🎄

Tree Of Files Fullstack Assignment

Welcome to your Token Security full-stack assignment.

In this assignment, you are tasked with creating a React web app and an HTTP server with a single endpoint. This endpoint should return a tree of files from a GitHub repository to be rendered in the web app.

The site should display an input field where users write a GitHub repository URL (like https://github.com/octocat/Hello-World) and a button. Upon clicking the button, a request is sent to the HTTP server with the repository URL, and the server returns the files in that repository.

Guidelines:

1. For the backend:

    a. Use an HTTP server library (We prefer Python FastAPI, but you can use anything you want) and validate parameters or body (if you opt for the POST method).

    b. **Don't use the recursive option when fetching content from the GitHub API. Instead, build your own recursive function.**

    c. The server should only support requests to render files from **public** GitHub repositories.

    d. There's no need to return the file content, just the file names.

2. For the frontend:

    a. Use React or Angular, preferably with TypeScript

    b. Choose a component library. (E.g antd)

    c. Select a library for HTTP request caching (e.g., react-query)

d. Implement routing

e. Apply typings for server objects (Zod, for example).

- **Before submitting the assignment, ensure you include a README file with instructions on how to run the project.**

Bonus Points (optional):

1. Dockerfiles for the backend and frontend, and a Docker compose file to run them
2. Incorporate a DB/cache layer to save the trees for faster load times.
3. JS frontend tooling system (like vite)
4. Login page
5. Icons for files/directories
6. Add a new endpoint to the server that sends back the content of a file. If a user clicks on a filename in the tree, a modal should appear with the content of the file.