NAAN MUDHALVAN

PROJECT PHASE - III

BY,

AI&DS Department 2nd Year

Team Leader : Mathavan V

Team Member : Mathavan V

Bharani
Prasana

John Vimal

Meedhun

Jose Kiri

Phase 3: Implementation of Project

Title: Energy Consumption Tracker App

Objective

The goal of Phase 3 is to implement the core components of the Energy Consumption Tracker App based on the plans and innovative solutions developed during Phase 2. This includes the development of the user interface, data logging functionality, data visualization, and cloud storage integration.

1. User Interface Development

Overview: The user interface (UI) is crucial for user interaction with the app. It must be intuitive and engaging to encourage regular use.

Implementation:

- **HTML/CSS Structure:** Develop a responsive layout using HTML and CSS that works seamlessly on both desktop and mobile devices.
- **JavaScript Functionality:** Implement JavaScript to handle user interactions, such as form submissions and data retrieval.
- **User Experience (UX) Testing:** Conduct usability testing with a small group of users to gather feedback on the interface design.

Outcome: By the end of this phase, the app will have a fully functional and visually appealing user interface that allows users to log energy consumption easily.

2. Data Logging Functionality

Overview: The app will allow users to input and store energy consumption data for various appliances.

Implementation:

- **Form Handling:** Develop a form for users to input appliance name, energy consumed (in kWh), and date of usage.
- **Local Storage:** Implement local storage to save user data temporarily for offline access.
- **Data Validation:** Ensure that the input data is validated to prevent errors (e.g., non-numeric values for energy consumption).

Outcome: Users will be able to log their energy consumption data accurately, and the app will store this data for future reference.

3. Data Visualization

Overview: Visualizing energy consumption data helps users understand their usage patterns and identify opportunities for savings.

Implementation:

- **Charting Library Integration:** Use Chart.js to create interactive charts that display energy consumption over time.

- **Dynamic Updates:** Ensure that the charts update automatically as new data is logged.
- **User Filters:** Allow users to filter data by date range and appliance type for more detailed analysis.

Outcome: The app will provide users with clear and informative visualizations of their energy consumption, making it easier to analyze trends.

4. Cloud Storage Integration

Overview: To ensure data accessibility across devices, the app will integrate cloud storage solutions.

Implementation:

- **Firebase Setup:** Configure Firebase for user authentication and data storage.
- **Data Synchronization:** Implement real-time data synchronization so that users can access their data from any device.
- **Security Measures:** Ensure that user data is stored securely and complies with data privacy regulations.

Outcome: Users will have secure access to their energy consumption data from any device, enhancing the app's usability and functionality.

5. Testing and Feedback Collection

Overview: Initial testing of the app will be carried out to evaluate its performance, accuracy, and user experience.

Implementation:

- **Test Groups:** Recruit a small group of users to test the app, focusing on data entry, visualization, and overall usability.
- **Feedback Loop:** Collect feedback regarding the app's functionality, ease of use, and any issues encountered during testing.

Outcome: The feedback gathered will guide improvements in the app, ensuring it meets user needs and expectations.

Challenges and Solutions

1. **User Engagement:**
    - **Challenge:** Users may forget to log their energy consumption regularly.
    - **Solution:** Implement reminders and notifications to encourage users to log their data.
2. **Data Accuracy:**
    - **Challenge:** Users may input incorrect data.
    - **Solution:** Implement data validation and provide clear instructions on how to log data accurately.
3. **Performance Issues:**

- **Challenge:** The app may experience slow performance with large datasets.
- **Solution:** Optimize data handling and implement pagination for visualizations.

Outcomes of Phase 3

By the end of Phase 3, the following milestones should be achieved:

