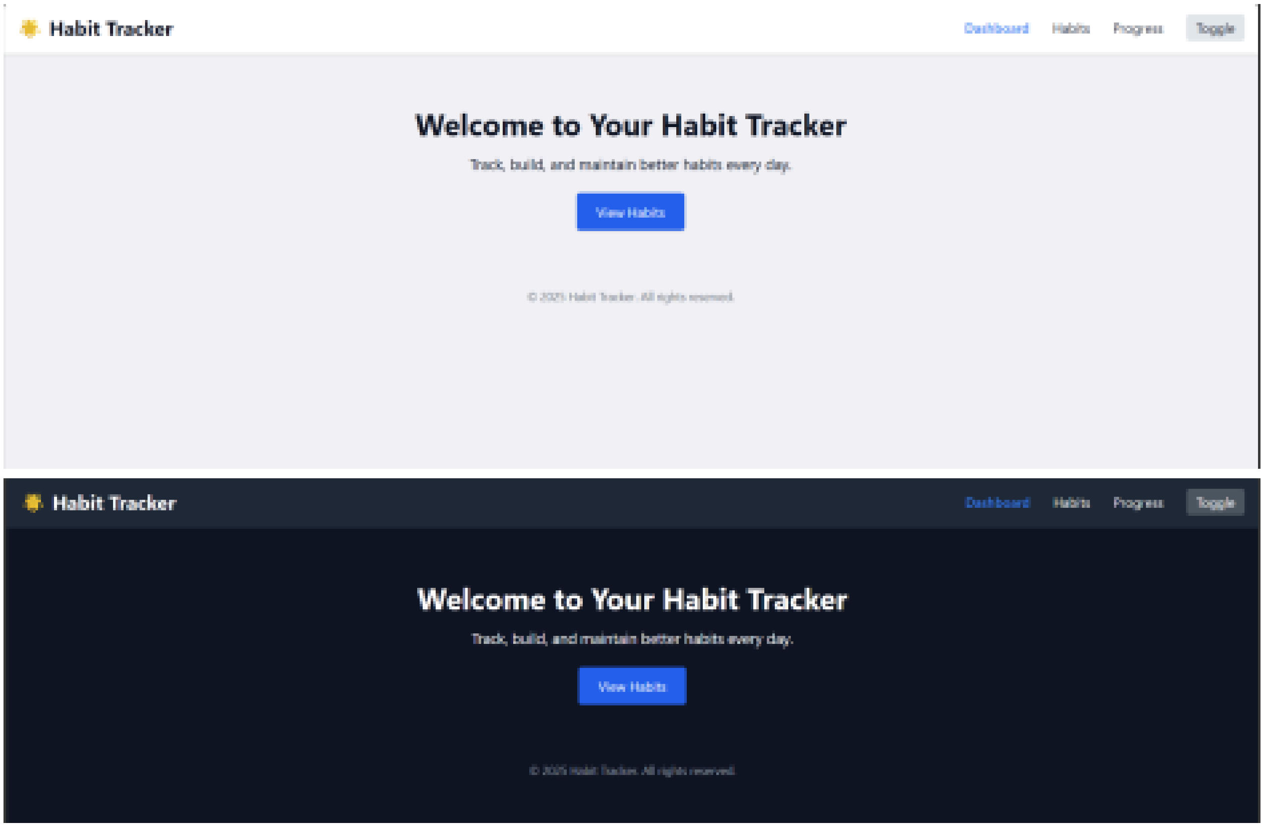
EXPERIMENT 1

Code and explanation

1. Dark Mode Toggle Button

The **Dashboard** of the Habit Tracker greets users with a clean and minimal welcome screen, encouraging them to start tracking their habits. It features a “View Habits” button for navigation and a **"Toggle" button** that switches between light and dark themes using the toggleTheme() JavaScript function. Tailwind CSS classes dynamically change the page appearance, enhancing user comfort in different lighting conditions.

```
 <script>
 function toggleTheme() {
 document.documentElement.classList.toggle('dark'); }
 </script>
 <button onclick="toggleTheme()" class="ml-3 px-3 py-1 bg-gray-200
dark:bg-gray-600 rounded">Toggle</button>
```



Welcome to Your Habit Tracker

Track, build, and maintain better habits every day.

