**CSE 6324: Advance Software Engineering**

**Project Name: Trade\_BNB**

**Team-1**

**Project Name:** Trade\_bnb

**Test Case Name: SP01 -** FrontEnd

**Test Case Id**: CSE6324/SP2022/Team1/FrontEnd

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case No.** | **Test Case Description -**  **Functional** | **Expected results** | **Outcome**  **Pass, Fail, Other (comments)** |
| TC1 | Verify if a user will be able to login with a valid username and valid password. | Successful login/ arrive on dashboard Page |  |
| TC2 | Verify if a user cannot login with a valid username and an invalid password. | Unsuccessful login/ stay on login page |  |
| TC3 | Verify the login page for both, when the field is blank and Submit button is clicked. | Unsuccessful login/ stay on login page |  |
| TC4 | Verify the ‘Forgot Password’ functionality. - To be able to view the forgot password utility-page | Successful at showing the page on click |  |
| TC5 | Verify the messages for invalid login. | To Show a warning message on invalid login |  |
| TC6 | Verify if the data in password field is either visible as asterisk or bullet signs. | Successful - at showing masked password fields using bullet signs |  |
| TC7 | Verify fields for user input on register page | Successful at verifying details |  |
| TC8 | Verify user registration through mail followup | Email link sent to the user |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case No.** | **Test Case Description -**  **Non-Functional** | **Expected results** | **Outcome**  **Pass, Fail, Other (comments)** |
| TC9 | Verify the login page by pressing ‘Back button’ of the browser. It should not allow you to enter into the system once you log out. | Successful at nullifying the token to restrict browser local data storage persistence |  |
| TC10 | Verify the Login page against SQL injection attack. | Successful at deflecting to login page |  |
| TC11 | Verify the timeout functionality of the login session. | 10 min to time out from a session |  |
| TC12 | Verify if the same user cannot be registered twice/duplication prevention | Successful at showing error message for user exists. |  |
| TC13 | System Design finalized |  |  |
| TC14 | Data Islands created |  |  |

**CSE 6324: Advance Software Engineering**

**Project Name: Trade\_BNB**

**Team-1**

**Project Name: TradeBnb**

**Test Case Name: SP01-Middleware**

**Test Case Id: CSE6324/SP2022/Team1/Middleware**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case No.** | **Test Case Description** | **Expected results** | **Outcome**  **Pass, Fail, Other (comments)** |
| TC1 | Post Call: Registering with a valid username, email id and password | API should return success status (200) and a password token should be generated. |  |
| TC2 | Post Call: Registering with a duplicate username | API should return success status (200) and a password token should be generated. |  |
| TC3 | Post Call: Registering with a duplicate email id | API should return a validation error status and registration should fail. |  |
| TC4 | Post Call: Registering with an invalid username or password | API should return a validation error status (shorter than minimum required length) and registration should fail. |  |
| TC5 | Post Call: Registering with an invalid email id. | API should return a validation error status (provide a valid email) and registration should fail. |  |
| TC6 | Post Call: Login of a successfully registered user | API should return success status (200). |  |
| TC7 | Post Call: Login of an unregistered user | API should return a validation error status (invalid username or password) and user is not allowed to login. |  |
| TC8 | Post Call: Login with an invalid password | API should return a validation error status (password does not match) and user is not allowed to login. |  |
| TC9 | Post Call: Login with an empty username and/or password | API should return a validation error status (provide email and password) and user is not allowed to login. |  |
| TC10 | Get Call: Password token verification | API should return success status (200) and provide access to landing page. |  |
| TC11 | POST Call: send info for forgot password | Should send mail link to user to reset password on given mail. |  |
| TC12 | PUT Call: receive patch for updating password field with new token for the given user | Should update the password in the Db along with new token |  |

**CSE 6324: Advance Software Engineering**

**Project Name: Trade\_BNB**

**Team-1**

**Project Name: TradeBnb**

**Test Case Name:**   **SP01- DataBase**

**Test Case Id: CSE6324/SP2022/Team1/DataBase**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case No.** | **Test Case Description** | **Expected results** | **Outcome**  **Pass, Fail, Other (comments)** |
| TC1 | To check whether the mapping of the database fields and columns in the backend is compatible with those mappings in the front-end. | The key mapping of frontend should be in sync with backend |  |
| TC2 | To check if access is given to registered users only | Access should be given if the credentials entered are correct |  |
| TC3 | To check if access is denied to users using incorrect credential | Access should be revoked if the credentials entered are incorrect |  |
| TC4 | To check if any anomalous data is stored for username field in database | No anomalous characters are stored in username attribute |  |
| TC5 | To check if any anomalous data is stored for password field in database | No anomalous characters are stored in password attribute |  |
| TC6 | To check if any anomalous data is stored for email field in database | No anomalous characters are stored in email attribute |  |
| TC7 | To check if updates are being made correctly (verifying concurrency control, dirty read and phantom updates in db) | Each row is unique for a given user and there are no duplicate fields |  |