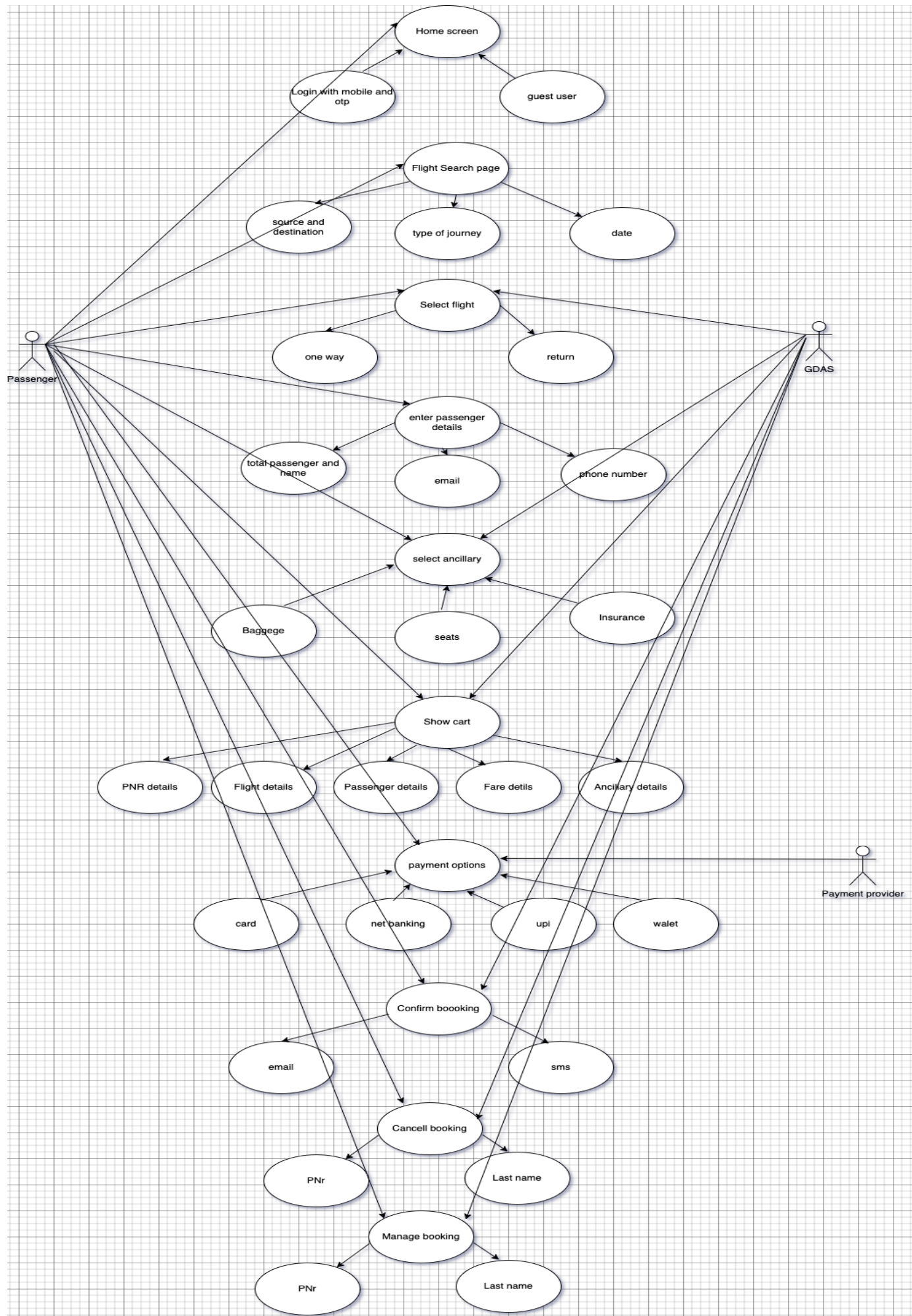
