

ISD TERM PROJECT

<Requirements Specification>

Flight Booking System



Project Advisor

ANEES UR RAHMAN

Presented by:

Group ID: 1

L1F20BSCS0029 MOHAMMAD ALIYAN

L1F20BSCS0025 HAMMAD QAYYOOM

Faculty of Information Technology

University of Central Punjab

Software Requirements Specification

Version 0

Flight Booking System

Advisor: ANEES UR RAHMAN

Group 1

Member Name	Primary Responsibility	Contribution %	Roll No	Serial #	Sign
Mohammad Aliyan		50%	L1F20BSCS002 9		
Hammad Qayyoom		50%	L1F20BSCS002 5		

Table of Contents

Table of Contents.....	iii
1. Introduction and Background.....	1
1.1 Product (Problem Statement).....	1
1.2 Background.....	1
1.3 Scope.....	1
1.4 Objective(s)/Aim(s)/Target(s).....	1
1.5 Challenges.....	1
1.6 Learning Outcomes.....	1
1.7 Nature of End Product.....	1
1.8 Completeness Criteria.....	1
1.9 Business Goals.....	1
1.10 Related Work/ Literature Survey/ Literature Review.....	1
1.11 Document Conventions.....	1
2. Overall Description.....	2
2.1 Product Features.....	2
2.2 User Classes and Characteristics.....	2
2.3 Operating Environment.....	2
2.4 Design and Implementation Constraints.....	2
2.5 Assumptions and Dependencies.....	2
3. Project Management.....	3
3.1 Work Breakdown Structure (WBS).....	3
3.2 Develop an Activity Graph.....	3
3.3 Find the Critical Path.....	3
3.4 Create a Gantt chart.....	3
~~~~~ <b>Phase II</b> ~~~~~.....	<b>3</b>
<b>4. Functional Requirements.....</b>	<b>3</b>
4.1 Name of Use-Case 1.....	3
4.2 Name of Use-Case 2 (and so on).....	4
Requirements Analysis and Modeling.....	4
<b>5. Nonfunctional Requirements*.....</b>	<b>4</b>
5.1 Performance Requirements.....	4
5.2 Safety Requirements.....	4
5.3 Security Requirements.....	4
5.4 Additional Software Quality Attributes.....	5
<b>6. Other Requirements.....</b>	<b>5</b>
<b>7. Designing.....</b>	<b>5</b>
7.1 Complete class diagram.....	5
7.2 Complete Data Flow Diagram (DFD).....	5
7.3 Complete ER Diagram.....	5
7.4 Physical design of your database.....	5
7.5 Information on use of design patterns while designing the modules.....	5
7.6 Make a Sequence and a Collaboration diagram of following.....	5
7.6.1 Scenario 1.....	5
7.6.2 Scenario 2.....	5
<b>8. Estimation.....</b>	<b>6</b>
8.1 Cost Benefit Estimation.....	6
8.1.1 Return of Investment (ROI) {1 st year}.....	6
8.1.2 %gain on ROI {1 st year}.....	6
8.1.3 Payback Period in years.....	6
8.2 FP based Estimation.....	6
8.3 COCOMO Estimation.....	6
<b>9. References.....</b>	<b>6</b>

Appendix A: Glossary.....	7
---------------------------	---

# 1. Introduction and Background

## 1.1 Product (Problem Statement)

We propose to develop a Flight Booking System. The old system of airline reservation was manual and not fully online. This system is expected to help people with all kinds of online processes of Reservation and Flight Bookings. It will allow people to look up flight times, seats, and their availability as well as track their plane locations.

