Sergio Andres Gomez Gonzalez

Part 3: Practical Questions

Write Test Cases UI

Context:

Website https://dashboard-test.zoefin.com/login

Please go to the UI, login as a User of your preference and Find in the menu

the "Add New Client" element, and explore the UI.

Household Member Management

- **Add Members**

Write three test cases for the functionality of adding a new member to a

household.

Test Case ID	CP_001	Test Case Description	Create a new member to a existing HouseHold

Tester's Name	Sergio Gomez	Date Tested	25-Jul-2024	Test Case (Pass/Fail/Not Executed)	Pass

S #	Prerequisites:
1	Logged in as an Advisor, Owner, or PM.
2	

S #	Test Data
1	Email = zoefin.advisor+qaroleadv01@gmail.com
2	Pass = QA.role123

Test Scenario	Create a new member to a existing HouseHold

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Click on the "Add New Client" button.	Add New Client Window should open	As Expected	Pass
2	Select option "Add Client to Existing Household"	Go to select household screen	As Expected	Pass
3	Select Household and click "Next" button	Go to form new member	As Expected	Pass
4	Fill the form and create new member	Create new member successfully	As Expected	Pass
5	Verify new member creation	View New Member in household	As Expected	Pass

Test Case ID	CP_002	Test Case Description	Duplicate Email Prevention	

Tester's Name	Sergio Gomez	Date Tested	25-Jul-2024	Test Case (Pass/Fail/Not Executed)	Pass

S #	Prerequisites:
1	Logged in as an Advisor, Owner, or PM.
2	

S #	Test Data
1	Email = zoefin.advisor+qaroleadv01@gmail.com
2	Pass = QA.role123

Test Scenario	Duplicate Email Prevention

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Click on the "Add New Client" button.	Add New Client Window should open	As Expected	Pass
2	Select option "Add Client to Existing HouseHold"	Go to select household screen	As Expected	Pass
3	Select HouseHold and click "Next" button	Go to form new member	As Expected	Pass
4	Enter an email address that already exists in the system.	An error message is displayed, indicating that the email address is already associated with another member.	As Expected	Pass
5	Attempt to save the member.	An error message is displayed, indicating that the email address is already associated with another member.	As Expected	Pass

Test Case ID	CP_003	Test Case Description	Invalid Mobile Details	

Tester's Name	Sergio Gomez	Date Tested	25-Jul-2024	Test Case (Pass/Fail/Not Executed)	Pass

S #	Prerequisites:
1	Logged in as an Advisor, Owner, or PM.
2	

S #	Test Data
1	Email = zoefin.advisor+qaroleadv01@gmail.com
2	Pass = QA.role123

Test Scenario	Invalid Mobile Details

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Click on the "Add New Client" button.	Add New Client Window should open	As Expected	Pass
2	Select option "Add Client to Existing HouseHold"	Go to select household screen	As Expected	Pass
3	Select HouseHold and click "Next" button	Go to form new member	As Expected	Pass
4	Write a wrong mobile number	An error message is displayed, indicating that the mobile number is wrong.	As Expected	Pass
5	Try to save the member.	An error message is displayed, indicating that the mobile number is wrong.	As Expected	Pass

Edit Members

Write three test cases for the functionality of editing the information of an existing member in a household.

Test Case ID	CP_004	Test Case Description	Valid Member Information Update

Tester's Name	Sergio Gomez	Date Tested	25-Jul-2024	Test Case (Pass/Fail/Not Executed)	Pass

S #	Prerequisites:
1	Logged in as an Advisor, Owner, or PM.
2	Created household member

S #	Test Data
1	Email = zoefin.advisor+qaroleadv01@gmail.com
2	Pass = QA.role123

Test Scenario	Valid Member Information Update

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Go to household member list	HouseHold member list	As Expected	Pass
2	Locate an existing member.	Select member	As Expected	Pass
3	Click on the "Edit" button next to their details.	Go to form edit member	As Expected	Pass
5	Modify any relevant information form and save changes	The member's details are updated successfully	As Expected	Pass

