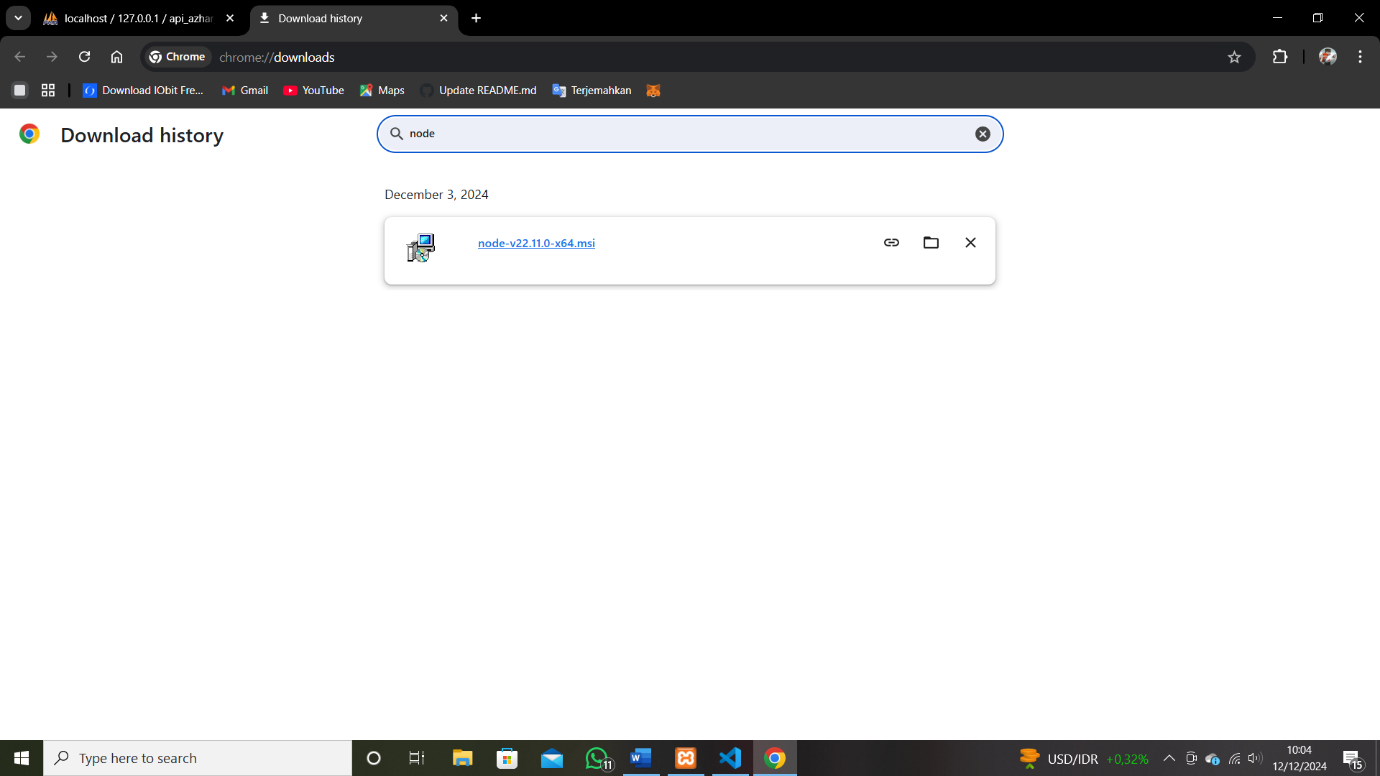
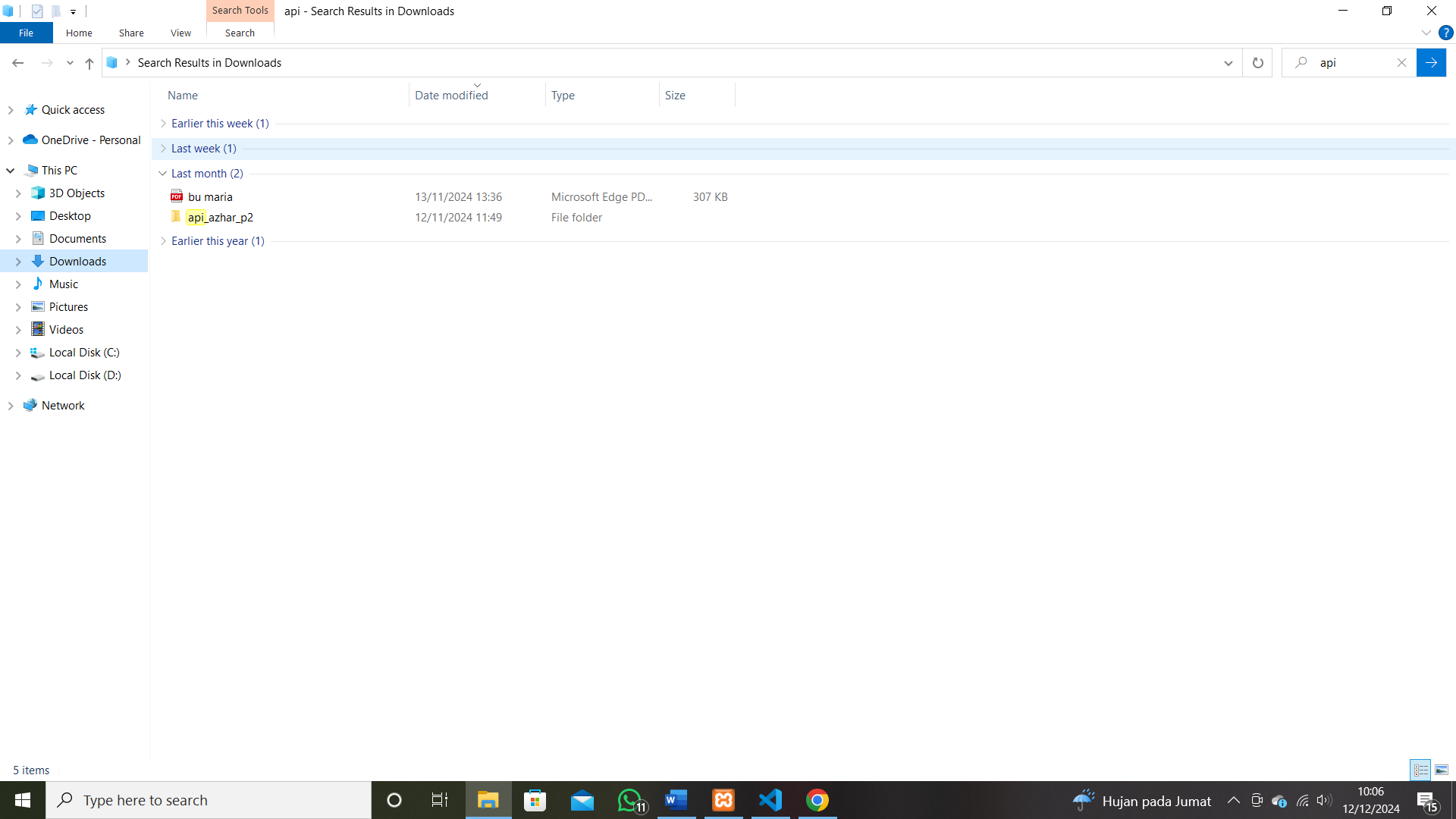
LANGKAH LANGKAH MEMBUAT API VIVA FANTASY

