

# FLYAIR

## An Airline Reservation System

Team Leader : Nareshkumar M. Sisodiya

Team Members: Arjun Lokhande

Binu Basil John

Anant Mathur

Khushboo Handa

# Outline

- Introduction
- Domain Analysis
- Project Implementation
- Tools Used
- Lesson Learnt

# Introduction

Who we are ?

- Fly Air - Airline Reservation System

What we do ?

- Prompt and Hassle Free Booking System
- User Friendly

# Domain Analysis

- Why we chose this domain ?
  - Personal experience.
  - Ease to comprehend.
- Study of the domain
  - Online website references.
  - Comparison of features to be provided.
- Analysis of the domain.
  - Technical implementation requirements.
  - We defined the various actors and their corresponding use cases.

# Team Goals

- Informal and friendly communication
- Time Management
- Respecting each other's point of view
- Appreciation and encouragement

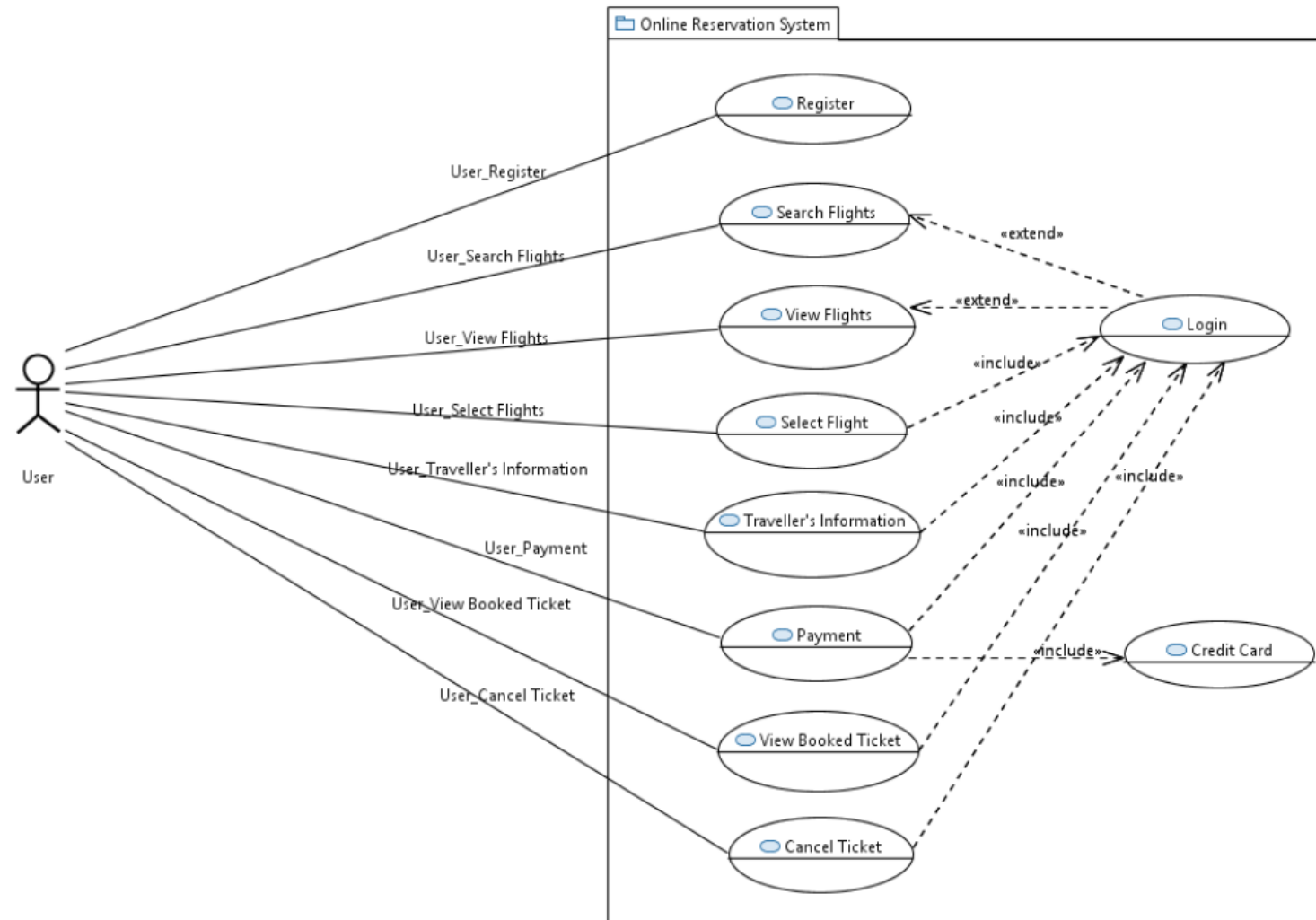
## **Meeting Schedule**

- Wednesdays: 18:00 - 21:00
- Fridays: 12:30 - 16:30

# Timeline

[illegible]