© 2025 Habit Tracker. All rights reserved.

Welcome to Your Habit Tracker

Track, build, and maintain better habits every day.

© 2025 Habit Tracker. All rights reserved.

Fig 1: Home page/ Dashboard

2. Modal for Adding Habits

The image displays a **dark-themed Habit Tracker web app** with a modal popup for adding a new habit. The modal includes a text input for the habit name, a dropdown to select the frequency (Daily or Weekly), and two buttons **Add**. The modal is styled with Tailwind CSS, featuring rounded edges, dark mode support, and a clean, centered layout for better user experience.

```
 <div id="habitModal" class="fixed inset-0 flex items-center
justify-center bg-black bg-opacity-50 hidden z-50">
 <div class="bg-white dark:bg-gray-800 p-6 rounded shadow-lg w-full
max-w-md">
 <h3 class="text-xl font-semibold mb-4">Add New Habit</h3> <input
type="text" id="habitName" placeholder="Habit name..."
class="w-full mb-3 px-3 py-2 rounded border dark:bg-gray-700
dark:border-gray-600" />
 <select id="habitFrequency" class="w-full mb-4 px-3 py-2 rounded
border dark:bg-gray-700 dark:border-gray-600">
 <option value="Daily">Daily</option>
 <option value="Weekly">Weekly</option>
 </select>
 <div class="flex justify-end space-x-2">
 <button onclick="closeModal()" class="px-4 py-2 bg-gray-300
dark:bg-gray-600 rounded">Cancel</button>
 <button onclick="addHabit()" class="px-4 py-2 bg-blue-600
text-white rounded hover:bg-blue-700">Add</button>
 </div>
 </div> </div>
```

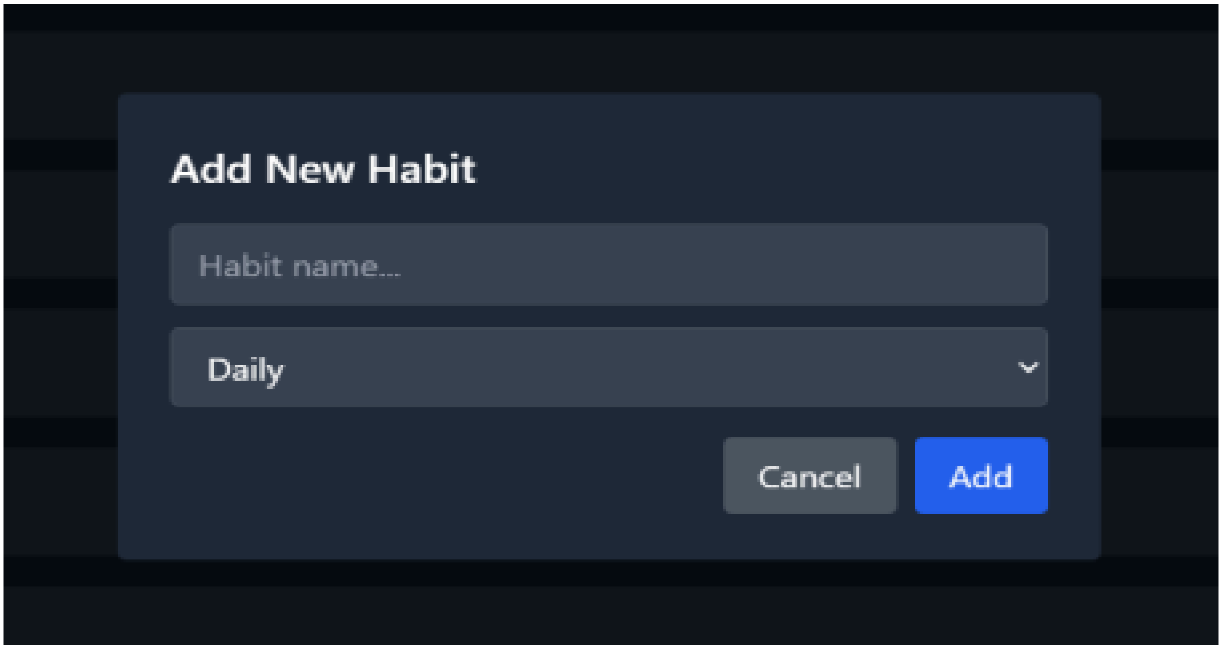
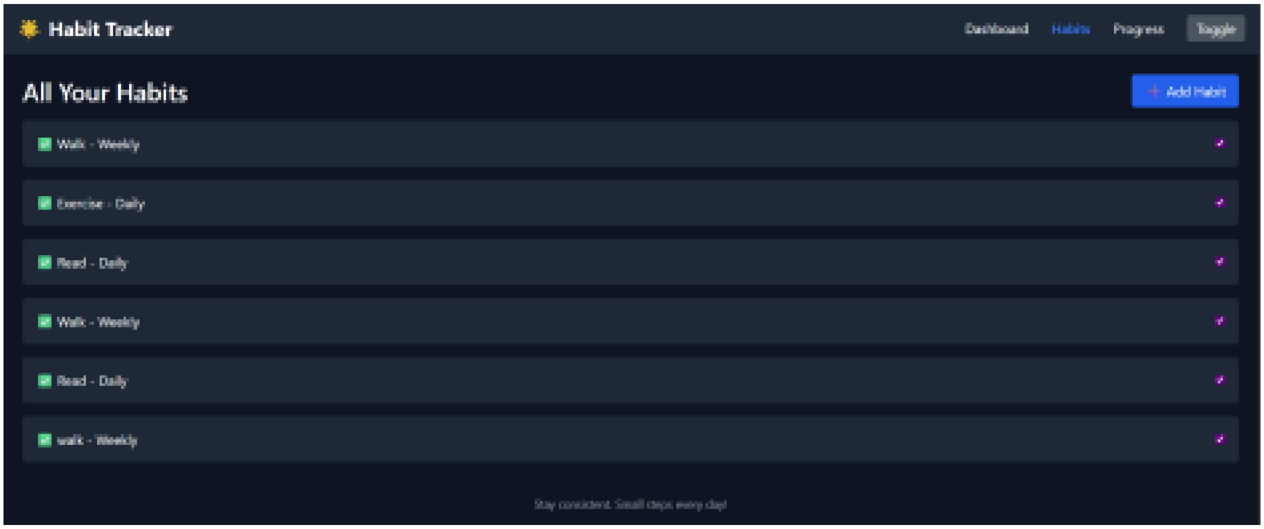