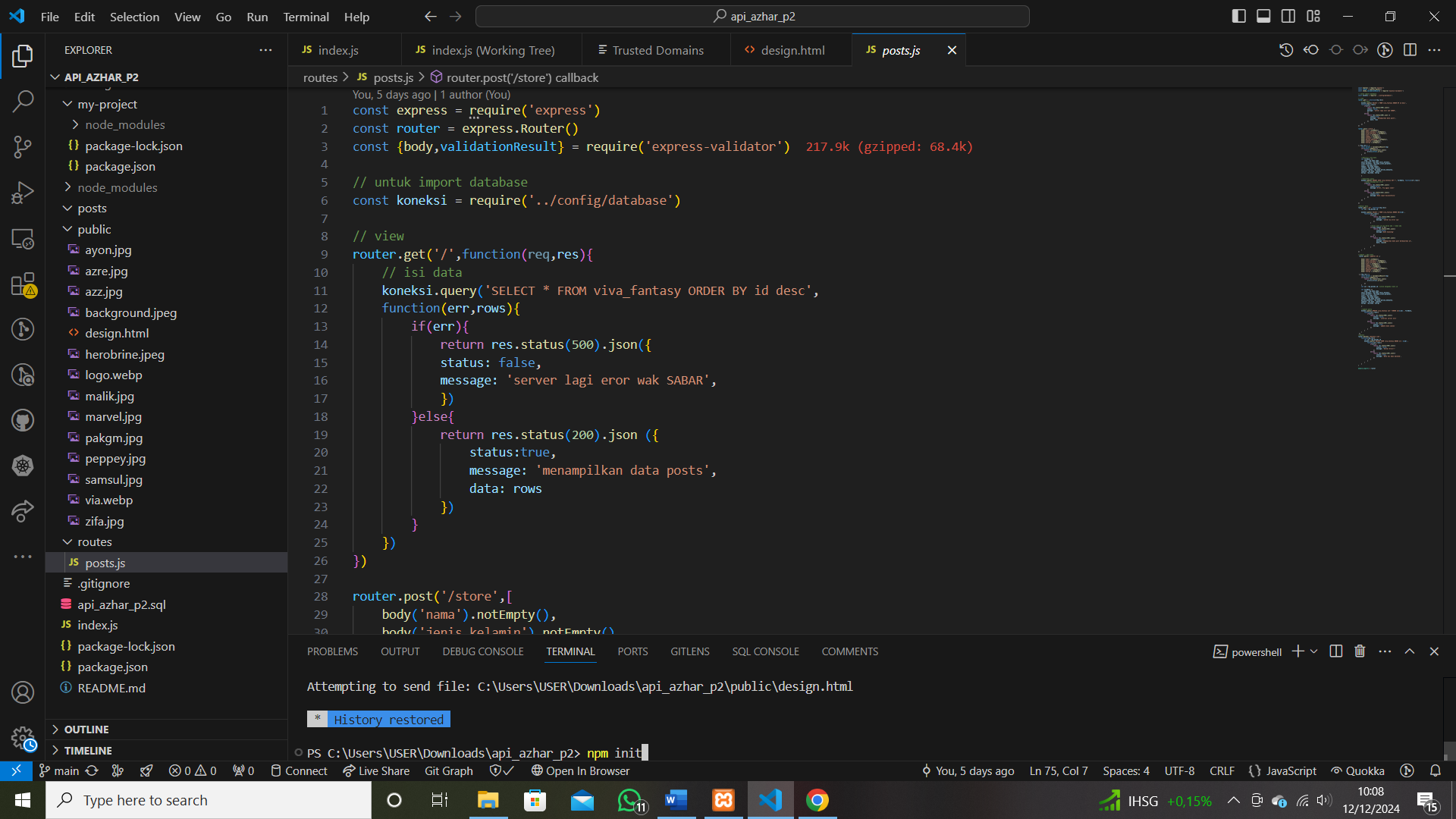
1. **Unduh Node.js**: Pertama-tama, saya mulai dengan mengunduh Node.js.



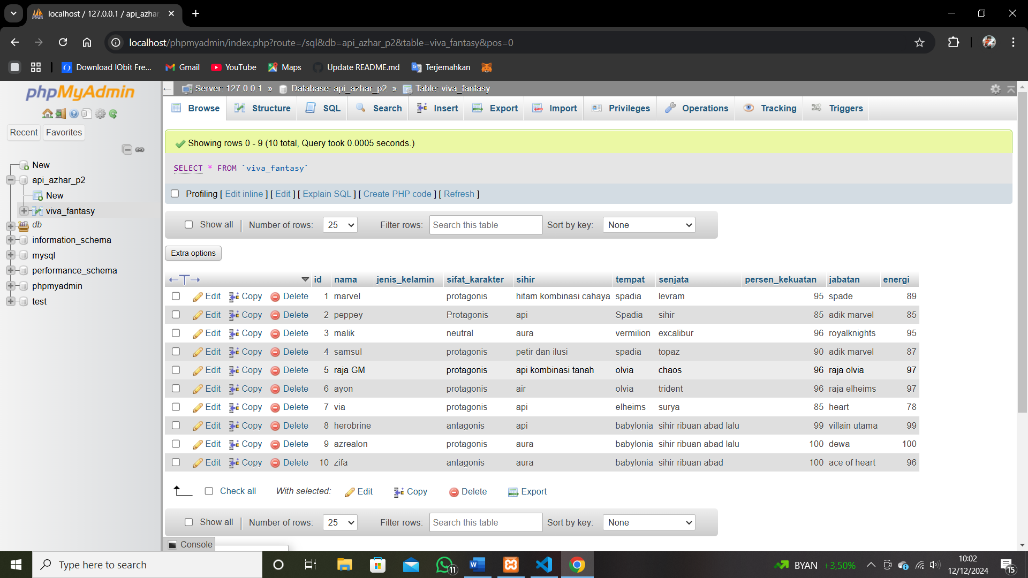
1. **Buat Folder untuk API**: Setelah itu, saya buat folder baru di komputer saya dan menamakannya "API\_azhar\_uas". Ini akan jadi tempat saya menyimpan semua file yang saya butuhkan.