## 1.2 Background

The objective of our project is to design and develop a software that will facilitate major airline operations such as online air ticket booking or any other operation through an efficient yet with simple GUI. Ordinary passenger who intends to travel by airways.

## 1.3 Scope

The Flight Booking System provides many facilities for customers who have had many problems with old seat booking systems. This type of system or software allows passengers to see their flights, seats and allotted times. It also offers the possibility to return the ticket or change that particular booking. It will also allow passengers to check their plane location so that they could be updated about their flight delays as soon as possible.

## 1.4 Objective(s)/Aim(s)/Target(s)

1. User can make flight bookings at any time that suits them 24/7.
2. User can compare prices and timing of different flights so they can make choice wisely.
3. System should be user friendly as possible so anyone can use.
4. User personal information should be secured.
5. The information should be Up-to-date about Flight Timings and Delays.
6. User can refund their ticket in the giving amount of period.

## 1.5 Challenges

There are many challenges which we faced in this project

- Security Issues
- Data storage problem
- Data management issues
- Flight tracking problem
- Handle Customer Complaints

## **1.6 Learning Outcomes**

Airline online booking system allows a customer to do all the work for you they are served up a selection of free spaces, they choose a date and time which are best for them. The main objective of this project is to test Airline online booking system various test case scenarios and make a detailed report. Easy to book tickets, Saves time and money. Provides every information about flight. 24/7 customer support through chat and calls.

## **1.7 Nature of End Product**

In this apps or site, you can reserve seats at any convenient time. you can cancel purchases and choose other flights suitable for you, and also be updated about delays.

## **1.8 Completeness Criteria**

The completeness criteria of this flight booking system have been set in such a way that it overcomes all the fundamental problems associated with flight booking. Users can easily view and compare the times and prices of different airlines and can book there chosen flight to reach their destination.

## **1.9 Business Goals**

The target audience for this flight booking system is all kinds of people who don't want to visit the travel agent's office to book a flight they just want to book a flight from the comfort of their homes. Users can easily download this app from the play store/app store they can also visit the website for booking.

## **1.10 Related Work/ Literature Survey/ Literature Review**

We have spoken to many customers about the issue they are facing and their only complaint is that refunds take a long time if they cancel a flight.

## **1.11 Document Conventions**

- Main heading Size 18 Bold (Calibri)
- Subheadings Size 14 Bold (Calibri)
- Paragraph Size 12 (Calibri)

## **2. Overall Description**

### **2.1 Product Features**

- Flight inquiry module.
- Flight Booking.
- Flight cancellation