Fig 2.1: Adding habits

3. Rendering Habits on the Habits Page

The updated Habit Tracker interface displays a **list of user habits** with a checkbox for tracking completion. Each habit shows its name and frequency (e.g., “Drink Water - Daily” or “Walk - Weekly”). The renderHabits() function dynamically creates and styles each habit entry using JavaScript, with visual feedback (✅ or ⬜) and checkboxes that reflect and update the completion status. The layout is styled using Tailwind CSS, ensuring a clean and modern dark theme.

```
function renderHabits() {
 const list = document.getElementById("habitList");
list.innerHTML = "";
 habits.forEach((habit, index) => {
 const li = document.createElement("li");
 li.className = "bg-white dark:bg-gray-800 p-4 rounded shadow flex
items-center justify-between";
 li.innerHTML = `
 <span>${habit.completed ? '✅' : '⬜'} ${habit.name} -
${habit.frequency}</span>
 <input type="checkbox" ${habit.completed ? "checked" : ""}
onchange="toggleComplete(${index})" />
  `;
 list.appendChild(li);
                                            });
```



All Your Habits

☑ Walk - Weekly

☑ Exercise - Daily

☑ Read - Daily

☑ Walk - Weekly

☑ Read - Daily

☑ walk - Weekly

Stay consistent. Small steps every day!