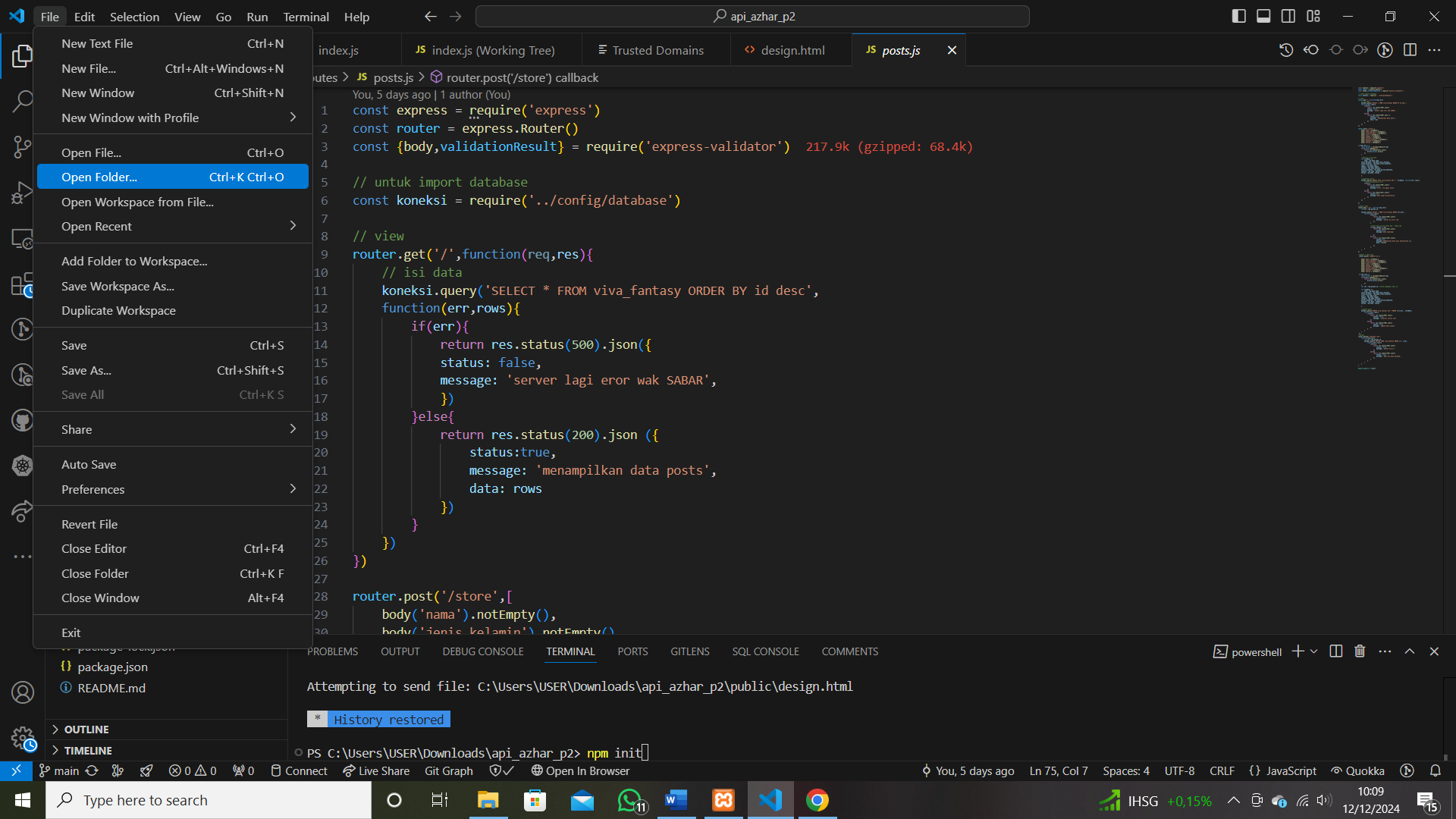
1. **Inisialisasi npm**: Selanjutnya, saya buka terminal di dalam folder itu dan menjalankan perintah **npm init**. Proses ini membantu saya membuat file **package.json**, berisi informasi tentang proyek.



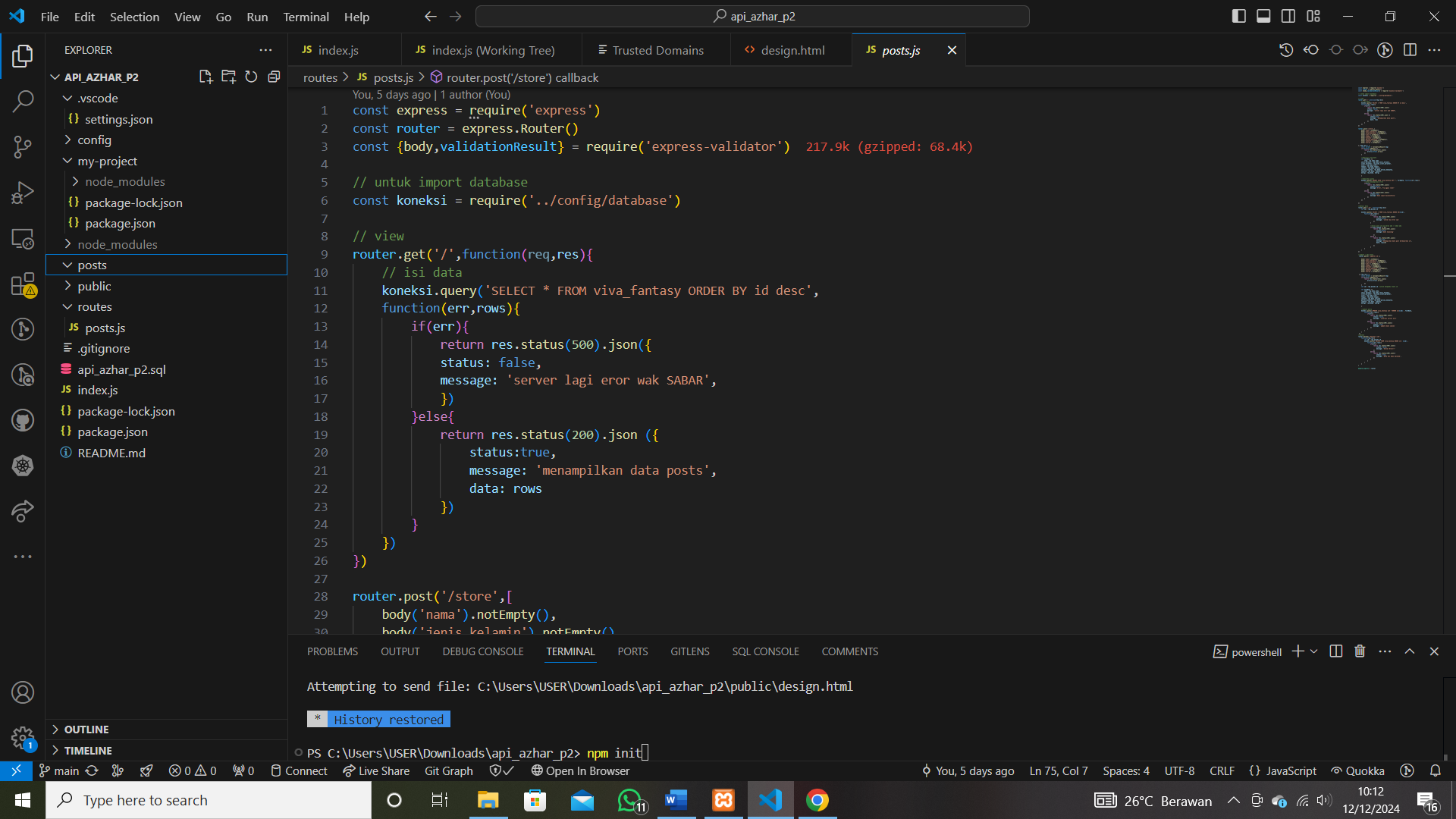
1. **Siapkan Database dan Tabel**: Saya juga menyiapkan database dan tabel dari php myadmin.



