

JABLOTRON - Node.js zadání

Služba monitorování endpointů

Vytvoř REST API JSON Node.js microservice, která umožňuje monitorovat zadané http/https url.

Služba by měla umět:

1. Vytvořit, Upravit a Smazat url na monitorování a Vypsat list monitorovaných URL pro daného uživatele (CRUD)
2. Monitorovat URL na pozadí a zapisovat si status kódy + vrácený payload
3. Pro danou monitorovanou URL dokázat zobrazit seznam 10 posledních monitorovacích výsledků

Datový model:

Toto je námi navrhovaný datový model. můžeš si ho upravit, pokud chceš.

MonitoredEndpoint má:

id: (typ necháme na tobě :-))
název : string
url: string
datum vytvoření : DateTime
datum posledního checku: DateTime
monitoredInterval: Integer (v sekundách)
vlastník: Uživatel

MonitoringResult má:

id: (typ necháme na tobě :-))
datum checku: DateTime
navracený http status kód: Integer
navracený payload: String
monitoredEndpointId: MonitoredEndpoint

Uživatel má

id: (typ necháme na tobě :-))
Jméno : String
Email : String
AccessToken : UUID like String

Pro uživatele, neočekáváme, že budeš mít CRUD, stačí je mít naseedované v databázi

Očekává se:

```
{
  name: "Jablotron",
  email: "info@jablotron.cz",
  accessToken: "93f39e2f-80de-4033-99ee-249d92736a25"
},
{
  name: "Batman",
  email: "batman@example.com",
  accessToken: "dcb20f8a-5657-4f1b-9f7f-ce65739b359e"
}
```

Podrobné zadání:

- Navrhni REST endpointy pro managování **MonitoredEndpoints** (pokud nevíš jak na to, zeptej se nás :-))
- Na pozadí monitoruj zadané endpointy a vytvářej **MonitoringResult**.
- Naimplementuj endpoint pro získání **MonitoringResult**
- Naimplementuj microservice v Node.js frameworku napsanou ideálně v **TypeScriptu** (pokud neumíš/nechceš, napiš ji v ES6 a výše). Jako databázi použij **MySQL**. Jako REST framework použij **Restify**.
- Autentizaci dělej dle tebou zvolené HTTP hlavičky, ve které ti pošleme accessToken
- Autorizace - uživatel může vidět pouze MonitoredEndpoints a Result, pro svoji osobu (dle accessTokenu)
- Nezapomeň na modelové validace (co všechno je třeba validovat necháme na tvém uvážení)
- Napiš na to jednoduché testy v **Mocha** nebo v **Jestu**
- Vše pushni do veřejného repozitáře na GitHubu
- Vyplň readme, kde popíšeš jak službu spustit a používat
- Pošli nám link na Github
- Kód i dokumentaci piš v angličtině
- Bonusový body: Vytvoř Dockerfile, dodej docker-compose a popiš, jak to spustíš v dockeru

Good luck, have fun!