

Otvoreno računarstvo

3. laboratorijska vježba

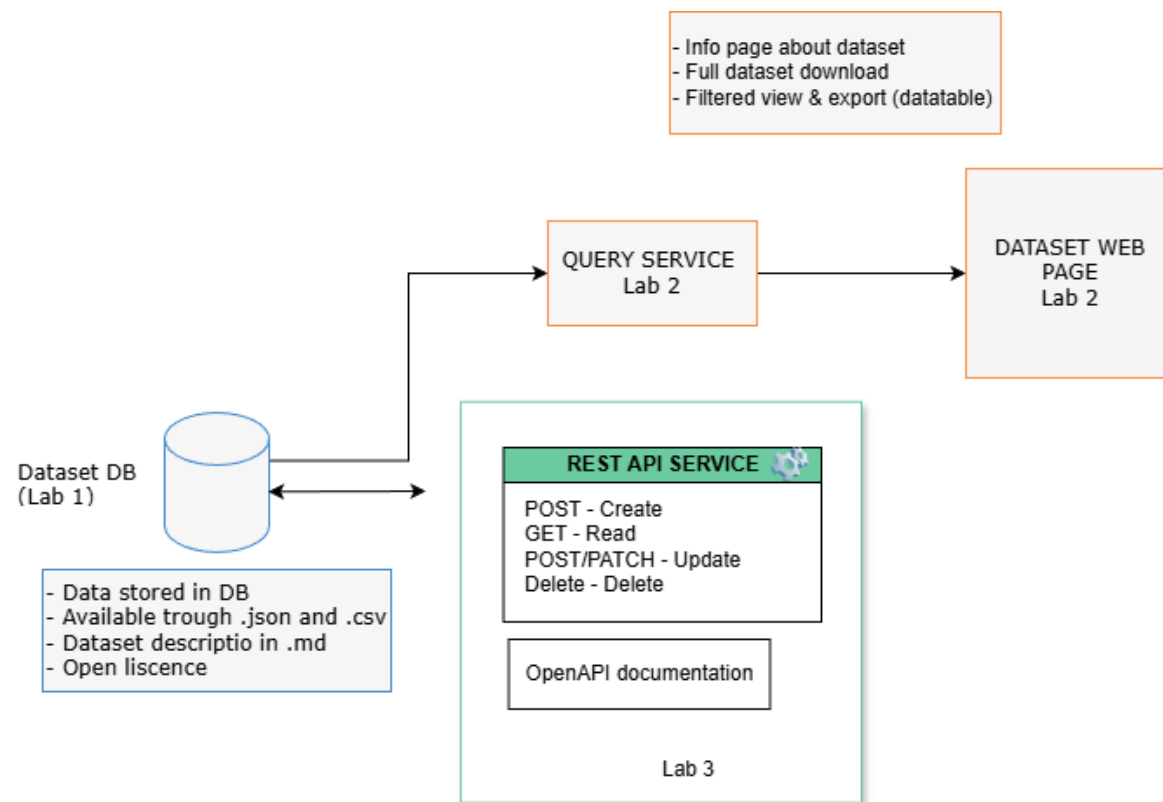
3. laboratorijska vježba

- Cilj vježbe:
 - Izraditi REST API koji izlaže otvoreni skup podataka kroz CRUD operacije.
- Potrebno predznanje:
 - *osnove HTTP-a i REST arhitekture,*
 - *rad s bazama podataka,*
 - *osnovno znanje programskog jezika po izboru (npr. Java, Python, Node.js, ...),*
 - Rok za predaju labosa: nedjelja, 04. 01. 2026. do 23:59
 - Temin labosa: četvrtak, 08.01.2026. po terminima od 08-13h

3. laboratorijska vježba

Gdje smo u laboratorijskim vježbama do sada?

- 1. Lab: Izrada otvorenog skupa podataka (osnovna otvorenost)
- 2. Lab: Izrada web sučelja (pristupačnost prema drugim korisnicima)
- 3. Lab: Izrada REST API-a (pristupačnost prema drugim servisima)



3. laboratorijska vježba

Primjeri skupova podataka otvorenih kroz REST API-e:

– Github REST API:

- Link: <https://docs.github.com/en/rest>
- Primjer: [Sažetak Octocat repozitorija](#)

– Wikimedia REST API:

- Link: https://en.wikipedia.org/api/rest_v1/
- Primjer: [Sažetak Wikipedia stranice o RH](#)

– The RESTful Pokémon API:

- Link: <https://pokeapi.co/>
- Primjer: [Karakteristike pokemona Pikachu](#)

– Open Meteo REST-like API:

