Web Technologies - Module D Project

Time: 02:30

Back-end

IFSC2024_TP17_EN

Iran's first International
Friendly Skills Competition

worldskills
I.R.IRAN

Introduction

Martin Deliver Company, after 10 years of experience in sending sensitive packages in Tehran with different couriers, is willing to offer its order process online to different companies. Martin Deliver does not have direct contact with the end user and offers its service directly to companies with high shipping orders. (B2B Service)

Description of project and tasks

Considering that Martin Deliver is a B2B service, you are asked to provide the system by implementing a **web service** in the form of RestFul API.

Companies in this system must be able to register or cancel their order after authentication.

Couriers can see the information of registered orders through the application they have and after accepting the order by the courier **(only one courier)**, they can refer to the place of receipt and transfer the order to the destination.

Martin Deliver should inform the companies that are working with them about the change of status in the form of **Webhook** method to the company's web service.

Companies side

Important information in providing courier orders from any company

- Start position (Latitude, Longitude)
- Delivery address
- Delivery provider name
- Delivery provider mobile
- Destination position (Latitude, Longitude)
- Destination address
- Recipient's name
- Recipient's mobile

Status

Companies can cancel their order if the package has not yet been received from the start position

Companies can cancel their order if the package has not yet been received from the start location by using their order ID and receive an appropriate message in response to their canceled order.

Companies can send their requests with the API Key they have.

Courier side

Couriers send requests after authentication (using username and password) and receiving a token with the authenticated token.

The courier should be able to see the information of the orders received and waiting for the courier, and at the decision of the courier, the order can be accepted and sent to the place of delivery.

To accept the receipt of the package, the courier must announce the status, and live location and move towards the destination.

Also, after delivering the package to the destination, the courier must announce the status of that order.

Each courier can have several orders at the moment.

company set their own orders

Martin Deliver side

Martin Deliver must have a web panel that can define the company and its webhook address after login and then receive the API KEY.

In the Martin Deliver panel, the orders registered by each company should be displayed and its status and which courier is responsible for the package should be displayed.

It also requires a page to define couriers, which requires the definition of username and courier name and hashed password in the database.

Also, a page to display the list of couriers should be implemented and the current orders of couriers should be displayed.

Also, on another page, past and current orders should be displayed for each courier

Instructions to the Competitor

Courier's access token should not be able to register an order in the form of a company, and the company should not be able to view courier orders directly with its token.

In endpoints related to courier and company, we need RestFul API and api documentation is required. (Postman Documentation)

For the Martin Deliver side, we need a login page, lists and requested forms. It is important to prevent RaceCondition at the moment of accepting orders by couriers.

Implementing a nice look for the Martin Deliverer side panel page is not very important to us.

In order to use the system, we need api documents for both the company side and the format for presenting the status to the company (the documents can be in postman format or even PDF files, but it is important to understand the documents)

Migrations must be used to create a database.

It is important to use different types of methods on endpoints.

The right error with the right message is important in the endpoints and use the correct http status code.

All request payloads must be provided with snake_case keys and the response should be JSON with snake_case keys.

The implemented project must be on Git and have different commits with a specific message. There is no need to push it on the server.

It is important to use the features of the framework.

Endpoints

Companies side

- Login
- Cancel a package
- Inquiry about the status of the registered order

Courier side endpoints

- Login and receive the correct token
- The list of packages that have not yet been accepted by any courier
- Accept a package
- Change Package Status
- Send the live location of the courier when he/she picks up the package.

There is no problem in providing more endpoints.For Martin Delivery admin's default access, please put the following information in your database

username	password
martin	martinpass

Finally, Push your final project in the Git.

Instructions

- You are only allowed to use Laravel.
- You need to implement migration and seed.
- You must follow Restful API standards.
- The project structure and clean code matters.
- Push your project to git on your repository - Module D

Note that the git folder should exist on the uploaded files on the server.

*"A **webhook** in web development is a method of augmenting or altering the behavior of a web page or web application with custom callbacks. These callbacks may be maintained, modified, and managed by third-party users and developers who may not necessarily be affiliated with the originating website or application. The term "webhook" was coined by Jeff Lindsay in 2007 from the computer programming term hook." -Wikipedia*