Laravel hiring test

Create a Laravel APIs application for a travel agency.

Glossary

- **Travel** is the main unit of the project: it contains all the necessary information, like the number of days, the images, title, etc. An example is `Japan: road to Wonder` or `Norway: the land of the ICE`;
- **Tour** is a specific dates-range of a travel with its own price and details. `Japan: road to Wonder` may have a *tour* from 10 to 27 May at €1899, another one from 10 to 15 September at €669 etc. At the end, you will book a *tour*, not a *travel*.

Goals

At the end, the project should have:

1. A private (admin) endpoint to create new users. If you want, this could also be an artisan command, as you like. It will mainly be used to generate users for this exercise;
2. A private (admin) endpoint to create new travels;
3. A private (admin) endpoint to create new tours for a travel;
4. A private (editor) endpoint to update a travel;
5. A public (no auth) endpoint to get a list of paginated travels. It must return only `public` travels;
6. A public (no auth) endpoint to get a list of paginated tours by the travel `slug` (e.g. all the tours of the travel `foo-bar`). Users can filter (search) the results by `priceFrom`, `priceTo`, `dateFrom` (from that `startingDate`) and `dateTo` (until that `startingDate`). User can sort the list by `price` asc and desc. They will **always** be sorted, after every additional user-provided filter, by `startingDate` asc.

Models

Users

- ID
- Email
- Password
- Roles (*M2M relationship*)

Roles

- ID
- Name

Travels

- ID
- Is Public (bool)
- Slug
- Name
- Description
- Number of days
- Number of nights (virtual, computed by `numberOfDays - 1`)

Tours

- ID
- Travel ID (*M2O relationship*)
- Name
- Starting date
- Ending date
- Price (integer, see below)

Notes

- Feel free to use the native Laravel authentication.
- We use UUIDs as primary keys instead of incremental IDs, but it's not required for you to use them, although highly appreciated;
- **Tours prices** are integer multiplied by 100: for example, €999 euro will be `99900`, but, when returned to Frontends, they will be formatted (`99900 / 100`);

- **Tours names** inside the `samples` are a kind-of what we use internally, but you can use whatever you want;
- Every `admin` user will also have the `editor` role;
- Every *creation* endpoint, of course, should create one and only one resource. You can't, for example, send an array of resource to create;
- Usage of *php-cs-fixer* and *larastan* are a **plus**;
- Creating docs is **big plus**;
- Feature tests are a **big big plus**.