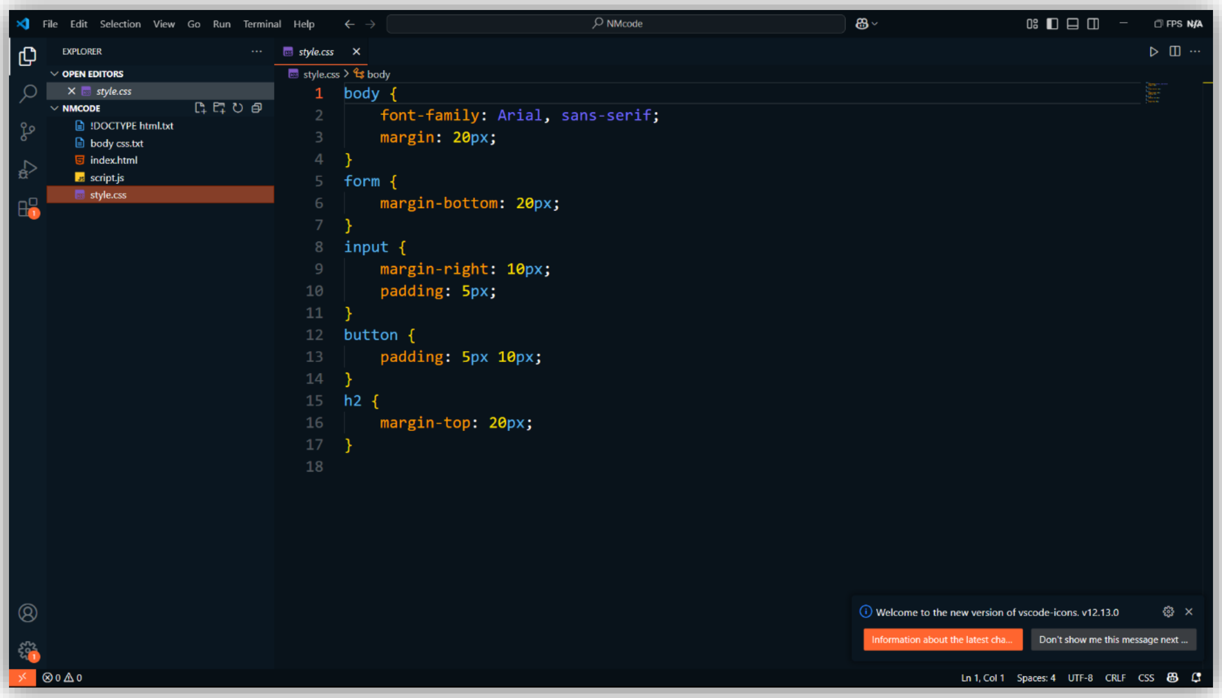
1. A fully functional user interface that allows easy data entry.
2. Accurate data logging functionality with local storage.
3. Interactive data visualizations that help users analyze their energy consumption.
4. Cloud storage integration for secure and accessible data.
5. Initial testing completed with user feedback collected for future improvements.

Next Steps for Phase 4

In Phase 4, the team will focus on:

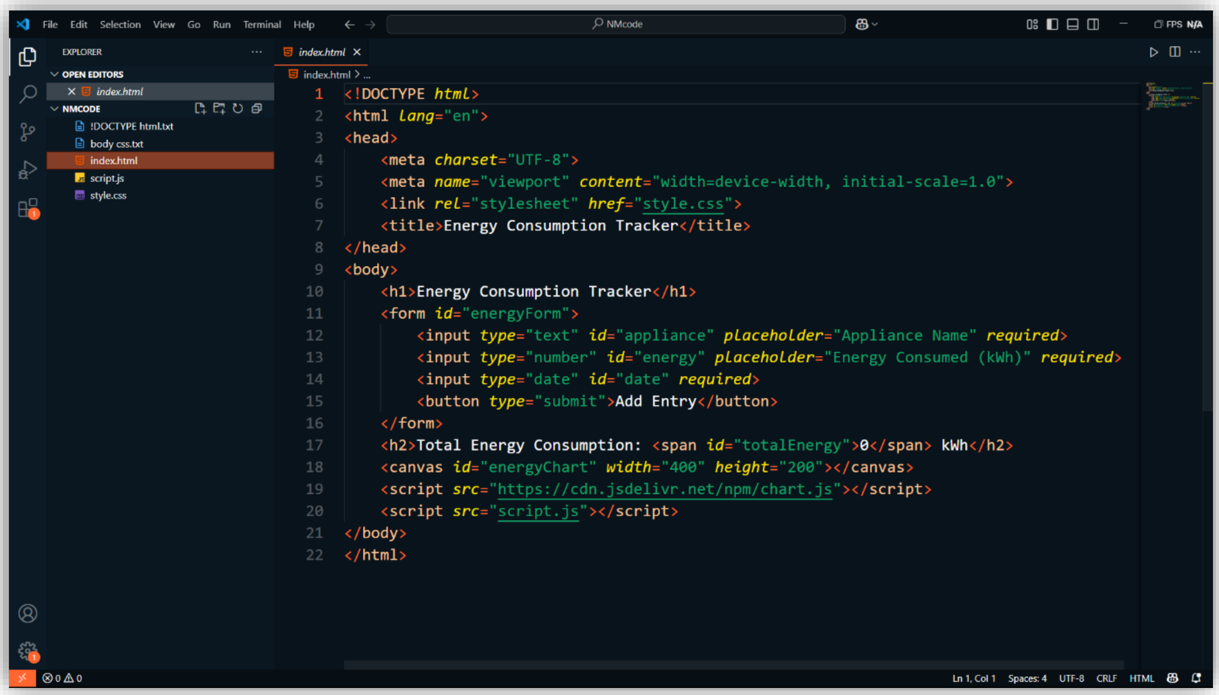
1. Enhancing the app based on user feedback and testing results.
2. Adding advanced features such as personalized energy-saving tips and user authentication.
3. Preparing for a wider launch and marketing the app to potential users.

