



Model Driven Software Engineering

(COEN 6312)

Project Deliverable 2

Submitted to

Dr. Abdelwahab Hamou-Lhadj

Submitted by

Nareshkumar M. Sisodiya 27650817

Arjun Lokhande 27411111

Binu Basil John 27421753

Anant Mathur 27323670

Khushboo Handa 27323794

February 8, 2016

Contents

List of Figures	ii
List of Tables	iii
1. Introduction.....	1
2. Functional Requirements	2
2.1 Use Case Diagrams	2
2.2 Use Case Descriptions	3
2.2.1 User.....	3
2.2.2 Administrator	10
3. Non Functional Requirements	13
Bibliography	14

List of Figures

Figure 1: User Use Case Diagram.....	2
Figure 2: Administrator Use Case Diagram.....	3

List of Tables

Table 1: Register Use Case	3
Table 2: Login Use Case.....	4
Table 3: Search Flights Use Case	5
Table 4: View Flights Use Case.....	6
Table 5: Select Flight Use Case	6
Table 6: Traveller's Information Use Case	7
Table 7: Payment Use Case	8
Table 8: View Booked Ticket Use Case	8
Table 9: Cancel Ticket Use Case	9
Table 10: Edit Flight Information Use Case	10
Table 11: Manage Flight Catalog Use Case.....	11
Table 12: Manage Ticket Booking Use Case.....	11
Table 13: Manage User Account Use Case	12

1. Introduction

Fly Air offers an online reservation for its flights between various pre-listed destinations within Canada at affordable costs. Our main aim involves facilitating an intuitive design for the reservation system which enables the user to book flights with minimum delay and makes the whole reservation experience simple and hassle free.

Fly Air online reservation system enables the user to search and view through the catalog of the available flights of Fly Air between various destinations without the necessity of registering oneself into the system. This helps the user to manage time efficiently during last minute bookings, in case the desired flight is not available or is fully booked. However, if user ascertains the flight to be booked, system requests the user to login or register oneself to proceed with the booking. Choice of individual or group booking is feasible with the requisite to provide valid information of each traveller. Furthermore, the user has the advantage to create multiple bookings, for instance the user requires to travel multiple destinations on consecutive days or wishes to book the return tickets at the same time. This practice caters maximum efficiency in booking and it's done to offer utmost priority to the convenience of the user. The payment can be done through credit or debit card by filling out relevant banking information to complete the transaction. The logged-in user has the privilege to access the details of recent and previous bookings and can get quick refunds in case of ticket cancellation. For any further assistance the user can contact the administrator through the help option.

In the back end the administrator plays a vital role in management of the system. Once logged into to the system the administrator has the authority to edit, manage and publish the information related to flights, tickets and user accounts. It is the sole responsibility of administrator to confirm that the information published in the system is accurate and relevant. This ensures that the user is not misguided with trivial errors and end up wasting their invaluable time and money. In short we expect to accomplish a refined reservation system for Fly Air that reflects their dignity and grace in business to achieve trust and satisfaction of every single customer.

2. Functional Requirements

The functional requirement defines the actors who use this system while supporting the targeted business processes, as well as the use cases this system provides to those actors. Each use cases has a specific task which enables the user to accomplish desired results out of the system. Detailed use case diagram and description of various use cases has been provided below.