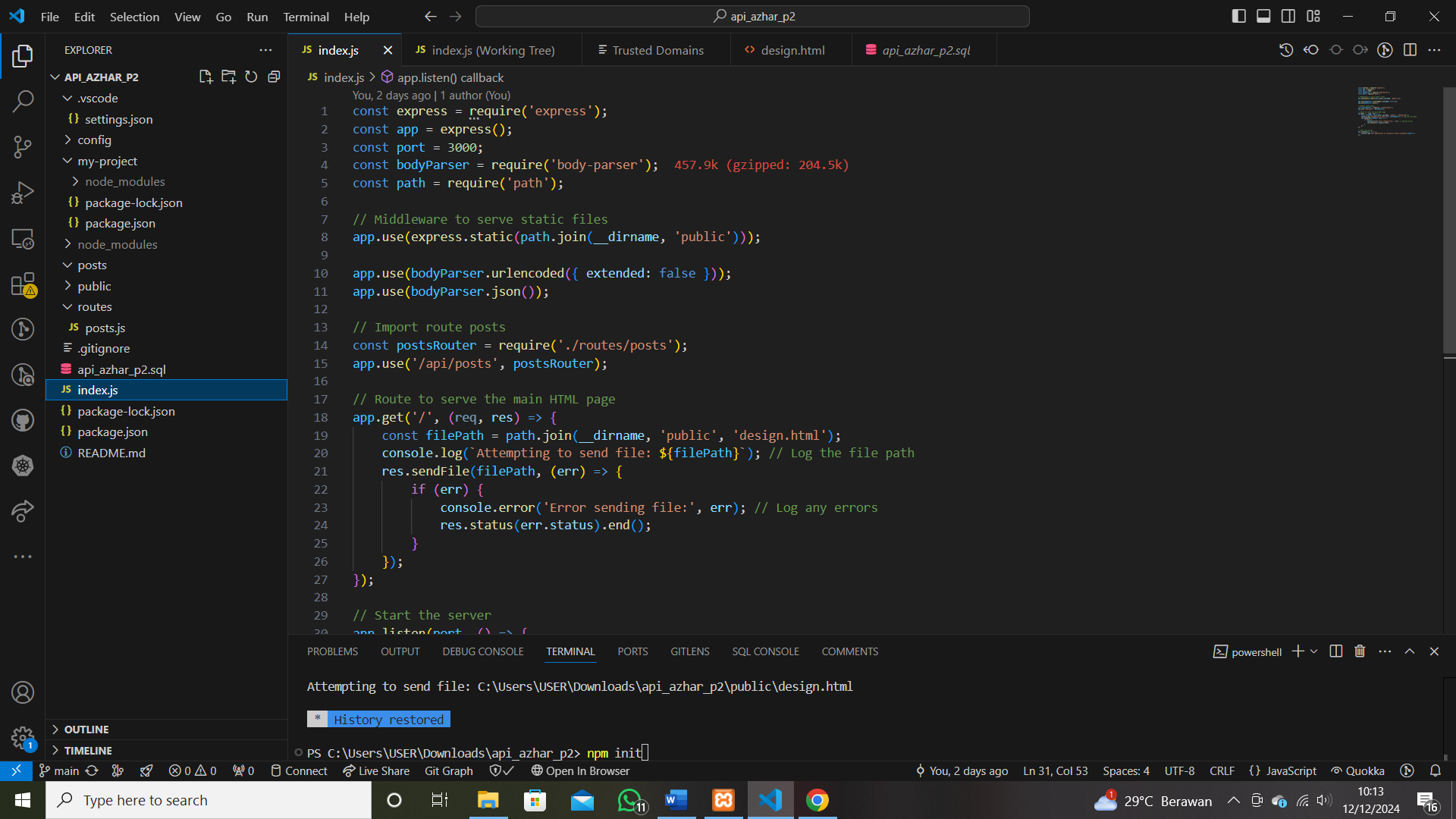
1. **Buka Folder di Visual Studio Code**: Setelah semua siap, saya buka folder "API" di Visual Studio Code.



