

Expense Calculator - Test Report

Test Case 1: Create and Update User

Test Case ID: 1

Test Case Title: Create and Update User in the Expense Tracker Application

Test Scenario:

- Create a new user.
- Update the user details.
- Verify that the user is correctly created and updated.

Preconditions:

- The expense tracker application should be running.
- The necessary services and repositories are initialized.

Test Data:

- User ID: "1"
- Initial Username: "Sownthari"
- Updated Username: "Sownthari123"

Test Steps:

1. Create a new user with the initial username.
2. Verify that the user is created with the correct details.
3. Update the username from "Sownthari" to "Sownthari123".
4. Verify that the username is updated correctly.

Iteration 1: Fail Case

Iteration Summary:

An intentional mistake in the user creation process where the username is incorrectly set.

Test Input:

- User ID: "1"
- Username: "sownthari"

Expected Output:

- User creation should fail or proceed with incorrect data.
- User update should either fail or proceed incorrectly.

Actual Output:

- User creation proceeds with the incorrect username "sownthari".
- User update fails due to the incorrect initial username.

Test Result: FAIL

Remarks:

- The username was not validated during creation.
- The system allowed an invalid username.

Iteration 2: Pass Case**Iteration Summary:**

The error from the first iteration is fixed, and the user is created and updated correctly.

Test Input:

- User ID: "1"
- Username: "Sownthari"

Expected Output:

- User should be created with the username "Sownthari".
- Username should be updated to "Sownthari123".

Actual Output:

- User created successfully with the username "Sownthari".
- Username updated successfully to "Sownthari123".

Test Result: PASS

Remarks:

- The issue from the first iteration was resolved.
- User creation and update flow works as expected.

Test Case 2: Add Transaction

Test Case ID: 2

Test Case Title: Add a New Transaction in the Expense Tracker Application

Test Scenario:

- Add a new transaction.
- Verify that the transaction is correctly added to the user's account.

Preconditions:

- The user should be logged in.
- The expense tracker application should be running.
- Categories and budgets should be predefined.

Test Data:

- Transaction ID: "1"
- User ID: "1"
- Category ID: "1"
- Amount: 100.00
- Type: "Expense"

- Date: "2024-08-01"

Test Steps:

1. Add a new transaction with the specified data.
2. Verify that the transaction is added with the correct details.

Iteration 1: Fail Case**Iteration Summary:**

An intentional mistake in the transaction amount.

Test Input:

- Transaction ID: "1"
- Amount: -100.00

Expected Output:

- Transaction creation should fail due to an invalid amount.

Actual Output:

- Transaction creation proceeds with an invalid amount.

Test Result: FAIL

Remarks:

- The transaction amount was not validated.

Iteration 2: Pass Case**Iteration Summary:**

The error from the first iteration is fixed, and the transaction is added correctly.

Test Input:

- Transaction ID: "1"
- Amount: 100.00

Expected Output:

- Transaction should be added with the correct amount.

Actual Output:

- Transaction added successfully.

Test Result: PASS

Remarks:

- The issue from the first iteration was resolved.
- Transaction creation works as expected.

Test Case 3: Create and Update Category

Test Case ID: 3

Test Case Title: Create and Update Category in the Expense Tracker Application

Test Scenario:

- Create a new category.
- Update the category name.
- Verify that the category is correctly created and updated.

Preconditions:

- The expense tracker application should be running.
- The necessary services and repositories are initialized.

Test Data:

- Category ID: "C001"
- Initial Category Name: "Groceries"
- Updated Category Name: "Food & Groceries"

Test Steps:

1. Create a new category with the initial name.
2. Verify that the category is created with the correct details.
3. Update the category name from "Groceries" to "Food & Groceries".
4. Verify that the category name is updated correctly.

Iteration 1: Fail Case**Iteration Summary:**

An intentional mistake in the category name update.

Test Input:

- Category ID: "1"
- Category Name: "Grocry" (Intentional typo)

Expected Output:

- Category update should either fail or proceed incorrectly.

Actual Output:

- Category update proceeds with the incorrect name "Grocry".

Test Result: FAIL

Remarks:

- The category name was not validated during the update.

Iteration 2: Pass Case**Iteration Summary:**

The error from the first iteration is fixed, and the category is updated correctly.

Test Input:

- Category ID: "1"
- Category Name: "Groceries"

Expected Output:

- Category name should be updated to "Food & Groceries".

Actual Output:

- Category updated successfully.

Test Result: PASS

Remarks:

- The issue from the first iteration was resolved.
- Category update flow works as expected.

Test Case 4: Set and Update Budget

Test Case ID: 4

Test Case Title: Set and Update Budget in the Expense Tracker Application

Test Scenario:

- Set a new budget for a category.
- Update the budget amount.
- Verify that the budget is correctly set and updated.

Preconditions:

- The expense tracker application should be running.
- The necessary services and repositories are initialized.

Test Data:

- Budget ID: "1"
- Category ID: "1"

- Initial Budget Amount: 500.00
- Updated Budget Amount: 600.00

Test Steps:

1. Set a new budget for a category.
2. Verify that the budget is set with the correct amount.
3. Update the budget amount from 500.00 to 600.00.
4. Verify that the budget amount is updated correctly.

Iteration 1: Fail Case**Iteration Summary:**

An intentional mistake in the budget amount update.

Test Input:

- Budget ID: "B001"
- Budget Amount: 100.00 (Less than current expenses)

Expected Output:

- Budget update should fail due to insufficient budget amount.

Actual Output:

- Budget update fails as expected.

Test Result: PASS**Remarks:**

- The system correctly identified the issue.

Iteration 2: Pass Case**Iteration Summary:**

The budget is set and updated correctly.

Test Input:

- Budget ID: "1"
- Budget Amount: 600.00

Expected Output:

- Budget should be updated to 600.00.

Actual Output:

- Budget updated successfully.

Test Result: PASS

Remarks:

- The issue from the first iteration was resolved.
- Budget setting and update flow works as expected.

Final Test Report Summary

Iteration	Test Case ID	Test Case Title	Result	Remarks
1	1	Create and Update User	FAIL	Typo in username, update failed.
2	1	Create and Update User	PASS	Fixed typo, user created and updated successfully.
1	2	Add Transaction	FAIL	Invalid transaction amount, creation failed.
2	2	Add Transaction	PASS	Fixed amount, transaction added successfully.

1	3	Create and Update Category	FAIL	Typo in category name, update failed.
2	3	Create and Update Category	PASS	Fixed typo, category updated successfully.
1	4	Set and Update Budget	PASS	Insufficient budget, update failed as expected.
2	4	Set and Update Budget	PASS	Budget updated successfully.