

# Deliverable #2 (Use Case Diagram)

---

**Project Topic: Online Flight Reservation System**

**Team: 6 Ideas**

**Team Members:**

<b>Family Name</b>	<b>First Name</b>	<b>Student ID</b>
Afrasiabi	Seyedeh Nazanin	40059181
Afrasiabi	Seyedeh Negar	40057810
Ahmed	Owais	40040018
Promwongsa	Nattakorn	40051020
Rasouli	Farinaz	40047293
Azadiabad	Siamak	26592856

February 11, 2018

Model Driven Software Engineering Course

**Table of Contents**

1. System Description ..... 3

2. Use Case Diagram ..... 5

3. Use Case Description ..... 6

4. Functional Requirements..... 13

**Table of Figures**

Figure 1: Online flight reservation system Use Case diagram ..... 5

# 1. System Description

## 1.1 Textual Description

Online flight reservation system is to facilitate customers to search and book flights. Moreover, it helps system managers to manage the system and keep the flights available to the customers, or generate needed reports. The main capabilities of this system from customers perspective are flight enquiry, user registration, flight reservation and cancelation. For the flight enquiry, customers can search flights for a given departure and destination airport and date, among available flights in the system which the result can be several different flights in terms of departure time. The travelers might not be necessary to register in order to search and select flights, but to make a payment to finalize the reservation, prior registration and logging in is mandatory. After the payment, the customer will be able to view and download the ticket. Also, a customer can cancel his/her own flights whenever he/she wants to do, before the deadline of flight cancelation, if he/she is logged in to the system.

On the other hand, this system allows managers to manage the system. They can modify the flight schedules such as updating the existing flights (price and time), add new ones or delete flights before the flights are booked by any customers. The managers are able to generate reports about flight schedules or the customer orders.

## 1.2 Actors

- Customers/Travelers
- Managers

## 1.3 Goals

- Allow Customers/Travelers to book a flight
- Allow managers to manage a flight system

## 1.4 Trigger

- Customers/Travelers or Managers access the system.

## 1.5 Precondition in Customers/Travelers' perspective

- Flights should be available on the system

## 1.6 Failed conclusion in Customers/Travelers' perspective

- Flights are available on the system
- Flights needed by customers are not available on the system
- Logging in fails
- Proof of a payment fails

## 1.7 Steps of Execution for booking a flight

1. Customers access to the system

- 1A. Flights are not available on the system

2. Flight are available on the system

3. Customers search for a desired flight by filling flight information (departure airport, destination airport and date)

- 3A. The desired flight does not exist

4. The desired flight exists

5. Customers select the flight (add companion, select a flight class and a seat, add insurance and increase a luggage size)

