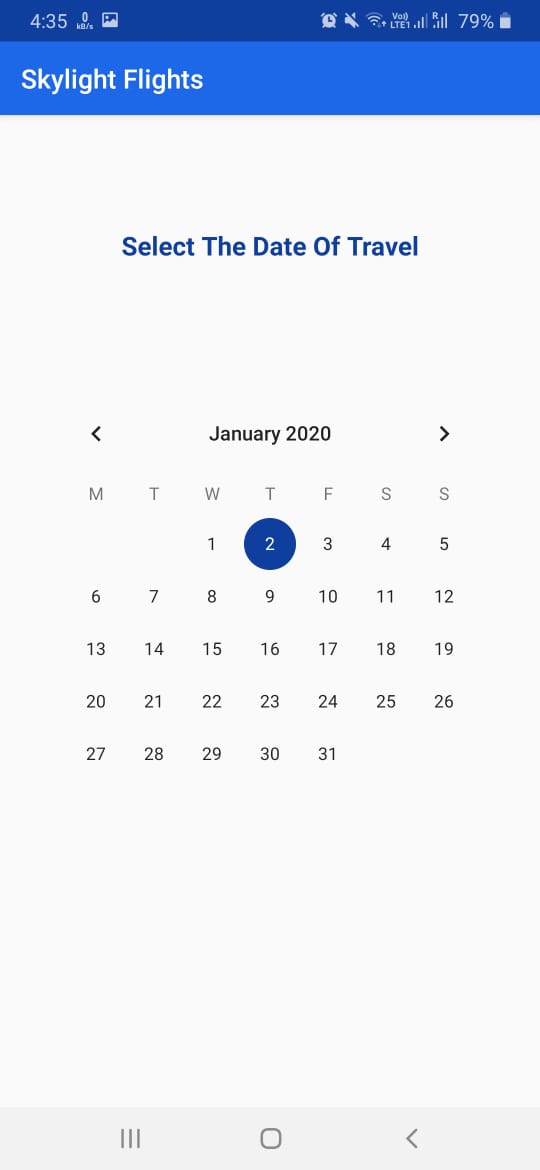
**Admin Login**

****

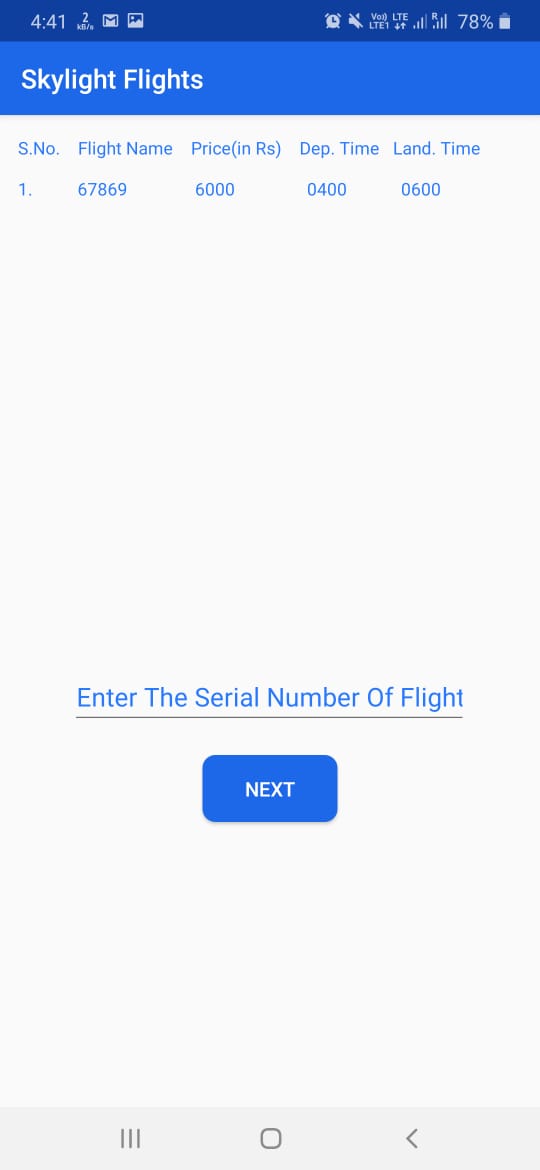
**Admin Adding New Flight**



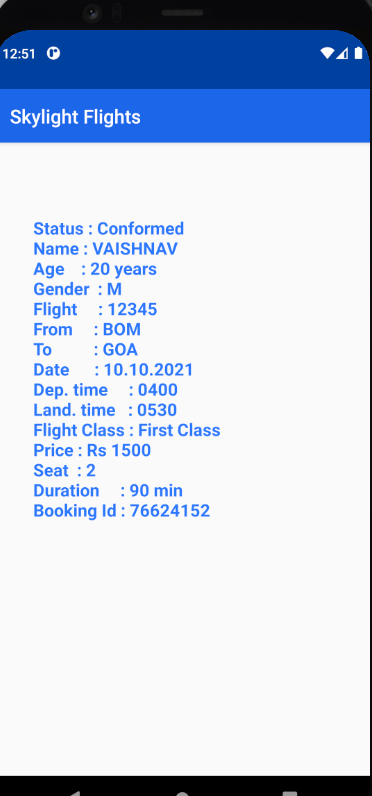
**Date Adding:**



**Display of Available Flights:**



**Getting a Existing Ticket:**



**CONCLUSION &**

**FUTURE SCOPE**

## Conclusion :

* Flight reservation system comprises of an online flight booking software which is designed with the main intension of generating an automated system for ticketing purchase, with the help of an online booking system which would be easy to use.
* An online flight reservation system for your mobile application would help in an easy management of reservation, ticket availability and data of the client who are using online flight reservation service.
* The customized features that a travel management company can add to a flight reservation system includes route scheduling, display the data in regards to availability of the seats, availability of the seat map for the travelers and the option for the travelers to select their seat from the seat map.

**Future Scope:**

* With the number of travelers increasing with each passing years, the demand for a well-equipped flight reservation service has increased. flight travel has gained a very eminent position when it comes to travelling within the country and even globally .
