# EAST WEST UNIVERSITY

## Software Requirements Specification (SRS) Report

ON

## Airline Management System

Submitted in partial fulfillment of the requirement for the Semester **Spring 2025**

COURSE CODE: **CSE412**; SECTION: **02**

Of

BACHELOR OF SCIENCE

IN

COMPUTER SCIENCE AND ENGINEERING

**Submitted by**:

1. *Seaum Insaniat Swapnil (2021-2-60-016)*
2. *Jerin Anan Proma (2022-1-60-132)*
3. *Shanta Islam (2022-1-60-288)*
4. *Nusrat Jahan Oishi (2022-2-60-033)*

**Submitted to:**

*Yasin Sazid*

Lecturer

Dept of Computer Science & Engineering

# 1. Introduction

## 1.1 Project Overview

The Airline Management System is a web-based application designed to streamline airline operations and enhance the overall passenger experience.

 As a **passenger,** I want a quick and straightforward way to find flights, compare prices and book tickets without unnecessary complexity. I expect to see the total price upfront, including taxes, baggage fee and any extras, with no hidden charges. I also want a variety of secure payment options, such as credit/debit cards, mobile banking and PayPal, to choose from. After booking, I want instant confirmation and immediate access to my e-ticket. I also want the flexibility to reschedule or cancel my flight easily, with clear refund policies in place. Real-time notifications about flight delays, cancellation  and status updates are essential, along with the ability to select my seat through a graphical interface and add extras like baggage or meals. Frequent flyers will benefit from loyalty rewards, discounts and priority benefits. The system should have an easy-to-use, mobile-friendly interface that allows offline ticket access and responsive customer support via live chat, phone, and email.

As an **admin**, I need an intuitive dashboard to efficiently manage flights, schedules and passenger records. I want to see all relevant records displayed clearly in my dashboard and have the ability to add, modify, or cancel flights with automatic notifications sent to passengers. Efficient handling of customer support queries and real-time assistance is crucial. Additionally, the system should ensure secure and transparent payment and refund processes, while also allowing me to view and manage passenger feedback and reviews to improve our services.

## 1.2 Purpose and Scope

**Purpose:**

The purpose of this project is to develop an **Airline Management System** that automates airline booking and administration processes while enhancing the user experience. The system aims to eliminate inefficiencies, improve accuracy  and reduce manual workload.

**In Scope:**

The system will provide the following functionalities:

- **For Passengers:**
  - Quick and hassle-free flight search and booking.
  - Transparent pricing with no hidden charges.
  - Multiple secure payment options.
  - Instant ticket generation and e-ticket access.
  - Easy rescheduling and refund processing.
  - User-friendly seat selection with a graphical interface.
  - Frequent flyer benefits, discounts and loyalty programs.
- **For Admins:**
  - Efficient management of flight schedules, passenger data, and bookings.
  - Easy addition, modification, or cancellation of flights.
  - Secure payment and refund handling.
  - Customer feedback and review management.
  - Flight Status Dashboard, Booking Analytics, Customer Support Analytics and Revenue and Payment Analytics.

**Out of Scope:**

  - Real-time flight status notifications and alerts.
  - Customer support via live chat, phone, and email.
  - Mobile-friendly interface with offline ticket access
  - Automated customer notifications for flight changes.

## 1.3 Stakeholders

**Primary Stakeholders:**

- **Passengers**: Individuals who book flights and use the platform for flight-related services.
- **Airline Administrators**: Staff responsible for managing flight schedules, bookings, and passenger records.

**Secondary Stakeholders:**

- **Travel Agencies**: Businesses that may use the platform to book flights for customers.
- **Payment Service Providers**: Entities handling secure financial transactions.
- **Developers & System Administrators**: Responsible for maintaining and upgrading the system.
- **Customer Support Teams**: Assisting passengers with queries, issues, and requests.

