PROJECT REPORT

ON

AIRLINE RESERVATION SYSTEM

B.Tech (CE) Sem-IV

In the Subject of

Software Engineering Pratices

PRANAV SAKARIYA (CE-114) (19CEUEG028)

VIDUR SAKHIYA(CE-115) (19CEUOG101)

Under the Guidance of

Prof. Brijesh S. Bhatt

Prof. Pincal C. Chauhan

Department Of Computer Engineering

Faculty Of Technology,

Dharmsinh Desai University, Nadiad.

DHARMSINH DESAI UNIVERSITY

College Road, NADIAD-387001 (Gujarat)

CERTIFICATE

This is to certify that the term work carried out in the subject of **Software Enineering Practice** and recorded in this report is bonafide

work of **Mr. Pranav Sakariya (Roll No.: 114, Id: 19CEUEG028)**

and

Mr. Vidur Sakhiya (Roll No.: 115, Id: 19CEUOG101)

of **B.Tech Semester 4**th in the branch of Computer Engineering during the academic year 2020-21.

Prof. Brijesh S. Bhatt   Dr. C.K. Bhensdadia
(Project Guide and Associate Professor)   Head of CE Dept.,
Faculty of Technology,   Faculty of Technology,
Dharmsinh Desai University,   Dharmsinh Desai
University,
Nadiad.   Nadiad.

Contents

Abstract

Airline reservation System is a computerized system used to store and retrieve information and conduct transactions related to air travel. The project is aimed at exposing the relevance and importance of Airline Reservation Systems. It is projected towards enhancing the relationship between customers and airline agencies through the use of ARSs, and thereby making it convenient for the customers to book the flights as when they require such that they can utilize this software to make reservations.

Introduction

- This software has two parts. First is user part and the administrator part. User part is used as a front end and administrator is the back end. Administrator is used by airline authority. It will allow the customers to access database and allow new customers to sign up for online access.

- The system allows the airline passenger to search for flights that are available between the two travel cities, namely the "Departure city" and "Arrival city" for a particular departure and arrival dates. The system displays all the flight's details such as flight no, name, price and duration of journey etc. After search the system display list of available flights and allows customer to choose a particular flight. Then the system checks for the availability of seats on the flight. If the seats are available then the system allows the passenger to book a seat

- . To book a flight the system asks the customer to enter his details such as name, address, city, state, and credit card number and contact number. Then it checks the validity of card and book the flight and update the airline database and user database. The system also allows the customer to cancel his/her reservation, if any problem occurs.

- The main purpose of this software is to reduce the manual errors involved in the airline reservation process and make it convenient for the customers to book the flights as when they require such that they can utilize this software to make reservations, modify reservations or cancel a particular reservation.

2. Software Requirement Specifications

Users of the System:-

- Customers
- System Administrator

functionality requirement of your system.

1. PASSENGER LOGIN

R.1 Registration

Description: for registration reader have should enter the detail.

Input: enter the first name, last name, mobile no, email id, password

Output: display successful message and reader id

R.2 login Description:

Readers must enter id and password for the login

Input: Enter reader id and password

Output: all options are display.

Next state: Passenger will choose any option if password is valid re-login if password is invalid

2.Manage flights details

R.1 Check flights that are going

Description : Admin of system can check total flights between given   departure and destination that will going on given time slot.

Input : flight departure and destination , time .

Output : All flights with flights id .

R.2 Change/Cancel flights

R.2.1: Change timing of flight

Description : Admin can reschedule flight

Input : Flight departure and destination,Flight Id

Output : Confirmation message of rescheduling.

R.2.2: Cancel flights

Description : Admin can remove available flights

Input : Flight departure and destination , Flight id

Output : confirmation message of deletion of flight.

3.Manage tickets details

R.1 Availibility of tickets

Description: user can see the available flight

Input: user has to enter origin ,destination, and date .

Next State: if flight is available within time then we can reserve the tickets.

R.2 Reservation of tickets

Description: user can reserve ticket for their particular root

Input: user has to enter origin ,destination, and date .

Next State: this will redirect to all flight availabe for particular root and   passenger can book ticket accordingly.

R.3 Ticket status:

The system shall allow a user to view all information about his trip.