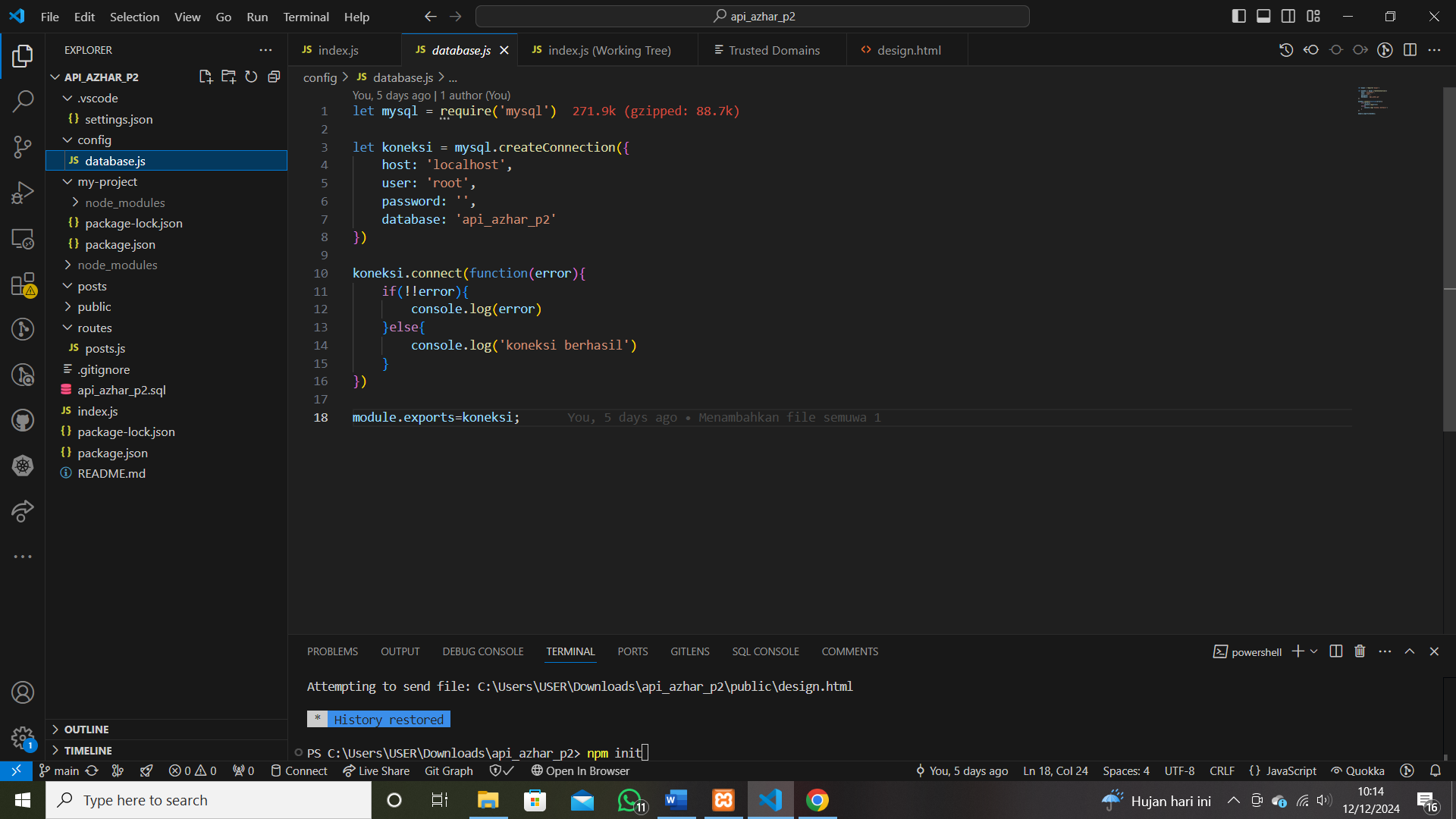
1. **Instalasi Paket yang Diperlukan**: Di terminal, saya mulai menginstal beberapa paket yang diperlukan, seperti Express dan body-parser. Saya menggunakan perintah **npm install <nama-paket> --save** untuk menambahkannya ke proyek saya.
2. **Buat File index.js**: Saya buat file baru bernama **index.js**. Di sini saya menulis kode untuk mengatur server dan rute API.
3. **Buat Folder Config**: Saya juga membuat folder baru bernama "config" untuk menyimpan file **database.js**
4. **Buat Folder Routes**: Selanjutnya, saya buat folder bernama "routes" untuk menyimpan file **posts.js**

6-9

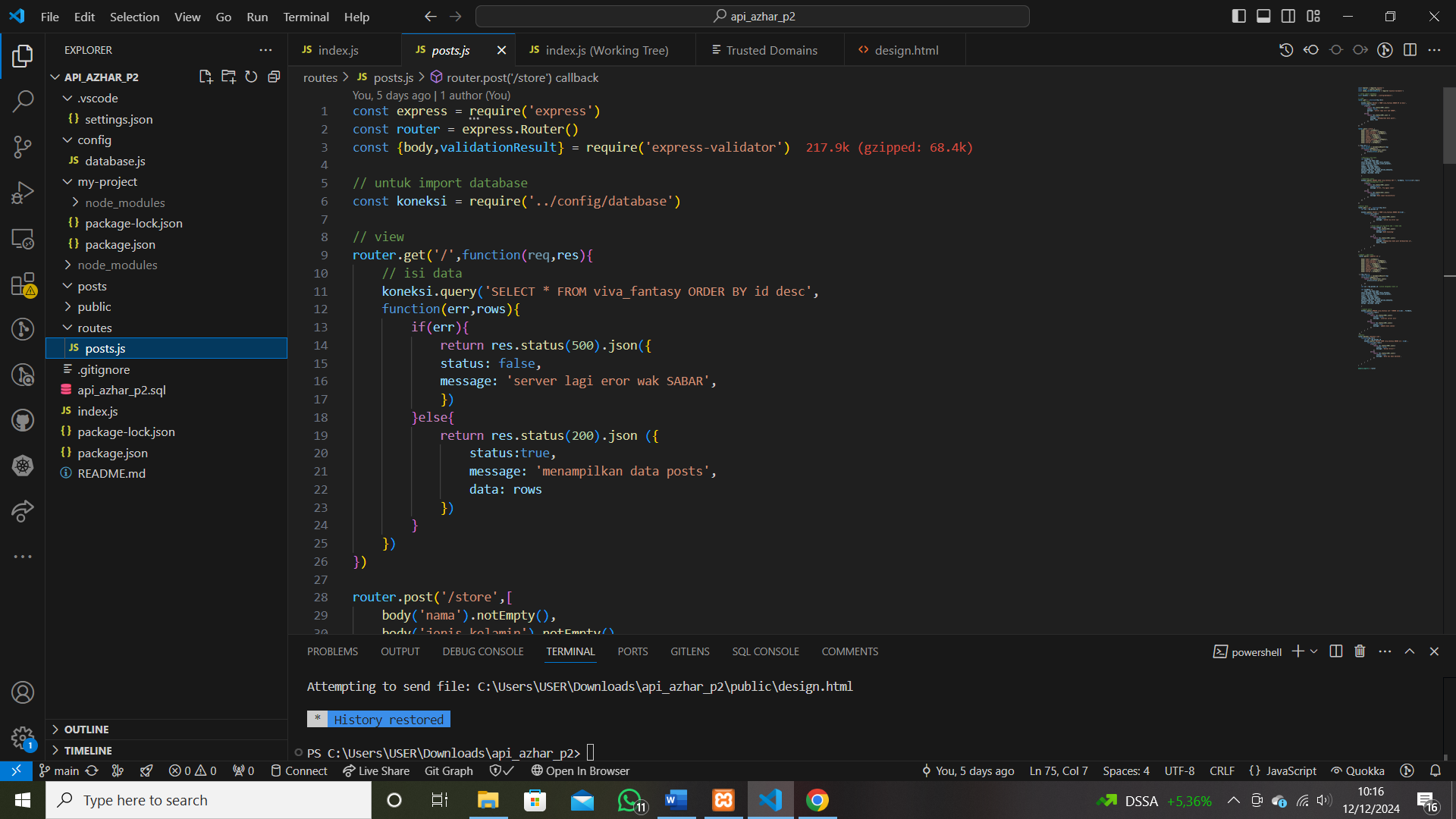
1. **Tulis Kode di index.js**: Di file **index.js**, saya mulai menulis kode untuk mengatur server.