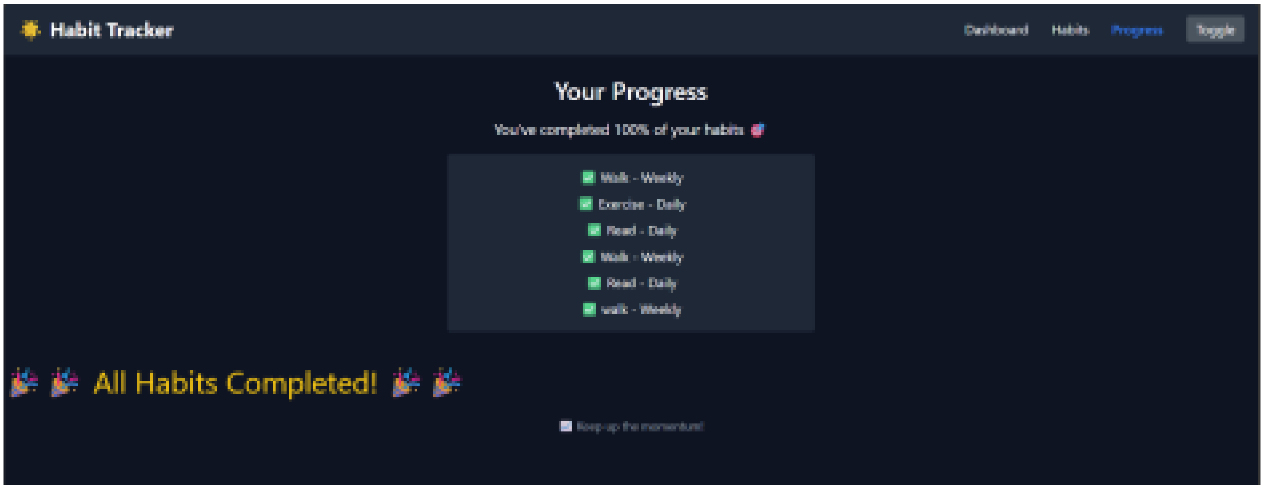
Fig 2.2: All Habits

4. Progress Calculation Logic

The **Progress** page of the Habit Tracker visually shows the percentage of completed habits. The code calculates progress by dividing the number of completed habits by the total number, then displays the result dynamically. If all habits are completed, a celebratory message like "�� All Habits Completed!" appears. Styled with Tailwind CSS, this section provides users with motivation and visual feedback to track their consistency in real-time

```
. let completed = habits.filter(h => h.completed).length;
 let total = habits.length;


 if (total === 0) {
 progressText.textContent = "No habits yet. Add some to start
tracking!";
  } else {
 let percentage = Math.round((completed / total) * 100);
progressText.textContent = "You have completed ${percentage}% of your
habits ��";
```



Your Progress

You've completed 100% of your habits

- ✅ Walk - Weekly
- ✅ Exercise - Daily
- ✅ Read - Daily
- ✅ Walk - Weekly
- ✅ Read - Daily
- ✅ walk - Weekly

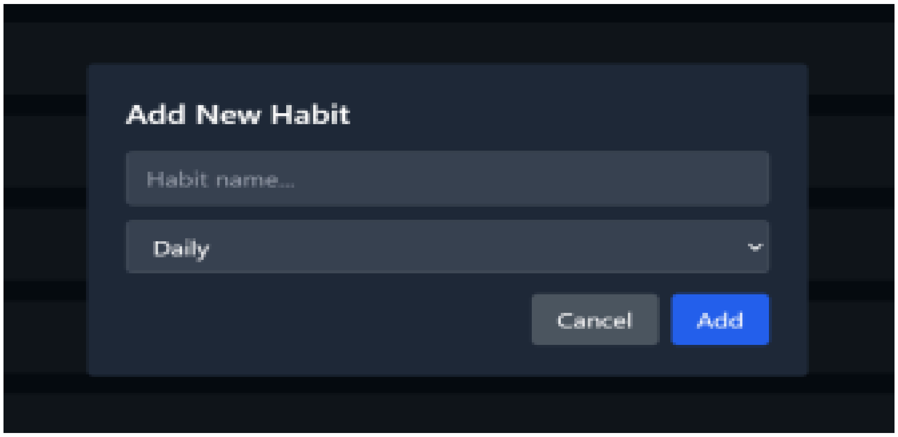
🎉 🎉 All Habits Completed! 🎉 🎉

Keep up the momentum!

Fig 3: Progress Page

5. Adding Habits with JavaScript

```
function addHabit() {
 const name =
document.getElementById('habitName').value.trim();
 const frequency =
document.getElementById('habitFrequency').value;
 if (name !== '') {
 habits.push({ name, frequency, completed: false });
localStorage.setItem("habits", JSON.stringify(habits));
closeModal();
 renderHabits();
  }
  }
```


Add New Habit

Daily