# 2. Requirements Engineering Process

## 2.1 Stakeholder Needs & Analysis

**Primary Stakeholders:**

- Passengers
- Admin (Airline staff)

**Secondary Stakeholders:**

- Payment Gateway Providers
- Developers & System Admins

**Methods Used for Requirement Elicitation:**

- **Interviews**: Conducted with potential users (passengers and admins) to understand expectations.
- **Competitive Analysis**: Researched existing airline booking systems to identify strengths and gaps.

## 2.2 List of Requirements

**Functional Requirements (FRs):**

1. Users should be able to register and log in securely.
2. Passengers should be able to search for flights based on date, destination, and airline.
3. The system should display real-time seat availability.
4. Passengers should be able to book tickets and receive e-tickets instantly.
5. Multiple secure payment options should be integrated into the system.
6. Users should be able to cancel or modify bookings with refund policies applied.
7. Admins should be able to add, update, and cancel flights.
8. Passengers should receive notifications for booking confirmations.
9. Seat selection and extra baggage options should be available during booking.
10. A loyalty program should provide discounts for frequent flyers.
11. The system should maintain booking history for users.
12. Admins should be able to manage customer support requests.
13. Discount offers should be displayed prominently on the website.
14. The system shall provide a Flight Status Dashboard with real-time flight status charts (on-time, delayed, canceled).
15. The system shall provide Customer Support Analytics with query distribution and resolution time charts.
16. The system shall display Revenue and Payment Analytics with revenue breakdown and payment method charts.

**Non-Functional Requirements (NFRs):**

1. The system shall run on the web.
2. The passwords shall be encrypted by the system.
3. It shall work fluently in case of many users and data.
4. The website shall have a user-friendly interface that is easy to navigate.
5. The system shall provide regular backups to prevent data loss.
6. It shall maintain the transactions flawlessly so that no error occurs.
7. The website shall respond to user requests within a specified time to get a smooth experience.

**Extraordinary Requirements (Wow Factors):**

1. **Price Prediction Tool**: Analyzes past trends to suggest the best time to book tickets.
2. **Personalized Travel Suggestions**: The system recommends flights based on past travel history.

## 2.3 House of Quality (QFD Integration)

**Customer Requirements (CRs) List:**

1. Quick and hassle-free booking
2. Transparent pricing with no hidden charges
3. Flexible payment options
4. Real-time flight updates
5. Easy cancellation and refund process
6. Seat selection and add-on services
7. Reliable customer support
8. Discounts and loyalty rewards

**Engineering Requirements (TRs) List:**

1. Secure authentication system
2. Efficient database management
3. Payment gateway integration
4. Notification system implementation
5. Optimized seat selection algorithm
6. Customer support ticketing system
7. Dynamic pricing module for discounts