Test Case ID	CP_005	Test Case Description	Update information with duplicate name

Tester's Name	Sergio Gomez	Date Tested	25-Jul-2024	Test Case (Pass/Fail/Not Executed)	Pass

S #	Prerequisites:
1	Logged in as an Advisor, Owner, or PM.
2	Created household member

S #	Test Data
1	Email = zoefin.advisor+qaroleadv01@gmail.com
2	Pass = QA.role123

Test Scenario	Update information with duplicate name

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Go to household member list	HouseHold member list	As Expected	Pass
2	Locate an existing member.	Select member	As Expected	Pass
3	Click on the "Edit" button next to their details.	Go to form edit member	As Expected	Pass
5	Modify name with duplicate information	An error message is displayed, indicating that the name is duplicated.	As Expected	Pass

Test Case ID	CP_006	Test Case Description	Cancel edit operation	
Tester's Name	Sergio Gomez	**Date Tested**	25-Jul-2024	**Test Case (Pass/Fail/Not Executed)** Pass

S #	Prerequisites:		S #	Test Data
1	Logged in as an Advisor, Owner, or PM.		1	Email = zoefin.advisor+qaroleadv01@gmail.com
2	Created household member		2	Pass = QA.role123

Test Scenario Cancel edit operation

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Go to household member list	HouseHold member list	As Expected	Pass
2	Locate an existing member.	Select member	As Expected	Pass
3	Click on the "Edit" button next to their details.	Go to form edit member	As Expected	Pass
5	Click the "Cancel" button	The changes are discarded, and the member's original details remain unchanged.	As Expected	Pass

Delete Members

Write two test cases for the functionality of deleting a member from a household.

Test Case ID	CP_007	Test Case Description	Delete a household member	
Tester's Name	Sergio Gomez	**Date Tested**	25-Jul-2024	**Test Case (Pass/Fail/Not Executed)** Pass

S #	Prerequisites:		S #	Test Data
1	Logged in as an Advisor, Owner, or PM.		1	Email = zoefin.advisor+qaroleadv01@gmail.com
2	Created household member		2	Pass = QA.role123

Test Scenario Delete a household member

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Go to household member list	HouseHold member list	As Expected	Pass
2	Locate an existing member.	Select member	As Expected	Pass
3	Click on the "Delete" button.	Go to delete form	As Expected	Pass
5	Click the "Delete" button	The member is removed from the household.	As Expected	Pass

Test Case ID	CP_008	Test Case Description	Cancel delete operation

Tester's Name	Sergio Gomez	Date Tested	25-Jul-2024	Test Case (Pass/Fail/Not Executed)	Pass

S #	Prerequisites:
1	Logged in as an Advisor, Owner, or PM.
2	Created household member

S #	Test Data
1	Email = zoefin.advisor+qaroleadv01@gmail.com
2	Pass = QA.role123

Test Scenario Cancel delete operation

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Go to household member list	HouseHold member list	As Expected	Pass
2	Locate an existing member.	Select member	As Expected	Pass
3	Click on the "Delete" button.	Go to delete form	As Expected	Pass
5	Click the "Cancel" button	The changes are discarded.	As Expected	Pass

Roles and Permissions

Write two test cases to verify functionalities for a specific role.

Test Case ID	CP_009	Test Case Description	Client Role Permissions Edit

Tester's Name	Sergio Gomez	Date Tested	25-Jul-2024	Test Case (Pass/Fail/Not Executed)	Pass

S #	Prerequisites:
1	Logged in as an Advisor, Owner, or PM.
2	Created household member

S #	Test Data
1	Email = zoefin.advisor+qaroleadv01@gmail.com
2	Pass = QA.role123

Test Scenario Client Role Permissions Edit

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Go to household member list	HouseHold member list	As Expected	Pass
2	Locate an existing member.	Select member	As Expected	Pass
3	Search the "Edit" button	The button is not displayed	As Expected	Pass

Test Case ID	CP_010	Test Case Description	Client Role Permissions Add Member	

Tester's Name	Sergio Gomez	Date Tested	25-Jul-2024	Test Case (Pass/Fail/Not Executed)	Pass