- Allow passengers to choose their seats.
- Display flight Details.
- Flight search option.

### **2.2 User Classes and Characteristics**

These are the kinds of users for this Airline Booking System.

- One of them customer and other is administrator.
- The Customer does not need any training for the application.
- However, the customer can book the flight ticket, check the flight details.
- The administrator need training for the application.
- Administrator can update/add/delete any information in online reservation database.

### **2.3 Operating Environment**

This application can run on smartphones like Android and iOS. Version 7 or above is required for android mobiles and version 10 or above for IOS. And the storage required to install this application is 100mb.

### **2.4 Design and Implementation Constraints**

Describe any items or issues that will limit the options available to the developers. These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer's organization will be responsible for maintaining the delivered software).

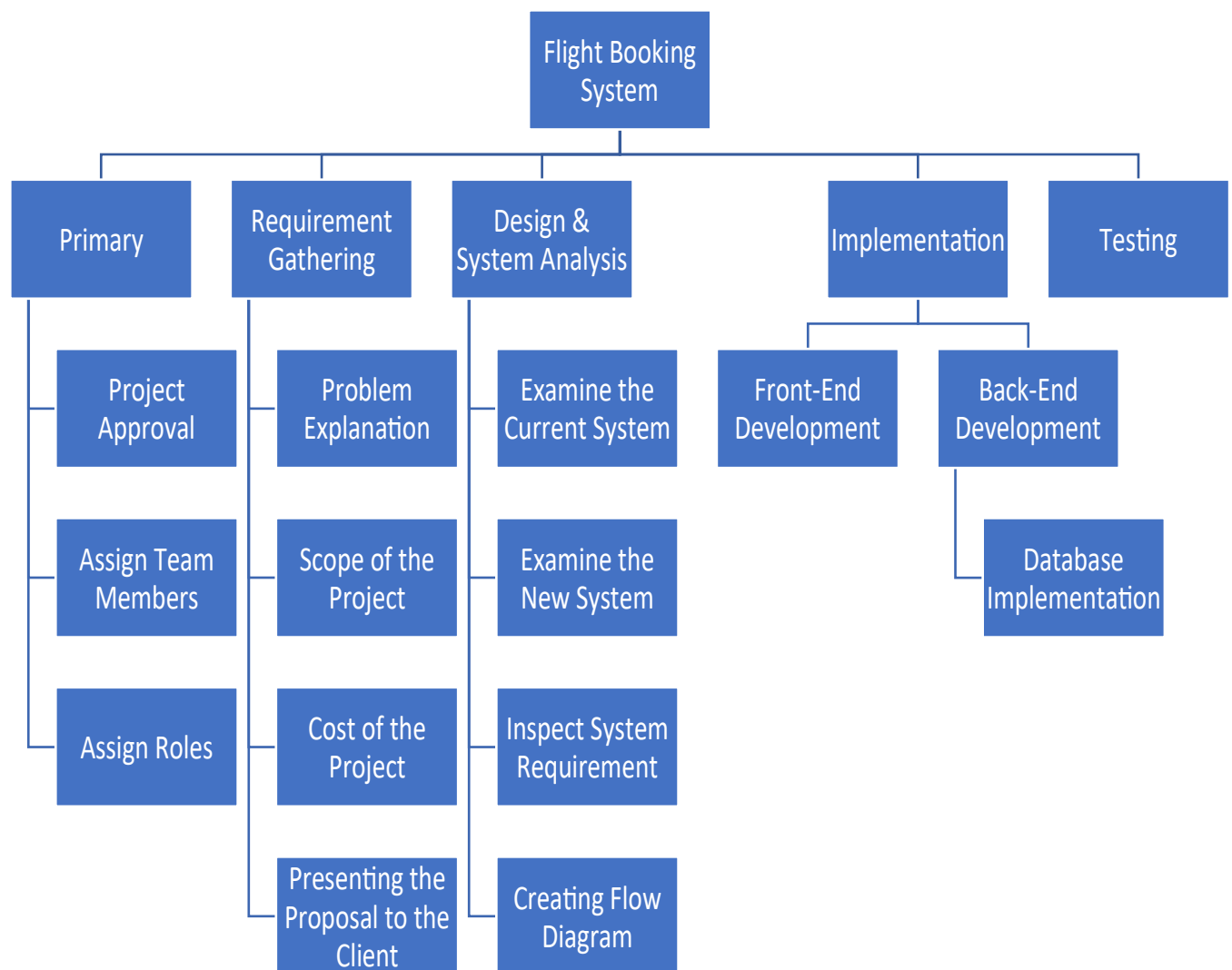
### **2.5 Assumptions and Dependencies**

It is believed that users of the airline reservation system have access to the internet on a mobile. Credit and debit card authority services are assumed to be available. Every airline's database is integrated with the airline reservation system, so it needs to be able to interface with these systems. It must also follow the business regulations of the airline.

### **3. Project Management.**

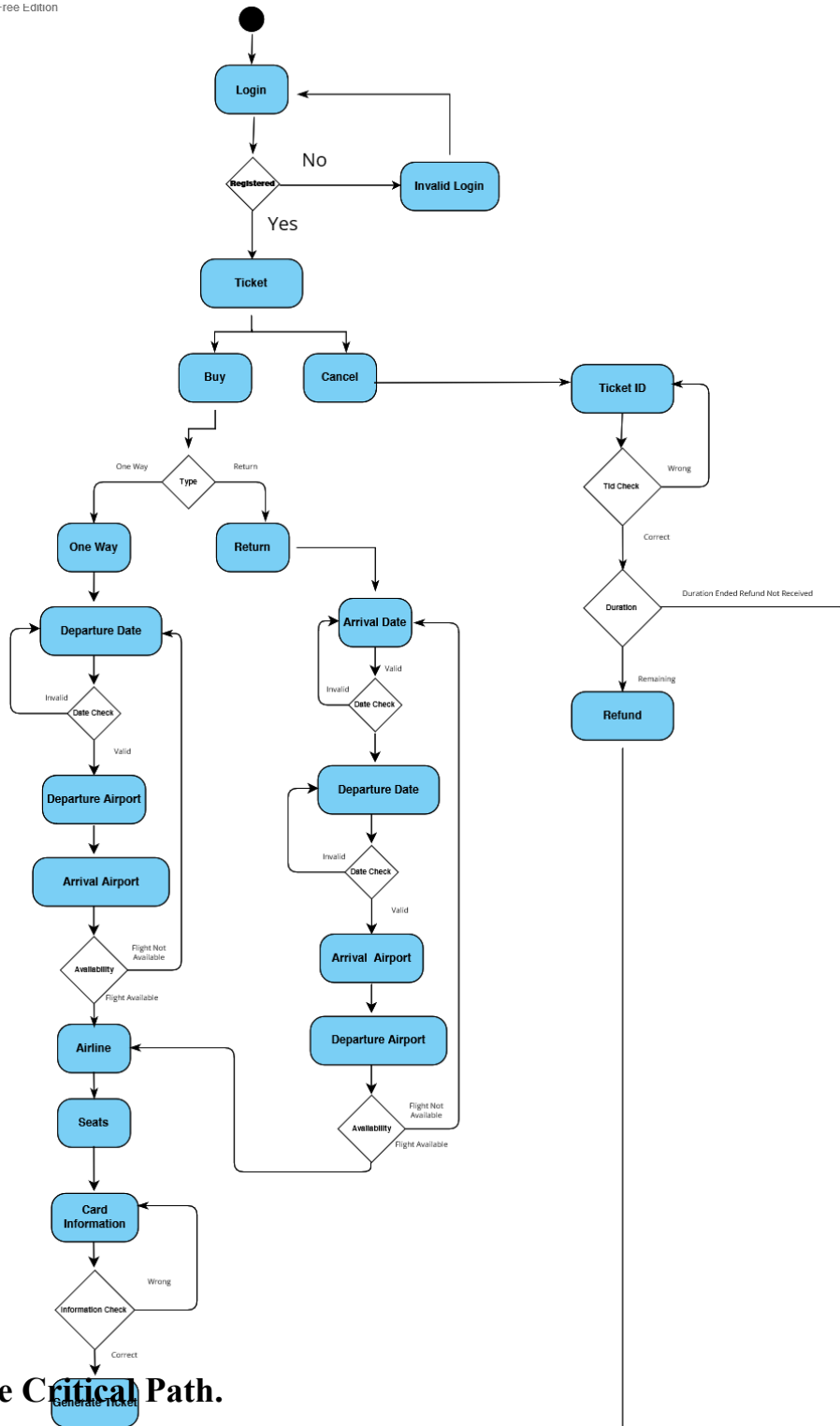


### 3.1 Work Breakdown Structure (WBS).



### 3.2 Develop an Activity Graph.

Visual Paradigm Online Free Edition

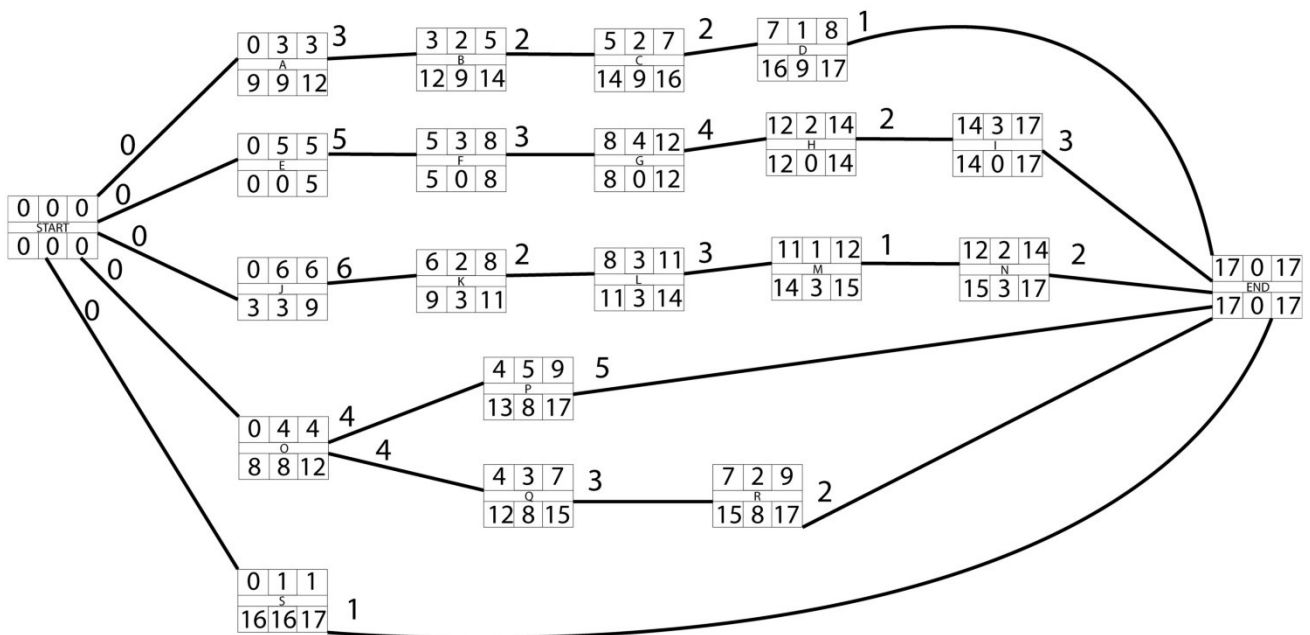


### 3.3 Find the Critical Path.

ID	ACTIVITY	DURATION (Days)	PREDECESSOR
A	Primary	3	-

**<Flight Booking System>**

B	Project Approval	2	A
C	Assign Team Members	2	B
D	Assign Roles	1	C
E	Requirement Gathering	5	-
F	Problem Explanation	3	E
G	Scope of the Project	4	F
H	Cost of the Project	2	G
I	Presenting the Proposal to the Client	3	H
J	Design & System Analysis	6	-
K	Examine the Current System	2	J
L	Examine the New System	3	K
M	Inspect System Requirement	1	L
N	Creating Flow Diagram	2	M
O	Implementation	4	-
P	Front-End Development	5	O
Q	Back-End Development	3	O
R	Database Implementation	2	Q
S	Testing	1	-

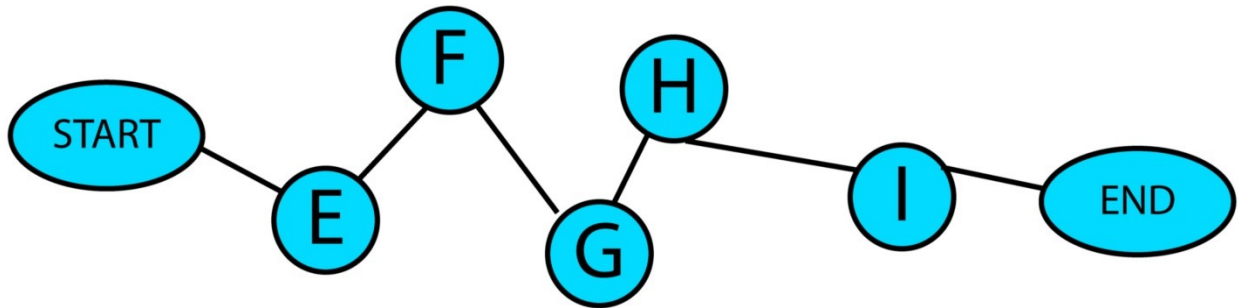


## <Flight Booking System>

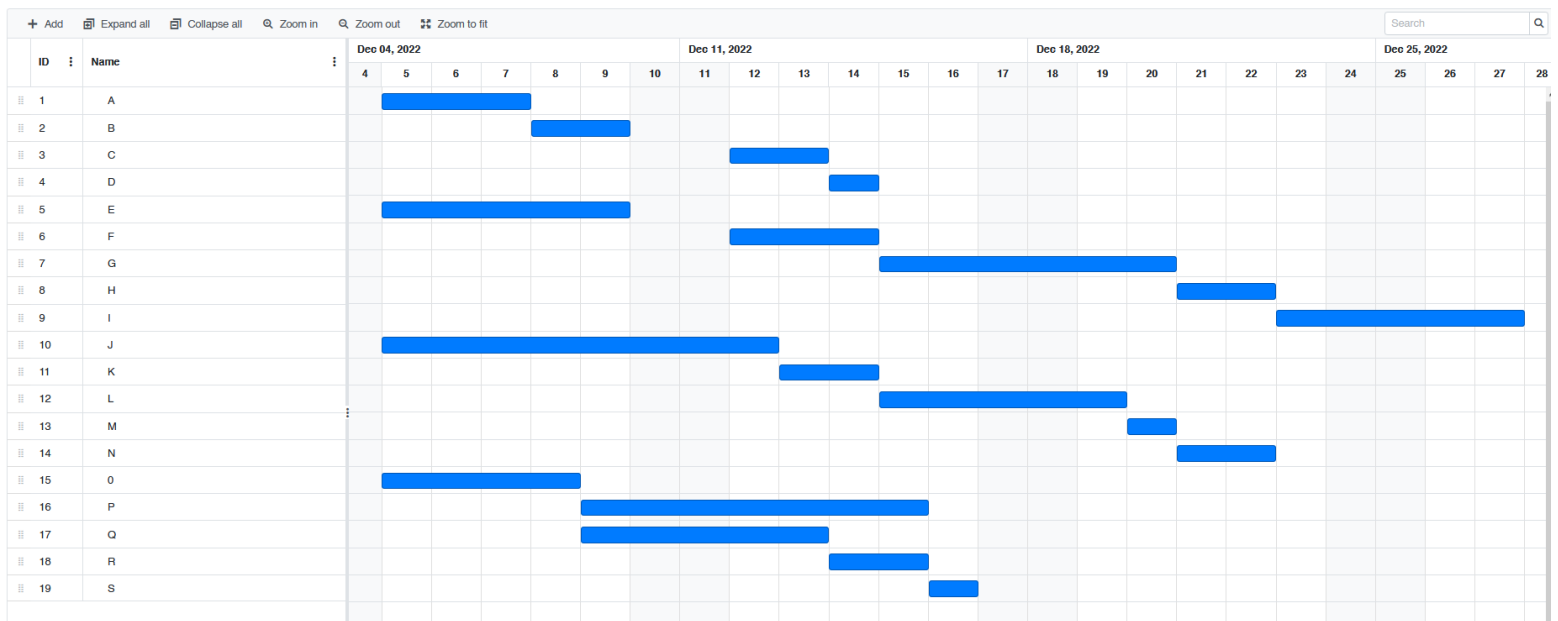
Paths: -

- 1) Start – A – B – C – D – End
- 2) Start – E – F – G – H – I – End
- 3) Start – J – K – L – M – N – End
- 4) Start – O – P – End
- 5) Start – O – Q – R – End
- 6) Start – S – End

Critical Path: -



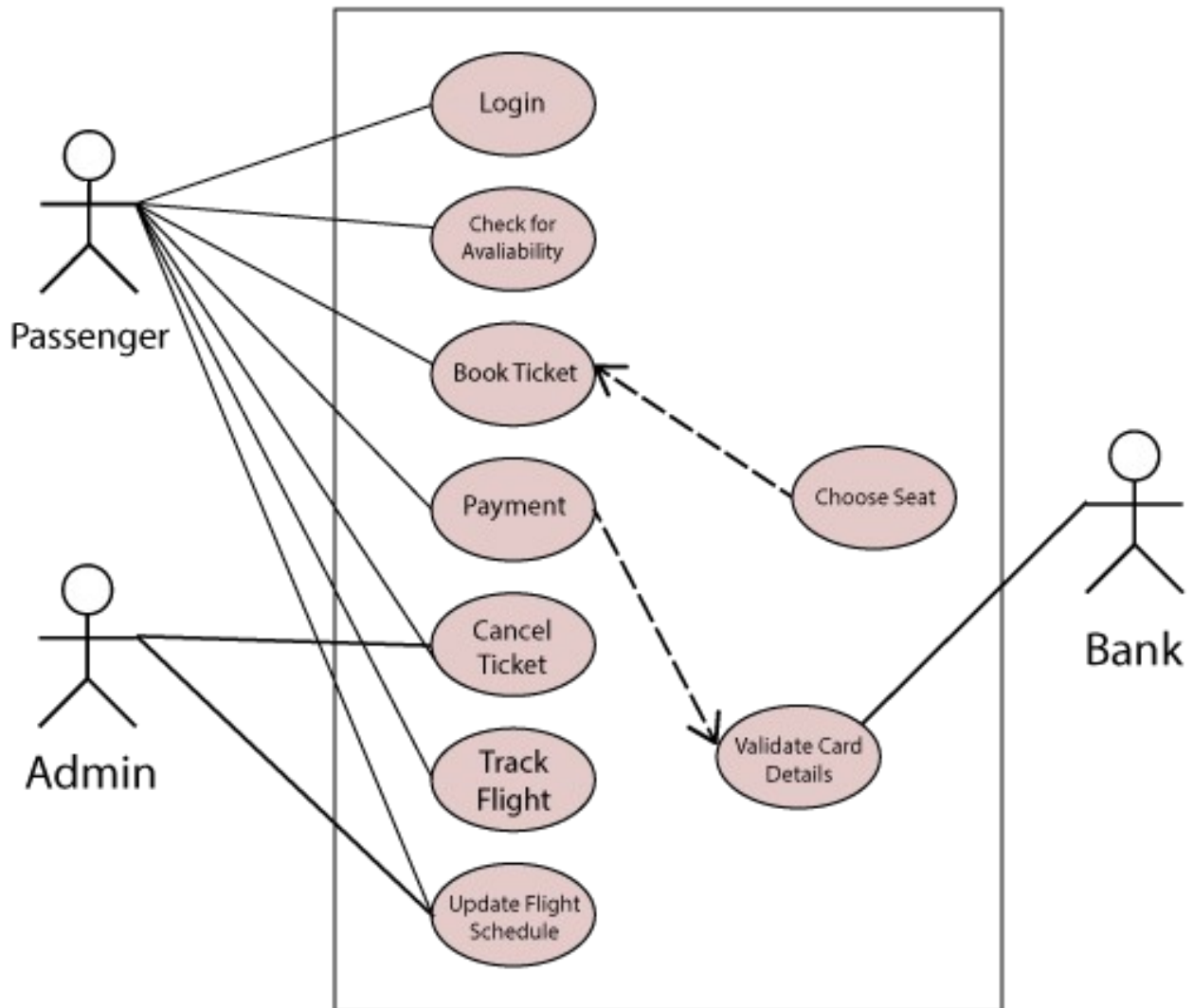
### 3.4 Create a Gantt chart.



## ~~~~~ Phase II ~~~~~

### 4. Functional Requirements

#### Use-Case Diagram



## 4.1 Name of Use-Case 1

### Flight Booking and Authentication

<b>Identifier</b>		UC-1
<b>Purpose</b>		This use-case describes the process of booking a flight ticket through the flight booking system
<b>Priority</b>		High
<b>Pre-conditions</b>		Launches the flight booking system
