Требования к бэкенду приложения

# Технологии

Язык приложения — JavaScript.

Выполняется в среде Node.js v12.17.0. Доступ к приложению по протоколу HTTP 1.1.

# Общие требования

Приложение использует базу данных. БД — PostgreSQL. Приложение

взаимодействует с БД через npm-пакет поверх пакета pg .

sequelize

— ORM для работы с различными БД в node.js.

sequelize

Запросы логируются через модуль winston . Документация к приложению осуществляется с помощью модуля apidoc .

Приложение должно отвечать требованиям REST.

В приложении должен быть глобальный обработчик ошибок. При возникновении исключений они должны быть обработаны, а приложение должно продолжить работу.

Ошибки приложения (неуспешно обработанные

запросы, исключения; ответы, отличные от 2XX) должны логироваться в отдельный файл

error.log

# Требования к URL

## Вспомогательные URL

Должен присутствовать URL, через который можно проверить, что бэкенд запущен и принимает запросы. При успешном ответе должен вернуться статус 200 OK .

Должен присутствовать URL, через который работает поиск станций метро. В случае успешного поиска должны вернуться номер станции, её цвет и название. Если станций несколько, для каждой должны возвращаться номер, цвет и название. Если станция не найдена, должен вернуться пустой список.

## URL для курьеров

**Должен присутствовать URL: при обращении к нему курьер может зарегистрироваться в приложении. URL должен принимать**

## логин, пароль и имя курьера. Логин, хэш пароля и имя курьера должны записываться в поля *login ,*

**и таблицы *Couriers*. В поле хранится**

***passwordHash***

***firstName***

**passwordHash**

## хэш пароля, генерируется стандартными функциями, поэтому соответствие хэш-пароль проверить можно через авторизацию.

**Поле login должно быть уникальным. При успешной регистрации соответствующая запись должна появиться в базе. При неуспешной должна вернуться ошибка. Подробнее об ошибках в**

## документации

**/docs/#api-Courier-CreateCourier**

Должен присутствовать URL для входа в учётную запись курьером. На вход должны отправляться логин и пароль курьера. При успешном входе должен

вернуться курьера. Если войти не удалось, должна вернуться ошибка.



id

## Должен присутствовать URL для удаления учётной записи курьера. На

**вход должен подаваться курьера в таблице Couriers. При удалении**



**id**

## связанные заказы в таблице Orders должны быть стёрты.

**URL для заказов**

Каждый раз, когда какой-нибудь из URL возвращает полные данные о заказе, ответ должен содержать и статус каждого заказа. В статусе должны быть такие значения:

* заказ создан, больше ничего с ним не происходило;



0

* заказ принят курьером;



1

* заказ завершён;



2

— заказ отменён.



-1

Статус должен вычисляться относительно значений полей в БД в таблице Orders (см. пункт «Описание содержимого базы данных»). Поля указаны в порядке приоритетности:

->

finished = true

status = 2

->

cancelled = true

status = -1

->

inDelivery = true

status = 1

->

Остальные случаи

status = 0

Должен присутствовать URL для создания заказа. При создании заказа указываются следующие параметры:

имя;

фамилия; адрес;

ближайшая станция метро; телефон;

количество дней аренды;

дата доставки; комментарий;

список подходящих цветов.

При создании заказа ему должен быть присвоен индивидуальный номер для отслеживания.

Orders

Переданные параметры записываются в таблицу имя:

firstName

фамилия:

lastName

адрес:

address

ближайшая станция метро: телефон:

metroStation

phone

количество дней аренды:

rentTime

дата доставки: комментарий:

deliveryDate

comment

список подходящих цветов: номер отслеживания:

color

track

следующим образом:

Если заказ создан успешно, должен вернуться его номер отслеживания. В противном случае должна вернуться ошибка. Подробнее об ошибках в документации:

/docs/#api-Orders-CreateOrder

## Должен присутствовать URL для получения данных о заказе по его номеру отслеживания. На вход должен подаваться номер. Если соответствующий заказ найден, должны вернуться данные о

**нём. Иначе должна вернуться ошибка.**

Должен присутствовать URL для принятия заказа курьером. URL принимает номер отслеживания заказа и id курьера. Если при принятии заказа возникли проблемы, должна вернуться ошибка.