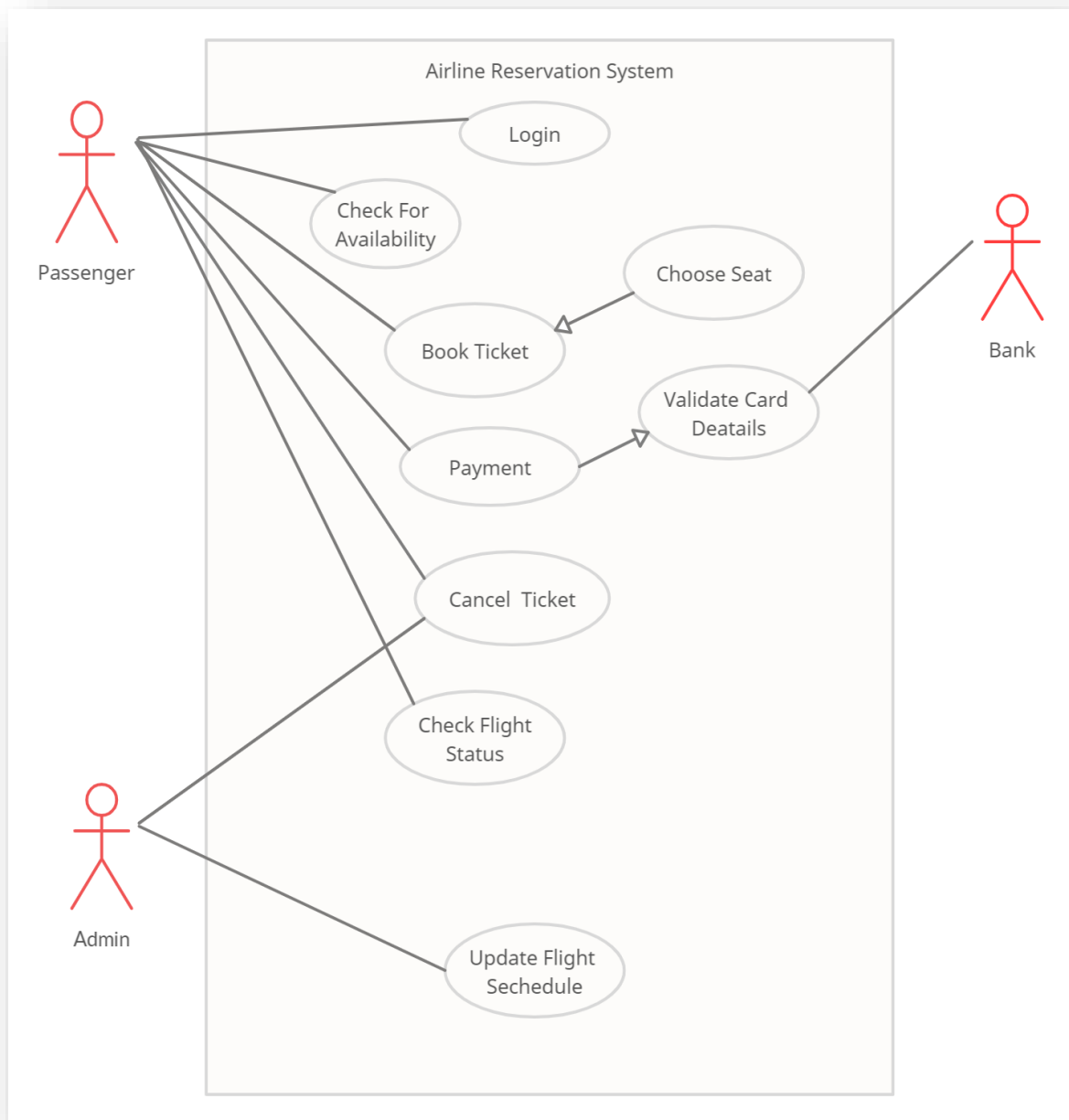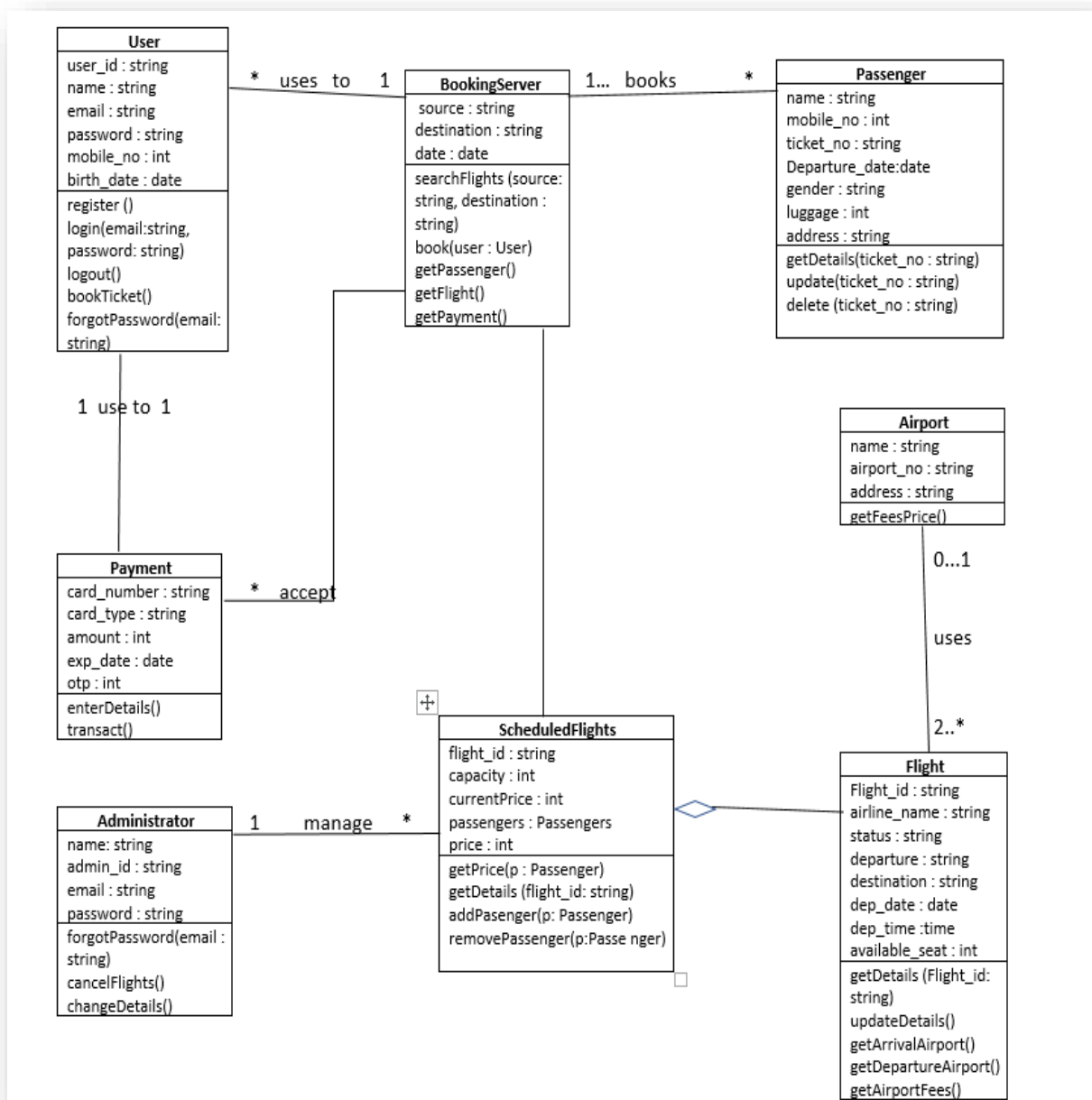### R.4 Cancellation of tickets