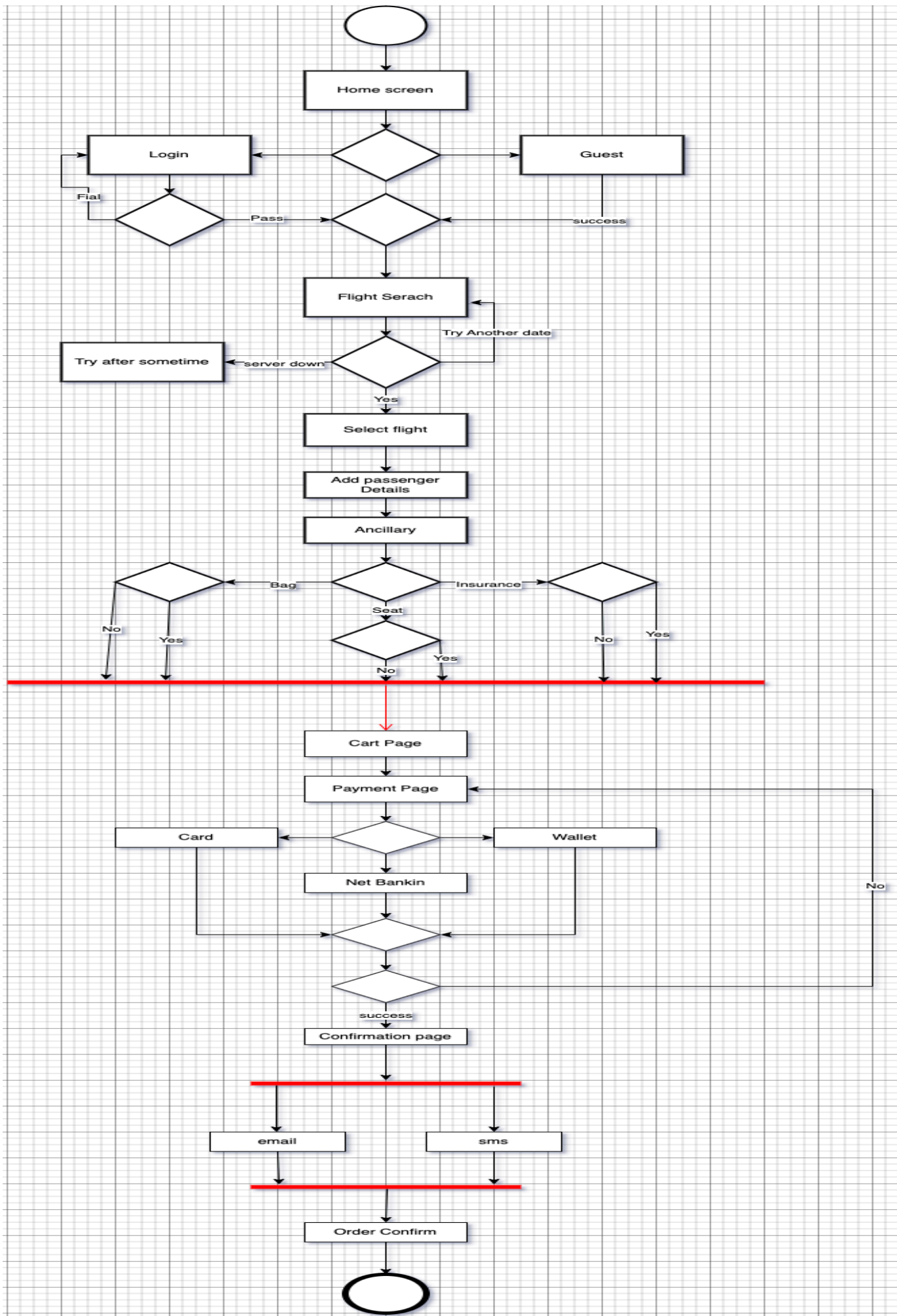


