Mercavus Backend Challenge

Write a backend API for a small app "User Hobbies" with following requirement: (Time: 1 - 2 hrs)

Tech stack: Nodejs with Typescript, Mongodb with mongoose, Test (Jest or Mocha or Jasmine)

Tech details:

- NodeJS with Typescript (Hint: Make your own nice directory structure of controller, service….etc)
- We use Hapi framework (but you can use Express if you like.)
- There are 2 different Mongo collections: User {id, name, hobbies}, Hobbies {id, passionLevel, name, year}
- Hobbies are **not** embedded in the User Schema. (please use Mongo **refs**)
- Endpoints will be CRUD of users and hobbies. (for more info on use cases, please refer to your FE contra-part).
- Validate the requests payload coming from FE.
- Setup Test using the tech stack mentioned above and write at least one unit test to demonstrate.

Once you are done, share the uploaded git repo link via email at pulkit.gupta@mercavus.com. We will get back to you asap.

FE application:

Feature:

- There would be 2 columns both resizable and scrollable.
- In User columns, user will be listed and added.
- Right column is blank grey screen until a user is selected.
- Data should be loaded initially from the mock at http service layer. (Keeping other flow untouched as if a production app)
- Once a user is selected: In Right column, user can now view, add or delete(with confirmation dialog) the hobbies.
- Hobby consist of three things mentioned below. Passion can be selected out of "Low", "Medium", "High" and "Very-High". (Preferably some validation).

Field to add new user / hobby

Hobbies with respect to selection on the left. If no selection is done on the left or the user is new without hobbies, this area should be empty

Selectable list to see the hobbies on the right

Scrollable and flexible to window resize.

Delete button will delete the respective hobby.