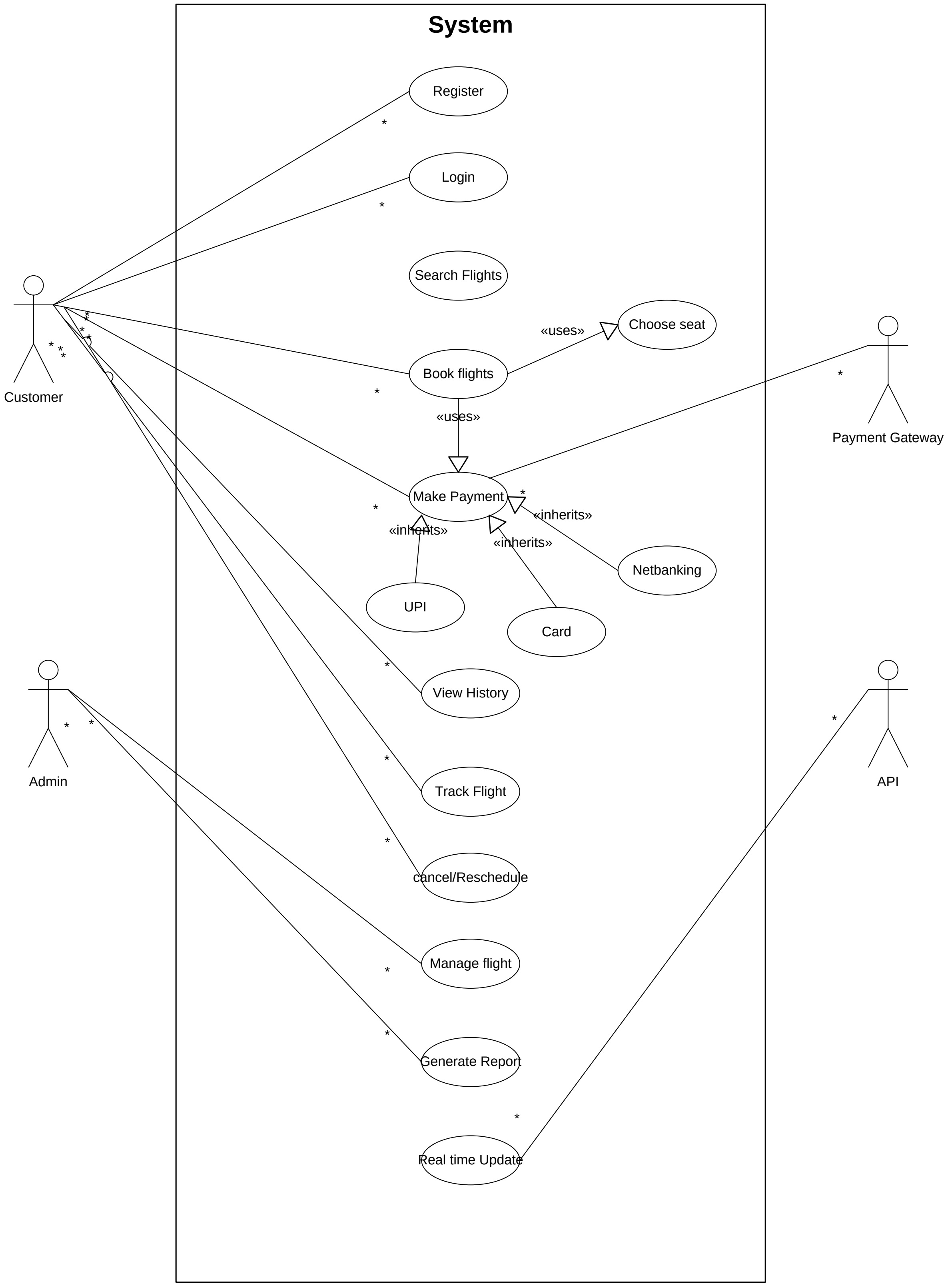
Live Project 1 Part 2

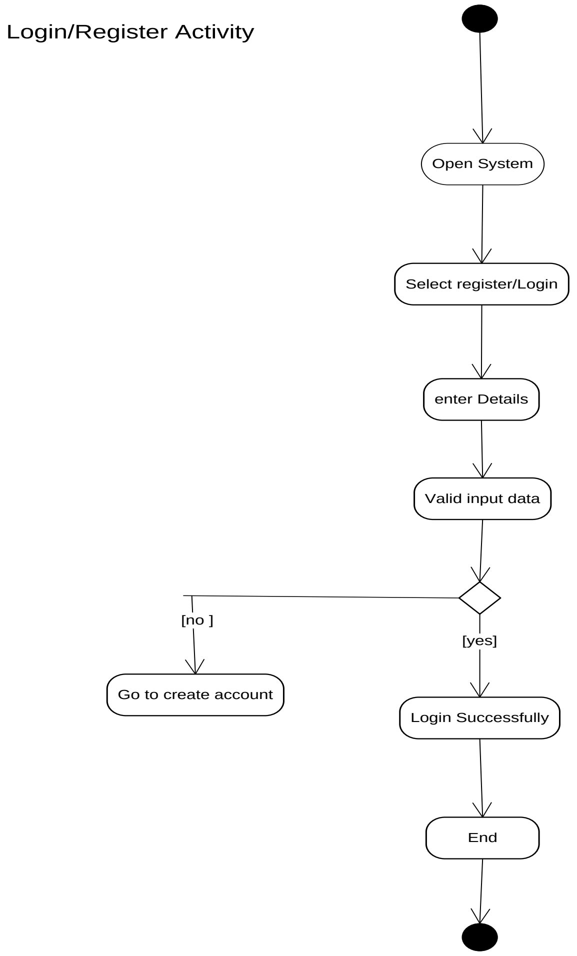
Document 6- Please prepare a use case diagram, activity diagram and a use case specification document.

