**Test Case:** Registration

|  |  |
| --- | --- |
| **Test Case #:** 1  **System:** Tourist Assist App  **Designed by:** Gan Xu  **Short Description:** This test case verifies if Tourist registration function works properly. | **Test Case Name:** Tourist Registration |

|  |
| --- |
| **Pre-conditions:**  The user hasn’t registered yet and has entered the registration page to create an account. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User fills in the field of  Email, address, phone number and date of birth with wrong value.  Example:  Email: abc.com  Address: a@b  Phone number: 312 OlL 1234  Date of birth: 32, Jan, 2000 | The system should not permit the user to register, instead, it will point out invalid field and prompt the user to enter the correct information. | As expected. |
| 2 | Confirm password does not match with password | The system should detect this error and indicate it to the user until they match. | As expected |
| 3 | The user fills every field with correct information and clicks “submit” button | The system should first check if the email is already registered, if it is, the system should reject this request and ask the user to provide another email. Else, the system will send a confirmation letter to user’s email. | As expected |
| 4 | User clicks the confirmation link in email | The system should store user’s information in database, informing the success to the user. Then return to login page and auto-fill user’s email. | As expected |

|  |
| --- |
| **Exceptions:**  1. Invalid data causes exceptions  2. Data is not saved in database |

|  |
| --- |
| **Post-conditions:**  The user successfully created an account. |

**Test Case:** Login

|  |  |
| --- | --- |
| **Test Case #:** 2  **System:** Tourist Assist App  **Designed by:** Gan Xu  **Short Description:** This test case verifies if Tourist login function works properly.. | **Test Case Name:** Tourist Login |

|  |
| --- |
| **Pre-conditions:**  The user has not logged in yet and the system is in the login page. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user clicks “forgot password” and enters his/her email in the popping dialog. | The system should send an email to user’s mailbox in order to let the user change his/her password | As expected |
| 2 | The user chooses to login with Facebook or Google+ and enters a wrong Facebook or Google+ account. | The system should verify account information with Facebook or Google and reject the login. The user shall get an error message. | As expected |
| 3 | The user chooses to login with Facebook or Google+ and enters a correct Facebook or Google+ account. | The system should verify account information with Facebook or Google and permit the login. | As expected |
| 4 | The user chooses to login with TouristAssist account and enters wrong information | The system should check the email&password combination in database and then reject the login. The user shall get an error message. | As expected |
| 5 | The user chooses to login with TouristAssist account and enters correct information | The system should check the email&password combination in database and then permit the login. | As expected |
| 6 | The user has logged in successfully. | The system should jump to the user’s dashboard. | As expected |
| 7 | The user closes the app and re-opens it later. | The user should stay his/her login status without entering account and password again. | As expected |
| 8 | The user clicks “logout” button | The user should not be logged in any more. | As expected |

|  |
| --- |
| **Exceptions:**   1. Login information not found in database 2. User doesn’t stay logged in when he/she closes the app and reopen it. |

|  |
| --- |
| **Post-conditions:**  Tourist should be able be login and logout. |

**Test Case:** Dashboard

|  |  |
| --- | --- |
| **Test Case #:** 3  **System:** Tourist Assist App  **Designed by:** Gan Xu  **Short Description:** This test case verifies if the user can view his/her dashboard correctly. | **Test Case Name:** View dashboard |

|  |
| --- |
| **Pre-conditions:**  The user has logged in successfully. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user logs in and stays on dashboard. | The system should display the user’s completed trips, reminders, notes, checklists and trip timeline in the dashboard. | As expected |
| 2 | The user modifies information on dashboard.  Example:   1. Adds a reminder 2. Deletes a note 3. Changes trip timeline   Then the user comes back to dashboard. | The dashboard should reflect these updates. | As expected. |
| 3 | The user clicks “logout”. | The system will pull back to login page. | As expected |

|  |
| --- |
| **Exceptions:**   1. Dashboard information doesn’t refresh in time. |

|  |
| --- |
| **Post-conditions:**  The user can view updated information on dashboard after login. |

**Test Case:** Profile

|  |  |
| --- | --- |
| **Test Case #:** 4  **System:** Tourist Assist App  **Designed by:** Gan Xu  **Short Description:** This test case verifies if the user can view and update his/her own profile. | **Test Case Name:** User’s Profile |