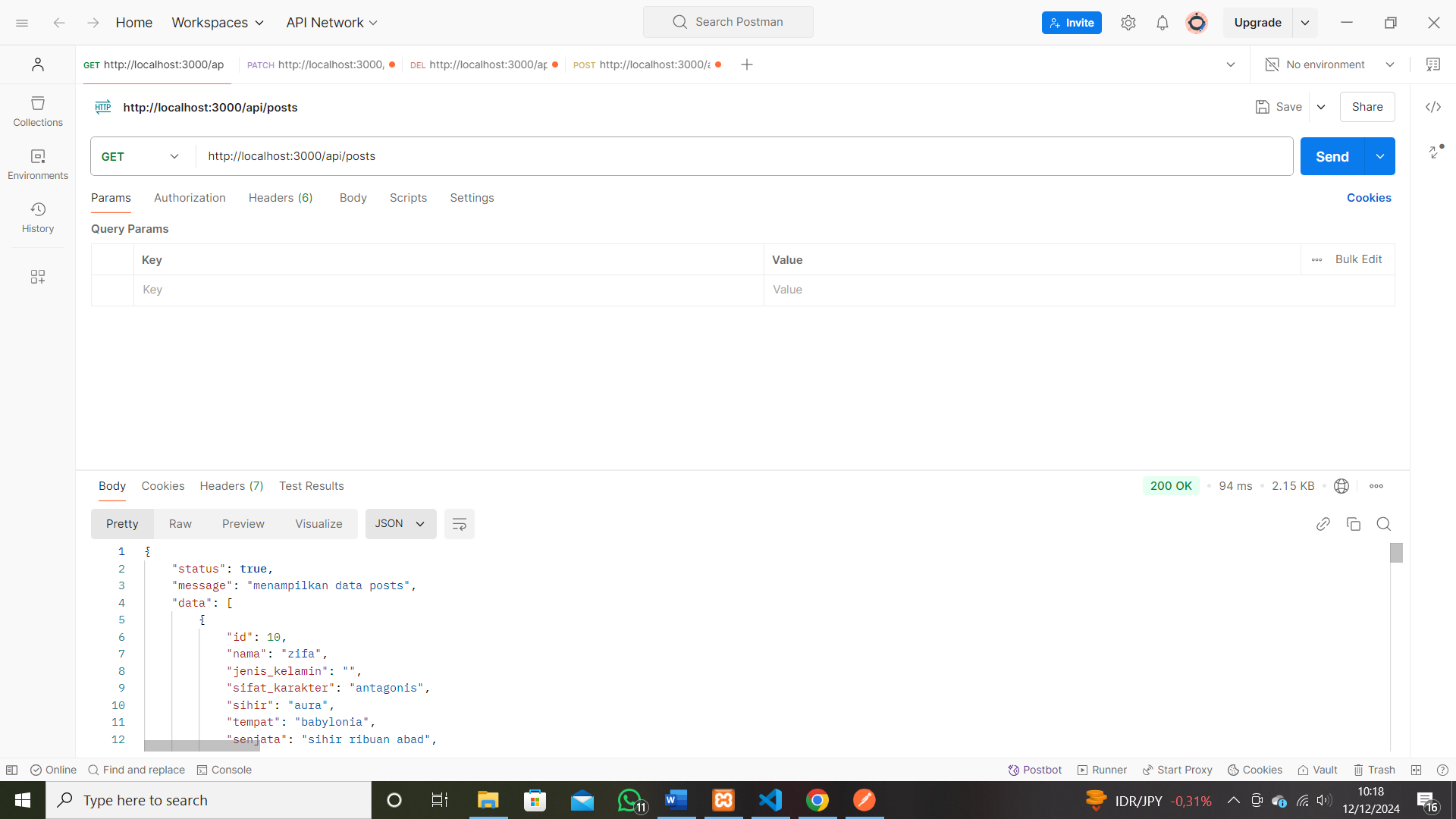
1. **Koneksi ke MySQL di database.js**: Di file **database.js**, saya menulis kode untuk menghubungkan aplikasi dengan database MySQL

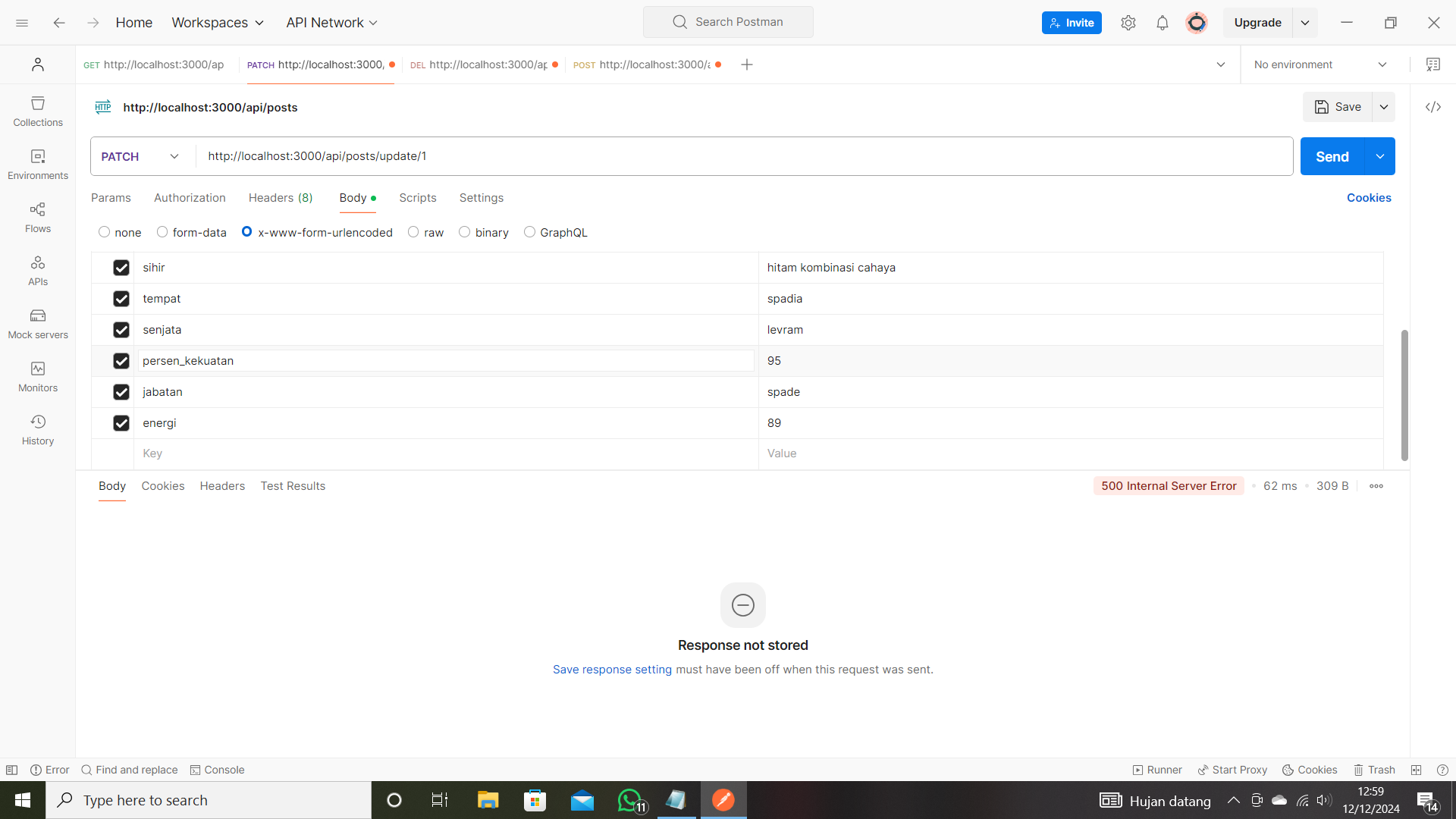


1. **Tulis Kode di posts.js**: Di file **posts.js**, saya membuat fungsi untuk menampilkan, menambah, memperbarui, dan menghapus data.

