

[Project] - RailRoad

Ouvert le : mercredi 9 octobre 2024, 16:00

À rendre : samedi 26 octobre 2024, 23:59

1 - Project overview

RailRoad Ltd is a company which aim to produce the best experience for your local and national commute.

Your task is to produce a full API using Node.js and Express.js (Fastify.js is also possible) to allow users to access information about train station or other relevant data. This should also allow people to "book" a ticket from one stop to another and provide an interface for specific user (let's call them employee) to verify the validity of the ticket

The API needs to be REST compliant (method, endpoint ...).

This project can be done in a group of 2 people.

This project is ONLY about building the API side, you don't have to create a frontend !

2 - Features

You need to develop multiples functionalities

- User management solution
 - Create, Read, Update, Delete user
 - User is at least {id, email, pseudo, password, role}
 - Normal users cannot read information about another user - but an employee can check information
 - You can create a new user even without being logged
 - You can only update yourself (other users cannot update you EXCEPT if admin)
 - You can only delete yourself (other users cannot delete you)
- Authentication need to be setup
 - Different solutions can be used, jwt token is recommended (see tips for more information)
 - All Read endpoints of train data are non-logged/anonymous
 - All Write endpoints need the request to be authenticated (stateless)
- A train endpoint
 - List all trains and allow you to sort by date, starting station, end station with a limit (default limit is 10 but can be changed with a parameter)
 - Create, Read, Update, Delete trains
 - Train is at least {id, name, start_station, end_station, time_of_departure}
 - Only an admin can create, update and delete a train
- A trainstation endpoint
 - List all trainstations and allow you to sort mission by name
 - Create, Read, Update, Delete trainstation
 - Only an admin can create, update or delete a train station (careful when you delete one train station, you need to think about trains!)
 - Trainstation is at least {id, name, open_hour, close_hour, image}. The image need to be resized to a 200*200px image if the upload is too big
- An endpoint to book a ticket and validate it
- It is important to provide good feedback to the users using the API so you will need to implement a simple type solution/validation like Joi or Yup or AJV
- On the same note, you need to use the valid HTTP code when returning information to the user
- Testing is important. You are required to implement a few tests on your project. Focus on testing the core functionalities

3 - Tips

Authentication:

- You do not need a full OAuth solution, just a way to generate a token
- You can use PassportJS and the strategy Passport-local-mongoose
- To test a jwt token you can use jwt.io

Validation/type solution

- Joi, Yup and AJV are working almost in a same way
- Only test the users input
- It can be interesting to build a middleware for that !

Documentation

- The documentation can be based on swagger/openapi standard
- For an easy writing you can use editor.swagger.io
- Try to define the endpoint with both the input value (url, body...) and the possibles returned values (success, error...)

4 - Deliverables

You need to submit an archive containing the source code of your project and the design of the API done with swagger/openAPI (or other standard solution - in that case precise which one).

With a readme for the installation steps and to be able to run your project: **if the project is not runnable a 0 grading can be attributed!**

5 - Graded items

Global architecture of the project: 4pts

Authentication: 3pts

Train feature:

- Operation by admin: 4pts
- Find a station or a specific train: 4pts
- Control the validity of a ticket: 3pts

Documentation and Tests: 2pts

Not providing a documentation or basics tests can result on negative points (up to -2)

Up to 50% of points for each categories can be lost to code quality and/or security. You have to work as you would do it in a company with impact for customers or finance !

Ajouter un travail

Statut de remise

Statut des travaux remis	Aucun devoir n'a encore été remis
Statut de l'évaluation	Non évalué
Temps restant	1 heure 7 min restants
Dernière modification	-

Commentaires	▶ Commentaires (0).
--------------	-------------------------------------

Contactez-nous



Suivez-nous



 Contacter l'assistance du site

Connecté sous le nom « Orlann Ferreira dias almeida » (Déconnexion)