**Test Cases**

**User**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | |  | | | 1 | | |
| **Title** | |  | | | Main Menu | | |
| **Description** | |  | | | To test the main menu works correctly | | |
|  | | |  | | | | |
| **Step** | **Description** | | | **Type** | | **Expected Result** | **Actual Result** |
| 1 | Run system | | |  | | System opens and main menu displays | X |
| 2 | Test all menu items load when input | | |  | | System displays correct submenu or page when valid menu input is entered |  |
| 3 | Enter a valid input | | |  | | System loads the correct area of the system. |  |
| 4 | Enter invalid data Test Data: ‘54’, ‘Three’ | | |  | | Enter message appears, re-runs the main menu loop. |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | |  | 2 | | | | |
| **Title** | |  | Create Account | | | | |
| **Description** | |  | The user can create an account that saves to the xml | | | | |
|  | | | |  | | | |
| **Step** | **Description** | | | | **Type** | **Expected Result** | **Actual Result** |
| 1 | Load Main Menu | | | |  | System opens and main menu when run | X |
| 2 | Select create account from the main menu, by typing ‘2’ | | | |  | Create account function runs, prompting the user to enter their details |  |
| 3 | Enters valid data | | | |  | System continues to the next detail without any error messages, then prompted with confirmation. |  |
| 4 | Enter invalid data | | | |  | User cannot continue to next detail - Error message asking the user to enter valid data such a phone number must be a length of 12 digits. |  |
| 5 | Data is added to xml file on confirmation | | | |  | When the customer xml file is loaded, the users details will be inside each field e.g <USER> <NAME> Andrew </NAME> <EMAIL> jj@gmail.com</ EMAIL> |  |
| 6 | Data appends when another user is added | | | |  |  |  |
| 7 | If create account is cancelled or not confirmed, system does not write to xml file | | | |  |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | |  | 1 | | | | |
| **Title** | |  | Log in as a User | | | | |
| **Description** | |  | To log in to the system as a customer | | | | |
|  | | | |  | | | |
| **Step** | **Description** | | | | **Type** | **Expected Result** | **Actual Result** |
| 1 | Select Log In from the main menu, by typing ‘1’ | | | |  | x | X |
| 2 | Input valid email, invalid password | | | |  | Error message occurs, user must re-enter their email and password |  |
| 3 | Input invalid email, valid password | | | |  | Error message occurs, user must re-enter their email and password |  |
| 4 | Input valid email and password | | | |  | Log in successful message displays |  |
| 5 | Input invalid email and password | | | |  | Error message occurs, user must re-enter their email and password |  |
| 6 | System returns user to the regular user menu, logged in as name | | | |  |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | |  | 2 | | | | |
| **Title** | |  | Log in as an Admin | | | | |
| **Description** | |  | To log in to the system as an administrator | | | | |
|  | | | |  | | | |
| **Step** | **Description** | | | | **Type** | **Expected Result** | **Actual Result** |
| 1 | Select Log In from the main menu, by typing ‘1’ | | | |  | x | x |
| 2 | Input invalid admin email and password | | | |  |  |  |
| 3 | Input valid email and password | | | |  |  |  |
| 4 | System returns user to the regular user menu, logged in as name, with admin abilities. | | | |  |  |  |

**Booking**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | |  | | | | 3 | |
| **Title** | |  | | | | Book a flight | |
| **Description** | |  | | | | To book a flight as a customer | |
|  | | |  | | | | |
| **Step** | **Description** | | | **Type** | **Expected Result** | | **Actual Result** |
| 1 | Select book flight from the main menu, by typing ‘3’ then ‘1’ on the booking menu. | | |  | Flight booking menu opens | | Flight booking menu opens |
| 2 | Choose an existing flight | | | Formative | User will be able to progress | | User will be able to progress |
| 3 | Choose a nonexisting flight | | | Formative | Error message asking the user to pick another seat | | Error message asking the user to pick another seat |
| 3 | Choose a valid seat “1, 1” | | | Formative | Users flight gets booked | | Users flight gets booked |
| 4 | Choose an invalid seat ‘AY-SEVEN’/ ‘A99999’ | | |  | Error message prompting the user to pick a valid seat | |  |
| 5 | Choose an already booked seat | | |  | Error message, asking user to book another seat | |  |
| 6 | User is able to book another seat if they ‘Y’ is entered after ‘Book another? (Y/N): | | |  |  | |  |
| 7 | Booking is saved to booking xml file upon confirmation | | |  |  | |  |
| 8 | If system is closed before confirmation, no data is entered into the xml file | | |  |  | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | | 4 | | |
| **Title** | | Edit Booking | | |
| **Description** | | A user is able to edit a previous booking | | |
|  | | | | |
| **Step** | **Description** | | **Expected Result** | **Actual Result** |
| 1 | Select book flight from the main menu, by typing ‘3’ then ‘2’ on the booking menu. | | Edit booking function begins | X |
|  | System displays current bookings for the user, user selects the booking they want to change | |  |  |
|  | User enters new booking data | |  |  |
|  | User confirms new booking | |  |  |
|  | Ensure booking xml file has been appended | |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | | 5 | | |
| **Title** | | Cancel a booking | | |
| **Description** | | A user trying to cancel a booking | | |
|  | | | | |
| **Step** | **Description** | | **Expected Result** | **Actual Result** |
| 1 | System displays current bookings made | | x | X |
| 2 | User confirms cancelling of booking | |  |  |
| 3 | Booking XML file is appended and booking is removed | |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | |  | | |
| **Title** | | View Flight Data | | |
| **Description** | | Test the timeboard works correctly | | |
|  | | | | |
| **Step** | **Description** | | **Expected Result** | **Actual Result** |
| 1 | Check a flight times are sorted | | All flight times display correctly, in order of time after begin passed through the sorting algorithm | x |
| 2 | Ensure the queue works correctly | | Ensure flight times in the timeboard enqueues and dequeues. |  |
| 3 | Ensure flight delays and cancellations work correctly | | Ensure when a flight is randomly delayed it appears on the timeboard and is added to the queue later when it next due to arrive/depart. |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | |  | | |
| **Title** | | Staff data | | |
| **Description** | | To ensure all staff related functions work correctly | | |
|  | | | | |
| **Step** | **Description** | | **Expected Result** | **Actual Result** |
| 1 | Ensure staff can view their roles can be viewed | | When the user Is signed in as staff when the menu item for viewing job data is selected, all job data is displayed on the system correctly | x |
|  | To ensure admin can assign roles to staff. | | When the user s signed in as admin, jobs can be added to the terminal queue by the admin, and viewed later by the staff and admin. |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | |  | | |
| **Title** | |  | | |
| **Description** | |  | | |
|  | | | | |
| **Step** | **Description** | | **Expected Result** | **Actual Result** |
| 1 | x | | x | x |