|  |
| --- |
| **Pre-conditions:**  The user has logged in and stays on dashboard. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user clicks “Profile” button | The system should go to user’s profile page, where the user can view his/her email, name, address, phone number, photo, date of birth, friends, previous trips and previous reviews. While email and name are not editable, friends, previous trips and reviews are auto-generated by other modules of this app, the other fields are editable. | As expected |
| 2 | The user updates his/her information with wrong value.  Example:  Address: a@b  Phone number: 312 OlL 1234  Date of birth: 32, Jan, 2000  Photo: Wrong format of image | The system should reject the update and indicate invalid fields. | As expected |
| 3 | The user updates his/her information with correct information and clicks “Save Changes” | The system should ask the user to enter a PIN or password to confirm the change, and then update data in database. | As expected |
| 4 | The user deletes a friend in the Travelmate module, marks a trip as “finished” and modified a previous review. Then the user comes back to the profile page. | The system should display updated information. | As expected. |
| 5 | The user clicks “logout” | The system will not allow the user to view profile | As expected |
| 6 | The user sets his/her profile to be “invisible to others” | Other users will not see the user’s personal profile. | As expected |

|  |
| --- |
| **Exceptions:**   1. Information may not be up-to-date 2. The editable property of these fields may be mixed |

|  |
| --- |
| **Post-conditions:**  The user should be able to view and update his/her own profile. |

**Test Case:** Search Travel Mates

|  |  |
| --- | --- |
| **Test Case #:** 5  **System:** Tourist Assist App  **Designed by:** Gan Xu  **Short Description:** This test case verifies if the user can find travel mates correctly. | **Test Case Name:** Search Travel Mates |

|  |
| --- |
| **Pre-conditions:**  The user has a planned trip and would like to find travel mates. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | In the current trip module, the user confirms his/her destinations and timetable, then clicks “Find Mates”. Test with different data for many times. | The system should search in the database for other users who’s going to the same destination at the same time.  Then the system will list all the searching results. If no users are found, the system will recommend mates with close timetable and ask the user if he/she would to change destinations or timetable. | As expected. |
| 2 | The user has got a list of possible travel mates. Then the user click one of them. | The system should display the selected mate’s profile and timetable if the mate doesn’t set it to be invisible. | As expected |
| 3 | The user clicks “send message”, fills in a message and click “Confirm”. | The selected mate should receive the message in the mailbox, he/she should be able to reply or delete the message. | As expected |

|  |
| --- |
| **Exceptions:**   1. The system doesn’t find any proper travel mates. 2. The system displays user’s profile even if it’s set invisible. |

|  |
| --- |
| **Post-conditions:**  The user should be able to find proper travel mates and talk with them. |

**Test Case:** Communicates with other users

|  |  |
| --- | --- |
| **Test Case #:** 6  **System:** Tourist Assist App  **Designed by:** Gan Xu  **Short Description:** This test case verifies if the user can add friends and create groups correctly. | **Test Case Name:** Friends & Groups |

|  |
| --- |
| **Pre-conditions:**  The user is viewing another user’s profile and they are not friends yet. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user clicks “Send friend request”, enters a message in the popping message box and clicks “submit”. | The other user should receive the request and be able to choose from accept, ignore and decline. The user should get correspondence feedback. | As expected |
| 2 | The user clicks “Create a group”, enters a group name in the popping box and clicks “submit”. | The system should create a group for the user, set the user as default group owner and allow the user to invite his/her friends to the group. | As expected |
| 3 | The user would like to talk with a friend or in a group | The system should provide chatting service just like whatsapp or any other social communication app. | As expected |
| 4. | The user talks about travel plan with a friend or in a group | The user can create a plan chart in the chat page, everyone involved in the chat can modify it (just like Google drive) until they all agreed on the plan, then they should be allowed to add the plan to everyone’s dashboard. | As expected |

|  |
| --- |
| **Exceptions:**   1. Mutual plan is not correctly generated (synchronize problem, etc) 2. Chatting-related problems (lost wifi connection, chatting history, etc) |

|  |
| --- |
| **Post-conditions:**  The user should be able to add friends, create groups and talk about travel plan. |

**Test Case:** Help

|  |  |
| --- | --- |
| **Test Case #:** 7  **System:** Tourist Assist App  **Designed by:** Gan Xu  **Short Description:** This test cases verifies the user can get necessary help in the Help module. | **Test Case Name:** Help |

|  |
| --- |
| **Pre-conditions:**  The user has logged in and stays on dashboard. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User clicks Help module | The system should go to the Help page, including a list of FAQ, a search bar, a slider bar to control the beginner's mode, admin’s linkified phone & email and an edit text view for sending in-app message to admin. | As expected |
| 2 | User searches some keywords in the search bar. | The system should list FAQs concerning keywords. | As expected |
| 3 | User clicks admin’s phone | The system should dial the number automatically for the user. | As expected |
| 4 | User clicks admin’s email | The system should open the phone’s default mail app and autofill a letter’s receiver with admin’s email address. | As expected |
| 5 | User enters a message in the edit text view and clicks “send” | The admin should receive a notification about the in-app message and reply it. | As expected. |
| 6 | The user activates the “beginner’s mode”. | The system should provide guide when the user enters each module at the first time. | As expected |
| 7 | The user deactivates the “beginner’s mode” | The system will no longer show guide messages | As expected |

|  |
| --- |
| **Exceptions:**   1. FAQ not correctly searched. 2. App doesn’t have permission to make phone calls and send emails. |

|  |
| --- |
| **Post-conditions:**  The user should be able to get help when needed. |

