**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 follow-up | Email link sent to the user |  |
| TC9 | Verify the login page by pressing ‘Back button’ of the browser. It should not allow you to enter 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 |  |  |

**Project Name:** Trade\_BNB

**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)** |
| TC15 | Post Call: Registering with a valid username, email id and password | API should return success status (200) and a password token should be generated. |  |
| TC16 | Post Call: Registering with a duplicate username | API should return success status (200) and a password token should be generated. |  |
| TC17 | Post Call: Registering with a duplicate email id | API should return a validation error status and registration should fail. |  |
| TC18 | 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. |  |
| TC20 | Post Call: Registering with an invalid email id. | API should return a validation error status (provide a valid email) and registration should fail. |  |
| TC21 | Post Call: Login of a successfully registered user | API should return success status (200). |  |
| TC22 | 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. |  |
| TC23 | 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. |  |
| TC24 | 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. |  |
| TC25 | Get Call: Password token verification | API should return success status (200) and provide access to landing page. |  |
| TC26 | POST Call: send info for forgot password | Should send mail link to user to reset password on given mail. |  |
| TC27 | 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 |  |

**Project Name:** Trade\_BNB

**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)** |
| TC28 | 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 |  |
| TC29 | To check if access is given to registered users only | Access should be given if the credentials entered are correct |  |
| TC30 | To check if access is denied to users using incorrect credential | Access should be revoked if the credentials entered are incorrect |  |
| TC31 | To check if any anomalous data is stored for username field in database | No anomalous characters are stored in username attribute |  |
| TC32 | To check if any anomalous data is stored for password field in database | No anomalous characters are stored in password attribute |  |
| TC33 | To check if any anomalous data is stored for email field in database | No anomalous characters are stored in email attribute |  |
| TC34 | 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 |  |

**Project Name:** Trade\_bnb

**Test Case Name:** SP02 - Frontend

**Test Case Id**: CSE6324/SP2022/Team0/Login-Frontend

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case No.** | **Test Case Description -**  **Functional** | **Expected results** | **Outcome**  **Pass, Fail, Other (comments)** |
| TC35 | Verification of the users for the login and register. No user with incorrect username or passwords should be able to login and register. | Verification will be Successful. |  |
| TC36 | Validation of the users for the login. No junk or anomalous characters username or passwords should be generated. | Validation will be Successful. |  |
| TC37 | Given a user can login and arrive on the dashboard, the user is granted access to features of the application based on privileges defined by the role. | Users will be logged in according to their given roles. |  |
| TC38 | A user identified as ADMIN has access to user activity for (users defined as INVESTOR, TRADER) | Admin can access users' activity successfully. |  |
| TC39 | A user defined as ADMIN cannot TRADE or INVEST. | Admin cannot do trading or investing. |  |
| TC40 | A user identified as INVESTOR cannot perform ADMIN activities (e.g., extending, or nullifying margin call) | Investor cannot access any user's activity with respect to trader/admin. |  |
| TC41 | A user identified as TRADER cannot perform ADMIN activities (e.g., extending, or nullifying margin call) | Traders cannot access any user's activity with respect to trader/admin. |  |
| TC42 | A user identified as INVESTOR cannot perform TRADER activities. | Investors see errors while trying to go to the trader's page. |  |
| TC43 | A user identified as TRADER can perform INVESTORS activities | Traders can trade and invest. |  |

**Project Name:** Trade\_BNB

**Test Case Name:** SP02 - Middleware

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case No.** | **Test Case Description** | **Expected results** | **Outcome**  **Pass, Fail, Other (comments)** |
| TC44 | Post Call: Registering with a valid username, email id and password and role. | API should return success status (200) and a password token should be generated. |  |
| TC45 | Empty roles data is a valid data | API should return success status (200) and a password token should be generated. |  |
| TC46 | Non-Empty roles is not a valid data object and is handled. | API should not pass any user role from front end as the roles are determined after verification of requests. |  |
| TC47 | While making GET call for the same user once the registration is complete the updated role should be reflected. | The data obtained for each user now has an additional field identifying a valid role (ADMIN, INVEST & TRADE), |  |
| TC48 | The get call received should identify the user either as: ADMIN, INVEST or TRADE | API should return a validation error status (provide a valid email) and registration should fail. |  |
| TC49 | The get call received having Unidentified/empty users are not allowed to land on the dashboard. | API should return success status (200).  But Users with empty roles are not allowed access to dashboard. (Identifier for frontend) |  |
| TC50 | Each user is identified uniquely using email and a Token encryption which is valid for session length. | The user was successfully assigned a token based on JWT token service |  |