Use Case Diagram



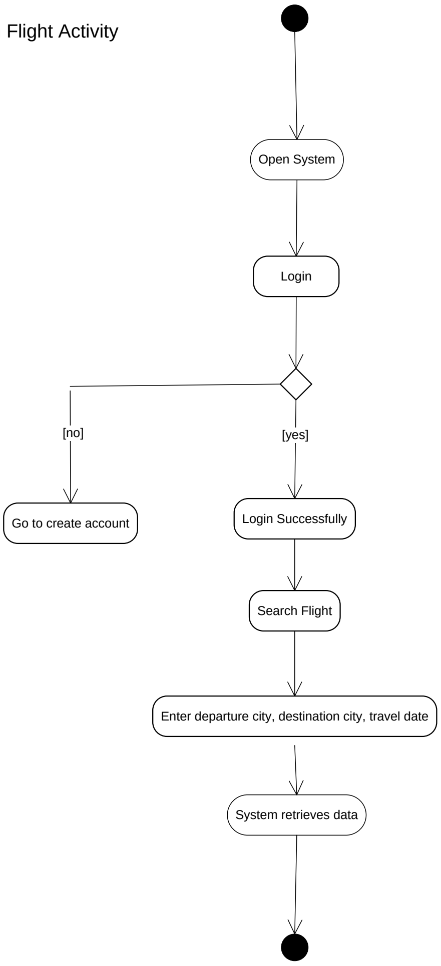
Activity diagram

1.Login /Register Activity



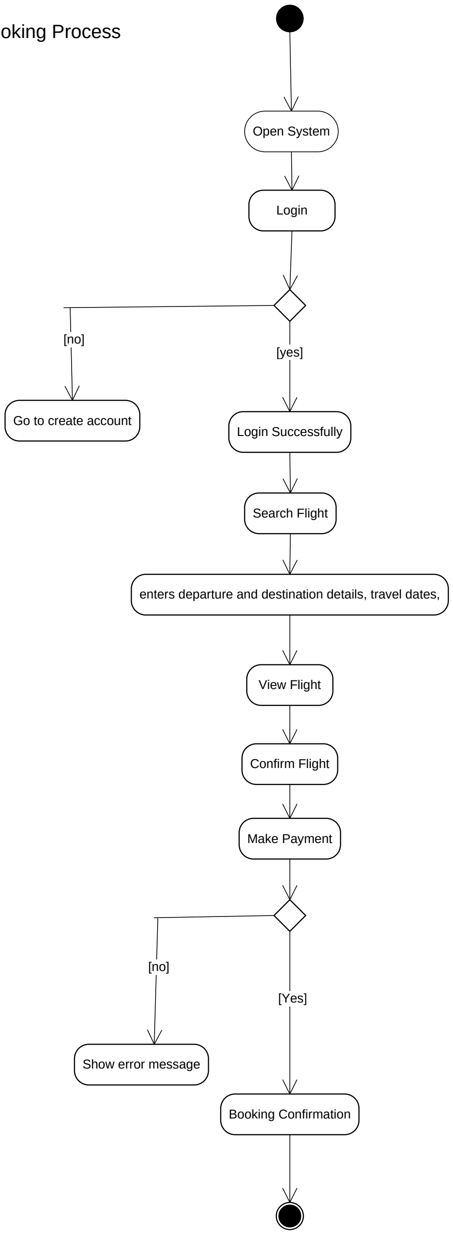
2.

Search Flight Activity



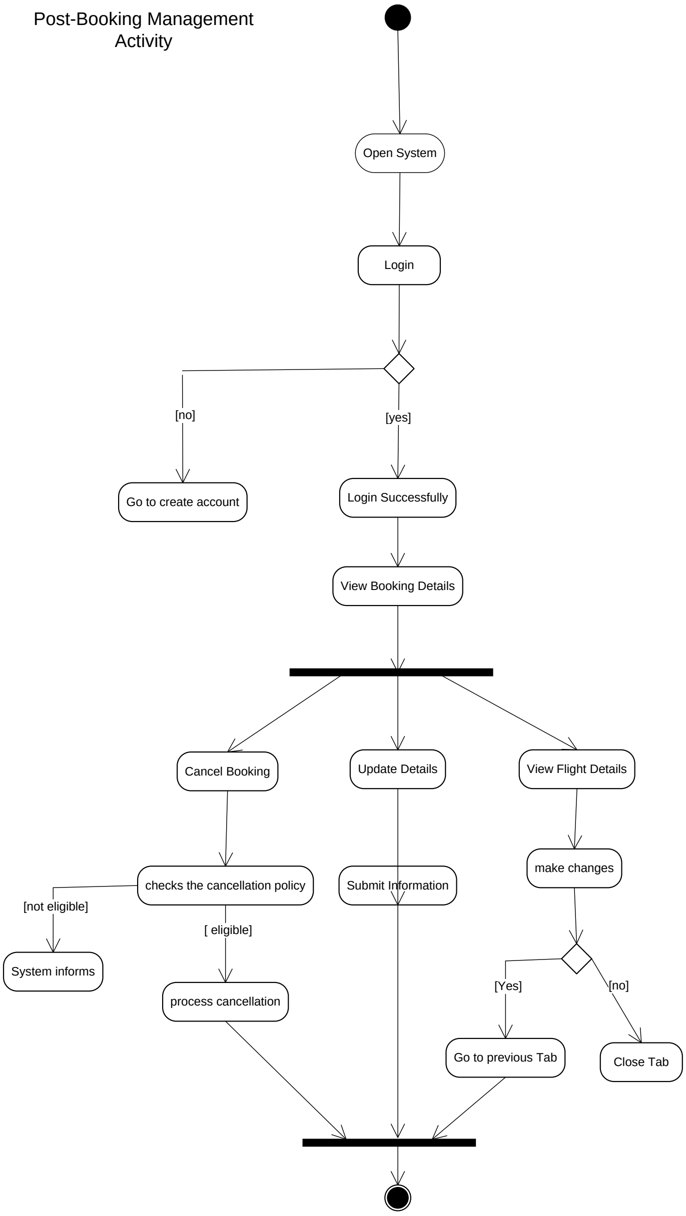
3.

Flight Booking Process

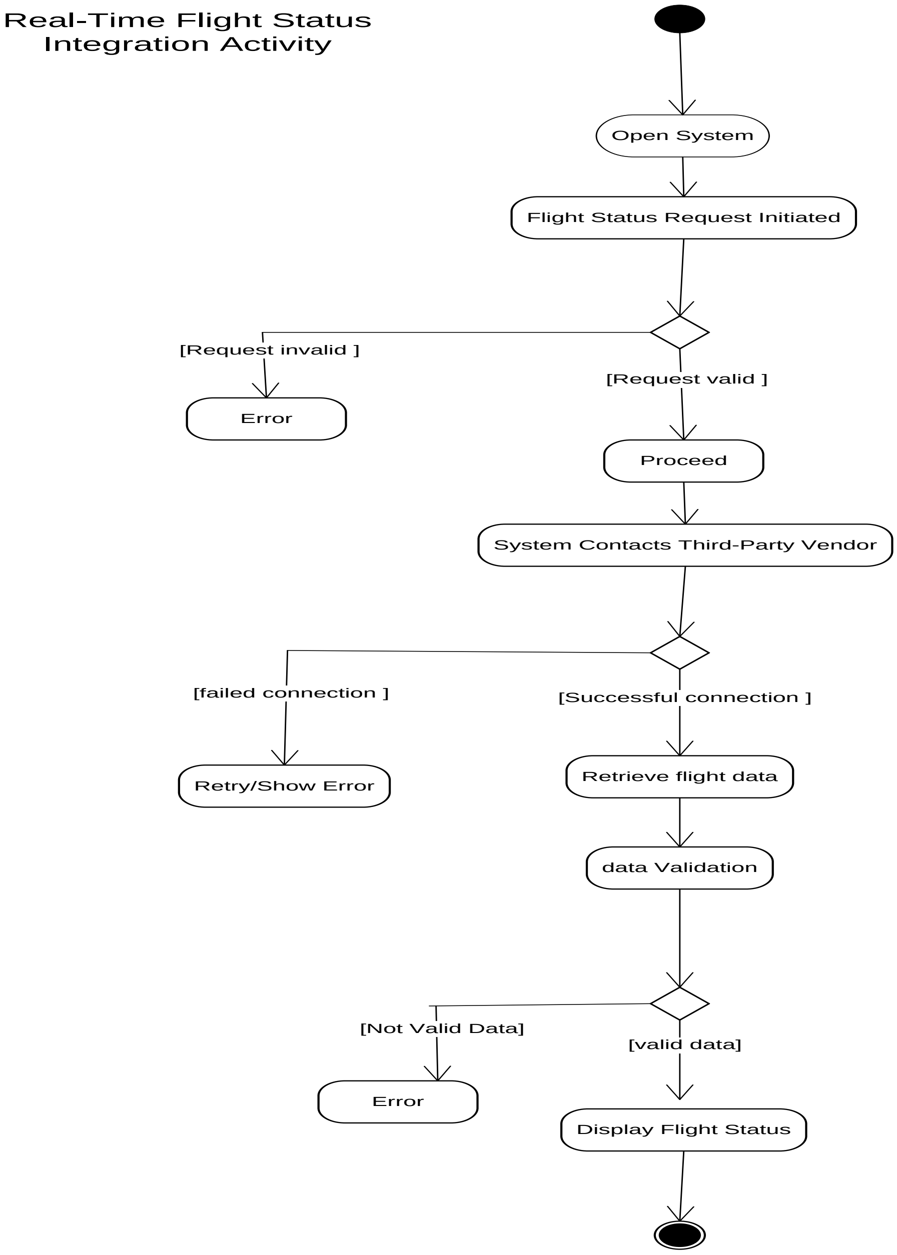


4.