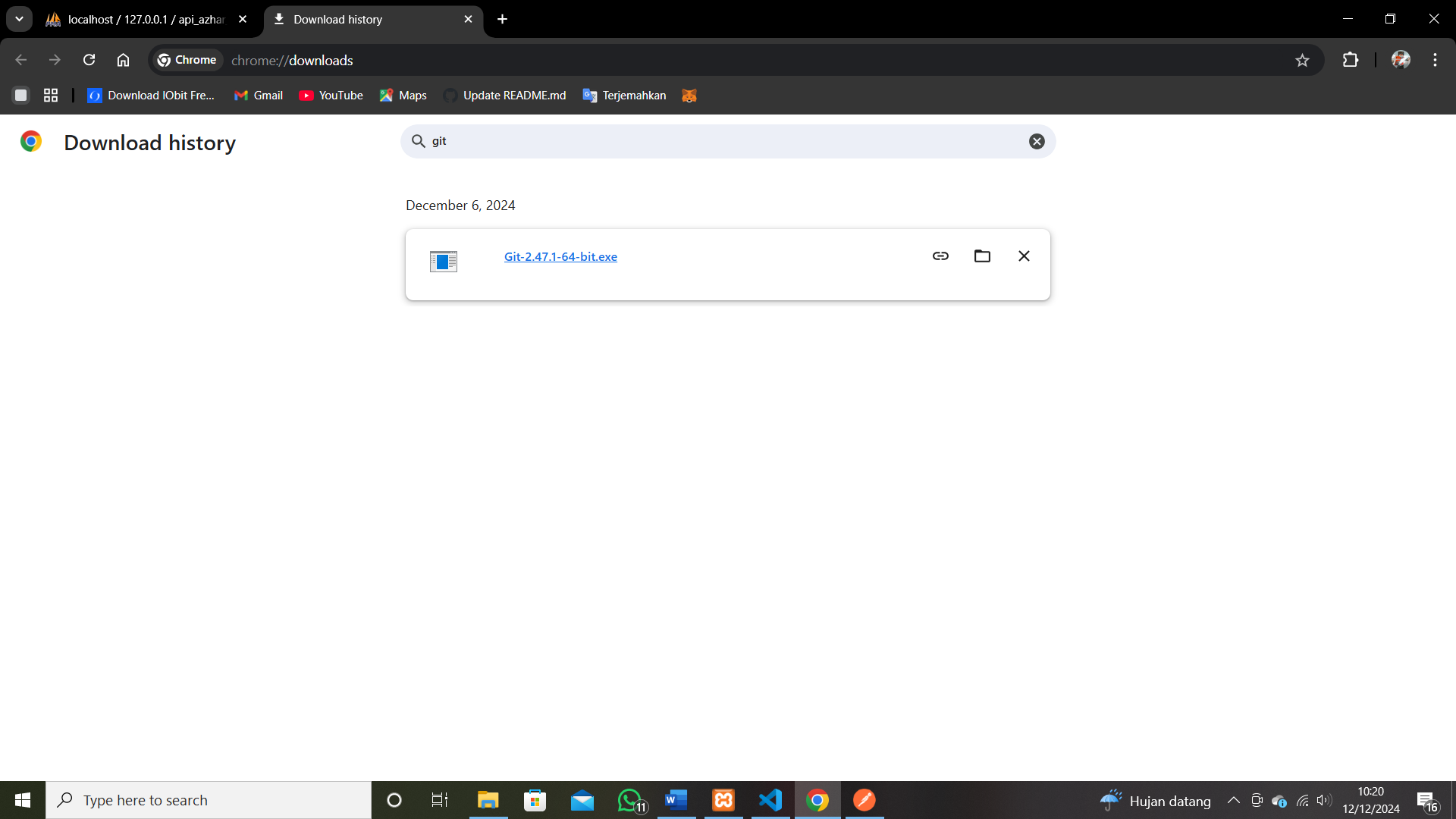


1. **Jalankan Aplikasi di Terminal**: Setelah semua siap, saya jalankan perintah **nodemon npm start** di terminal untuk melihat apakah aplikasi saya berjalan dengan baik. Jika ada kesalahan, saya bisa langsung memperbaikinya.
2. **Buka Postman**: Sekarang, saya buka Postman. untuk menguji API yang saya buat. Dengan Postman, saya bisa mengirim permintaan dan melihat respons dari server.
3. **Uji Coba API**: Saya mulai menguji semua fungsi API yang sudah saya buat.memastikan semuanya berjalan sesuai harapan, seperti menampilkan, menambah, memperbarui, dan menghapus data.