# Use Case Diagram



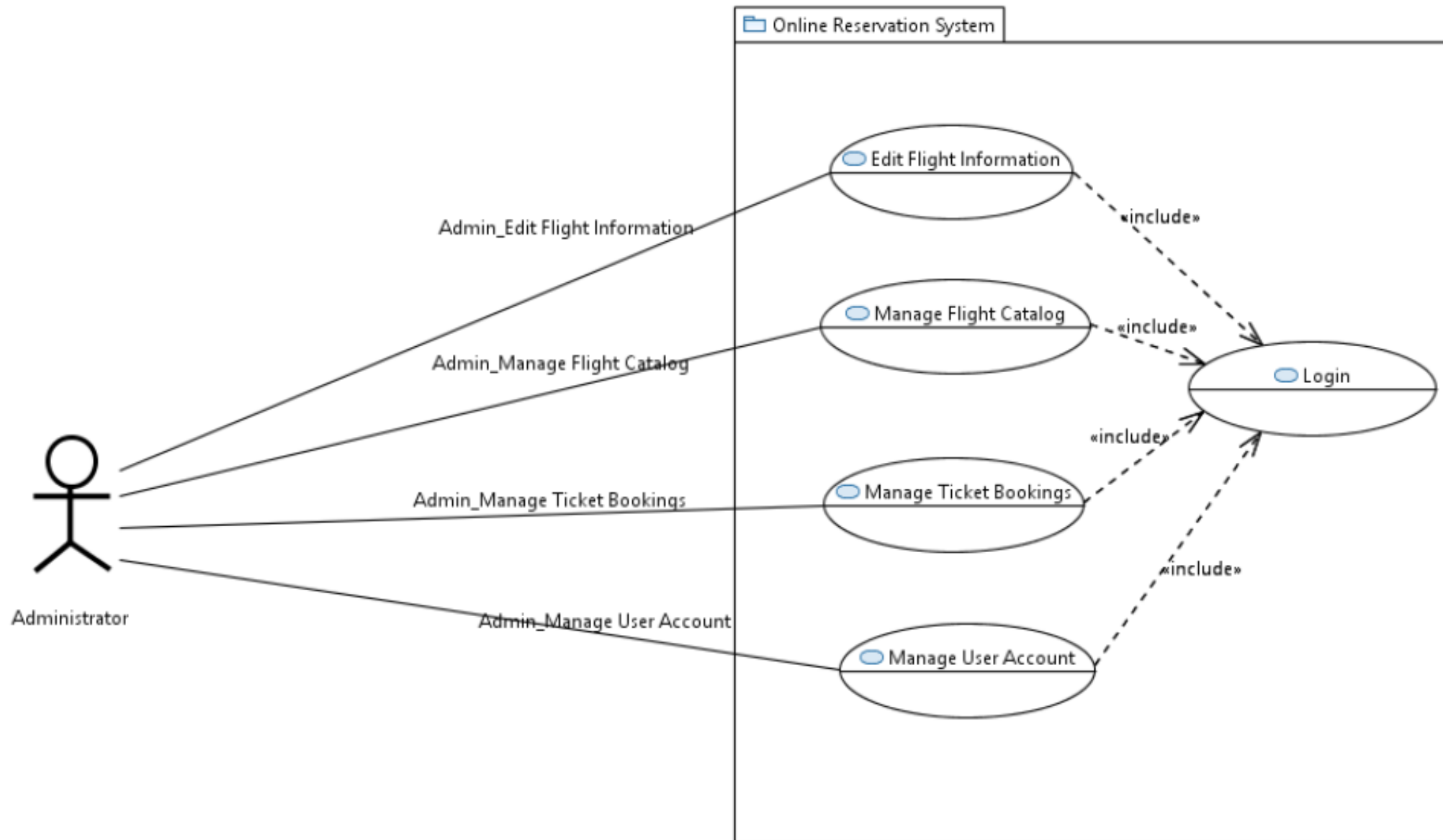
