ENSF 614: Advanced System Analysis and Software Design

Fall 2025

Group 3 Term Project Design Document

Members:
Cameron Dunn, Ibrahim Khan, Cassius Samaco, Edmund Yu

Submission Date: November 30, 2025

Table of Contents

1. Introduction

The Flight Reservation System is a Java-based desktop application designed to facilitate the end-to-end process of booking airline flights. The system provides a comprehensive platform that connects customers seeking air travel with available flight options while enabling airline staff to manage reservations and flight operations efficiently.

The system supports three distinct user roles with varying levels of access and functionality:

- Customers can search for available flights based on origin, destination, and travel dates; make, modify, and cancel bookings; process payments for reservations; and view monthly promotional news and updates.
- Flight Agents have the ability to manage customer profiles, assist customers with booking operations, and view flight and payment information to provide customer support.
- System Administrators are responsible for managing the flight inventory, including adding new flights, updating existing flight details (such as pricing and seat availability), and removing discontinued routes.

The core entities of the system include

- Flights (with attributes such as airline, origin, destination, departure/arrival times, pricing, and seat availability)
- Bookings (linking customers to specific flights with seat assignments and status tracking)
- Payments (recording transaction details and payment methods) Users (storing authentication credentials and role assignments)
- MonthlyNews (promotional content for registered customers).

Major System Processes

- User Login: Users authenticate by entering their credentials on the login screen. The system verifies the email and password against stored records in the database. Upon successful authentication, the system identifies the user's role and directs them to the appropriate menu (Customer Menu, Agent Menu, or Admin Dashboard). Failed login attempts to display an error message and allow the user to retry. This process is detailed in Activity Diagram 2.1.
- Browsing and Selecting a Flight: Customers search for flights by navigating to the flight search screen and entering their criteria including origin city, destination city, departure date, and optionally a preferred airline. The system queries the flight database and displays matching results in a tabular format showing flight details, available seats, and pricing. If no flights match the criteria, an appropriate message is displayed. Customers can then select a specific flight to proceed with booking. This process is detailed in Activity Diagram 2.2.
- Booking a Flight Ticket: After selecting a flight, the system displays the flight details and prompts the customer to choose an available seat and enter passenger information. Upon confirmation, the system validates seat availability, creates a booking record with a

"Pending" status, decreases the available seat count, and redirects the customer to the payment screen. This process is detailed in Activity Diagram 2.3.

- Making a Payment: The payment process begins after a booking is made. The system displays the amount due and presents a form for credit card details (card number, cardholder name, expiry date, CVV). Upon submission, the system validates the information and processes the transaction. Successful payment updates the booking status to "Confirmed" and generates a booking confirmation displaying the confirmation number, flight details, passenger information, and seat assignment. This process is detailed in Activity Diagram 2.4.

System Architecture

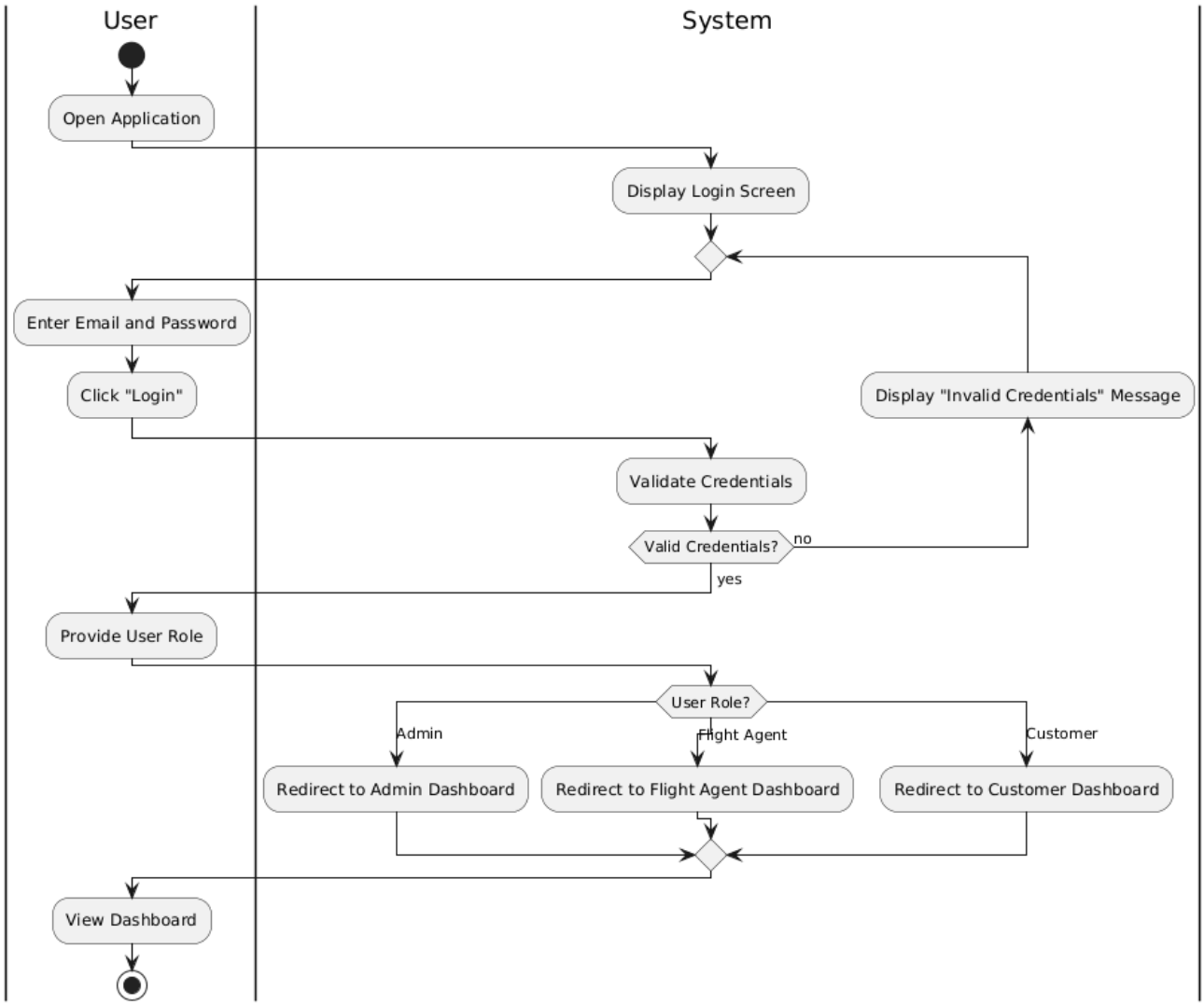
The system utilizes a MySQL relational database for persistent data storage and follows a three-tier layered architecture:

- Data Layer: Handles database connectivity and persistence through the DBConnection class and repository classes (BookingRepository, FlightRepository, PaymentRepository, UserRepository, MonthlyNewsRepository)
- Business Logic Layer: Contains service classes (AuthService, UserService, FlightService, BookingService, PaymentService, NewsService) coordinated by a central AppController
- Presentation Layer: Fifteen Swing-based GUI views including LoginView, RoleSelectionView, role-specific menu views (CustomerMenuView, AgentMenuView, AdminMenuView), and functional views for flight search, booking, payment, and administration
- Design PatternsThe system incorporates several software design patterns to promote code reusability, maintainability, and separation of concerns:
- Singleton Pattern: Both the DBConnection class and the AppController class implement the Singleton pattern. DBConnection ensures a single, shared database connection instance throughout the application, preventing resource exhaustion and connection conflicts. AppController serves as the centralized point of interaction between the presentation layer and the business logic layer, coordinating all service calls.
- Repository Pattern: An abstract Repository<T> class defines a generic interface for standard CRUD (Create, Read, Update, Delete) operations. Concrete repository classes such as FlightRepository, BookingRepository, UserRepository, PaymentRepository, and MonthlyNewsRepository extend this base class to handle persistence for their respective domain entities.
- Strategy Pattern: The payment processing functionality utilizes the Strategy pattern through the processPaymentWithStrategy() method in the AppController, allowing different payment strategies to be applied at runtime for flexible payment processing.
- Observer Pattern: The AppController implements the Observer pattern through the addViewSubscriber() method, enabling views to subscribe for updates and ensuring the user interface remains synchronized with underlying data changes.
- MVC (Model-View-Controller) Architecture: The system follows the MVC architectural pattern, separating domain entities (Flight, Booking, User, Payment, MonthlyNews),

