



**COEN 6312**

**Model Driven Software Engineering**

**Project Deliverable 4**

**State Machines**

**Course Instructor**

Dr. Wahab Hamou-Lhadj

**Team: Techno\_Stars**

**Project Title: Airline Reservation System**

**Submitted by**

<b>Sravan Kumar Thumati</b>	<b>40070088</b>
<b>Prashanth reddy Somireddy</b>	<b>40047831</b>
<b>Nayana Raj Cheluvaraju</b>	<b>40071318</b>
<b>Sri Akhil Varma Alluri</b>	<b>40082333</b>
<b>Santosh reddy chelluri</b>	<b>40041074</b>

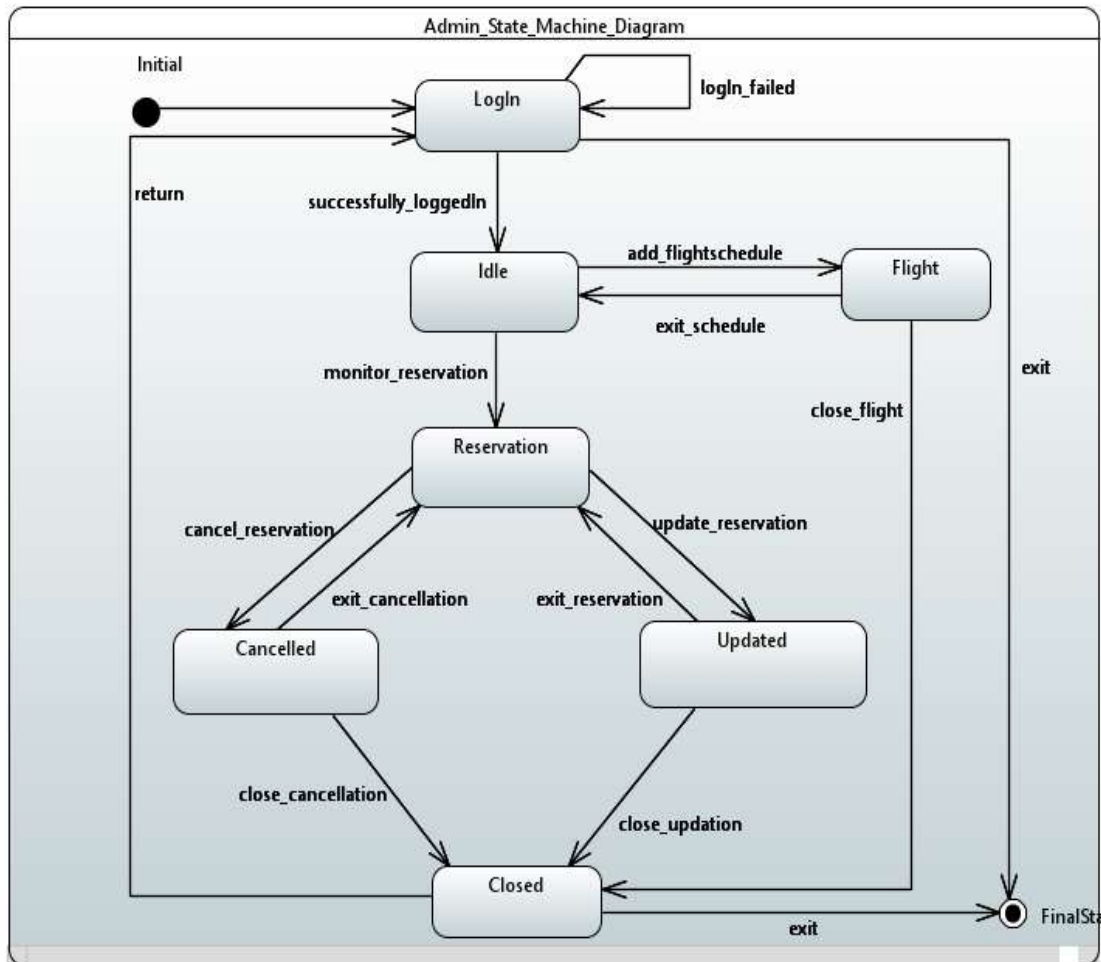
**Table of Contents**

**1. Admin State Diagram.....3**

**2. Customer State Diagram.....5**

**3. Booking State Diagram.....8**

## 1. Admin State Diagram:



### State Description:

The state diagram shows the process and functionalities of admin in flight reservation system. This state diagram consists of 7 states which are: Login state, Idle state, Flight state, Reservation state, cancelled state, Updated state and Closed state. Each state is having its own transition relationship. It will enter the other states only when there occurs a specified set of conditions or actions.