**QFD Matrix (House of Quality)**

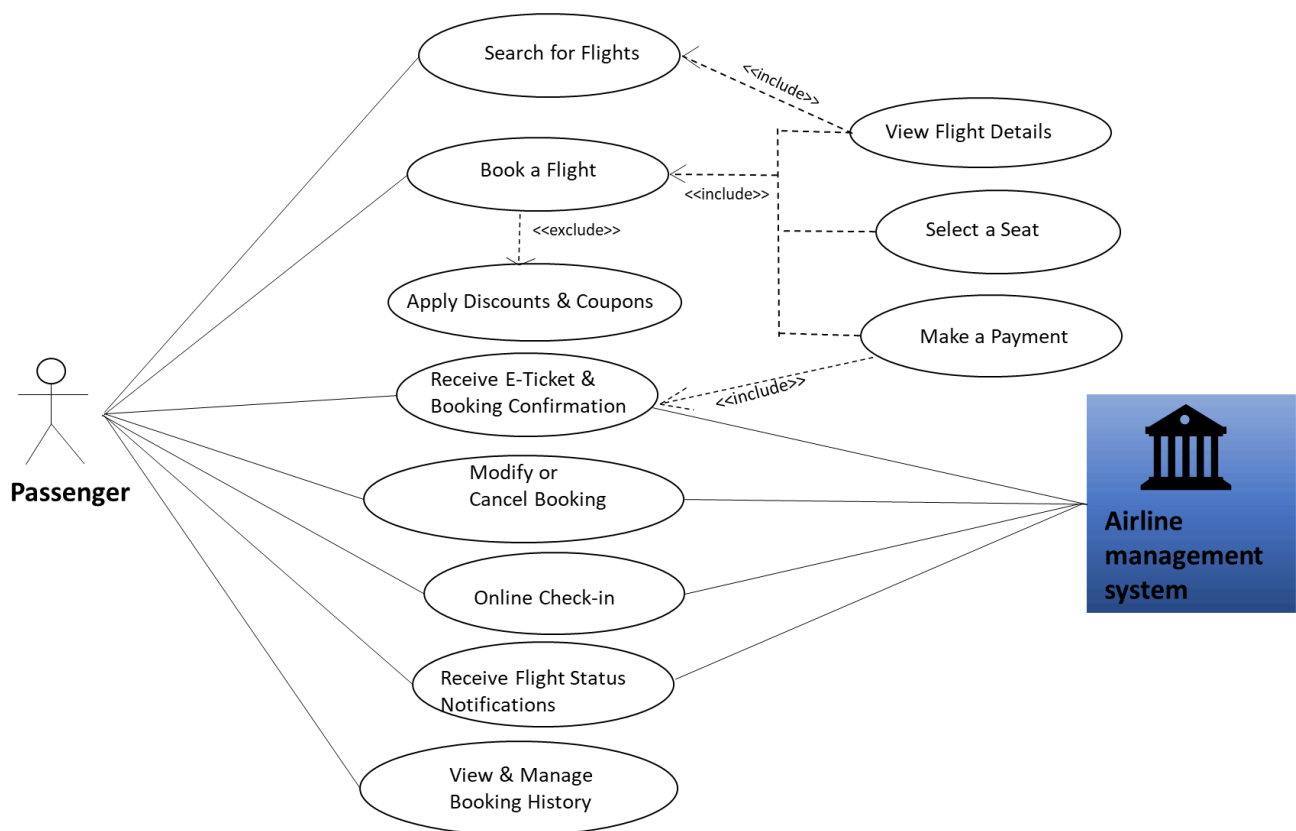| User Requirements | Performance | Security | Usability | Reliability | Scalability | Speed | Customer Support | Aesthetic Design | Notifications | Loyalty Program | Discount Offers | Price Prediction Tool | Personalized Travel Suggestions |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Register and log in securely** | Weak | Strong | Medium | Medium | Weak | Weak | Medium | Medium | Medium | Weak | Weak | Weak | Weak |
| **Search flights based on date, destination and airline** | Weak | Weak | Medium | Medium | Weak | Weak | Medium | Medium | Medium | Weak | Weak | Weak | Weak |
| **Display real-time seat availability** | Medium | Medium | Strong | Medium | Medium | Strong | Weak | Medium | Medium | Weak | Weak | Medium | Medium |
| **Book tickets and receive e-tickets instantly** | Medium | Strong | Strong | Strong | Medium | Strong | Medium | Medium | Strong | Weak | Weak | Weak | Weak |
| **Secure payment options integration** | Medium | Strong | Medium | Medium | Medium | Medium | Medium | Weak | Medium | Weak | Weak | Weak | Weak |
| **Cancel or modify bookings with refund policies** | Medium | Strong | Medium | Strong | Medium | Weak | Medium | Weak | Strong | Weak | Weak | Weak | Weak |
| **Add, update, and cancel flights (Admin)** | Medium | Strong | Medium | Strong | Strong | Weak | Medium | Weak | Weak | Weak | Weak | Weak | Weak |
| **Receive notifications for booking confirmations** | Medium | Medium | Strong | Strong | Medium | Medium | Strong | Medium | Strong | Weak | Weak | Medium | Medium |
| **Seat selection and extra baggage options** | Medium | Medium | Strong | Medium | Medium | Medium | Medium | Strong | Medium | Weak | Weak | Weak | Weak |
| **Loyalty program for frequent flyers** | Weak | Medium | Strong | Medium | Medium | Weak | Medium | Strong | Medium | Strong | Strong | Medium | Medium |
| **Display discount offers prominently** | Medium | Weak | Strong | Weak | Medium | Weak | Weak | Strong | Medium | Weak | Strong | Weak | Weak |