Description: user can cancel the available flight

The system shall also give the user an option to cancel a confirmed ticket

3. Designs

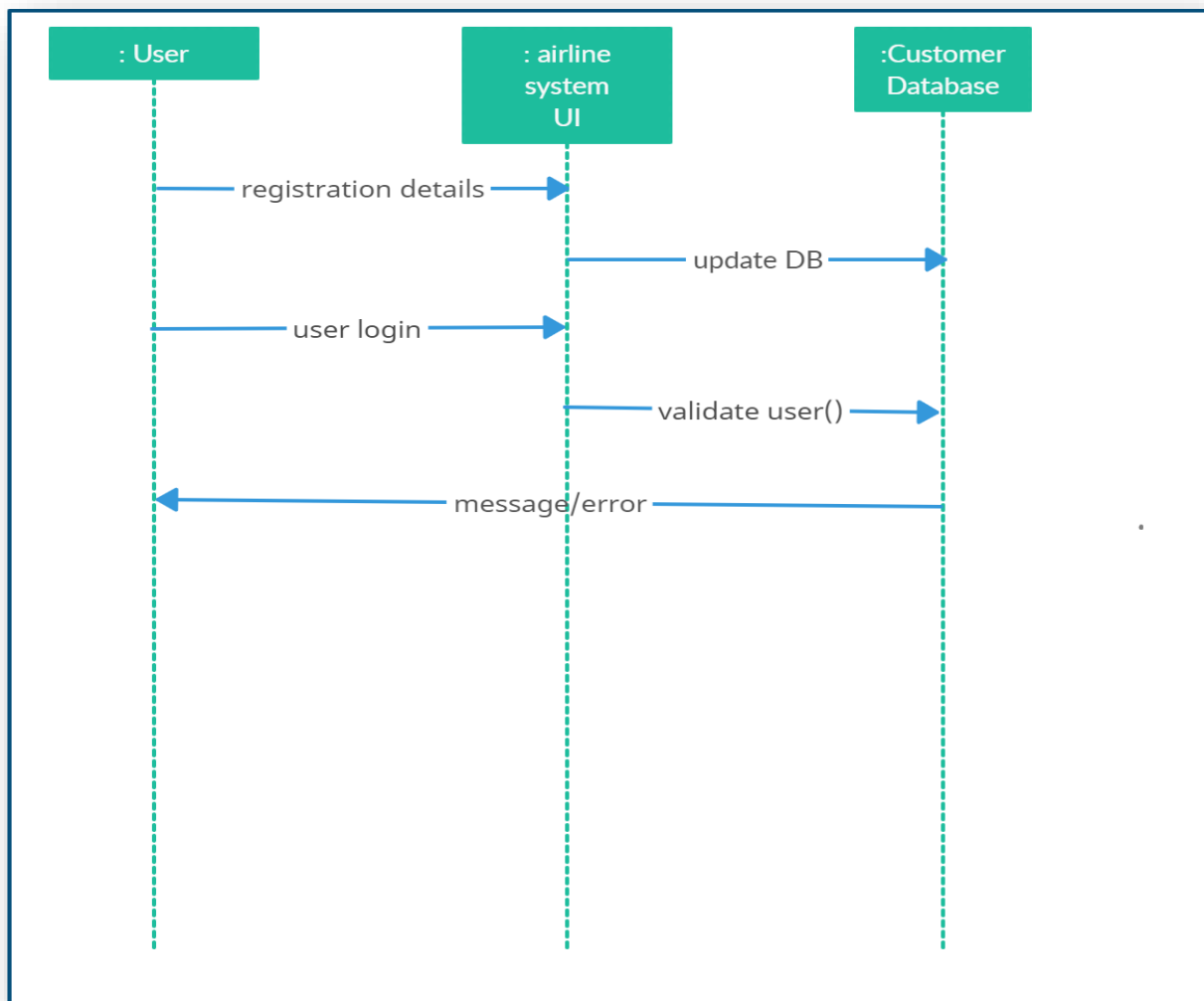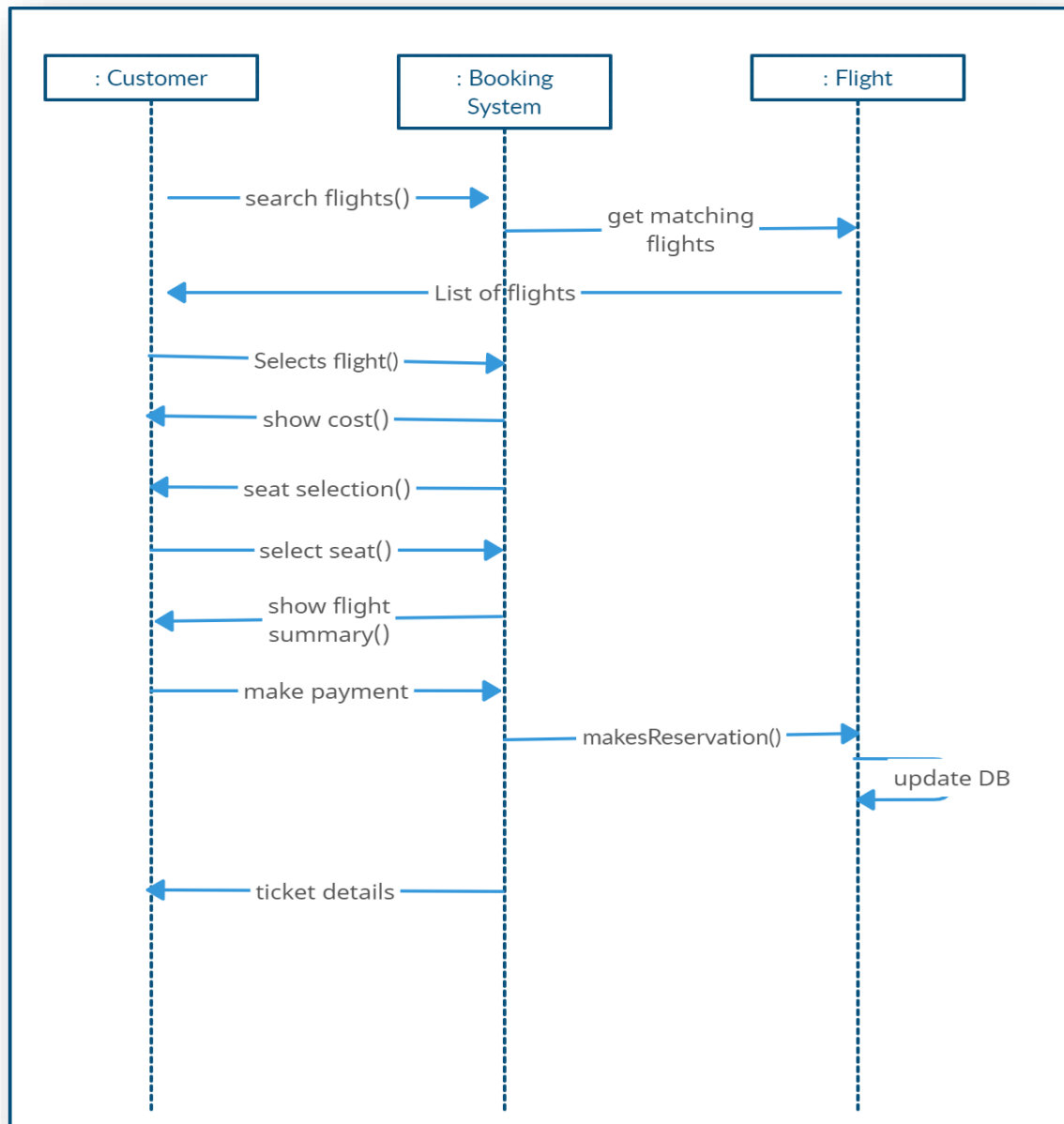## 1. Usecase Diagram :-

