
System Requirements Specification Index

For

Expense Manager Application

Version 1.0

IIHT Pvt. Ltd.
fullstack@iiht.com

Contents

| | | |
|-------|---------------------------|---|
| 1 | Business-Requirement: | 3 |
| 1.1 | Problem Statement: | 3 |
| 1.1.1 | Expense Manager | 3 |
| 2. | Template Code Structure | |
| 2.1 | Expense Controller | |
| 3. | Resources AVAILABLE: | 4 |
| 4 | Suggested WIREFRAMES: | 5 |
| 5 | Business Validations | 9 |
| 6 | Considerations | 9 |
| 7 | Execution Steps to Follow | 9 |

1 BUSINESS-REQUIREMENT:

1.1 PROBLEM STATEMENT:

The purpose of this application is to allow a person to manage all expenses.

1.1.1 Expense Manager:

The Expense Manager allows you to:

1. Access the home page (**must show option to add expense, search expense and show list of expenses with number of page for pagination**).
2. Should be able to add a new expense.
3. It can have basic fields like Description, Amount, Month & Year.
4. Should be able to get the list of expenses (**should return data in pages and must have buttons as Asc and Desc in each column to sort list ascending or descending respectively**).
5. Should be able to edit any expense.
6. Should be able to search any expense on description basis (**should be implemented using custom query**).

2. TEMPLATE CODE STRUCTURE:

2.1 EXPENSE CONTROLLER

| Method Exposed | Purpose |
|---------------------|--|
| getForm() | Should return page "form" with required data. |
| submitForm() | Should save a new expense and show all expenses via page "expenses." |
| getExpenses() | Should return page "expenselist" with required data. |
| showFormForDelete() | Should delete an expense and return page "expenses". |

3. RESOURCES AVAILABLE:

| Description | View Pages Name | Remarks |
|--------------|-----------------|--|
| Common UI | | |
| Home Page | form | Contains a homepage which shows a form to add expenses and an option to list all expenses along with an option to update any expense.. |
| All Expenses | expenselist | |

3 SUGGESTED WIREFRAMES:

1. Homepage – Visitor Landing Page

←

↺

📘

localhost:8081/expenses

[Add Expense](#)

[Expense List](#)

Expenses

Search Expense

Search By Desc

▼

Search By Year

Search

Total Expenses : 221

1

| Sno. | Month | Desc | Asc | Year | Desc | Asc | Amount | Desc | Asc | Description | Desc | Asc | |
|------|-------|------|-----|------|------|-----|--------|------|-----|-------------|------|-----|---|
| 1 | 2 | | | 2023 | | | 121 | | | rrty | | | Update Delete |
| 2 | 2 | | | 2023 | | | 100 | | | vvg | | | Update Delete |

2. Add an Expense

The screenshot shows a web browser window with the address bar displaying 'localhost:8081'. The page has two navigation links at the top: 'Add Expense' and 'Expense List'. The main heading is 'Expense Form'. Below the heading are four input fields, each with a label and a placeholder text: 'Description : Description', 'Amount : Amount', 'Month : Month', and 'Year : Year'. At the bottom left of the form is a 'Submit' button.

← ↻ ⓘ localhost:8081

[Add Expense](#) [Expense List](#)

Expense Form

Description :

Amount :

Month :

Year :

3. All Expenses

The screenshot shows a web browser window with the address bar displaying 'localhost:8081/?id=2'. The page has two links at the top: 'Add Expense' and 'Expense List'. Below them is the title 'Expense Form'. The form contains four input fields: 'Description : vvg', 'Amount : 100', 'Month : 2', and 'Year : 2023'. At the bottom of the form is a 'Submit' button.

4 BUSINESS VALIDATIONS

1. Id must be of type id.
2. Description value should not be blank, min 3 and max 50 characters.
3. Amount value should not be null.
4. Month value should not be null, min 1 and max 12.
5. Year value should not be null.

5 CONSIDERATIONS

The Code template already contains skeleton methods for service and controller layer. Please write your logic in it.

6 EXECUTION STEPS TO FOLLOW

1. All actions like build, compile, running application, running test cases will be through Command Terminal.
2. To open the command terminal the test takers, need to go to Application menu
(Three horizontal lines at left top) → Terminal → New Terminal
3. To build your project use command:
mvn clean package -Dmaven.test.skip
4. To launch your application:
java -jar springboot-expense-tracker-0.0.1-SNAPSHOT.jar
5. This editor Auto Saves the code
6. If you want to exit(logout) and continue the coding later anytime (using Save & Exit option on Assessment Landing Page) then you need to use **CTRL+Shift+B**-command compulsorily on code IDE. This will push or save the updated contents in the internal git/repository. Else the code will not be available in the next login.
7. These are time bound assessments the timer would stop if you logout and while logging in back using the same credentials the timer would resume from the same time it was stopped from the previous logout.
8. To test any Restful application, the last option on the left panel of IDE, you can find ThunderClient, which is the lightweight equivalent of POSTMAN.
9. This is a web-based application, to run the application on a browser, use the internal browser in the workspace. Click on the second last option on the left panel of IDE, you can find Browser Preview, where you can launch the application.
Note: The application will not run in the local browser
10. Default credentials for MySQL:
 - a. Username: **root**
 - b. Password: **pass@word1**
11. To login to mysql instance: Open new terminal and use following command:
 - a. **sudo systemctl enable mysql**
 - b. **sudo systemctl start mysql**

c. **mysql -u root -p**

The last command will ask for password which is '**pass@word1**'

12. Mandatory: Before final submission run the following command:

mvn test

13. You need to use **CTRL+Shift+B** - command compulsorily on code IDE, before final submission as well. This will push or save the updated contents in the internal git/repository, and will be used to evaluate the code quality.