6. Customers confirm the flight reservation

7. Customers log in.

- 7A. Logging in fails

8. Logging in is successful

9. Customers make a payment

- 9A. Proof of the payment fails

10. Proof of the payment is successful

11. Customers receive the ticket

#### 1.8 Steps of Execution for cancelling a flight

1. Customers access to the system

2. Customers log in

- 2A. Logging in fails

3. Logging in is successful

4. Customers cancel the flight

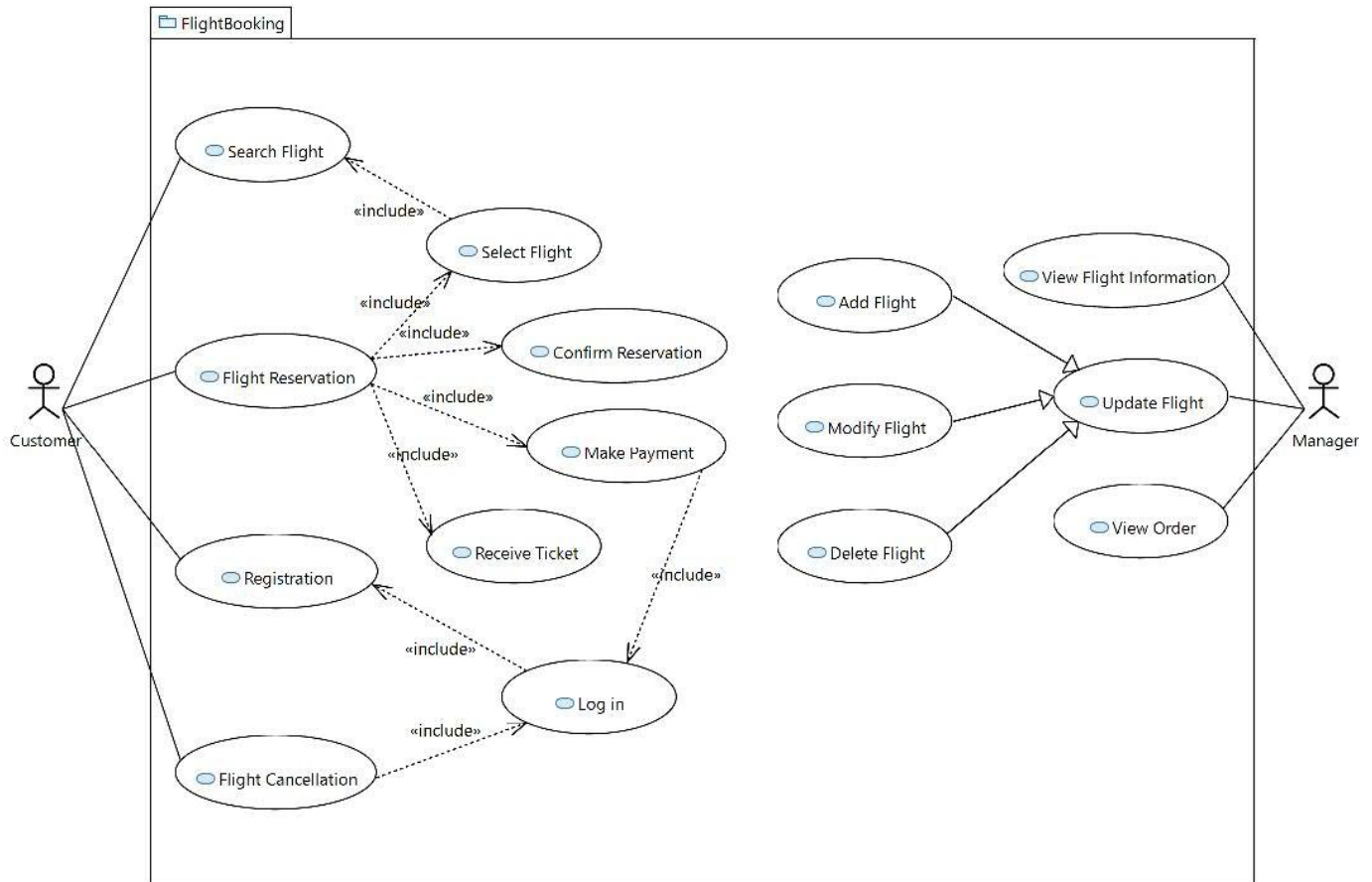
#### 1.9 Steps of Execution for viewing flight information, updating flights and viewing customers' orders

1. Managers access to the system

2. Managers view flight information, update flight or view customers' orders

## 2. Use Case Diagram

Following diagram depicts online flight reservation system Use Case diagram based on system description in the previous section:



**Figure 1: Online flight reservation system Use Case diagram**

### 3. Use Case Description

#### 3.1 UC-01: Search Flight

<b>Use-Case ID</b>	UC-01		
<b>Use-Case Name</b>	Search Flight		
<b>Created By</b>	6 Ideas	<b>Last Updated By</b>	6 Ideas
<b>Date Created</b>	Feb. 08 2018	<b>Date Last Updated</b>	Feb. 11 2018

<b>Actors</b>	Customer
<b>Description</b>	The customer search flight based on his/her need
<b>Preconditions</b>	NA
<b>Post conditions</b>	NA
<b>Normal Flow</b>	<ul style="list-style-type: none"><li>• Choose one round, round-trip.</li><li>• Fill in departure and arrival date.</li><li>• Select departure arrival airport.</li></ul>
<b>Alternative Flow</b>	The customer can sort the flight by time, price and promotion.
<b>Exceptions Flow</b>	The previous day is not accepted.
<b>Includes</b>	UC-02 and UC-03

