

# Software Requirements Analysis

## Airline Reservation System



Ömer Talha BAYSAN  
20.07.2022

# Airline Reservation System

## 1. Introduction

This Project is a desktop application where the airline reservation system is coded with C# programming language. The application starts with a login form that leads to a login form. The login form will be given an authentication mechanism, if the admin id and password match the condition, the admin will login to the main form. The admin can navigate to the flights form from the main form. Can add, edit, delete and view flights. The admin can manage different Passengers, add, edit or update Passenger. The admin can manage different flight schedules that allocate a particular flight to them. The admin can add, edit and delete flight schedules. The admin can Book a Ticket for a particular Trip. Finally, the user can cancel a reservation.

### 1.1. Purpose of the system

The airline reservation system includes details about flight schedules, passenger reservations and ticket registrations.

### 1.2. Scope of the system

The users of the system consist of the system administrator and registrar. The aircraft, passengers, travel and reservation information in the system are under the control of the system users.

## 2. Current System

In several countries, if someone wants to book a flight ticket, they follow one of these things: Elle goes to the airport and book the ticket. Downloading the ticket as a paper document. Fill out the ticket from the System and print it out and deliver it to the airport.

Booking the ticket online at some registered ticket offices. Even the above approaches are booking tickets online, not completely online. The traveler may not have much freedom over this approach. Therefore, the Passenger may or may not be satisfied with this approach as it involves manual intervention such as traveling to the airport to book their ticket.

## 3. Proposed System

The proposed system provides complete freedom for the user, where the user can log in to this application and book her ticket on her own system.

### 3.1. Functional requirements

#### 3.1.1. Flights

In the Flights module, we can enter flight code, date, number of seats, source and destination information, as well as view registered flights. We can update and delete saved flights on the flight logs view screen.

#### 3.1.2. Passangers

To register a passenger in the Passengers module, we must enter the passenger code, name, number, address, nation, telephone number and gender. We can update or delete registered passengers from the passenger records display screen.

#### 3.1.3. Tickets

In order to reserve a ticket in the ticket reservation module, we need to enter the ticket and flight code, passenger code, name, nationality, passport number and ticket price information.

### 3.1.4. Cancellation

In the ticket cancellation module, we need to enter the cancellation, ticket and flight code and date information for ticket cancellation.

## 3.2. Nonfunctional requirements

### 3.2.1. Usability

Airline Reservation System serves a variety of people, including those who are not inclined to use computers. Therefore, My Wallet must be clear and easy to use.

### 3.2.2. Security

The system contains a lot of information such as flight, passenger and reservation. The data is kept in a database. Database security protects data against data theft.

### 3.2.3. Interface

The system offers multiple tools to the user. It is important that the interface has been developed for the most efficient and easiest use of these tools for the user.