**Live Project Part - 3:**

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





use case specification document

Use cases described in a document called use case specification or document

What is the objective of the use case? what is the pre condition and what should be the post condition? do we have any assumptions on use case. what is the frequency for this use case?

FR-001: User Registration and Login

Actor

- User
- System

Pre-Conditions

- User must have a valid email, phone number, or social media account.
- System must be available and responsive.

Post-Conditions

- User is successfully registered and logged in.
- System stores user credentials securely.

Basic Flow

1. User selects "Register" or "Login" option.
2. User enters credentials (email/phone/social media login).
3. System verifies credentials.
4. If new user, system registers the user and stores credentials securely.
5. User is logged in successfully.

Exceptional Flow

- Incorrect credentials: The system displays an error message.
- Email/phone already registered: System prompts for password reset.
- Social media login fails: System suggests alternative login methods.

Assumptions

- User has internet access.
- System has access to authentication services.

Frequency of Use

- High (frequently used feature).

---

## FR-002: Flight Search

### Actor

- User
- System

### Pre-Conditions

- User has access to the system.
- System has updated flight data.

### Post-Conditions

- User is presented with a list of available flights based on search criteria.

### Basic Flow

1. User enters origin, destination, and travel dates.
2. System fetches available flights.
3. System displays matching flights.

### Exceptional Flow

- No flights available: System displays an appropriate message.
- Invalid date format: System prompts for correction.

### Assumptions

- System integrates with airline databases.

### Frequency of Use

- High
- 

## FR-003: Flight Booking

### Actor

- User
- System

### Pre-Conditions

- User has selected a flight.
- User is logged in.

#### Post-Conditions

- Flight booking is confirmed.
- User receives a confirmation message.

#### Basic Flow

1. User selects a flight.
2. User enters passenger details.
3. User selects payment method.
4. System processes the payment.
5. Booking confirmation is displayed and sent via email/SMS.

#### Exceptional Flow

- Payment failure: System prompts for retry or alternate payment.
- Invalid passenger details: System requests correction.

#### Assumptions

- Payment gateway is operational.

#### Frequency of Use

- Moderate
- 

### FR-004: Fare Comparison

#### Actor

- User
- System

#### Pre-Conditions

- User initiates a flight search.
- System has access to fare data.

#### Post-Conditions

- User is presented with fare trends.

### Basic Flow

1. User searches for a flight.
2. System retrieves fare trends for adjacent dates.
3. System displays fare comparison.

### Exceptional Flow

- No fare data available: System displays an error message.

### Assumptions

- System regularly updates fare data.

### Frequency of Use

- Moderate
- 

## FR-005: Real-Time Flight Updates

### Actor

- User
- System

### Pre-Conditions

- User has a booked flight.
- System has access to real-time flight data.

### Post-Conditions

- User receives flight status updates.

### Basic Flow

1. System retrieves flight status.
2. System notifies user of any changes.

### Exceptional Flow

- Data fetch failure: System retries and alerts user.

### Assumptions

- Airline APIs provide real-time updates.

#### Frequency of Use

- Moderate to High
- 

#### FR-006: Notifications

##### Actor

- User
- System

##### Pre-Conditions

- User has an active booking.

##### Post-Conditions

- User receives timely notifications.

##### Basic Flow

1. System monitors booking and flight updates.
2. System sends relevant notifications.

##### Exceptional Flow

- Delivery failure: System retries sending notification.

##### Assumptions

- Notification service is functional.

#### Frequency of Use

- High

#### FR-007: Multilingual Support

##### Actor

- User
- System



### Pre-Conditions

- User accesses the system.

### Post-Conditions

- System displays content in the user's preferred language.

### Basic Flow

1. User selects preferred language.
2. System updates the UI and content accordingly.

### Exceptional Flow

- Language data not available: System defaults to English.

### Assumptions

- System has language packs installed.

### Frequency of Use

- Moderate
- 

## FR-008: Payment Integration

### Actor

- User
- System
- Payment Gateway

### Pre-Conditions

- User has selected a flight and entered payment details.

### Post-Conditions

- Payment is processed successfully.

### Basic Flow

1. User selects a payment method.
2. System redirects to the payment gateway.

3. Payment is processed.
4. System confirms the payment.

#### Exceptional Flow

- Payment failure: System prompts user to retry.