<b>Post-conditions</b>		The customer has successfully booked a flight ticket
<b>Typical Course of Action</b>		
<b>S#</b>	<b>Actor Action</b>	<b>System Response</b>
1	Launches the flight booking system	Flight booking system is displayed
2	Selects departure and arrival destinations, dates, and number of travelers	List of available flights matching criteria is displayed
3	Selects a flight and proceeds to the booking process	Flight details are displayed, and the customer is asked to confirm the booking
4	Confirms the booking and makes the payment	The booking is confirmed, and a ticket is issued
<b>Alternate Course of Action</b>		
<b>S#</b>	<b>Actor Action</b>	<b>System Response</b>
1	Enters Email and password	System authenticates the user credentials
2	Requests to reset password	User is granted access to the system if the credentials are valid, otherwise an error message is displayed
3	Requests to reset password	System sends password reset instructions to the user's email

**Table 1: UC-1**

## 4.2 Name of Use-Case 2 (and so on)

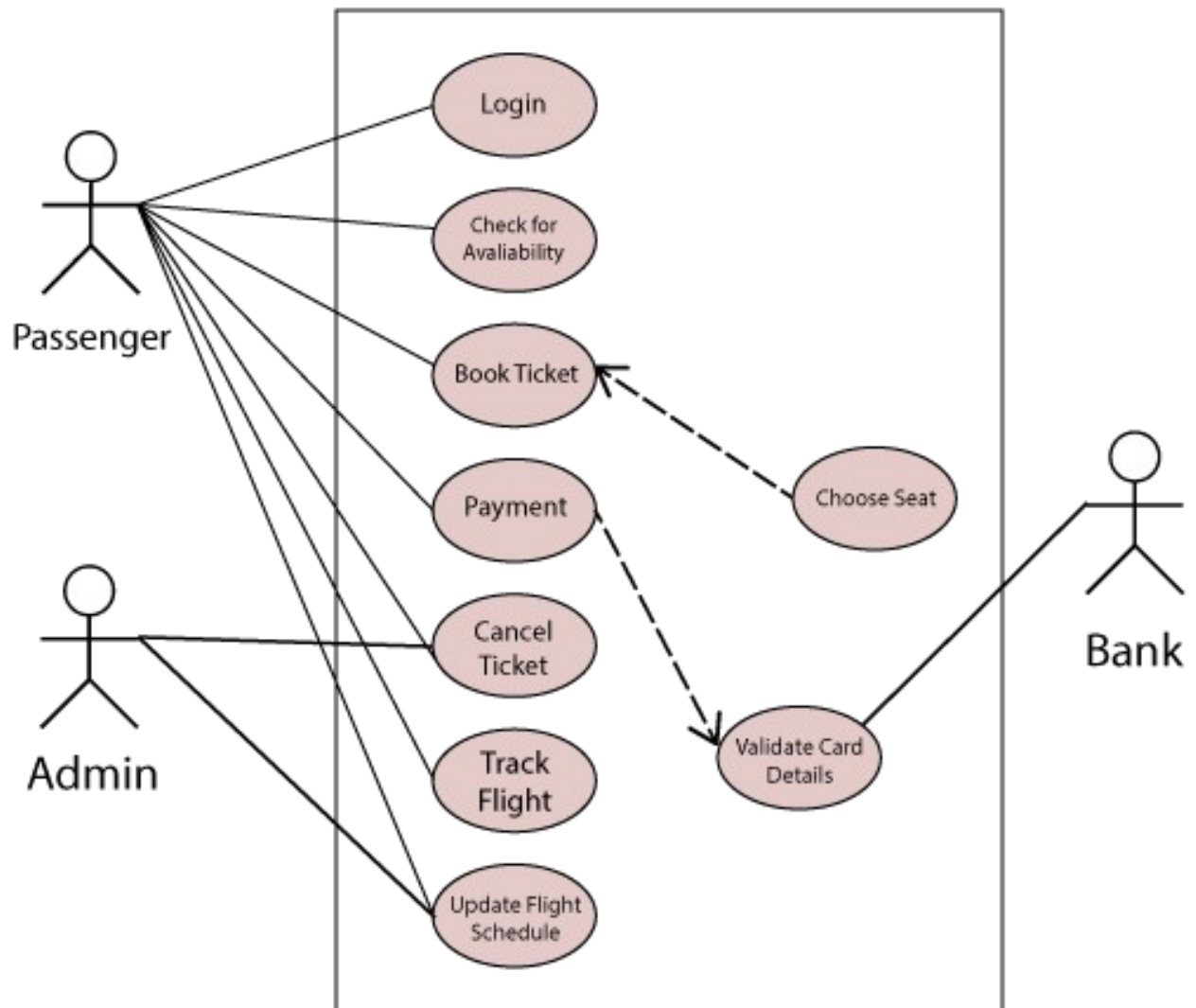
### Flight Availability and Booking

<b>Identifier</b>	UC-2
<b>Purpose</b>	The purpose of a flight booking system is to provide customers with an efficient and convenient way to search for, book, and manage their flight reservations.
<b>Priority</b>	High
<b>Pre-conditions</b>	The pre-conditions for using a flight booking system include having access to the internet, a computer or mobile device, and a valid payment method.
<b>Post-conditions</b>	The post-conditions of a flight booking system include a confirmed flight reservation, a receipt, and an itinerary.
<b>Typical Course of Action</b>	

S#	Actor Action	System Response
1	Check Flight Availability	The user can check the availability of flights based on their preferred departure and arrival times and dates.
2	Compare Flight Options	List of available flights matching criteria is displayed
<b>Alternate Course of Action</b>		
S#	Actor Action	System Response
1	Process Payment	The user can process payment for their flight booking using various payment methods such as credit cards, PayPal, or bank transfer.
2	Confirm Booking	The user can confirm their flight booking after successful payment and receive a booking confirmation. Message is displayed
3	The user can view their past flight bookings, including details such as booking dates, flight itineraries, and payment information.	The user can view their past flight bookings, including details such as booking dates, flight itineraries, and payment information.
4	Send Booking Confirmation	The system can automatically send a booking confirmation to the user's email or mobile phone after a successful flight booking.

## Requirements Analysis and Modeling

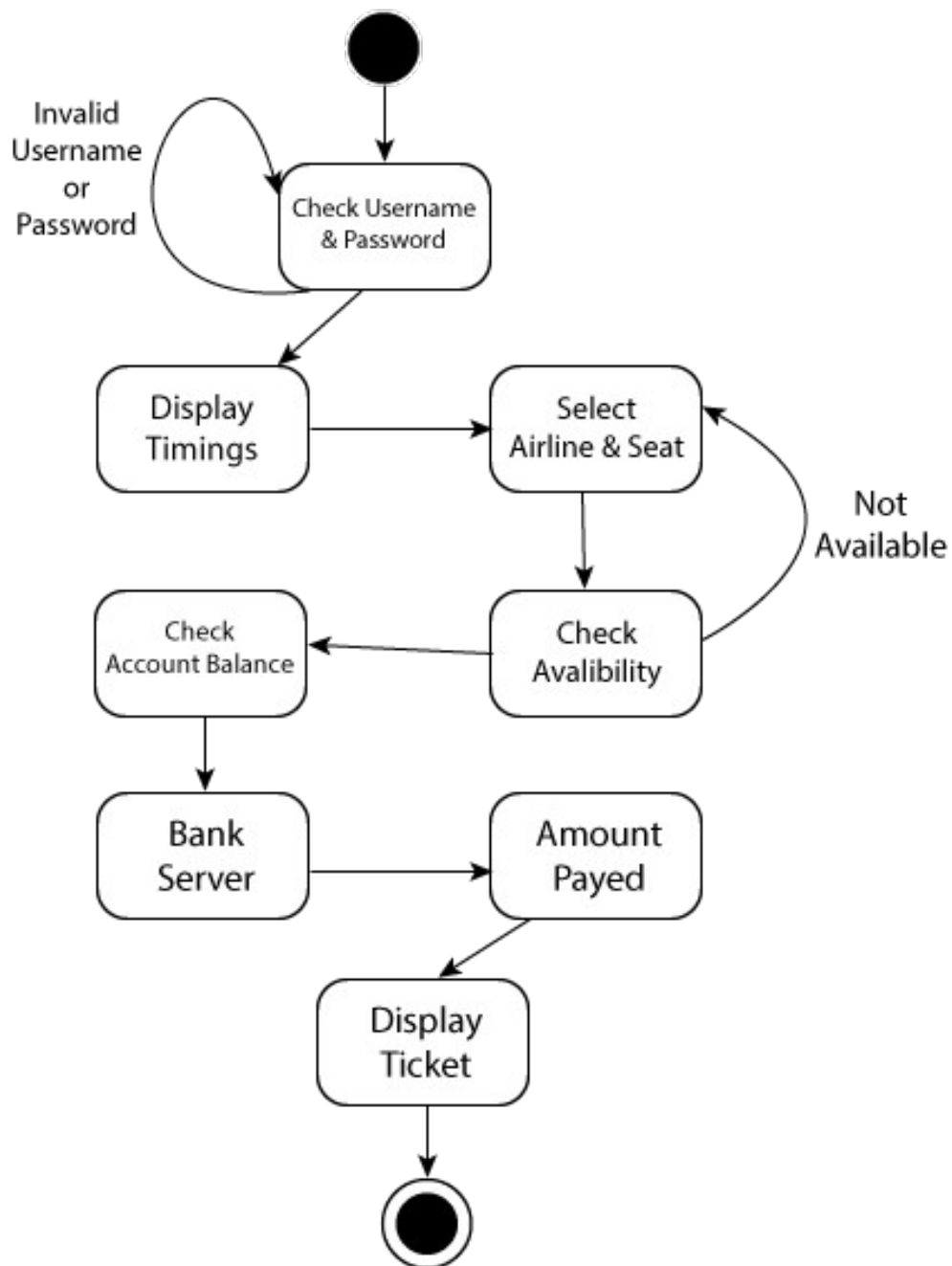
### Use-Case Diagram





<Flight Booking System>

State Diagram



### Decision Table

Conditions	Case 1	Case 2	Case 3	Case 4	Case 5
Login	T	T	T	T	F
Timings	T	T	T	T	F
Airline	T	T	F	T	F
Seat	T	F	T	T	F
Ticket Available	T	F	F	T	F
Account Balance	T	T	T	F	F
<b>Actions</b>					
Ticket Booked	✓	x	x	x	x
