

Detailing a Use Case

Use Case: Register

Actors	UserAccount (primary actor) User Database (secondary actor)
Pre-conditions	1. User does not have an account with Travpedia 2. User has an email address
Flow of events	<ol style="list-style-type: none">1. User opens the Travpedia website2. User is on the homepage and clicks 'Create an Account' link3. The system presents the user with an interface with empty fields asking for the following details:<ul style="list-style-type: none">- Name- Email Address- Password- Confirm Password4. User fills these fields in and clicks 'Next'5. User then has the option to provide their mobile number and tick 'please text and email me with Travpedia deals and promotional offers' if they wish to receive this service6. User ticks 'I have read and agree to the Terms and Conditions'7. User clicks 'Create Account' button<ul style="list-style-type: none">- If any details are invalid or a field has not been filled, the user will be shown an error message and asked to re-enter their information8. The system validates the users details and sends a confirmation email to the users email address9. The system displays a message telling the user that they should have received a confirmation email and that they should open this email to verify their email address10. User opens the confirmation email and clicks on the link to verify their email address11. The system redirects them back to Travpedia website and informs the user that their registration was successful

Post-conditions	<ol style="list-style-type: none"> 1. Details are stored in the user database and backed up 2. The user is now registered and is signed in 3. The user can now search and make a booking
Scenarios	<ol style="list-style-type: none"> 1. User forgets to tick the box confirming they have read and agree to the terms and conditions – cannot proceed to next page 2. User does not confirm their email address - their registration is incomplete 3. User leaves the interface before clicking 'Create Account' – registration is incomplete
Additional notes	The user has the opportunity to update their profile by going into their account, here they can specify travel and hotel preferences

Use Case: Make a booking (Hotel only)

Actors	UserAccount (primary actor) User Database (secondary actor) Product Database (secondary actor) Third Party Consortium (secondary actor)
Pre-conditions	<ol style="list-style-type: none"> 1. User is registered with Travpedia 2. User has signed in
Flow of events	<ol style="list-style-type: none"> 1. User is on the homepage where they can begin entering their search criteria 2. User ticks 'Accommodation'. Other options include: <ul style="list-style-type: none"> - Travel - Package Holiday 3. The interface changes slightly to accommodate booking a hotel only, the following details are required:

	<ul style="list-style-type: none"> - Destination/Hotel name (User can enter postcode, city, region or specific hotel) - Check in date (link to a calendar) - Check out date (link to a calendar) - How many rooms - Target price range <p>4. User clicks 'Search'</p> <p>5. The system presents the user with a list of available hotels, who have subscribed to Travpedia, matching their search criteria</p> <p>6. User browses the search results. Can filter by:</p> <ul style="list-style-type: none"> - Price - Star rating <p>And sort by:</p> <ul style="list-style-type: none"> - Price - Review score - Star rating <p>7. User click on the hotel they would like to book</p> <p>8. The system presents the user with a more detailed view of the hotel including photos</p> <p>9. User selects the number of room they require</p> <p>10. User clicks 'Book Now'</p> <p>11. The system presents the user with an interface with empty fields asking for various payment details:</p> <ul style="list-style-type: none"> - Name on card - Billing address - Card type - Card number - Expiry date - Security number - Voucher code (if entered, the system will reload the page displaying the new price to pay) - Option to have Travpedia save payment details <p>12. User fills in all the fields</p> <p>13. User ticks to agree to terms and conditions</p> <p>14. User clicks 'Pay Now'</p> <p>15. User is presented with another screen summarising booking and payments details</p> <p>16. User clicks 'Confirm'</p> <p>17. The transaction information is securely passed</p>
--	---

	<p>by the system to the credit/debit card consortium where the user's payment details are validated</p> <p>18. The booking is processed and handled by the GDS</p> <p>19. The system redirects the users to a page thanking them for their booking</p>
Post-conditions	<ol style="list-style-type: none"> 1. User receives a confirmation email 2. Details of the booking are stored in the user database – behaviour of user is used to determine future promotional offers and recommendations 3. Payment details are securely stored in the user database (if user selects to save payment details) 4. Details of the booking are available to view in 'Manage Bookings' 5. Hotel company is notified of the booking
Scenarios	<ol style="list-style-type: none"> 1. The destination or hotel name the user has specified produces no matches 2. User enters the incorrect account number – consortium confirms this, system generates an error message 3. User may think booking is complete after clicking 'Pay Now' – leaves website before clicking 'Confirm' – booking is incomplete 4. When user clicks 'Confirm', the system loads for several minutes and eventually times out – booking is incomplete

