



# Requisitos para el back-end de la aplicación

## Tecnologías

El lenguaje de la aplicación es JavaScript.

Se ejecuta en el entorno Node.js v12.17.0.

Acceso a la aplicación a través del protocolo HTTP 1.1.

## Requisitos generales

La aplicación utiliza la base de datos PostgreSQL. La aplicación interactúa con la base de datos a través del paquete npm `sequelize` sobre el paquete `pg`. `sequelize` es un ORM para trabajar con diferentes bases de datos en node.js.

Las solicitudes se registran a través del módulo `winston`. La documentación de la aplicación se organiza mediante el módulo `apidoc`.

La aplicación debe cumplir con los requisitos REST.

La aplicación debe tener un controlador de errores global. Si ocurren excepciones, deben manejarse y la aplicación debe continuar ejecutándose.

Los errores de la aplicación (solicitudes fallidas, excepciones, respuestas que no sean 2XX) se deben registrar en un archivo `error.log` separado.

## Requisitos de la URL

### URL auxiliares

- Debe haber una URL que te ayude a comprobar que el back-end se esté ejecutando y aceptando solicitudes. El estado `200 OK` debería regresar si la respuesta es exitosa.
- Debe existir una URL a través de la cual funcione la búsqueda de estaciones de metro. Si la búsqueda tiene éxito, se debe devolver el número, el color y el

nombre de la estación. Si hay más de una estación, se debe devolver el número, el color y el nombre de cada estación. Si no se encuentra ninguna estación, se debe devolver una lista vacía.

## URL para repartidores o repartidoras

- **La URL debe estar presente: al acceder, el repartidor o repartidora puede registrarse en la aplicación. La URL debe aceptar el nombre de usuario, la contraseña y el nombre del repartidor o repartidora. El inicio de sesión, el hash de la contraseña y el nombre del repartidor o repartidora deben escribirse en los campos `login`, `passwordHash` y `firstName` de la tabla *Couriers* (Servicio de entrega). El campo `passwordHash` almacena el hash de la contraseña. Se genera mediante funciones estándar, por lo que puedes comprobar la coincidencia hash-contraseña mediante la autorización.**
- **El campo de inicio de sesión debe ser único. Si el registro es exitoso, la entrada correspondiente debe aparecer en la base de datos; si no tiene éxito, se debe devolver un error. Aprende más sobre errores en la documentación </docs/en/#api-Orders-CreateOrder>**
- Debe haber una URL para iniciar sesión en la cuenta de repartidor o repartidora. El nombre de usuario y la contraseña de repartidor o repartidora deben enviarse al iniciar sesión. Se debe devolver la `ID` del repartidor o repartidora cuando se haya iniciado sesión correctamente. Si el inicio de sesión falla, se debe devolver un error.
- **La URL para eliminar la cuenta del repartidor o repartidora debe estar presente. Al acceder se debe obtener la `ID` del repartidor o repartidora en la tabla *Couriers*. Al eliminar, los pedidos vinculados en la tabla *Orders* (Pedidos) deben borrarse.**

## URL para los pedidos

Siempre que alguna de las URL devuelva datos completos del pedido, la respuesta también debe contener el estado de cada pedido. El estado debe tener estos valores:

- `0`: se creó el pedido, no pasó nada más;
- `1`: el repartidor o repartidora ha aceptado el pedido;
- `2`: el pedido ha sido completado;

- `1`: el pedido ha sido cancelado.

El estado se debe calcular en relación con los valores de los campos en la base de datos en la tabla Orders (ver "Descripción del contenido de la base de datos").

Los campos se enumeran en orden de prioridad:

- `finished = true` → `status = 2`
- `cancelled = true` → `status = -1`
- `inDelivery = true` → `status = 1`
- `Other cases` → `status = 0`

Debe haber una URL para crear un pedido; al crear un pedido, se especifican los siguientes parámetros:

- nombre
- apellido
- dirección
- estación de metro más cercana
- teléfono
- número de días de alquiler
- fecha de entrega
- comentario
- lista de colores que coinciden

Se debe asignar un número de seguimiento individual al pedido cuando se crea.

