

This project is a Habit Tracker web application that allows users to create, track, and delete daily habits. Users can enter a habit name and description, add it to a list, and mark it as completed each day. Each habit keeps track of the number of days it has been completed. The interface features two main sections: one for adding a habit and another for displaying the added habits. These sections are visually styled with drop shadows and beveled boxes to improve user experience. The server is built using Express.js and handles CRUD operations for the habits.

Example AI Prompts Used:

- Encase both the Habit Adder and the Added Habits in a beveled box with a drop shadow, the added habits box should be slightly wider in order to fit the descriptions.
- What method would I need to use to remove a habit from the habit array?
- Logic for ensuring both a name a description are provided when creating a new habit?
- Scale up the entire program in the styling so that there is less white space.
- What would the process be to choose different sounds for each habit when checking the completed box? How would I allow the user to select from a list of sound effects?

Habit Tracker

Add Habit

Your Habits

Push-Ups 25 a day.	Completed: 0 days	Delete	Check Off
Leetcode 1 question a day.	Completed: 0 days	Delete	Check Off

ulyscd / HabitTracker

Q Type ↵ to search

<> Code

🕒 Issues

🔗 Pull requests

🎬 Actions

📁 Projects

📖 Wiki

🛡 Security

📈 Insights

⚙ Settings

HabitTracker

Public

📌 Pin

👁 Watch 0

🔗 main ▾

🔗 1 Branch

🏷 0 Tags

🔍 Go to file

t

Add file ▾

<> Code ▾

ulyscd

Add files via upload

ddfec9 · now

🕒 1 Commit

📁 public

Add files via upload

now

📄 p6-server.js

Add files via upload

now

📖 README

📖

Add a README

Help people interested in this repository understand your project by adding a README.

Add a README

JS p6-server.js x < index.html # style.css JS script.js

p6 > p6 > JS p6-server.js > app.delete("/habits/:id") callback

```
32 app.put("/habits/:id", (req, res) => {
33     const habit = habits.find(h => h.id === habitId);
34
35     if (!habit) {
36         return res.status(404).json({ error: "Habit not found" });
37     }
38
39     habit.completedDays += 1;
40     res.json(habit);
41 });
42
43
44 app.delete("/habits/:id", (req, res) => {
45     const habitId = parseInt(req.params.id);
46     const habitIndex = habits.findIndex(h => h.id === habitId);
47
48     if (habitIndex === -1) {
49         return res.status(404).json({ error: "Habit not found" });
50     }
51
52     habits.splice(habitIndex, 1);
53     res.status(204).send();
54 });
55
56 app.use((req, res) => {
57     res.status(404).json({ error: "Route not found" });
58 });
59
60 app.listen(port, () => {
61     console.log(`Server is running at http://localhost:${port}`);
62 });
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

node - p6 + v ... ^ x

ulyscd@dyn-10-108-19-191 p6 % node p6-server.js
Server is running at http://localhost:4000
^