Feature 1: Search Bar

Objective:

Verify that users can successfully search for stocks using the search bar and receive relevant results.

Test Cases:

1. **Basic Search Functionality**
    - **Data**: Known stock symbols or company names.
    - **Activity**: Type the stock symbols or company names into the search bar and initiate the search.
    - **Expected Result**: The search results should accurately reflect the query, displaying proper information about the entered stocks.
2. **Empty and Invalid Queries**
    - **Data**: Empty strings, irrelevant strings, or symbols not related to stocks.
    - **Activity**: Enter these invalid queries into the search bar.
    - **Expected Result**: The system should handle invalid queries, showing an appropriate message (e.g., "No results found" or "Please enter a valid stock symbol").

Test Environment

- To test this, the environment should remain local until the above cases are satisfied. Then released to end-users with varying levels of investment experience to test the intuitiveness and accuracy of the search feature.

Feature 2: Favorite Stocks List

Objective:

To ensure users can add, view, and remove stocks from their favorite stocks list efficiently.

Test Cases:

1. **Addition of Stocks to Favorites**
    - **Data**: Various stocks selected by the user.
    - **Activity**: Add chosen stocks to the favorites list through the interface.
    - **Expected Result**: Selected stocks should be consistently added to the user's favorites list.
2. **Removal from Favorites**
    - **Data**: Specific stocks currently in the favorites list.
    - **Activity**: Remove selected stocks from the favorites list.
    - **Expected Result**: Stocks should be immediately removed from the list and no longer accessible from the favorites section.

Test Environment

- Testing will begin on localhost and then move to a group of potential users, including those who frequently use stock watchlists or portfolios.

Feature 3: Portfolio Tracker

Objective:

To confirm that the portfolio tracker accurately manages and reflects the users' stock holdings, including performance metrics.

Test Cases:

1. **Portfolio Creation and Management**
    - **Data**: A selection of stocks and quantities to simulate a real portfolio.
    - **Activity**: Create a portfolio and add the selected stocks with their respective quantities.
    - **Expected Result**: The system should accurately create the portfolio and reflect the correct holdings and quantities.
2. **Modifying the Portfolio**
    - **Data**: Changes in stock quantities, addition of new stocks, and removal of existing stocks from the portfolio.
    - **Activity**: Update the portfolio according to the specified changes and observe the tracker's response.
    - **Expected Result**: The tracker should immediately reflect the changes, updating the performance and value metrics accordingly.

Test Environment

- Initial testing in a local environment, focusing on the functional accuracy, data integrity, and user experience of the portfolio tracker. Testers will include experienced investors, casual users, and financial analysts to cover a wide range of user interactions and expectations.