2.1 Use Case Diagrams

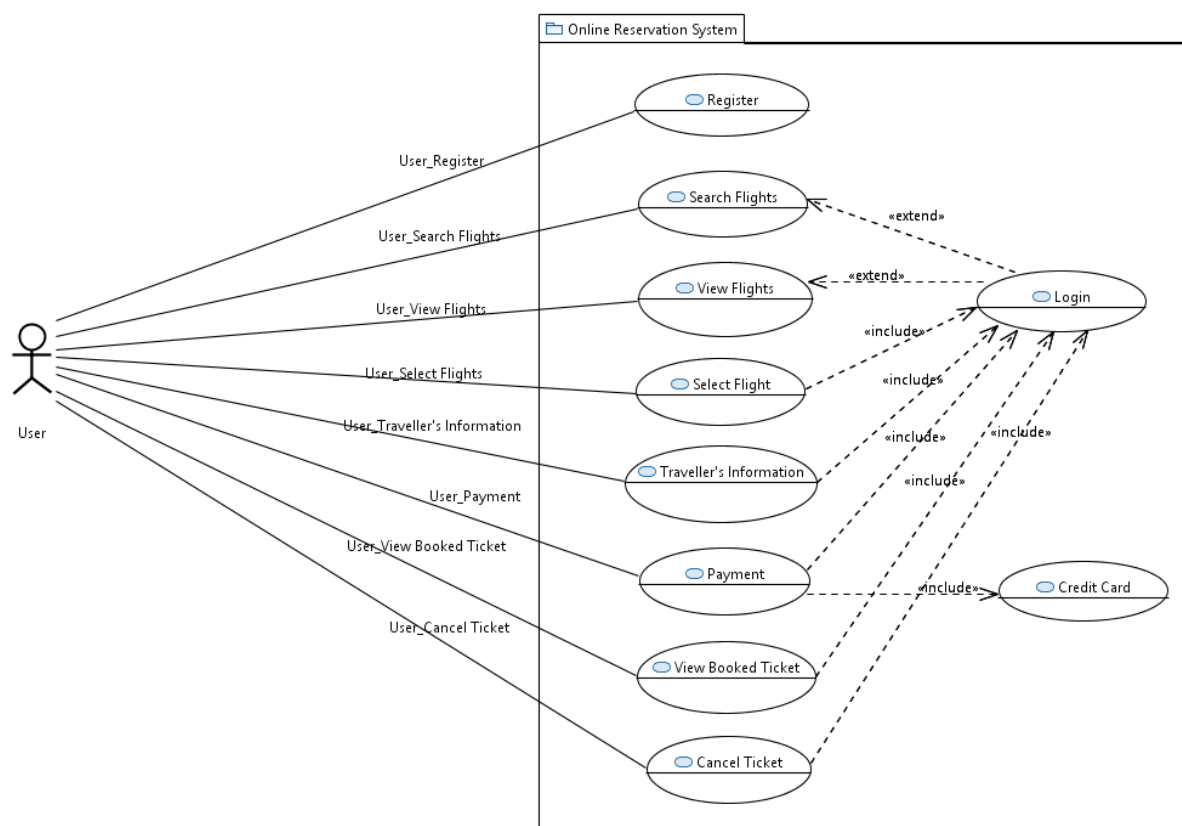


Figure 1: User Use Case Diagram

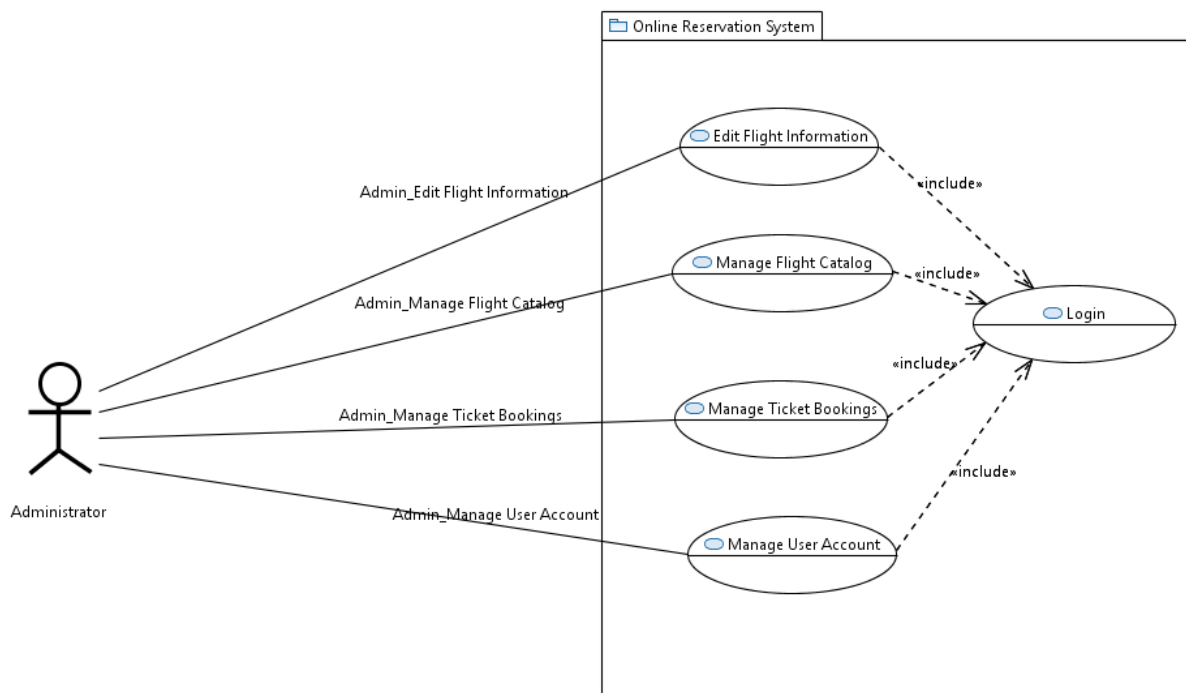


Figure 2: Administrator Use Case Diagram

2.2 Use Case Descriptions

2.2.1 User

1. Register

Table 1: Register Use Case

Use Case ID	UC-01
Use Case Name	Register
Description	Registration of a new user to generate a new account in the system database
Primary Actor	User
Pre-Conditions	User does not have an account.
Normal Flow	<ol style="list-style-type: none"> 1. The user clicks on the link for registration on the webpage 2. The system displays the registration page. 3. The user inputs all of his necessary personal information. 4. The system verifies the entered information.

	<p>5. The system updates the database.</p> <p>6. The confirmation message for the new account is displayed to the user.</p>
Alternative Flow	N/A
Exceptions	<p>In step4: The user registration form is missing mandatory information or has incorrect characters or duplicate email ID.</p> <ol style="list-style-type: none"> 1. The system highlights the sections having error. 2. The system notifies with an error message that alerts the user about the error. 3. The system prompt the user to fix the wrong information or to add the missing information. <p>In step 5 : The system detects a problem with the database connection</p> <ol style="list-style-type: none"> 1. The system informs the user of the problem with connection 2. The user registration fails.
Priority	High

2. Login

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"> 1. The system display the login page. 2. The User enters the user information (username and password) 3. The system verifies the User information 4. The system displays the User homepage.
Alternative Flow	N/A

Exceptions	<p>Step 1: The User enters a wrong login information.</p> <ol style="list-style-type: none"> 1. The system shows an error message displaying invalid username and/or password. 2. The system prompts the User to re-enter the login information.
Includes	N/A
Priority	High

3. Search Flights

Table 3: Search Flights Use Case

Use Case ID	UC-03
Use Case Name	Search Flights
Description	Searches the available flights as entered by the User
Primary Actor	User and Administrator
Pre-Conditions	User has logged into the system
Normal Flow	<ol style="list-style-type: none"> 1. The user navigates to the Search Flights section. 2. The User enters the Travel information. 3. The system retrieves all the relevant flight information from the database.
Alternative Flow	The User can search the flights without Login
Exceptions	<p>In step 2:</p> <ol style="list-style-type: none"> 1. The user enters invalid Travel Information 2. The systems displays the error message to the user. <p>In step 3: There is no flight available at the requested time.</p>
Includes	N/A
Priority	High

