**Test Cases -** Ishmail Hoque

|  |  |
| --- | --- |
| ID:TM08 | Use case: recordTransaction |
| Test Number:  1 | |
| Objective: Test the main flow | |
| Set Up:   1. Tester will make a transaction. 2. Monitor that the record of transaction is recorded in the SQL database | |
| Expected Results:   1. A record of the transactions is shown in the SQL database | |
| Test:   1. Tester makes a transaction. 2. Tester makes sure that the transactions are all stored into the correct corresponding folders. 3. Tester types in amounts paid by the different transaction types. 4. Tester checks if transaction type and amounts are correctly logged in the database. | |
| Test record: The expected behaviour is observed. | |
| Date: | Tester: |
| Test Result: | |

|  |  |
| --- | --- |
| ID:TM08.1 | Use case: connectionFailure |
| Test Number:  2 | |
| Objective: Test the alternative flow | |
| Set Up:   1. Tester is logged into the system. 2. Tester will monitor a failed connection between the transaction and the database. | |
| Expected Results:   1. Tester will see different outputs based on the connection. 2. Each output will be displayed to the user. | |
| Test:   1. Tester makes a transaction. 2. Tester closes window before transaction is gone through. 3. Tester reopens window to check error message output. 4. Tester retries to establish a connection. 5. Tester does not establish a connection. 6. Tester establishes a connection | |
| Test record: | |
| Date: | Tester: |
| Test Result: | |

|  |  |
| --- | --- |
| ID:TM09 | Use case: sellTicket |
| Test Number:  3 | |
| Objective: Test the main flow | |
| Set Up:   1. Tester will be logged into the system. 2. Tester will record the outcome of a ticket sale | |
| Expected Results:   1. System should create a ticket using the blank ticket. 2. System asks for payment. | |
| Test:   1. Tester makes a ticket through entering the correct details. 2. The system makes a ticket. 3. Tester makes sure all details are on the ticket. 4. System prompts Tester to make a payment. 5. Tester verifies payment. 6. System creates a record of the created ticket. | |
| Test record: | |
| Date: | Tester: |
| Test Result: | |

|  |  |
| --- | --- |
| ID:TM09.1 | Use case: declinedPayment |
| Test Number:  4 | |
| Objective: Test the alternative flow | |
| Set Up:   1. Tester will be logged into system. 2. Tester will make a transaction | |
| Expected Results:   1. Payment does not go through. 2. System does not record as a successful transaction | |
| Test:   * 1. Tester makes a transaction.   2. Tester uses wrong details when entering payment details.   3. System prompts that incorrect information was given.   4. System does not allow transaction.   5. System prompts to use another payment method. | |
| Test record: | |
| Date: | Tester: |
| Test Result: | |

|  |  |
| --- | --- |
| ID:TM10 | Use case: alteringTravelAdvisor |
| Test Number:  5 | |
| Objective: Test the main flow | |
| Set Up:   1. Tester has system admin permissions. 2. Tester is logged into the system. | |
| Expected Results:   1. Tester can successfully change the travel advisor details. | |
| Test:   1. Tester accesses the travel advisor management section. 2. Tester attempts to change the name and/or address of a travel advisor. 3. Tester verifies that the changes are reflected correctly in the database. 4. Tester attempt to change the name and/or address of a non-existent travel advisor and verify that an appropriate error message is displayed. 5. Tester verifies that the travel advisor's associated customer records are also updated with the new name and/or address. 6. Tester verifies that the changes made to the travel advisor details do not affect any other functionality in the system. | |
| Test record: The expected behaviour is observed. | |
| Date: | Tester: |
| Test Result: | |