|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Scenario: | Description: | Values: | Expected Outcome: | Actual Outcome: |
| 1.0 | Accessing the ‘Home’ page or Main page | Home page url: 127.0.0.1:8000/home/ | Goes to the home Page | Goes to the home Page |
| 1.1 | Test whether ‘Home’ page contains Resume | Home page url:  127.0.0.1:8000/home/ | Goes to home page and resume shown | Goes to home page and resume shown |
| 1.2 | Test whether ‘Home’ page contains Self intro | Home page url:  127.0.0.1:8000/home/ | Goes to home page and Self Introduction shown | Goes to home page and Self Introduction shown |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Scenario: | Description: | Values: | Expected Outcome: | Actual Outcome: |
| 2.0 | Access ‘Admin’ page when not logged in. | Admin page url:  <http://127.0.0.1:8000/admin/> | Bring user to login page | Bring user to login page |
| 2.1 | Logging in with invalid credentials to see if it remains in ‘Log In’ page | Login url: [http://127.0.0.1:8000/admin/login/ ?next=/admin/](http://127.0.0.1:8000/admin/login/%20?next=/admin/)  Username: testing  Password: testing | Remains on same page. | Remain in same page. |
| 2.2 | Logging in without entering any user credentials to see if it remains in ‘Log in’ page | Login url: [http://127.0.0.1:8000/admin/login/ ?next=/admin/](http://127.0.0.1:8000/admin/login/%20?next=/admin/)  Username: “”  Password: “” | Remain in same page | Remain in same page |
| 2.3 | Logging in with only valid username to see if it remains in ‘Log in’ page | Login url: [http://127.0.0.1:8000/admin/login/ ?next=/admin/](http://127.0.0.1:8000/admin/login/%20?next=/admin/)  Username: “admin”  Password: “” | Remain in same page | Remain in same page |
| 2.4 | Logging in with only the password to see if it remains in ‘Log in’ page | Login url: [http://127.0.0.1:8000/admin/login/ ?next=/admin/](http://127.0.0.1:8000/admin/login/%20?next=/admin/)  Username: “”  Password: “password” | Remain in same page | Remain in same page |
| 2.5 | Redirection to ‘Admin’ page after logging in | Admin page url:  <http://127.0.0.1:8000/admin/>  username: Eshwar  password: Eshwar123456 | Redirected to admin page after logging in. | Redirected to admin page after logging in. |

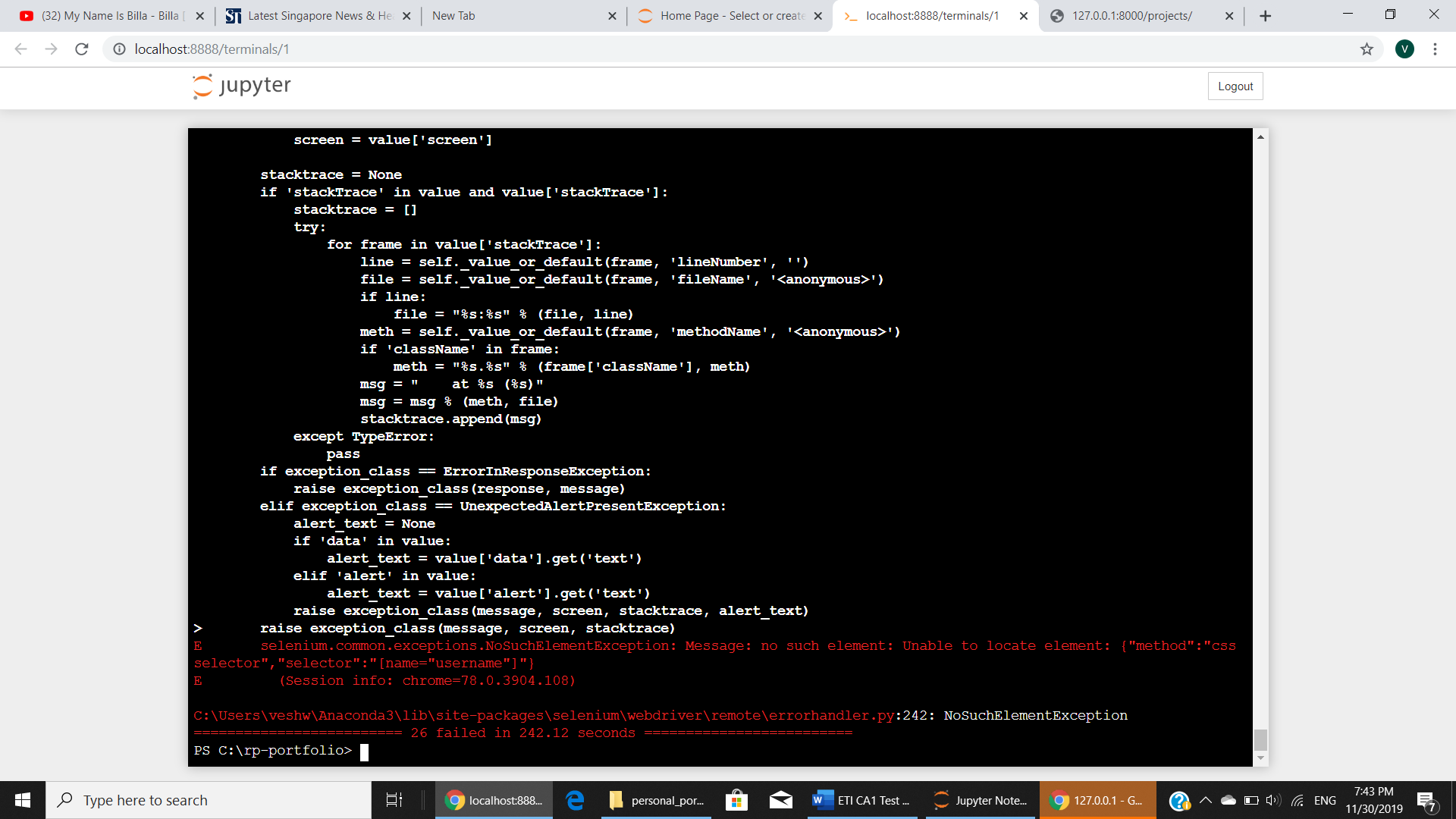
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Scenario: | Description: | Values: | Expected Outcome: | Actual Outcome: |
| 3.0 | Test whether able to view list of created users | Users page url:  <http://127.0.0.1:8000/admin/auth/user/> | Directed to page with list of created users | Directed to page with list of created users |
| 3.1 | Test whether able to select user whose username is ‘admin’ and click to view more info | Users page url:  <http://127.0.0.1:8000/admin/auth/user/>  user = ‘Eshwar’ | Went into page ‘admin’ user page  Username of user is ‘Eshwar’ | Went into page ‘admin’ user page  Username of user is ‘Eshwar’ |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Scenario: | Description: | Values: | Expected Outcome: | Actual Outcome |
| 4.0 | Test to be able to create a new ‘User’ with empty fields | User creation page url:  <http://127.0.0.1:8000/admin/auth/user/add/>  new username: “”  new password: “”  new confirm password: “” | User is not created and remain in same page  Error message shown = “Please correct the errors below.” & “This field is required.” | User is not created and remain in same page  Error message shown = “Please correct the errors below.” & “This field is required.” |
| 4.1 | Test to be able to create ‘User’ with an already existing username ‘admin’ and empty password fields | User creation page url:  <http://127.0.0.1:8000/admin/auth/user/add/>  new username: “Eshwar”  new password: “”  new confirm password: “” | User is not created and remain in same page  Error message shown = “Please correct the errors below.”, “A user with that username already exists.” & 'This field is required.' | User is not created and remain in same page  Error message shown = “Please correct the errors below.”, “A user with that username already exists.” & 'This field is required.' |