Должен присутствовать URL для отмены заказа. URL принимает номер для отслеживания заказа. В случае неуспешной отмены должна вернуться ошибка.

Должен присутствовать URL для завершения заказа. На вход подаётся номер заказа. В случае неуспешного завершения должна вернуться ошибка.

Должен присутствовать URL для получения всех заказов, которые соответствуют заданным параметрам. Параметры поиска — ближайшая станция метро и id курьера. Также должны быть переданы ограничения по количеству выводимых записей на странице и номер страницы. Подробнее об ошибках и кейсах применения в документации:

GetOrdersPageByPage

/docs/#api-Orders-

Должен присутствовать URL для получения количества выполненных заказов курьера. На вход должен подаваться id курьера. Подробнее об ошибках и кейсах использования в документации:

GetOrdersCountByCourierId

/docs/#api-Couriers-

**Ограничение полей**

|  |  |  |
| --- | --- | --- |
| Элемент | Требования | Обязательность |
| [login](https://www.notion.so/login-c8208aa7f71c4cb68580d2826a357da1) | Только английские буквы. Длина текста не менее 2 и не более более 10 символов. |  |
| [firstName](https://www.notion.so/firstName-454e17c72db841b89d3fe780263a33ce) | Только английские и русские буквы. Длина текста не менее 2 и не более более 10 символов. |  |
| [password](https://www.notion.so/password-325abf8e6dd54fc2a30a686359bf6603) | Только целые числа. Длина 4 символа — не больше и не меньше. |  |

# Описание содержимого базы данных

БД состоит из двух таблиц: Couriers и Orders. Первая таблица содержит данные о курьерах, вторая — данные о заказах.

**Couriers**

|  |  |  |
| --- | --- | --- |
| Название | Тип | Назначение |
| [login](https://www.notion.so/login-057525b6eee542fba8b9289be85db418) | строка | логин курьера |
| [passwordHash](https://www.notion.so/passwordHash-d43223e4f5c44d2da7516d788e8bde71) | строка | захэшированный пароль курьера |
| [firstName](https://www.notion.so/firstName-682185ff27df4598a4cad2c4ed41e959) | строка | имя курьера |

**Orders**

|  |  |  |
| --- | --- | --- |
| Название | Тип | Назначение |
| [courierId](https://www.notion.so/courierId-f44ebbd1ca7947a7b3b3fb2138c4acf3) | число | поле, связывающее эту таблицу с таблицей Couriers |
| [firstName](https://www.notion.so/firstName-29ba7bc14b5343fe9b3a01b8216fa25f) | строка | имя заказчика |
| [lastName](https://www.notion.so/lastName-110e653b3dd54cfba174904c8bd2c3ff) | строка | фамилия заказчика |
| [address](https://www.notion.so/address-07d52d9785d143e8975fe74d3ac6194e) | строка | адрес |
| [metroStation](https://www.notion.so/metroStation-d41358a2100f47ef8b8e5de1091b86ca) | строка | ближайшая станция метро |
| [phone](https://www.notion.so/phone-2467b82d712840dc972623da43b0d6a0) | строка | номер телефона |
| [rentTime](https://www.notion.so/rentTime-af73f68298644b8d9b97bb072ef77a4d) | число | срок аренды |
| [deliveryDate](https://www.notion.so/deliveryDate-5ada3e42ab8148419a45213e906e7502) | дата | дата доставки |
| [track](https://www.notion.so/track-e571290cf4fa4b77a2c45f9f8945e32e) | число | номер отслеживания |
| [color](https://www.notion.so/color-96bd542e84124091b3226afdd1e431c7) | массив строк | предпочитаемые цвета самоката |
| [comment](https://www.notion.so/comment-51e428f3a49f4c228cec1bfbd11f4f7b) | строка | комментарий курьеру |
| [cancelled](https://www.notion.so/cancelled-b658c330f9b94009a24879fd80a2f465) | логический | заказ отменён |
| [finished](https://www.notion.so/finished-f8df7f82f8cb4b36b95a5f84beb7e42f) | логический | заказ доставлен |
| [inDelivery](https://www.notion.so/inDelivery-58deec2552a946df943bbd03e47eee7d) | логический | заказ в процессе доставки |