Airline Reservation System

Passenger

Login

Check For Availability

Choose Seat

Book Ticket

Validate Card Deatails

Bank

Payment

Cancel Ticket

Check Flight Status

Admin

Update Flight Sechedule

2. Class Diagram :-

User
user_id : string
name : string
email : string
password : string
mobile_no : int
birth_date : date
register ()
login(email:string, password: string)
logout()
bookTicket()
forgotPassword(email: string)

BookingServer
source : string
destination : string
date : date
searchFlights (source: string, destination : string)
book(user : User)
getPassenger()
getFlight()
getPayment()

Passenger
name : string
mobile_no : int
ticket_no : string
Departure_date:date
gender : string
luggage : int
address : string
getDetails(ticket_no : string)
update(ticket_no : string)
delete (ticket_no : string)

* uses to 1   1... books   *

1 use to 1

Payment
card_number : string
card_type : string
amount : int
exp_date : date
otp : int
enterDetails()
transact()

* accept

Airport
name : string
airport_no : string
address : string
getFeesPrice()

0...1

uses

2..*

Administrator
name: string
admin_id : string
email : string
password : string
forgotPassword(email : string)
cancelFlights()
changeDetails()

1   manage   *

ScheduledFlights
flight_id : string
capacity : int
currentPrice : int
passengers : Passengers
price : int
getPrice(p : Passenger)
getDetails (flight_id: string)
addPasenger(p: Passenger)
removePassenger(p:Passe nger)

Flight
Flight_id : string
airline_name : string
status : string
departure : string
destination : string
dep_date : date
dep_time :time
available_seat : int
getDetails (Flight_id: string)
updateDetails()
getArrivalAirport()
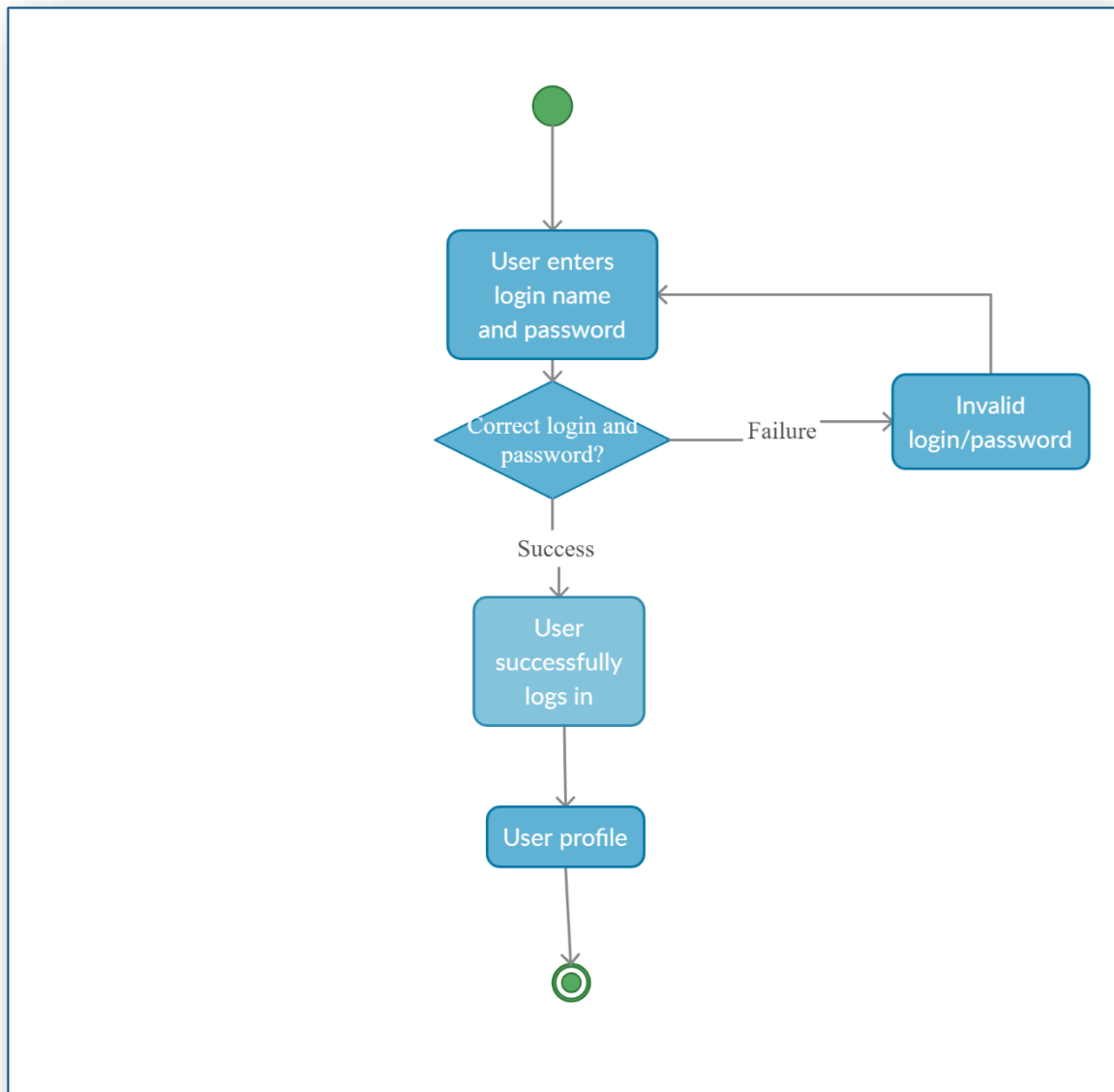getDepartureAirport()
getAirportFees()