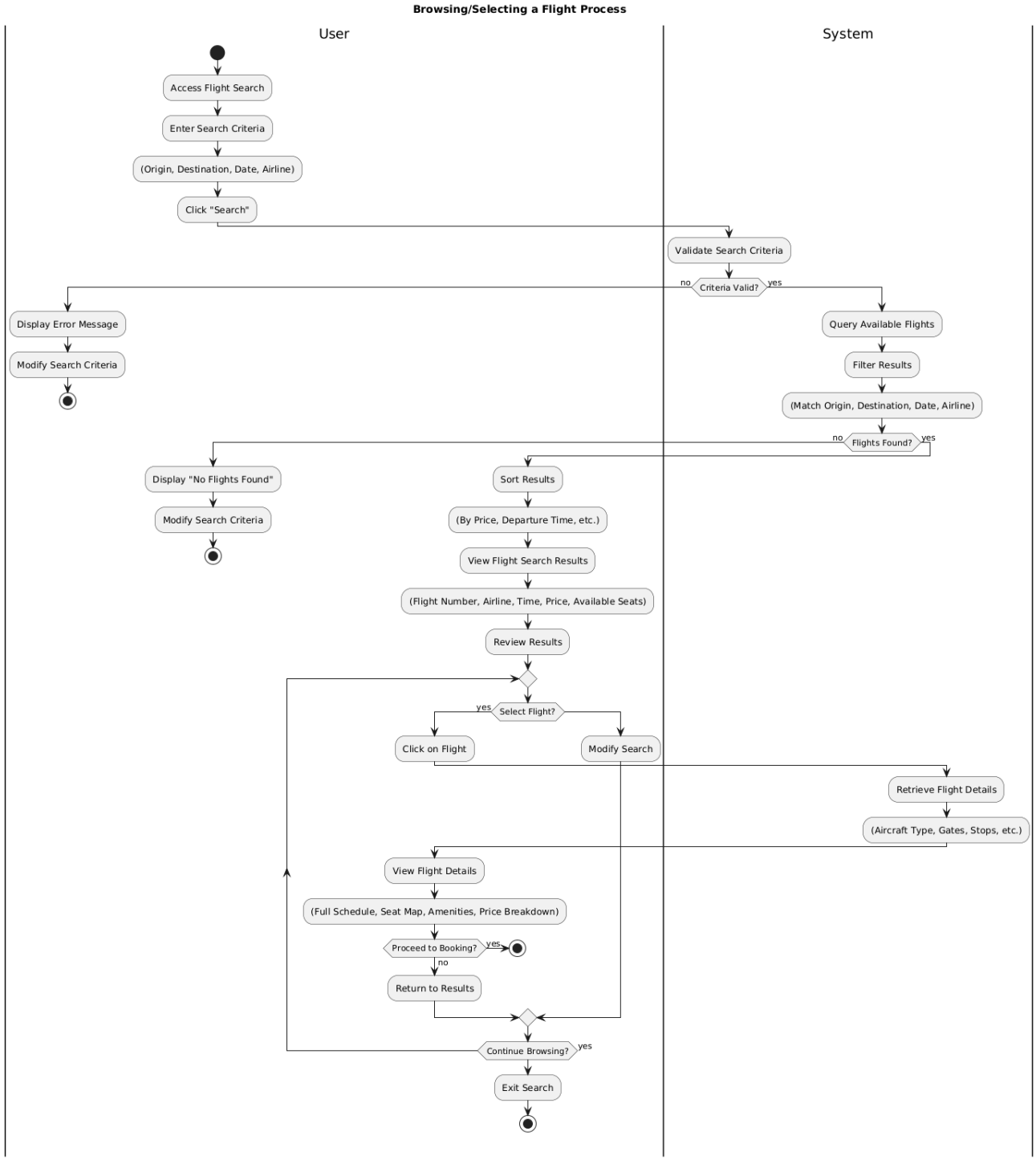
presentation views (15 Swing-based UI components), and controller/service logic, enabling independent development and testing of each layer.