S #	Prerequisites:
1	Logged in as an Advisor, Owner, or PM.
2	Created household member

S #	Test Data
1	Email = zoefin.advisor+qaroleadv01@gmail.com
2	Pass = QA.role123

Test Scenario	Client Role Permissions Add Member

| Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Go to household member list	HouseHold member list	As Expected	Pass
2	Search the "Add Member" button	The button is not displayed	As Expected	Pass

Test Automation

Context:

Use the UI to create the automation

Website https://dashboard-test.zoefin.com/login

Instructions:

1. The Project is located on the route:
   https://github.com/Sergomez94/PruebaZoefin.git

2. The file with the html report with evidence of the successfully executed test cases is located in the path:
   target/site/serenity/index.html

3. The first point was developed, but on the afternoon of July 25, the page in the AddNewClient module stopped working (evidence attached), therefore the identification and creation of the script was carried out by an alternative path from the list of members of the household with the addMember button, the original code was commented in the page object class
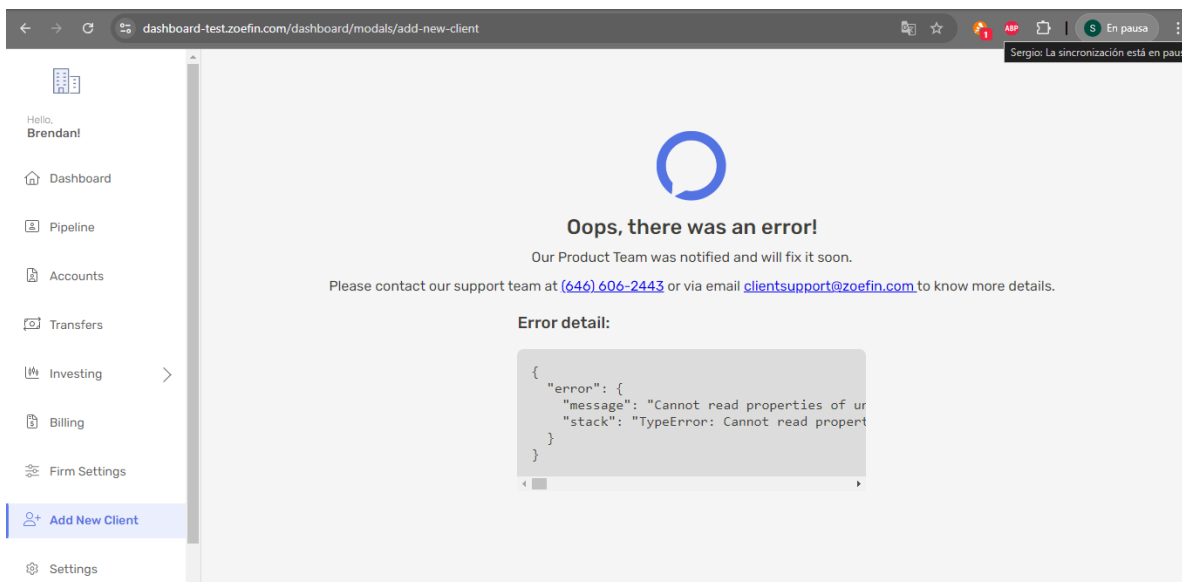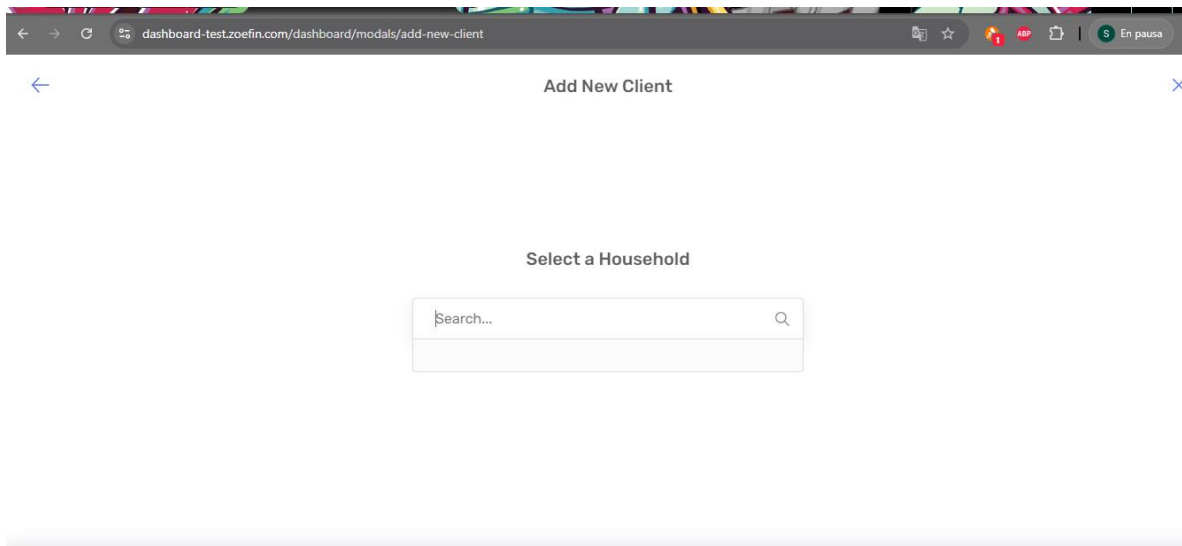
4. To run the Project it is necessary to have Windows and Google Chrome version 127 installed since chromedriver.exe is for that version of Chrome and for Windows. The chromedriver is located in the path: src/test/resources/webDriver/chromedriver.exe

