**Laborator 11 IoT – documentatie**

Creati un serviciu RESTful simplu care permite gestionarea unui set de fisiere continut intr-un director. Se doresc urmatoarele functionalitati:

1. **Listarea continutului directorului (GET)**

**Structura** Structura proiectului meu este urmatoarea:

project/

│

├── app.py           ← serverul Flask

├── files/           ← directorul unde sunt pastrate fișierele

└── index.html       ← pagina HTML

Când accesez GET http://localhost:5000/files din browser sau cu fetch() din HTML, serverul va răspunde cu o listă JSON a fișierelor existente în directorul files/ cum se vede mai jos.

A screenshot of a computer

AI-generated content may be incorrect.

Dacă HTML-ul este servit de pe alt port decât Flask (ex: port 8000 pentru HTML, 5000 pentru API), browserul va bloca cererea dacă serverul Flask nu acceptă CORS. De accea trebuie inclus CROS(app) in aplicatie.

Acesta de mai jos este folderul files in care sunt incluse fisierele care urmeaza a fi listate:

A screenshot of a computer

AI-generated content may be incorrect.

Acum rulez un server local pentru HTML intr-un terminal nou:

python -m http.server 8000

Apoi deschid browserul și accesez:

<http://localhost:8000/index.html>

Apasand pe butonul List Files se vor lista toate fisierele din folderul files, dupa cum se poate vedea mai jos:

A screenshot of a computer

AI-generated content may be incorrect.

1. **Listarea continutului unui fisier text, specificat prin nume (GET)**

Accesand in browser: [127.0.0.1:5000/files/text.txt](http://127.0.0.1:5000/files/text.txt) , vom obtine:

**A computer screen with a black background

AI-generated content may be incorrect.**

Apoi in pagina HTML trebuie introdus ca si input fisierul txt care se doreste a fi citit si se va afisa continutul acelui fisier, cum se poate vedea mai jos:

A screenshot of a computer

AI-generated content may be incorrect.

In caz contrar, va afisa eroare:

A screenshot of a computer

AI-generated content may be incorrect.

1. **Crearea unui fisier specificat prin nume si continut (POST)**

In prima caseta la punctul 3 se va adauga numele fisierului care urmeaza sa fie creiat sau suprascris. In a doua caseta se adauga continutul fisierului. Daca fisierul deja exista vom avea un mesaj de eroare, altfel se va creia un fisier nou cu continutul specificat

A screenshot of a computer

AI-generated content may be incorrect.

In imaginea de mai jos se poate vedea fisierul nou creiat.

A screenshot of a computer

AI-generated content may be incorrect.

1. **Crearea unui fisier specificat prin continut (POST)**

Cu ajutorul metodei POST la acest subpunct se pot creia fisiere doar prins specificarea continutului fisierului, iar numele acestora este generat in mod automat.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

1. **Stergerea unui fisier specificat prin nume (DELETE)**

Cu ajutorul acestei metode am reusit sa implementez functia care va sterge un fisier din folderul meu specificat prin nume. Daca fisierul nu este gasit va da eroare: File not found.

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

1. **Modificarea continutului unui fisier specificat prin nume (PUT)**

Pentru acest subpunct am creiat o noua functie cu ajutorul metodei PUT prin care se va face update la continutul unui fisier in cazul in care acesta exista, alfel vom avea eroare.

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**