# 3. Requirements Modeling

## 3.1 Use Case Diagram
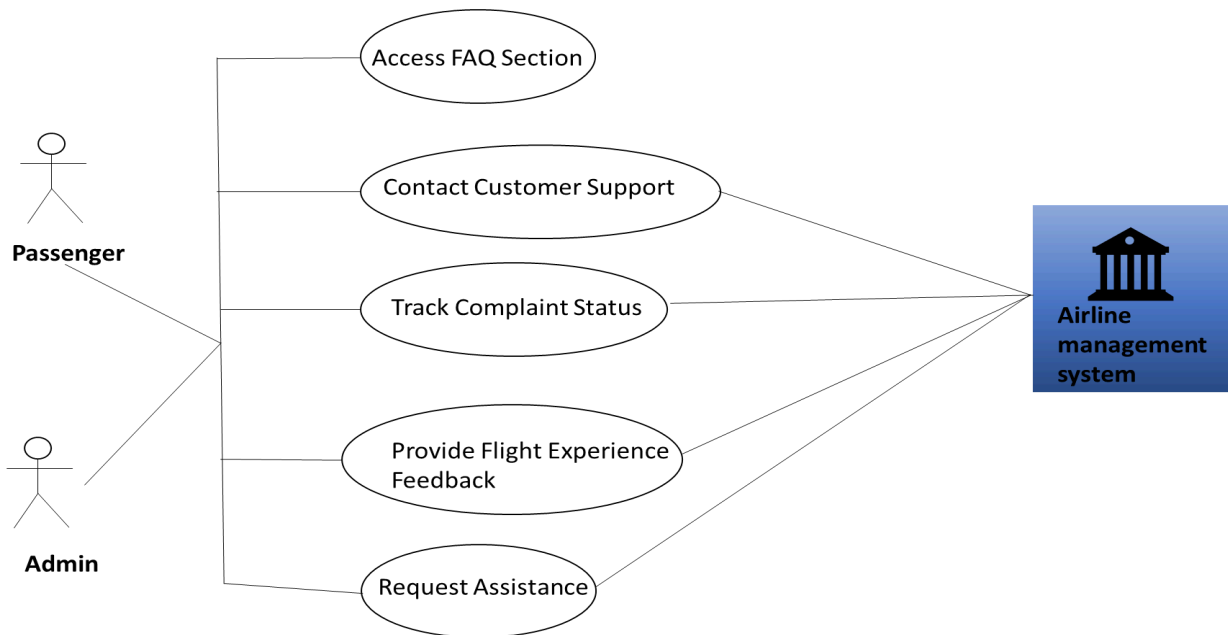
## Actors:

- Passenger (End User)
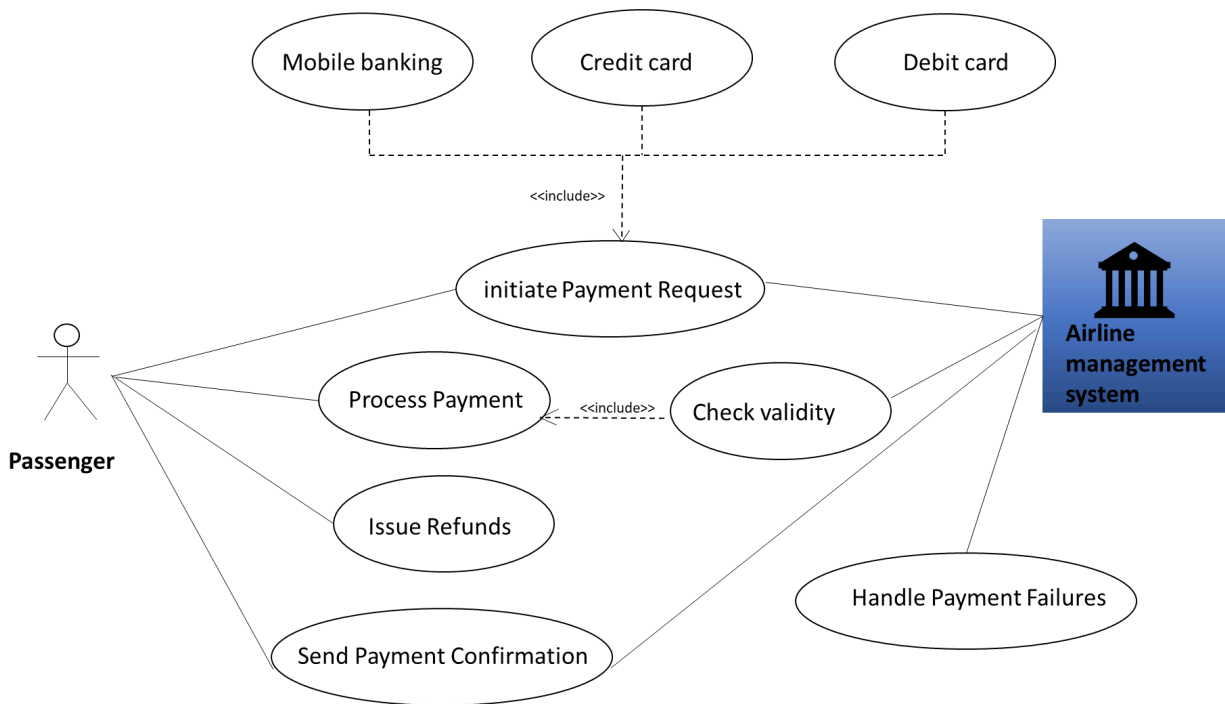- Admin (Airline Staff)

### Use Case Diagram (Passenger)



### Use Case Diagram (Admin)

Manage Passenger Details

Add a Flight Schedule