3. Sequence Diagram :-

Sequence diagram of login/Register
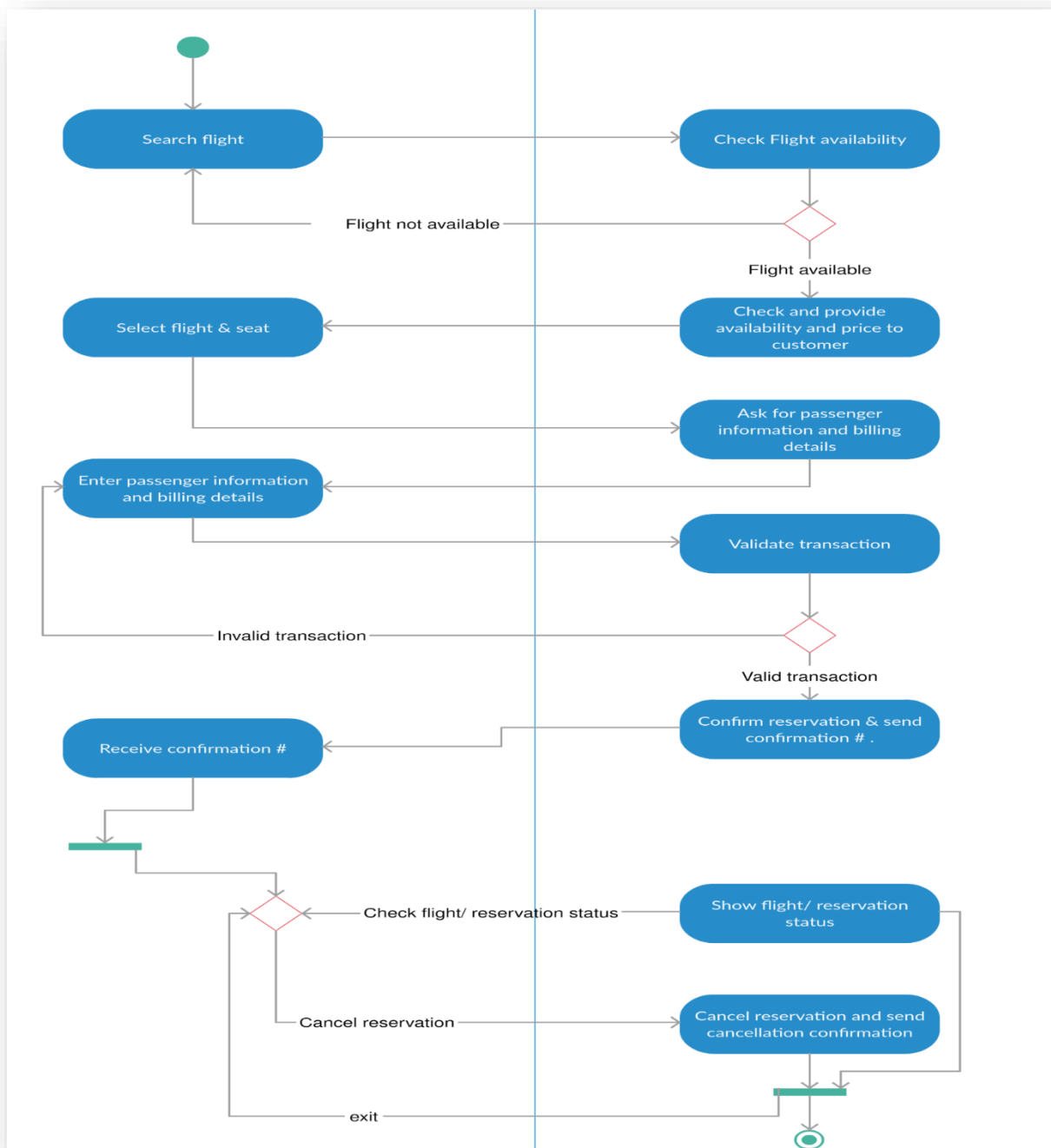
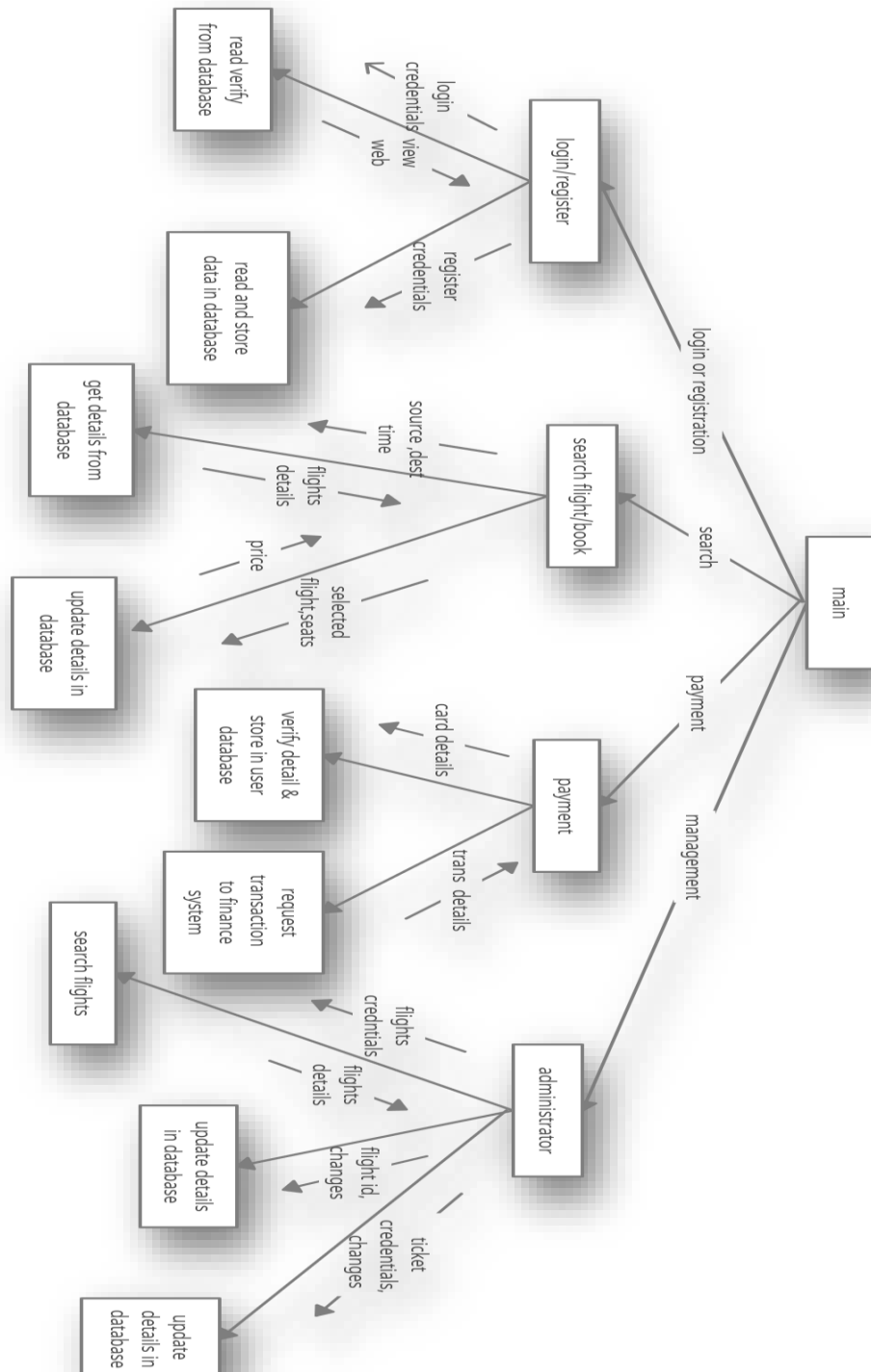**Sequence diagram of Book Tickets**