Post-Booking Management Activity

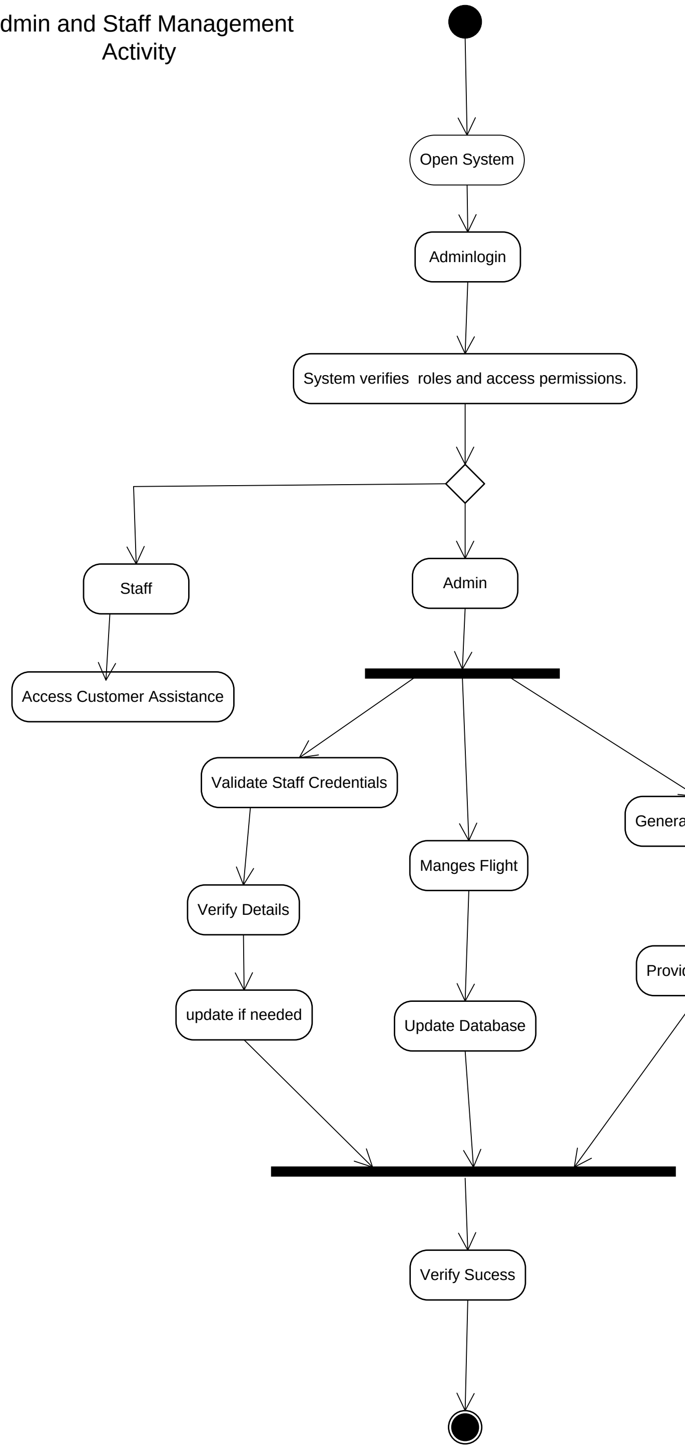


5.

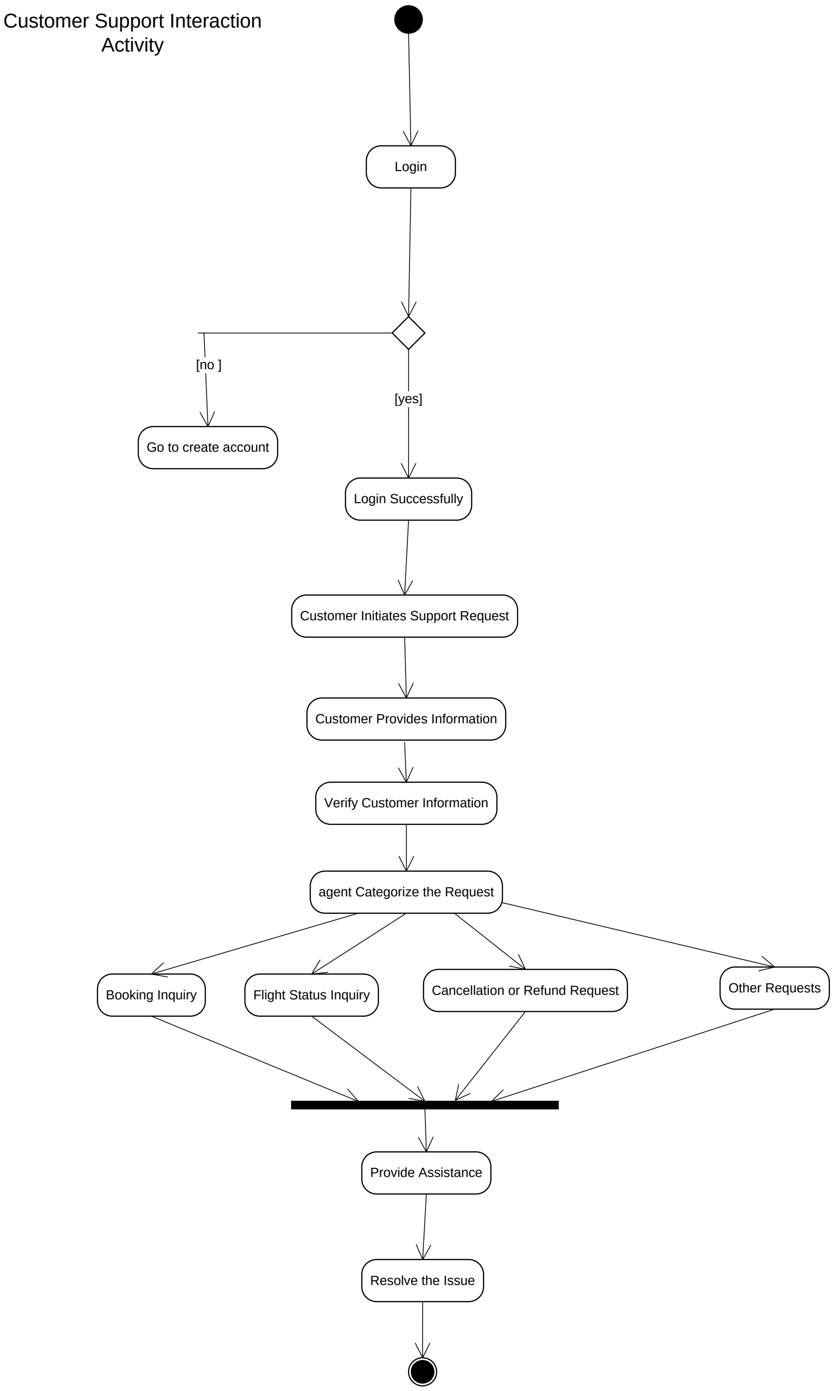

Real-Time Flight Status
Integration Activity
Open System
Flight Status Request Initiated
[Request invalid]
Error
[Request valid]
Proceed
System Contacts Third-Party Vendor
[failed connection]
Retry/Show Error
[Successful connection]
Retrieve flight data
data Validation
[Not Valid Data]
Error
[valid data]
Display Flight Status

6.