| 4.2 | Test to able to create ‘User’ with a password similar to the username entered | User creation page url:  <http://127.0.0.1:8000/admin/auth/user/add/>  new username: “eshwar”  new password: “Eshwar7272”  new confirm password: “Eshwar7272” | User is not created and remain in same page  Error message shown = “Please correct the error below.”  & ‘The password is too similar to the username.’ | User is not created and remain in same page  Error message shown = “Please correct the error below.”  & ‘The password is too similar to the username.’ |
| 4.3 | Test to be able to create ‘User’ with a password less than 8 characters | User creation page url:  <http://127.0.0.1:8000/admin/auth/user/add/>  new username: “eshwar”  new password: “passwor”  new confirm password: “passwor” | User is not created and remain in same page  Error message shown = “Please correct the error below.”  & ‘This password is too short. It must contain at least 8 characters.’ | Error message shown = “Please correct the error below.”  & ‘This password is too short. It must contain at least 8 characters.’ |
| 4.4 | Test to be able to create ‘User’ with numeric password | User creation page url:  <http://127.0.0.1:8000/admin/auth/user/add/>  new username: “eshwar”  new password: “123456789”  new confirm password: “123456789” | User is not created and remain in same page  Error message shown = “Please correct the error below.”  & ‘This password is too common.’ | User is not created and remain in same page  Error message shown = “Please correct the error below.”  & ‘This password is entirely numeric.’ |
| 4.5 | Test whether able to create a ‘User’ with a commonly used password | User creation page url:  <http://127.0.0.1:8000/admin/auth/user/add/>  new username: “eshwar”  new password: “password”  new confirm password: “password” | User is not created and remain in same page  Error message shown = “Please correct the error below.”  & ‘This password is too common.’ | A common password is very subjective hence it’s hard to test to determine what is commonly used or not. |
| 4.6 | Test to be able to create ‘User’ when password and confirm password are different | User creation page url:  <http://127.0.0.1:8000/admin/auth/user/add/>  new username: eshwar123  new password: Str0ngP@ww0rd  new confirm password: Str0ngP@ww0rd123 | User is not created and remain in same page  Error message shown = “Please correct the error below.”  & ‘The two password fields didn't match.’ | User is not created and remain in same page  Error message shown = “Please correct the error below.”  & ‘The two password fields didn't match.’ |
| 4.7 | Test to be able to create ‘User’ when username field is empty | User creation page url:  <http://127.0.0.1:8000/admin/auth/user/add/>  new username: “”  new password: Str0ngP@ww0rd1234  new confirm password: Str0ngP@ww0rd1234 | User is not created and remain in same page  Error message shown = “Please correct the error below.”  & ‘This field is required.’ | User is not created and remain in same page  Error message shown = “Please correct the error below.”  & ‘This field is required.’ |
| 4.8 | Test whether able to create ‘User’ when one of the password fields is empty | User creation page url:  <http://127.0.0.1:8000/admin/auth/user/add/>  new username: “testing2k00”  new password: Str0ngP@ww0rd  new confirm password: | User not created and remain in same page  Error message shown = “Please correct the error below.”  & ‘This field is required.’ | User not created and remain in same page  Error message shown = “Please correct the error below.”  & ‘This field is required.’ |
| 4.9 | Test whether able to create ‘User’ with valid credentials | User creation page url:  <http://127.0.0.1:8000/admin/auth/user/add/>  new username: testing2k00  new password: Str0nkP@ssw0rd  new confirm password: Str0nkP@ssw0rd | New user created  Number of row of users increased by 1. | New user created  Number of row of users increased by 1. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Scenario: | Description: | Values: | Expected Outcome: | Actual Outcome: |
