



Zadatak

Jezik	PHP
Framework	Symfony 5+
VCS	git
GitHub API Dokumentacija	https://docs.github.com/en/rest/reference/search#search-issues-and-pull-requests

Potrebno je izraditi sustav koji računa popularnost određene riječi. Sustav bi za danu riječ trebao pretražiti GitHub issue koristeći broj rezultata za {riječ} rocks kao pozitivan rezultat i {riječ} sucks kao negativan. Rezultat bi trebao biti ocjena popularnosti zadane riječi od 0-10 kao omjer pozitivnog rezultata i ukupnog broja rezultata. Rezultati bi trebali biti spremljeni u lokalnu bazu kako bi budući upiti za iste riječi bili brži. U budućnosti se očekuje dodavanje/izmjena providera (npr. umjesto GitHuba će se koristiti Twitter) pa bi sustav trebao biti dizajniran u skladu s tim.

Očekivani rezultat

Endpoint

Endpoint koji vraća rezultat u JSON obliku.

npr.

```
GET https://example.com/score?term=php
```

```
{
  term: "php",
  score: 3.33
}
```

Url je samo dan kao primjer te može biti izveden u bilo kojem obliku (path parameter, query parameter, ...). Response je također primjer te može biti u bilo kakvom obliku, ali mora sadržavati minimalno ova 2 polja (term i score).

README

Uz source kod potrebno je napraviti README.md u kojem će biti objašnjen način na koji se JSON API konzumira uz nekoliko praktičnih primjera. Pored toga očekuje se da je dokumentirano postavljanje projekta za daljni razvoj (što učiniti nakon kloniranja repozitorija kako bi mogli početi raditi na projektu).

Generalna uputstva

Projekt je potrebno postaviti kao da se radi o komercijalnom projektu koji će se nastaviti razvijati bez obzira što ima samo 1 endpoint.

Poželjno je da je sustav moguće testirati pokretanjem testova (PHPUnit, Codeception, Behat, ...).

Bonus

Ovaj dio nije nužno rješavati!

Ako se želiš iskazati i dobiti dodatne bodove možeš nam pokazati što znaš i riješiti jedan od dodatnih i zahtjevnijih zadataka.

- OAuth - postaviti osnovni OAuth2 sustav bez korisnika (samo client credentials)
- Uskladiti API s JSONAPI specifikacijom (<http://jsonapi.org/>)
- API v2 - uvesti verzioniranje API-a i napraviti novu verziju endpointa za dohvat rezultata koje odgovara JSONAPI specifikaciji (<http://jsonapi.org/>)
- CI / CD - postaviti osnovni CI/CD sustav koristeći neke od besplatnih CI sustava (npr. GitLab CI, <https://travis-ci.org/>)
- Docker - postaviti projekt koristeći Docker platformu

- Napraviti OpenAPI 3 specifikaciju projekta
- Prije početka rješavanja dati svoje estimate potrebnog vremena za rješavanje
- Tijekom rješavanja uključiti tracking vremena