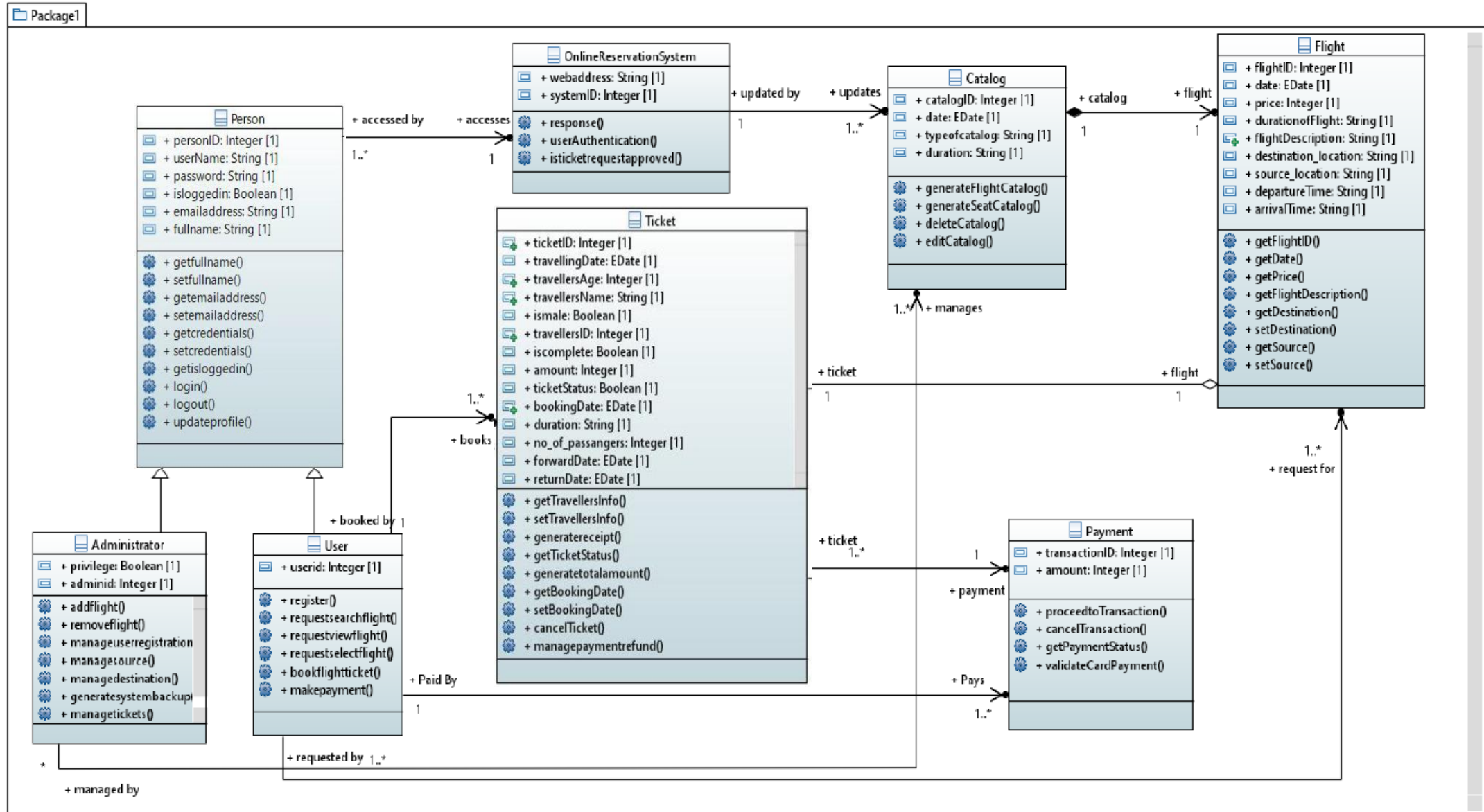