Admin and Staff Management Activity



7.



Customer Support Interaction Activity

Login
[no]
[yes]
Go to create account
Login Successfully
Customer Initiates Support Request
Customer Provides Information
Verify Customer Information
agent Categorize the Request
Booking Inquiry
Flight Status Inquiry
Cancellation or Refund Request
Other Requests
Provide Assistance
Resolve the Issue

Use case specification document –

1. Register Account

Attribute	**Description**
Use Case Name	Register Account
Use Case Description	The customer registers for an account on the platform by providing personal details to generate a unique username and password.
Primary Actors	Customer
Secondary Actors	System
Basic Flow	1. Customer navigates to the registration page. 2. Customer enters personal details (name, email, etc.). 3. The system validates the input and creates an account. 4. A unique username and password are generated and provided to the customer.
Alternate Flow	If the customer is new, they are prompted to create an account before proceeding with further actions.
Exceptional Flows	If the system detects duplicate or invalid information (e.g., email already exists), it prompts the customer to correct it.
Pre-Conditions	The customer is not registered in the system.
Post-Conditions	A new user account is created, and login credentials are provided.
Assumptions	The customer has internet access to complete the registration process.
Constraints	Registration requires valid email and phone number.
Dependencies	The system must have a user database and a backend for validating and storing account details.
Inputs and Outputs	Input: Personal details (name, email, etc.) Output: Unique username and password.
Business Rules	The username must be unique.
Miscellaneous Information	The registration process must be secure, ensuring user data is stored safely.

2. Login

Attribute	Description
Use Case Name	Login
Use Case Description	The customer logs into their account using the username and password to access the system and make transactions.
Primary Actors	Customer
Secondary Actors	System
Basic Flow	1. Customer navigates to the login page. 2. Customer enters username and password. 3. The system validates the credentials and grants access to the customer.
Alternate Flow	If credentials are incorrect, the system prompts the user to re-enter them or request password recovery.
Exceptional Flows	If the username/password is incorrect three times, the account is temporarily locked, and the customer is asked to reset their password.
Pre-Conditions	The customer must have a registered account.
Post-Conditions	The customer is successfully logged in to the system.
Assumptions	The customer remembers their login credentials.
Constraints	The system must enforce secure password storage and login attempts to prevent brute force attacks.
Dependencies	The system must have a login authentication mechanism in place.
Inputs and Outputs	Input: Username and password Output: Login success or error message.
Business Rules	Password must be encrypted and never stored in plaintext.
Miscellaneous Information	N/A

3. Search Flight Options

Attribute	**Description**
Use Case Name	Search Flight Options
Use Case Description	Customer searches for available flight options based on input criteria such as date, destination, and number of passengers.
Primary Actors	Customer
Secondary Actors	System
Basic Flow	1. Customer enters search criteria (date, destination, passengers). 2. The system queries the database for available flights. 3. The system displays matching flights.
Alternate Flow	If no flights match the criteria, the system displays a message and suggests alternative options.
Exceptional Flows	If the system cannot retrieve flight data due to a database error, an error message is shown.
Pre-Conditions	The customer is logged in and has entered search criteria.
Post-Conditions	The system displays available flight options.
Assumptions	Flight data is available in the system's database.
Constraints	Flights must be available for the selected criteria.
Dependencies	The system must have access to a flight database with up-to-date information.
Inputs and Outputs	Input: Search criteria (date, destination, passengers) Output: List of available flights.
Business Rules	Flight availability must be updated regularly.
Miscellaneous Information	N/A

4. Book Flight

Attribute	Description
Use Case Name	Book Flight
Use Case Description	Customer selects a flight and proceeds with booking by providing personal details and making a payment.
Primary Actors	Customer
Secondary Actors	System
Basic Flow	1. Customer selects a flight from the search results. 2. Customer provides personal and payment details. 3. The system processes the booking and reserves the flight.
Alternate Flow	If payment fails, the customer is prompted to try again with a different payment method.
Exceptional Flows	If the flight is unavailable at the time of booking, the system informs the customer and suggests alternative flights.
Pre-Conditions	Customer has selected a valid flight option.
Post-Conditions	The flight is successfully booked and the reservation is saved in the system.
Assumptions	The customer has valid payment information and sufficient funds.
Constraints	The system must support multiple payment methods (UPI, card, net banking).
Dependencies	The system must have access to a payment gateway for processing payments.
Inputs and Outputs	Input: Personal and payment details Output: Booking confirmation.
Business Rules	Flight booking must adhere to airline rules (e.g., cancellation policy).
Miscellaneous Information	N/A

5. Make Payment

Attribute	**Description**
Use Case Name	Make Payment
Use Case Description	Customer pays for the flight booking through available payment methods such as UPI, credit card, debit card, or net banking.
Primary Actors	Customer
Secondary Actors	Payment Gateway
Basic Flow	1. Customer selects the payment method. 2. The system directs the customer to the payment gateway. 3. Payment is processed and confirmed. 4. The booking is confirmed.
Alternate Flow	If the payment fails, the customer is prompted to retry or choose a different payment method.
Exceptional Flows	If the payment gateway fails or encounters an error, an error message is shown to the customer.
Pre-Conditions	The customer has selected a flight and is ready to make payment.
Post-Conditions	The payment is successfully processed, and the booking is confirmed.
Assumptions	The payment gateway is functional and available.
Constraints	The system supports various payment methods, including UPI, credit card, debit card, and net banking.
Dependencies	The system must integrate with a payment gateway for processing transactions.
Inputs and Outputs	Input: Payment details (card info, UPI, etc.) Output: Payment confirmation.
Business Rules	Payment must be processed securely and confirmed before proceeding with the booking.
Miscellaneous Information	N/A

6. Cancel/Reschedule Flight

Attribute	**Description**
Use Case Name	Cancel/Reschedule Flight
Use Case Description	Customer cancels or reschedules an existing flight booking, following the airline's cancellation or rescheduling policy.
Primary Actors	Customer
Secondary Actors	System
Basic Flow	1. Customer selects a flight from their booking history. 2. Customer chooses to either cancel or reschedule. 3. System processes the request according to the policy and confirms the action.
Alternate Flow	If rescheduling is not possible due to availability, the system prompts the customer to cancel instead.
Exceptional Flows	If cancellation or rescheduling fails (due to policy or system issues), an error message is displayed to the customer.
Pre-Conditions	The customer has an existing flight booking.
Post-Conditions	The flight is either canceled or rescheduled as per the customer's request.
Assumptions	The customer is aware of the airline's cancellation or rescheduling policies.
Constraints	The system enforces airline-specific cancellation or rescheduling policies, including fees or time restrictions.
Dependencies	The system must be integrated with the airline's booking and scheduling system to check availability and process cancellations or reschedules.
Inputs and Outputs	Input: Flight booking details, customer request to cancel or reschedule. Output: Confirmation of cancellation or rescheduling.
Business Rules	Cancellation or rescheduling must adhere to the airline's policy, including fees or penalties.
Miscellaneous Information	N/A

