

DAILY HEALTH TRACKER

IIHT

Time To Complete: 2 hr

CONTENTS

1	Problem Statement	3
2	Proposed Daily Health Tracker Wireframe	4
2.1	Welcome Page	4
3	Business-Requirement	7
4	Validations	7
5	Constraints	8
6	Mandatory Assessment Guidelines	8

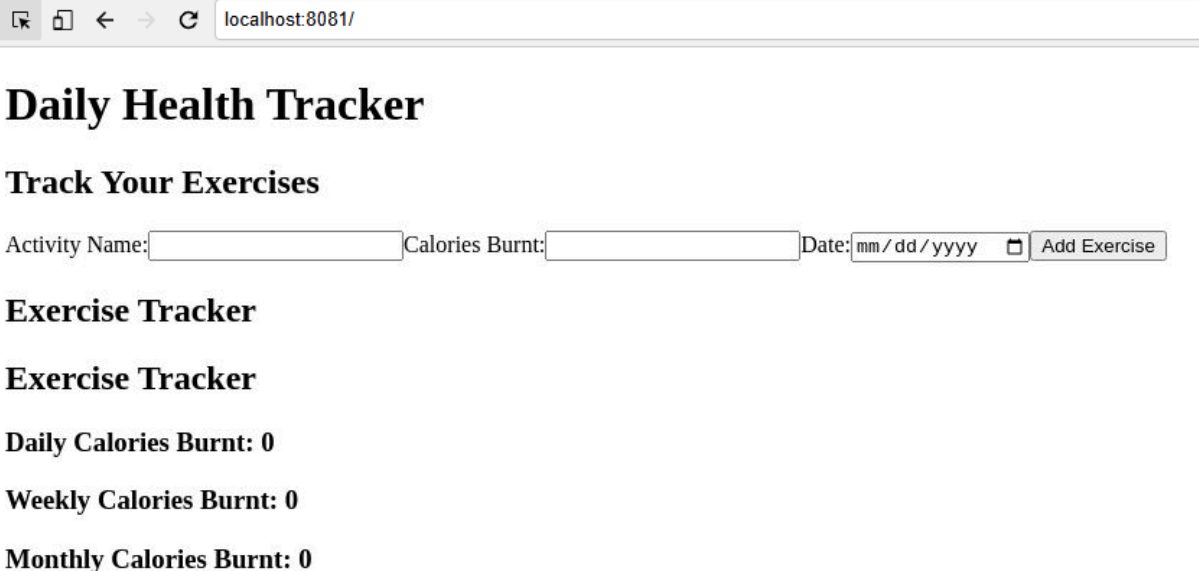
1 PROBLEM STATEMENT

Daily Health Tracker is a Single Page Application (SPA) that allows users to track their physical activities with details such as activity name, calories burnt, and the date of the activity. The application provides a summary of daily, weekly, and monthly calories burnt and displays a detailed list of all recorded activities. Users can add new exercises to the tracker, which updates the overall calorie counts and lists the activities accordingly. The interface includes options to input activity details and view the progress in different time frames.

2 PROPOSED DAILY HEALTH TRACKER WIREFRAME

UI needs improvisation and modification as per given use case and to make test cases passed.

