**Report on Expense Tracker**

**Introduction**

This report provides an overview and analysis of the Expense Tracker application, which is a web-based tool designed to help users manage their personal

finances by tracking their income, expenses, and overall financial balance. The application was developed using HTML, CSS, and JavaScript, and includes

features such as the ability to add, edit, and delete transactions, as well as view summary information about income, expenses, and the current balance.

The purpose of this report is to summarize the key features and functionality of the Expense Tracker application, as well as to provide a detailed analysis

of the code and design decisions that were made during the development process.

**Section 1: User Interface and Functionality**

- The Expense Tracker application has a clean and intuitive user interface, with a header that displays the application name and a wallet icon.

- The main content area is divided into several sections:

- The input form, which allows users to enter details about a new transaction, including the date, amount, transaction type (income or expense), and a description.

- A set of cards that display the total income, total expenses, and current balance.

- A table that displays a list of all transactions, including the amount, transaction type, date, and a delete button.

- The application uses local storage to persist transaction data, so that users can access their transaction history even after closing and reopening the app.

- The application includes validation checks to ensure that users enter valid data, such as ensuring that the date is not in the future and that the amount

is a positive number.

**Section 2: Code Structure and Design**

- The Expense Tracker application is built using HTML, CSS, and JavaScript.

- The HTML file (`Tracker.html`) defines the structure of the page, including the input form, cards, and transaction table.

- The CSS file (`Tracker.css`) defines the styles for the various UI elements, including the header, input form, cards, and transaction table.

- The JavaScript code (`Tracker.html`) handles the functionality of the application, including:

- Adding new transactions to the table and updating the summary cards

- Deleting transactions from the table and updating the summary cards

- Saving and loading transaction data from local storage

- Validating user input to ensure that it is valid and consistent

- The JavaScript code uses a combination of DOM manipulation, event handling, and data storage/retrieval to implement the application's functionality.

**Conclusion**

The Expense Tracker application provides a simple and effective way for users to manage their personal finances by tracking their income, expenses, and overall

financial balance. The application's clean and intuitive user interface, combined with its robust functionality and data persistence features, make it a valuable

tool for anyone looking to better understand and manage their spending habits.

Overall, the Expense Tracker application is a well-designed and well-implemented project that demonstrates the power of modern web development technologies

and techniques.

**Technologies Used in Expense Tracker**

The Expense Tracker website is built using the following technologies:

1. **HTML (Hypertext Markup Language)**: HTML is the standard markup language used to create the structure and content of web pages.

In the Expense Tracker application, HTML is used to define the overall structure of the web page, including the input form, cards, and transaction table.

2. **CSS (Cascading Style Sheets)**: CSS is the style sheet language used to control the presentation and visual appearance of web pages.

In the Expense Tracker application, CSS is used to define the styles for the various UI elements, such as the header, input form, cards, and transaction table.

3. **JavaScript**: JavaScript is a programming language used to add interactivity and dynamic behavior to web pages. In the Expense Tracker application,

JavaScript is used to handle the functionality of the application, including adding new transactions, deleting transactions, updating the summary cards, and saving/loading transaction data from local storage.

4. **Bootstrap**: Bootstrap is a popular CSS framework that provides a set of pre-designed UI components and styles. In the Expense Tracker application, Bootstrap is used to provide a responsive and mobile-friendly design, as well as to style the various UI elements, such as the input form and buttons.

5. **Local Storage**: The Expense Tracker application uses the browser's local storage API to persist transaction data, so that users can access their transaction history even after closing and reopening the app.

Together, these technologies work together to create a functional and visually appealing Expense Tracker application that allows users to manage their personal finances.

**By…**

J. HARINATH