Roomit – tests summary page

**Unit Tests:**

User Module –

|  |  |  |
| --- | --- | --- |
| Test Method Name | Title | Description |
| test\_change\_status\_get\_insert\_to\_enter | Change Status - Get (Insert to Enter) | Tests the behavior of changing the user's status from "Insert" to "Enter" using a GET request. |
| test\_change\_status\_get\_enter\_to\_insert | Change Status - Get (Enter to Insert) | Tests the behavior of changing the user's status from "Enter" to "Insert" using a GET request. |
| test\_display\_property\_offer\_get | Display Property Offer - GET | Tests the behavior of retrieving the property offer display page using a GET request. |
| test\_display\_property\_offer\_post\_valid\_form | Display Property Offer - POST (Valid Form) | Tests the behavior of submitting a valid form for the property offer display page using a POST request. |
| test\_display\_property\_offer\_post\_invalid\_form | Display Property Offer - POST (Invalid Form) | Tests the behavior of submitting an invalid form for the property offer display page using a POST request. |
| test\_display\_property\_reqs\_get | Display Property Reqs - GET | Tests the behavior of retrieving the property requirements display page using a GET request. |
| test\_display\_property\_reqs\_post\_invalid\_form | Display Property Reqs - POST (Invalid Form) | Tests the behavior of submitting an invalid form for the property requirements display page using a POST request. |
| test\_info\_get\_user\_logged | Info - GET (User Logged) | Tests the behavior of retrieving the user's info page when the user is logged in. |
| test\_info\_get\_user\_unlogged | Info - GET (User Unlogged) | Tests the behavior of retrieving the user's info page when the user is unlogged. |
| test\_info\_post\_valid\_form | Info - POST (Valid Form) | Tests the behavior of submitting a valid form on the user's info page using a POST request. |
| test\_info\_post\_invalid\_form | Info - POST (Invalid Form) | Tests the behavior of submitting an invalid form on the user's info page using a POST request. |
| test\_profile\_get\_user\_logged | Profile - GET (User Logged) | Tests the behavior of retrieving the user's profile page when the user is logged in. |
| test\_profile\_get\_user\_unlogged | Profile - GET (User Unlogged) | Tests the behavior of retrieving the user's profile page when the user is unlogged. |
| test\_profile\_get\_read\_only | Profile - GET (Read-Only) | Tests the behavior of retrieving the profile page of another user in read-only mode. |
| test\_profile\_post\_valid\_form | Profile - POST (Valid Form) | Tests the behavior of submitting a valid form on the user's profile page using a POST request. |
| test\_info\_post\_invalid\_form | Profile - POST (Invalid Form) | Tests the behavior of submitting an invalid form on the user's profile page using a POST request. |
| test\_correct | Register - Correct Credentials | Tests the authentication of a user with correct credentials. |
| test\_wrong\_username | Register - Incorrect Username | Tests the authentication of a user with an incorrect username. |
| test\_wrong\_password | Register - Incorrect Password | Tests the authentication of a user with an incorrect password. |
| test\_set\_status\_get\_redirect\_insert | Set Status - Get (Redirect Insert) | Tests the behavior of redirecting the user to the "Insert" page when accessing the status page with the status set to "Insert". |
| test\_set\_status\_get\_redirect\_enter | Set Status - Get (Redirect Enter) | Tests the behavior of redirecting the user to the "Enter" page when accessing the status page with the status set to "Enter". |

