

Cerințe proiect Programarea aplicațiilor în NodeJS

Proiectul final pentru optionalul “Programarea aplicațiilor în NodeJS” consta în crearea și prezentarea unui **REST API** care să îndeplinească anumite cerințe enunțate în baremul de mai jos. Există atât cerințe obligatorii pentru nota de trecere, adică 5, cât și cerințe opționale pentru a crește nota până la maximum de 10.

Proiectul trebuie să fie gândit ca o aplicație care are un scop bine definit și acoperă o nevoie reală pentru un anumit utilizator țintă. Se va ține cont de congruența și interdependența dintre API-uri în a atinge acest scop. Altfel spus, nu ne dorim “vanatoare de puncte” în care să bifăm anumite cerințe doar de dragul de a le bifa.

Obligatoriu pentru nota 5:

- Implementarea operațiilor CRUD (create, read, update, delete) pentru cel puțin **3 entități** care au o relație între ele (e.g. entitatea User și entitatea Post au o relație de one-to-many, adică un user are mai multe postări)
- Salvarea datelor într-o bază de date
 - Folosirea unui **ORM**
- **Rute private și rute publice**
 - Pentru rute private este nevoie de autentificare, cel mai simplu ar fi cu JWT
 - Implicit trebuie o metodă de signup și signin pentru a face rost de un JWT
- **Error handling**
- **Middleware** care validează payload/parametrii request-ului
- **Explicați** unul sau mai multe **flow-uri** prin care un potențial client (e.g. o aplicație web) ar trece pentru a folosi aplicația
 - E.g: Aplicație de rețete: Userul se înregistrează în aplicație, după care poate vedea rețetele altor utilizatori, poate să voteze și să lase comentarii.

Optional:

- Unit tests
 - Cel puțin 4 unit tests **(1 punct)**
- Implementarea unor operații CRUD mai avansate **(pentru maxim 2 puncte)**, de ales dintre:
 - Filtrare + Sortare **(1 punct)**
 - Paginare **(1 punct)**
 - Tranzacție unde se înserează/updatează mai multe entități interdependente (exemplu: împrumut o carte de la bibliotecă) **(1 punct)**
- Implementarea și folosirea unor custom middlewares **(0.5 puncte per middleware, maxim 1 punct)**
 - Implementarea și folosirea implică să faceți voi unul de la 0, nu să folosiți middlewares gata făcute din alte librării

- Autentificare avansata (**1 punct la alegere**)
 - Role based access control cel puțin 2-3 roluri
 - Integrare cu OAuth
- Documentatie (**1 punct**)
 - README complet, descrierea aplicației, pașii de setup, explicarea env variables
 - Diagrama bazei de date
 - Documentarea API-ului (e.g. puteți folosi Swagger)
- Flow Chart Diagram (**0.5 puncte**)
 - O diagrama care arată flow-urile principale prin care API-ul vostru va fi folosit de către un client
- Image upload (**1 punct**)
 - Multipart/Form-Data sau binary data
 - Salvarea unei referințe a imaginii în baza de date
- Cloud deploy (**0.5 puncte**)
 - Deploy al aplicației (Node.JS + baza de date) pe orice serviciu cloud (e.g. Heroku, AWS, Azure, Vercel, etc.)
 - Multe astfel de servicii oferă free trials pentru proiecte de tip hobby

Aplicația se va prezenta unuia dintre cei doi profesori în sesiune, ar fi indicat să pregătiți în Postman request uri, să aveți un database viewer prin care să ne arătați cum se modifică datele în baza de date, și alte tools care să demonstreze cât mai ușor și rapid corectitudinea cerințelor rezolvate.