Ticket Not Booked	x	✓	✓	✓	✓

## 5. Nonfunctional Requirements

### 5.1 Performance Requirements

- **Response time for booking a flight:** This requirement specifies that the system should respond within 5 seconds when a user books a flight. This is an important performance requirement as users expect fast and efficient booking processes. A slow response time can result in frustration and decreased user satisfaction.
- **Handling of concurrent users:** The system should be able to handle at least 1000 concurrent users. This requirement ensures that the system can handle a large number of users simultaneously without slowing down or crashing. This is important for the system's scalability and to provide a good user experience.

## 5.2 Safety Requirements

- **Backup mechanism:** The system should have a backup mechanism in place to ensure that user data is not lost in case of system failure. This requirement is important for data protection and to prevent loss of user information.
- **Compliance with aviation safety regulations:** The system should comply with aviation safety regulations and standards. This requirement ensures that the system meets legal requirements and operates in a safe manner.

## 5.3 Security Requirements

- **Secure login and authentication:** The system should have secure login and authentication processes to protect user data. This requirement ensures that user data is protected from unauthorized access.
- **Encryption of payment information:** Payment information should be encrypted during transmission and storage. This requirement ensures that sensitive information is protected and secure.
- **Regular security updates:** The system should have regular security updates and vulnerability scans to ensure protection against threats. This requirement ensures that the system is kept secure and protected against potential security breaches.

## 5.4 Additional Software Quality Attributes

- **User-friendly interface:** The system should be user-friendly and easy to navigate. This requirement ensures that users have a positive experience and can easily find what they need.
- **Accessibility on multiple devices:** The system should be accessible on multiple devices, including desktops, laptops, and mobile devices. This requirement ensures that users can access the system from a variety of devices and locations.

- **Error handling:** The system should have a clear and concise error handling process to ensure a smooth user experience. This requirement helps prevent frustration and ensures that users can resolve any issues quickly and easily.

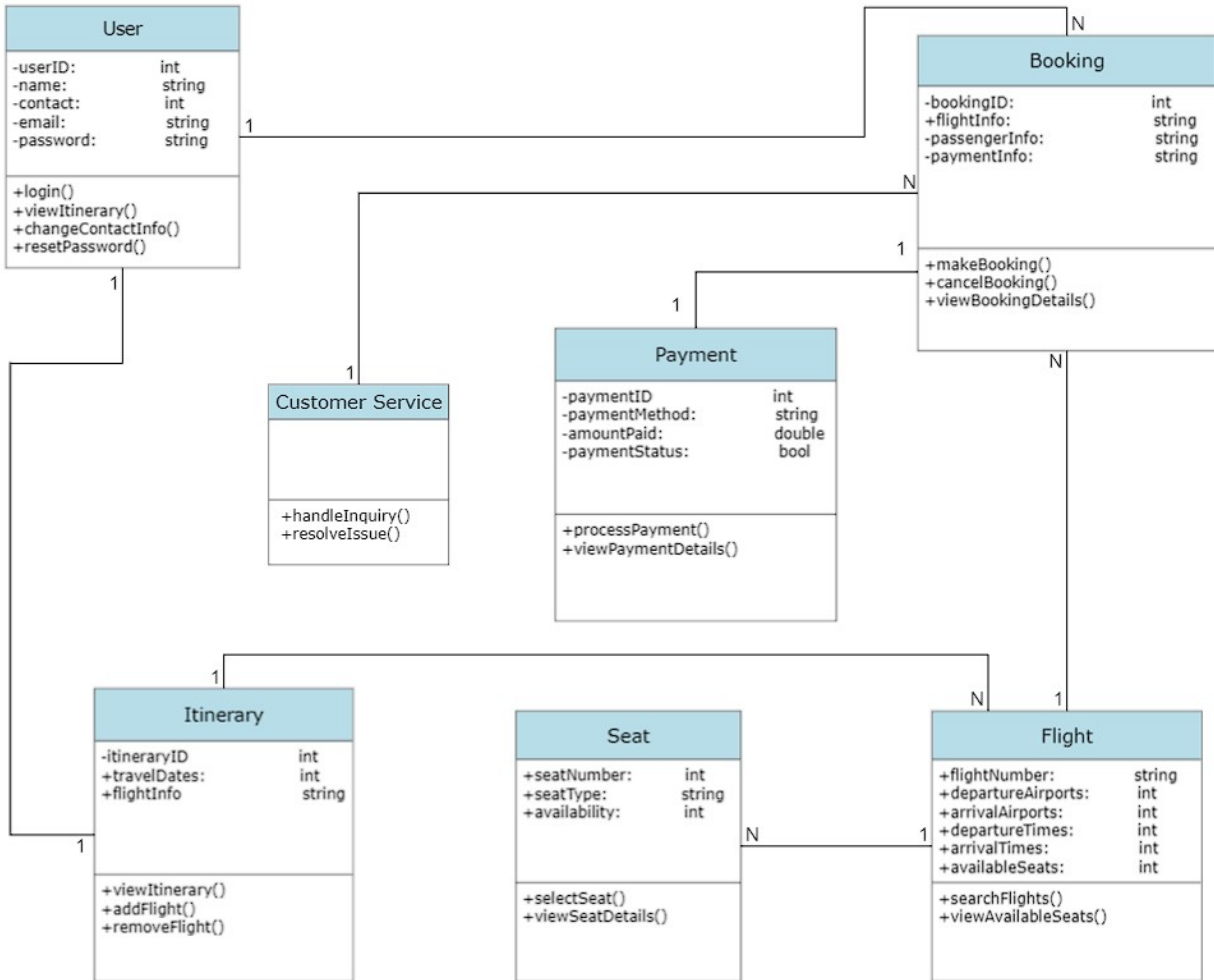
## 6. Other Requirements

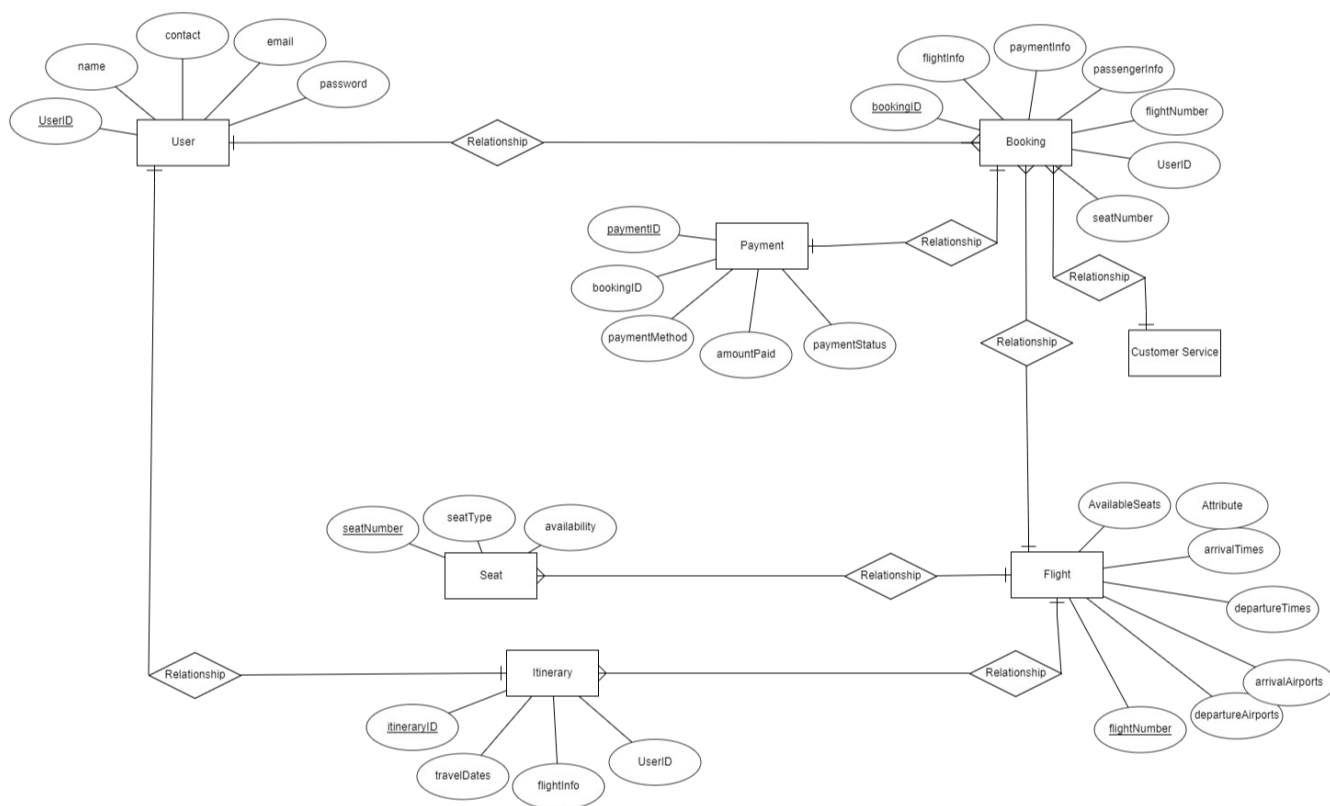
- **Seat selection and special requirements:** The system should provide options for seat selection and special requirements (e.g. meal preferences, accessibility needs). This requirement ensures that users can customize their flight experience to meet their needs.
- **Modification and cancellation:** The system should allow users to modify and cancel their bookings. This requirement provides users with flexibility and convenience.