**Project Name:** Trade\_BNB

**Test Case Name:** SP02 - Database

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case No.** | **Test Case Description** | **Expected results** | **Outcome**  **Pass, Fail, Other (comments)** |
| TC51 | Verify passwords stored in the database are encrypted. | Retrieved passwords in the database are encrypted. |  |
| TC52 | Verify that the database schema contains a field called roles. | Fetched data should indicate roles field for each user. |  |
| TC53 | Verify that roles field has no data by default on each user creation. | For regular user roles field should be empty. |  |
| TC54 | Verify that we can modify the roles field to admin after receiving a go ahead from business team. | Admin role should be assigned properly. |  |
| TC55 | Verify that the admin role can be revoked successfully after receiving a go ahead from business team. | Admin role should be removed successfully upon request. |  |

**Project Name:** Trade\_bnb

**Test Case Name:** SP3 - Homepage

**Test Case Id**: CSE6324/SP2022/Team1/Login-Frontend

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case No.** | **Test Case Description -**  **Functional** | **Expected results** | **Outcome**  **Pass, Fail, Other (comments)** |
| TC56 | Customer can now be able to login with login button which will show to login pop-up. | Login Button is implemented successfully |  |
| TC57 | Customer can now be able logout by one click on logout icon. | Logout Button is implemented successfully |  |
| TC58 | In Homepage, Trading privilege shows different options in tiles. **stocks** will be shown, and the link will be provided for more details. | Stocks tile displayed successfully. |  |
| TC59 | In Homepage, Trading privilege shows different options in tiles. **options** will be shown. | Options tile displayed successfully. |  |
| TC60 | In Homepage, Trading privilege shows different options in tiles. **ETF** will be shown. | ETF tile displayed successfully. |  |
| TC61 | In Homepage, Trading privilege shows different options in tiles. **Crypto** will be shown. | Crypto tile displayed successfully. |  |
| TC62 | In Account types, information about IRA will be displayed along with the image. | IRA information displayed successfully. |  |
| TC63 | In Account types, information about IBA will be displayed along with the image. | IBA information displayed successfully. |  |
| TC64 | Open account will be a placeholder given for opening the trade BNB individual brokerage and IRA accounts. | Open account placeholder displayed successfully. |  |

**Project Name:** Trade\_BNB

**Test Case Name:** SP3 - Market Watch / Portfolio

**Test Case Id**: CSE6324/SP2022/Team1/Market Watch

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case No.** | **Test Case Description -**  **Functional** | **Expected results** | **Outcome**  **Pass, Fail, Other (comments)** |
| TC65 | To check if premarket data, current and historical data is available | Data for Premarket session, current session historical data is available |  |
| TC66 | To check if Symbol, name, price, after hours, change, % change and open price is available | Data is available for the current page |  |
| TC67 | To check if color code is used to display gain / loss | Gain should be displayed using green color and loss should be displayed using red color |  |
| TC68 | To check if search functionality is working | Data for the searched symbol should be available |  |
| TC69 | To check if search box gives an error if invalid input is given | It should display an error saying no result found |  |
| TC70 | To check on hover Buy / Sell / Delete option is visible | It should be able to buy / sell / delete the selected stock |  |
| TC71 | To check if Open, High, Low, Close, Prev. Close price is available of the selected stock | It should display the Open, High, Low, Close, Prev. Close price of the selected page |  |
| TC72 | To check if Traded Volume, Traded Quantity is available for the selected stock | It should display the Traded Volume, Traded Quantity of the selected page |  |

**Project Name:** Trade\_BNB

**Test Case Name:** SP03 - Dashboard

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case No.** | **Test Case Description -**  **Functional** | **Expected results** | **Outcome**  **Pass, Fail, Other (comments)** |
| TC73 | Customer should be able to see configuration page chart for stocks | Configuration page chart displayed successfully |  |
| TC74 | Customer should be able to see advance and decline chart. | Advance and decline chart displayed successfully |  |
| TC75 | Customer should be able to see broader market chart. | Broader market chart displayed successfully |  |
| TC76 | Customer should be able to see data grid chart with live news. | Data grid chart and live news displayed successfully |  |
| TC77 | Customer should be able to see the rows per pages. | Row per pages displayed successfully |  |
| TC78 | Customer should be able to see 5 rows – ID, source name, provider name, breaking news and publish data. | 5 rows displayed successfully |  |
| TC79 | Customer should be able to see live news sorted by timestamp | Live news with sorted timestamp displayed successfully |  |