Use Case: Make a booking (Travel by plane)

Actors	<p>UserAccount (primary actor)</p> <p>User Database (secondary actor)</p> <p>Product Database (secondary actor)</p> <p>Third Party Consortium (secondary actor)</p>
Pre-conditions	<ol style="list-style-type: none"> 1. User is registered with Travpedia 2. User has signed in
Flow of events	<ol style="list-style-type: none"> 1. User is on the homepage where they can

	<p>begin entering their search criteria</p> <p>2. User ticks 'Travel'. Other options include:</p> <ul style="list-style-type: none"> - Accommodation - Package Holiday <p>3. The interface changes slightly to accommodate booking a journey</p> <p>4. User ticks 'One way', the following details are required:</p> <ul style="list-style-type: none"> - Type (of transport) - Date (link to calendar) - Start location - End location - Number of children - Number of adults <p>5. User clicks 'Search'</p> <p>6. The system presents the user with a list of available flights, whose airline companies have subscribed to Travpedia, matching their search criteria</p> <p>7. User browses the search results. Can filter by price and sort by:</p> <ul style="list-style-type: none"> - Price - Departure time <p>8. User clicks on the journey they would like to book</p> <p>8. The system presents the user with a more detailed view of the flight</p> <p>9. User selects how many items of luggage they wish to take on the flight</p> <p>10. User clicks 'Book Now'</p> <p>11. The system presents the user with an interface with the total price and empty fields asking for various payment details:</p> <ul style="list-style-type: none"> - Passport number - Passport issue date - Name on card - Billing address - Card type - Card number - Expiry date - Security number - Voucher code (if entered, the system
--	---

	<p>will reload the page displaying the new price to pay)</p> <p>- Option to have Travpedia save payment details</p> <p>12. User fills in all the fields</p> <p>13. User ticks to agree to terms and conditions</p> <p>14. User clicks 'Pay Now'</p> <p>15. User is presented with another screen summarising booking and payments details</p> <p>16. User clicks 'Confirm'</p> <p>17. The transaction information is securely passed by the system to the credit/debit card consortium where the user's payment details are validated</p> <p>18. The booking is processed and handled by the GDS</p> <p>19. The system redirects the users to a page thanking them for their booking</p>
Post-conditions	<p>1. User receives a confirmation email</p> <p>2. Details of the booking are stored in the user database – behaviour of user is used to determine future promotional offers and recommendations</p> <p>3. Payment details are securely stored in the user database (if user selects to save payment details)</p> <p>4. Details of the booking are available to view in 'Manage Bookings'</p> <p>5. Airline company is notified of the booking</p>
scenarios	<p>1. The search criteria produces no results</p> <p>2. User enters an invalid passport number - error message</p>

Use Case: Cancel a booking

Actors	<p>1. UserAccount (primary actor)</p> <p>2. Product Database (secondary actor)</p> <p>3. User Database (secondary actor)</p>
---------------	--

Pre-conditions	<ol style="list-style-type: none"> 1. User is registered 2. User is logged in 3. User has made a booking 4. The terms and conditions of the booking in question specify that cancellations are permitted
Flow of events	<ol style="list-style-type: none"> 1. User clicks 'Manage Bookings' on the website's homepage 2. The system directs the user to a page showing their past and current bookings 3. User clicks on the booking they wish to cancel 4. The system shows the booking in more detail with an option to cancel the booking 5. User clicks 'Cancel booking' 6. User is asked to confirm that they are sure they want to cancel the booking 7. User clicks 'Confirm Cancellation' 8. This information is relayed to the credit/debit card consortium which will handle the refund 9. The system redirects the user to a page confirming their cancellation was successful
Post-conditions	<ol style="list-style-type: none"> 1. Hotel company is notified of the cancellation 2. User receives email confirmation of cancellation 3. Booking is removed from the user's account in 'Manage Bookings' 4. Both databases are updated of changes 5. User receives appropriate refund
Scenarios	<ol style="list-style-type: none"> 1. User disputes a no refund policy – not possible using website. User must contact the company directly 2. User leaves the website after clicking 'Cancel Booking' – User has not clicked 'Confirm Cancellation' – Booking has not

	been cancelled
--	----------------