2. Activity Diagrams

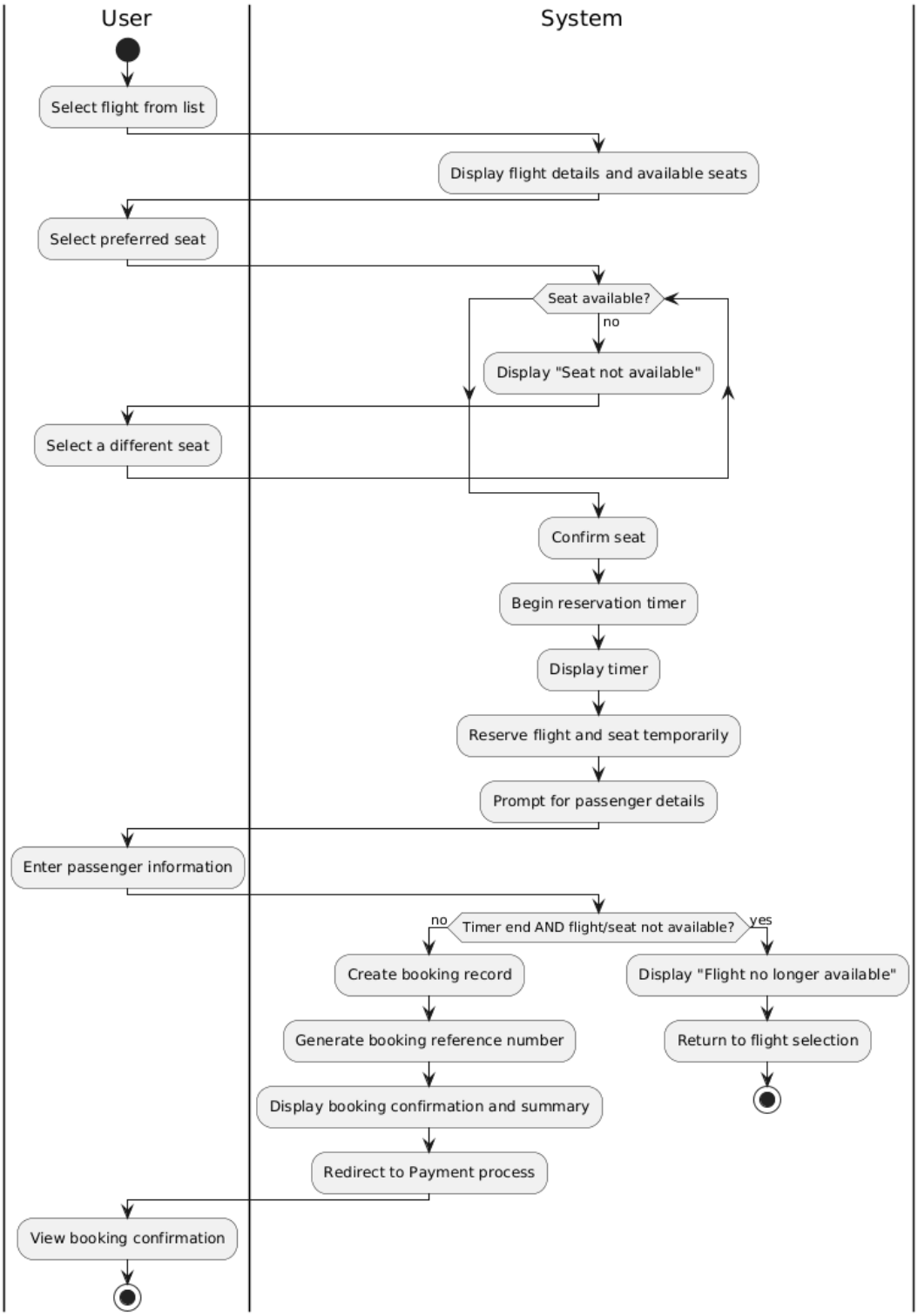
2.1 Activity Diagram – Login



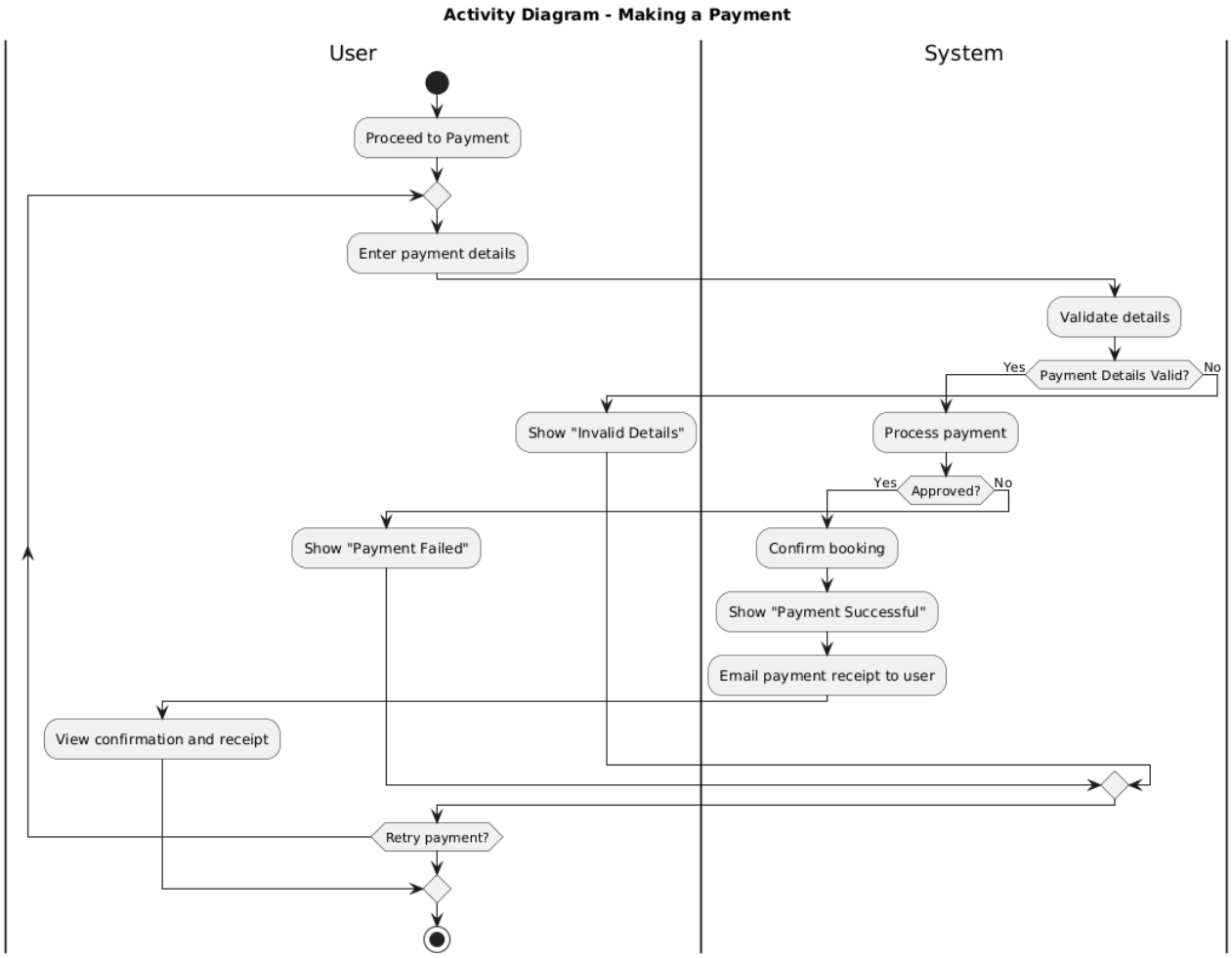
2.2 Activity Diagram – Browse/Select Flight



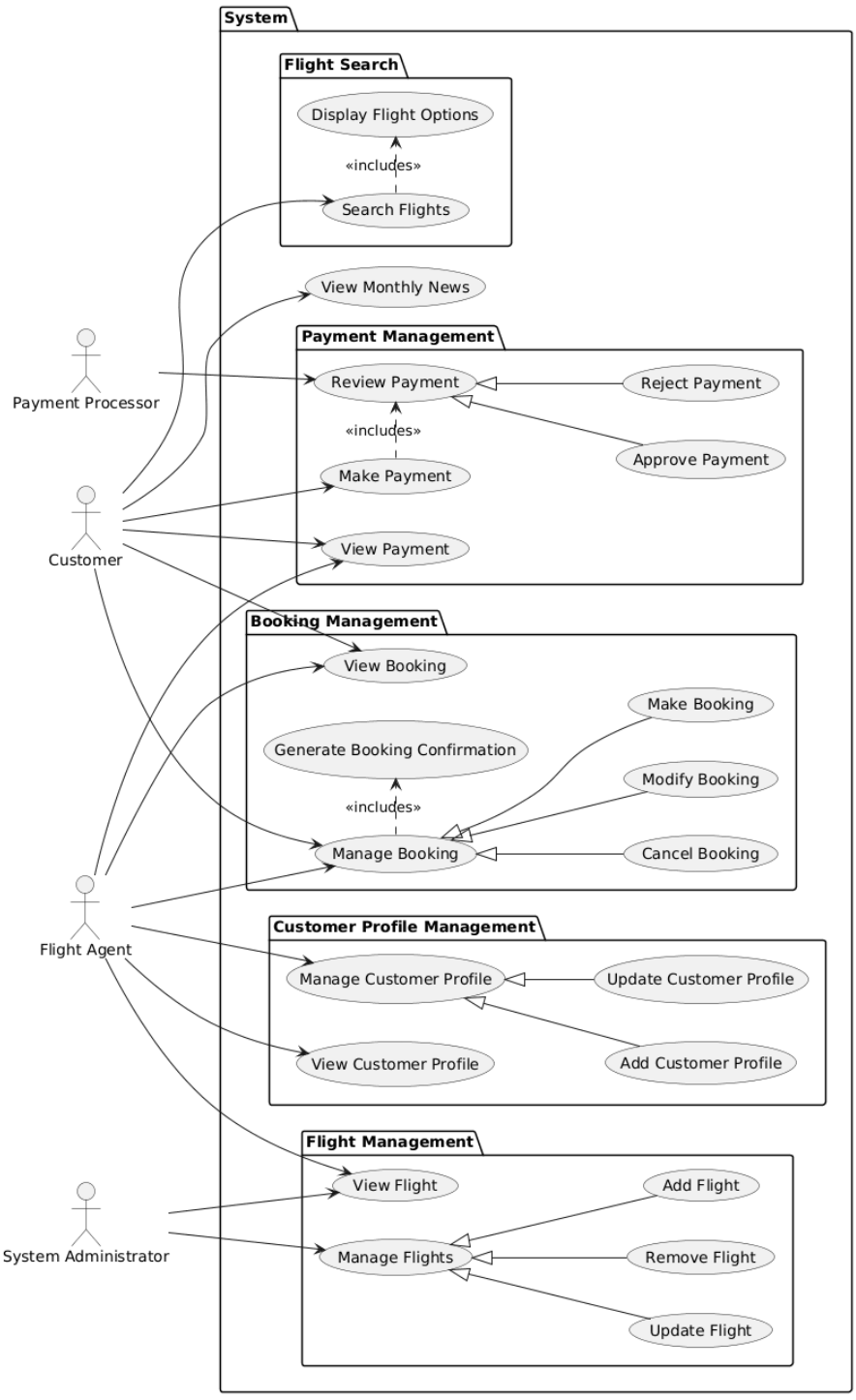
2.3 Activity Diagram – Booking



2.4 Activity Diagram – Payment



3. Systems Use-Case Diagram



4. Use-Case Scenarios

Flight Search

Search Flights (UC1)