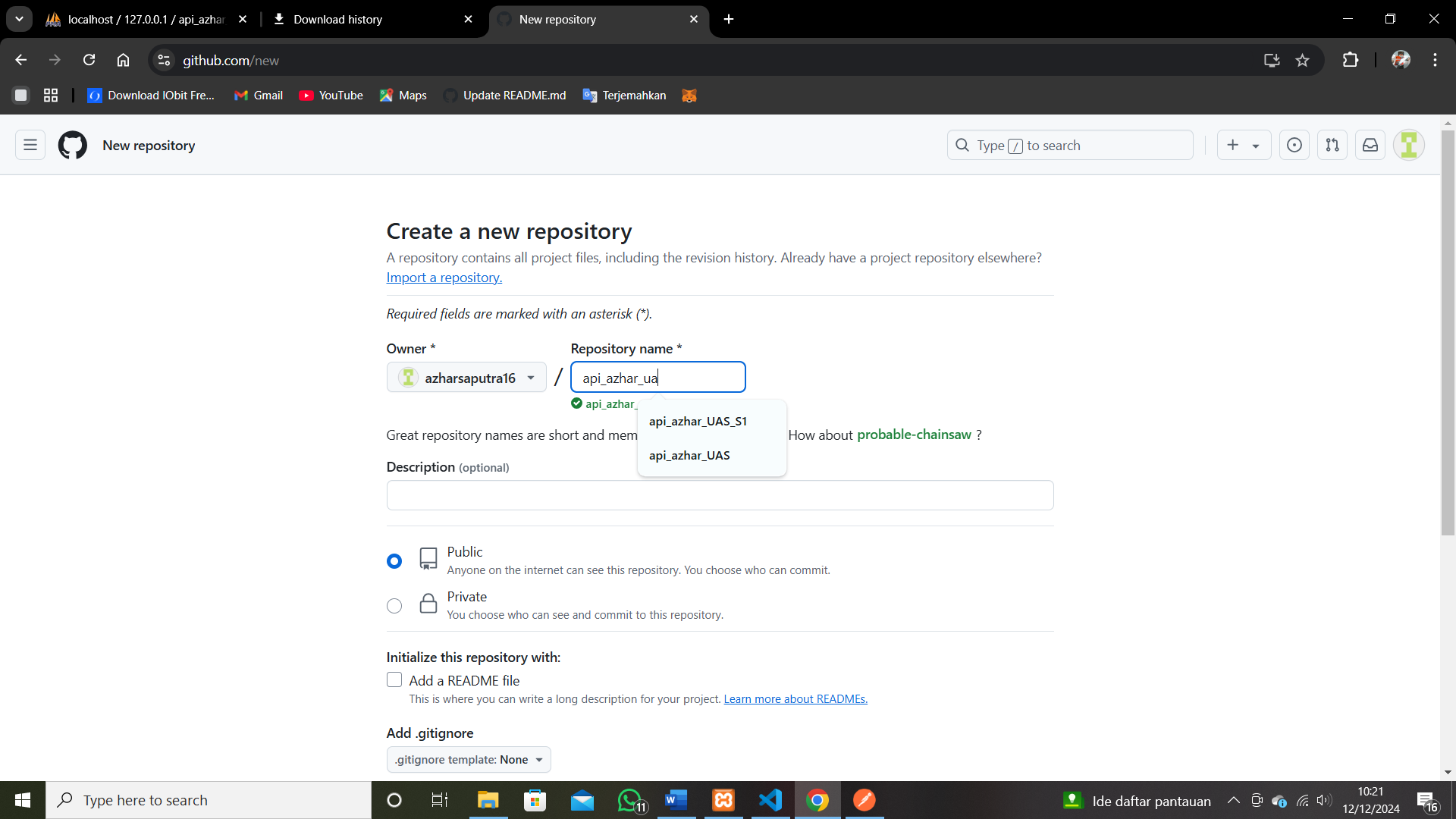


14-15

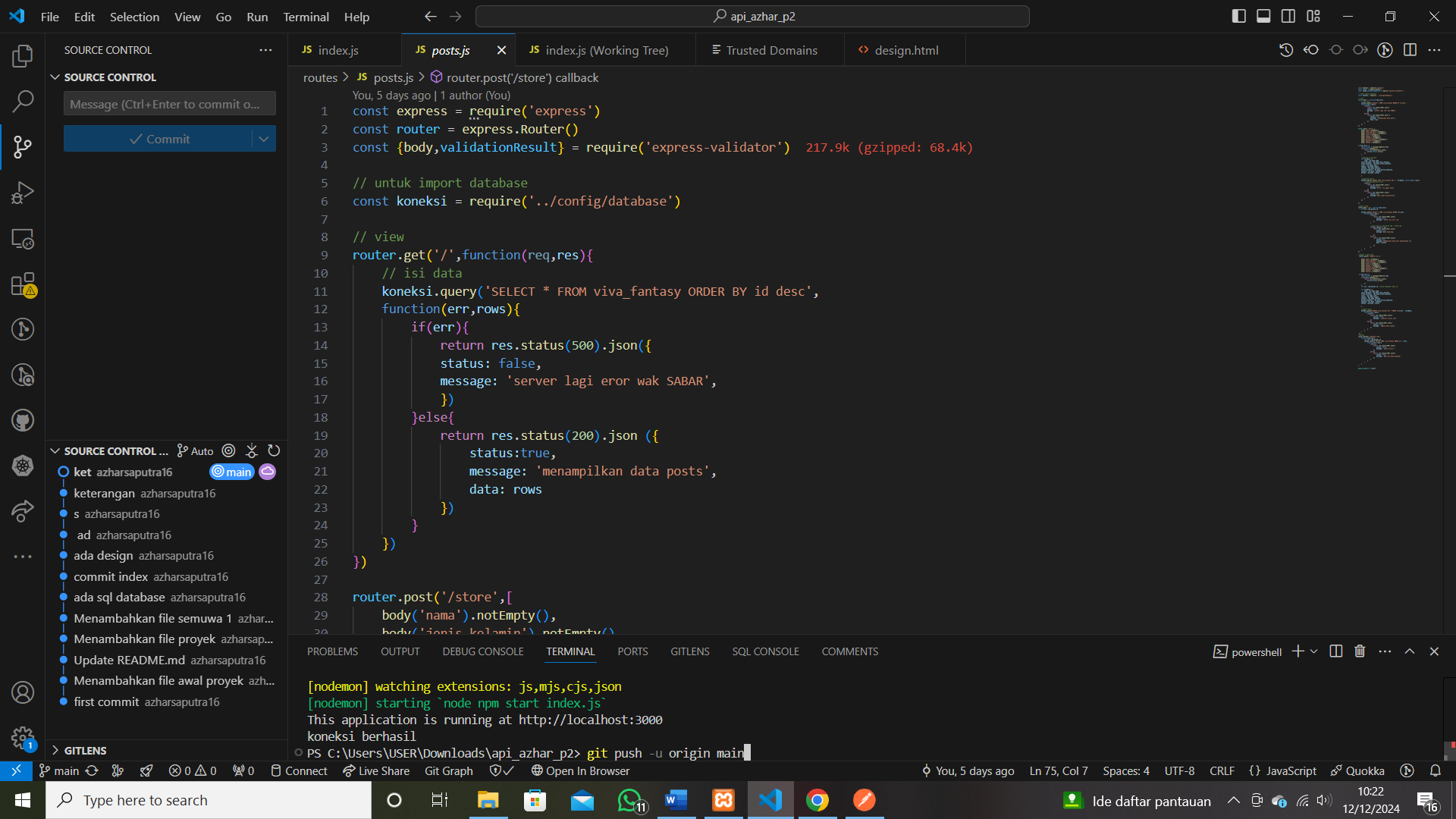
1. **Instal Git dan Login ke GitHub**: menginstal Git dan masuk ke akun GitHub. untuk menyimpan dan berbagi kode.



1. **Buat Repositori di GitHub**: Saya buat repositori baru di GitHub untuk proyek API ini. Ini akan jadi tempat saya menyimpan semua perubahan yang saya buat.



1. **Commit dan Push Perubahan**: Setelah melakukan beberapa perubahan, saya lakukan commit dengan pesan yang jelas. Kemudian, saya push file saya ke repositori GitHub agar semua orang bisa melihat dan mengakses kode saya.

commit

Git add .

Git commit -m “pesan commit”

Git push -u origin main