Modify or Cancel Flights

Manage Seat Availability

Handle Refund & Cancellation Requests

Manage Discounts & Promotions

Send Automated Customer Notifications

Admin

Airline management system

<<exclude>>
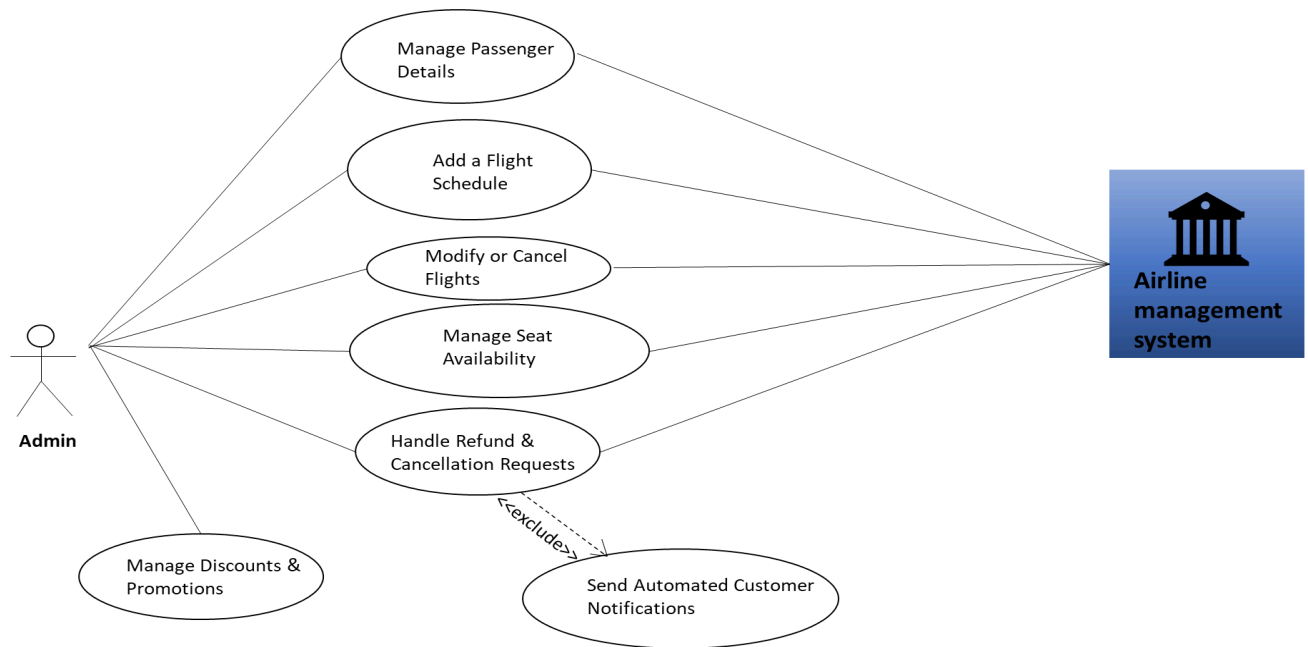
**Use Case Diagram (Customer Support & Feedback)**