- Link: <https://open-meteo.com/en/docs>
- Primjer: [Prognoza temperature ZG](#)

3. laboratorijska vježba

Što čini API RESTlike? (*Richardson maturity model*)

- L1: Bazirano na resursima:
 - /repos/{owner}/{repo}
 - /users/{username}
 - /issues/{id}
- L2: Koriste se HTTP metode:
 - POST - create read (kreiranje resursa)
 - GET – read (dohvaćanje resursa)
 - PUT / PATCH – update (ažuriranje resursa)
 - DELETE – delete (brisanje resursa)
- Uniformiran format odgovora (*wrapped JSON*)

```
POST    /repos/{owner}/{repo}/issues
GET     /repos/{owner}/{repo}/issues/{issue_number}
PATCH  /repos/{owner}/{repo}/issues/{issue_number}
DELETE  /repos/{owner}/{repo}/issues/comments/{comment_id}
```

3. laboratorijska vježba

Što čini API Restful?

- L3: Hypermedia-like linking (HATEOAS)
 - API se može navigirati kroz URL-ove
- *Stateless* / bez spremanje među stanja:
 - Sve ključne informacije se nalaze u zahtjevu i resursu.
 - Nema spremanja među-stanja u mehanizme poput serverskih sesija.

```
{...  
  "url": "https://api.github.com/repos/octocat/Hello-World",  
  "forks_url": "https://api.github.com/repos/octocat/Hello-World/forks",  
  "keys_url": "https://api.github.com/repos/octocat/Hello-World/keys{/key_id}",  
  "collaborators_url": "https://api.github.com/repos/octocat/Hello-World/collaborators{/collaborator}",  
  ...}
```

3. laboratorijska vježba

Kako dokumentirati API-e?

- Postoji mnogo različitih formata za opisivanje API-ja
- Jedan format postao **de facto standard** za dokumentiranje JSON REST API-ja:

OpenAPI

- Širok spektar mogućnosti
- Koristit ćemo ga za **dokumentiranje**
- Primjer: [OpenAPI 3.0 Swagger Petstore](#)

3. laboratorijska vježba - Zadatak

Izraditi REST API za dataset iz ranijih vježbi.

Obavezne krajnje točke:

- GET kolekcije
 - GET pojedinačnog resursa (ID)
 - 3 proizvoljna dodatna GET endpointa
 - POST za dodavanje resursa
 - PUT za ažuriranje resursa
 - DELETE za brisanje prema ID-u
-
- Svaki odgovor mora imati *status kod, poruku i response podatke*.
 - Implementirati robusno **rukovanje greškama** (404, 400, 500...).
 - Krajnje točke i nazivi resursa biraju se slobodno, prema temi.
 - API operacije se izvršavaju nad bazom podataka.

3. laboratorijska vježba - Zadatak

Izraditi OpenAPI specifikaciju (openapi.json).

Mora sadržavati obavezna polja:

- **openapi** – verzija (npr. 3.0.3)
- **info** – osnovni podaci, *contact* i *license*
- **paths** – sve krajnje točke API-ja (GET, POST, PUT, DELETE)

- **Dokumentirati sve obavezne rute** iz REST API-a.
- Za svaku operaciju navesti:
 - **summary / description**
 - **parametre** (npr. ID)
 - **moguće odgovore** (200, 400, 404...)

3. laboratorijska vježba

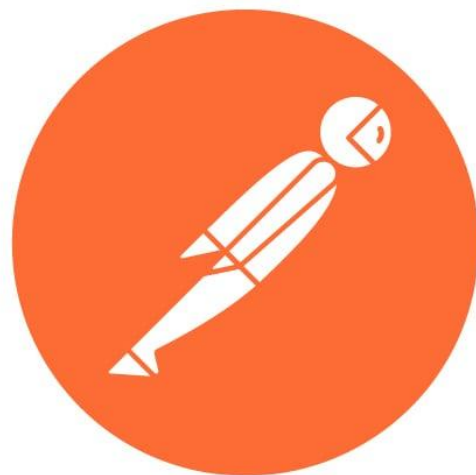
- Predaja vježbe – *Release 3.0*
- Popis datoteka:
 - README.md
 - LICENSE
 - Dataset: naziv_skupa.json, naziv_skupa.csv, shema_skupa.json
 - index.html, datatable.html
 - Dump baze podataka
 - Pozadinski (backend) kôd s implementiranim REST API-jem – LAB 3
 - openapi.json – dokumentacija API-ja prema OpenAPI specifikaciji – LAB3

Pitanja?



Primjer testiranja API-a u Postmanu

[Poveznica za download alata Postman](#)



POSTMAN