Information	Description
Precondition(s)	• Customer must be logged in • Customer must be on home screen
Actor	Customer
Main Scenario	1. Customer selects Search Flights 2. Customer inputs their criteria 3. System triggers Display Flight Options (UC2)
Alternative Scenario	1. Customer selects Search Flights 2. Customer clicks on the back button 3. System returns to the home screen

Display Flight Options (UC2)

Information	Description
Precondition(s)	• Customer must be logged in • Flight options passed by Search Flights (UC1)
Actor	Customer
Main Scenario	1. System gets data and returns a list of results 2. Customer views the options and selects an option 3. System registers the selection and returns a confirmation
Alternative Scenario	1. System finds no data and return “No matching flights” 2. Customer is prompted to go back to Search Flights (UC1) to make a new selection

Booking Management

Manage Booking (UC3)

Information	Description
Precondition(s)	• User must be logged in • User must be on home screen
Actor	Customer, Flight Agent
Main Scenario	1. User selects Manage Bookings 2. If user is Flight Agent, user enters customerID 3. System displays all bookings and available booking management operations (Make new booking, cancel booking, modify booking) 4. User selects one of the operations 5. System follows into one of UC4, UC5, UC6
Alternative Scenario	1. User selects Manage Bookings 2. If user is Flight Agent, user enters customerID 3. System displays all bookings and available booking management operations (Make new booking, cancel booking, modify booking) 4. User clicks on the back button 5. System returns to the home screen

Make Booking (UC4)

Information	Description
Precondition(s)	• User must be logged in • User have selected “Make Booking” from Manage Booking (UC3) • If user is a Flight Agent, a valid customerID must be selected
Actor	Flight Agent, Customer
Main Scenario	1. System displays available flights and prompts for trip details (origin, destination, dates, number of passengers). 2. User enters required trip details. 3. System displays matching flights and fare options. 4. User selects a flight and fare. 5. System displays booking summary and prompts for confirmation.

	6. User confirms the booking. 7. System creates the booking record. 8. System transitions to Generate Booking Confirmation (UC7).
Alternative Scenario	1. User selects Make New Booking. 2. System displays flight search interface. 3. User clicks Back. 4. System returns to the Manage Bookings (UC3).

Cancel Booking (UC5)

Information	Description
Precondition(s)	• User must be logged in • User have selected "Cancel Booking" from Manage Booking (UC3) • If user is a Flight Agent, a valid customerID must be selected • A booking with the customerID must exist
Actor	Flight Agent, Customer
Main Scenario	1. System displays a list of active bookings. 2. User selects a booking to cancel. 3. System displays cancellation details, including refund/penalty information. 4. User confirms cancellation. 5. System cancels the booking and updates records. 6. System transitions to Generate Booking Confirmation (UC7) for cancellation notice.
Alternative Scenario	1. System displays active bookings. 2. User clicks Back. 3. System returns to the Manage Bookings screen.

Modify Booking (UC6)

Information	Description
Precondition(s)	• User must be logged in • User have selected "Modify Booking" from Manage Booking (UC3) • If user is a Flight Agent, a valid customerID must be selected • A booking with the customerID must exist
Actor	Flight Agent, Customer
Main Scenario	1. System displays a list of bookings. 2. User selects a booking to modify.

	3. System displays modification options (flight change, passenger info change, seat selection, etc.). 4. User selects a modification option and enters updated information. 5. If "flight change" option chosen, new flight options are displayed for selection. 6. System validates changes and shows updated booking summary. 7. User confirms modifications. 8. System updates the booking record. 9. System transitions to Generate Booking Confirmation (UC7) for modification notice.
Alternative Scenario	1. System displays active bookings. 2. User clicks Back. 3. System returns to the Manage Bookings screen.

Generate Booking Confirmation (UC7)

Information	Description
Precondition(s)	• A booking has just been created, modified, or canceled • Triggered automatically from UC4, UC5, or UC6
Actor	Customer, Flight Agent
Main Scenario	1. System compiles booking details (new, modified, or canceled). 2. System generates a booking confirmation document (email or PDF). 3. System sends the confirmation to the customer's registered contact information. 4. System displays a success message to the user.
Alternative Scenario	1. System compiles booking details (new, modified, or canceled). 2. System generates a booking confirmation document (email or PDF). 3. A delivery error occurs (e.g., invalid email). 4. System logs the error and displays a notification instructing the user to update contact details.

View Booking (UC8)

Information	Description
Precondition(s)	• User must be logged in • User must be on the home screen
Actor	Customer, Flight Agent
Main Scenario	1. User selects View Booking. 2. If user is a Flight Agent, system prompts for and validates customerID. 3. User enters booking reference number. 4. System retrieves booking details. 5. System displays full booking information (flight, passenger details, payment, status).
Alternative Scenario	1. User selects View Booking. 2. User clicks back 3. System returned to home page

Customer Profile Management

Manage Customer Profile (UC9)

Information	Description
Precondition(s)	• Flight Agent must be logged in • Flight Agent must be on home screen
Actor	Flight Agent
Main Scenario	1. Flight Agent selects Customer Profile Management 2. System displays all profiles and available profile operations (Add Profile, Update Profile, View Profile) 3. Flight Agent selects one of the operations 4. System follows into one of UC9, UC10, UC11
Alternative Scenario	1. Flight Agent selects Customer Profile Management 2. System displays all profiles and available profile operations (Add Profile, Update Profile, View Profile) 3. Flight Agent clicks on the back button 4. System returns to the home screen

Add Customer Profile (UC10)

Information	Description

Precondition(s)	• Flight Agent must be logged in • Flight Agent have selected “Add Profile” from Manage Customer Profile (UC8)
Actor	Flight Agent
Main Scenario	1. Flight Agent enters customer details 2. Flight Agent clicks on create button 3. System validates, stores customer details and responds with a confirmation message 4. Flight Agent returns to Manage Customer Profile page
Alternative Scenario	1. Flight Agent enters customer details 2. Flight Agent clicks on create button 3. System determines an error with the customer details and sends an error message 4. Flight Agent reattempts or returns to Manage Customer Profile page

Update Customer Profile (UC11)

Information	Description
Precondition(s)	• Flight Agent must be logged in • Flight Agent have selected “Update Profile” from Manage Customer Profile (UC8)
Actor	Flight Agent
Main Scenario	1. Flight Agent enters customer details 2. Flight Agent clicks on update button 3. System validates, updates customer details and responds with a confirmation message 4. Flight Agent returns to Manage Customer Profile page
Alternative Scenario	1. Flight Agent enters customer details 2. Flight Agent clicks on create button 3. System determines an error with the customer details and sends an error message 4. Flight agent either reattempts or returns to Manage Customer Profile Page

View Customer Profile (UC12)

Information	Description
Precondition(s)	• Flight Agent must be logged in

	• Flight Agent have selected “View Profile” from Manage Customer Profile (UC8)
Actor	Flight Agent
Main Scenario	1. Flight Agent views all customer details 2. Flight Agent returns to Manage Customer Profile page
Alternative Scenario	

Flight Management

Manage Flights (UC13)

Information	Description
Precondition(s)	• System Administrator must be logged in. • System Administrator must be on the Admin Dashboard.
Actor	System Administrator
Main Scenario	1. System Administrator selects "Flight Management" from the dashboard. 2. System displays all flight management options (Add Flight, Update Flight, Remove Flight, View Flight). 3. System Administrator selects one of the operations. 4. System redirects to the selected use case (UC13, UC14, UC15, or UC16).
Alternative Scenario	1. System Administrator selects "Flight Management". 2. System displays flight management options. 3. System Administrator clicks the "Back" button. 4. System returns to the Admin Dashboard.