#### Assumptions

- Payment gateway is functional.

#### Frequency of Use

- High

#### R-009 - Rescheduling Flights

1. Use Case Name: Rescheduling Flights
2. Actor(s): User, Airline System, Payment Gateway
3. Pre-Conditions:

- User must be logged into the application.
- User must have an existing confirmed flight booking.
- Rescheduling must be allowed as per airline policies.
- Sufficient funds must be available if additional payment is required.

#### Post-Conditions:

- The flight is successfully rescheduled.
- The user receives confirmation of the updated booking details.
- Any applicable charges are processed successfully.

#### Basic Flow:

1. User logs into the application.
2. User navigates to "My Bookings."
3. User selects the flight they wish to reschedule.
4. System checks if the flight is eligible for rescheduling.
5. User selects a new date and time for the flight.
6. System displays available flight options and any applicable fare differences.
7. User confirms the new selection and agrees to any additional charges.
8. If payment is required, the user completes the transaction via the payment gateway.
9. System updates the booking details.
10. System sends a confirmation notification to the user via email/SMS/in-app.

#### Alternate Flow:

- If the selected new flight is unavailable, the system notifies the user and allows them to choose another option.
- If there are multiple passengers in the booking, the system allows the user to reschedule individually or for all.

#### Exceptional Flow:

- If rescheduling is not allowed for the flight, the system displays an error message.
- If payment fails, the system prompts the user to retry or choose another payment method.
- If system connectivity issues occur, the user is notified to try again later.

#### Assumptions:

- The user has access to the internet and a valid payment method if required.
- The airline allows online rescheduling for the selected ticket type.
- User has provided valid contact details to receive the confirmation.

#### Frequency of Use:

High

## **Document 7- Screens and pages**

### Screens and Pages for the Voyage Airline Booking System

#### 1. Home Page

- Search bar for flights (departure, destination, date, passengers, class)
- Featured deals and promotions
- Login/Register option
- Quick links to important features (Manage Booking, Check-in, Support)

#### 2. User Authentication Pages

- Login Page: Email/phone number login with password
- Registration Page: User details, password setup, and verification
- Forgot Password Page: Password reset via email or SMS

#### 3. Flight Search Results Page

- Filter options (price, airline, layovers, duration, time)
- Sorting (cheapest, fastest, best match)
- Detailed flight information (timings, layovers, airline, baggage allowance)
- Compare flights feature

#### 4. Flight Booking & Payment Page

- Passenger details (name, contact, passport info for international flights)
- Seat selection
- Baggage options and add-ons (extra luggage, meals, travel insurance)
- Payment gateway (credit/debit card, net banking, wallets, UPI, PayPal)
- Booking summary with fare breakdown
- Confirmation with PNR generation

#### 5. Booking Management Page

- View upcoming and past bookings
- Modify/cancel bookings
- Reschedule flights
- Download e-tickets

#### 6. Check-in Page

- Online check-in options
- Seat selection
- Boarding pass generation (download/print option)

## 7. Flight Status & Notifications Page

- Real-time flight tracking
- Status updates (on-time, delayed, canceled)
- Notification settings for updates

