

Tema 1 – DATC

REST - Representational State Transfer

REST este un stil arhitectural pentru furnizarea de standarde între sistemele informatice de pe web, facilitând astfel comunicarea sistemelor între ele. Nu este un standard, ci un set de constrângeri, cum ar fi: este fără stare (stateless), are o relație client/server și o interfață uniformă. REST nu este strict legat de HTTP, dar este de obicei asociat cu acesta.

Separarea dintre client și server

În stilul arhitectural REST, implementarea clientului și implementarea serverului se pot face independent fără ca fiecare să știe despre celălalt. Aceasta înseamnă că codul de pe partea clientului poate fi modificat în orice moment fără a afecta funcționarea serverului, și vice versa.

Apatriditate (Statelessness)

Sistemele care urmăresc paradigma REST sunt apatrizii, ceea ce înseamnă că serverul nu are nevoie să știe nimic despre starea în care se află clientul și vice versa. În acest fel, serverul și clientul pot înțelege orice mesaj primit, chiar fără a vedea mesajele anterioare.

Comunicarea între client și server

În arhitectura REST, clienții trimit solicitări de recuperare sau modificare a resurselor, iar serverele trimit răspunsuri la aceste solicitări. REST cere ca un client să facă o cerere către server pentru a prelua sau a modifica datele de pe server. O solicitare în general constă în:

- un verb HTTP, care definește ce fel de operație trebuie efectuată
- un antet, care permite clientului să transmită informații despre cerere
- cale către o resursă
- un corp opțional de mesaj care conține date

Există 4 verbe HTTP de bază pe care le folosim în cererile de interacțiune cu resurse într-un sistem REST:

- GET - preia o anumită resursă (prin id) sau o colecție de resurse
- POST - creează o nouă resursă
- PUT - actualizează o resursă specifică (după id)
- DELETE - elimină o resursă specifică prin id

În antetul solicitării, clientul trimite tipul de conținut pe care îl poate primi de la server. Acesta este numit câmpul Accept și asigură faptul că serverul nu trimite date care nu pot fi înțelese sau procesate de client. Solicitățile trebuie să conțină o cale spre o resursă pe care trebuie efectuată operația.

Un avantaj primordial al utilizării REST, atât din perspectiva unui client cât și a serverului, sunt interacțiunile bazate pe REST, care se întâmplă utilizând constructe cunoscute de oricine este obișnuit să utilizeze HTTP.