**Test Case:** Admin

|  |  |
| --- | --- |
| **Test Case #:** 8  **System:** Tourist Assist App  **Designed by:** Gan Xu  **Short Description:** This test case verifies if the admin can modify database and provide help to users. | **Test Case Name:** Admin |

|  |
| --- |
| **Pre-conditions:**  The admin has logged in with admin’s account |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The admin adds/deletes/updates data on destinations, attractions, restaurants, hotels and FAQ | The database should reflect admin’s modification. | As expected |
| 2 | The admin enters the mail box. | The admin should see messages from users and reply them to provide help. | As expected |

|  |
| --- |
| **Exceptions:**   1. Database update error |

|  |
| --- |
| **Post-conditions:**  The admin should be able to serve normally. |

**Test Case:** Adding new trips.

|  |  |
| --- | --- |
| **Test Case #:** 9  **System:** Tourist Assist App  **Designed by:** Nishant Sharma  **Short Description:** This test case verifies if a user can add a new trip. | **Test Case Name:** Adding new trip. |

|  |
| --- |
| **Pre-conditions:**  The user has logged in with his account. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User clicks on the new trip information tab. | System opens a new page where user can add a new trip. | As expected |
| 2 | The tourist enters the desired destination, date, hotel details etc. | User can see his review in the text box. | As expected |
| 3 | User clicks on save after completing all the details. | New trip gets saved to the system database. | As expected |

|  |
| --- |
| **Exceptions:**   1. Database update error. 2. The desire location was not found in the database. |

|  |
| --- |
| **Post-conditions:**  The user is able to add a new trip normally. |

**Test Case:** Adding reviews.

|  |  |
| --- | --- |
| **Test Case #:** 10  **System:** Tourist Assist App  **Designed by:** Nishant Sharma  **Short Description:** This test case verifies if a user can add reviews for destination, attraction, restaurant or travel mate. | **Test Case Name:** Review |

|  |
| --- |
| **Pre-conditions:**  The user has logged in with his account. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User opens a destination, restaurants or an attraction page. | System opens the desired page. | As expected. |
| 2 | User clicks on the add a review tab. | System opens a text box where user can add his reviews. | As expected |
| 3 | User enters his review. | User can see his review in the text box. | As expected |
| 4 | User clicks on save after writing the review. | The review gets saved to the system database. | As expected |

|  |
| --- |
| **Exceptions:**   1. Database update error 2. The user enters a spam content |

|  |
| --- |
| **Post-conditions:**  The user is able to add his reviews normally. |

**Test Case:** Modifying/Deleting reviews.

|  |  |
| --- | --- |
| **Test Case #:** 11  **System:** Tourist Assist App  **Designed by:** Nishant Sharma  **Short Description:** This test case verifies if a user can modify/delete his previous reviews. | **Test Case Name:** Review |

|  |
| --- |
| **Pre-conditions:**  The user has logged in with his account. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User selects the review from any of the places reviewed by him. | System displays the option to edit or delete the review. | As expected |
| 2 | User edits or deletes his previous review. | System edits or deletes user review. | As expected |

|  |
| --- |
| **Exceptions:**   1. Database update error 2. The user enters a spam content |

|  |
| --- |
| **Post-conditions:**  The user is able to modify/delete reviews normally. |

**Test Case:** Adding photos.

|  |  |
| --- | --- |
| **Test Case #:** 12  **System:** Tourist Assist App  **Designed by:** Nishant Sharma  **Short Description:** This test case verifies if a user can add photos for destination, attraction, and restaurant. | **Test Case Name:** Adding photos. |

|  |
| --- |
| **Pre-conditions:**  The user has logged in with his account. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User opens a destination, restaurants or an attraction page. | System opens the desired page. | As expected |
| 2 | User clicks on add a photo tab. | System opens a new box for browsing the photo in users phone. | As expected |
| 3 | User browse and select the file from the computer and clicks on upload tab. | The photo gets saved to the system database. | As expected |

|  |
| --- |
| **Exceptions:**   1. Database update error 2. The user enters a spam content |

|  |
| --- |
| **Post-conditions:**  The user is able to add photos normally. |

**Test Case:** Edit/Delete photos.

|  |  |
| --- | --- |
| **Test Case #:** 13  **System:** Tourist Assist App  **Designed by:** Nishant Sharma  **Short Description:** This test case verifies if a user can edit/delete his previously uploaded photos. | **Test Case Name:** Edit/delete photos. |