4. View Flights

Table 4: View Flights Use Case

Use Case ID	UC-04
Use Case Name	View Flights
Description	This use case describes that the user will view list of available flights
Primary Actor	User
Pre-Conditions	1. User searches a Flight 2. Flight exists in the system
Normal Flow	1. The flight information has been retrieved and shown to the user. 2. The user navigates through the list of flights. 3. The user selects the desired flight.
Alternative Flow	The user can view of the list of Flights page without login.
Exceptions	1. The system detects a problem with retrieving the desired flight information from the database. 2. An error message is displayed to the user.
Includes	N/A
Priority	High

5. Select Flight

Table 5: Select Flight Use Case

Use Case ID	UC-05
Use Case Name	Select Flight
Description	The User can select a Flight from list of all available flights.
Primary Actor	User
Pre-Conditions	1. User has logged into the system 2. User has viewed the list of flights
Normal Flow	1. The User navigates through the flight selection section

	2. The system retrieves the flight information from the database and checks for any conflicts. 3. The system updates the database. 4. The system displays the confirmed selection.
Alternative Flow	N/A
Exceptions	1. The user is unable to select the desired flight due to technical issues. 2. The Database is inaccessible.
Includes	The User must Login
Priority	High

6. Traveller's Information

Table 6: Traveller's Information Use Case

Use Case ID	UC-06
Use Case Name	Enter Traveller's Information
Description	The User must enter the traveller's information
Primary Actor	User
Pre-Conditions	1. User has logged into the system 2. User has viewed and selected the flight
Normal Flow	1. The User has selected the flight and entered traveller's information 2. The system updates the database. 3. The system displays the information and prompts to confirm to proceed further .
Alternative Flow	N/A
Exceptions	1. The Database is inaccessible. 2. The user has not filled all the mandatory fields.. 3. An error message is displayed to the user.
Includes	The User must Login
Priority	High

7. Payment

Table 7: Payment Use Case

Use Case ID	UC-07
Use Case Name	Payment
Description	The user makes payment for the selected tickets
Primary Actor	User
Pre-Conditions	<ol style="list-style-type: none">1. User has logged into the system2. User has viewed the list of flights3. User has selected the flight and entered the travelers' information.4. The payment method has been selected by user.
Normal Flow	<ol style="list-style-type: none">1. The user makes flight selection and enter travellers' information.2. The user selects the payment method and enter the valid information for the payment3. The system validates the user credentials for the payment.4. The system informs the user of the successful payment.
Alternative Flow	N/A
Exceptions	<ol style="list-style-type: none">1. Unsuccessful payment due to insufficient balance.2. Payment unsuccessful due to wrong payment or card information.3. Communication failure between system and bank server.4. User is informed about the unsuccessful transaction and prompted for another payment.
Includes	The User must Login
Priority	High

8. View Booked Ticket

Table 8: View Booked Ticket Use Case

Use Case ID	UC-08
Use Case Name	View Booked Ticket

Description	The User can view the tickets
Primary Actor	User
Pre-Conditions	1. User has logged into the system 2. User has booked a ticket.
Normal Flow	1. The User views the ticket he/she has booked 2. The system displays the information and prompts to confirm to proceed further .
Alternative Flow	N/A
Exceptions	1. The booked ticket has been deleted from the history. 2. The Database is inaccessible.
Includes	The User must Login
Priority	High

9. Cancel Ticket

Table 9: Cancel Ticket Use Case

Use Case ID	UC-09
Use Case Name	Cancel Ticket
Description	The User can cancel ticket and get refund
Primary Actor	User
Pre-Conditions	1. User has logged into the system 2. User has booked a ticket.
Normal Flow	1. The User selects booked the ticket he/she wants to cancel 2. The system displays the information and prompts to confirm to proceed further.
Alternative Flow	N/A

Exceptions	1.The Database is inaccessible. 2.The user could not cancel ticket 3. An error message is displayed to the user.
Includes	The User must Login
Priority	High