| 5.0 | Test to be able to create ‘Category’ with empty fields | Category creation page url:  <http://127.0.0.1:8000/admin/blog/category/add/>  Category: “” | Category is not created, and it remains in the same page  Error message shown = “Please correct the error below.” & “This field is required.” | Category is not created, and it remains in the same page  Error message shown = “Please correct the error below.” & “This field is required.” |
| 5.1 | Test to be able to create ‘Category’ | Admin page url:  <http://127.0.0.1:8000/admin/>  category: CCA | Category created and redirected to Categories page  Number of row of Categories increased by 1. | Category created and redirected to Categories page  Number of row of Categories increased by 1. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Scenario: | Description: | Values: | Expected Outcome: | Actual Outcome: |
| 6.0 | Test whether able to create ‘Post’ with empty fields | Blog creation page url:  <http://127.0.0.1:8000/admin/blog/post/add/>  Title: “”  Body: “”  Categories: “” | Post is not created, and it remains in same page  Error message shown = “Please correct the error below.” & “This field is required.” | Post is not created, and it remains in same page  Error message shown = “Please correct the error below.” & “This field is required.” |
| 6.1 | Test to be able to create ‘Post’ when Body and categories is empty | Blog creation page url:  <http://127.0.0.1:8000/admin/blog/post/add/>  Title: “Test”  Body: “”  Categories: “” | Post is not created and remains in same page  Error message shown = “Please correct the error below.” & “This field is required.” | Post is not created and remains in same page  Error message shown = “Please correct the error below.” & “This field is required.” |
| 6.2 | Test to be able to create ‘Post’ when categories is empty | Blog creation page url:  <http://127.0.0.1:8000/admin/blog/post/add/>  Title: “Test”  Body: “Test”  Categories: “” | Post is not created and remains in same page  Error message shown = “Please correct the error below.” & “This field is required.” | Post is not created and remains in same page  Error message shown = “Please correct the error below.” & “This field is required.” |
| 6.3 | Test to be able to create ‘Post’ when Title is empty | Blog creation page url:  <http://127.0.0.1:8000/admin/blog/post/add/>  Title: “”  Body: “Test”  Categories: “1” | Post not created and remain in same page  Error message shown = “Please correct the error below.” & “This field is required.”  Page title = ‘Add post | Jw Portfolio Portal’ | Post not created and remain in same page  Error message shown = “Please correct the error below.” & “This field is required.”  Page title = ‘Add post | Jw Portfolio Portal’ |
| 6.4 | Test to be able to create a new unique ‘Post’ | Post page url: <http://127.0.0.1:8000/admin/blog/post/>  Title: “Track & Field”  Body: “I did …”  Categories: “1” | Post created and redirected to Post page. | Post created and redirected to Post page. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Scenario: | Description: | Values: | Expected Outcome: | Actual Outcome: |
| 7.0 | Test to be able to access ‘Blog’ page | Blog page url:  <http://127.0.0.1:8000/blog/> | Goes to the blog Page | Goes to the blog Page |
| 7.1 | Test to be able to access ‘Blog’ categories ‘Projects’ | Blog page categories url:  [http://127.0.0.1:8000/blog/Proje cts/](http://127.0.0.1:8000/blog/Proje%20%20cts/) | Goes to the Blog Categories page | Goes to the Blog Categories page |



The tests failed due to unforeseen circumstances which is heavily attributed to errors in methods and the structure in coding the tests cases mentioned above. From this, further redevelopment would be needed on the website and the coding to ensure that these tests will pass.