

<i>Use case name</i>	SearchFlight
<i>Participating actors</i>	Initiated by the Customer
<i>Flow of events</i>	<ol style="list-style-type: none"> 1. The Customer activates the “SearchFlight” function of his terminal. 2. FlyByNightAirlines responds by presenting an Interface to the Customer. The interface includes a flight type menu (<i>round trip or one way trip</i>), departure and arrival information (<i>dates, specific time, specific airport</i>), seat types choices (<i>first class or coach</i>), number of stopovers list. 3. The Customer completes making detailed choices of his flight by <i>typing the locations and choosing flight and seat types, dates, times, numbers of stopovers</i>. The Customer then submits these information after making all choices. 4. FlyByNightAirlines receives these informations and <i>verifies if there is any possible typo among the information that the Customer input</i>. If has, it response a dialog. After verifying, it searches matching flights in database, and then <i>lists all satisfactory flights based on Customer’s requirements</i>, except those flights which has <i>no available seats left</i>. Each flight displayed should <i>include departure, arrival and travel time for each leg of a flight, the time for layovers, specific airport names of all stopovers, names of airplane companies, numbers of current available seats and total prices of all flight plans</i>. There is also a filter contains details and choices about flights to help Customer make choices. 5. The Customer chooses one of flight plans displayed and activates the MakeReservation use case.
<i>Entry condition</i>	<ul style="list-style-type: none"> • The Customer opens FlyByNightAirlines.
<i>Exit conditions</i>	<ul style="list-style-type: none"> • The Customer activates MakeReservation use case, OR • The Customer activates ReserveByAgency use case, OR • The Customer closes FlyByNightAirlines.
<i>Quality requirements</i>	<ul style="list-style-type: none"> • All the flight plans listed after searching should be reasonable. • Verification of the information inputed by the Customer should be accurate.

<i>Use case name</i>	MakeReservation
<i>Participating actors</i>	Initiated by the Customer OR the Travel Agency
<i>Flow of events</i>	<ol style="list-style-type: none"> 1. The Customer activates the “MakeReservation” function. 2. FlyByNightAirlines responds by presenting an interface to the Customer. <i>The interface includes personal information (name, birthday, gender), contract information (email, mobile phone number).</i> 3. The Customer fills in the information and then submits this personal information to FlyByNightAirlines. 4. FlyByNightAirlines receives the Customer’s information and checks the validity of information, if correct, then system activates the “MakePayments” function. Otherwise, system returns message pointing out wrong information at top of interface.
<i>Entry condition</i>	<ul style="list-style-type: none"> • The Customer activates MakeReservation.
<i>Exit conditions</i>	<ul style="list-style-type: none"> • The Customer activates MakePayment use case OR • The Customer closes MakeReservation OR • The Customer closes FlyByNightAirlines.

<i>Use case name</i>	CancelReservation
<i>Participating actors</i>	Initiated by the Customer OR the Travel Agency
<i>Flow of events</i>	<ol style="list-style-type: none"> 1. The Customer activates the “CancelReservation” function. 2. FlyByNightAirlines responds by presenting an interface to the Customer. <i>The interface includes flight information and flight order number which need the Customer to type.</i> 3. The Customer types information and then submits it to system. 4. FlyByNightAirlines receives the Customer’s information, if correct, system will process the CancelReservation request and return to the Customer a message that the request has been received. Otherwise, system returns message pointing out wrong information at top of interface.
<i>Entry condition</i>	<ul style="list-style-type: none"> • The Customer activates CancelReservation.
<i>Exit conditions</i>	<ul style="list-style-type: none"> • The Customer closes CancelReservation OR • The Customer closes FlyByNightAirlines.

<i>Use case name</i>	MakePayment
<i>Participating actors</i>	Initiated by the Customer Communicates with the Bank
<i>Flow of events</i>	<ol style="list-style-type: none"> 1. The Customer activates the “MakePayment” function of his terminal. 2. FlyByNightAirlines locks the Database, warns and gives the Customer 5 minutes to complete the payment. It sends another interface to Customer. The interface includes a table for Payment Information. It includes a Payment method menu(Visa, MasterCard, American Express) and blanks for Payment, Billing and Shipping information. 3. The Customer completes all the blanks and then submits these information. 4. FlyByNightAirlines checks the validity of these information, if fail, sends former interface again with <i>Wrong information warning on top</i>; if success, then making the payment to the Bank and waiting for the Success Information from the Bank. It modifies the Database’s information according to Customer’s request and release the locking of the Database. Then FlyByNightAirlines sends an interface shows “Payment success” to the Customer. It also sends an e-mail including order details and a conformation details(flight information).
<i>Entry condition</i>	<ul style="list-style-type: none"> • The Customer activates MakePayment.
<i>Exit conditions</i>	<ul style="list-style-type: none"> • The Customer closes FlyByNightAirlines.