- **Real-time flight information:** The system should provide real-time flight information and updates. This requirement ensures that users have accurate and up-to-date information about their flight

## 7. Designing

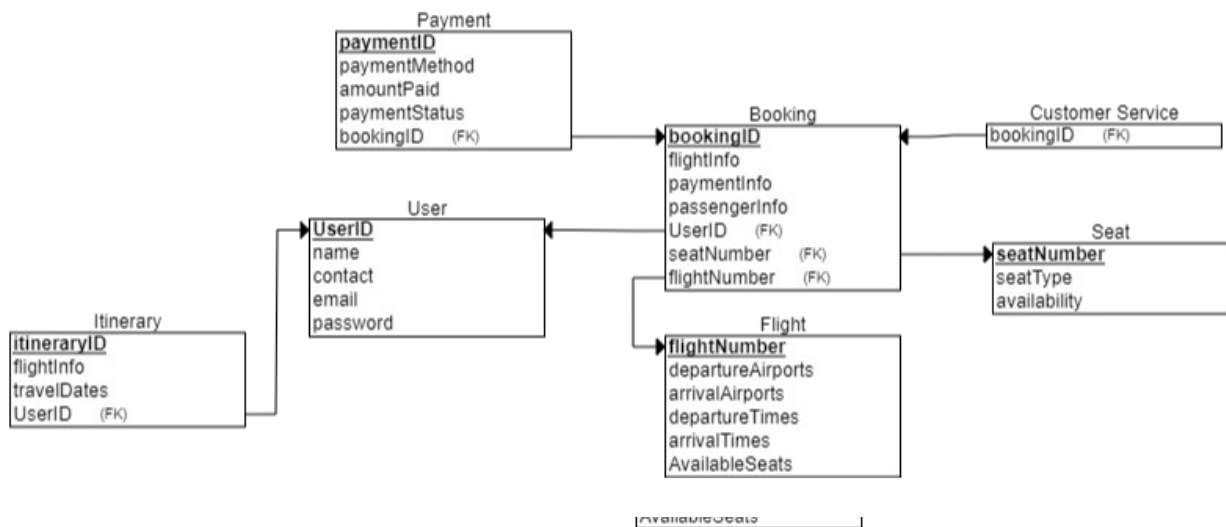
### 7.1 Complete class diagram

# <Flight Booking System>





## 7.4 Physical design of your database



## 7.5 Information on use of design patterns while designing the modules

**User:** This class would represent a user of the system and would contain attributes such as user ID, name, contact information, and password.

**Flight:** This class would represent a flight and would contain attributes such as flight number, departure airport, arrival airport, departure time, arrival time, and number of available seats.

**Booking:** This class would represent a booking made by a user and would contain attributes such as booking ID, flight number, user ID, seat number, and payment information.

**Payment:** This class would represent a payment made by a user for a booking and would contain attributes such as payment ID, booking ID, payment method, and amount paid.

**Seat:** This class would represent a seat on a flight and would contain attributes such as seat number, seat type, and availability.

**Itinerary:** This class would represent a collection of flights for a user and would contain attributes such as itinerary ID and user ID.

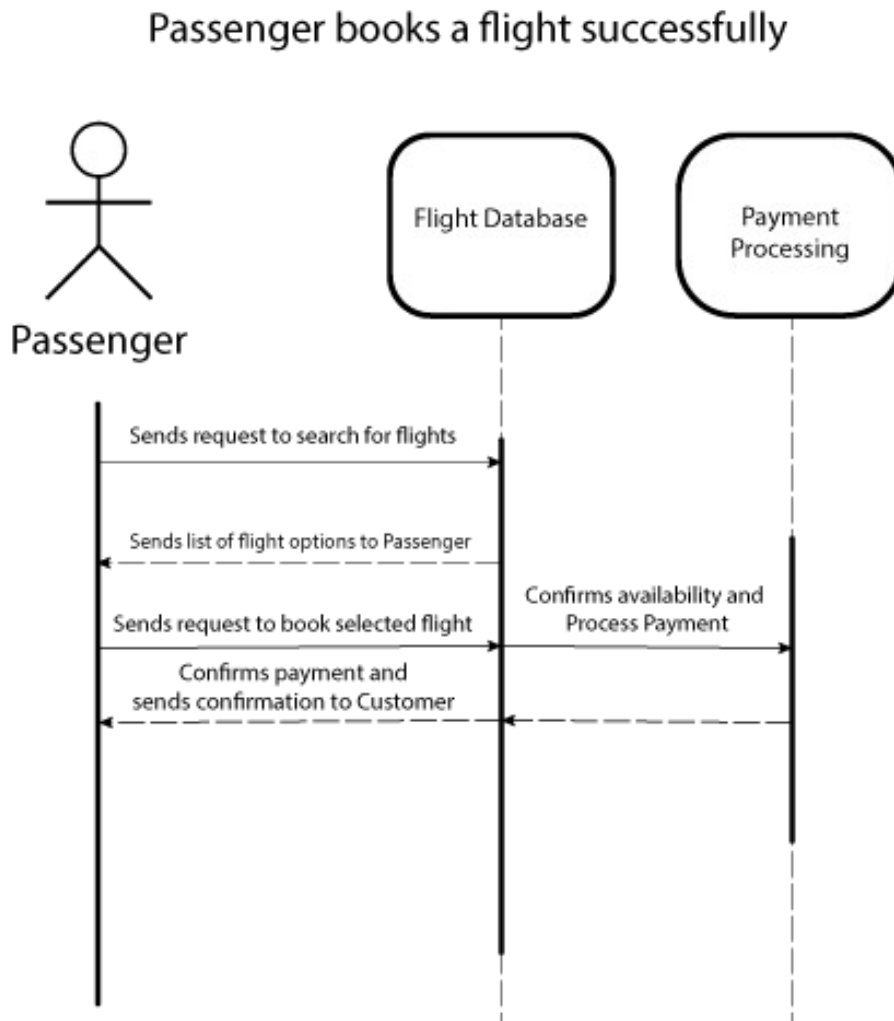
**Customer Service:** class would represent the customer service component of the Flight Booking System. It would be responsible for handling customer inquiries, resolving customer complaints, and providing assistance with the booking process. The class would contain the following attributes:

The class diagram would show the relationships between these classes, such as the relationship between the User and Booking classes, or the relationship between the Flight and Seat classes

## 7.6 Make a Sequence and a Collaboration diagram of following.

### 7.6.1 Scenario 1

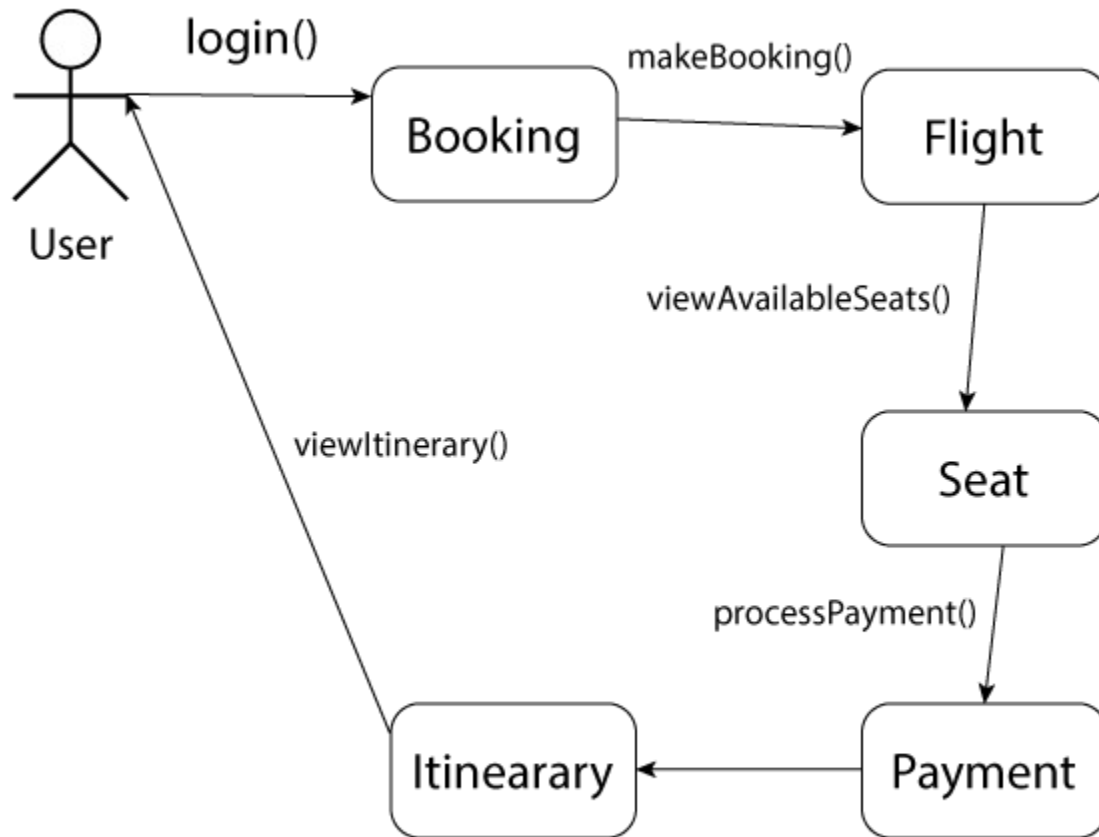
#### Sequence Diagram



#### Collaboration Diagram



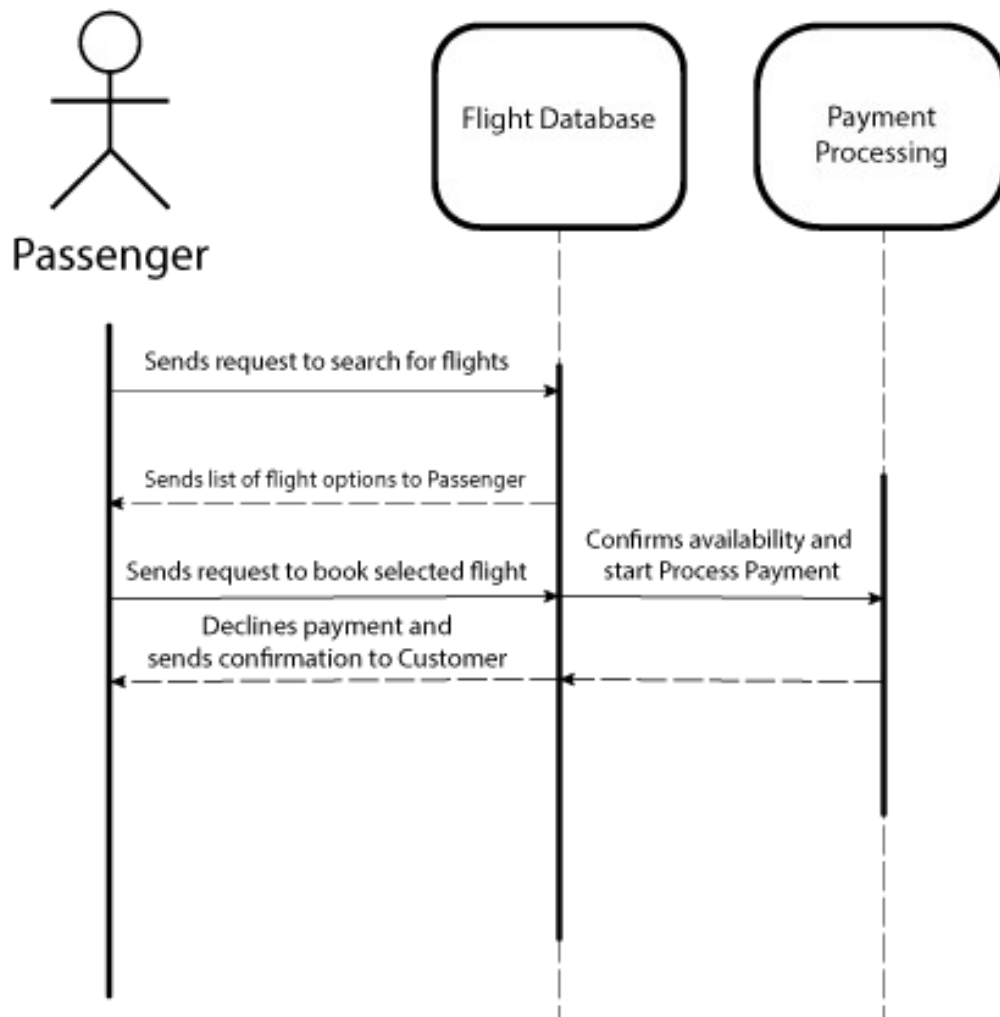
<Flight Booking System>



7.6.2 Scenario 2

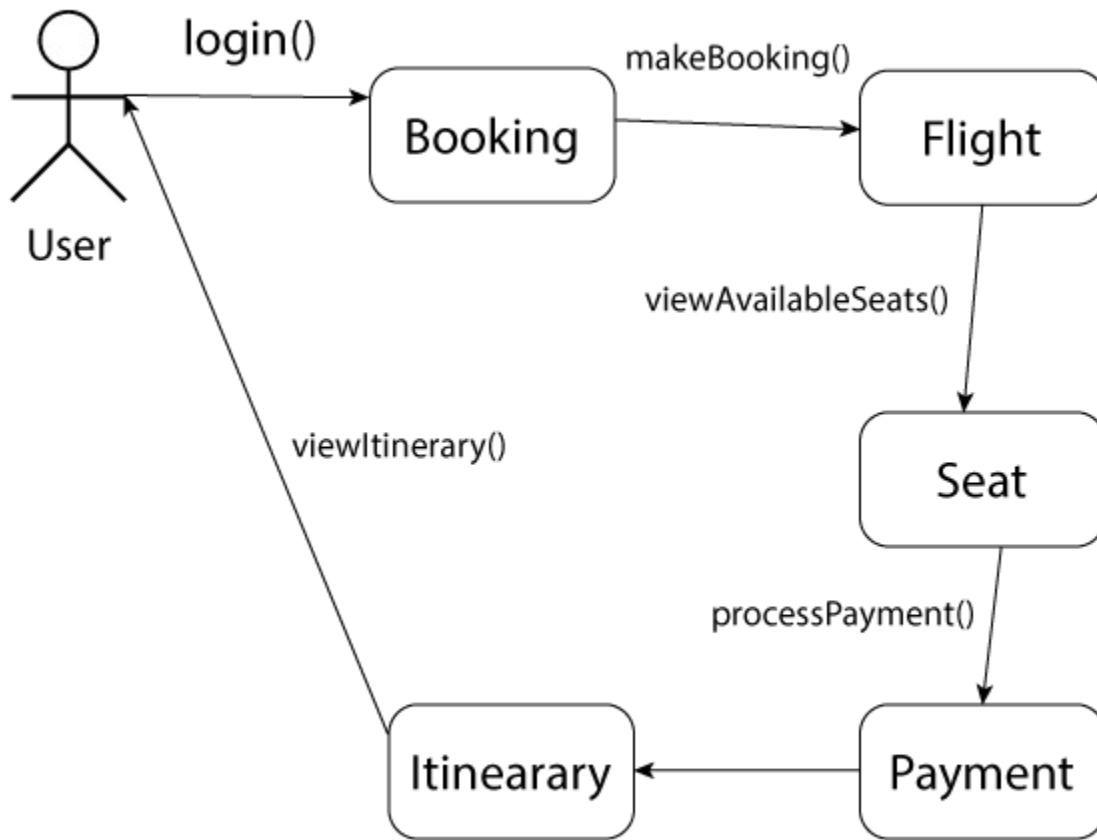
Sequence Diagram

Passenger payment is declined



<Flight Booking System>

Collaboration Diagram



## 8. Estimation

### 8.1 Cost Benefit Estimation

	Initial Cost	Year 1	Year 2	Year 2
<b>Cost</b>				
Software Development	\$120,000	\$50,000	\$50,000	\$50,000
Hardware	\$30,000	\$5,000	\$5,000	\$5,000
Marketing and Advertising	\$10,000	\$15,000	\$15,000	\$15,000
Operational Expenses	\$15,000	\$20,000	\$20,000	\$20,000
<b>Total Cost</b>	<b>\$175,000</b>	<b>\$90,000</b>	<b>\$90,000</b>	<b>\$90,000</b>
<b>Benefits</b>				
Ticket Sales		\$200,000	\$200,000	\$200,000
Fees for Additional Services		\$15,000	\$15,000	\$15,000
<b>Total Benefit</b>		<b>\$215,000</b>	<b>\$215,000</b>	<b>\$215,000</b>
Productivity		\$215,000	\$215,000	\$215,000
<b>Net Benefit</b>		<b>\$95,000</b>	<b>\$95,000</b>	<b>\$95,000</b>

