



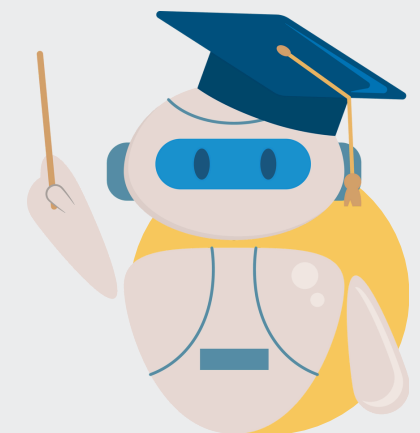
Chat Bot



What you need to do:

1. Make a backend server:

- o Use Node.js with Express (a tool that helps build websites).
- o Use Socket.IO to send messages in real-time between people.
- o Save all messages in a file (called messages.json) so messages stay even if the server restarts.
- o When someone sends a message, save it to the file and send it to all other users immediately.





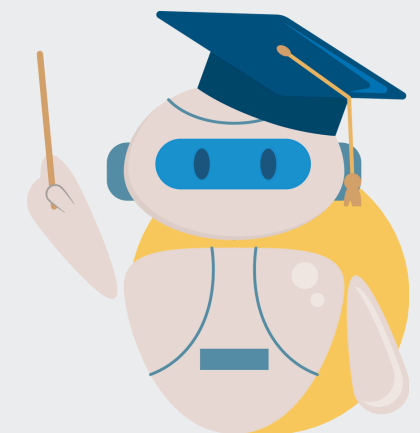
Chat Bot



What you need to do:

2. Make a frontend page:

- o Build a webpage using HTML, CSS, and JavaScript.
- o Show all old messages when the page loads.
- o Show new messages instantly when someone sends them.
- o Let users type a message and send it by clicking a button.
- o Show the username, message text, and time for each message.





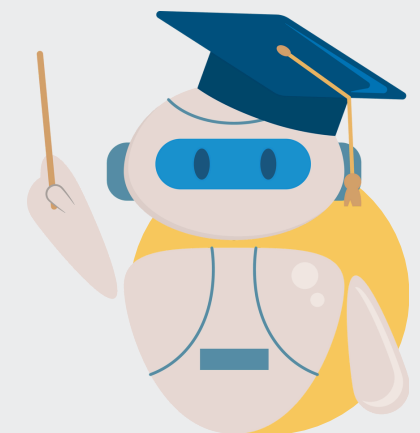
Chat Bot



What you need to do:

3. How files are organized:

- o server.js — the backend program
- o public/index.html — the webpage code
- o public/style.css — styling for the webpage
- o public/script.js — code that makes the page interactive
- o messages.json — the file where messages are saved



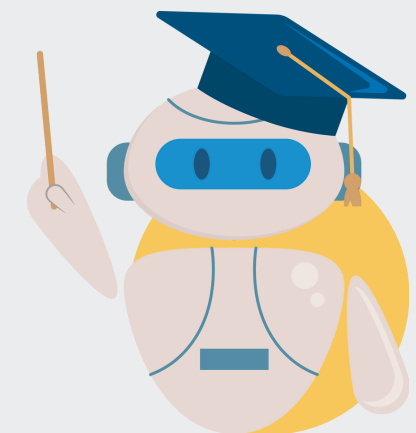


Chat Bot



How to run the app:

- Open the terminal (command line).
- Type `npm install` to get necessary tools.
- Run the server with `node server.js`.
- Open the browser and go to `http://localhost:3000` to see your chat app.



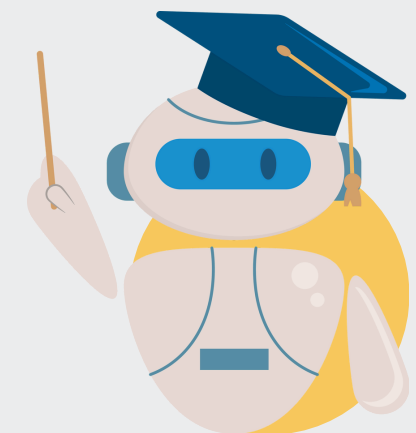


Chat Bot



What you will learn:

- How to build a program that talks to many people at once.
- How to save and read data in a file.
- How to build a webpage that updates automatically without refreshing.





Chat Bot



Tips for using Cursor:

- Ask Cursor to help write or fix any file (server.js, index.html, style.css, script.js).
- Use Cursor to explain any code you don't understand.
- Test your app often to see how it works.