Table 2: Login Use Case

Use Case ID	UC-02
Use Case Name	Login
Description	The user logs in to the system using the username and password.
Primary Actor	User, Administrator
Pre-Conditions	Active account in the system
Normal Flow	<ol style="list-style-type: none"><li>1. The system display the login page.</li><li>2. The User enters the user information (username and password)</li><li>3. The system verifies the User information</li><li>4. The system displays the User homepage.</li></ol>
Alternative Flow	N/A

# Class Diagram



# OCL

- Each person who logs in to the system to book a ticket must have an unique ID :

Context : Person

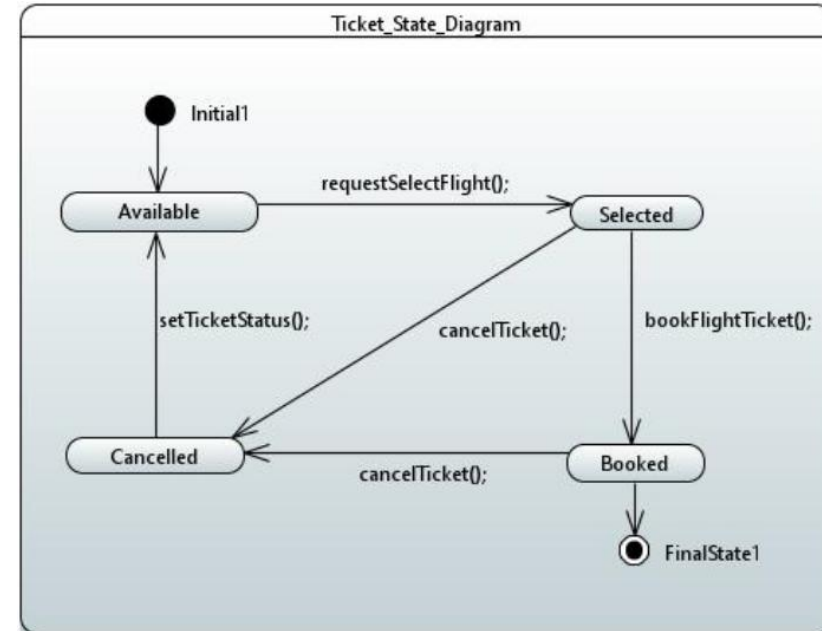
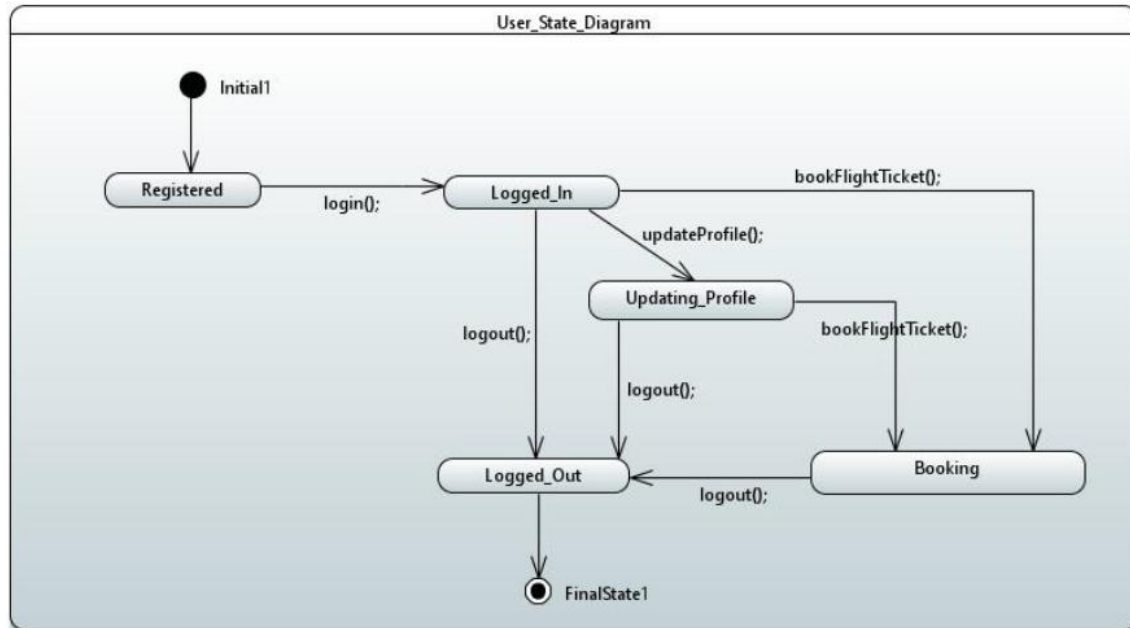
inv: self.allinstances()->forall(P1,P2:Person | P1 <> P2 implies  
P1.userID <> P2.userID);

- Each Ticket booked by the user for scheduled travel should have an unique ID :

Context : Ticket

inv: self.allinstances()->forall(T1,T2:Tickets | T1<>T2 implies  
T1.ticketID <> T2.ticketID);