## 4. System models

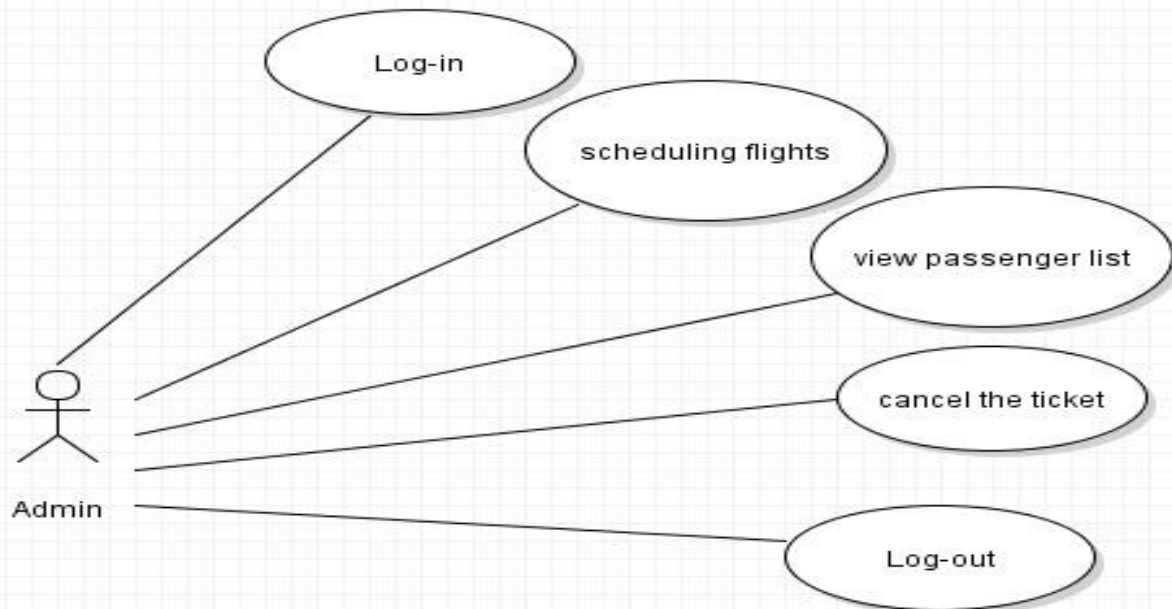
### 4.1. Scenarios

Use case Scenario	Record New Flights
Primary Actor	The system administrator
Relevant and Expectations	The system administrator wants to enter the system and enter the new flight information and add the flight record to the system.
Prerequisites	The system administrator must be authenticated.
Final Conditions	Flight information is entered into the system without error and completely.
Main Scenario	<ol style="list-style-type: none"> <li>1. The system administrator determines the flight code of the new flight.</li> <li>2. The system administrator determines the source and destination address of the new flight.</li> <li>3. The system administrator selects the date of the flight.</li> <li>4. The system administrator determines the capacity of the flight.</li> <li>5. The system saves information.</li> </ol>
Alternative Scenario	<ol style="list-style-type: none"> <li>1. There is a flight in the same cod. <ol style="list-style-type: none"> <li>a. The system issues an administrator warning message and requests a new flight code.</li> </ol> </li> <li>4. The seat limit has been reached. <ol style="list-style-type: none"> <li>a. The system gives an error when trying to add more than a certain number of passengers.</li> </ol> </li> </ol>

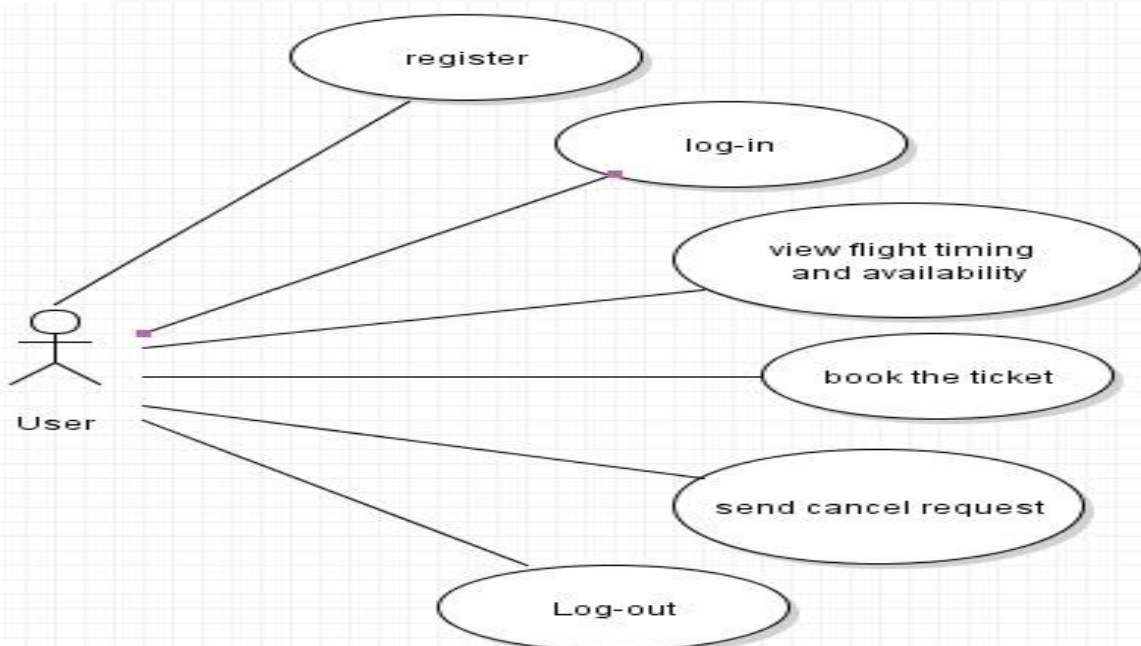
Use case Scenario	Passenger Registration
Primary Actor	Passenger
Relevant and Expectations	Passenger asks to enter the system, enter the passenger information correctly into the system and list the registered passengers properly.
Prerequisites	Passenger identity must be verified.
Final Conditions	Passenger information is entered into the system without error and completely.
Main Scenario	<ol style="list-style-type: none"> <li>1. The passenger enters the application.</li> <li>2. Logs into the Passengers module.</li> <li>3. Passenger code, name, phone number, address, gender and nationality information are entered into the system.</li> <li>4. Registration is complete.</li> </ol>
Alternative Scenario	<ol style="list-style-type: none"> <li>4. Passenger may want to view registered passengers. <ol style="list-style-type: none"> <li>a. It can easily look from where the registered passengers are listed.</li> </ol> </li> </ol>