* Flight reservation system has become an important part of the travel technology and travel portal development initiatives.
* Travel portal development companies are working towards the development of flight reservation system and mobile apps, which are user-friendly for the travelers and a source of revenue generation for the travel management companies.
* Our responsive bus booking system with mobile apps (Android), is specially customized for travel management companies, destination management companies, travel aggregators, business 2 business (B2B), business 2 customer’s (B2C) travel agencies and tour operators to cater global customers.

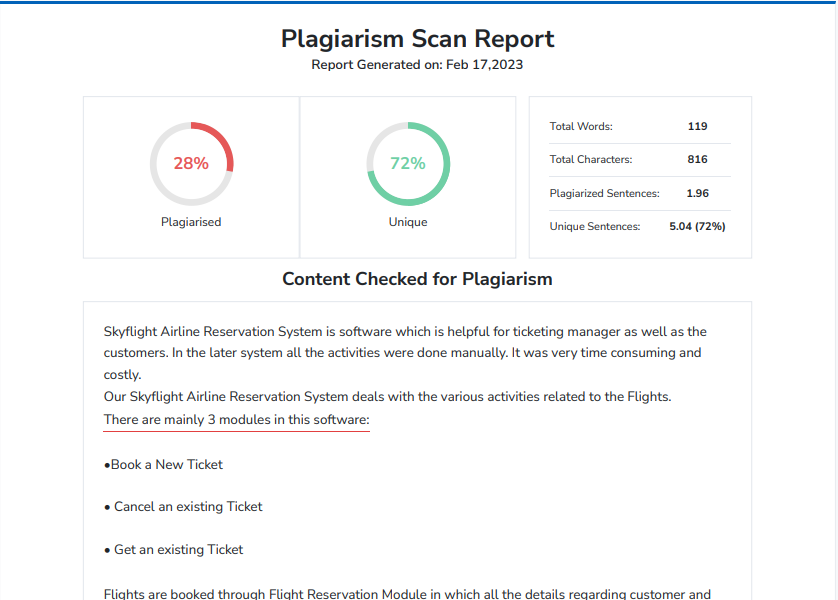
# 

# 

**PLAGIARISM**

**REPORT**

# 



**REFERENCES**

## References:

[1]Kamarulazhar Daud “Sky flight app using andriod studio”

https://iopscience.iop.org/article/10.1088/1757-899X/1045/1/012043

September 07 2022.

[2] Syed Abdul Gaffar Shakhadri “Sky flight app for Firebase” Blog

https://www.analyticsvidhya.com/blog/2021/07/building-a-hand-tracking-system-using-opencv/ July 08 2022

### Websites:

* <https://towardsdatascience.com/sign-language-recognition-using-deep-learning-6549268c60bd>
* <https://ieeexplore.ieee.org/document/7916786>
* <https://stackoverflow.com/>