#### 8.1.1 Return of Investment (ROI) {1st year}

$$\text{ROI} = (\text{Benefits} - \text{Cost}) / \text{Cost}$$

the ROI for the first year would be:  $(\$215,000 - \$175,000) / \$175,000 = 0.2286$  or 22.86%

#### 8.1.2 %gain on ROI {1st year}

Use the formula:  $\% \text{ Gain on ROI} = (\text{ROI}) \times 100$

% gain on ROI would be:  $22.86\% \times 100 = 2286\%$ .

#### 8.1.3 Payback Period in years

The total initial cost is \$175,000, the total costs for years 1, 2, and 3 are \$90,000 each, and the net benefits for years 1, 2, and 3 are \$95,000 each.

The payback period for year 1 can be calculated as:  $\$175,000 / \$95,000 = 1.84$  years. This means it takes 1.84 years to recover the initial cost in year 1.

For year 2, the payback period can be calculated as:  $(\$175,000 - \$90,000) / \$95,000 = 0.84$  years. This means it takes 0.84 years to recover the remaining initial cost in year 2.

## 8.2 FP based Estimation

Information Domain Value	Count	Simple	Average	Complex	Total
External Inputs (EI)	20	3	4	6	80
External Outputs (EO)	15	4	5	7	75
External Inquiries (EQ)	10	3	4	6	40
Input Logical Files (ILF)	30	7	10	15	300
External Logical Files (EIF)	15	5	7	10	105
Total					600

Questions	Weighting Factor
Reliable backup and recovery?	4
Are data communications required?	1
Are there distributed processing functions?	0
Is performance critical?	3
Will system run in existing heavily utilized operating environment?	5
Online Data Entry?	4
Online data entry require input transaction over multiple screens or operations?	4
Are ILF updated online?	3
Are input, output, files or inquires complex?	3
Is the internal processing complex?	2
Code designed to be reusable?	2
Conversion and installation included in design?	4
System designed for multiple installations in different organizations?	5
Application designed to facilitate change and ease of use by the user?	4
$\Sigma fi =$	44

Cost = \$7500/Month  
 Effort = 36.9 P.M  
 UFP = 600  
 $\Sigma fi = IDI = 44$   
 COST = 7744

$VPF = (0.01 * 44) + 0.65 = 1.09$   
 $FP = 1.09 * 600 = 654$   
 $Productivity = 654/36.9 = 17.72$   
 $Cost\ Per\ function = 423.25$   
 $Total\ Project\ Cost = 423.25 * 654 = 276969$   
 $LOC = 654 * 60 = 39240$   
 $KLOC = 39240/1000 = 39.24$

### 8.3 COCOMO Estimation

	a	b	c	d
Organic	2.4	1.05	2.5	0.34
Semidetached	3.0	1.12	2.5	0.35
Embedded	3.6	1.20	2.5	0.32

KLOC = 39.24

#### Organic

$Effort = 2.4(39.24)^{1.05} = 113.14$

$Time\ Development = 2.5(113.14)^{0.34} = 12.47$

$Average\ Staff\ Time = 113.14/12.47 = 9.07$

Cocomo Productivity = 0.34

## 9. References

*<List all books, conference papers, journal articles, websites, etc. used in preparing the content of this SRS. Provide enough information so that the reader could access a copy of each reference, including title, author, volume/edition number, page number(s), and publication year. Mention complete URLs for websites.>*

*< Must use Zotero reference Manager to maintain references in every phase in this section.>*