5. The data file is located in the path: src/test/resources/data/DataFile.xlsx

6. The cucumber file where the test cases are written is in the path: src/test/resources/features/pruebaTecnica.feature

Part 4: Case Studies

Case Study 1

You are faced with a problem where Advisors, Owners, and PMs report that they cannot add new members to households. Describe the steps you would take to investigate and resolve the problem.

1. **Collect Information and Reproduce the Issue:**

   o Start by collecting details about the problem.

   o Reproduce the issue if possible. Try adding a new member to a household using the same process they follow. Note down any errors or unexpected behavior.

2. **Database Investigation:**

   o Check the database.

   o Verify that the database schema aligns with the application's expectations.

3. **Permissions and Access Control:**

   o Verify the permissions of Advisors, Owners, and PMs.

   o Check if there are any role-based access control (RBAC) rules preventing this action.

4. **Test Environment and Data:**

   o Test in different environments (e.g., development, staging, production).

   o Use test data to ensure that the problem isn't related to specific records or data.

5. **Collaborate with Stakeholders:**

   o Talk to the affected users (Advisors, Owners, PMs).

   o Understand their workflows.

Case Study 2

The functionality of editing household members is intermittently failing, allowing users to edit members who already have investment accounts. What measures would you take to identify the root cause and prevent similar issues in the future?

Debugging

Describe your process for identifying and resolving a problem where the edit and delete buttons are shown for members who already have an investment account

1. **Reproduce the Issue:**
   - First, try to consistently reproduce the problem. Identify the specific conditions under which the edit and delete buttons incorrectly appear.
   - Note down any patterns or scenarios where this issue occurs.

2. **Inspect the Code and UI Logic:**
   - Dive into the codebase responsible for rendering the member details and buttons.
   - Look for logic related to displaying these buttons
   - Verify that the UI components (buttons, forms) are correctly wired to the backend data.

3. **Database Queries and Joins:**
   - Review the database queries used to fetch member information.
   - Specifically, check if the query correctly excludes members with existing investment accounts from being editable.

4. **Test Scenarios:**
   - Create test scenarios covering different cases: members with and without investment accounts.
   - Verify that the buttons behave as expected in each scenario.

5. **Logging and Debugging:**
   - Add detailed logging around the code responsible for rendering member details and buttons.
   - Monitor logs during reproduction to identify any unexpected behavior.

6. **Collaborate with Developers Teams:**
   - Discuss the issue with frontend developers and backend engineers.
   - Sometimes the issue might span both frontend and backend components.