**Login:**The admin enters the login credentials. If the credentials entered are valid, admin enters the Idle state. If the credentials are invalid, admin have to re-enter the credentials or can exit.

**Idle:**

Once admin provides valid login credentials and enters Idle state, he/she can choose the transition state between Flight state and Reservation state.

**Flight:**

To enter this state the admin should choose the transition relationship of add flight schedule. Here admin can add new flight schedules. Admin can exit schedule to get back to Idle state.

**Reservation:**

Admin can enter Reservation state from Idle state if he/she choose to monitor reservations. From here the admin can choose the transition between Cancelled state and Update state.

**Cancelled:**

Admin can transit from Reservation state to Cancelled state if admin want to cancel a particular or set of reservations. Admin can enter reservation state from this state by performing exit cancellation action.

**Updated:**

Admin can transit from Reservation state to Updates state of he/she wants to update a particular or set of reservations. Example is to update flight timings, seat preferences etc., Admin can again enter reservation state from this state by exiting the update.

**Closed:**

Finally, if the admin wants to log out the session from cancelled/ updated/ flight states he/she enters closed state and can return to the login state or end the system.

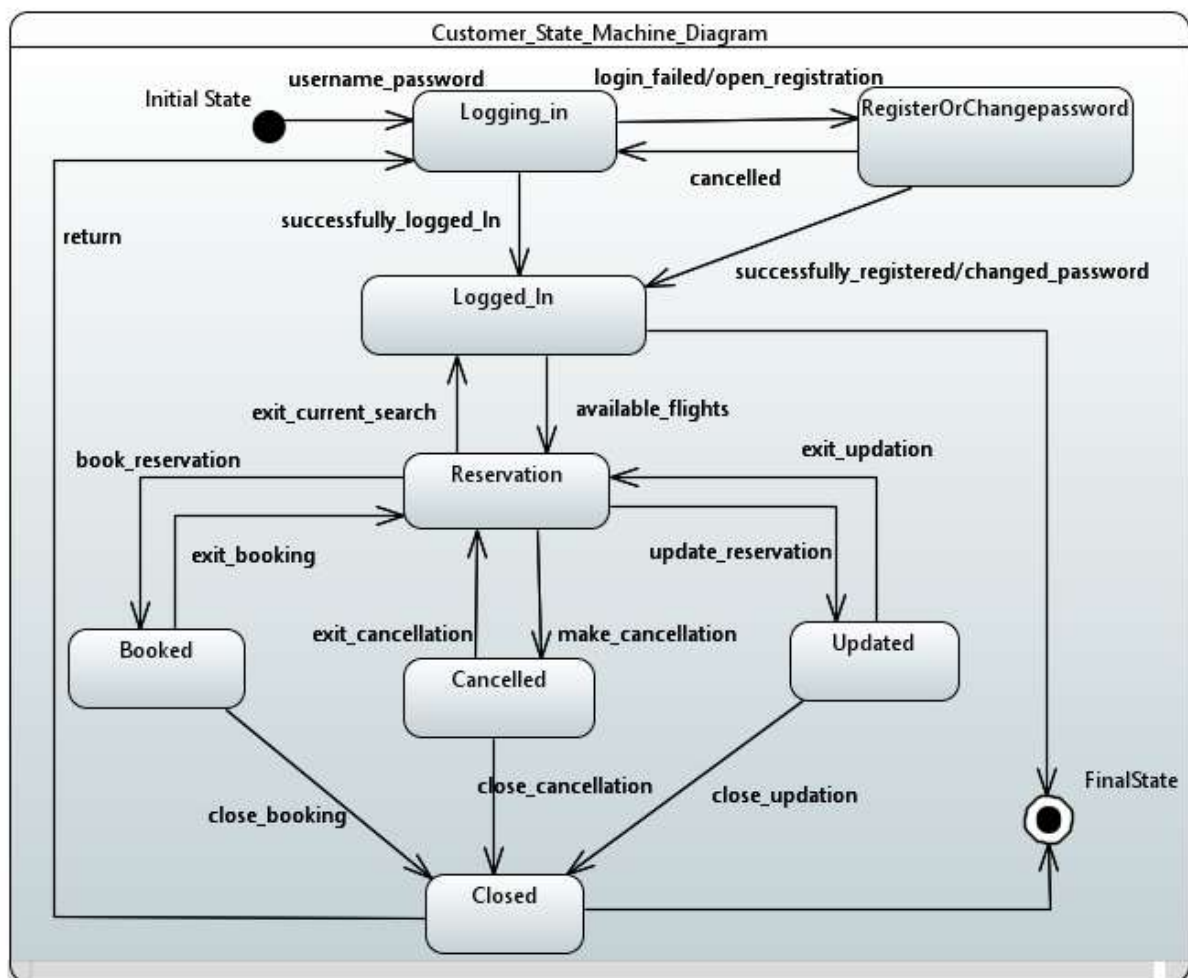
**Action Description:**

The below table describes the different transitions in detail.