|  |
| --- |
| **Pre-conditions:**  The user has logged in with his account. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User opens a page where he has previously added a photo. | System opens the desired page. | As expected |
| 2 | User clicks on edit/delete photo tab. | System edits/deletes the desired photo. | As expected |

|  |
| --- |
| **Exceptions:**   1. Database update error 2. The user enters a spam content |

|  |
| --- |
| **Post-conditions:**  The user is able to add photos normally. |

**Test Case:** Comment on reviews.

|  |  |
| --- | --- |
| **Test Case #:** 14  **System:** Tourist Assist App  **Designed by:** Nishant Sharma  **Short Description:** This test case verifies if a user can comment on a review by other user. | **Test Case Name:** Comment |

|  |
| --- |
| **Pre-conditions:**  The user has logged in with his account. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User selects any review or photos from the list of reviews or photos of a destination, attraction, restaurant or a different user. | System shows the option of replying to a review. | As expected |
| 2 | User types the comment to a reply. | Comment gets saved in the database. | As expected |

|  |
| --- |
| **Exceptions:**   1. Database update error 2. The user enters a spam content |

|  |
| --- |
| **Post-conditions:**  The user is able to add a comment to review normally. |

**Test Case:** Search destinations.

|  |  |
| --- | --- |
| **Test Case #:** 15  **System:** Tourist Assist App  **Designed by:** Nishant Sharma  **Short Description:** This test case verifies if a user can search a desired destination. | **Test Case Name:** Search destination |

|  |
| --- |
| **Pre-conditions:**  The user has logged in with his account. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User clicks the search destination tab. | System opens a text box for user to type the desired destination. | As expected |
| 2 | User enters the desired destination and click on search tab. | System displays information about the destination. | As expected |

|  |
| --- |
| **Exceptions:**   1. User location not found 2. User location does not exist in the database |

|  |
| --- |
| **Post-conditions:**  The user is able to search a destination normally. |

**Test Case:** View destinations.

|  |  |
| --- | --- |
| **Test Case #:** 16  **System:** Tourist Assist App  **Designed by:** Nishant Sharma  **Short Description:** This test case verifies if a user can view a desired destination. | **Test Case Name:** Search destination |

|  |
| --- |
| **Pre-conditions:**  The user has logged in with his account. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User clicks the search destination tab. | | System opens a textbox for user to type the desired destination. | As expected |
| 2 | User selects the desired destination. | | System displays information about the selected destination. | As expected |

|  |
| --- |
| **Exceptions:**   1. User location not found 2. User location does not exist in the database |

|  |
| --- |
| **Post-conditions:**  The user is able to search a destination normally. |

**Test Case:** View attraction.

|  |  |
| --- | --- |
| **Test Case #:** 17  **System:** Tourist Assist App  **Designed by:** Nishant Sharma  **Short Description:** This test case verifies if a tourist is able to view an attraction. | **Test Case Name:** View attraction |

|  |
| --- |
| **Pre-conditions:**  The user has logged in with his account. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User search for his desired location | System search for user’s desired location. | As expected |
| 2 | User select’s particular attraction. | System opens a page with all details of that attraction. | As expected |

|  |
| --- |
| **Exceptions:**   1. The direction does not exist 2. The direction is not found in the database |

|  |
| --- |
| **Post-conditions:**  The user is able to see details of a desired attraction. |

**Test Case:** Direction to attraction.

|  |  |
| --- | --- |
| **Test Case #:** 18  **System:** Tourist Assist App  **Designed by:** Nishant Sharma  **Short Description:** This test case verifies if a tourist is able to view the direction to his desired attraction location | **Test Case Name:** Direction |

|  |
| --- |
| **Pre-conditions:**  The user has logged in with his account. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User search for his desired location | System search for user’s desired location. | As expected |
| 2 | User select’s particular attraction. | System opens a page with all details of that attraction. | As expected |
| 3 | User clicks on direction to attraction. | System displays the direction to the tourist desired attraction. | As expected |

|  |
| --- |
| **Exceptions:**   1. The direction does not exist 2. The direction is not found in the database |

|  |
| --- |
| **Post-conditions:**  The user is able to see directions to the desired attraction. |

**Test Case:** Add attraction to trip timeline:

|  |  |
| --- | --- |
| **Test Case #:** 19  **System:** Tourist Assist App  **Designed by:** Nishant Sharma  **Short Description:** This test case verifies if a user can add attraction to trip timeline. | **Test Case Name:** Add attraction to trip timeline |

|  |
| --- |
| **Pre-conditions:**  The user has logged in with his account. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User selects the desired attraction. | System displays information and options about the attraction. | As expected |
| 2 | User clicks on the Add attraction to trip timeline tab. | System adds the selected attraction to the trip timeline. | As expected |

|  |
| --- |
| **Exceptions:**   1. User attraction not found 2. User attraction does not exist in the database |

|  |
| --- |
| **Post-conditions:**  The user is able to search add attraction to trip timeline normally. |

