**Back-end Test** (**LARAVEL**).

* Para dar respuesta a el caso práctico Prueba **Técnica - Programador (Back-end) LARAVEL)** utilizando el framework de laravel, después de hacer un análisis según los requerimientos y las descripciones de los casos de usos, se realizó el diseño y construcción de la base de datos que permita la automatización del proceso en cuestión.
* El sistema de gestor de base de datos seleccionado fue mysql, el cual se integra muy bien con el framework de php.
* La base de datos tiene dos tablas:
* **Vehiculos (**chapa, tipo, marca**)**
* **Registro\_vehs (**chapa, horaEnt, horaSal, tiempoEst, montoPagar**)**
* **Funcionamiento**
* Primero se debe se debe registrar los vehículos puesto que entre estas dos tablas hay una relación de 1:m y la tabla registro\_vehs tiene como llave foránea chapa, primero se registran los vehículos y después es que se le da entrada en el registro de vehículos.
* También se hizo un endpoint que permite ver todos los registros de la tabla vehículos, además de la chapa, se le agrego el tipo y la marca como parte de la flexibilidad que se pretende que tenga la aplicación en el futuro.
* **EndPoints**

1. <http://Nombre-host/api/todos>(GET) -> Devuelve todos los registros de la tabla vehículos.
2. http://Nombre-host /api/guadar/vehiculo(POST) -> Inserta un vehículo en la tabla vehículos.
3. http://Nombre-host /api/entrada/vehiculo(POST) -> Registra la entrada de un vehículo al estacionamiento, que previamente se haya insertado en el sistema y se recoge la hora en la que el vehículo entro a dicho estacionamiento.
4. http://Nombre-host /api/salida/vehiculo(PUT) -> Registra la salida de un vehículo del estacionamiento, y se recoge la hora en la que el vehículo salio a dicho estacionamiento, además de que se calcula el tiempo de estancia y la cantidad que debe pagar por dicho tiempo.
5. http://Nombre-host /api/informe/clientes(GET) -> Muestra el informe correspondiente a los clientes.

* **Repositorio**
* <https://github.com/menelik8402/backend-test.git>