# State Diagram



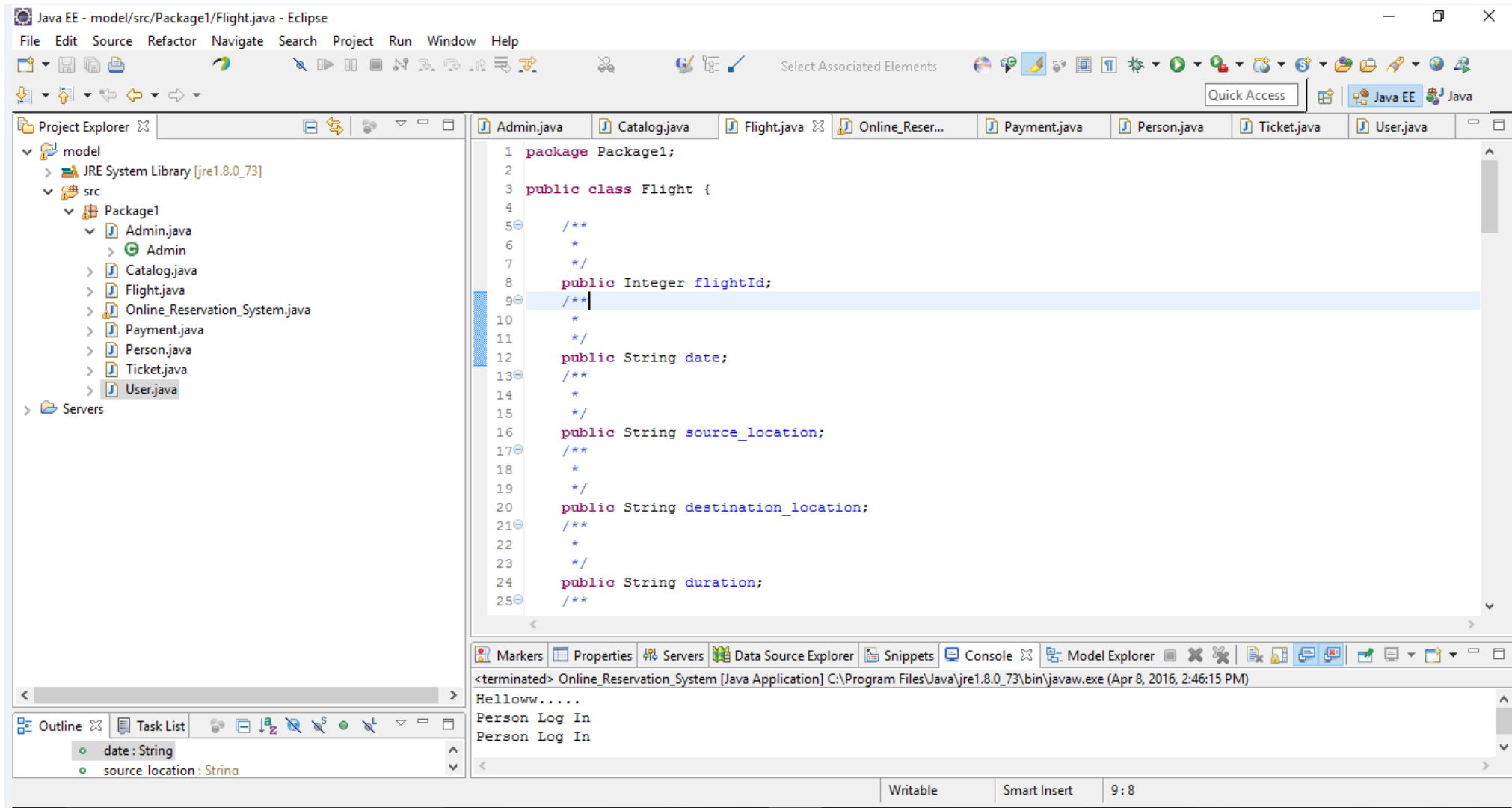
# Action Specification Language

```
if(authenticate())
{
    System.out.println("User Successfully Logged in");

    System.out.println("Please Select the operation:");
    System.out.println("\t\t\t1. Book Flight");
    System.out.println("\t\t\t2. Update User Profile");
    System.out.println("\t\t\t3. Logout");
    int op;
    op = sc.nextInt();

    switch (op) {
    case 1:
        bookFlight();
        break;
    case 2:
        updateProfile();
        break;
    case 3:
        System.out.println("User Logged Out Succesfully.");
        break;
    default:
        break;
    }
}
else
{
    System.out.println("Invalid Username or password");
}
```