Access FAQ Section

Contact Customer Support

Track Complaint Status

Provide Flight Experience Feedback

Request Assistance

Passenger

Admin

Airline management system

**Use Case Diagram (Payment Processing )**

| Mobile banking | Credit card | Debit card |

Passenger

initiate Payment Request

<<include>>

Process Payment   <<include>>   Check validity

Issue Refunds

Send Payment Confirmation

Handle Payment Failures

**Airline management system**

## 3.2 Activity diagram

**Activity Diagram for Log In**
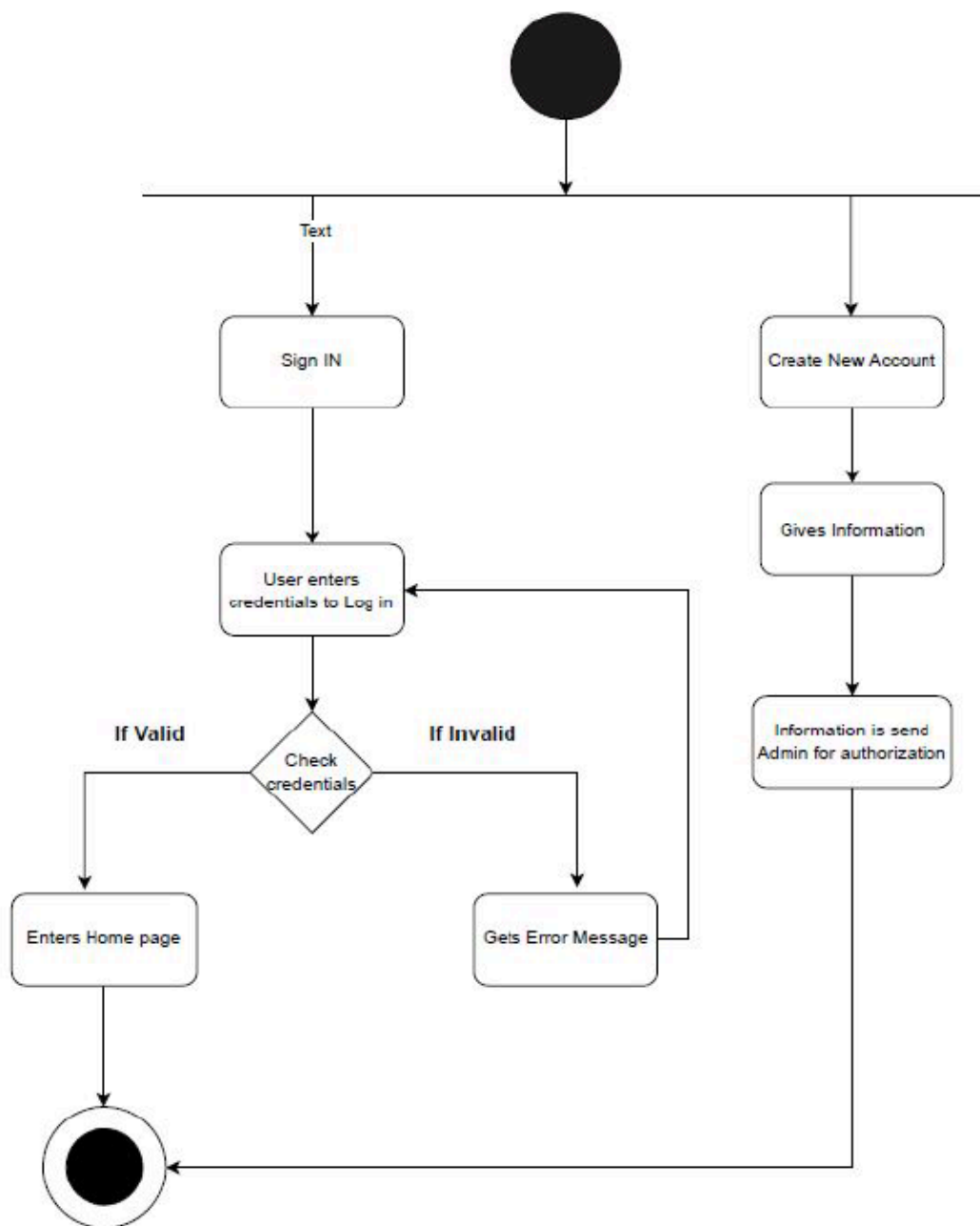
**Activity Diagram for Pay Bill**
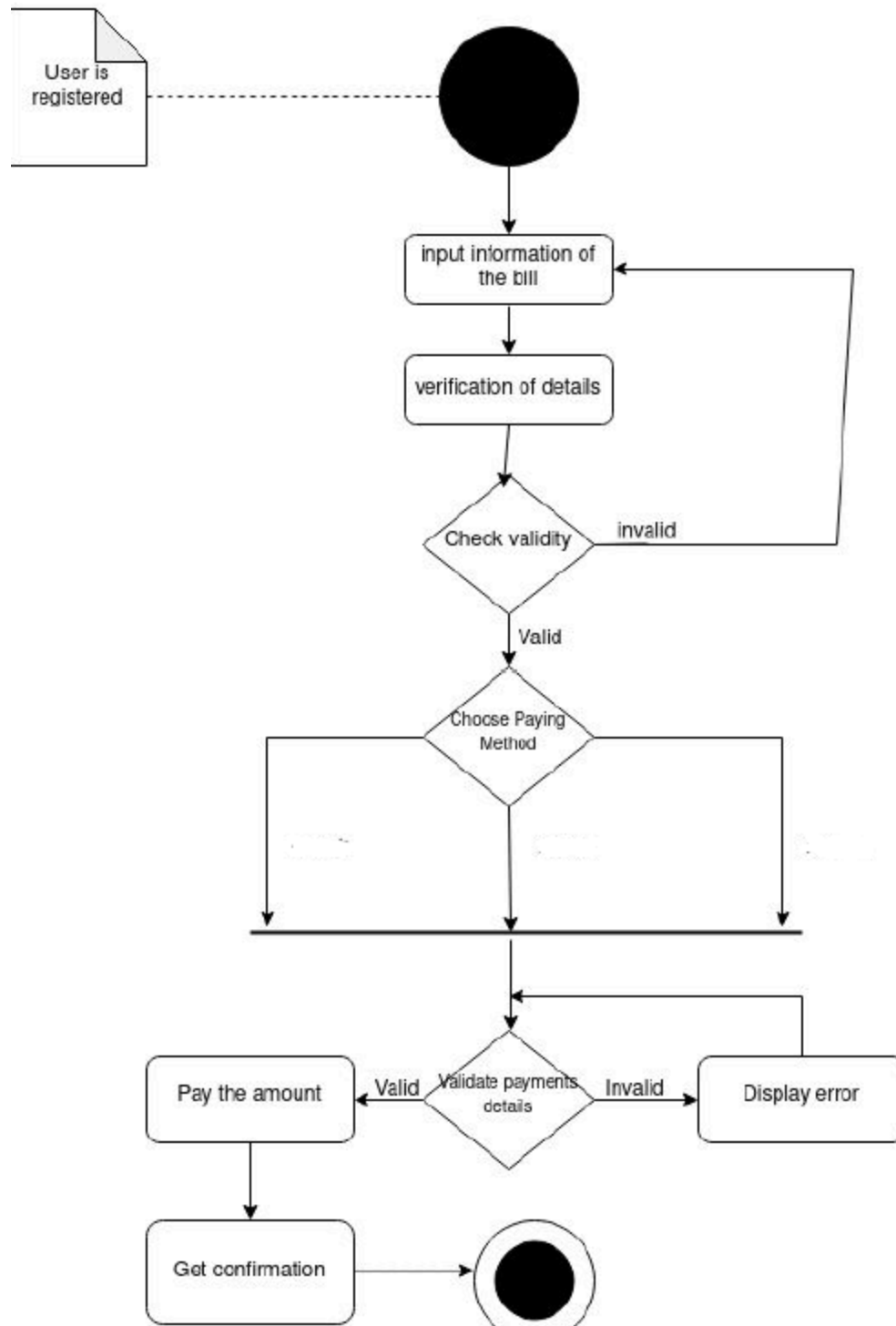
# 3.3 Prototyping using wireframes