roomit\_app module –

|  |  |  |
| --- | --- | --- |
| Method Name | Title | Description |
| test\_calculate\_score | Calculate Score Test | Tests the calculation of a score based on certain properties and requirements. |
| test\_update\_scores\_only\_creates\_new\_scores | Update Scores Test | Tests that updating scores creates new scores. |
| test\_update\_scores\_correctly\_calculates\_enter\_scores | Update Scores Enter Test | Tests that updating scores correctly calculates enter scores. |
| test\_update\_scores\_correctly\_calculates\_insert\_scores | Update Scores Insert Test | Tests that updating scores correctly calculates insert scores. |
| test\_update\_scores\_enter | Update Scores Enter Test | Tests the update of scores for the "enter" profile. |
| test\_update\_scores\_insert | Update Scores Insert Test | Tests the update of scores for the "insert" profile. |
| test\_make\_requirementsP | Make RequirementsP Test | Tests the creation of requirementsP object. |
| test\_make\_requirementsR | Make RequirementsR Test | Tests the creation of requirementsR object. |
| test\_requirementsP\_authenticated | Authenticated requirementsP Test | Tests the requirementsP view for an authenticated user. |
| test\_requirementsP\_unauthenticated | Unauthenticated requirementsP Test | Tests the requirementsP view for an unauthenticated user. |
| test\_requirementsR\_authenticated | Authenticated requirementsR Test | Tests the requirementsR view for an authenticated user. |
| test\_requirementsR\_unauthenticated | Unauthenticated requirementsR Test | Tests the requirementsR view for an unauthenticated user. |
| test\_likes\_me\_authenticated | Authenticated likes\_me Test | Tests the likes\_me view for an authenticated user. |
| test\_likes\_me\_unauthenticated | Unauthenticated likes\_me Test | Tests the likes\_me view for an unauthenticated user. |
| test\_i\_like\_authenticated | Authenticated i\_like Test | Tests the i\_like view for an authenticated user. |
| test\_i\_like\_unauthenticated | Unauthenticated i\_like Test | Tests the i\_like view for an unauthenticated user. |
| test\_more\_authenticated\_insert | Authenticated more (insert) Test | Tests the more view for an authenticated user with profile status "insert". |
| test\_more\_authenticated\_enter | Authenticated more (enter) Test | Tests the more view for an authenticated user with profile status "enter". |
| test\_more\_unauthenticated | Unauthenticated more Test | Tests the more view for an unauthenticated user. |
| test\_post\_list\_authenticated\_insert | Authenticated post\_list (insert) Test | Tests the post\_list view for an authenticated user with profile status "insert". |
| test\_post\_list\_authenticated\_enter | Authenticated post\_list (enter) Test | Tests the post\_list view for an authenticated user with profile status "enter". |

Requirements:

* **Y/N requirements –**

|  |  |  |
| --- | --- | --- |
| Test Method Name | Title | Description |
| test\_calculate\_score\_desired\_answer\_yes | Calculate Score - Desired Answer: Yes | Tests the calculation of the score when the desired answer is "Yes". |
| test\_calculate\_score\_desired\_answer\_no | Calculate Score - Desired Answer: No | Tests the calculation of the score when the desired answer is "No". |
| test\_calculate\_score\_no\_desired\_answer | Calculate Score - No Desired Answer | Tests the calculation of the score when there is no desired answer. |
| test\_calculate\_score\_answer\_yes\_not\_desired\_answer | Calculate Score - Answer: Yes, Not Desired Answer | Tests the calculation of the score when the answer is "Yes" but not the desired answer. |
| test\_calculate\_score\_answer\_no\_not\_desired\_answer | Calculate Score - Answer: No, Not Desired Answer | Tests the calculation of the score when the answer is "No" but not the desired answer. |
| test\_calculate\_score\_answer\_empty\_string | Calculate Score - Answer: Empty String | Tests the calculation of the score when the answer is an empty string. |
| test\_calculate\_score\_answer\_false | Calculate Score - Answer: False | Tests the calculation of the score when the answer is False. |
| test\_calculate\_score\_answer\_one | Calculate Score - Answer: 1 | Tests the calculation of the score when the answer is 1. |
| test\_calculate\_score\_answer\_empty\_list | Calculate Score - Answer: Empty List | Tests the calculation of the score when the answer is an empty list. |
| test\_calculate\_score\_answer\_non\_empty\_list | Calculate Score - Answer: Non-Empty List | Tests the calculation of the score when the answer is a non-empty list. |
| test\_calculate\_score\_with\_desired\_answer | Calculate Score - With Desired Answer | Tests the calculation of the score with a desired answer specified. |
| test\_calculate\_score\_without\_desired\_answer | Calculate Score - Without Desired Answer | Tests the calculation of the score without a desired answer specified. |
| test\_calculate\_score\_with\_non\_bool\_answer | Calculate Score - With Non-Boolean Answer | Tests the calculation of the score with a non-boolean answer. |
| test\_desired\_answer\_none | Desired Answer: None | Tests the desired answer as **None**. |
| test\_matching\_bool\_answer | Matching Boolean Answer | Tests the calculation of the score with a matching boolean answer. |
| test\_non\_matching\_bool\_answer | Non-Matching Boolean Answer | Tests the calculation of the score with a non-matching boolean answer. |
| test\_answer\_not\_bool | Answer: Not Boolean | Tests the calculation of the score with a non-boolean answer. |