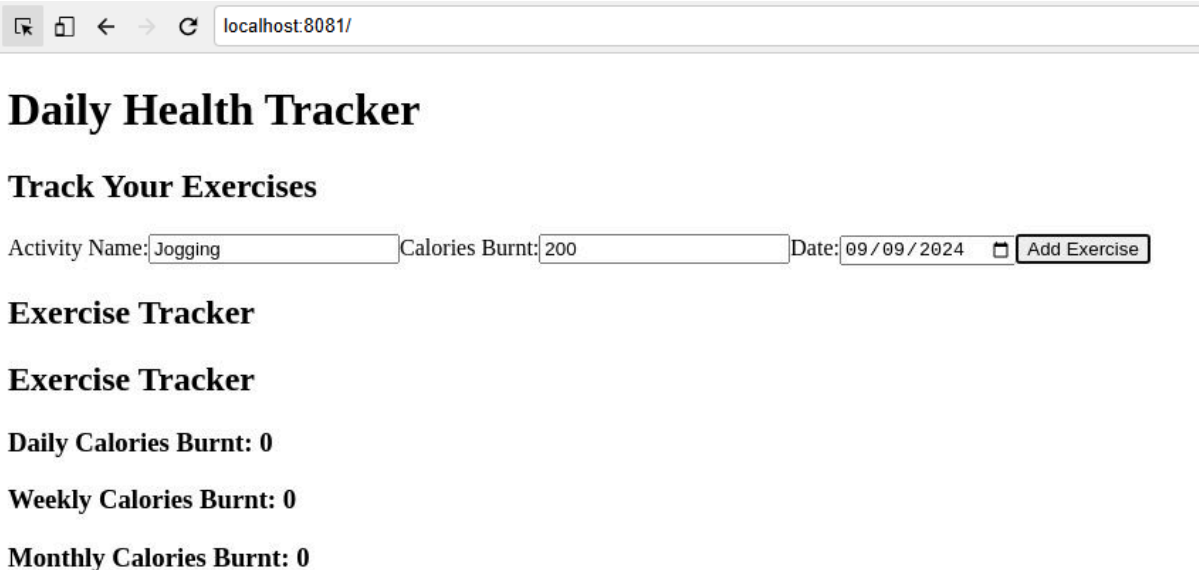
2.1 WELCOME PAGE



The wireframe shows a web browser window with the address bar displaying 'localhost:8081/'. The page title is 'Daily Health Tracker'. Below the title is a section header 'Track Your Exercises'. This is followed by a form with three input fields: 'Activity Name:', 'Calories Burnt:', and 'Date:'. The 'Date:' field has a placeholder 'mm/dd/yyyy' and a calendar icon. To the right of the 'Date:' field is a button labeled 'Add Exercise'. Below the form are three lines of text: 'Exercise Tracker', 'Exercise Tracker', and 'Daily Calories Burnt: 0'. Below these are two more lines of text: 'Weekly Calories Burnt: 0' and 'Monthly Calories Burnt: 0'.

SCREEN SHOTS

ADD TRACKER



The screenshot shows the same web browser window as the wireframe, but with sample data entered in the form. The 'Activity Name:' field contains 'Jogging', the 'Calories Burnt:' field contains '200', and the 'Date:' field contains '09/09/2024'. The 'Add Exercise' button is highlighted with a black border. Below the form are three lines of text: 'Exercise Tracker', 'Exercise Tracker', and 'Daily Calories Burnt: 0'. Below these are two more lines of text: 'Weekly Calories Burnt: 0' and 'Monthly Calories Burnt: 0'.

Daily Health Tracker

Track Your Exercises

Activity Name: Calories Burnt: Date:

Exercise Tracker

Exercise Tracker

Daily Calories Burnt: 200

Weekly Calories Burnt: 200

Monthly Calories Burnt: 200

- **Activity Name:** Jogging
Calories Burnt: 200
Date: 2024-09-09

Daily Health Tracker

Track Your Exercises

Activity Name: Calories Burnt: Date:

Exercise Tracker

Exercise Tracker

Daily Calories Burnt: 450

Weekly Calories Burnt: 450

Monthly Calories Burnt: 450

- **Activity Name:** Jogging
Calories Burnt: 200
Date: 2024-09-09
- **Activity Name:** Swimming
Calories Burnt: 250
Date: 2024-09-09

3 BUSINESS-REQUIREMENT:

As an application developer, develop the Daily Task Tracker (Single Page App) with below guidelines:

User Story #	User Story Name	User Story
US_01	Welcome Page	<p>As a user I should be able to visit the welcome page as default page.</p> <p>Acceptance criteria:</p> <ol style="list-style-type: none">1. Users should be able to see a form to add a new exercise.2. The form must include fields for Activity Name, Calories Burnt, and Date along with an Add Exercise button.3. Users should be able to view a summary of the total calories burnt, including daily, weekly, and monthly totals.4. Users should be able to see a detailed list of all recorded exercises with their respective details.5. Each exercise entry should display the Activity Name, Calories Burnt, and Date.6. The application should update the summary totals automatically when a new exercise is added. <p>** Kindly refer to the screenshots for any clarifications. **</p>

4 VALIDATIONS

- All required fields must be fulfilled with valid data.
- In the starting "Add Exercise" button should be disabled.
- Only after validating fields, "Add Exercise" button should be enabled.
- Once we click on "Add Exercise", A new exercise must be added to the list of recorded exercises.

5 CONSTRAINTS

- You should be able to press the "TAB" key and "SHIFT + TAB" to navigate from top field to bottom field and vice-versa.
- On clicking the "Add Exercise" button, a new task must be added with entered fields.
- "Add Exercise" button will be disabled until all validations are fulfilled.

6 MANDATORY ASSESSMENT GUIDELINES

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. This editor Auto Saves the code.
4. 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.
5. 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.
6. 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.
7. 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

8. You can follow series of command to setup React environment once you are in your project-name folder:
 - a. npm install -> Will install all dependencies -> takes 10 to 15 min
 - b. npm run start -> To compile and deploy the project in browser. You can press the <Ctrl> key while clicking on localhost:8080/8081 to open the project in the browser -> takes 2 to 3 min.
 - a. npm run jest -> to run all test cases and see the summary. It takes 5 to 6 min to run.
 - c. npm run test -> to run all test cases. **It is mandatory to run this command before submission of workspace -> takes 5 to 6 min**

9. You may also run “npm run jest” while developing the solution to refactor the code to pass the test-cases.
10. Once you are done with development and ready with submission, you may navigate to the previous tab and submit the workspace. It is mandatory to click on **“Submit Assessment”** after you are done with code.
11. 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.