Transition Name	Description
login_failed	If the login authentication is failed, this transition is used to re-enter the credentials.
successfully_loggedIn	This transition is used to enter Idle state after entering valid credentials
add_flightschedule	This transition is used to add new schedule to flights
exit_schedule	This transition is used to exit flight state and go back to Idle state
close_flight	This transition is used if user want to log off the session
monitor_reservation	This transition is used if the admin wants to cancel/update the reservation details or even to monitor the reservations.

cancel_reservation	If admin wants to cancel the ticket reservation/s this transition is used
exit_cancellation	This transition helps admin to go back to reservation state after ticket/s are cancelled
update_reservation	If admin wants to update the ticket reservation/s this transition is used to enter Updated state.
exit_reservation	This transition helps admin to go back to reservation state after updating the ticket details.
close_cancellation	Used to end the session by logging off and enter closed state.
close_updatation	Used to end the session by logging off and enter closed state.
return	This transition is used to enter the start of the module again, i.e., login.

## 2. Customer State Diagram:



## **State Description:**

The state diagram shows the process and functionalities of customer in flight reservation system. This state diagram consists of 8 states which are: Logging\_in state, Register or change password state, Logged\_in state, Reservation state, booked state, Cancelled state, Updated state and Closed state. Each state is having its own transition relationship. It will enter the other states only when there occurs a specified set of conditions or actions.

### **Logging\_in:**

The customer enters the login credentials. If the credentials entered are valid, customer enters the Logged in state. If the credentials are invalid, customer will be redirected to Register or change password state.

### **Register or change password:**

If the Logged in failed due to wrong credentials, then the customer is redirected to Register or change password state where the customer can register themselves or change password on successful registration or changed password they will be transmitted to Logged in state. He or she cancels the registration again it goes backs to logging in state.

### **Logged\_in:**

Once the customer is successfully logged in then he or she will be transmitted to the reservation state when he enters the availability flights for the current search.

### **Reservation:**

Customer can enter Reservation state from Logged\_In state if there are available flights and they can also go back to the Logged\_In state by choosing Exit from the current search. From reservation customer can choose the next state by selecting the action he or she wants to execute.

### **Booked:**

Customer can transmit from Reservation state to Booked state if customer wants to book a particular or set of reservations and once the booking of reservation is successful customer can choose to go back to the reservation state for next reservation or can close the booking by transmitting to the close State.

### **Cancelled:**

Customer can transit from Reservation state to Cancelled state if customer wants to cancel a

or set of reservations. Customer can enter reservation state or to the closed state from this State.

### **Updated:**

Customer can transit from Reservation state to Updates state of he/she wants to update a particular or set of reservations. Example is to change the Time preferences or , seat preferences etc., Customer can enter reservation state or close state from this state.

### **Closed:**

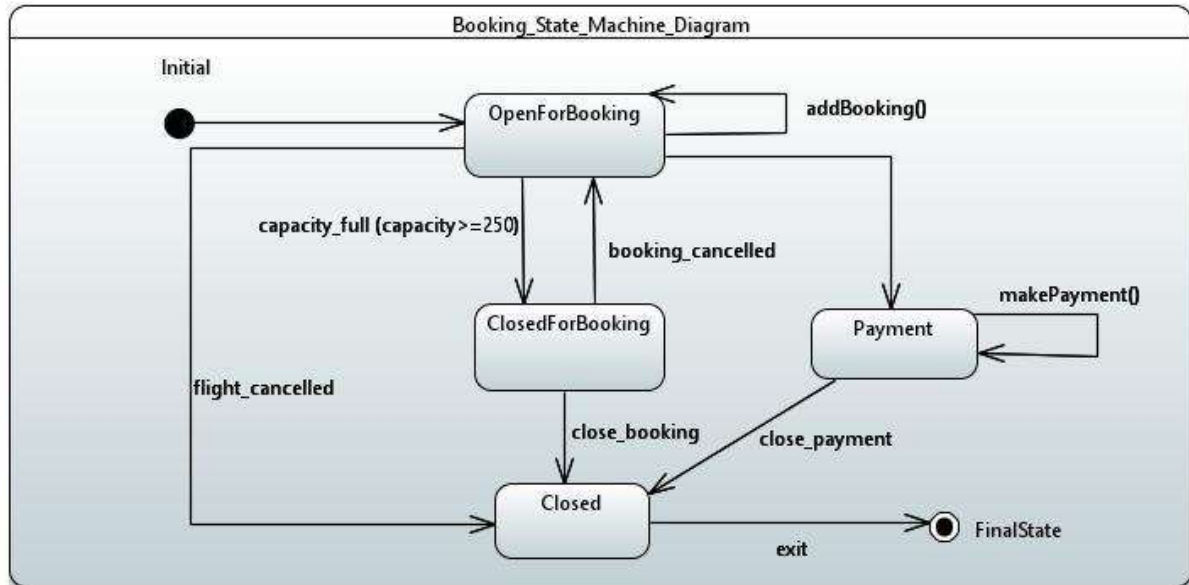
Finally, if the customer wants to log out the session from booked/cancelled/ updated/ flight states he/she enters closed state.