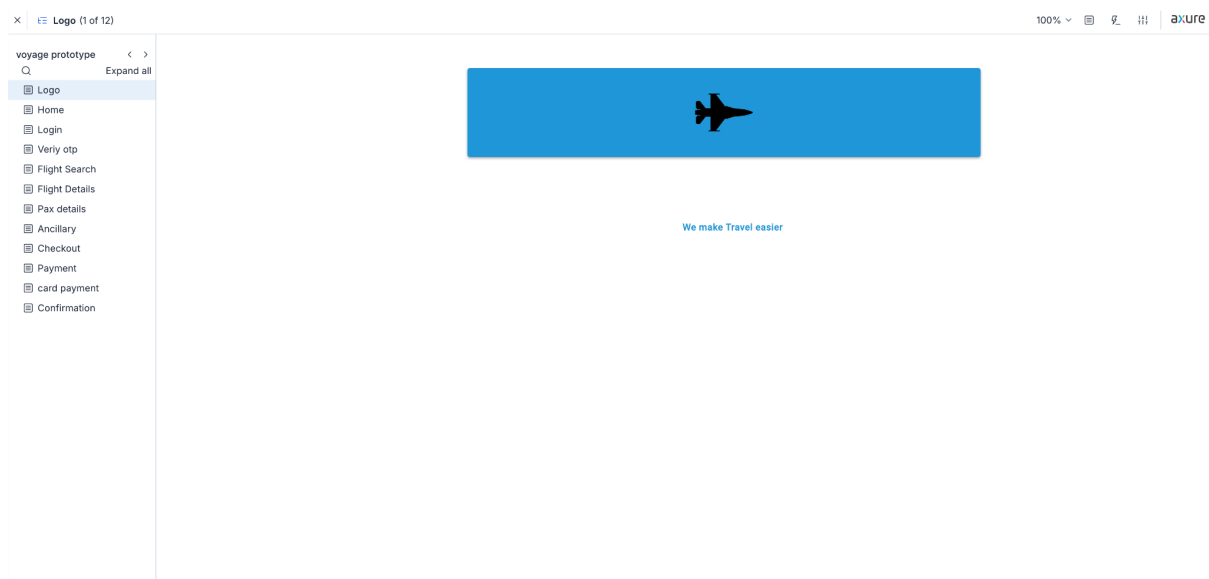
## 8. User Profile & Settings Page

- Personal information
- Saved payment methods
- Frequent flyer details
- Preferences (notifications, language, currency)

## 9. Support & Help Center Page

- FAQs and self-help guides
- Live chat and support tickets
- Contact customer care

### Azure:



voyage prototype < >  
Q Expand all

- Home
- Login
- Verify otp
- Flight Search
- Flight Details
- Pax details
- Ancillary
- Checkout
- Payment
- card payment
- Confirmation



We make Travel easier

Login

Guest

- voyage prototype < > Expand all
- Q

Logo

Home

Login

Verify otp

Flight Search

Flight Details

Pax details

Ancillary

Checkout

Payment

card payment

Confirmation



We make Travel easier

Mobile

Enter mobile with country code

OR



E-mail

Enter a valid email

Sent OTP

- voyage prototype < > Expand all
- Q

Logo

Home

Login

Verify otp

Flight Search

Flight Details

Pax details

Ancillary

Checkout

Payment

card payment

Confirmation



We make Travel easier

OTP

Enter Otp received on Mobile or email

Verify

- voyage prototype < > Expand all
- Q

Logo

Home

Login

Verify otp

Flight Search

Flight Details

Pax details

Ancillary

Checkout

Payment

card payment

Confirmation




We make Travel easier

OTP

Enter Otp received on Mobile or email

Verify

- voyage prototype
- Q Expand all
- Logo
  - Home
  - Login
  - Verify otp
  - Flight Search
  - Flight Details
  - Pax details
  - Ancillary
  - Checkout
  - Payment
  - card payment
  - Confirmation



1 Flight Search Journey

2 Passenger Add Passenger

2 Ancillary Add Baggage, Seat, etc.

3 Payment Perform Payment

Source

Select Source

Destination

Select Destination

Journey Type


Select Journey

Date

Pick a date

Submit

- voyage prototype
- Q Expand all
- Logo
  - Home
  - Login
  - Verify otp
  - Flight Search
  - Flight Details
  - Pax details
  - Ancillary
  - Checkout
  - Payment
  - card payment
  - Confirmation