Los parámetros enviados se escriben en la tabla `Orders` (Pedidos) de la siguiente manera:

- nombre: `firstName`
- apellido: `lastName`
- dirección: `address`
- estación de metro más cercana: `subwayStation`
- teléfono: `phone`

- periodo de alquiler: `rentTime`
- fecha de entrega: `deliveryDate`
- comentario: `comment`
- lista de colores que coinciden: `color`
- número de seguimiento: `track`

Si el pedido se creó con éxito, se debe devolver su número de seguimiento; de lo contrario, se debe devolver un error. Lee más sobre los errores en la documentación: </docs/en/#api-Orders-CreateOrder>

- **Debe haber una URL para recuperar los datos del pedido desde su número de seguimiento. Se debe obtener un número en el acceso. Si se encuentra un pedido que coincida, se deben devolver sus datos; de lo contrario, se debe devolver un error.**
- Debe haber una URL para que el repartidor o repartidora acepte el pedido. La URL toma el número de seguimiento del pedido y la ID del repartidor o repartidora. Si hay un problema al aceptar el pedido, se debe devolver un error.
- Debe haber una URL para cancelar el pedido. La URL acepta el número de seguimiento del pedido. En caso de una cancelación sin éxito, se debe devolver un error.
- Debe haber una URL para completar el pedido. El número de pedido se envía al iniciar sesión. En caso de que no se complete correctamente, se debe devolver un error.
- Debe haber una URL para recuperar todos los pedidos que coincidan con los parámetros especificados. Los parámetros de búsqueda son la estación de metro más cercana y la ID del repartidor o repartidora. También se deben enviar los límites en el número de registros de salida por página y el número de página. Lee más sobre errores y casos de uso en la documentación: </docs/#api-Orders-GetOrdersPageByPage>
- La URL para el número de pedidos de repartidor o repartidora completados debe estar presente. Se debe obtener la ID del repartidor o repartidora al

iniciar sesión. Lee más sobre errores y casos de uso en la documentación:

</docs/en/#api-Couriers-GetOrdersCountByCourierId>

## Restricciones de los campos

Elemento	Requisitos	Obligatorio
<b>iniciar sesión</b>	Solo letras latinas. La longitud del texto es de 2 a 10 caracteres.	<input checked="" type="checkbox"/>
<b>firstName</b>	Solo letras latinas. La longitud del texto es de 2 a 10 caracteres.	<input type="checkbox"/>
<b>contraseña</b>	Solo números enteros. La longitud es exactamente de 4 caracteres.	<input checked="" type="checkbox"/>

## Descripción del contenido de la base de datos

La base de datos consiste en dos tablas: Couriers y Orders. La primera tabla contiene datos sobre repartidores y repartidoras, la segunda almacena datos sobre pedidos.

### Couriers

Nombre	Tipo	Asignación
<b>iniciar sesión</b>	string	inicio de sesión del repartidor o repartidora
<b>passwordHash</b>	string	contraseña hash del repartidor o repartidora
<b>firstName</b>	string	contraseña hash del repartidor o repartidora

### Orders

Nombre	Tipo	Asignación
<b>courierId</b>	número	el campo que vincula esta tabla con la tabla Couriers
<b>firstName</b>	string	nombre del cliente
<b>lastName</b>	string	apellido del cliente

Nombre	Tipo	Asignación
<b>address</b>	string	dirección
<b>metroStation</b>	string	estación de metro más cercana
<b>phone</b>	string	número de teléfono
<b>rentTime</b>	número	periodo de alquiler
<b>deliveryDate</b>	datos	fecha de entrega
<b>track</b>	número	número de seguimiento
<b>color</b>	un array de strings	colores de scooter preferidos
<b>comment</b>	string	un comentario para el repartidor o repartidora
<b>cancelled</b>	lógico	el pedido ha sido cancelado
<b>finished</b>	lógico	el pedido ha sido enviado
<b>inDelivery</b>	lógico	el pedido se está entregando