**User is able to enter their monthly income and expenses and sort them into categories:**

**Action:**

* Upon visiting the "New income/expenses" page, users are able to enter their monthly income and expenses into boxes and click the submit button which then adds the income or expense to the database.

**Reaction:**

* If the user’s input is not valid (e.g. strings, negative integers) a message is displayed to the user clearly describing the error.
* When the data is added to the database, it must be structured correctly and go into the correct rows and columns.

**Environment:**

* The test data will include values that cause the net income to be negative, and values that include the net income to be positive to ensure that all parts of our system can handle these values correctly.
* The test will take place on a docker environment running locally.
* The user acceptance tester will be a college student who can make use of the app to assist them in managing their finances.

**User is able to view an overview of their monthly finances:**

**Action:**

* The user navigates to the monthly expenses or income page to view an overview of their monthly finances. The user can select a certain category from the drop down menu to display the expenses or income within that category.

**Reaction:**

* When visiting the monthly expenses or income page, the user is shown a diagram showing their income or expenses for the month.
* When viewing the monthly expenses page, the user is shown their total expenses and total savings.

**Environment:**

* The test data will include extreme values to check whether the diagram is able to handle such values.
* The test will take place on a docker environment running locally.
* The user acceptance tester will be a college student who can make use of the app to assist them in managing their finances.

**User is able to generate an expense/income report:**

**Action:**

* The user can request an expense or income report for a specific month by selecting the month and clicking "show expense report".

**Reaction:**

* When the user requests an expense or income report for a specific month, data from the database is displayed in an easily readable table containing the expenses or income from that specific month.
* The data will be compared to the data in the tables to ensure that the correct data is being queried.
* The report must display the correct date that was assigned to the expense.
* The expense or income report should display the data in an easy to read format with the name of each expense being shown along with the category it belongs to.

**Environment:**

* The test data will include the data used for testing the monthly income and expense case.
* The test will take place on a docker environment running locally.
* The user acceptance tester will be a college student who can make use of the app to assist them in managing their finances.