1 Flight Search Journey

2 Passenger Add Passenger

2 Ancillary Add Baggage, Seat, etc.

3 Payment Perform Payment

BOM - NYC

June 10 \$1000

June 11 \$1000

June 12 \$1000

June 13 No Flight

June 14 \$1000

June 15 \$1000

June 16 \$1000

June 17 No Flight

June 18 \$1000

BOM 00:30 13-Jun

Direct 10 hr

NYC 6:00 14-Jun

BOM 4:00 13-Jun

One Stop 20hr

NYC 13:00 14-Jun

BOM 00:30 13-Jun

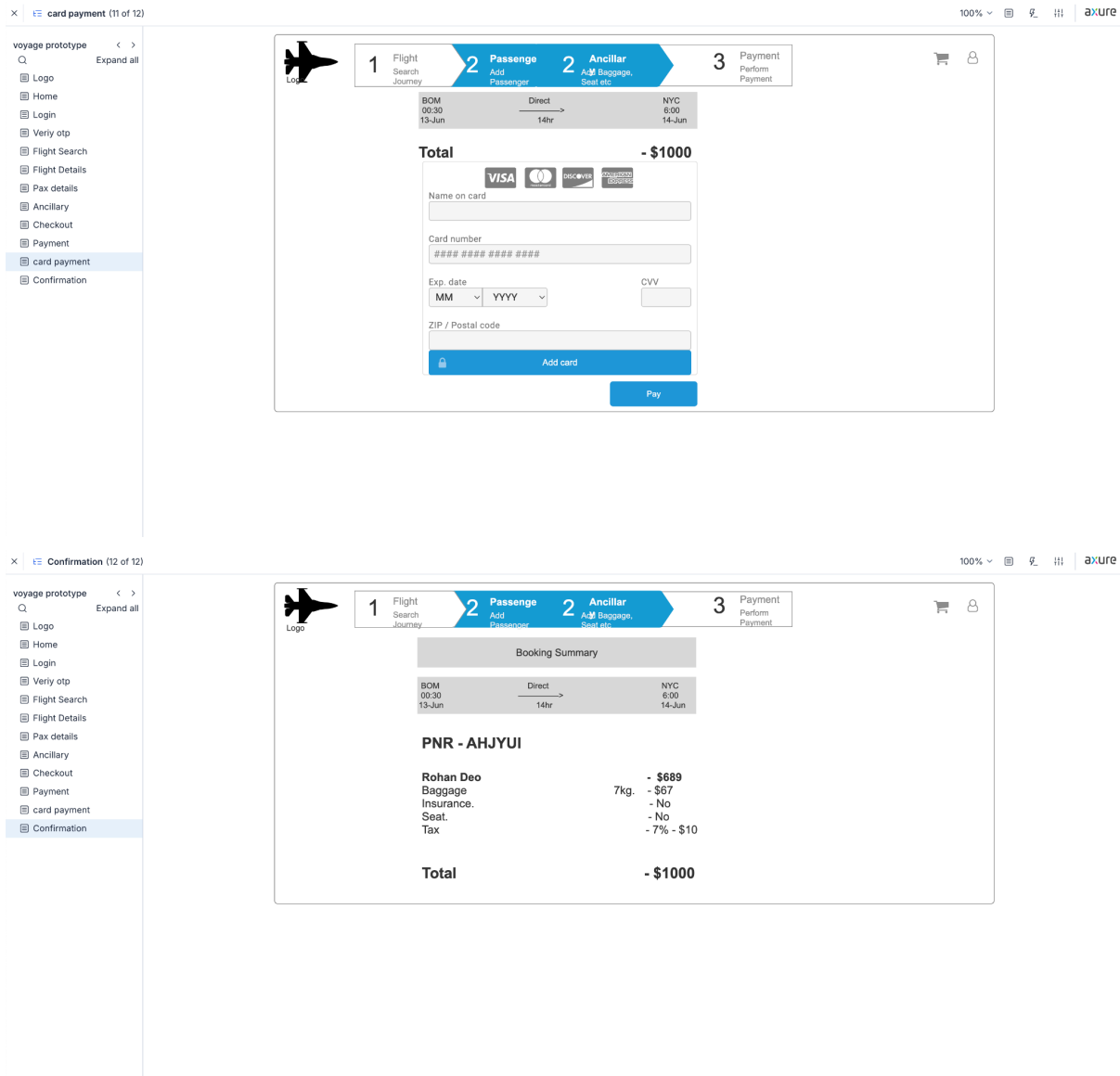
Direct 14hr

NYC 6:00 14-Jun









## Document 8- Tools-Visio and Axure Write a paragraph on your experience using Visio and Axure for the project.

I used Microsoft Visio and Axure RP to design workflows, wireframes, and prototypes for this project.

