

PHP Back-End Challenge

Candidate: Houssine (*Huss*)

Summary

Let's start by listing and defining the technology I decided to use in this challenge. To easy up the process I solution is dockerized, the container includes all what is needed to build and test the application!

Technology	Version
Laravel	7
PHP	7.4
Swagger (API specification)	8.4
Nginx	Alpine
MySQL	5.7
PhpMyAdmin	latest

More about the challenge

Sample data:

A seeder named SampleDataSeeder includes a bunch of data to make the application testable and ready to go after building the solution. Structure of the seeder:

- Hospital Clínic de Barcelona
 - **subjects:**
 - Marta Cases
 - Paolo Costa
 - **projects:**
 - Digestive Diseases
 - **Enrolled subjects:** *Paolo and Marta*
 - Musculoskeletal Diseases
 - **Enrolled subjects:** *Paolo*

Description API:

Full documentation (Swagger): <http://localhost:8000/api/documentation>

I create some extra endpoint that I thought would be useful for the testing, like for example the retrieving of all subjects and the project they are attached to. All the endpoints are protected using the protocol OAuth2 except for two routes (/register and /token), the first one lets the user create a new repository and the other create a token. This token will be REQUIRED for all the other endpoints inside the headers as Authorization Bearer <token>, this will work as identifier of who makes the petition. For this reason, is not necessary to include the repository id in the URLs for this scenario, but I decided to follow and maintain the current structure of the core API cmrad.

Build and Run:

- 1) We make sure we are inside the folder cmrad
- 2) Run the following commands sequentially:
 1. docker-compose build app
 2. docker-compose up -d
 3. docker-compose exec app composer install
 4. docker-compose exec app php artisan migrate --seed
 5. docker-compose exec app php artisan passport:install
 - i. You may get (Encryption keys already exist. Use the --force option to overwrite them), its fine.

Tests:

To run some tests I created, run the following command:

- 1) `docker-compose exec app php artisan test`

API collection: (postman)

A postman collection is available at the root folder `cmrad_challenge`, substitute de parameters for the once you want to request, also replace the `<token>` for the token generated by the endpoint `/token`.

URL's

Service	Url
Localhost	localhost:8000 <i>(will redirect to API doc)</i>
API Documentation (Swagger)	localhost:8000/api/documentation
PhpMyAdmin	localhost:8082 <i>(user: huss / pass: secret)</i>