**Project Report: Money Tracker Web App**

**1. Introduction**

The Money Tracker Web App is a simple yet effective tool designed to help users keep track of their expenses. It allows users to categorize their expenditures, input the amount spent, and record the date of the expense. The application also provides the functionality to view the list of all expenses, calculate the total amount spent, and delete individual expense entries.

**2. Project Objective**

The primary objective of this project is to develop a web application that helps users manage their finances by tracking their daily expenses. The application should be user-friendly, efficient, and visually appealing.

**3. Technologies Used**

* **HTML**: To structure the content of the web app.
* **CSS**: For styling the web app to make it visually appealing.
* **JavaScript**: To add interactivity and handle the logic for managing expenses.

**4. Features**

**4.1 Input Section**

* **Category Selection**: Users can select a category for their expense from a predefined list (e.g., College Fee, Rent, Transport, Food, Shopping, Cool Drinks).
* **Amount Input**: Users can enter the amount spent.
* **Date Input**: Users can record the date of the expense.
* **Add Button**: Allows users to add the expense to the list.

**4.2 Expenses List**

* **Expense Display**: Displays a list of all recorded expenses in a tabular format.
* **Delete Functionality**: Each expense entry has a delete button that allows users to remove the entry.
* **Total Amount Calculation**: Displays the total amount spent based on the recorded expenses.

**5. Implementation Details**

**5.1 HTML Structure**

The HTML structure consists of:

* An input section for adding expenses.
* A table to display the list of expenses and the total amount.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="stylesheet" href="style.css">

<title>Money Tracker Web App</title>

</head>

<body>

<h1>Money Tracker Web App</h1>

<div class="input-section">

<label for="category-select">Category:</label>

<select id="category-select">

<option value="College Fee">College Fee</option>

<option value="Rent">Rent</option>

<option value="Transport">Transport</option>

<option value="Food">Food</option>

<option value="Shopping">Shopping</option>

<option value="Cool Drinks">Cool Drinks</option>

</select>

<label for="amount-input">Amount:</label>

<input type="number" id="amount-input">

<label for="date-input">Date:</label>

<input type="date" id="date-input">

<button id="add-btn">Add</button>

</div>

<div class="expenses-list">

<h2>Expenses List</h2>

<table>

<thead>

<tr>

<th>Category</th>

<th>Amount</th>

<th>Date</th>

<th>Delete</th>

</tr>

</thead>

<tbody id="expense-table-body">

</tbody>

<tfoot>

<td>Total:</td>

<td id="total-amount"></td>

<td></td>

<td></td>

</tfoot>

</table>

</div>

<script src="script.js"></script>

</body>

</html>

**5.2 CSS Styling**

The CSS provides basic styling for the web app, making it visually appealing and user-friendly.

body {

font-family: Arial, Helvetica, sans-serif;

margin: 0;

}

h1, h2 {

text-align: center;

}

.input-section {

display: flex;

flex-direction: row;

justify-content: space-between;

align-items: center;

padding: 10px;

}

.input-section label {

font-weight: bold;

margin-right: 10px;

}

.input-section input[type="number"],

.input-section input[type="date"] {

padding: 5px;

margin-right: 10px;

}

.input-section button {

padding: 5px 10px;

background-color: rgb(238, 9, 9);

color: white;

border: none;

border-radius: 4px;

cursor: pointer;

}

.expenses-list {

margin: 20px;

}

table {

width: 100%;

border-collapse: collapse;

}

th, td {

border: 1px solid #ddd;

padding: 8px;

text-align: left;

}

th {

background-color: rgb(215, 222, 38);

color: rgb(12, 1, 1);

}

tfoot td {

font-weight: bold;

}

.delete-btn {

padding: 5px 10px;

background-color: rgb(238, 9, 9);

color: white;

border: none;

border-radius: 4px;

cursor: pointer;

}

**5.3 JavaScript Functionality**

The JavaScript adds interactivity and handles the logic for managing expenses.

document.addEventListener("DOMContentLoaded", function () {

const addBtn = document.getElementById("add-btn");

const categorySelect = document.getElementById("category-select");

const amountInput = document.getElementById("amount-input");

const dateInput = document.getElementById("date-input");

const expenseTableBody = document.getElementById("expense-table-body");

const totalAmountDisplay = document.getElementById("total-amount");

let totalAmount = 0;

addBtn.addEventListener("click", function () {

const category = categorySelect.value;

const amount = parseFloat(amountInput.value);

const date = dateInput.value;

if (isNaN(amount) || amount <= 0 || !date) {

alert("Please enter a valid amount and date.");

return;

}

const tr = document.createElement("tr");

const tdCategory = document.createElement("td");

tdCategory.textContent = category;

const tdAmount = document.createElement("td");

tdAmount.textContent = amount;

const tdDate = document.createElement("td");

tdDate.textContent = date;

const tdDelete = document.createElement("td");

const deleteBtn = document.createElement("button");

deleteBtn.textContent = "Delete";

deleteBtn.className = "delete-btn";

deleteBtn.addEventListener("click", function () {

totalAmount -= amount;

totalAmountDisplay.textContent = totalAmount.toFixed(2);

tr.remove();

});

tdDelete.appendChild(deleteBtn);

tr.appendChild(tdCategory);

tr.appendChild(tdAmount);

tr.appendChild(tdDate);

tr.appendChild(tdDelete);

expenseTableBody.appendChild(tr);

totalAmount += amount;

totalAmountDisplay.textContent = totalAmount.toFixed(2);

// Clear input fields

amountInput.value = "";

dateInput.value = "";

});

});

**6. Conclusion**

The Money Tracker Web App is a simple and effective tool for managing personal expenses. The app allows users to add, view, and delete expenses, providing a clear overview of their spending. The project successfully demonstrates the use of HTML, CSS, and JavaScript to create a functional and user-friendly web application.

**7. Future Enhancements**

* **Data Persistence**: Implementing data persistence using local storage or a database to save expenses even after the browser is closed.
* **User Authentication**: Adding user authentication to allow multiple users to use the app with their individual accounts.
* **Expense Analysis**: Providing graphs and charts to visualize spending patterns and categories.