**TASK: Customer Creates Account**

|  |  |
| --- | --- |
| **ID** | **Create account - Customer Creation - POST Test - 1** |
| Purpose | To ensure that a POST request can be sent to the customer database |
| Set Up | The customer enters in the email “test123@wiz” and other attributes required to create the account. |
| Step 1. | The system is expected to send a GET response to the API endpoint using the email (primary key) to check whether the user already exists. |
| Step 2. | If this information is not found, send a POST request in JSON form to insert data into the system. |
|  |  |
| Expected Result | The POST request will register the new customer. |

|  |  |
| --- | --- |
| **ID** | **Reject Duplicates - CC - P - 2** |
| Purpose | Ensure that duplicate data cannot be sent to /customers resource |
| Set Up | A customer exists in the database with email ‘Test@Trude.wiz’, other values for other fields |
| Step 1. | Send a POST request to the /customers resource attempting to write a customer object with email ‘Test@Trude.wiz’, other values for other fields. |
| Step 2. | Output ‘True’ if request is accepted, ‘false’ if rejected |
| Expected Result | Output is ‘false’. |

|  |  |
| --- | --- |
| **ID** | **Reject Blanks - CC - P - 3** |
| Purpose | Ensure that missing data cannot be sent to /customers resource |
| Set Up | The /customer resource exists |
| Step 1. | Send a POST request to the /customers resource attempting to write a customer object with email ‘Test@Trude.wiz’, blank data in either username, password or name fields |
| Step 2. | Output ‘True’ if request is accepted, ‘false’ if rejected |
| Expected Result | Output is ‘false’. |

|  |  |
| --- | --- |
| **ID** | **Concurrent requests - CC - P - 4** |
| Purpose | Ensure that the same requests cannot be sent concurrently. |
| Set Up | Have two sessions enter in the same email (primary key) and the relevant attributes and submit. |
| Step 1. | The system is expected to send a GET response to the API endpoint using the email (primary key) to check whether the user already exists. |
| Step 2. | For the request that arrives first, a POST request in JSON form is inserted into the database, initiating the creation of an account. |
| Step 3. | POST request will return a successful or fail response |
| Step 4. | When the second request arrives, the system will perform a GET request. |
| Step 5. | The outcome of the GET request is displayed onto the page. |
| Step 6. | POST request will return a successful or fail response |
| Expected Result | The system is expected to process the first POST request (successful response)but reject the second (fail response) when it recognises that the same account already exists in the system. |