Junior Fullstack Engineer Technical Assessment

Part 1

Solution Architecture

User Interface (Presentation Layer) - In the client side, a User Interface able to make http requests to the REST API Server. To note that in part 2 this client side app was not implemented.

Routes - Define and expose API endpoints. Then redirects the incoming requests for the correct controller.

Middleware - Allows to have functions between layers. Here we can have middleware handlers for things like parsing and validation of requests, logging and error handling. Because we can process requests before they reach the controllers.

Controllers - Controllers initiate the processing for incoming requests, calling services to perform some logic and interact with data, and at the end prepare the response to the client.

Services (Business Logic Layer) - Responsible for performing business logic, and interacting with data from whichever source through models/data access objects.

Models (Data Access Layer) and Database (Database Layer) - Models contain the entities or DAO (Data Access Objects) which will interact with the data. This solution is under the assumption that both the routes and the highlights/interest points are persisted in a database and that the origin and destination users may choose is limited to the ones belonging to these routes. Nevertheless, the source of the data could be, instead of a database, an external API, in charge of computing routes and getting interest points.

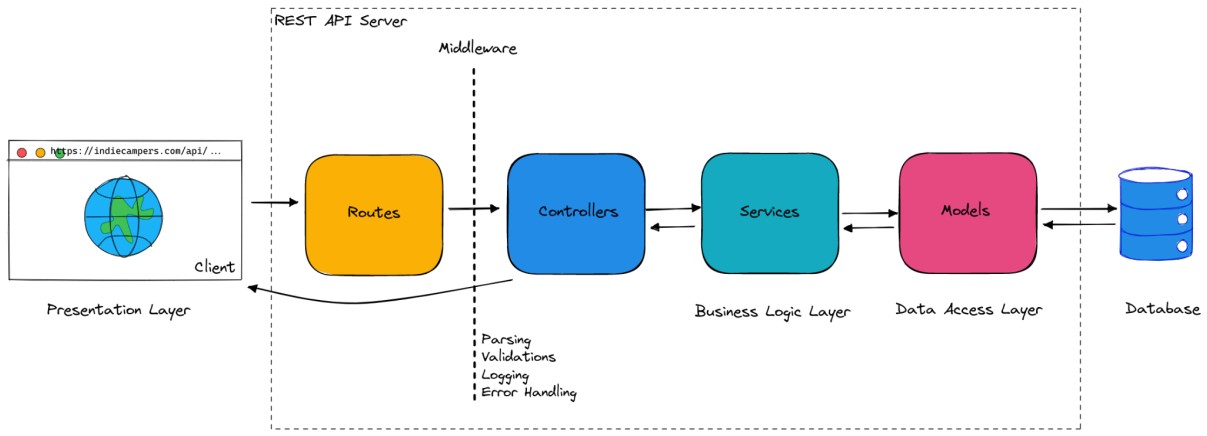

Fig 1 - Architecture Diagram

Data Model

The next figure represents a **Data Model** for my solution of the problem.

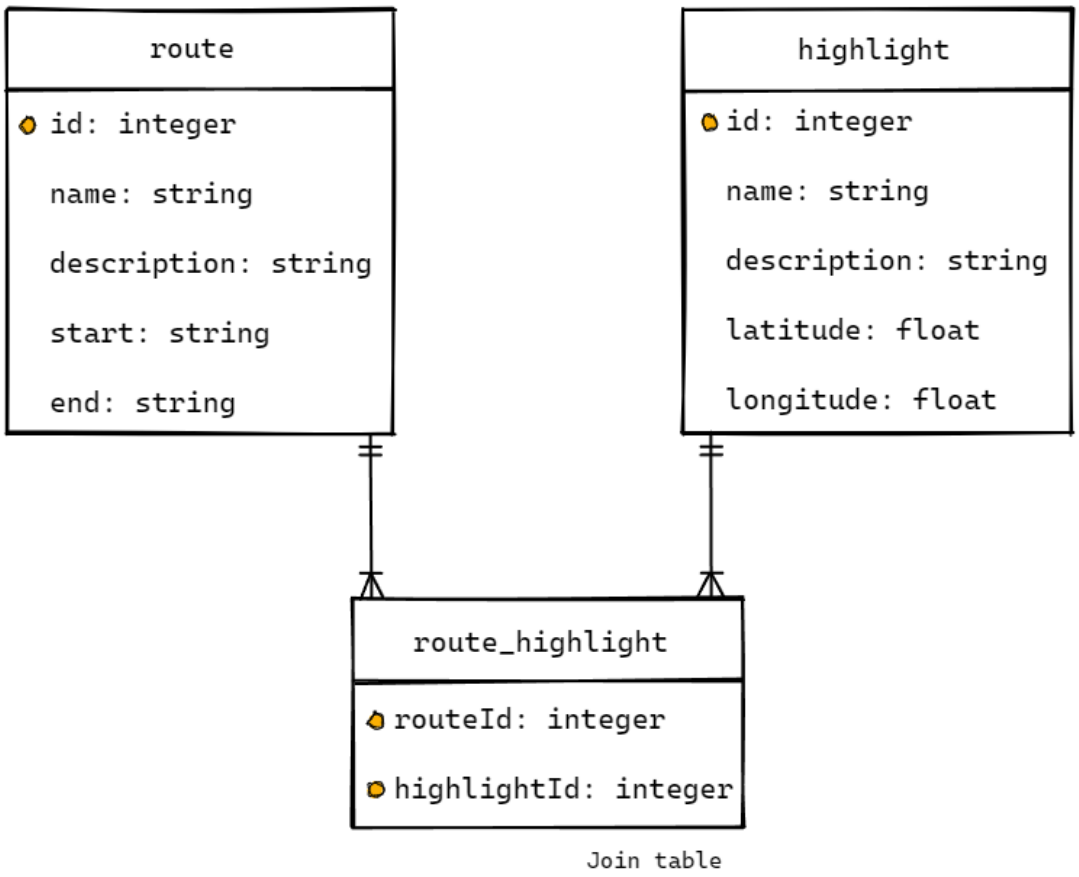

Fig 2 - Data Model

A **route** is characterized by:

- **id** - identifier
- **name** - name for the route
- **description** - detailed description for the route
- **start** - the origin of the route
- **end** - the destination of the route.

An **highlight** is characterized by:

- **id** - identifier
- **name** - the interest point name
- **description** - detailed description of the interest point
- **latitude** - latitude of the coordinates of the interest point
- **longitude** - longitude of the coordinates of the interest point

Additionally, there will be a relationship of **many to many** between routes and highlights, because not only a route may go through multiple interest points, but an interest point may be part of multiple routes. This relationship is represented by **route_highlight**.

Part 2

https://github.com/MarcofSilva/indiecampers-highlights

Marco Silva
marcofgd.silva@gmail.com
August 2022