Nile University

Course name: CSCI313

Project title : Airline Reservation System

Team member:

Nada Radwan 211001128

Sama Ayman Zaki 211001876

Ahmed Amin 211002251

Mohamed Farag 211001654

Ahmed Moussa 211001639

Table of content

1.1 Purpose of the Document

The purpose of this document is to define the requirements and specifications for the development of the airline reservation application, which will be implemented using Flutter. It serves as a comprehensive guide for all stakeholders involved in the project, ensuring a common understanding of the system's objectives and functionality.

1.2 Scope of the Project

The airline reservation application aims to provide a user-friendly platform for customers to browse, select, and book flights, as well as for administrators to manage flight data and reservations. The scope of the project includes features such as flight search, passenger information management, booking, and payment processing.

1.3 Technologies Used

The application will be developed using the following technologies:

- Flutter: The primary programming language for the application.
- Flutter: Used for building the graphical user interface (GUI) to enhance user interaction.
- Relational Database (e.g., MySQL): For data storage and retrieval.
- Web Services: To access real-time flight data and services.

1.4 Intended Audience

This document is intended for various stakeholders, including:

- Development Team: Developers and programmers responsible for creating the application.
- Project Managers: Those overseeing the project's development and progress.
- Quality Assurance Team: Responsible for testing and ensuring the application's quality.
- Client/Project Sponsors: Those funding and requesting the project.
- End Users: Customers who will use the airline reservation system.

1.5 Overview of the Document

The document is structured into several chapters, each addressing a specific aspect of the airline reservation application. It will provide a detailed description of functional and non-functional requirements, use cases, system architecture, and data flow, among other critical aspects of the project. It will serve as a reference point for design, development, testing, and project management throughout the application's lifecycle

2 Overall Description

Airlo will be an Mobile application that provides perfect Airline Reservation System that provides the best tickets for passengers to choose from and get the perfect travel experience.

The system has two sign up options, as passengers user or to continue on as an airline staff, both have different features and structures.

2.1 Product Perspective:

- **Airline Systems Integration:** The application interfaces with airline databases to fetch real-time flight information, availability, and pricing.
- **Payment Gateways:** Integration with secure payment gateways for processing transactions and ensuring financial transactions are smooth and secure.
- **User Authentication:** Integration with user authentication systems to ensure secure access to user accounts and protect personal information.

2.2 Product Function:

2.2.1 Flight Search and Selection:

- **Real-time Flight Information:** Users can search for flights based on various criteria such as destination, departure date, and class
- **Flight Availability:** Display of available flights from different companies, along with details like departure and arrival times, stopovers, and pricing.
- **Seat Selection:** Users can view the seating layout acxnd choose their preferred seats when booking.

2.2.2 Passenger Information Management:

- **User Profiles:** Users can create and manage profiles, storing personal information, preferences, and past booking history.
- **Passenger Details:** Ability to add and manage passenger details for multiple travelers, including special requirements and preferences.

2.2.3 Booking Process:

- **Booking Confirmation:** Users can confirm and finalize their flight selection, review booking details, and receive a confirmation with a unique booking reference.

2.2.4 Payment Processing:

- **Secure Transactions:** Integration with secure payment gateways for processing payments using various methods (credit cards, online banking, etc.).
- **Payment Confirmation:** Immediate confirmation of successful payment transactions and issuance of electronic tickets.

2.2.5 Administrative Functions:

- **Flight Management:** Administrators can add, edit, or remove flights, update schedules, and manage seat availability.
- **Reservation Management:** Access to a dashboard to view and manage customer reservations, handle cancellations, and resolve booking-related issues.

2.3 User Characteristics:

- The user should have basic knowledge of how to use android phones.
- The user should have accessibility to internet connection.
- Administrators have in-depth knowledge and use the system regularly for flight management.

2.4 Constraints:

- Sign-in and password will be required to identify users and Administrators.

- The phone will have to be connected to internet for the application to work, otherwise the user won’t be able to complete the reservation.

2.5 Assumptions and Dependencies:

- The phone that the user will be using is expected to have a GPS-Navigator since it is crucial to be able to access the user’s location to provide you with the nearest flight to your location.
- Database Management System: A reliable database system to store and retrieve flight data, user information, and booking details.

3 Functional Requirements

3.1 User Class 1: The Passenger

3.1.1 Functional Requirements

Title: User Registration

Description: The system will enable passengers to create an account to facilitate future logins.
Required Information:

- Full Name
- Password (at least 8 characters)
- Email (unique and not registered)
- Phone number
- Address
- Passport information
- Photo (optional)

3.1.2 Functional Requirements

Title: Login

Description: The system will allow passengers to log in with a valid username and password.

3.1.3 Functional Requirements

Title: Search Flights

Description: The system will enable passengers to search for available flights based on specified criteria.

Required Information:

- Departure city
- Destination city
- Travel date
- Number of passengers

3.1.4 Functional Requirements

Title: Book a Flight

Description: The system will allow passengers to book a selected flight.

Required Information:

- Passenger details for all travelers
- Payment information

3.1.5 Functional Requirements

Title: View Booking History

Description: The system will allow passengers to view their past flight bookings.

3.1.6 Functional Requirements

Title: Cancel Reservation

Description: The system will allow passengers to cancel a booked flight within a specified timeframe.

3.1.7 Functional Requirements

Title: Modify Reservation

Description: The system will enable passengers to modify their existing flight reservations.

3.1.8 Functional Requirements

Title: Receive Booking Confirmation

Description: The system will send a confirmation email to the passenger upon successful booking.

3.1.9 Functional Requirements

Title: View Flight Details

Description: The system will provide detailed information about selected flights, including timings, layovers, and airlines.

3.1.10 Functional Requirements

Title: Notify User

Description: The system will notify the user of any changes to their flight status, such as delays or cancellations.

3.1.11 Functional Requirements

Title: Seat Selection

Description: The system will allow passengers to choose their seats during the booking process.

3.2 User Class 2: Airline Staff

3.2.1 Functional Requirements

Title: Staff Login

Description: The system will allow airline staff to log in with valid credentials.

3.2.2 Functional Requirements

Title: Manage Flights

Description: The system will enable staff to add, update, or cancel flights.

3.2.3 Functional Requirements

Title: Check-in Passengers

Description: The system will allow staff to check in passengers for their flights.

3.2.4 Functional Requirements

Title: Handle Reservations

Description: The system will allow staff to assist passengers with reservations, modifications, and cancellations.

3.2.5 Functional Requirements

Title: View Passenger Information

Description: The system will provide staff with access to passenger details and booking history.

3.2.6 Functional Requirements

Title: Issue Boarding Pass

Description: The system will allow staff to issue boarding passes to checked-in passengers.

3.2.7 Functional Requirements

Title: Monitor Flight Status

Description: The system will provide real-time updates on the status of each flight.

3.2.8 Functional Requirements

Title: Handle Customer Inquiries

Description: The system will support staff in responding to customer inquiries and helping**.**

4: Non-Functional Requirements

4.1 Performance:

- The application should respond to user requests within 2 seconds.
- Support a minimum of 1000 concurrent users.

4.2 Security:

- Encrypt sensitive data, such as user credentials and payment information.
- Implement access control to ensure that administrators can only access relevant data.

4.3 Usability:

- The user interface should be intuitive and user-friendly to cater to a wide range of users.
- Provide online help and guidance.

4.4 Reliability:

- Ensure the system is available 99.9% of the time.
- Regularly back up data to prevent data loss.

4.5 Scalability:

- Design the system to handle a growing number of customers and flights.

5 System Interface:

When the user open the app in the beginning, there will be a starting page as shown in figure (1) , Then it will open a sign in page to sign in with your account as shown in figure (2), if you don’t have one you will select create account and it opens create account page as shown in figure (3), the user must enter his First name, second name , username , email address, password and confirm password.

