

# Zadatak: Razvoj gRPC API-ja za upravljanje korisničkim nalogima

## Cilj zadatka

Razviti jednostavan **gRPC API** koristeći **Node.js** i **gRPC** tehnologiju, koji omogućava:

- Kreiranje novih korisničkih naloga
  - Dobavljanje liste svih korisničkih naloga
- 

## Tehnološki zahtevi

- **Backend:** Node.js, [@grpc/grpc-js](#), Protocol Buffers ([.proto](#))
  - **Baza podataka / skladištenje:** Po izboru:
    - SQLite
    - PostgreSQL
    - MongoDB
    - *ili jednostavan niz u memoriji*
- 

## Funkcionalni zahtevi

### 1. Definisanje Protobuf šeme

Kreirati [.proto](#) fajl sa sledećom strukturom:

 **Tip User:**

- `id` (int32) – automatski generisan
- `name` (string) – obavezno
- `email` (string) – obavezno

#### gRPC metode:

- `GetUsers()` → vraća listu svih korisnika
  - `CreateUser(name, email)` → kreira korisnika i vraća novokreiranog korisnika
- 

## 2. Implementacija gRPC servera

- Kreirati gRPC server koristeći `grpc-js` biblioteku
  - Implementirati metode:
    - `GetUsers`
    - `CreateUser`
  - Korisnici se mogu čuvati:
    - U bazi (SQLite, PostgreSQL, MongoDB)
    - Ili u nizu u memoriji (za jednostavnu implementaciju)
- 

## 3. Implementacija gRPC klijenta

Kreirati jednostavnu Node.js klijentsku aplikaciju koja:

- Poziva `CreateUser` kako bi kreirala korisnika
- Poziva `GetUsers` i ispisuje rezultat (listu korisnika) u konzoli

---

## ☀ Dodatni izazovi (opciono)

- ☒ Validacija email adrese pre kreiranja korisnika
- ☒ Filter po imenu u `GetUsers` metodi (pretraga korisnika po imenu)
- ☒ Dodavanje `DeleteUser` RPC metode

---

## 📦 Predaja zadatka

- 🔗 Push-ovati projekat na **GitHub**
- 📺 Postaviti video demonstraciju (npr. **YouTube**, **Vimeo** ili drugi servis)
- 📄 Napisati **README.md** koji sadrži:
  - Uputstvo za pokretanje servera i klijenta
  - Tehnologije korišćene u projektu
- ✉ Poslati link ka:
  - GitHub repozitorijumu
  - Video demonstraciji