**Test Case:** View map of attraction.

|  |  |
| --- | --- |
| **Test Case #:** 20  **System:** Tourist Assist App  **Designed by:** Nishant Sharma  **Short Description:** This test case verifies if a tourist is able to view a map of all the attractions near his attraction. | **Test Case Name:** Map of attraction |

|  |
| --- |
| **Pre-conditions:**  The user has logged in with his account. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User search for his desired location | System search for user’s desired location. | As expected |
| 2 | User select’s particular attraction. | System opens a page with all details of that attraction. | As expected |
| 3 | User clicks on view map. | System displays the map of all the attractions near user’s attraction. | As expected |

|  |
| --- |
| **Exceptions:**   1. The direction does not exist 2. The direction is not found in the database |

|  |
| --- |
| **Post-conditions:**  The user is able to see directions to the desired attraction. |

**Test Case:** View hotel list.

|  |  |
| --- | --- |
| **Test Case #:** 21  **System:** Tourist Assist App  **Designed by:** Nishant Sharma  **Short Description:** This test case verifies if a tourist is able to view list of hotels near his attraction. | **Test Case Name:** Hotel list |

|  |
| --- |
| **Pre-conditions:**  The user has logged in with his account. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User search for his desired location | System search for user’s desired location. | As expected |
| 2 | User select’s particular attraction. | System opens a page with all details of that attraction. | As expected |
| 3 | User clicks on view hotel list tab. | System displays the list of all the hotels near user’s attraction. | As expected |

|  |
| --- |
| **Exceptions:**   1. No hotels found near user’s location. |

|  |
| --- |
| **Post-conditions:**  The user is able to see list of hotels near his attraction. |

**Test Case:** Add hotel to trip.

|  |  |
| --- | --- |
| **Test Case #:** 22  **System:** Tourist Assist App  **Designed by:** Nishant Sharma  **Short Description:** This test case verifies if a tourist is able to add a hotel to his trip. | **Test Case Name:** Hotel list |

|  |
| --- |
| **Pre-conditions:**  The user has logged in with his account. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User searches for the trip to which he wants to add. | System displays the trip. | As expected |
| 2 | User select’s add a hotel to a trip tab. | System shows the list of the hotels available. | As expected |
| 3 | User selects the hotels from the list. | System adds selected hotel to the trip. | As expected |

|  |
| --- |
| **Exceptions:**   1. Database update error |

|  |
| --- |
| **Post-conditions:**  The user is able to add hotel to his trip. |

**Test Case:** View restaurants.

|  |  |
| --- | --- |
| **Test Case #:** 23  **System:** Tourist Assist App  **Designed by:** Nishant Sharma  **Short Description:** This test case verifies if a tourist is able to view a list of restaurants. | **Test Case Name:** View restaurants |

|  |
| --- |
| **Pre-conditions:**  The user has logged in with his account. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | User search for his desired location | System search for user’s desired location. | As expected |
| 2 | User select’s particular attraction. | System opens a page with all details of that attraction. | As expected |
| 3 | User clicks on view restaurants list tab. | System displays the list of all the restaurants near user’s attraction. | As expected |

|  |
| --- |
| **Exceptions:**   1. No restaurants near user’s attraction. |

|  |
| --- |
| **Post-conditions:**  The user is able to view list of restaurants near his attraction. |

**Test Case: View Completed Trips**

|  |  |
| --- | --- |
| **Test Case #:** 24  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** The test is to verify that the user can view his previous trips | **Test Case Name:** Completed Trips: |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Trip page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The User clicks on the trips module and click on the completed trip | The system is expected to display all store previous trip by the user on the page | As expected |
| 2 | The user selects one from the option and clicks to view more | The system is expected to display all details of the trip including the time, dates and location | As expected |
| 3 | The user clicks the close button to return back to the previous page | The system is expected to take the user back to the previous viewed page | As expected |

|  |
| --- |
| **Exceptions:** The user travel history is not stored on the data base  The Admin restricts the user from viewing his completed trip history |

|  |
| --- |
| **Post-conditions:** The user views his completed trip history page |