* **Range requirements –**

|  |  |  |
| --- | --- | --- |
| **Test Method Name** | Title | Description |
| test\_calculate\_score\_when\_min\_and\_max\_are\_none | Calculate Score - Min and Max: None | Tests the calculation of the score when both the minimum and maximum values are None. |
| test\_calculate\_score\_when\_max\_is\_none | Calculate Score - Max: None | Tests the calculation of the score when the maximum value is None. |
| test\_calculate\_score\_when\_answer\_is\_none | Calculate Score - Answer: None | Tests the calculation of the score when the answer is None. |
| test\_calculate\_score\_when\_answer\_is\_not\_an\_int | Calculate Score - Answer: Not an Integer | Tests the calculation of the score when the answer is not an integer. |
| test\_calculate\_score\_when\_answer\_is\_within\_range | Calculate Score - Answer: Within Range | Tests the calculation of the score when the answer is within the specified range. |
| test\_calculate\_score\_when\_answer\_is\_outside\_  range\_and\_outside\_tolerance | Calculate Score - Answer: Outside Range and Tolerance | Tests the calculation of the score when the answer is outside the specified range and tolerance. |
| test\_both\_min\_max\_none | Min and Max: None | Tests the scenario when both the minimum and maximum values are none. |
| test\_answer\_not\_int | Answer: Not an Integer | Tests the scenario when the answer is not an integer. |
| test\_answer\_negative\_number | Answer: Negative Number | Tests the scenario when the answer is a negative number. |
| test\_answer\_outside\_range | Answer: Outside Range | Tests the scenario when the answer is outside the specified range. |
| test\_answer\_equal\_to\_min\_max | Answer: Equal to Min/Max | Tests the scenario when the answer is equal to the minimum or maximum value. |
| test\_answer\_is\_none | Answer: None | Tests the scenario when the answer is None. |
| test\_answer\_is\_not\_a\_number | Answer: Not a Number | Tests the scenario when the answer is not a number. |
| test\_answer\_is\_negative | Answer: Negative | Tests the scenario when the answer is a negative number. |
| test\_answer\_is\_greater\_than\_max | Answer: Greater than Max | Tests the scenario when the answer is greater than the maximum value. |
| test\_answer\_is\_less\_than\_min | Answer: Less than Min | Tests the scenario when the answer is less than the minimum value. |
| test\_answer\_is\_equal\_to\_min | Answer: Equal to Min | Tests the scenario when the answer is equal to the minimum value. |
| test\_answer\_is\_equal\_to\_max | Answer: Equal to Max | Tests the scenario when the answer is equal to the maximum value. |
| test\_answer\_is\_in\_range | Answer: In Range | Tests the scenario when the answer is within the specified range. |
| test\_answer\_birthdate\_is\_out\_of\_range | Answer: Birthdate out of Range | Tests the scenario when the answer (birthdate) is outside the specified range. |
| test\_answer\_birthdate\_is\_on\_range | Answer: Birthdate on Range | Tests the scenario when the answer (birthdate) is on the specified range. |
| test\_answer\_birthdate\_is\_in\_range | Answer: Birthdate in Range | Tests the scenario when the answer (birthdate) is within the specified range. |
| test\_answer\_is\_not\_birthdate | Answer: Not a Birthdate | Tests the scenario when the answer is not a birthdate. |
| test\_answer\_birthday\_is\_in\_future | Answer: Birthday in Future | Tests the scenario when the answer (birthday) is in the future. |
| test\_non\_datetime\_birthdate | Non-Datetime Birthdate | Tests the scenario when the answer (birthdate) is a non-datetime value. |
| test\_non\_numeric\_min\_or\_max | Non-Numeric Min/Max | Tests the scenario when the minimum or maximum value is non-numeric. |
| test\_non\_numeric\_answer | Non-Numeric Answer | Tests the scenario when the answer is non-numeric. |