Figure (1)

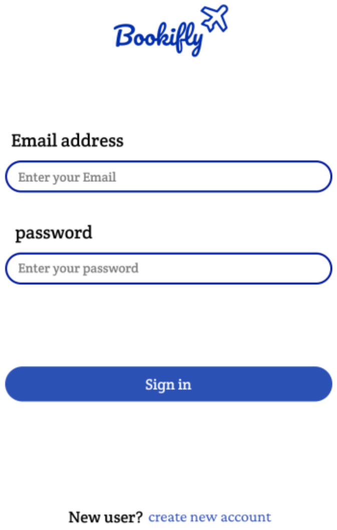
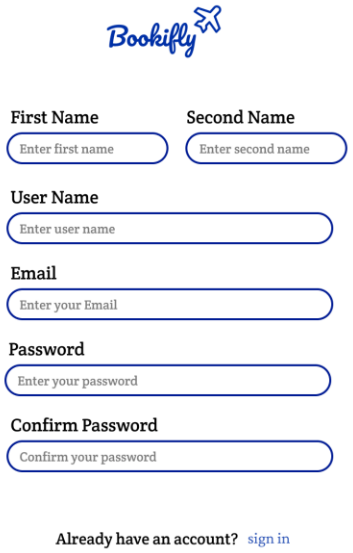
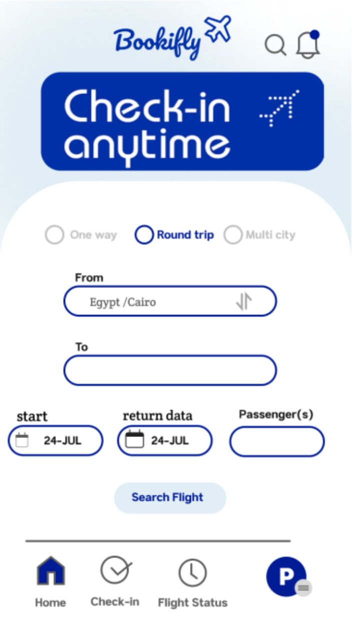


Figure (2)



Figure(3)

In the landing page the user have an option to go to check in page or search for flights, by selecting the destination and the date, also it have an options if it is one way or round trip as shown in figure (4), After searching for flight it will open a new page contains all the flights available for booking as shown in figure (5), the user can select the suitable time then it will open a page contains the price as shown in figure (6)



From: Egypt /Cairo

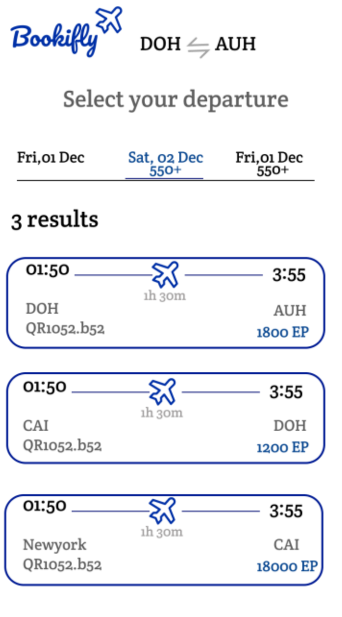
To:

start: 24-JUL

return data: 24-JUL

Passenger(s):

Figure (4)



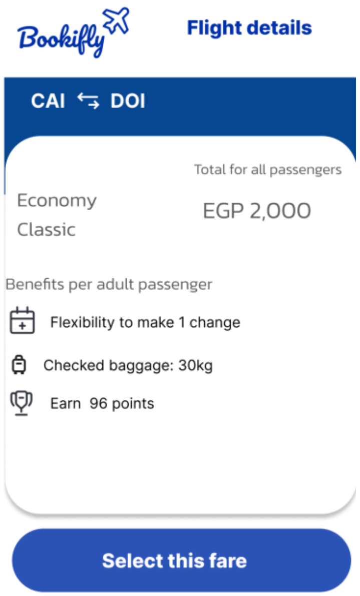
3 results

01:50 — 1h 30m — 3:55
DOH — AUH
QR1052.b52 — 1800 EP

01:50 — 1h 30m — 3:55
CAI — DOH
QR1052.b52 — 1200 EP

01:50 — 1h 30m — 3:55
Newyork — CAI
QR1052.b52 — 18000 EP

Figure (5)



Economy Classic — Total for all passengers: EGP 2,000

Benefits per adult passenger

Flexibility to make 1 change

Checked baggage: 30kg

Earn 96 points

Figure (6)

The payment page contains the passport number , full name , date of birth , date of issue , date of expiary, phone number and email address as shown in figure (7), Then the user will need to add his credit card details in order to complete payment as shown in figure (8) ,then it will show Payment successfully as shown in figure (9)

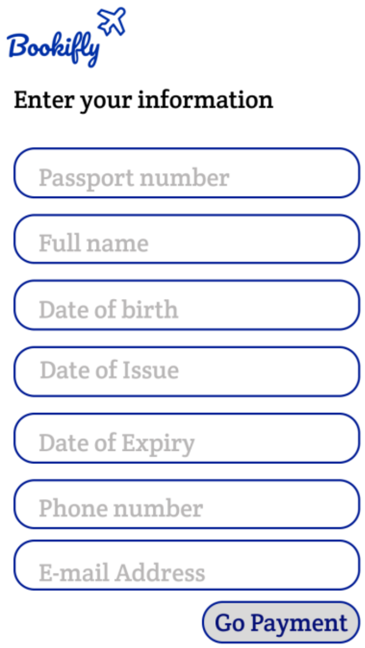


Figure (7)

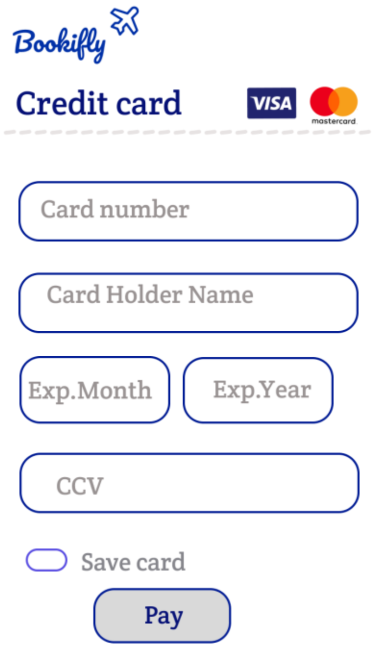
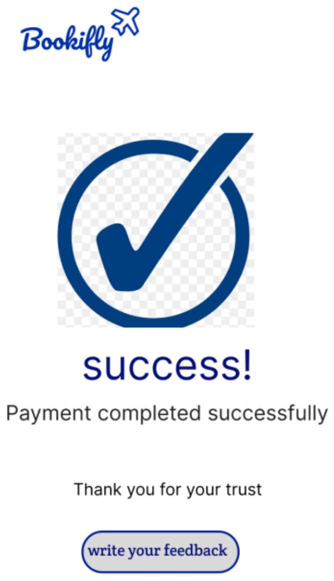


Figure (8)



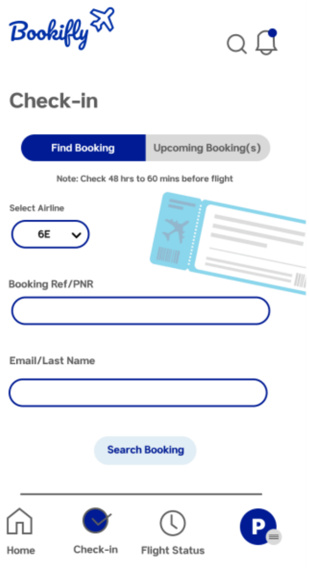
success!