# Code Generated from Class Diagram



# Modified Code

The screenshot displays an IDE interface with the following components:

- Menu Bar:** File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help.
- Toolbar:** Includes icons for file operations (save, open, delete), navigation (back, forward), and development tools (run, debug, test).
- Project Explorer (Left):** Shows a project structure with folders like 'AirlineReservation', 'JAX-WS Web Services', 'Java Resources', 'src', 'Libraries', 'JavaScript Resources', 'WebContent', 'Servers', and 'Test\_Pro'. The 'src' folder is expanded, showing a package hierarchy: 'com.arjun.auction.commons', 'com.arjun.auction.dao', 'com.arjun.auction.dbutils', 'com.arjun.auction.exception', 'com.arjun.auction.pojo', 'com.arjun.auction.service', and 'com.arjun.auction.servlet'. The 'com.arjun.auction.pojo' package is selected.
- Navigator (Top):** Shows the current file being edited: 'UserDaoImpl.java'.
- Code Editor (Center):** Displays the source code for 'UserDaoImpl.java'. The code is as follows:

```
1 package com.arjun.auction.pojo;
2
3 public class Person {
4     private String id;
5     private String fullname;
6     private String email;
7     private String contactno;
8     private String role;
9     public String getId() {
10         return id;
11     }
12     public void setId(String id) {
13         this.id = id;
14     }
15     public String getFullname() {
16         return fullname;
17     }
18     public void setFullname(String fullname) {
19         this.fullname = fullname;
20     }
21     public String getEmail() {
22         return email;
23     }
24     public void setEmail(String email) {
25         this.email = email;
26     }
27     public String getContactno() {
28         return contactno;
29     }
30 }
```
- Bottom Panel:** Contains tabs for 'Markers', 'Properties', 'Servers', 'Data Source Explorer', 'Snippets', 'Problems', 'Console', and 'Model Explorer'. The 'Servers' tab is active, showing 'Tomcat v7.0 Server at localhost [Stopped, Synchronized]'.
- Status Bar (Bottom):** Shows 'Writable', 'Smart Insert', and '1 : 1'.

# Search Flight:

localhost:8080/AirlineRese × Login × Search Flight ×

localhost:8080/AirlineReservation/

Apps ★ Bookmarks folder hadoop folder concordia Google Translate Top 10 Algorithms for folder Spark,cassandra Tech Experiences COMP 6651 - Algorith folder project delta folder interview WhatsApp Web

Login, New user ? Register.

SEARCH FLIGHT

FlyAir

Search Flight

From\*

Montreal ▾

To\*

Toronto ▾

Select Travel Date\*

09/04/2016

Total traveller\*

1 ▾

Search

Developed by Fly Air & Team



# Login:

Login

localhost:8080/AirlineReservation/login.jsp

Apps ★ Bookmarks folders hadoop concordia Google Translate Top 10 Algorithms for Spark,cassandra Tech Experiences COMP 6651 - Algorithm project delta interview WhatsApp Web

Google Chrome isn't your default browser. Set as default Don't ask again

Login, New user ? Register.

SEARCH FLIGHT

FlyAir

Login

Username

Arjun123

Password

\*\*\*\*\*

New user? click for sign up

login

Developed by Fly Air & Team

Search the web and Windows

2:08 ENG US 6:37 PM 2016-04-08

# Ticket Booking:

Flights

localhost:8080/AirlineReservation/flight?pagename=book\_flight&totaltraveller=1&id=22

Apps Bookmarks hadoop concordia Google Translate Top 10 Algorithms for Spark,cassandra Tech Experiences COMP 6651 - Algorithm project delta interview WhatsApp Web

## Book Flight

### Flight Details

From :	To :	Travel Date :	Time :	Duration :
Toronto	Montreal	2016-04-09	14:45	2

Complete your transaction within -> 03:46

### Passenger Details

Traveller 1 Name		Arjun Lokhande	
Contact No	1234543	Email	ar0j.l@gmail.com
Total Fare : 232			

### Payment Details

Card Number	121312
CVV	546
Name On Card	Arjun Lokhande
<input type="button" value="Reset"/> <input type="button" value="Book"/>	

Search the web and Windows

ENG US 6:44 PM 2016-04-08

# Ticket Confirmation:

Success

localhost:8080/AirlineReservation/flight?pagename=book\_ticket&id=22

Apps Bookmarks hadoop concordia Google Translate Top 10 Algorithms for Spark,cassandra Tech Experiences COMP 6651 - Algorith project delta interview WhatsApp Web

Welcome Arjun S Lokhande, Logout

SEARCH FLIGHT BOOKING HISTORY

FlyAir

Ticket Details

Ticket booked successfully.