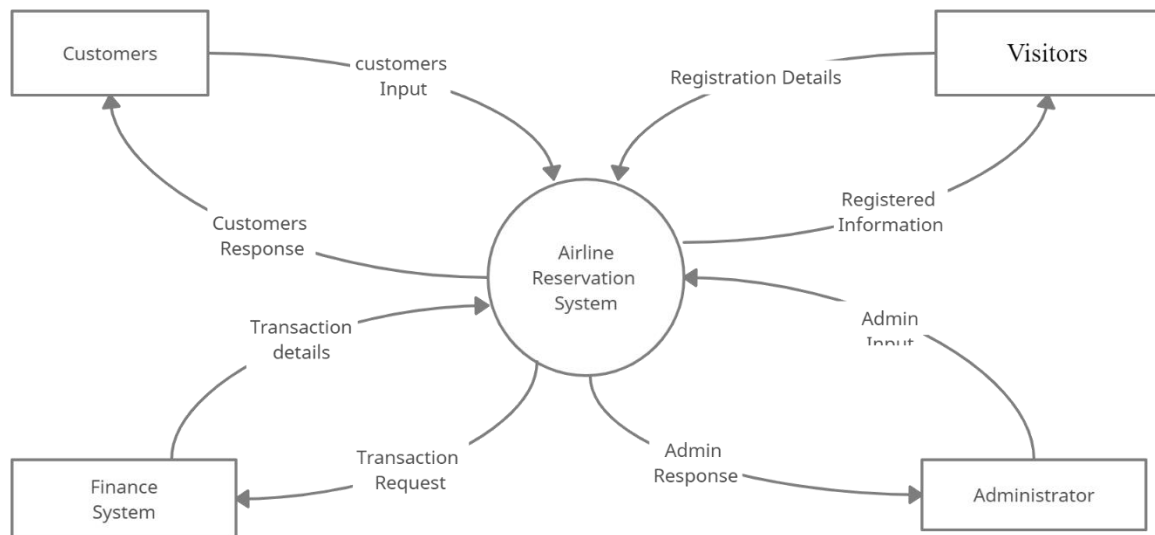# 4. Activity Diagram :-

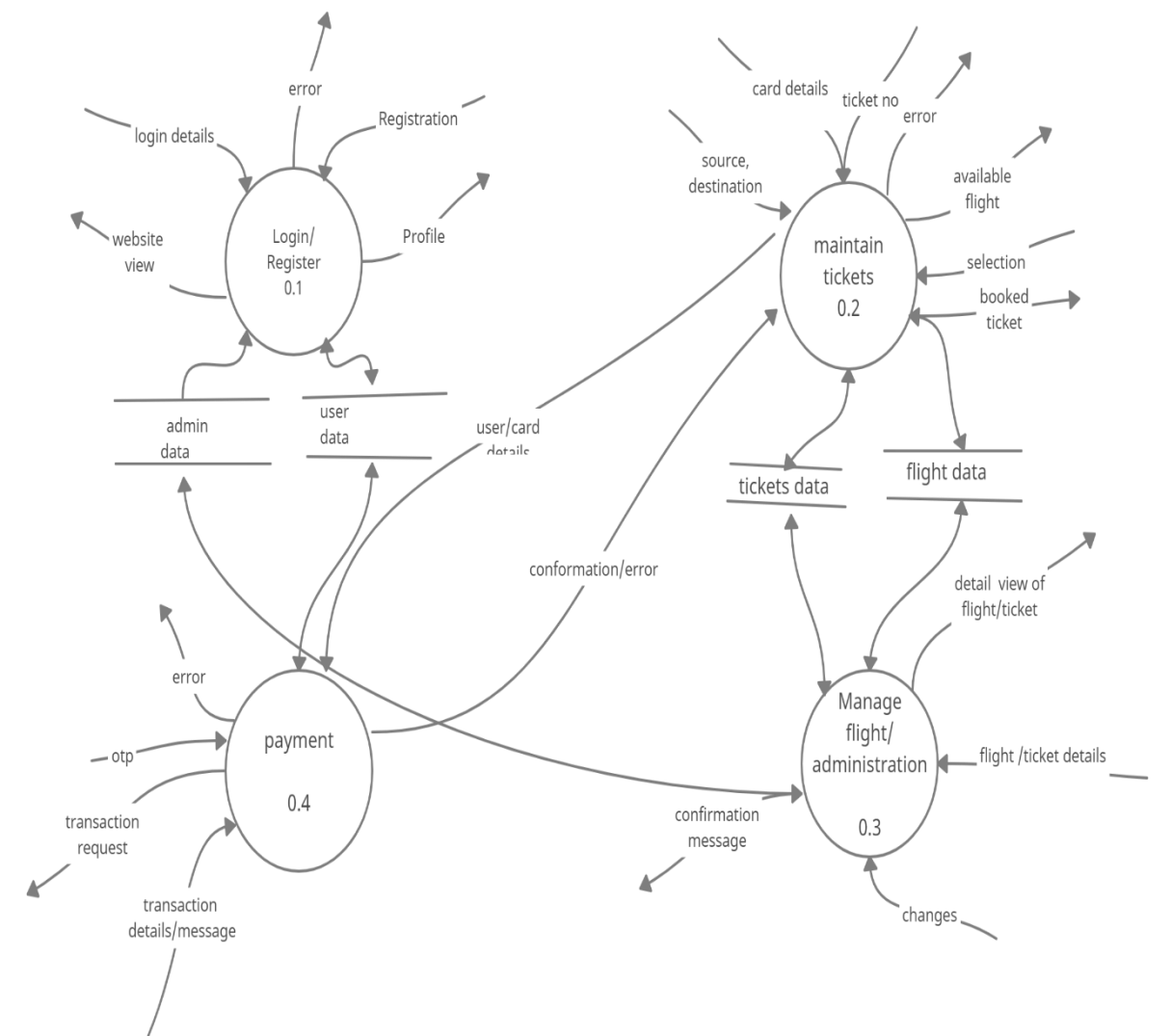# Activity Diagrm of Login/Register

Activity Diagrm of Book Ticket

5.Structure Chart :-

The structure chart contains the following components:

- **main** connects to: login/register (login or registration), search flight/book (search), payment (payment), administrator (management)
- **login/register** connects to: read verify from database (login credentials, view web), read and store data in database (register credentials)
- **search flight/book** connects to: get details from database (source, dest, time / flights details), update details in database (price / selected flight, seats)
- **payment** connects to: verify detail & store in user database (card details), request transaction to finance system (trans details)
- **administrator** connects to: search flights (flights credentials / flights details), update details in database (flight id, changes), update details in database (ticket credentials, changes)

5. DFD Model :-

Level 0

Level 1

website view

profile/
error

email, name,
mobile no

**register
user
0.1.1**

user data

**validation
write
0.1.2**

login data

**login
0.1.3**

email , password

user data

admin data

source , destination

available
flights

transaction
request

selection

**search
flights
0.2.1**

available
flights

**display and
select
0.2.2**

user details
and
card details

**read or
write card
details
0.3.1**

card
details

**make
payment
0.3.2**

successful
transaction
reciept

flights data

cancel ticket
no

select flight
and seat

error

confirmation payment details/
refund details

refund
request

card details

**book/cancel
tickets
0.2.3**

user
details

**refund
0.3.3**

booked ticket/
error/refund message

tickets data

**update
flights
details
0.4.1**

flight id or
source,dest

flights details

changes

**remove
ticket
0.4.2**

ticket no

deletion
confirm

Level 2

4. Implementation Details

4.1 Description of Modules :-

1.Account module

Basic information of user taken by system and stored in database. When user is try to login in system , system authenticate to user and then user entered in system.

2.Travel module

When user want to Book Tickets ,system will allow user To book ticket at their specific time and date with required source and destination.