Payment completed successfully

Thank you for your trust

Figure (9)

In the check in page the user Select Airline then enter the Booking Ref/PNR and the Email address / Last name as shown in figure (10), t it will show check in successfully as shown in figure (11), Also he can view the flight status by entering the date and the flight number as shown in Figure (12)



Check-in

Note: Check 48 hrs to 60 mins before flight

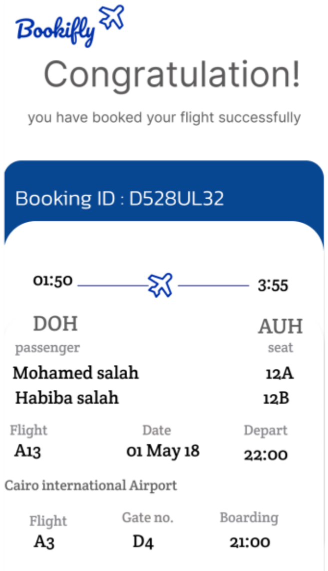
Select Airline

6E

Booking Ref/PNR

Email/Last Name

Figure (10)



Congratulation!

you have booked your flight successfully

Booking ID : D528UL32

01:50 3:55

DOH AUH

passenger seat

Mohamed salah 12A

Habiba salah 12B

Flight Date Depart

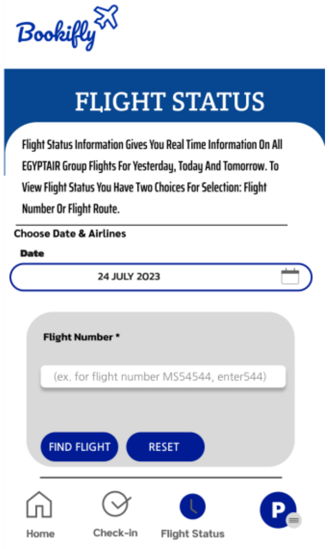
A13 01 May 18 22:00

Cairo international Airport

Flight Gate no. Boarding

A3 D4 21:00

Figure (11)



FLIGHT STATUS

Flight Status Information Gives You Real Time Information On All EGYPTAIR Group Flights For Yesterday, Today And Tomorrow. To View Flight Status You Have Two Choices For Selection: Flight Number Or Flight Route.

Choose Date & Airlines

Date

24 JULY 2023

Flight Number *

(ex. for flight number MS54544, enter544)

Figure (12)

Lastly the user can view their profile as shown in figure (13), they can edit their profile photo, edit their account, change password , log out and more settings as shown in figure (14), also the user can change password as shown in figure (15), choose currency as shown in figure(16), choose the language as shown in figure (17) and he can contact us as shown in figure (18)

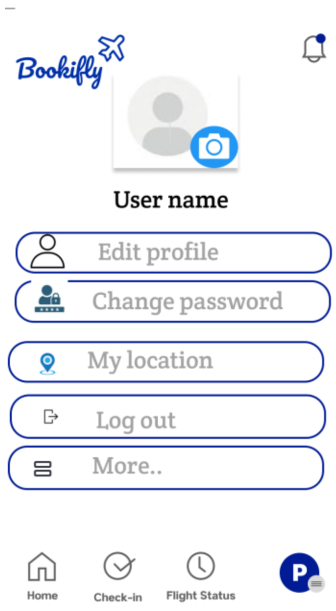
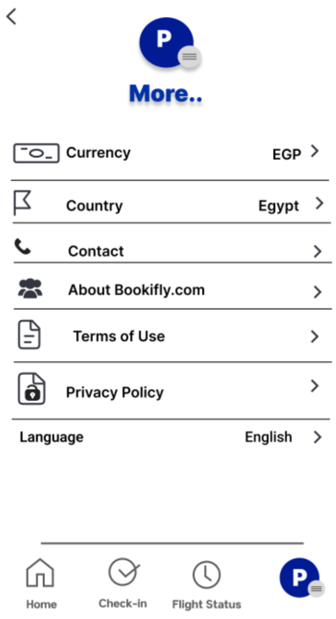


Figure (13)



Currency EGP

Country Egypt

Language English

Figure (14)

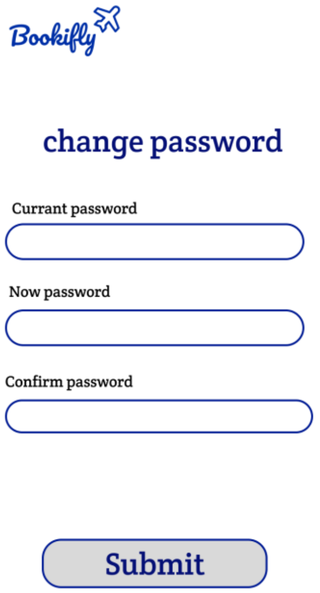
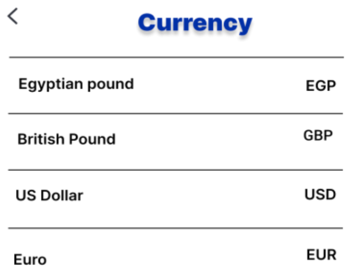


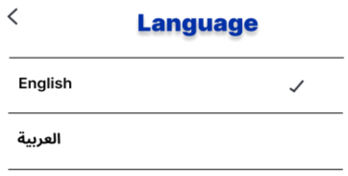
Figure (15)



Currency

Egyptian pound	EGP
British Pound	GBP
US Dollar	USD
Euro	EUR

Figure (16)

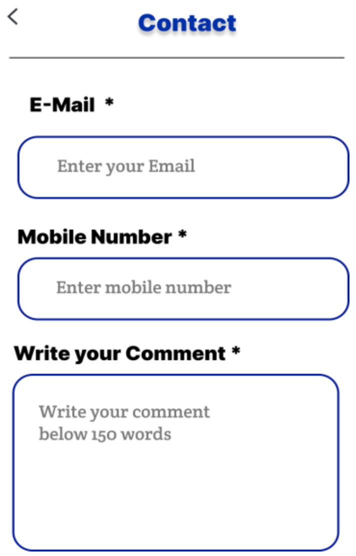


Language

English ✓

العربية

Figure (17)



Contact

E-Mail *

Enter your Email

Mobile Number *

Enter mobile number

Write your Comment *

Write your comment below 150 words

Figure (18)