Visio helped me create clear flowcharts and diagrams to map out processes like Login, search, pax detailing, and Payment. It made it easy to organize and visualize complex workflows.

Axure RP was useful for building interactive wireframes and prototypes, allowing stakeholders to see and test the design before development. Its features, like interactive panels and logic-based actions, helped us create a more user-friendly experience. Both tools made planning, designing, and improving the interaction look and feel better.

Azure provides all the prototypes, and sample templates, provides ideas to color the text and boxes, and links the interaction as per design.

## Document 9- BA experience

### My Experience as a Business Analyst Across Project Phases

#### 1. Requirement Gathering

##### Challenges & Workarounds:

- Used the **MOSCOW technique** to prioritize requirements effectively
- identified and removed duplicate or repetitive requirements to maintain clarity
- validated the requirement using **FURPS technique**
- When key stakeholders were unavailable, I reached out to secondary points of contact and documented assumptions for later validation.
- Used **prototyping and real-world examples** to help clients better articulate their needs.
- Conducted **requirement reconciliation meetings** to resolve differences and finalize requirements.

#### 2. Requirement Analysis

##### Challenges & Workarounds:

- I created **UML diagrams** and **activity diagrams** to illustrate process flows clearly
- Preparing the **Business Requirement Specification (BRS)** and **Software Requirement Specification (SRS)** documents.
- **Team Disagreements on Diagrams:** Addressed concerns by organizing **collaborative review sessions** and incorporating valid feedback.
- **Changing Requirements Midway:** Maintained a **Change Log** and updated impacted artifacts accordingly.
- **Lack of Documentation Standards:** Followed best practices and templates to ensure clarity and consistency.

#### 3. Design

##### Challenges & Workarounds:

- 
- I prepared **test data**
- I included both **positive and negative scenarios**, ensuring no critical test case was missed to prevent future development risks
- **Overlooked Scenarios in Test Cases:** Conducted **peer reviews** and **walkthroughs** to ensure all possible scenarios were covered.
- **Unclear Business Rules for Testing:** Held **clarification sessions** with SMEs and developers to refine business logic.
- **Scope Creep:** Used **RTM** to track requirements and prevent undocumented changes.

#### 4. Development

##### Challenges & Workarounds:

- I facilitated **Joint Application Development (JAD) sessions** to align stakeholders.
- **Developer Resistance to Changes:** Provided **detailed impact analysis** to justify necessary modifications.
- **Delayed Feedback from Clients:** Created **structured feedback cycles** with deadlines to speed up approvals.
- **Misalignment Between Business & Tech Teams:** Organized **bridge meetings** to ensure mutual understanding.

## 5. Testing

### Challenges & Workarounds:

- I prepared **test cases** based on **use cases** and conducted **high-level testing** to verify system functionality.
- **Insufficient Test Data:** Requested sample data early and collaborated with the client to generate realistic test scenarios.
- **Test Environment Issues:** Worked closely with DevOps and IT teams to resolve setup problems.
- Make sure all test cases get executed by team and provided the support if required.
- Make sure team save all the test results to avoid any confusion in future
- Make sure to have proper sign off from team.
- Conducted regular triage call to avoid any delay in bug fixing.
- Make sure test cases were ready and reviewed before development get completed.
- **Delays in UAT Signoff:** Ensured **early engagement** with business users and provided **step-by-step guidance** during testing.

## 6. Deployment

### Challenges & Workarounds:

- **End-User Resistance to Change:** Conducted **user training workshops** and provided FAQs to ease adoption.
- **Missing Critical Signoffs:** Set **reminder emails and follow-ups** to ensure all approvals were secured on time.
- **Unavailability of Key Stakeholders:** Recorded training sessions and created **self-help guides** for future reference.
- Make sure all the necessary teams and support is available during the deployment.