

**COEN - 6312**

**Model Driven Software Engineering**

**Deliverable 4**

|  |  |
| --- | --- |
| **Student Name** | **Student ID** |
| **Ankit Chitlangia** | **27485573** |
| **Armaan Gill** | **27380003** |
| **Gurpreet Rehal** | **27632215** |
| **Mitesh Kaura** | **27284756** |
| **Simranjeet Singh** | **27376782** |

Contents

[State Diagrams 3](#_Toc446693329)

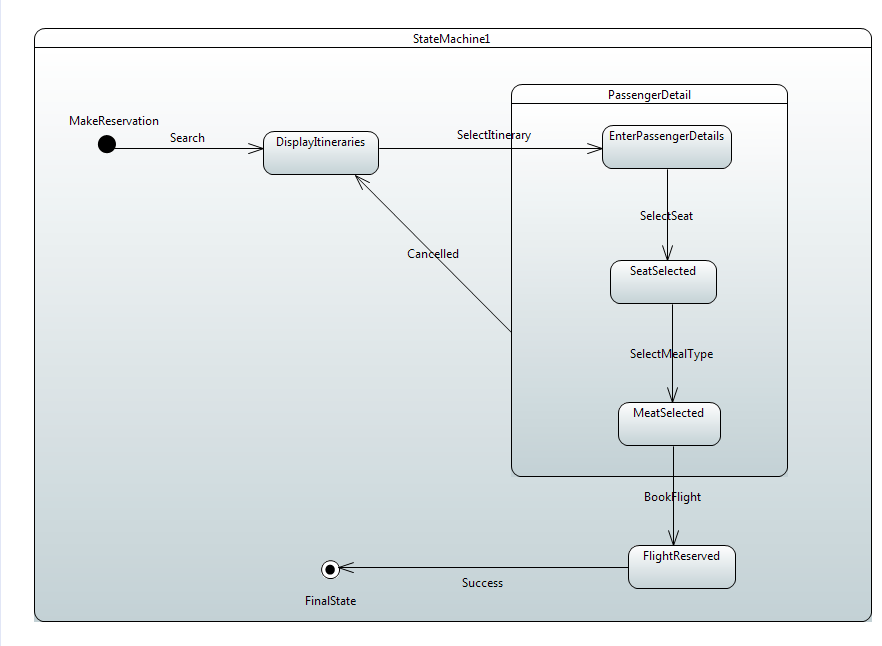
[BookFlight: 3](#_Toc446693330)

[Payment 5](#_Toc446693331)

[User Registration 8](#_Toc446693332)

State Diagrams:

## BookFlight:



**Action Specification**

**public** **class** BookFlight

{

**public** **static** bookFlight(String member\_name,passenger\_name,phone\_number//others)

{

**if**(member\_name!=**null** and passenger\_name!=**null** and phone\_number!=**null**)

{

//here we code to book flights.

bookingDetails(details);//// this is to display booking details like displaying id

}

**else**

generateAlert() // display error message

}

**public** **static** bookingDetails(String details)

{

//display the ticket details once the flight is booked

}

}

**public** **class** Cancellation

{

String cancel\_status;

**public** String cancelFlight(Long booking\_id)

{

If (booking\_id!=**null**)

{

cancel\_status=flightCancellationConfirmation(booking\_id)//code for canceling flight

**return**(cancel\_status);

}

**else** generateAlert() // display error message

**return** (message);

}

**public** string flightCancellationConfirmation(Long booking\_id)

{

**return**(status);//code for cancellation of flight

}

}

:searchFlight (arrival\_date, departure\_date, departure\_city, No\_of\_passenger, arival\_city)

If This.arrival\_date>=this.departure\_date and This.arival\_city!=this.departure\_city and

No\_of\_passenger!=null and trip!=null

Then

This.viewFlightInformation();

Else

Alert:= [Alert] = A1 : generateAlert [] on searchFlight

generateAlert()

endif

:bookingFlight(passenger\_name, seat\_preference, meal\_preference, phone\_number, email\_id)

If This. passenger\_name!=null and This. seat\_preference!=null and this. meal\_preference!=null and this. phone\_number!=null and this. email\_id!=null

Then

this.makePayment();

Else

Alert:= [Alert] = A1 : generateAlert [] on bookingFlight

generateAlert()

endif;

:cancelFlight( booking\_id, cancel\_status)

If this.booking\_id!=null

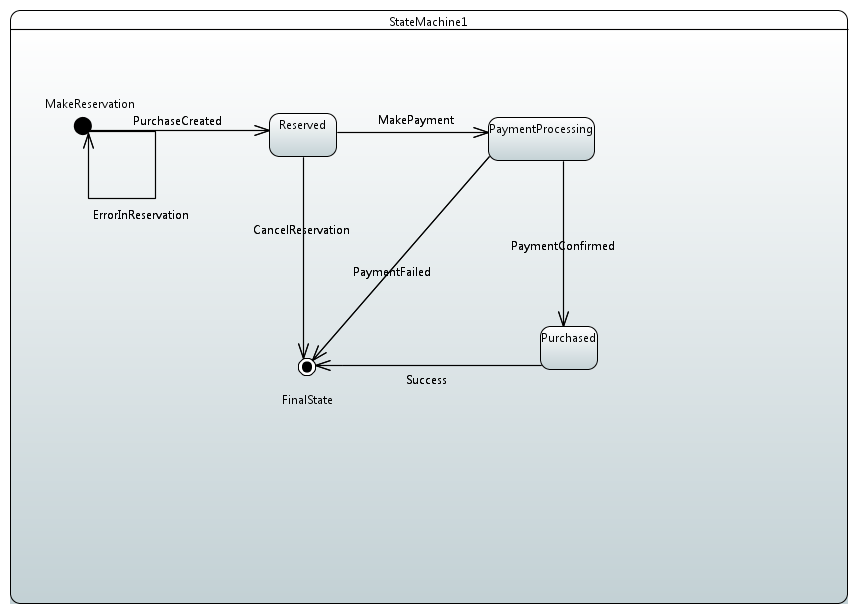
Then this.cancel\_status =this.