**<Flight Booking System>**

**Title:** Software Requirements Specification for Airline Ticket Reservation

**Author:** *David J. Wardell*

**volume/edition:** 4.0

**Page Number:** 27

**Publication year:** December 16, 2018

**Link:** [https://courses.cs.ut.ee/MTAT.03.306/2018_fall/uploads/Main/team6.pdf](https://courses.cs.ut.ee/MTAT.03.306/2018_fall/uploads/Main/team6.pdf)

**Title:** AIRLINE RESERVATION SYSTEMS

**Author:** Elchin Aghazada

**volume/edition:** 1.0

**Page Number:** 114

**Publication year:** July 5, 1991

**Link:** [https://www.researchgate.net/profile/David-Wardell/publication/333812517_AIRLINE_R.pdf](https://www.researchgate.net/profile/David-Wardell/publication/333812517_AIRLINE_R.pdf)

**Title:** *AIRLINE AND AIRPORT MANAGEMENT*

**Author:** *Mr. Yugender, Asso. Prof*

**volume/edition:** 1.0

**Page Number:** 200

**Publication year:** June 8, 2022

**Link:** [https://mrcet.com/downloads/digital_notes/AE/IV%20Year/Airline%20and%20Airport%20Management.pdf](https://mrcet.com/downloads/digital_notes/AE/IV%20Year/Airline%20and%20Airport%20Management.pdf)

**Title:** Airline Reservations Systems.

**Author:** *Mr. Yugender, Asso. Prof*

**volume/edition:** 2

**Page Number:** 18

**Publication year:** September 12, 1998

**Link:** <https://www.jstor.org/stable/249202>

**Link:** <https://rezdy.com/blog/what-is-online-booking-system/>

## Appendix A: Glossary

**Actor:** An individual or entity that interacts with the system.

**Customer:** An individual who uses the flight booking system to search for, book, and manage their flight reservations.

**Agent:** An individual who works for the airline and uses the flight booking system to assist customers with their flight bookings.

**Administrator:** An individual who has access to the back-end of the flight booking system and is responsible for maintaining and updating the system.

**Use Case:** A set of actions that an actor can perform in the system.

**Search Flights:** A use case where the customer can search for available flights based on their travel dates, destination, and other criteria.

**View Flight Info:** A use case where the customer can view detailed information about a flight, including its schedule, seat availability, and fare.

**Select Flight:** A use case where the customer can select a flight and proceed to the next step in the booking process.

**Manage Passenger:** A use case where the customer can enter information about their passengers, such as their names, dates of birth, and passport information.

**Payment:** A use case where the customer can enter their payment information and complete the booking process.

**Confirm Booking:** A use case where the system confirms the customer's flight reservation and provides them with a receipt and itinerary.

**Retrieve Booking:** A use case where the customer can retrieve their flight reservation information by entering their booking reference or other identifying information.

**Cancel Booking:** A use case where the customer can cancel their flight reservation.

**Manage Flight Info:** A use case where the administrator can add, update, and remove flight information in the system.