Flight Details

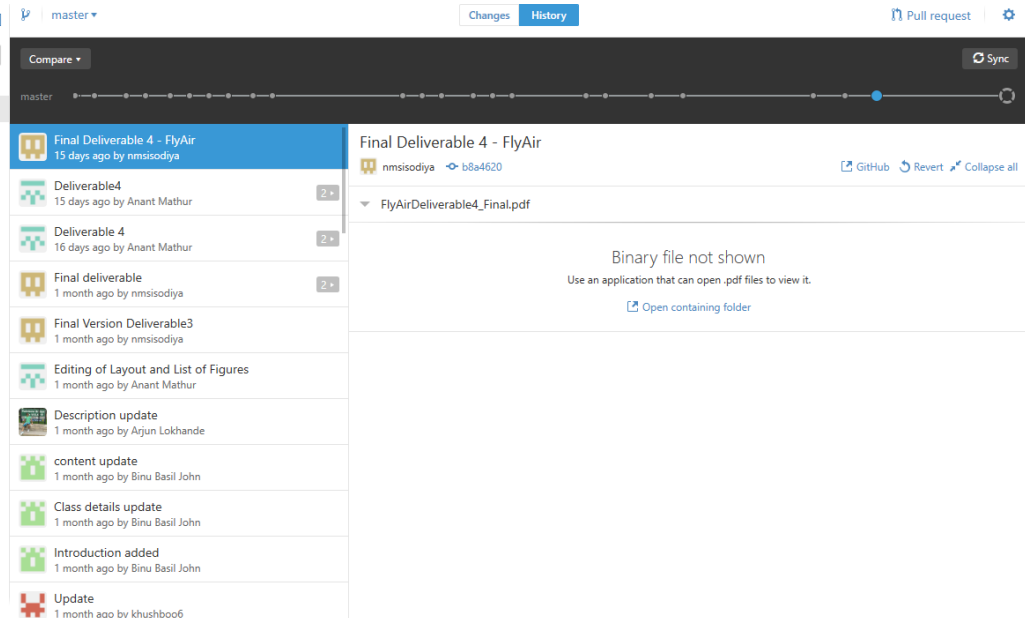
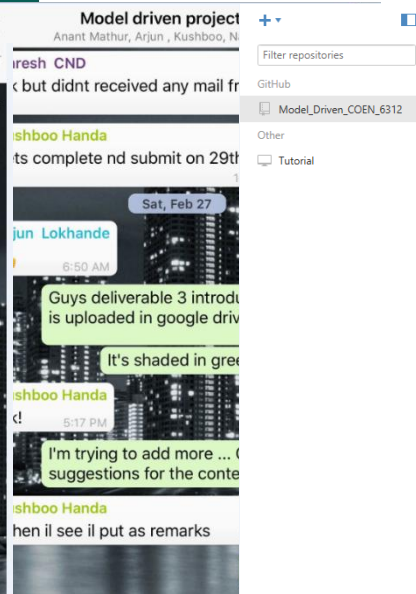
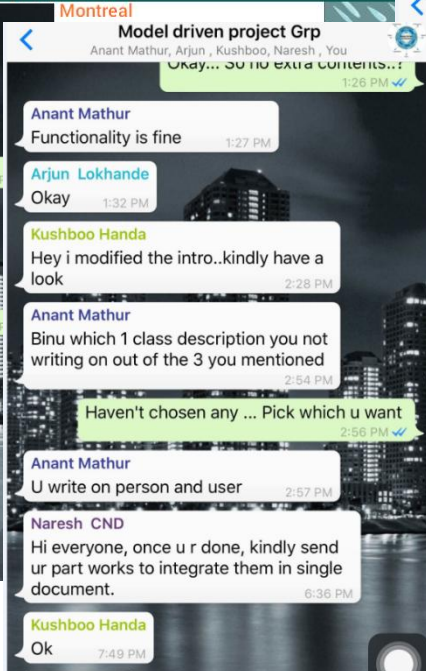
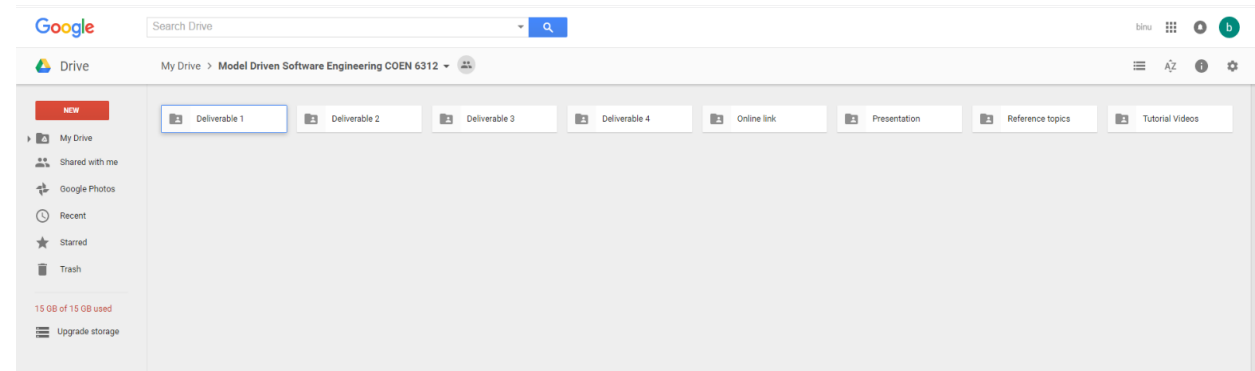
From :	To :	Travel Date :	Time :	Duration :
Toronto	Montreal	2016-04-09	14:45	2