**Test Case: View current trip**

|  |  |
| --- | --- |
| **Test Case #: 25**  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify the user is able to view his View current trip | **Test Case Name:** View current trips |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Trip page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The User clicks on the trips module and click on view current trip | The system is expected to take him to the page where he can find timeline of the trip, completed destinations, remaining destinations, visited attractions, remaining attractions, photos from current trips and restaurants to visit. | As expected |
| 2 | The user selects one from the option and clicks to view more | The system is expected to display all details of the current trip including all details | As expected |
| 3 | The user clicks the close button to return back to the previous page | The system is expected to take the user back to the previous viewed page | As expected |

|  |
| --- |
| **Exceptions:** The user current history details is not stored on the database  The Admin restricts the user from viewing his current history details  The user is not logged in |

|  |
| --- |
| **Post-conditions:** The user views is current trip details |

**Test Case: Start Trip**

|  |  |
| --- | --- |
| **Test Case #: 26**  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify the user is able to select and update his trip from the planned trips | **Test Case Name:** start trip |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Trip page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The User clicks on the trips module and click on start trip from the planned trips | The system updates the trip details from planned trip to current trip | As Expected |
| 2 | The user views his current trip section | The system is expected to update the user current trip section to the selected trip | As Expected |
| 3 | The user views his dashboard to also check his current trip | The system is expected to update the user dashboard with the selected star trip | As Expected |

|  |
| --- |
| **Exceptions:** Database error  The user selects a location blocked by the Admin  The user did not click the start trip button |

|  |
| --- |
| **Post-conditions:** The user current trip will be updated with the chosen trip from the start trip button |

**Test Case: Plan Trip:**

|  |  |
| --- | --- |
| **Test Case #:** 27  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify that the Plan trip module work correctly | **Test Case Name:** Plan trip |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Current Trip Page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user clicks the tab to plan new tip by editing the date, time, destination and more | The user should be able to plan his new trip by editing all necessary information | As Expected |
| 2 | The user clicks the tab to update the set date and destination for a planned trip | The user should be able to update the set information and the system is expected to store the new information | As Expected |
| 3 | The users clicks the tab to add destination, attractions and plan restaurants | The user should be able to add new details to his planned trip and the system is expected to store all information | As Expected |
| 4 | The user clicks the tab to add events to the calendar | The user should be able to add new events to his calendar and the system is expected to store all information | As Expected |
| **Exceptions:** Database Error  The admin restricts the user's account the features it can perform  The user is not logged in | | | |

|  |
| --- |
| **Post-conditions:** The user is able to add new plan trip, edit and update all information about his trip |

**Test Case: View Timeline**

|  |  |
| --- | --- |
| **Test Case #:** 28  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify that the view timeline module work correctly | **Test Case Name:** View Timeline |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Current Trip Page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user clicks on the Timeline tab to view a highlight details of his trip | The system is expected to display all trip information in a Chart fashion | As Expected |

|  |
| --- |
| **Exceptions:** Database Error  The user haven’t added any destination details  The user is not logged in  The user account is restricted by the Amin |

|  |
| --- |
| **Post-conditions:** The User is expected to view the details of his trip information |

**Test Case: Add Hotel/Stay in selected destination:**

|  |  |
| --- | --- |
| **Test Case #:** 29  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify that the user can add/stay in hotels in selected location | **Test Case Name:**  Add Hotel/Stay in selected destination: |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Current Trip Page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user click the link to add/stay in a hotel in a selected location | The system is expected to add the selected hotel detail to his travel details | As Expected |

|  |
| --- |
| **Exceptions:** Database Error  The User is not logged in  The User account is restricted from performing this action |

|  |
| --- |
| **Post-conditions:** The User is able to add a hotel he wants to stay to his selected destination |

**Test Case: Complete Trip**

|  |  |
| --- | --- |
| **Test Case #:** 30  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify that the user can complete the trip | **Test Case Name:** complete trip. |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Current Trip Page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user clicks the link to complete a current trip | The system is expected to confirm from the user if he really want to complete the trip before removing the trip from the current trip to completed trip | As Expected |

|  |
| --- |
| **Exceptions:** Database Error  The user is not logged in  The Admin restrict the functionality from the users account |

|  |
| --- |
| **Post-conditions:** The user completed the trip and his dashboard is updated with all details |

**Test Case: Manage Notes**

|  |  |
| --- | --- |
| **Test Case #: 31**  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify that the user can manage Notes | **Test Case Name:** Manage Notes |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Utilities Page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user click the link to add a new note to the planned trip or current trip | The system is expected to create a note, ask the user to name it and store the new note to the database | As Expected |
| 2 | The user clicks the link to edit/delete his created note | The system is expected to edit the required note to the new details and store it to the database | As Expected |

|  |
| --- |
| **Exceptions:** Database Error  The user is not logged in  The Admin restrict the functionality from the users account |

|  |
| --- |
| **Post-conditions:** The user is able to perform his desired activity either to create, edit or delete his notes |