#### 3.2 UC-02: Flight Reservation

<b>Use-Case ID</b>	UC-02		
<b>Use-Case Name</b>	Flight Reservation		
<b>Created By</b>	6 Ideas	<b>Last Updated By</b>	6 Ideas
<b>Date Created</b>	Feb. 08 2018	<b>Date Last Updated</b>	Feb. 11 2018

<b>Actors</b>	Customer
<b>Description</b>	Customer book desirable flight
<b>Preconditions</b>	NA
<b>Post conditions</b>	NA
<b>Normal Flow</b>	NA
<b>Alternative Flow</b>	NA
<b>Exceptions Flow</b>	NA
<b>Includes</b>	UC-05, UC-06, UC-07, UC-08

#### 3.3 UC-03: Registration

<b>Use-Case ID</b>	UC-03		
<b>Use-Case Name</b>	Registration		
<b>Created By</b>	6 Ideas	<b>Last Updated By</b>	6 Ideas

<b>Date Created</b>	Feb. 08 2018	<b>Date Last Updated</b>	Feb. 11 2018
---------------------	--------------	--------------------------	--------------

<b>Actors</b>	Customer
<b>Description</b>	Customer must Register on the system in order to make a payment and cancel the flight.
<b>Preconditions</b>	NA
<b>Post conditions</b>	NA
<b>Normal Flow</b>	Customer should enter the valid information in the registration form such as phone number and email address
<b>Alternative Flow</b>	NA
<b>Exceptions Flow</b>	NA
<b>Includes</b>	NA

### 3.4 UC-04: Flight Cancellation

<b>Use-Case ID</b>	UC-04		
<b>Use-Case Name</b>	Flight Cancellation		
<b>Created By</b>	6 Ideas	<b>Last Updated By</b>	6 Ideas
<b>Date Created</b>	Feb. 08 2018	<b>Date Last Updated</b>	Feb. 11 2018

<b>Actors</b>	Customer
<b>Description</b>	The flight can be canceled by the customer.
<b>Preconditions</b>	NA
<b>Post conditions</b>	NA
<b>Normal Flow</b>	NA
<b>Alternative Flow</b>	NA
<b>Exceptions Flow</b>	The flight is missed by the customer
<b>Includes</b>	UC-09

### 3.5 UC-05: Select Flight

<b>Use-Case ID</b>	UC-05		
<b>Use-Case Name</b>	Select Flight		
<b>Created By</b>	6 Ideas	<b>Last Updated By</b>	6 Ideas
<b>Date Created</b>	Feb. 08 2018	<b>Date Last Updated</b>	Feb. 11 2018

<b>Actors</b>	Customer
<b>Description</b>	Customer select desired flight.
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>Customer should search for the flight based on date, time and cost.</li> <li>Customer state should be Flight Reservation</li> </ul>



<b>Post conditions</b>	NA
<b>Normal Flow</b>	NA
<b>Alternative Flow</b>	NA
<b>Exceptions Flow</b>	NA
<b>Includes</b>	UC-01

### 3.6 UC-06: Confirm Reservation

<b>Use-Case ID</b>	UC-06		
<b>Use-Case Name</b>	Confirm Reservation		
<b>Created By</b>	6 Ideas	<b>Last Updated By</b>	6 Ideas
<b>Date Created</b>	Feb. 08 2018	<b>Date Last Updated</b>	Feb. 11 2018

<b>Actors</b>	Customer
<b>Description</b>	Customer confirms the reserved Flight.
<b>Preconditions</b>	Customer state should be Flight Reservation
<b>Post conditions</b>	NA
<b>Normal Flow</b>	NA
<b>Alternative Flow</b>	NA
<b>Exceptions Flow</b>	The reservation information might be incorrect and the customer wants to correct it.
<b>Includes</b>	NA