## 4.2. Use Case Model

### - Admin Use Case Model

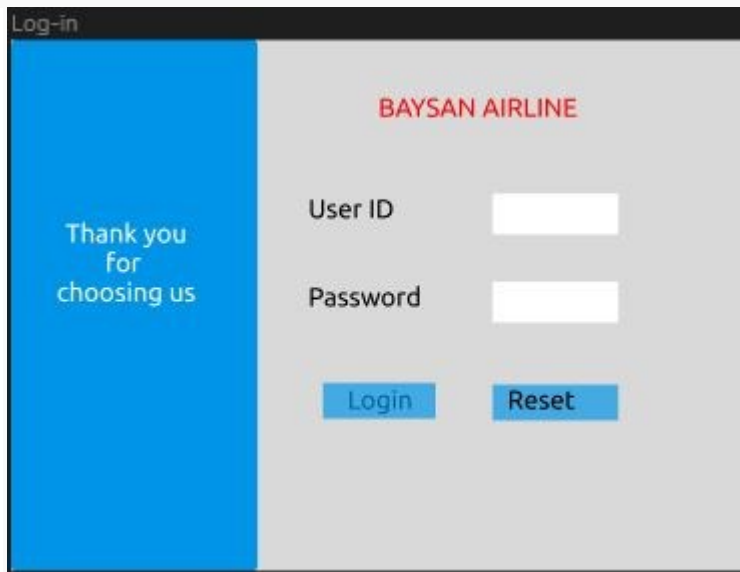


### - User Use Case Model



## 4.3. Screen Mock-ups

- Log-in



Log-in

Thank you for choosing us

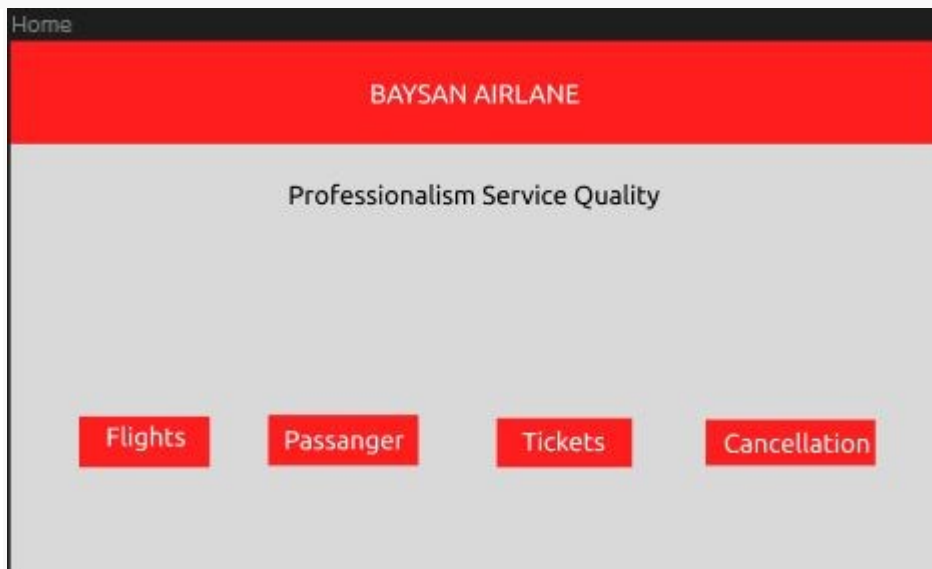
BAYSAN AIRLINE

User ID

Password

This mock-up shows a log-in interface. On the left, a blue vertical bar contains the text "Thank you for choosing us". The main area has a light gray background. At the top, "BAYSAN AIRLINE" is written in red. Below this, there are two input fields: "User ID" and "Password". At the bottom, there are two blue buttons labeled "Log in" and "Reset".

- Home



Home

BAYSAN AIRLINE

Professionalism Service Quality

This mock-up shows a home interface. At the top, a red horizontal bar contains "BAYSAN AIRLINE" in white. Below this, on a light gray background, is the text "Professionalism Service Quality". At the bottom, there are four red buttons labeled "Flights", "Passanger", "Tickets", and "Cancellation".

- Flights

Flights

BAYSAN AIRLINE

Record New Flight

Flight Code

Source

Destination

Take of Date

Num of Seats

Record

Reset

Back

View Flights

- View Schdule Flights

View Schdule Flights

BAYSAN AIRLINE

View Schdule Flights

Flight Code

Source

Take of Date

Destination

Num of Seats

Update

Delete

Reset

Back

Fcode

Fsource

Fdest

Fdate

Fcap



- Passengers

Passengers

**BAYSAN AIRLINE**  
Record New Passenger

Passanger ID

Passanger Name

Passport Number

Passanger Address

Nationality

Gender

Phone

- View Passengers

View Passenger

**BAYSAN AIRLINE**  
View Passengers

Passanger ID  Name

Passport  Address

Nationality  Gender

Phone

PassID	PassName	Passport	PassAd	PassGend	PassNat	Phone
--------	----------	----------	--------	----------	---------	-------

- Tickets

Tickets

BAYSAN AIRLINE  
Ticket Booking

Ticket ID

Flight Code

Passanger Id

Name

Passport

Nationalitiy

Amount

Book

Reset

Back

Bookings

Tid	Fcode	PId	PName	PPass	PNation	Amount
-----	-------	-----	-------	-------	---------	--------

- Cancellation

Cancellation

BAYSAN AIRLINE  
Ticket Cancellation

Cancel ID

Ticket ID

Flight Code

Date

Cancel

Reset

Back

Cancellation List

Cancel id	Ticket id	Flight code	Cancel Date
-----------	-----------	-------------	-------------