2.2.2 Administrator

10. Edit Flight Information

Table 10: Edit Flight Information Use Case

Use Case ID	UC-10
Use Case Name	Edit Flight Information
Description	The Admin can edit the flight information if some changes are made by the airlines management.
Primary Actor	Administrator
Pre-Conditions	The Admin is logged into the system
Post-Conditions	The Admin has successfully edited the flight information
Normal Flow	1.The Administrator logs into the system. 2.The Administrator can edit the existing flight information.
Alternative Flow	N/A
Exceptions:	The Administrator forgets the password of editing the information.
Includes	Administrator Login
Priority	High

11. Manage Flight Catalog

Table 11: Manage Flight Catalog Use Case

Use Case ID	UC-11
Use Case Name	Manage Flight Schedule
Description	The Admin manages the flight catalog and edits the flight schedule if necessary
Primary Actor	Administrator
Pre-Conditions	The Admin is logged into the system
Post-Conditions	The Admin has updated the flight catalog.
Normal Flow	1. The Administrator logs into the system. 2. The Administrator manages the flight catalog.
Alternative Flow	N/A
Exceptions	The Administrator forgets the secondary password for editing the catalog.
Includes	Administrator Login
Priority	High

12. Manage Ticket Booking

Table 12: Manage Ticket Booking Use Case

Use Case ID	UC- 12
Use Case Name	Manage Ticket Bookings
Description	The Admin manages the ticket bookings of the scheduled flights.
Primary Actor	Administrator

Pre-Conditions	The Admin is logged into the system
Post-Conditions	The Admin is able to manage the bookings.
Normal Flow	1. The Administrator logs into the system. 2. The Administrator is able to manage the tickets bookings.
Alternative Flow	N/A
Exceptions	The Administrator forgets the password for editing the information.
Includes	Administrator Login
Priority	High

13. Manage User Account

Table 13: Manage User Account Use Case

Use Case ID	UC-13
Use Case Name	Manage User Accounts
Description	The Admin can edit the information if some changes are made by the airlines management.
Primary Actor	Administrator
Pre-Conditions	The Admin is logged into the system
Normal Flow	1. The Administrator logs into the system. 2. The Administrator can edit the account information of the user if the user requests.
Alternative Flow	N/A
Exceptions	The Administrator forgets the password of editing the information.
Includes	Administrator login
Priority	High

3. Non Functional Requirements

- **Performance:** Overall response time of the system should be as small as possible. System will be required to provide list of flights in response of flight search request. The system should respond in acceptable time frame. Accepting payment, validating transaction from the bank and confirming the ticket booking should be done very quickly. Also the process to cancel tickets and get refunds should be instantaneous and convenient.
- **Availability:** The system should be available 24 x 7. The supporting systems and software should be stable and operational all the time.
- **Reliability:** The system should not crash frequently. System should save data at frequent intervals so that in case of failure, system can restore the user data and ongoing sessions.
- **Maintainability:** The online system must be kept up to date in terms of flight status and flight availability.
- **Serviceability:** The reservation system must be able to service the demands of all the users. Number of people who can access the system at a time should be utmost.
- **Security:** System must handle unauthorized access and notify user immediately. Password must have special characters and complex combinations. User should be notified if he/she selects weak password. User should be automatically logged off in case of more than 10 minutes of inactivity. The payment process must happen over a secured session.
- **Privacy:** The user/traveller information must be protected from unwanted usage.
- **Usability:** Users must be able to search and view flights without logging / registering.

Bibliography

1. Marco Brambilla, J. C. *Model-Driven Software Engineering in Practice*. Morgan & Claypool .
2. *Papyrus User Guide*. (n.d.). Retrieved from https://wiki.eclipse.org/Papyrus_User_Guide
3. Thomas Stahl, M. V. *Model-Driven Software Development: Technology, Engineering, Management*. Wiley.
4. *UML Use Case Diagrams: Guidelines*. (n.d.). Retrieved from <https://msdn.microsoft.com/en-us/library/dd409432.aspx>