HTML/CSS Structure:



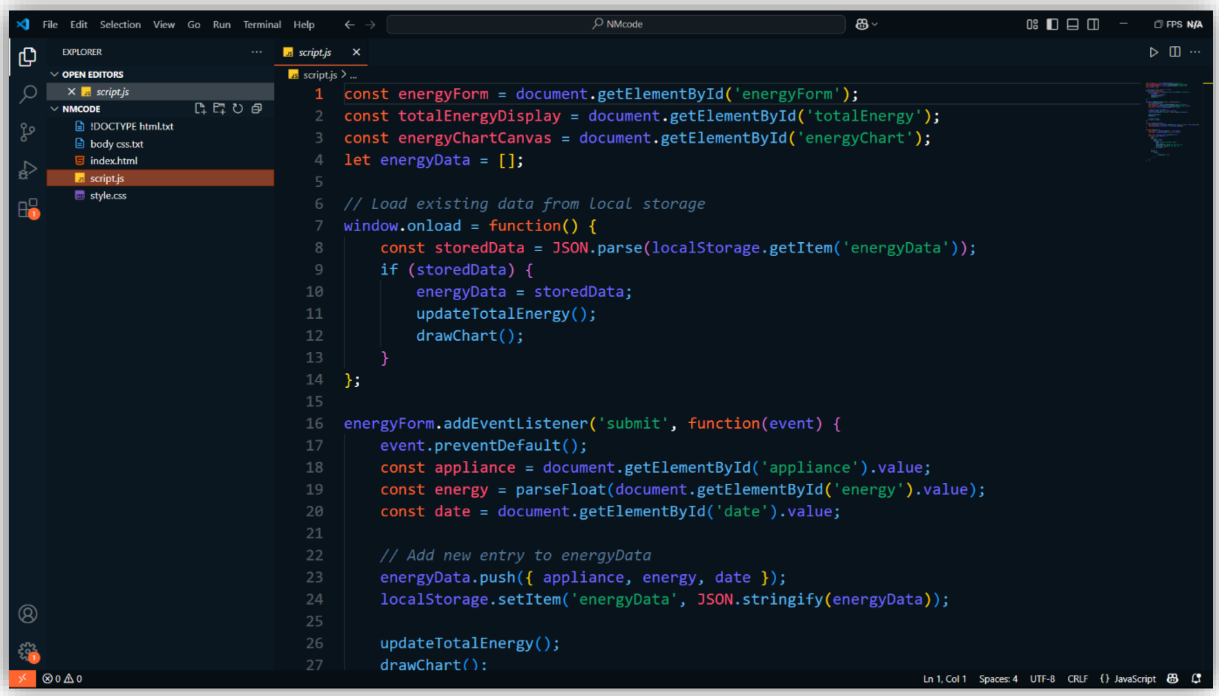
```css
body {
    font-family: Arial, sans-serif;
    margin: 20px;
}
form {
    margin-bottom: 20px;
}
input {
    margin-right: 10px;
    padding: 5px;
}
button {
    padding: 5px 10px;
}
h2 {
    margin-top: 20px;
}

```

Welcome to the new version of vscode-icons. v12.13.0



```html
<!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>Energy Consumption Tracker</title>
</head>
<body>
    <h1>Energy Consumption Tracker</h1>
    <form id="energyForm">
        <input type="text" id="appliance" placeholder="Appliance Name" required>
        <input type="number" id="energy" placeholder="Energy Consumed (kWh)" required>
        <input type="date" id="date" required>
        <button type="submit">Add Entry</button>
    </form>
    <h2>Total Energy Consumption: <span id="totalEnergy">0</span> kWh</h2>
    <canvas id="energyChart" width="400" height="200"></canvas>
    <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
    <script src="script.js"></script>
</body>
</html>
```

JavaScript:



```javascript
const energyForm = document.getElementById('energyForm');
const totalEnergyDisplay = document.getElementById('totalEnergy');
const energyChartCanvas = document.getElementById('energyChart');
let energyData = [];

// Load existing data from local storage
window.onload = function() {
    const storedData = JSON.parse(localStorage.getItem('energyData'));
    if (storedData) {
        energyData = storedData;
        updateTotalEnergy();
        drawChart();
    }
};

energyForm.addEventListener('submit', function(event) {
    event.preventDefault();
    const appliance = document.getElementById('appliance').value;
    const energy = parseFloat(document.getElementById('energy').value);
    const date = document.getElementById('date').value;

    // Add new entry to energyData
    energyData.push({ appliance, energy, date });
    localStorage.setItem('energyData', JSON.stringify(energyData));

    updateTotalEnergy();
    drawChart();
```