Add Flight (UC14)

Information	Description
Precondition	• System Administrator must be logged in.

	• System Administrator must have selected "Add Flight" from Manage Flights (UC12).
Actor	System Administrator
Main Scenario	1. System displays a form to enter new flight details (e.g., flight number, origin, destination, time, aircraft type, price). 2. System Administrator enters the flight details. 3. System Administrator clicks the "Create" button. 4. System validates the input, creates the new flight record in the database, and displays a success confirmation. 5. System returns to the Manage Flights screen.
Alternative Scenario	1. System Administrator enters flight details and clicks "Create". 2. System determines the flight number already exists or data is invalid. 3. System displays an error message indicating the invalid fields. 4. System Administrator creates a corrected entry or cancels the operation to return to the Manage Flights screen.

Update Flight (UC15)

Information	Description
Precondition	1. System Administrator must be logged in. 2. System Administrator must have selected "Update Flight" from Manage Flights (UC12).
Actor	System Administrator
Main Scenario	1. System displays a list of existing flights. 2. System Administrator selects a specific flight to edit. 3. System displays the current details of the flight in an editable form. 4. System Administrator modifies the necessary details.

	5. System Administrator clicks the "Update" button. 6. System validates the changes, updates the database, and displays a success confirmation.
Alternative Scenario	1. System Administrator modifies details and clicks "Update". 2. System detects an error. 3. System displays an error message. 4. System Administrator corrects the data and re-submits or clicks "Cancel" to abort changes.

Remove Flight (UC16)

Information	Description
Precondition	1. System Administrator must be logged in. 2. System Administrator must have selected "Remove Flight" from Manage Flights (UC12).
Actor	System Administrator
Main Scenario	1. System displays a list of existing flights. 2. System Administrator selects the flight to remove. 3. System prompts for confirmation to delete the flight. 4. System Administrator confirms the deletion. 5. System removes the flight record from the database and displays a success message.
Alternative Scenario	1. System Administrator selects a flight to remove. 2. System checks constraints and finds active bookings associated with this flight. 3. System prevents deletion and displays an error message: "Cannot remove flight with active reservations." 4. System Administrator acknowledges and returns to the flight list.

View Flight (UC17)

Information	Description

Precondition	1. User must be logged in as System Administrator OR Flight Agent. 2. User must have selected "View Flight" from their respective dashboard/menu.
Actor	System Administrator, Flight Agent
Main Scenario	1. System displays a search bar and a list of all flights. 2. User enters a flight number or selects a flight from the list. 3. System displays the full details of the selected flight (Schedule, Aircraft, Route, Current Bookings). 4. User reviews the information. 5. User clicks "Back" to return to the flight list.
Alternative Scenario	1. User enters a flight number in the search bar. 2. System cannot find a match for the entered flight number. 3. System displays a "Flight not found" message. 4. User re-enters criteria or clears the search.

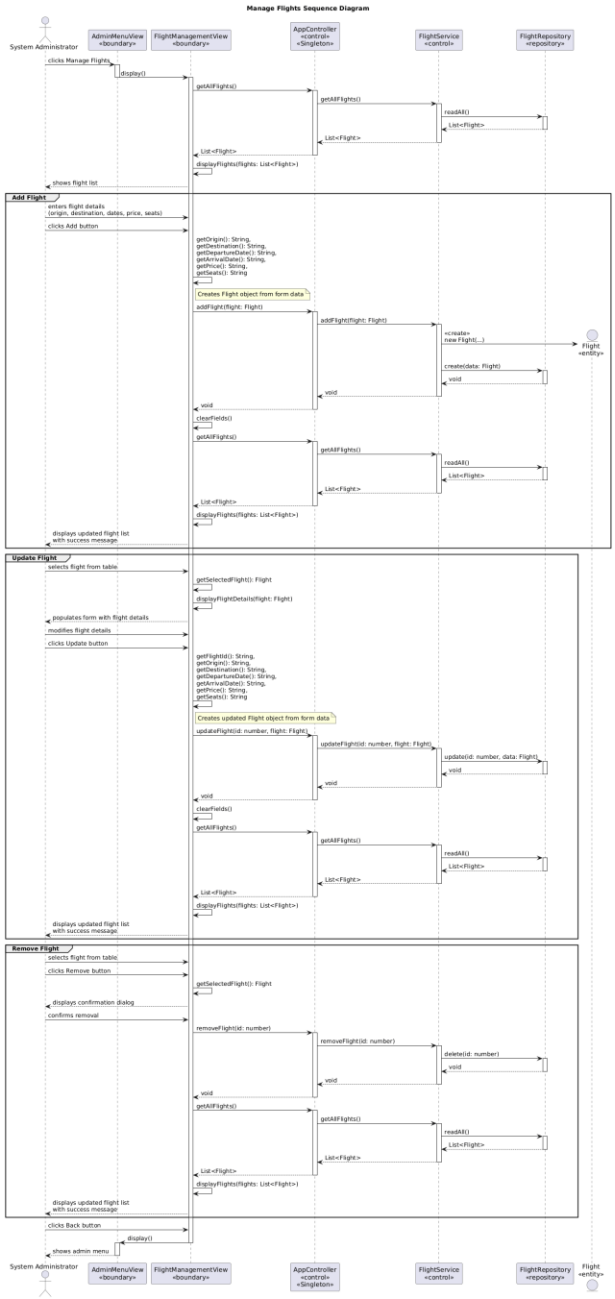
News

View Monthly News (UC18)

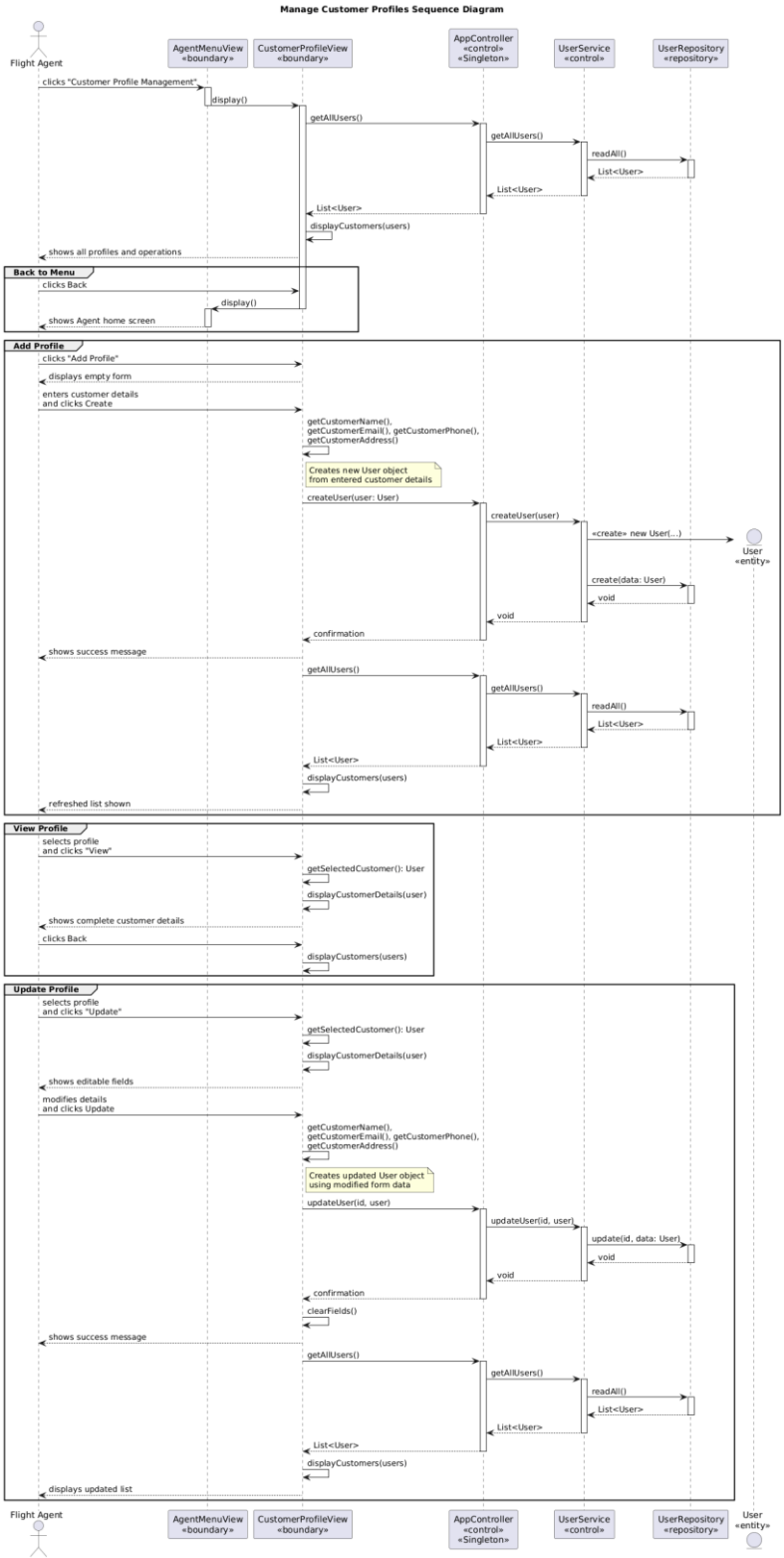
Information	Description
Precondition(s)	• Customer must be logged in • Customer must be on home screen
Actor	Customer
Main Scenario	1. Customer selects View Monthly News 2. System shows monthly news
Alternative Scenario	1. Customer selects View Monthly News 2. System can’t find news and shows “No new news”