### 3.7 UC-07: Make Payment

<b>Use-Case ID</b>	UC-07		
<b>Use-Case Name</b>	Make Payment		
<b>Created By</b>	6 Ideas	<b>Last Updated By</b>	6 Ideas
<b>Date Created</b>	Feb. 08 2018	<b>Date Last Updated</b>	Feb. 11 2018

<b>Actors</b>	Customer
<b>Description</b>	The desirable Flight should be paid by the customer.
<b>Preconditions</b>	The customer should Log-in to the system.
<b>Post conditions</b>	NA
<b>Normal Flow</b>	Customer enter valid bank account information.
<b>Alternative Flow</b>	NA
<b>Exceptions Flow</b>	<ul style="list-style-type: none"> <li>• The system might not accept credit Card</li> <li>• The system might not accept the specific Bank</li> <li>• The customer enter invalid card number or password</li> </ul>
<b>Includes</b>	UC-09

### 3.8 UC-08: Receive Ticket

<b>Use-Case ID</b>	UC-08		
<b>Use-Case Name</b>	Receive Ticket		
<b>Created By</b>	6 Ideas	<b>Last Updated By</b>	6 Ideas
<b>Date Created</b>	Feb. 08 2018	<b>Date Last Updated</b>	Feb. 11 2018

<b>Actors</b>	Customer
<b>Description</b>	The system provide the ticket to the customer by sending to his/her email.
<b>Preconditions</b>	Customer state should be Flight Reservation
<b>Post conditions</b>	Customer see the payment Confirmation
<b>Normal Flow</b>	Customer enter valid bank account information
<b>Alternative Flow</b>	NA
<b>Exceptions Flow</b>	<ul style="list-style-type: none"><li>• The system might not accept credit Card</li><li>• The system might not accept the specific Bank</li><li>• The customer enter invalid card number or password</li></ul>
<b>Includes</b>	NA

### 3.9 UC-09: Log in

<b>Use-Case ID</b>	UC-09		
<b>Use-Case Name</b>	Log in		
<b>Created By</b>	6 Ideas	<b>Last Updated By</b>	6 Ideas
<b>Date Created</b>	Feb. 08 2018	<b>Date Last Updated</b>	Feb. 11 2018

<b>Actors</b>	Customer
<b>Description</b>	The customer enter the username and password in order to log in to the system.
<b>Preconditions</b>	NA
<b>Post conditions</b>	Customer log in to the system so as to make a payment or cancel the flight.
<b>Normal Flow</b>	NA
<b>Alternative Flow</b>	<ul style="list-style-type: none"><li>• The ticket can be email</li><li>• The flight information send as a text Message</li></ul>
<b>Exceptions Flow</b>	NA
<b>Includes</b>	UC-03

### 3.10 UC-10: View Flight Information

<b>Use-Case ID</b>	UC-10		
<b>Use-Case Name</b>	View Information		
<b>Created By</b>	6 Ideas	<b>Last Updated By</b>	6 Ideas
<b>Date Created</b>	Feb. 08 2018	<b>Date Last Updated</b>	Feb. 11 2018

<b>Actors</b>	Manager
<b>Description</b>	Manger can view and access to the system information.
<b>Preconditions</b>	NA
<b>Post conditions</b>	NA
<b>Normal Flow</b>	NA
<b>Alternative Flow</b>	NA
<b>Exceptions Flow</b>	NA
<b>Includes</b>	NA

### 3.11 UC-11: Update Flight

<b>Use-Case ID</b>	UC-17		
<b>Use-Case Name</b>	Update Flight		
<b>Created By</b>	6 Ideas	<b>Last Updated By</b>	6 Ideas
<b>Date Created</b>	Feb. 08 2018	<b>Date Last Updated</b>	Feb. 11 2018

<b>Actors</b>	Manager
<b>Description</b>	Manger can update the system information (Add Flight, Modify Flight and Delete Flight).
<b>Preconditions</b>	NA
<b>Post conditions</b>	NA
<b>Normal Flow</b>	NA
<b>Alternative Flow</b>	NA
<b>Exceptions Flow</b>	System does not need any update.
<b>Includes</b>	NA