flightCancellationConfirmation()

Alert(cancel\_status);

endif;

## Payment



**Action Specification**

**public** **class** Payment

{

String payment\_status;

**public** **static** makePayment(**int** amount,**int** card\_number, String card\_holder\_name)

{

**if** (validateDetails(card\_number,card\_holder\_name)==**true**

and sufficientAmount(amount)==**true**)

{

payment\_status=paymentConfirm();//display the status about payment

System.out.println(payment\_status)

}

**else** generateAlert() // display error message

}

**public** String validateDetails(**int** card\_number, String card\_holder\_name )

{

//validate card information

**return** status;// status will be local variable which confirm the card details

}

**public** **int** sufficientAmount(**int** amount )

{

//validate sufficient balance

**return** status;// status will be local variable which display the sufficient amount

}

**public** String paymentConfirm()

{

//display message

**return** status;// status will be local variable which display the payment status

}

}

**public** **class** Cancellation

{

String cancel\_status;

**public** String cancelFlight(Long booking\_id)

{

If (booking\_id!=**null**)

{

cancel\_status=flightCancellationConfirmation(booking\_id)//code for canceling flight

**return**(cancel\_status);

}

**else** generateAlert() // display error message

**return** (message);

}

**public** string flightCancellationConfirmation(Long booking\_id)

{

**return**(status);//code for cancellation of flight

}

}

:makePayment(amount, payment\_status ,card\_number,card\_hoder\_name)

If this.validateCard(card\_number,card\_hoder\_name)==true and this.sufficientAmount(amount)==true

Then This.payment\_status= This.paymentConfirm()

Alert(payment\_status);

endif;

:cancelFlight( booking\_id, cancel\_status)

If this.booking\_id!=null

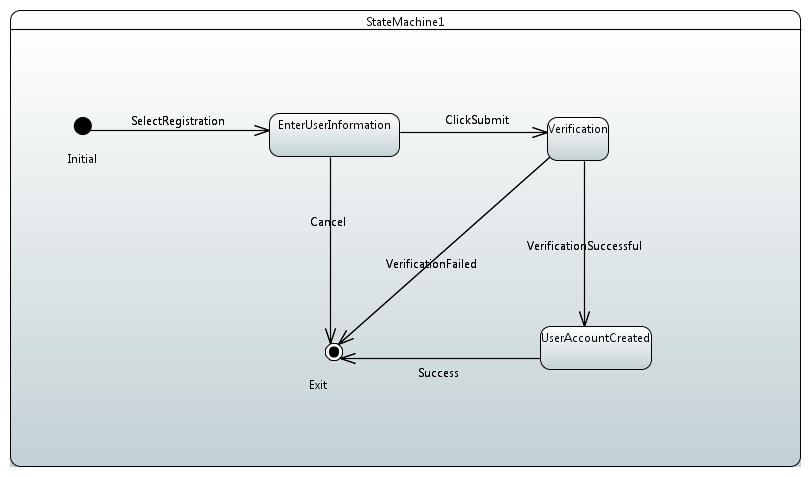
Then this.cancel\_status =this.

flightCancellationConfirmation()

Alert(cancel\_status);

endif;

## User Registration



**Action Specification**

**public** **class** RegisterUser {

**public** **static** **void** saveUser(String user\_name, **int** password,String first\_name,String last\_name,//others)

{

//code to register user by using all the parameters

}

}

**public** **class** Flight{

Public **static** **void** searchFlight(String arrival\_city,departure\_city,arrival\_date,departure\_date

number\_of\_passenger)

{

**if**(arrival\_city!=**null** and departure\_city!=**null** and arrival\_date!=**null** and departure\_date!=**null**)

{// here we code for search flights

// String result is local vairable to store search result.

ViewFlightInformation( result); // this is for displaying search result

}

**else**

generateAlert() // display error message

}

}

public **static** ViewFlightInformation(String result)

{ // display search result

// System.out.print(result) to display search}

}

}

:saveUser(String user\_name, int password,String first\_name,String last\_name)

If this. user\_name!=null and this.password!=null and this. first\_name!=null and this. last\_name!=null

Then This.userSaved = This.saveUser ()

Alert(userSaved);

endif;

:searchFlight(String arrival\_city,departure\_city,arrival\_date,departure\_date,number\_of\_passenger)

If this.arrival\_city!=null and this.departure\_city!=null and this.arrival\_date!=null and this.departure\_date!=null

Then This.ViewFlightInformation = This.FlightInformationViewed()

Alert(ViewFlightInformation);

endif;