5. Sequence Diagrams

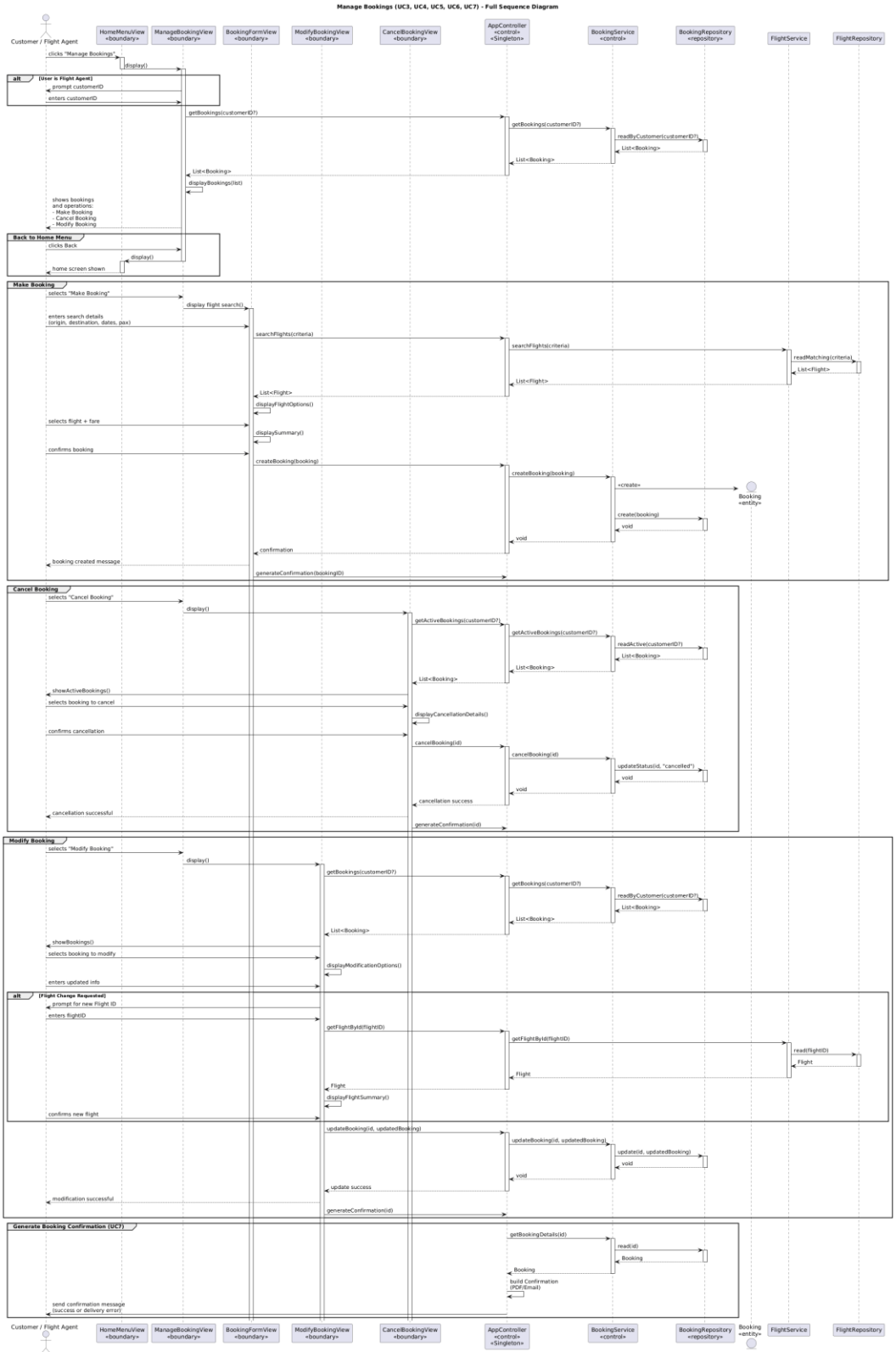
5.1 Sequence Diagram – Manage Flights



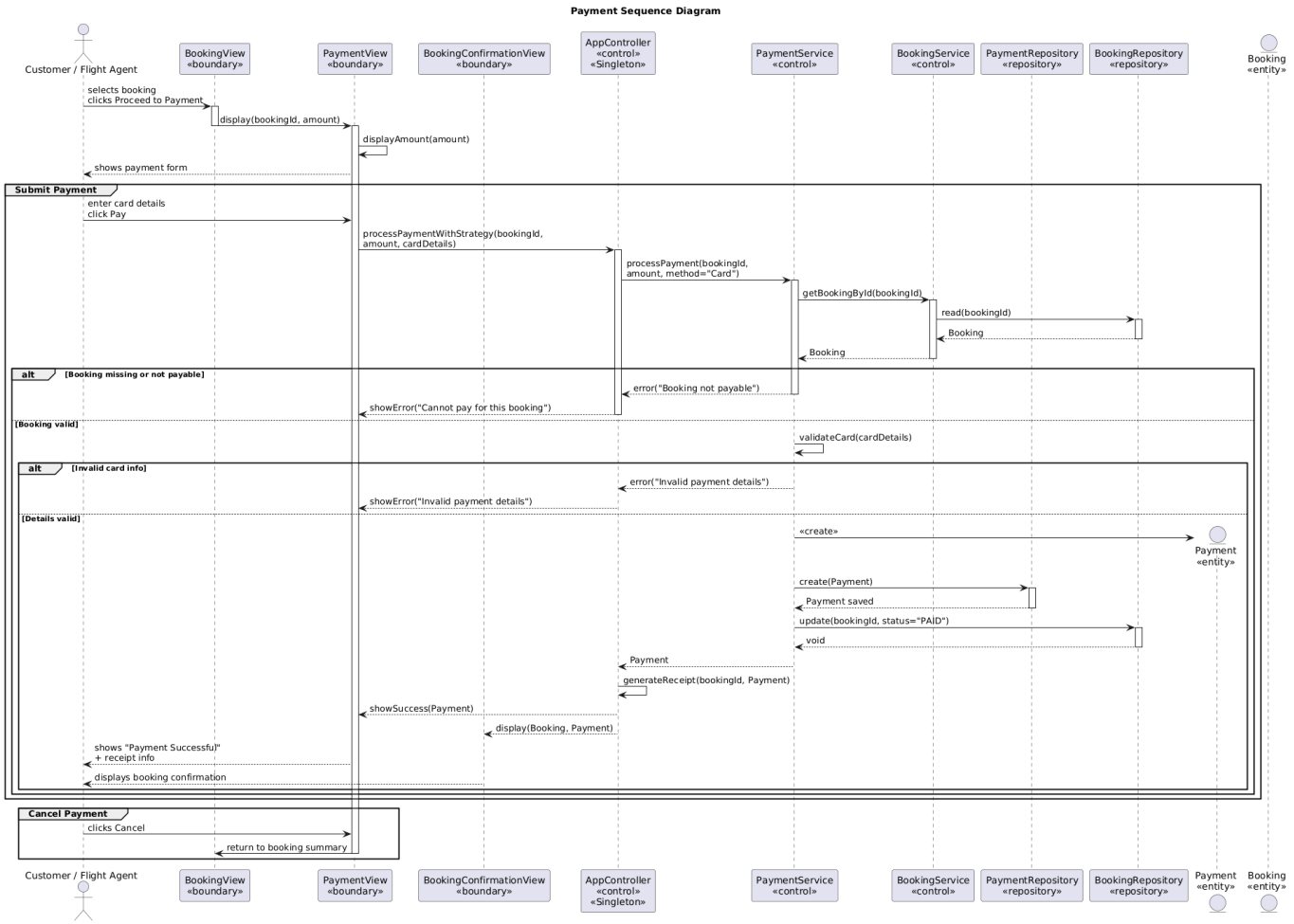
5.2 Sequence Diagram – Customer Profile



5.3 Sequence Diagram – Manage Booking

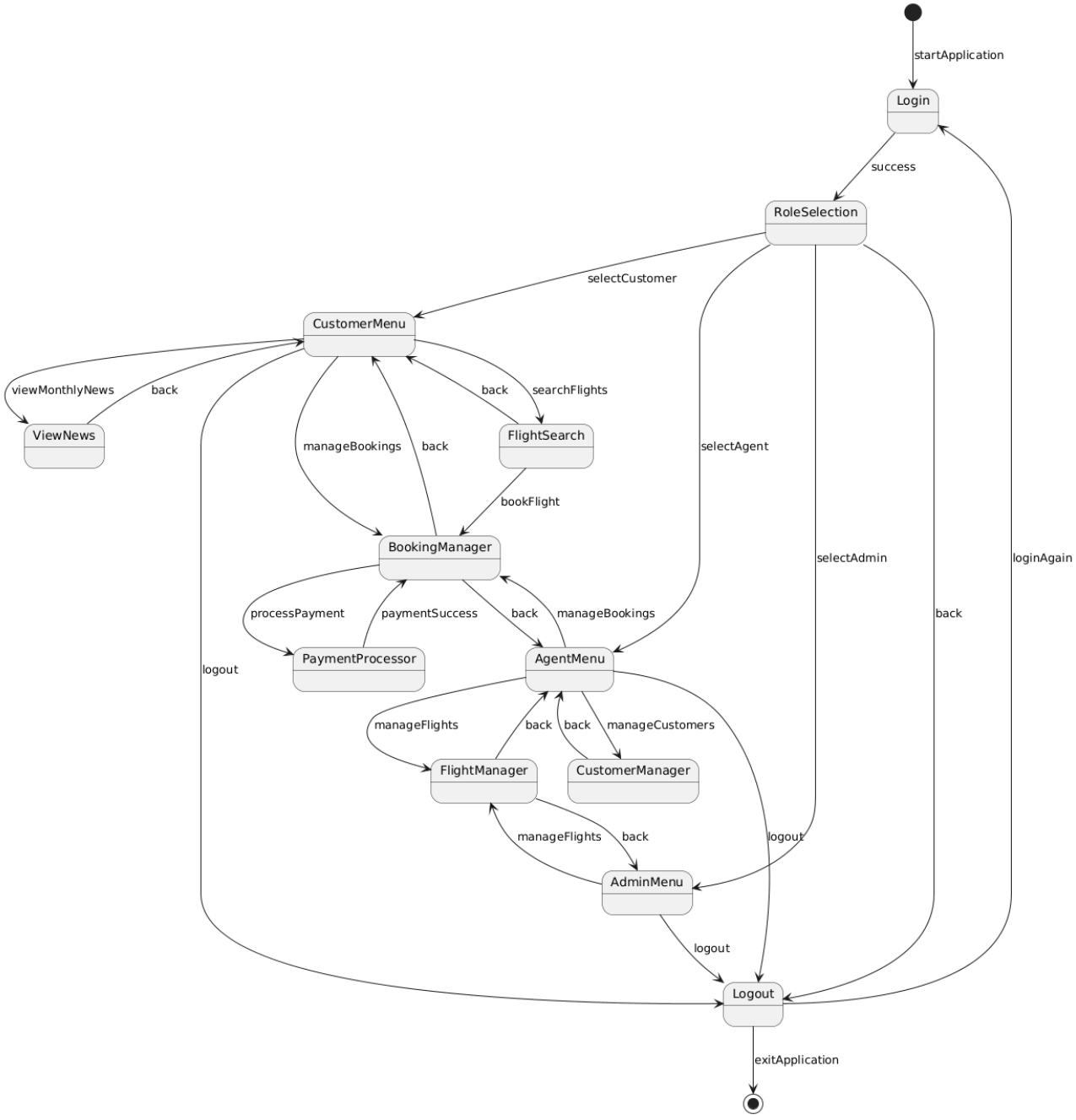