### 3.12 UC-12: View Order

<b>Use-Case ID</b>	UC-12		
<b>Use-Case Name</b>	View Order		
<b>Created By</b>	6 Ideas	<b>Last Updated By</b>	6 Ideas
<b>Date Created</b>	Feb. 08 2018	<b>Date Last Updated</b>	Feb. 11 2018

<b>Actors</b>	Manager
<b>Description</b>	The customers' orders can be viewed by the Manger
<b>Preconditions</b>	NA
<b>Post conditions</b>	NA
<b>Normal Flow</b>	NA
<b>Alternative Flow</b>	NA
<b>Exceptions Flow</b>	NA
<b>Includes</b>	NA

### 3.13 UC-13: Add Flight

<b>Use-Case ID</b>	UC-13		
<b>Use-Case Name</b>	Add Flight		
<b>Created By</b>	6 Ideas	<b>Last Updated By</b>	6 Ideas
<b>Date Created</b>	Feb. 08 2018	<b>Date Last Updated</b>	Feb. 11 2018

<b>Actors</b>	Manager
<b>Description</b>	The Manger must be able to add a new flight to the system.
<b>Preconditions</b>	NA
<b>Post conditions</b>	NA
<b>Normal Flow</b>	NA
<b>Alternative Flow</b>	NA
<b>Exceptions Flow</b>	The manger does not need to add a new flight to the system.
<b>Includes</b>	NA

### 3.14 UC-14: Modify Flight

<b>Use-Case ID</b>	UC-14		
<b>Use-Case Name</b>	Modify Flight		
<b>Created By</b>	6 Ideas	<b>Last Updated By</b>	6 Ideas
<b>Date Created</b>	Feb. 08 2018	<b>Date Last Updated</b>	Feb. 11 2018

<b>Actors</b>	Manager
<b>Description</b>	The Manger must be able to modify flight information such as flight number, time, price, etc.
<b>Preconditions</b>	NA
<b>Post conditions</b>	NA
<b>Normal Flow</b>	NA
<b>Alternative Flow</b>	NA
<b>Exceptions Flow</b>	The flight information does not need any update.
<b>Includes</b>	NA

### 3.15 UC-15: Delete Flight

<b>Use-Case ID</b>	UC-15		
<b>Use-Case Name</b>	Delete Flight		
<b>Created By</b>	6 Ideas	<b>Last Updated By</b>	6 Ideas
<b>Date Created</b>	Feb. 08 2018	<b>Date Last Updated</b>	Feb. 11 2018

<b>Actors</b>	Manager
<b>Description</b>	The Manger must be able to delete a flight from the system.
<b>Preconditions</b>	NA
<b>Post conditions</b>	NA
<b>Normal Flow</b>	NA
<b>Alternative Flow</b>	NA
<b>Exceptions Flow</b>	The manger does not need to delete any flight.
<b>Includes</b>	NA

## **4. Functional Requirements**

### **4.1 Search Flight**

The customer must search the flight based on the destination and time.

### **4.2 Flight Reservation**

The customer must select flight reservation for booking the flight.

### **4.3 User Registration**

The customer should register to the system in order to log in to the system.

### **4.4 Flight Cancellation**

The customer must select flight cancelation for canceling the flight

### **4.5 Select Flight**

The customer must select flight reservation for booking the flight.

### **4.6 Confirm Reservation**

The customer should able to confirm the reserved flight.

### **4.7 Make payment**

The Customer should able to pay the expense of the ticket to the airline.

### **4.8 Receive Ticket**

The system must be able to provide online ticket to the customer.

### **4.9 Log in**

The customer should have user name and password in order to login to the system.

### **4.10 View Information**

The system must show the flight's information to the manger.

### **4.11 Update Flight**

The manger must able to update the system's information.

### **4.12 View Order**

The system must show the customers' order to the manger.

#### **4.13 Add Flight**

The manger must be able to add a new flight to the system.

#### **4.14 Modify Flight**

The manger must be able to modify the flight information.

#### **4.15 Delete Flight**

The manger must be able to delete a flight.