6. Screenshots

1. Home page:
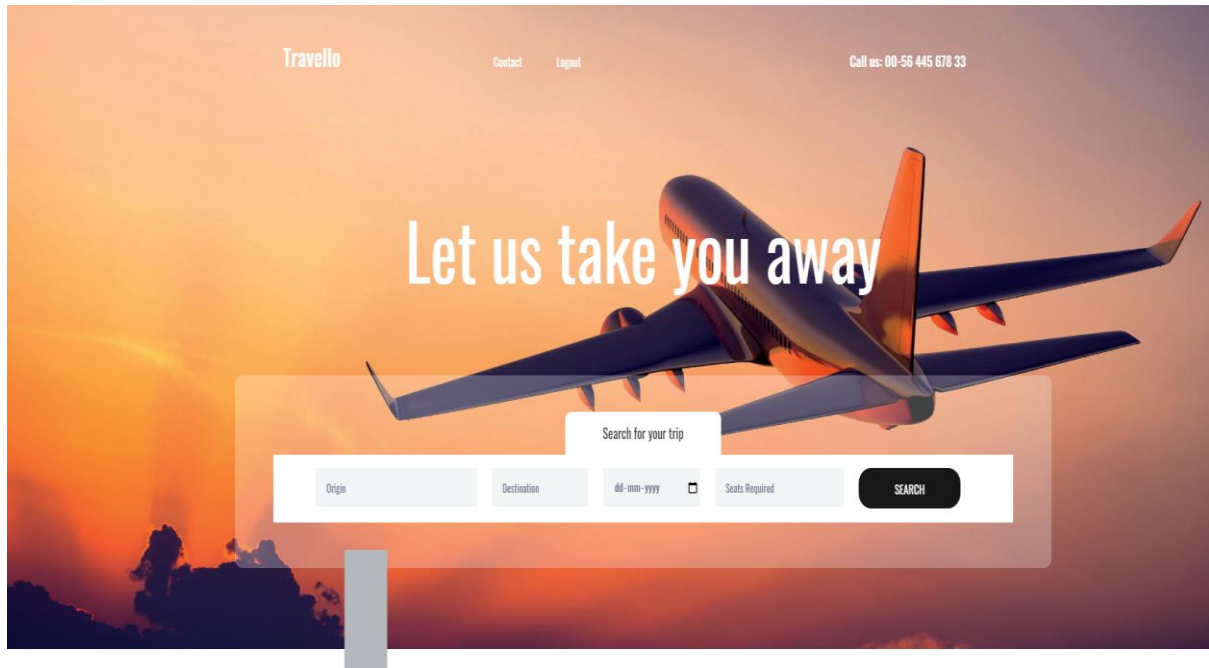
2. Registration page :-

3. Login page:-
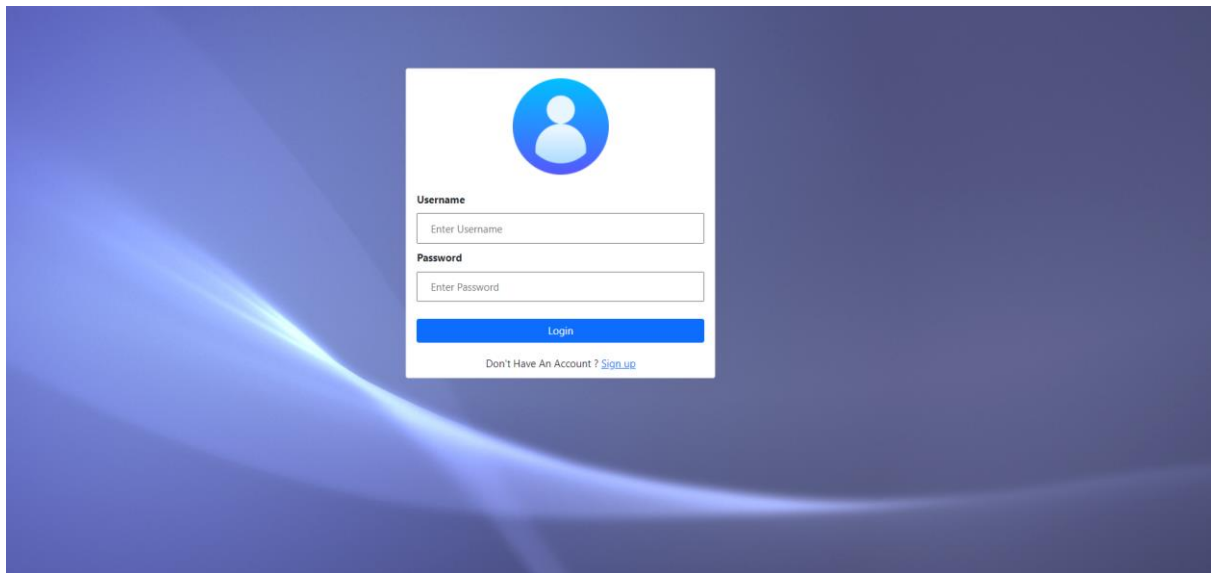
4. Available Flights

5.Payment

total amount   3000

Your payment details

Card Number

MM/YY   CVC

pay
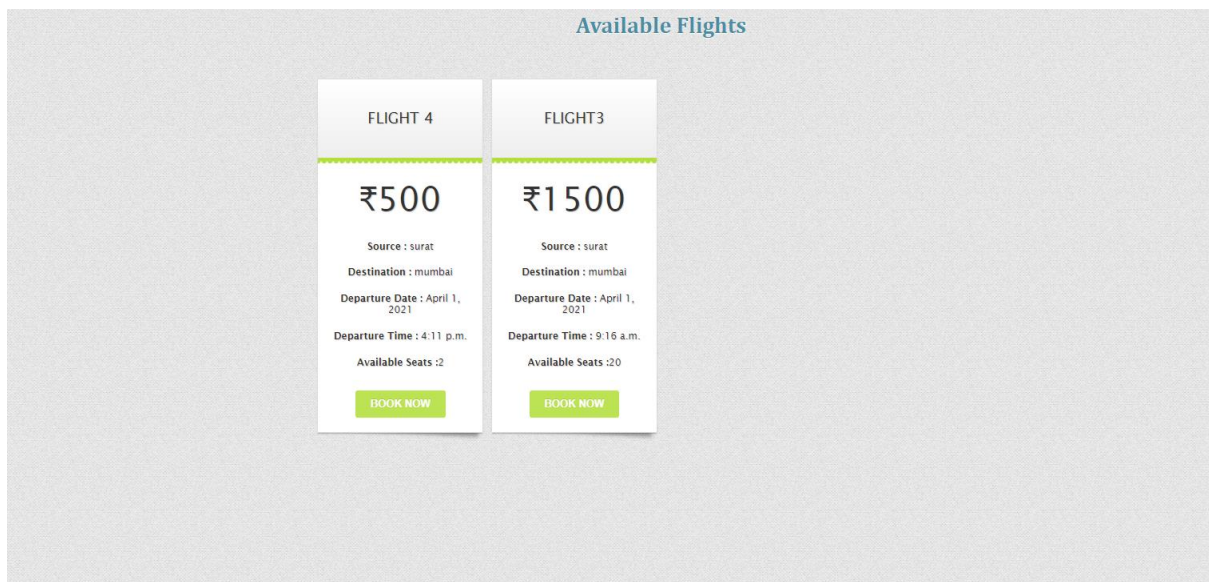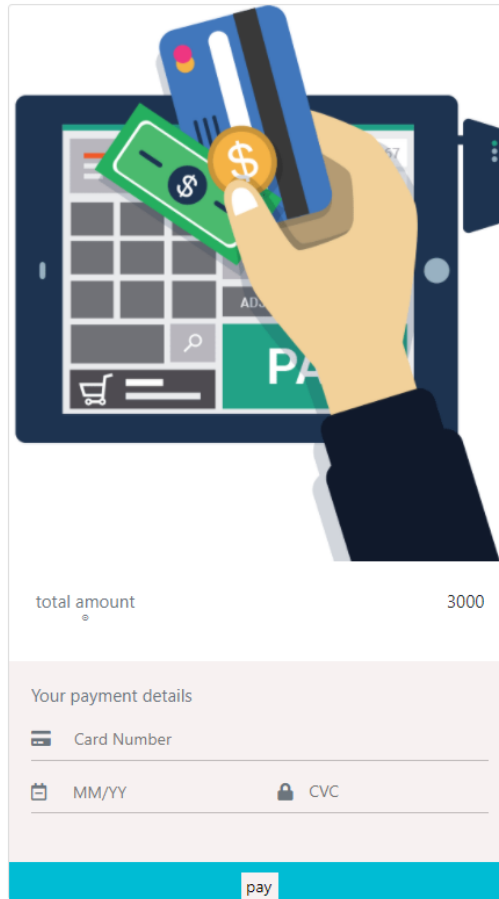
7. Conclusion

The Airline reservation system allow the user to register and login. After Login Airline reservation system allow user to search flights by entering source, destination, date and required-seats. Airline reservation system shows available flights as per user requirements. User can confirm his/her ticket(s) by providing necessary details. This software is working properly and meeting to all user requirements.

8. Limitation and future extension

Limitation:

Background processes can be handled more efficiently. Process load due to concurrent thread execution can be reduced. Defalut Django administration is used for Admin panel but it can be made as per website layouts.

Future Extension:

User interface will be improved to provide better interaction with system. Background processes and Thread execution will be efficient and process load will be reduced.

9. Bibliography

Websites :

https://stackoverflow.com

https://www.djangoproject.com/