5.4 Sequence Diagram – Payment

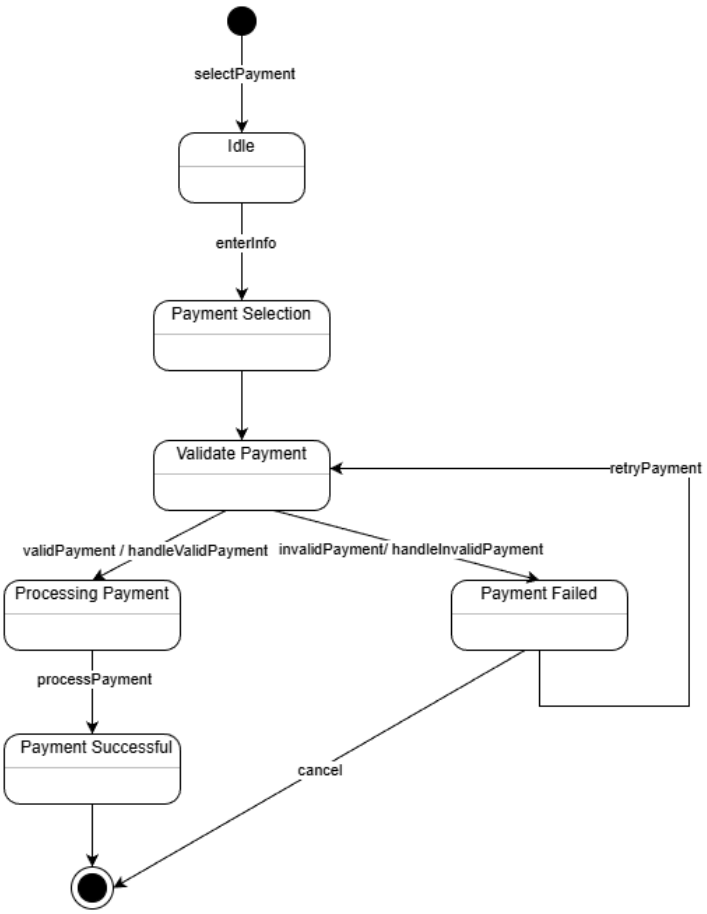


6. State Diagrams

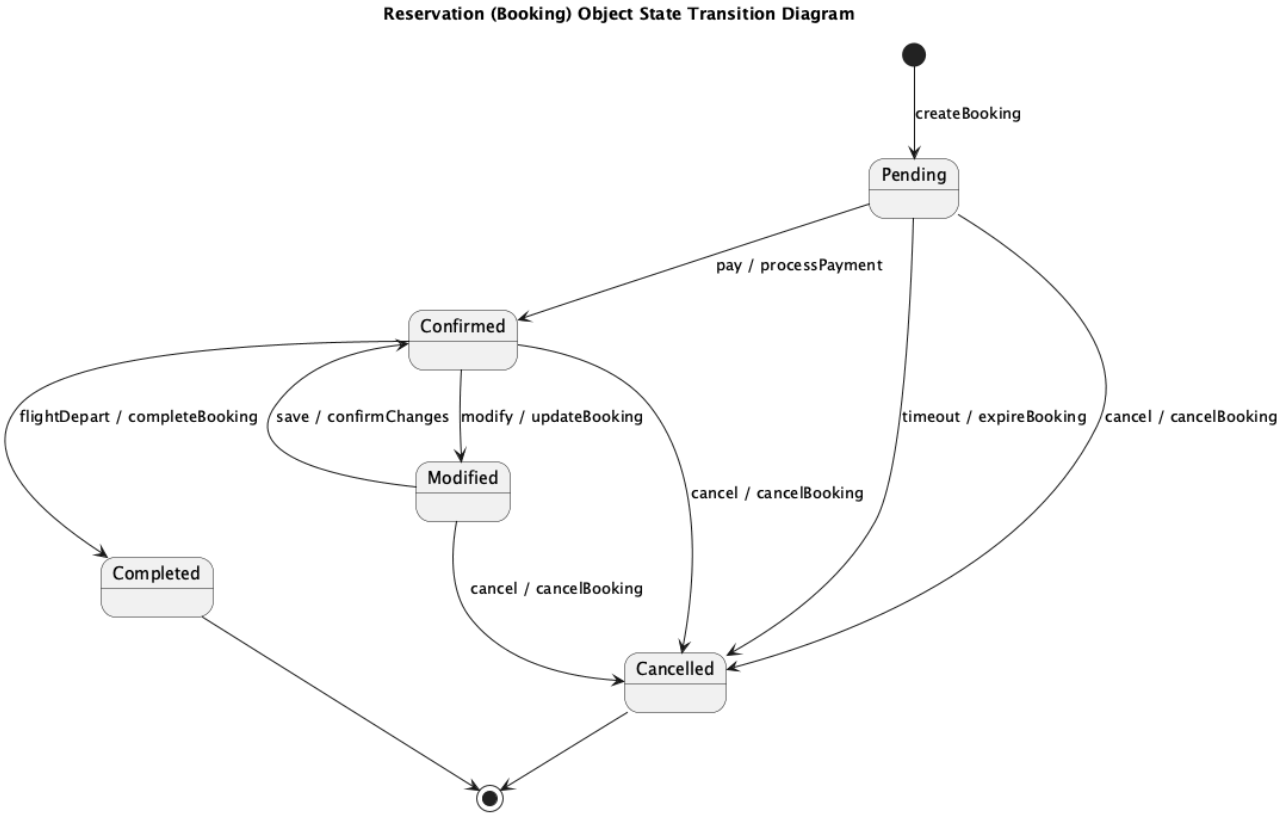
6.1 State Diagram – System



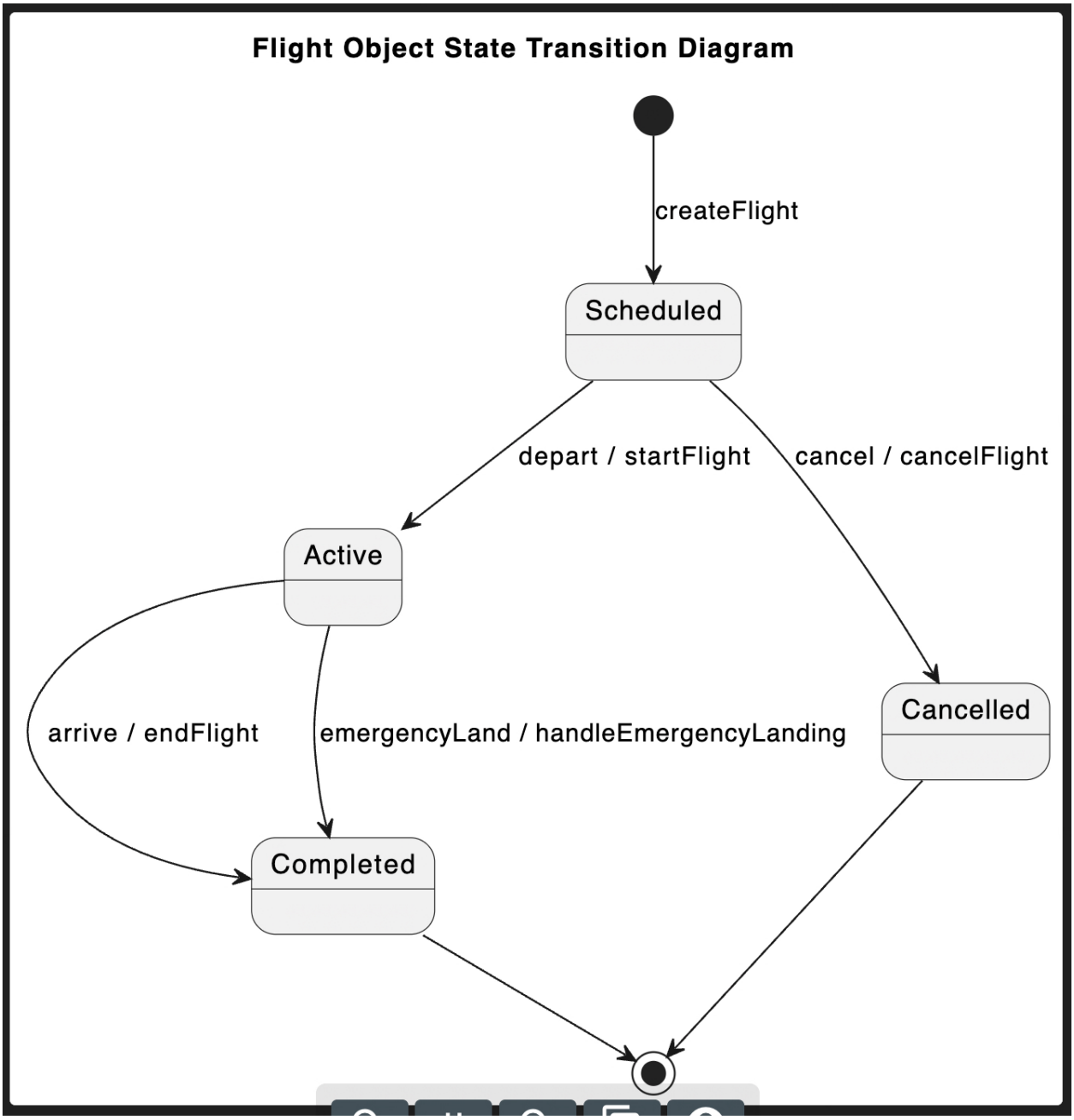
6.2 State Diagram – Payment



6.3 State Diagram – Reservation



6.4 State Diagram – Flight



7. Class Diagram

The class diagram is provided as a separate PNG file due to its large size, which makes the attributes difficult to read when embedded directly in the document.

8. Package Diagram