5.1 Software Interface:

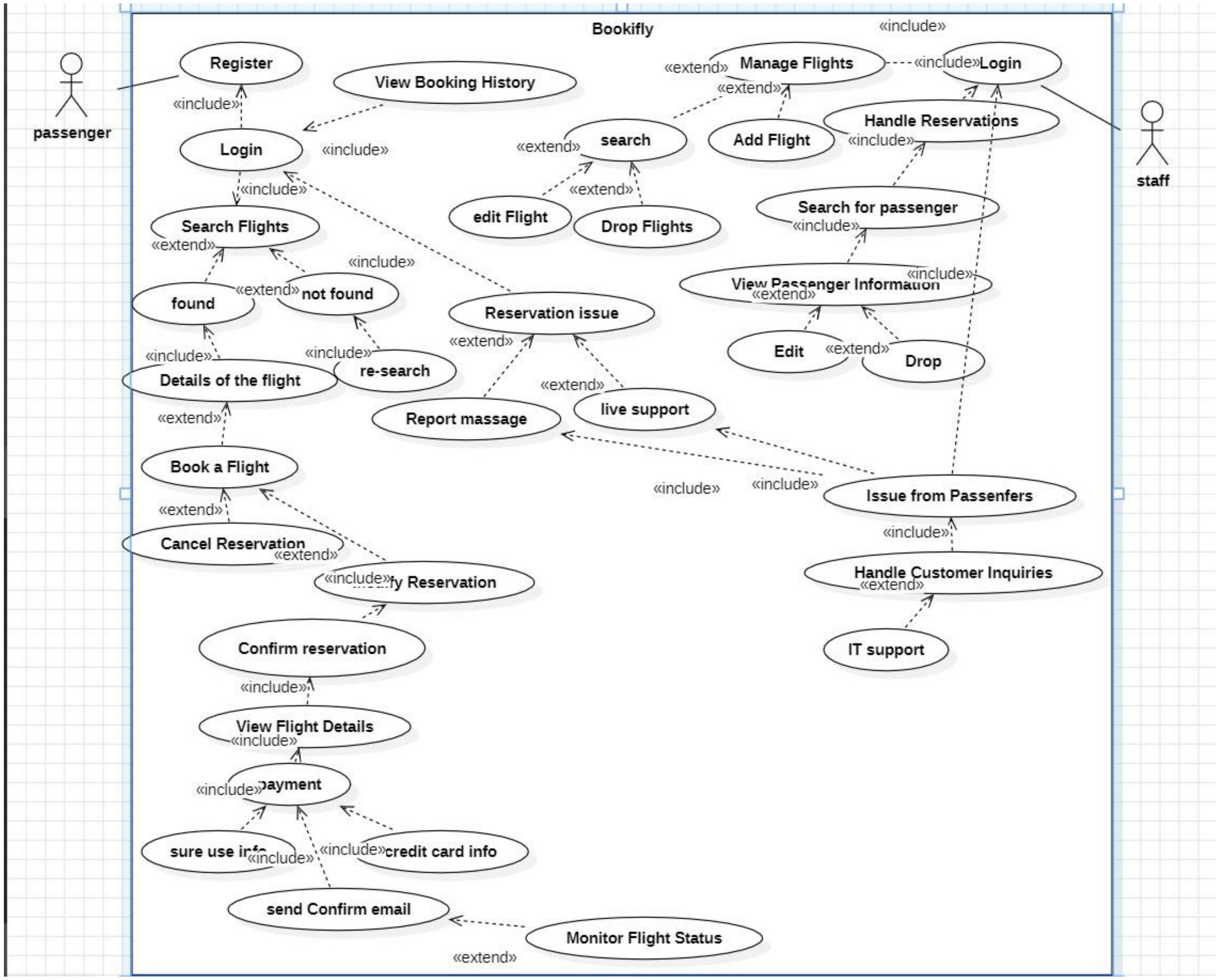
The flight booking system, "Bookifly" is intricately designed using Flutter SDK, Dart SDK, and Android SDK tools. Development is facilitated through Visual Studio Code, incorporating essential packages and libraries for seamless functionality specific to flight reservations. " Bookifly " offers an intuitive interface for users to input flight details, receive real-time notifications, access updates, and share relevant information. These user interactions are transmitted to the server as Outgoing data, ensuring a fluid booking experience.

5.2 Hardware Interface:

For optimal functionality, the "Bookifly" flight booking application mandates a reliable internet connection. Additionally, mobile devices must support GPS functionality to enhance location-based services crucial for flight reservations. "Bookifly" leverages Flutter for its cross-platform capabilities, ensuring a consistent booking experience across various devices. Data storage and retrieval are streamlined through the integration of specialized tools, enhancing the efficiency of the flight booking system.

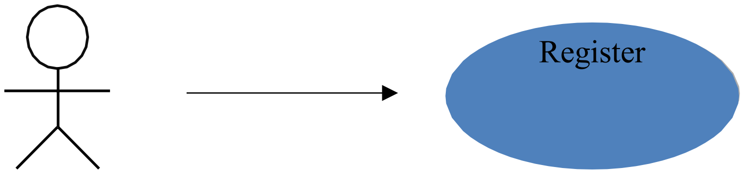
6 Diagrams

6.1Use case diagram



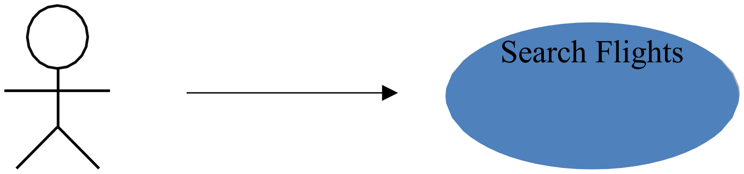
Use case scenarios

Use case 1

	
Use case Name	Register
Actors	User
Main success scenario	1. System asks the user to enter the Full name. 2. System asks the user to give their information: username, password (at least 8 characters), email (unregistered email), phone number, address, photo (optional). 3. System asks the user to enter the Passport information. 4. User gives the system the required information. 5. User clicks on the sign in button. 6. System checks that the email is not registered. 7. System checks that the username is not registered. 8. System creates the account.
Exceptions	1. User clicks on the sign in button without filling all of the required information. 2. Email is registered. 3. Username is registered.

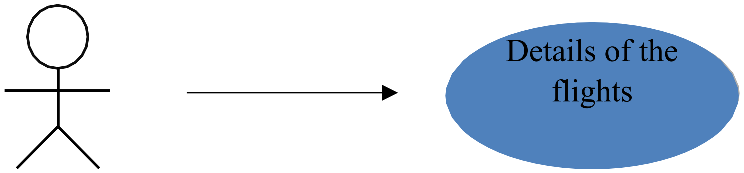
Actions	1. System display to the user an alert “Please fill in all the required fields”. 2. User fills the missing information. 3. System display to the user “Email already registered”. 4. User enters another email. 5. System display to the user “Username already exist”. 6. User enters another username.
Pre-Condition	• User had to download the application.
Post Condition	• Another user is added to the system. • User is successfully registered. • User information is stored in the database.

Use case 2

	
Use case Name	Search Flights
Actors	User

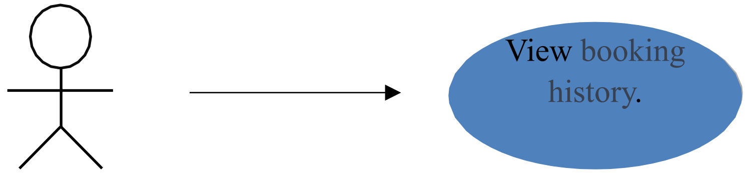
Main success scenario	1. System asks the user to enter departure city. 2. System asks the user to enter destination city. 3. System asks the user to give their information: Travel date and Number of passengers. 4. User clicks on the search button. 5. System checks that the available fights in that day.
Exceptions	1. Checks the departure and destination city and the date. 2. When click search button shows the available flights provides an appropriate error message and recommends trying again later.
Actions	1. Check the user's input for validity and completeness. 2. Ensure that the required information is provided and is in the correct format.
Pre-Condition	• The search parameters (departure city, destination, date, etc.) must be valid.
Post Condition	• Relevant flight options are displayed to the user. • The system should present a list of available flights based on the provided criteria.

Use case 3

	
Use case Name	Book flight.
Actors	User
Main success scenario	1. The system searches the database for available flights based on the provided criteria. 2. A list of relevant flights is displayed to the user. 3. User explores the displayed flight options. 4. User chooses a preferred flight from the list.
Exceptions	1. If the user provides invalid or incomplete search parameters, the system displays an error message and prompts the user to correct the input. 2. If there are no flights matching the provided criteria, the system informs the user and suggests alternative search options. 3. In case of system errors or connectivity issues, the system provides an appropriate error message and recommends trying again later.

Actions	1. Enter search parameters. 2. Review displayed flight options. 3. Select a preferred flight. 4. Optionally filter or sort results. 5. Proceed to book.
Pre-Condition	• The user is logged into the system. • Valid and complete search parameters are provided. • The system has up-to-date flight data.
Post Condition	• A list of relevant flights matching the search criteria is displayed. • The user can proceed to book the selected flight. • The system remains in a stable state, ready for the next user interaction.

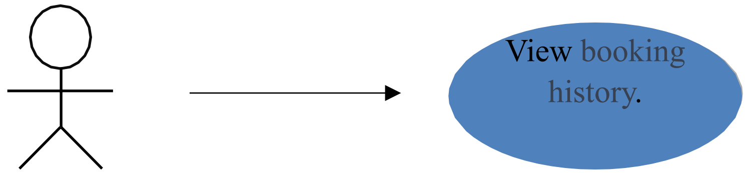
Use case 4

	
Use case Name	Book flight.