**Test Case: Manage Reminder**

|  |  |
| --- | --- |
| **Test Case #: 32**  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify that the user can manage Reminder | **Test Case Name:** Manage Reminder |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Utilities Page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user click the link to add a new reminder | The system is expected to add a new reminder and store it on the database | As Expected |
| 2 | The user click the link to add the date, time and reminder text | The system is expected to save to the details to the database and remind the user on the desired time | As Expected |

|  |
| --- |
| **Exceptions:** Database Error  The user is not logged in  The Admin restrict the functionality from the user's account |

|  |
| --- |
| **Post-conditions:** The user sets a reminder and the reminder notifies the user at the desired time |

**Test Case: Manage checklist**

|  |  |
| --- | --- |
| **Test Case #: 33**  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify that the user can manage Reminder | **Test Case Name:** Manage Checklist |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Utilities Page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user click the link to add a new checklist to the planned or current trip | The system is expected a new checklist and store the details when the user is done adding all details | As Expected |
| 2 | The user clicks the link to edit the checklist | The system is expected to allow the user to edit the created check list and store the new details in the database | As expected |

|  |
| --- |
| **Exceptions:** Database Error  The user is not logged in  The Admin restrict the functionality from the user's account |

|  |
| --- |
| **Post-conditions:** The user is able to add, edit and delete his checklist |

**Test Case: View Calendar**

|  |  |
| --- | --- |
| **Test Case #: 34**  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify that the user can View the Calendar | **Test Case Name:** View Calendar |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Utilities Page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user clicks the link to view the calendar events | The system is expected to show all details of the planned trip to the user | As Expected |
| 2 | The user clicks on any trip for more information | The system is expected to show more information to the user | As Expected |

|  |
| --- |
| **Exceptions:** Database Error  The user is not logged in  The Admin restrict the functionality from the user's account |

|  |
| --- |
| **Post-conditions:** The user is able to view his calendar and details about his trips |

**Test Case: View Completed Trips**

|  |  |
| --- | --- |
| **Test Case #:** 35  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** The test is to verify that the user can view his previous trips | **Test Case Name:** Completed Trips: |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Trip page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The User clicks on the trips module and click on the completed trip | The system is expected to display all store previous trip by the user on the page | As expected |
| 2 | The user selects one from the option and clicks to view more | The system is expected to display all details of the trip including the time, dates and location | As expected |
| 3 | The user clicks the close button to return back to the previous page | The system is expected to take the user back to the previous viewed page | As expected |

|  |
| --- |
| **Exceptions:** The user travel history is not stored on the data base  The Admin restricts the user from viewing his completed trip history |

|  |
| --- |
| **Post-conditions:** The user views his completed trip history page |

**Test Case: View current trip**

|  |  |
| --- | --- |
| **Test Case #: 36**  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify the user is able to view his View current trip | **Test Case Name:** View current trips |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Trip page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The User clicks on the trips module and click on view current trip | The system is expected to take him to the page where he can find timeline of the trip, completed destinations, remaining destinations, visited attractions, remaining attractions, photos from current trips and restaurants to visit. | As expected |
| 2 | The user selects one from the option and clicks to view more | The system is expected to display all details of the current trip including all details | As expected |
| 3 | The user clicks the close button to return back to the previous page | The system is expected to take the user back to the previous viewed page | As expected |

|  |
| --- |
| **Exceptions:** The user current history details is not stored on the database  The Admin restricts the user from viewing his current history details  The user is not logged in |

|  |
| --- |
| **Post-conditions:** The user views is current trip details |

**Test Case: Start Trip**

|  |  |
| --- | --- |
| **Test Case #: 37**  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify the user is able to select and update his trip from the planned trips | **Test Case Name:** start trip |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Trip page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The User clicks on the trips module and click on start trip from the planned trips | The system updates the trip details from planned trip to current trip | As Expected |
| 2 | The user views his current trip section | The system is epected to update the user current trip section to the selected trip | As Expected |
| 3 | The user views his dashboard to also check his current trip | The system is expected to update the user dashboard with the selected star trip | As Expected |

|  |
| --- |
| **Exceptions:** Database error  The user selects a location blocked by the Admin  The user did not click the start trip button |

|  |
| --- |
| **Post-conditions:** The user current trip will be updated with the chosen trip from the start trip button |

**Test Case: Plan Trip:**

|  |  |
| --- | --- |
| **Test Case #:** 38  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify that the Plan trip module work correctly | **Test Case Name:** Plan trip |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Current Trip Page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user clicks the tab to plan new tip by editing the date, time, destination and more | The user should be able to plan his new trip by editing all necessary information | As Expected |
| 2 | The user clicks the tab to update the set date and destination for a planned trip | The user should be able to update the set information and the system is expected to store the new information | As Expected |
| 3 | The users clicks the tab to add destination, attractions and plan restaurants | The user should be able to add new details to his planned trip and the system is expected to store all information | As Expected |
| 4 | The user clicks the tab to add events to the calendar | The user should be able to add new events to his calendar and the system is expected to store all information | As Expected |
| **Exceptions:** Database Error  The admin restricts the user's account the the features it can perform  The user is not logged in | | | |

|  |
| --- |
| **Post-conditions:** The user is able to add new plan trip, edit and update all information about his trip |