7. Manage Flight

Attribute	**Description**
Use Case Name	Manage Flight
Use Case Description	Admin manages flight information, including adding, editing, or deleting flight details such as timings, destinations, and available seats.
Primary Actors	Admin
Secondary Actors	System
Basic Flow	1. Admin logs in to the system. 2. Admin selects a flight to manage. 3. Admin adds, edits, or deletes flight details. 4. The system updates flight information.
Alternate Flow	If editing is not possible (e.g., flight is already booked), the system shows an error or restriction message.
Exceptional Flows	If the flight details cannot be updated due to system errors, the system displays an error message.
Pre-Conditions	The admin is logged in and has the required permissions to manage flights.
Post-Conditions	Flight details are updated in the system.
Assumptions	The admin is authorized to manage flights and has the necessary details.
Constraints	The system may have restrictions on managing certain flight details once they are booked or near departure time.
Dependencies	The system must have access to flight data and be integrated with the booking and scheduling system.
Inputs and Outputs	Input: Flight details to add, edit, or delete. Output: Confirmation of changes to flight details.
Business Rules	Only authorized admins should be allowed to manage flight details.
Miscellaneous Information	N/A

8. Track Real-Time Flight Status

Attribute	Description
Use Case Name	Track Real-Time Flight Status
Use Case Description	The system receives and displays real-time flight status updates (e.g., delayed, canceled, on-time) from a third-party vendor.
Primary Actors	Customer
Secondary Actors	Third-Party Vendor, System
Basic Flow	1. Customer enters flight details (flight number, date). 2. The system queries the third-party vendor for real-time updates. 3. The system displays flight status updates.
Alternate Flow	If the flight status changes (e.g., a delay), the system updates the status automatically.
Exceptional Flows	If the third-party vendor fails to provide updates, the system shows a "status unavailable" message.
Pre-Conditions	The customer has a valid flight number and date for tracking.
Post-Conditions	The system displays the real-time flight status.
Assumptions	The system is integrated with a third-party service providing real-time flight status.
Constraints	The availability of real-time flight data depends on the third-party vendor's service.
Dependencies	The system must have access to a third-party vendor's API for flight status.
Inputs and Outputs	Input: Flight number and date Output: Real-time flight status (on-time, delayed, canceled, etc.)
Business Rules	The flight status must be updated based on real-time information.
Miscellaneous Information	N/A

9. Generate Reports

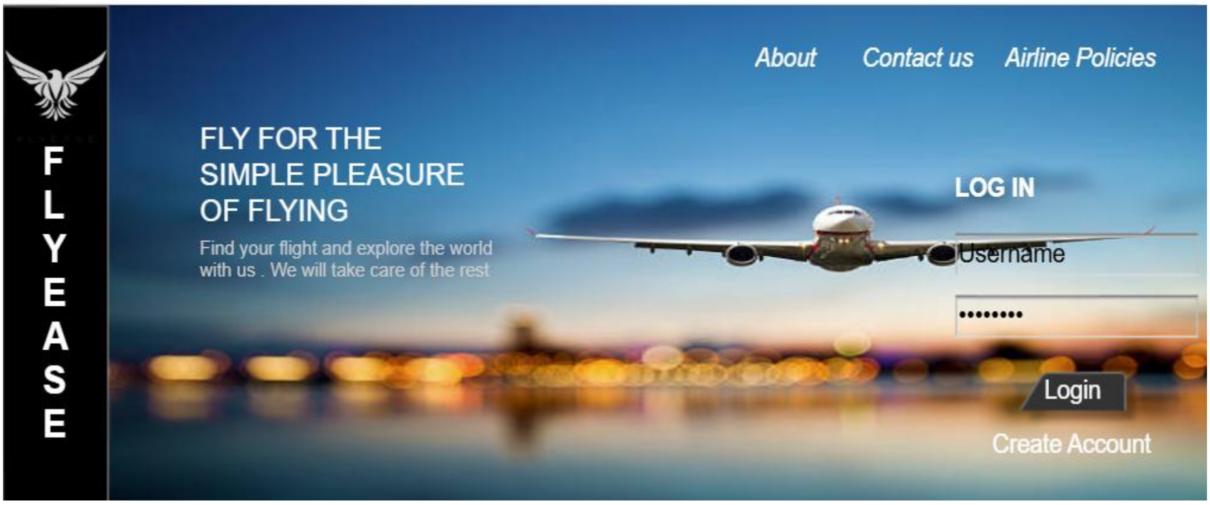
Attribute	**Description**
Use Case Name	Generate Reports
Use Case Description	Admin generates various reports, such as flight bookings, revenue, and user activity, to assist in decision-making.
Primary Actors	Admin
Secondary Actors	System
Basic Flow	1. Admin selects the type of report to generate (e.g., bookings, revenue). 2. Admin specifies report criteria (date range, filters). 3. The system generates the report and displays it.
Alternate Flow	If no data is available for the selected criteria, the system shows a "no data available" message.
Exceptional Flows	If the report generation fails due to system error, an error message is displayed.
Pre-Conditions	Admin must be logged in and have access to report-generation functionality.
Post-Conditions	A report is generated and displayed.
Assumptions	The system has up-to-date data to generate reports.
Constraints	Report generation must comply with system performance limitations (e.g., large datasets).
Dependencies	The system must have data collection and processing capabilities.
Inputs and Outputs	Input: Report criteria (date range, filters) Output: Generated report.
Business Rules	Reports should be accurate and based on available data.
Miscellaneous Information	N/A

10. View History of Bookings

Attribute	Description
Use Case Name	View History of Bookings
Use Case Description	The customer views their past flight bookings, including flight details and payment history.
Primary Actors	Customer
Secondary Actors	System
Basic Flow	1. Customer navigates to the "Booking History" section. 2. The system displays a list of past bookings. 3. Customer can view details for each booking.
Alternate Flow	If no bookings are found, the system displays a message stating "No history available."
Exceptional Flows	If there is an error retrieving booking data, the system displays an error message.
Pre-Conditions	The customer must be logged in.
Post-Conditions	The customer can view details of past bookings.
Assumptions	The customer has made previous bookings.
Constraints	Booking history is limited by the retention policy of the airline system.
Dependencies	The system must have a record of all the customer’s past bookings.
Inputs and Outputs	Input: Customer request to view booking history Output: List of past bookings with details.
Business Rules	Past bookings must be retrievable by the customer for a set period (e.g., 1 year).
Miscellaneous Information	N/A