Actors	User
Main success scenario	1. User logs into the system. 2. User enters the flight search criteria (departure city, destination, date, etc.). 3. System validates the input criteria. 4. System displays a list of available flights based on the criteria. 5. User selects a preferred flight. 6. System confirms the selected flight details. 7. User provides necessary passenger details. 8. System validates the passenger information. 9. User proceeds to payment. 10. System processes the payment. 11. System issues a booking confirmation with the flight details. 12. User receives a booking confirmation.
Exceptions	• If the booking processing fails, the system notifies the user and provides guidance on resolving the issue or using an alternative booking method.
Actions	1. Enter search parameters. 2. Review displayed flight options. 3. Select a preferred flight. 4. Proceed to book.
Pre-Condition	• The user must be logged into the system. • Valid flight data must be available in the system. • The user should have valid information to book.
Post Condition	• The user has successfully booked a flight. • A booking confirmation with flight details is generated and sent to the user.

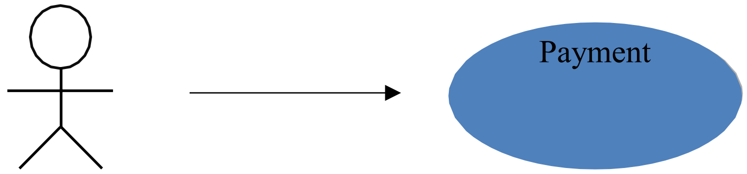
	• The system is updated with the new booking information.

Use case 5

	
Use case Name	View booking history.
Actors	User
Main success scenario	1. The customer logs into the system. 2. The customer navigates to the "Booking History" section. 3. The system retrieves and displays a list of past bookings associated with the customer's account. 4. The customer can view details of each booking, such as flight information, dates, and prices.
Exceptions	1. If the customer is not logged in, the system prompts them to log in before accessing the booking history. 2. If there are no past bookings for the customer, the system informs the customer that the booking history is empty.

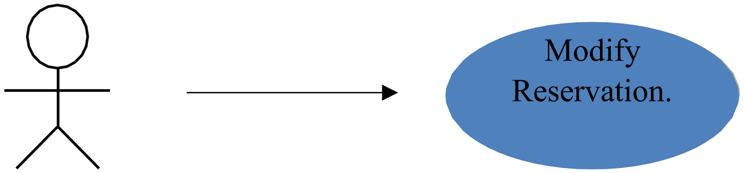
Actions	1. Customer logs in. 2. Customer navigates to the "Booking History" section. 3. System retrieves and displays past bookings. 4. Customer views details of each booking.
Pre-Condition	• The customer must be authenticated and logged into the system.
Post Condition	• The system displays the customer's booking history, allowing them to review details of past bookings.

Use case 6

	
Use case Name	Payment.
Actors	User
Main success scenario	1. The customer selects a flight and proceeds to the payment step. 2. The system prompts the customer to enter payment details (credit card, PayPal, etc.). 3. The customer provides valid payment information. 4. The system contacts the payment gateway for transaction processing. 5. The payment gateway confirms the success of the transaction. 6. The system updates the booking status to "Confirmed." 7. The system sends a booking confirmation to the customer.

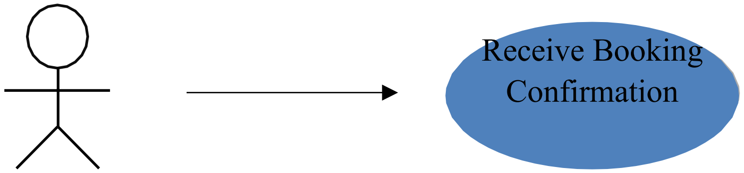
Exceptions	1. If the customer provides invalid payment details, the system notifies the user and prompts them to correct the information. 2. If the payment gateway reports a transaction failure, the system notifies the user and provides guidance on the next steps.
Actions	1. Prompt the user to enter payment details. 2. Send payment details to the payment gateway for processing. 3. Receive and interpret the response from the payment gateway. 4. If the transaction is successful, update the booking status to "Confirmed."
Pre-Condition	• The customer has selected a flight for booking. • The customer is authenticated and has provided necessary personal information. • The flight availability has been confirmed.
Post Condition	• The customer has selected a flight for booking. • The customer is authenticated and has provided necessary personal information. • The flight availability has been confirmed.

Use case 7

	
Use case Name	Modify Reservation.
Actors	User
Main success scenario	1. User logs into the system. 2. User navigates to the "My Reservations" or a similar section. 3. User selects the reservation they want to modify. 4. User makes the necessary modifications (e.g., change of date, passenger name, or seat preference). 5. User confirms the modifications. 6. System updates the reservation with the user-provided changes. 7. System displays a confirmation message.
Exceptions	1. If the user attempts to make an invalid modification (e.g., changing a non-changeable parameter), the system displays an error message and prompts the user to correct the issue. 2. If the selected reservation is not found in the system, the user is notified, and they may be prompted to re-enter the reservation details.

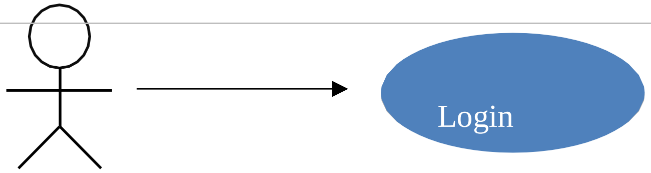
Actions	1. User input validation: Validate user input to ensure that modifications adhere to the system's rules and constraints. 2. Reservation update: Modify the reservation data according to the user's input. 3. Confirmation message: Display a confirmation message to the user after successful modification.
Pre-Condition	• The user must be authenticated and logged into the system. • The reservation to be modified must exist in the system.
Post Condition	• The reservation is successfully updated with the user's modifications. • The system provides a confirmation to the user, indicating that the reservation has been modified. • If applicable, any associated details such as updated itinerary or payment information are reflected accurately.

Use case 8

	
Use case Name	Receive Booking Confirmation
Actors	User
Main success scenario	1. The customer successfully selects a flight and proceeds to book it. 2. The system processes the booking request. 3. The booking confirmation is generated. 4. The system sends the booking confirmation to the customer via the chosen communication method (e.g., email, SMS).departure time, arrival time, and any delays.
Exceptions	1. If the payment transaction fails, the system notifies the customer and provides guidance on resolving the issue. 2. If the selected flight becomes unavailable during the booking process, the system notifies the customer and suggests alternative flights.
Actions	1. The customer selects a specific flight for booking. 2. The system processes the payment transaction securely. 3. Upon successful payment, the system generates a booking confirmation containing relevant details. 4. The system sends the booking confirmation to the customer through the selected communication channel
Pre-Condition	• The customer selects a specific flight for booking. • The system processes the payment transaction securely.

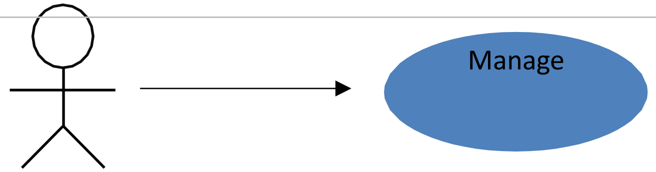
	- Upon successful payment, the system generates a booking confirmation containing relevant details. - The system sends the booking confirmation to the customer through the selected communication channel
Post Condition	- The customer receives a booking confirmation. - The booking information is stored securely in the system. - If applicable, the customer's account reflects the booked flight

Use case 9

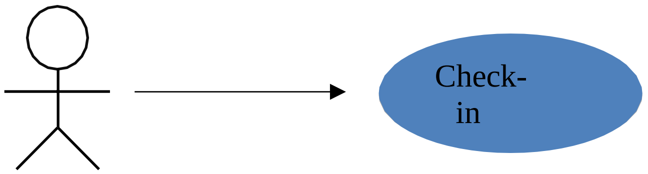
	
Use case Name	Staff Login
Actors	Airline Staff
Main success scenario	1. The system prompts the airline staff to enter their valid credentials (username and password). 2. Airline staff enters their login information. 3. The system verifies the credentials and grants access to the staff. 4. The staff gains access to the system's functionalities for airline staff.
Exceptions	1. Invalid credentials are entered. 2. The system displays an alert indicating that the credentials are invalid. 3. The staff re-enters the correct credentials.