**Test Case: View Timeline**

|  |  |
| --- | --- |
| **Test Case #:** 39  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify that the view timeline module work correctly | **Test Case Name:** View Timeline |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Current Trip Page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user clicks on the Timeline tab to view a highlight details of his trip | The system is expected to display all trip information in a Chart fashion | As Expected |

|  |
| --- |
| **Exceptions:** Database Error  The user haven’t added any destination details  The user is not logged in  The user account is restricted by the Amin |

|  |
| --- |
| **Post-conditions:** The User is expected to view the details of his trip information |

**Test Case: Add Hotel/Stay in selected destination:**

|  |  |
| --- | --- |
| **Test Case #:** 40  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify that the user can add/stay in hotels in selected location | **Test Case Name:**  Add Hotel/Stay in selected destination: |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Current Trip Page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user click the link to add/stay in a hotel in a selected location | The system is expected to add the selected hotel detail to his travel details | As Expected |

|  |
| --- |
| **Exceptions:** Database Error  The User is not logged in  The User account is restricted from performing this action |

|  |
| --- |
| **Post-conditions:** The User is able to add a hotel he wants to stay to his selected destination |

**Test Case: Complete Trip**

|  |  |
| --- | --- |
| **Test Case #:** 41  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify that the user can complete the trip | **Test Case Name:** complete trip. |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Current Trip Page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user clicks the link to complete a current trip | The system is expected to confirm from the user if he really want to complete the trip before removing the trip from the current trip to completed trip | As Expected |

|  |
| --- |
| **Exceptions:** Database Error  The user is not logged in  The Admin restrict the functionality from the user’s account |

|  |
| --- |
| **Post-conditions:** The user completed the trip and his dashboard is updated with **all details** |

**Test Case: Manage Notes**

|  |  |
| --- | --- |
| **Test Case #: 42**  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify that the user can manage Notes | **Test Case Name:** Manage Notes |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Utilities Page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user click the link to add a new note to the planned trip or current trip | The system is expected to create a note, ask the user to name it and store the new note to the database | As Expected |
| 2 | The user clicks the link to edit/delete his created note | The system is expected to edit the required note to the new details and store it to the database | As Expected |

|  |
| --- |
| **Exceptions:** Database Error  The user is not logged in  The Admin restrict the functionality from the user’s account |

|  |
| --- |
| **Post-conditions:** The user is able to perform his desired activity either to create, edit or delete his notes |

**Test Case: Manage Reminder**

|  |  |
| --- | --- |
| **Test Case #: 43**  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify that the user can manage Reminder | **Test Case Name:** Manage Reminder |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Utilities Page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user click the link to add a new reminder | The system is expected to add a new reminder and store it on the database | As Expected |
| 2 | The user click the link to add the date, time and reminder text | The system is expected to save to the details to the database and remind the user on the desired time | As Expected |

|  |
| --- |
| **Exceptions:** Database Error  The user is not logged in  The Admin restrict the functionality from the user’s account |

|  |
| --- |
| **Post-conditions:** The user sets a reminder and the reminder notifies the user at the desired time |

**Test Case: Manage checklist**

|  |  |
| --- | --- |
| **Test Case #: 44**  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify that the user can manage Reminder | **Test Case Name:** |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Utilities Page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user click the link to add a new checklist to the planned or current trip | The system is expected a new checklist and store the details when the user is done adding all details | As Expected |
| 2 | The user clicks the link to edit the checklist | The system is expected to allow the user to edit the created check list and store the new details in the database | As expected |

|  |
| --- |
| **Exceptions:** Database Error  The user is not logged in  The Admin restrict the functionality from the user's account |

|  |
| --- |
| **Post-conditions:** The user is able to add, edit and delete his checklist |

**Test Case: View Calendar**

|  |  |
| --- | --- |
| **Test Case #: 45**  **System:** Tourist Assist App  **Designed by:** Ericson Akatakpo  **Short Description:** This test is to verify that the user can View the Calendar | **Test Case Name:** View Calendar |

|  |
| --- |
| **Pre-conditions:** The user must be logged in with a valid Username and password and must be on the Utilities Page |

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Expected System Response** | **Actual Result** |
| 1 | The user clicks the link to view the calendar events | The system is expected to show all details of the planned trip to the user | As Expected |
| 2 | The user clicks on any trip for more information | The system is expected to show more information to the user | As Expected |

|  |
| --- |
| **Exceptions:** Database Error  The user is not logged in  The Admin restrict the functionality from the user’s account |

|  |
| --- |
| **Post-conditions:** The user is able to view his calendar and details about his trips |