Document 7- Screens and pages

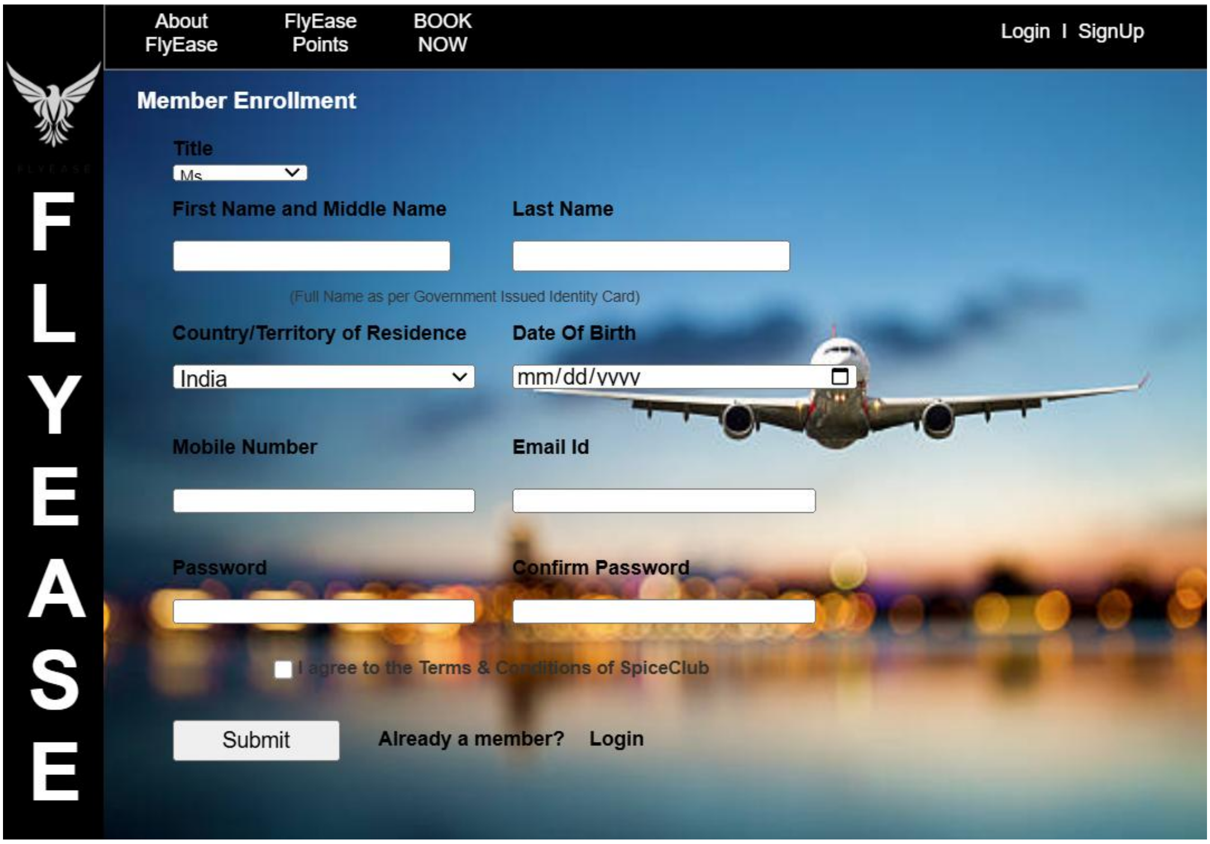
1.Home Page –



FLY FOR THE SIMPLE PLEASURE OF FLYING

Find your flight and explore the world with us . We will take care of the rest

2.Register Page



Member Enrollment

Title: Ms

First Name and Middle Name

Last Name

(Full Name as per Government Issued Identity Card)

Country/Territory of Residence: India

Date Of Birth: mm/dd/yyyy

Mobile Number

Email Id

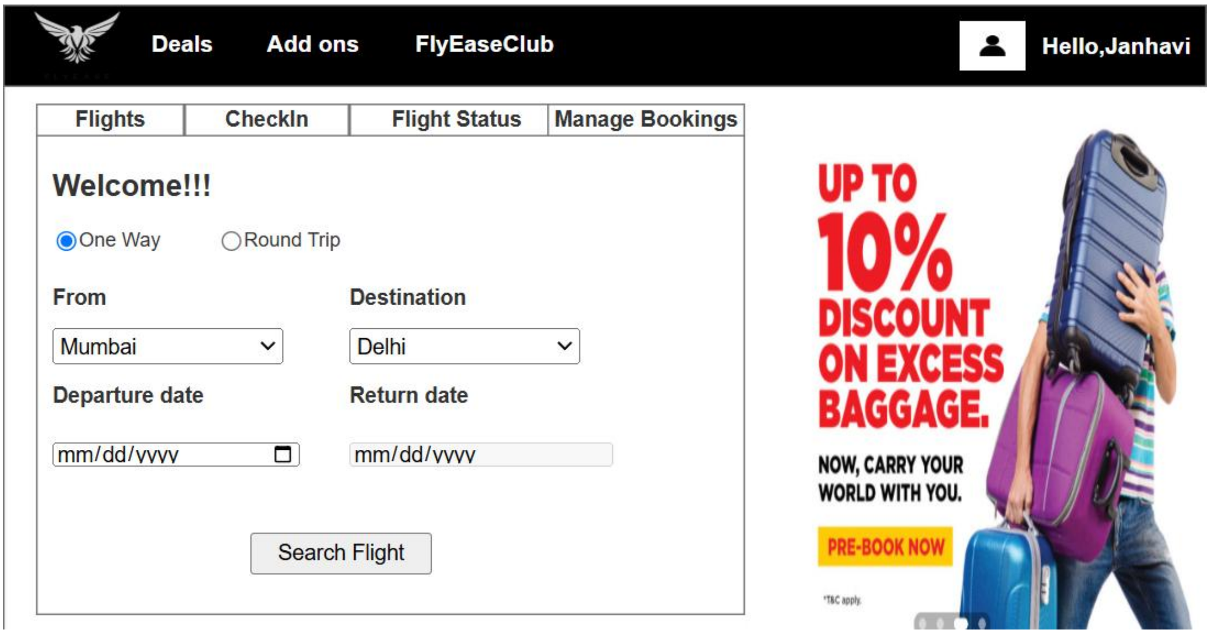
Password

Confirm Password

□ I agree to the Terms & Conditions of SpiceClub

Already a member? Login

3. Search Flights -



Welcome!!!