Actions	1. The system displays an alert: "Invalid credentials. Please enter valid username and password." 2. The staff re-enters the correct credentials.
Pre-Condition	• The airline staff has been provided with valid login credentials.
Post Condition	• The airline staff gains access to the system.

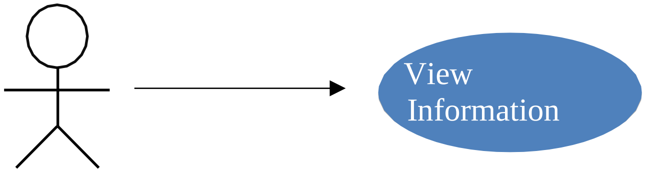
Use case 10

	
Use case Name	Manage Flights
Actors	Airline Staff
Main success scenario	1. The system provides the airline staff with options to add, update, or cancel flights. 2. Airline staff selects the desired operation (add/update/cancel). 3. The staff enters the necessary information for the selected operation. 4. The system processes the request and updates the flight information accordingly
Exceptions	1. The staff attempts an operation with incomplete or invalid information. 2. The system displays an error message. 3. The staff corrects the information and resubmits the request.
Actions	1. The system displays an error message: "Incomplete or invalid information. Please provide the required details." 2. The staff corrects the information and resubmits the request.
Pre-Condition	• The airline staff is logged into the system.
Post Condition	• Flight information is updated in the system.

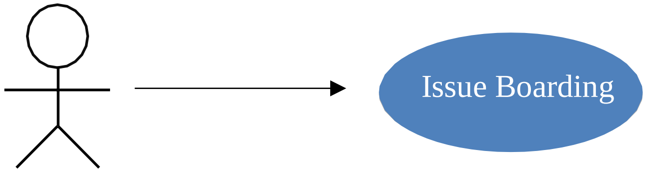
Use case 11

	
Use case Name	Report a Stray
Actors	Airline Staff
Main success scenario	1. The system allows airline staff to access the check-in functionality. 2. Airline staff selects the flight for which passengers need to be checked in. 3. The system displays a list of passengers booked for the selected flight. 4. The staff checks in passengers as they arrive.
Exceptions	1. The system cannot retrieve the passenger list. 2. The system displays an error message. 3. The staff retries accessing the passenger list.
Actions	1. The system displays an error message: "Please select a reservation to assist." 2. The staff selects a valid reservation. 3. The system displays an error message: "Invalid action. Please choose a valid action for the reservation." 4. The staff corrects the action or informs the passenger of limitations.
Pre-Condition	• The airline staff is logged into the system.
Post Condition	• Passengers are checked in for the selected flight.

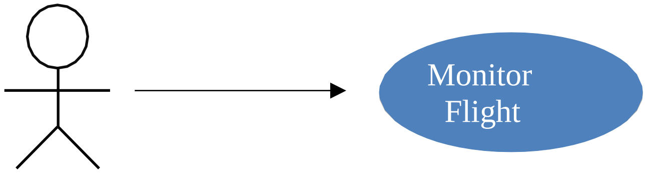
Use case 12

 	
Use case Name	View Passenger Information
Actors	Airline Staff
Main success scenario	1. The system allows airline staff to access passenger details and booking history. 2. Airline staff enters the passenger's name or booking reference. 3. The system retrieves and displays detailed information about the passenger, including previous and upcoming flights.
Exceptions	1. The system cannot retrieve passenger information. 2. The system displays an error message. 3. The staff verifies the entered information and retries.
Actions	1. The system displays an error message: "Passenger information not found. Please verify the entered details." 2. The staff verifies the entered information and retries.
Pre-Condition	• The airline staff is logged into the system.
Post Condition	• The staff has access to detailed passenger information.

Use case 13

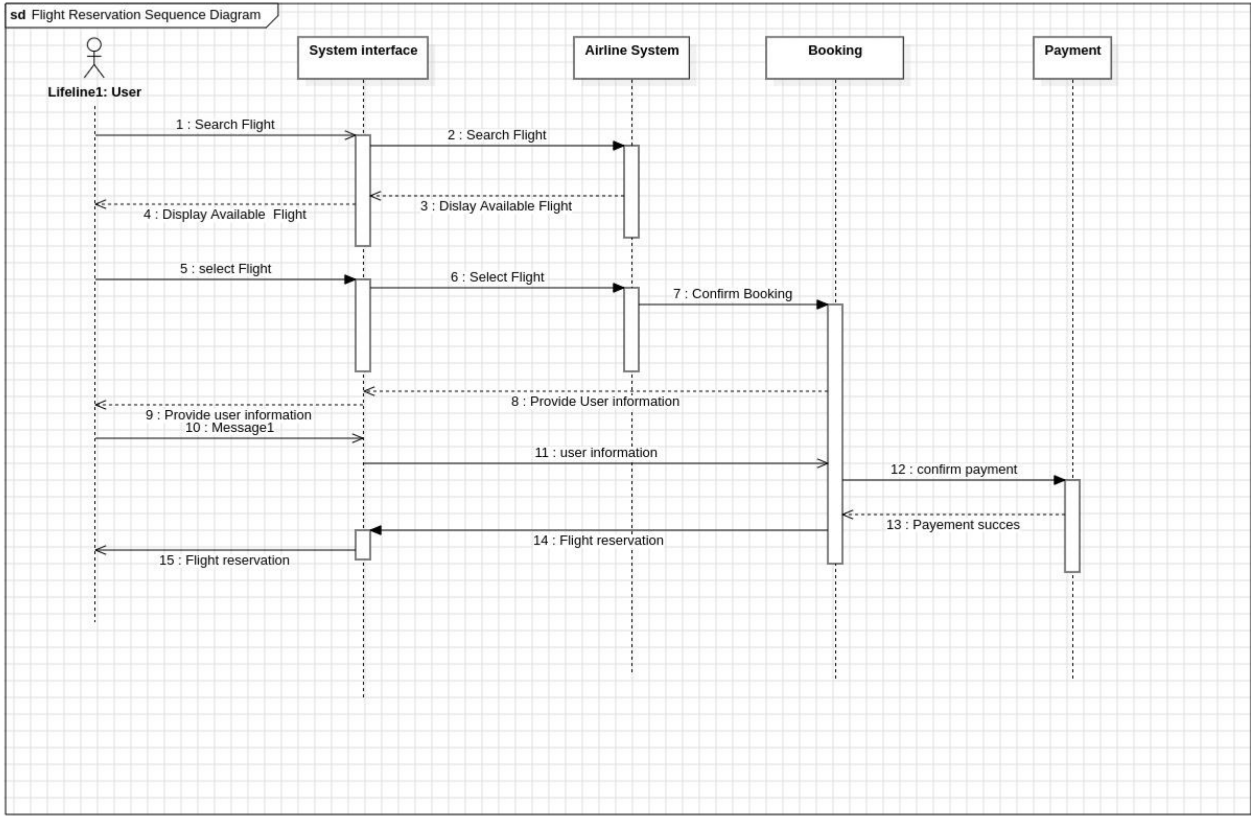
	
Use case Name	Issue Boarding Pass
Actors	Airline Staff
Main success scenario	1. The system allows airline staff to issue boarding passes to checked-in passengers. 2. Airline staff selects the flight and passenger for which the boarding pass needs to be issued. 3. The system generates and displays the boarding pass with relevant flight details and passenger information.