### **Action Description:**

The below table describes the different transitions in detail.

<b>Transition Name</b>	<b>Description</b>
login_failed/Open_Registration	If the login authentication is failed, this transition is used to register a customer.
successfully_loggedIn	This transition is used to enter Logged_in after entering valid credentials.
Available_flights	This transition is used to check the available flights for reservation.
exit_current_search	This transition is used to exit reservation state and go back to Logged_In state.
Book_Reservation	This transition is used if user want to book a reservation.
Exit_Booking	This transition is used if the customer wants to exit the Booked state and return back to the reservation state.
Close_booking	If customer wants to close the booked state this transition is used.
Make_cancellation	This transition helps customer to cancel the reservation.
Exit_Cancelling	This transition helps customer to go back to reservation state after ticket's are cancelled.
Update_Reservation	This transition helps customer to update a reservation for example updating the ticket details.
exit_updation	This transition helps customer to go back to reservation state after updating the ticket details.
close_updation	Used to end the session by logging off and enter closed state.
close_cancellation	Used to end the session by logging off and enter closed state.
return	This transition is used to enter the start of the module again, i.e., login.

### 3.Booking State Diagram:



#### State Description:

The booking state diagram shows the process for booking a flight ticket. This state diagram consists of 4 states like **OpenForBooking**, **ClosedForBooking**, **Payment** and **Closed**.

#### OpenForBooking:

Customers/Admin are able to book their ticket after proper authentication, in this state the customers/Admin are allowed for booking if the capacity of the flight is not full and further action is jumped to payment state.

#### ClosedForBooking:

If the capacity of the flight reached to maximum, the action enters this state and further bookings cannot be done, until any of the booking is cancelled.

#### Payment:

Customers enters this state once their booking is selected and payment is calculated.

#### Closed:

The closed state is entered if some of the following actions take place like, if the flight which is scheduled is cancelled or if the booking is closed or the payment for the booking is successful.



### Action Description:

The below table describes the different transitions in detail.

addBooking()	If customer/admin wants to book a ticket, this action is called.
flight_cancelled	If the flight is cancelled,the action transmits to Closed state.
capacity_full(capacity>=250 )	If the capacity of the flight reached to maximum, this action encounters and transmits to ClosedForBooking state.
booking_cancelled	It encounters when a booking is to be cancelled.
makePayment()	It encounters for a payment action for Payment state.
close_booking	Used to end the booking and move to closed state.
close_payment	Used to end the payment action and move to closed state.
exit	Used to exit from the entire booking process.

### Action Specification for Customer State Diagram:

```

switch(state) {

case LOGGING_IN:
if (keysuccessfully_logged_in)
    state = LOGGED_IN
if(keylogin_failed/open_registration)
    state = REGISTERORCHANGEPASSWORD
case REGISTERORCHANGEPASSWORD:
if(keycancelled)
    state = LOGGING_IN
if(keysuccessfully_registered/changedpassword)
    state = LOGGED_IN
case LOGGED_IN:
if(keyavailable_flights)
    state = RESERVATION
case RESERVATION:
if(keyexit_current_search)
    state = LOGGED_IN
if(keybook_reservation)

```

```
        state = BOOKED
    if(keymake_cancellation)
        state = CANCELLED
    if(keyupdate_reservation)
        state = UPDATED
    case BOOKED:
    if(keyexit_booking)
        state = RESERVATION
    if(keyclose_booking)
        state = CLOSED
    case CANCELLED:
    if(keyexit_cancellation)
        state = RESERVATION
    if(keyclose_cancellation)
        state = CLOSED
    case UPDATED:
    if(keyexit_updatation)
        state = RESERVATION
    if(keyclose_updatation)
        state = CLOSED
    case CLOSED:
    if(keyreturn)
        state = LOGGING_IN
```