● One Way ○ Round Trip

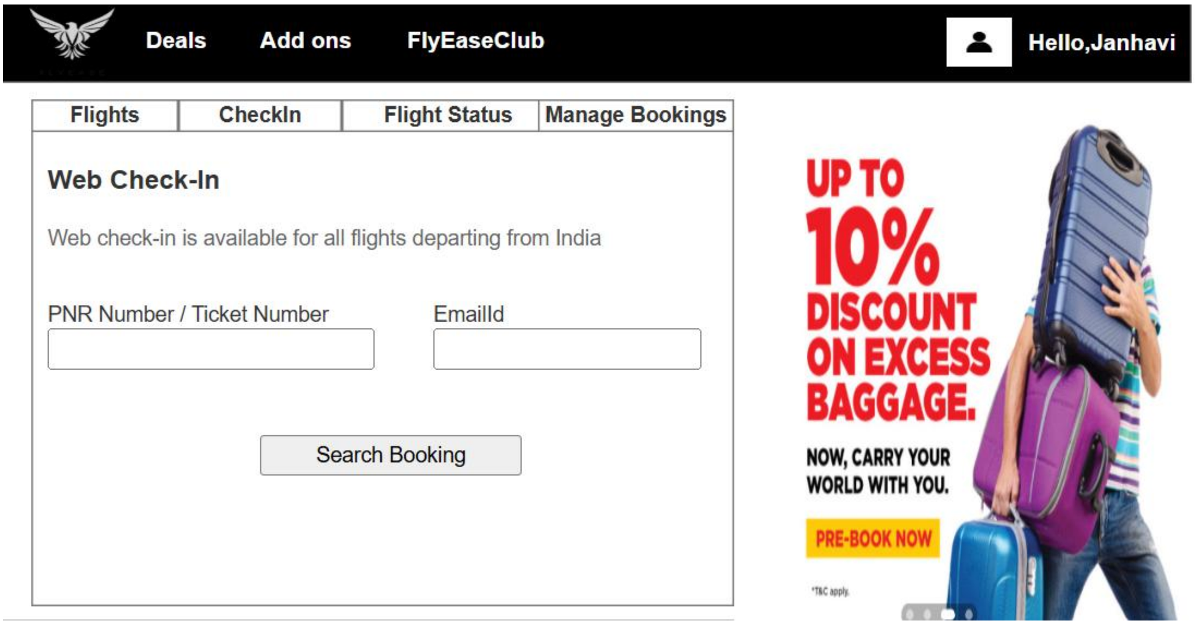
From: Mumbai

Destination: Delhi

Departure date: mm/dd/yyyy

Return date: mm/dd/yyyy

4. Check in for Flights



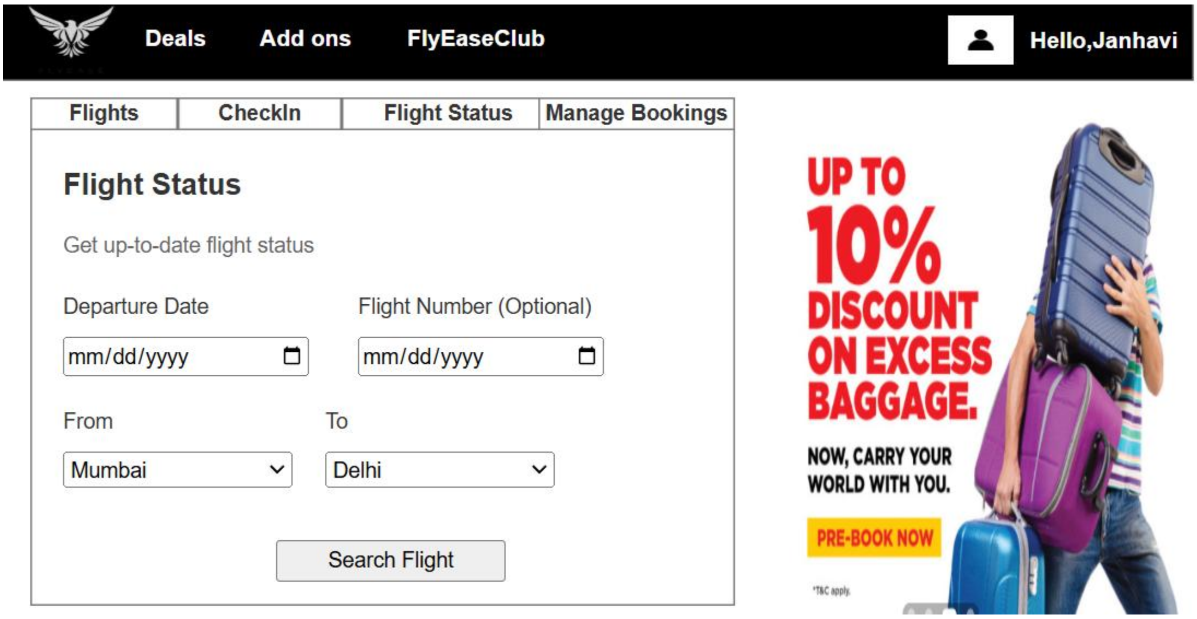
Web Check-In

Web check-in is available for all flights departing from India

PNR Number / Ticket Number

EmailId

5. Flight Status



Flight Status

Get up-to-date flight status

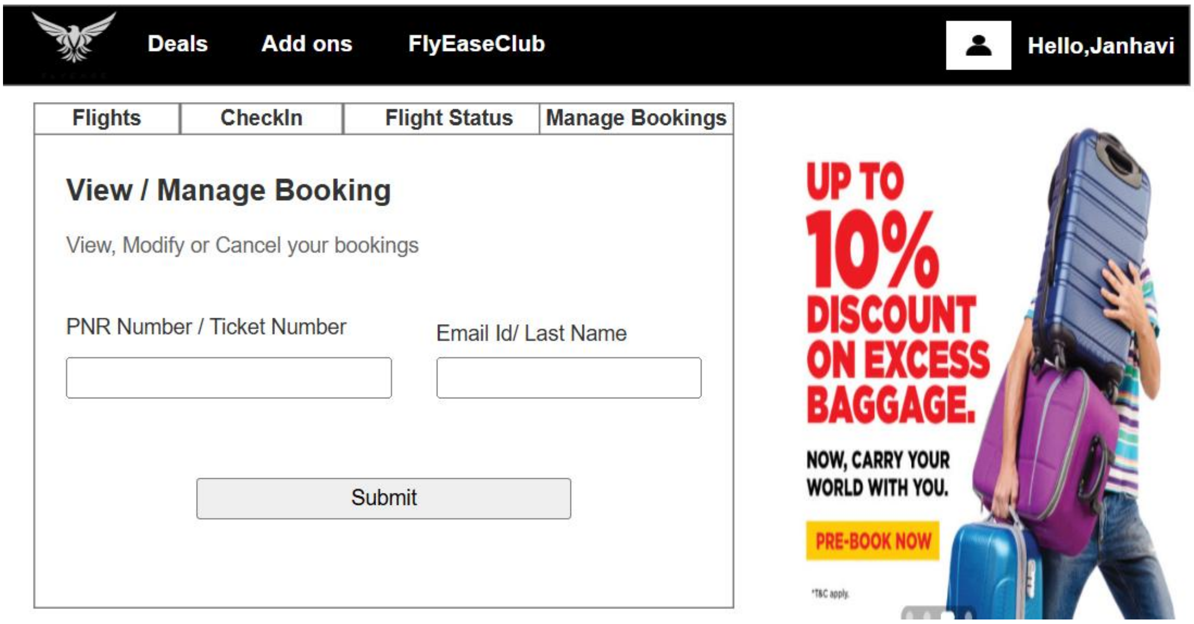
Departure Date: mm/dd/yyyy

Flight Number (Optional): mm/dd/yyyy

From: Mumbai

To: Delhi

6. Manage Booking



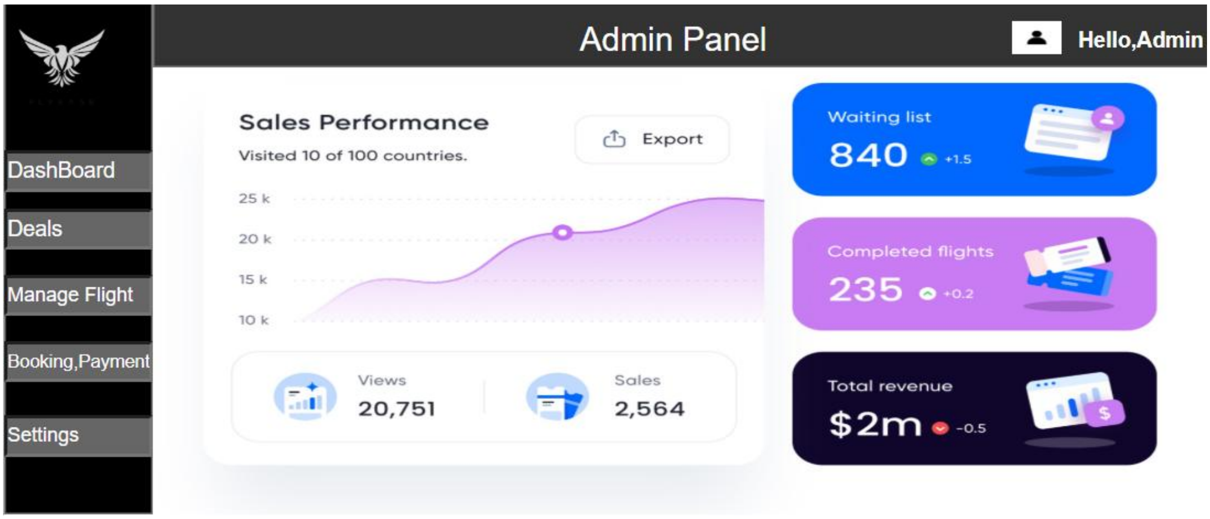
View / Manage Booking

View, Modify or Cancel your bookings

PNR Number / Ticket Number

Email Id/ Last Name

7. Admin Panel



Document 8- Tools-Visio and Axure

Throughout the project, I leveraged Visio and Axure to create visual representations and interactive prototypes of the airline system, enhancing both the design and development phases.