Use case 14

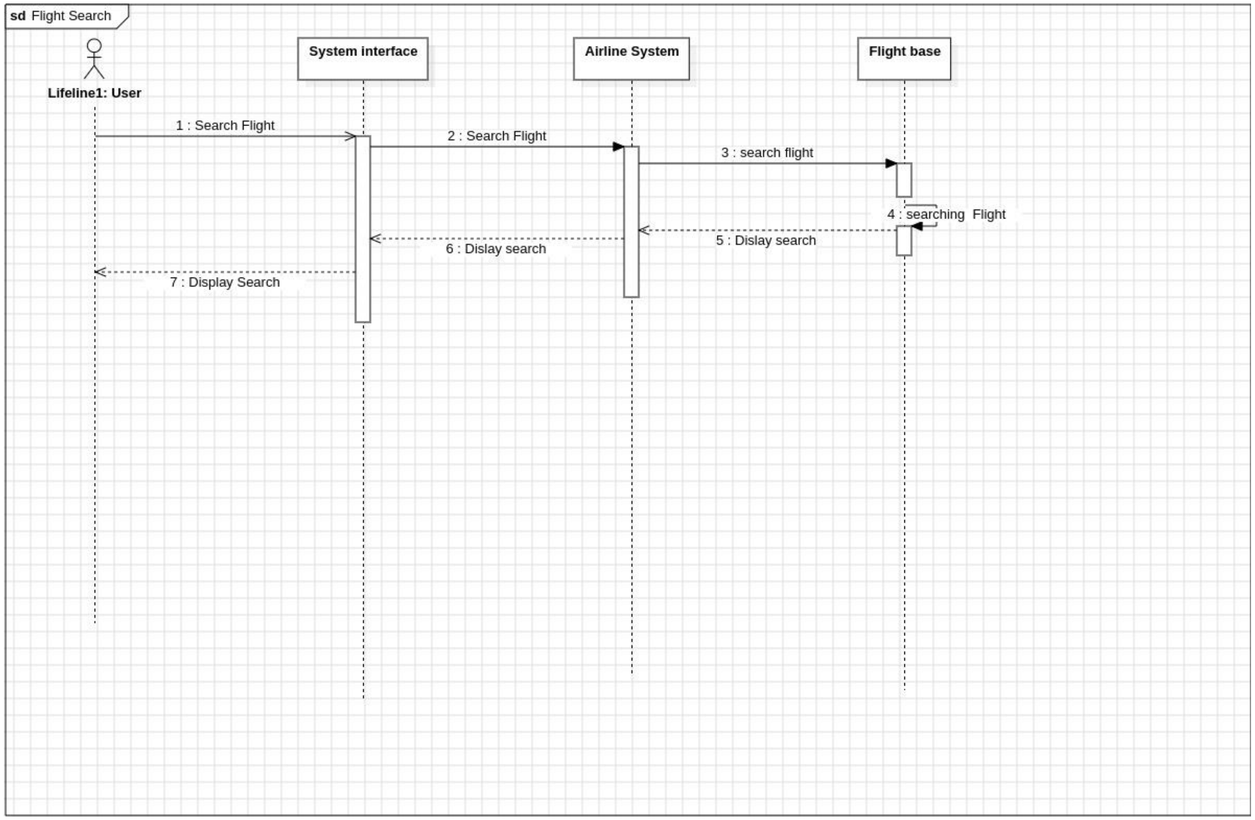
	
Use case Name	Donate to Shelter
Actors	Airline Staff
Main success scenario	1. The system provides real-time updates on the status of each flight. 2. Airline staff selects the flight they want to monitor. 3. The system displays current information on the flight status, including departure time, arrival time, and any delays.
Exceptions	1. The system cannot retrieve the flight status information. 2. The system displays an error message. 3. The staff retries accessing the flight status.
Actions	1. The system displays an error message: "Unable to retrieve flight status. Please try again later." 2. The staff retries accessing the flight status.
Pre-Condition	• The airline staff is logged into the system.
Post Condition	• The staff has real-time information on the selected flight's status.

6.1.2 Sequence Diagram:

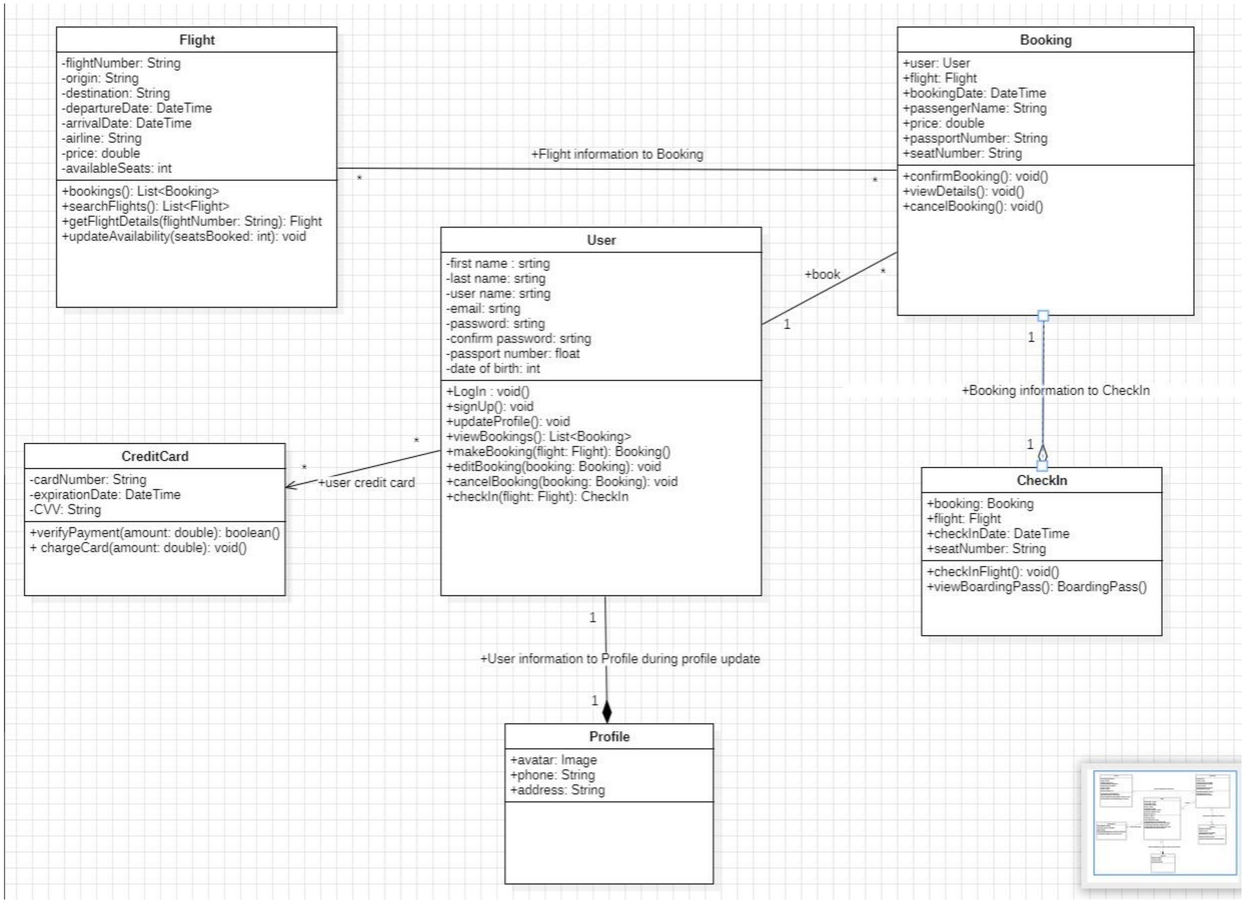
6.2.1 Flight Reservation Sequence Diagram



6.2.2Flight Search



6.3 Class Diagram:



7 Testcases

Test Case Template

Test Case ID: Fun_10	**Test Designed by:** Nada Radwan
Test Priority (Low/Medium/High): Med	**Test Designed date:** Nada Radwan
Module Name: Google login screen	**Test Executed by:** Nada Radwan
Test Title: Verify login with valid username and password	**Test Execution date:** 24/12/2023
Description: Test the Google login page	

Pre-conditions: User has valid username and password

Dependencies:

Step	Test Steps	Test Data	Expected Result	Actual Result	Status (Pass/Fail)	Notes
1	Navigate to login page		User should be able to login	As Expected	Pass	
2	Provide valid username	User= Nada@gmail.com	Credential can be entered	As Expected	Pass	
3	Provide valid	Password: 1234	Credential can be	As Expected	Pass	

	password		entered			
4	Click on Login button		User Logged-in	User Logged-in successfully	pass	

Post-conditions:

User is validated with database and successfully login to account. The account session details are logged in database.