* **List requirements –**

|  |  |  |
| --- | --- | --- |
| **Test Method Name** | Title | Description |
| test\_calculate\_score\_desired\_answer\_none | Calculate Score - Desired Answer: None | Tests the calculation of the score when the desired answer is **None**. |
| test\_calculate\_score\_answer\_none | Calculate Score - Answer: None | Tests the calculation of the score when the answer is **None**. |
| test\_calculate\_score\_desired\_answer\_is\_disjoint | Calculate Score - Desired Answer: Disjoint | Tests the calculation of the score when the desired answer is disjoint from the actual answer. |
| test\_calculate\_score\_answer\_is\_subset | Calculate Score - Answer: Subset | Tests the calculation of the score when the actual answer is a subset of the desired answer. |
| test\_calculate\_score\_answer\_is\_superset | Calculate Score - Answer: Superset | Tests the calculation of the score when the actual answer is a superset of the desired answer. |
| test\_calculate\_score\_desired\_answer\_is\_empty | Calculate Score - Desired Answer: Empty | Tests the calculation of the score when the desired answer is an empty set. |
| test\_calculate\_score\_answer\_is\_none | Calculate Score - Answer: None | Tests the calculation of the score when the answer is **None**. |
| test\_calculate\_score\_answer\_does\_not\_  intersect\_desired\_answer | Calculate Score - Answer: No Intersection | Tests the calculation of the score when the answer does not intersect with the desired answer. |
| test\_calculate\_score\_answer\_intersects\_desired\_answer | Calculate Score - Answer: Intersects | Tests the calculation of the score when the answer intersects with the desired answer. |
| test\_calculate\_score\_no\_desired\_answer | Calculate Score - No Desired Answer | Tests the calculation of the score when there is no desired answer. |
| test\_calculate\_score\_desired\_answer\_dont\_care | Calculate Score - Desired Answer: Don't Care | Tests the calculation of the score when the desired answer is "D" (don't care). |
| test\_calculate\_score\_no\_answer | Calculate Score - No Answer | Tests the calculation of the score when there is no answer. |
| test\_calculate\_score\_answer\_in\_desired\_answer | Calculate Score - Answer in Desired Answer | Tests the calculation of the score when the answer is in the desired answer list. |
| test\_calculate\_score\_answer\_not\_in\_desired\_answer | Calculate Score - Answer not in Desired Answer | Tests the calculation of the score when the answer is not in the desired answer list. |
| test\_calculate\_score\_with\_empty\_desired\_answer\_list | Calculate Score - Empty Desired Answer List | Tests the calculation of the score when the desired answer list is empty. |
| test\_calculate\_score\_with\_single\_desired\_answer | Calculate Score - Single Desired Answer | Tests the calculation of the score when there is a single desired answer. |
| test\_calculate\_score\_with\_non\_list\_answer | Calculate Score - Non-List Answer | Tests the calculation of the score when the answer is not a list. |
| test\_calculate\_score\_with\_non\_string\_answer | Calculate Score - Non-String Answer | Tests the calculation of the score when the answer contains non-string elements. |

Security tests –

|  |  |  |
| --- | --- | --- |
| **Test Method Name** | Title | Description |
| test\_unauthorized\_access | Unauthorized Access | Verifies that when no user is logged in and attempts to access the admin page, they are redirected to the login page. |
| test\_invalid\_credentials | Invalid Credentials | Tests the scenario where the admin tries to log in with the wrong password and checks if the correct error message is displayed. |
| test\_password\_reset | Password Reset | Tests the functionality of password reset by sending a request and verifying the appropriate redirection. |
| test\_role\_permissions | Role Permissions | Tests the role-based permissions. Checks if a regular user is redirected to the login page when accessing the admin page and if the admin has access. |
| test\_input\_sanitization | Input Sanitization | Verifies that user input is sanitized to prevent cross-site scripting (XSS) attacks. |

Performance tests –

The **test\_performance** module is designed for performance testing using Locust, where multiple users simulate various tasks on a web application. This module focuses on simulating user registration, filling out user information, and performing tasks related to viewing the homepage and user profiles.

The performance test consists of three tasks:

* **registration:** Simulates user registration to the system and filling out their personal information and requirements.
* **view\_homepage:** Simulates users accessing the homepage of the web application by sending a GET request to the root URL.
* **view\_profile:** Simulates users viewing user profiles by sending a GET request to a randomly selected user's profile page. The username for the profile is chosen from the usernames list, which contains the usernames of all registered users.

The selected tasks for performance testing were intentionally chosen as they represent the heaviest operations in terms of database interactions and computational complexity. By focusing on these tasks, we aim to evaluate the system's performance under demanding scenarios that involve substantial database operations and computational calculations.

By running this performance test with Locust and scaling up the number of users, it becomes possible to measure the web application's performance under load. This module can help identify potential performance bottlenecks, such as slow response times or resource limitations, allowing developers to optimize and improve the application's scalability.