In **Visio**, I utilized its powerful diagramming tools to create detailed flowcharts, activity diagrams, and use case diagrams. The intuitive drag-and-drop interface allowed me to map out complex workflows such as flight bookings, cancellations, and real-time flight status tracking in a clear, structured manner. Visio's ability to connect various shapes and objects helped me maintain a consistent, organized visual structure, ensuring that stakeholders could easily understand the system's processes. Additionally, its integration with Microsoft tools made collaboration seamless, especially when incorporating feedback from team members or during meetings with the client.

On the other hand, I used **Axure** for creating interactive wireframes and high-fidelity prototypes that showcased the user interface (UI) and user experience (UX) of the airline system. Axure's robust functionality allowed me to simulate dynamic interactions, such as flight searches, bookings, and payment flows, helping stakeholders visualize the actual user experience. The ability to add conditional logic and interactions made the prototype highly interactive, providing a realistic representation of how users would navigate the system. This was especially useful during user testing sessions, where real-time feedback could be gathered to iterate on design improvements.

Both tools served distinct but complementary roles in the project. Visio helped in the planning and documentation phases, offering clarity in system workflows, while Axure was crucial for bringing the design to life, allowing stakeholders to interact with the system and providing valuable insights into user behavior and system usability.

Document 9- BA experience

My Experience as a Business Analyst in the Airline System Project:

1. Requirement Gathering:

In this phase, my primary goal was to understand and capture all the requirements for the Airline System accurately and in detail. We used the MOSCOW technique (Must Have, Should Have, Could Have, Won't Have) to prioritize the requirements based on the client's needs. Before starting with MOSCOW, I used elicitation techniques such as interviews, surveys, and workshops to gather all possible inputs from stakeholders.

Since the client was unavailable for a period during this phase, I proactively sourced alternative points of contact from the client's side, ensuring no communication gaps. I organized regular meetings with these alternate contacts, which allowed me to gather the required information and prevent delays.

Once the initial requirements were gathered, I validated them using the FURPS technique (Functionality, Usability, Reliability, Performance, and Supportability). This helped in ensuring that the requirements were clear, feasible, and realistic. There were instances where duplicate or redundant requirements were identified, so I worked with the stakeholders to remove these duplicates early on to avoid confusion and rework.

During the requirement refinement process, we used prototyping to create visual mockups and simulations, allowing the stakeholders to provide more detailed and specific feedback. The prototypes helped stakeholders better understand their expectations and facilitated a more collaborative approach to refining the requirements.

2. Requirement Analysis:

After gathering the requirements, I used UML diagrams to visually represent the system's structure and components. These diagrams helped in mapping the functional requirements to specific use cases. I also created activity diagrams to visualize the flow of actions, decision points, and interactions between users and the system.

One of the challenges during this phase was dealing with disagreements among the technical team regarding how the system should be structured. I took the initiative to facilitate open discussions and encourage constructive feedback. I also ensured that everyone understood the potential impact of any changes on the project scope, timeline, and cost.

As a result of these discussions, the team reached a consensus, and I was able to update the Business Requirements Specification (BRS) and System Requirements Specification (SRS) documents. These documents outlined the agreed-upon requirements in detail, ensuring that there was no ambiguity moving forward.

3. Design:

Once the requirements were analyzed and agreed upon, I moved to the design phase. I created use case diagrams to capture the system's functional behavior, detailing how users would interact with the system's components. These diagrams were used to derive test cases, both positive and negative, ensuring that the system was robust enough to handle real-world scenarios. I paid special attention to edge cases during the testing preparation to ensure that the system could handle unusual or unexpected inputs without failure.

While preparing the test cases, I made sure to include negative test cases, which were vital to identify potential system failures, especially in critical areas such as booking, payment, and flight status updates. I also worked on creating the necessary test data and validated that all the test cases were traceable to the specific requirements in the Requirements Traceability Matrix (RTM). This matrix ensured that all the requirements were covered in the design and testing phases.

I also collaborated closely with the technical team and client, ensuring that the design documents accurately represented the business requirements and were aligned with the project's goals. I took into account feedback from the technical team to ensure that the design was feasible and scalable. This constant communication helped avoid delays in the design phase.

4. Development:

During the development phase, I organized and led Joint Application Development (JAD) sessions with both business and technical teams to clarify queries and ensure that the project was on track. These sessions were pivotal in aligning the technical team's understanding with the business requirements. However, some team members did not initially agree with certain aspects of the design. I managed these situations by holding one-on-one meetings to address their concerns, explaining the rationale behind decisions, and ensuring that they understood the broader impact of the changes.

To maintain team collaboration and avoid disruptions, I encouraged open communication and a healthy work environment. I ensured that the technical team had the necessary documentation and resources to move forward. Regular meetings with the client were held to update them on progress, and I made sure that all team members attended these meetings. In cases where some team members couldn't attend, I recorded the sessions and shared the recordings with them for further discussions.

5. Testing:

In the testing phase, I worked closely with the QA team to prepare test cases from the use cases. The test cases included both functional and non-functional aspects, ensuring comprehensive testing. I also created test data to simulate different use scenarios and validated them against the Requirements Traceability Matrix (RTM) to confirm that all requirements had been covered.

After conducting high-level testing, I helped take client sign-off on the testing phase, ensuring that the system met the client's requirements and expectations. I also prepared the client for User Acceptance Testing (UAT) by guiding them through the testing process, explaining the necessary steps, and ensuring they were comfortable with the testing environment**.**

6. Deployment:

In the deployment phase, I took responsibility for ensuring that all the necessary documentation, including the RTM, was forwarded to the client as part of the project closure. I coordinated with the client to create end-user manuals and training materials, ensuring that users were equipped to operate the system once deployed.

I organized training sessions for end-users to ensure that they understood how to use the system effectively. I also ensured that all stakeholders attended the training sessions and that the training was comprehensive, covering all necessary features. After deployment, I continued to support the project, ensuring that any issues were promptly addressed.

Throughout the project, I acted as the bridge between the business stakeholders, technical team, and the client. I leveraged various techniques such as elicitation, prototyping, JAD sessions, and UML diagrams to ensure that all requirements were clearly defined, validated, and communicated effectively. I maintained strong communication with the client and technical team, ensuring that everyone was aligned with the project's goals.

As a Business Analyst, I played a vital role in ensuring that the project was completed on time, within scope, and met all of the client's requirements. My involvement in all phases, from requirement gathering to deployment, allowed me to contribute significantly to the success of the airline system project.