Test Case ID: Fun_13

Test Priority (Low/Medium/High): Med

Test Designed by: Nada Radwan

Test Designed date: Nada Radwan

Module Name: login screen

Test Title: Verify login with invalid username and password

Description: Test the Google login page

Test Executed by: Nada Radwan

Test Execution date: 24/12/2023

Pre-conditions: User has invalid username and password

Dependencies:

Step	Test Steps	Test Data	Expected Result	Actual Result	Status (Pass/Fail)	Notes
1	Navigate to login page		User should be able to login	As Expected	Pass	
2	Provide valid username	User= Nada@gmail.com (invalid username)	Credential can be entered	As Expected	Pass	
3	Provide valid password	Password: 134 (invalid username)	Credential can be entered	As Expected	Pass	
4	Click on Login button		User Logged-in	Unsuccessful log-in system because invalid username or password	Fail	

Post-conditions:

User is validated with database and failed login to account. The account session details are not logged in database.

Test Case ID: Fun_01	**Test Designed by:** Sama Zali
Test Priority (Low/Medium/High): High	**Test Designed date:** Sama Zali
Module Name: login screen	**Test Executed by:** Sama Zali
Test Title: Verify flight search with valid criteria	**Test Execution date:** 24/12/2023
Description: Test the flight search functionality with valid departure, destination, date, and number of passengers	

Pre-conditions: User is logged into the system

Dependencies: The system has up-to-date flight data

Step	Test Steps	Test Data	Expected Result	Actual Result	Status (Pass/Fail)	Notes
	Navigate to flight search page		User should be able to flight search		Pass	
1	User should be able to enter the departure city	departure = Paris	Credential can be entered	As Expected	Pass	
2	User should be able to enter the destination city	destination = London	Credential can be entered	As Expected	Pass	
3	User should be able to enter the travel date	Date = 10/06/2024	Credential can be entered	As Expected	Pass	
4	Enter valid number of passengers	number = 3	Credential can be entered	As Expected	Pass	
5	Click on search button		User search flight	User search flight successfully	Pass	

Post-conditions:

Users can view and select a flight from the list of available flights.

Test Case ID: Fun_07	**Test Designed by:** Ahmed Amin
Test Priority (Low/Medium/High): High	**Test Designed date:** Ahmed Amin
Module Name: Flight Booking	**Test Executed by:** Ahmed Amin
Test Title: Verify flight booking with valid payment information	**Test Execution date:** 24/12/2023
Description: Test the flight booking functionality with valid passenger payment information	

Pre-conditions: User has selected a flight from the search results

Dependencies: The system has a secure payment gateway

Step	Test Steps	Test Data	Expected Result	Actual Result	Status (Pass/Fail)	Notes
1	User should be able to enter Card number	Card number : 2553687426	Credential can be entered	As Expected	Pass	
2	User should be able to enter the card holder name	Mane : Ahmed Amin	Credential can be entered	As Expected	Pass	
3	User should be able to enter Exp. Month	Exp .month = 12	Credential can be entered	As Expected	Pass	
4	User should be able to enter Exp. Year	Exp . year = 2025	Credential can be entered	As Expected	Pass	
5	User should be able to enter CCV	CCV = 368	Credential can be entered	As Expected	Pass	
6	Click Pay now button		Users enter his payment info	User enter his payment info successfully	Pass	

Post-conditions:

User has successfully booked a flight and received a booking confirmation. The booking information is stored in the system and the payment transaction is processed.

Test Case ID: Fun_02	**Test Designed by:** Ahmed Amin
Test Priority (Low/Medium/High): High	**Test Designed date:** Ahmed Amin
Module Name: Flight Booking	**Test Executed by:** Ahmed Amin
Test Title: Verify flight booking with valid payment information	**Test Execution date:** 24/12/2023
Description: Test the flight booking functionality with valid passenger payment information	

Pre-conditions: User has selected a flight from the search results

Dependencies: The system has a secure payment gateway

Step	Test Steps	Test Data	Expected Result	Actual Result	Status (Pass/Fail)	Notes
1	User should be able to enter Card number	Card number : 2553687426(invalid card name)	Credential can be entered	As Expected	Pass	
2	User should be able to enter the card holder name	Mane : Ahmed Amin	Credential can be entered	As Expected	Pass	
3	User should be able to enter Exp. Month	Exp .month = 12	Credential can be entered	As Expected	Pass	
4	User should be able to enter Exp. Year	Exp . year = 2025	Credential can be entered	As Expected	Pass	
5	User should be able to enter CCV	CCV = 368	Credential can be entered	As Expected	Pass	
6	Click Pay now button		Users enter his payment info	Unsuccessful booking because invalid card data	Fail	

Post-conditions:

User has successfully booked a flight and received a booking confirmation. The booking information is stored in the system and the payment transaction is processed.

Test Case ID: Fun_04	**Test Designed by:** Ahmed Mousa
Test Priority (Low/Medium/High): High	**Test Designed date:** Ahmed Mousa
Module Name: search for User ticket	**Test Executed by:** Ahmed Mousa
Test Title: Verify ticket ID from User	**Test Execution date:** 24/12/2023
Description: Test the ticket searching functionality with valid passenger ticket ID	

Pre-conditions: User is logged into the system

Step	Test Steps	Test Data	Expected Result	Actual Result	Status (Pass/Fail)	Notes
1	Navigate to check ticket page		User should be able to check ticket	As Expected	Pass	
2	Provide valid ticket ID	D8652UL75	Credential can be entered	As Expected	Pass	
3	Provide valid Email or Last Name	Name: Mousa	Credential can be entered	As Expected	Pass	
4	Click on search button		Search in database	User Get his ticket successfully	pass	

Post-conditions:

User has successfully searched for his flight ticket ID and received details for his ticket.

Test Case ID: Fun_05	**Test Designed by:** Mohamed Farag
Test Priority (Low/Medium/High): med	**Test Designed date:** Mohamed Farag
Module Name: search for User ticket	**Test Executed by:** Mohamed Farag
Test Title: Change password for User	**Test Execution date:** 24/12/2023
Description: Change password for User	

Pre-conditions: User is logged into the system

Step	Test Steps	Test Data	Expected Result	Actual Result	Status (Pass/Fail)	Notes
1	Navigate to change password page		User should be able to	As Expected	Pass	

			check ticket			
2	Provide valid currant password	Curr = sama20	Credential can be entered	As Expected	Pass	
3	Provide valid new password	New = sama2002	Credential can be entered	As Expected	Pass	
4	Provide new password again	New = sama2002	Credential can be entered	As Expected	Pass	
5	Click on submit button		Save in database	User change password successfully	pass	

Post-conditions:

User has successfully changed his password .

Scrum video link : https://nileuniversity-my.sharepoint.com/:v:/g/personal/a_amin2151_nu_edu_eg/Ef6cTonI0xVGjK5AJq7NxoMBfVE8ssJhJRAuSg5JTAukKw?e=o3CDEw&nav=eyJyZWZlcnJhbEluZm8iOnsicmVmZXJyYWxBcHAiOiJTdHJlYW1XZWJBcHAiLCJyZWZlcnJhbFZpZXciOiJTaGFyZURpYWxvZy1MaW5rIiwicmVmZXJyYWxBcHBQbGF0Zm9ybSI6IldlYiIsInJlZmVycmFsTW9kZSI6InZpZXcifX0%3D

This Is the Link For our Design On Figma :

https://www.figma.com/file/8kI08eqFvnrPFvzMCJ0VOJ/software-project?type=design&node-id=0%3A1&mode=design&t=k4vuxLfPwCTr1SL5-1