## For Passengers



🔍 **Search Flights**

| 🛫 From | 🛬 To | dd/mm/yyyy 📅 |

✈️ Search Flights

✈️ **Flight Booking Options**

| 🛫 Economy | 🎒 Extra Baggage (kg) | 🚫 No Food |
| 📋 Single Route | 🎫 Enter Promo Code | 🏷️ Apply Promo Code |

🚫 Cancel Flight

✏️ Modify Flight

Deployed by **Noob Coders**



🛬 **AirTicket**    🏠 Home  ✈️ Flights  ❓ Help Center  👤 Admin Panel  🕘 Passenger History  🔑 Login

🎫 **Book Your Flight Now**
Find the best deals on airline tickets

🔍 **Search Flights**

| 🛫 From | 🛬 To | dd/mm/yyyy 📅 |

✈️ Search Flights

✈️ **Flight Booking Options**

| 🛫 Economy | 🎒 Extra Baggage (kg) | 🚫 No Food |
| 📋 Single Route | 🎫 Enter Promo Code | 🏷️ Apply Promo Code |

🚫 Cancel Flight

✏️ Modify Flight



🛬 **AirTicket**    🏠 Home  ✈️ Flights  ❓ Help Center  👤 Adm

🎫 **Book Your Flight Now**
Find the best deals on airline tickets

# FOR ADMIN

✈ **AirTicket Admin**  🏠 Dashboard  ✈ Manage Flights  👥 Manage Users  ➡ Logout

## ⚙ Admin Dashboard

| Total Flights 120 | Available Tickets 3000 | Ticket Demand High |
|---|---|---|

| Employees Working Today 50 | Average Customer Rating 4.5/5 | |

## ✈ Manage Flights

Add Flight

| Flight No | From | To | Departure Time | Seats Available | Price ($) | Actions | Notice |
|---|---|---|---|---|---|---|---|
| AI-202 | New York | London | 10:30 AM | 150 | 500 | Cancel Modify Adjust Price | Enter notice<br>Send Notice |

**Deployed by Noob Coders**