Passenger Details

Traveller 1 Name		Arjun Lokhande	
Contact No	1234543	Email	ar0j.l@gmail.com
Total Fare : 232			

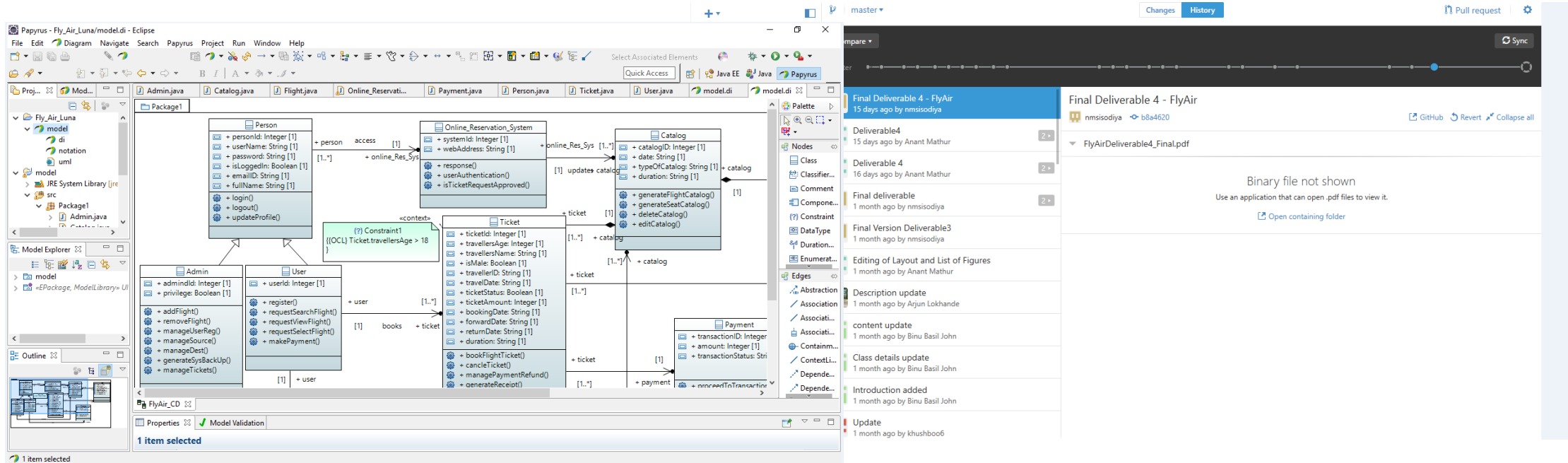
# Collaboration Tools

- GITHUB
- Whatsapp Application
- Google Drive



# Tools Used

- Development Environment: Eclipse, Papyrus
- Version control system: Github



# Lessons Learnt

- Model Driven Software Engineering Approach
- Model Diagrams in Papyrus
- Code Generation by Model Driven Approach
- Importance of Software Practices
- Team Work and Time Management
- Github

# Difficulties Encountered

- Documentation in Github –Unable to explore the full features
- Using papyrus standalone, later using papyrus as a plugin on eclipse as demonstrated and suggested by the TA

Thank You !!