**ОБЛАСТНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ**

**ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

**«ТОМСКИЙ ЭКОНОМИКО-ПРОМЫШЛЕННЫЙ КОЛЛЕДЖ»**

**ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА REST API СЕРВЕРА САМОКАТ В ФРЕЙМВОРКЕ LARAVEL**

***Специальность 09.02.07 информационные системы и программирование***

Выполнили:

Студенты группы 1912с

\_\_\_\_\_\_\_\_\_\_\_\_ / Ладяев Н.А.

\_\_\_\_\_\_\_\_\_\_\_\_ / Рюгин А.И.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2025 г.

Проверил:

Преподаватель

\_\_\_\_\_\_\_\_\_\_\_\_ / Уляхин В.А.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2025 г.

Томск 2025

Оглавление

[Введение 3](#_Toc183277088)

[Технологии и программное обеспечение 4](#_Toc183277089)

[Используемые языка программирования и фреймворки 4](#_Toc183277090)

[Используемое программное обеспечение 6](#_Toc183277091)

[Глава 1. Техническое задание на разработку информационного продукта 7](#_Toc183277092)

[Описание проекта и задач 7](#_Toc183277093)

[Специфические требования к API 9](#_Toc183277094)

[Функционал неавторизированного пользователя 9](#_Toc183277095)

[Функционал авторизированного пользователя 13](#_Toc183277096)

[Функционал администратора 17](#_Toc183277097)

[Функционал курьера 23](#_Toc183277098)

# Введение

Актуальность проекта: в условиях стремительного развития электронной коммерции и растущего спроса на быструю и удобную доставку, "Самокат Про" предлагает инновационное решение для курьерских служб. Наше приложение предоставляет удобный и эффективный способ управления курьерскими сменами, позволяя курьерам выставлять свою доступность, а администраторам подтверждать смены.

Цель проекта: разработка API и мобильного приложения "Самокат Про", обеспечивающих удобство, эффективность и надежность управления курьерскими сменами.

Технические требования: API должно быть разработано с использованием современных технологий и соответствовать стандартам безопасности. Мобильное приложение должно быть разработано для платформ Android.

Задачи проекта:

1. Разработка пользовательского интерфейса мобильного приложения, обеспечивающего простоту и интуитивность использования как для курьеров, так и для администраторов.
2. Создание API, обеспечивающего взаимодействие мобильного приложения с серверной частью системы.
3. Реализация функционала выставления доступности курьерами для выхода на смену.
4. Разработка системы подтверждения смен администраторами.
5. Интеграция системы мониторинга и управления курьерскими сменами.

Краткое описание функционала:

1. Авторизация курьеров и администраторов.
2. Выставление доступности курьерами для выхода на смену.
3. Подтверждение смен администраторами.
4. Управление курьерскими сменами.

# Технологии и программное обеспечение

## Используемые языка программирования и фреймворки

PHP - это широко используемый язык программирования общего назначения, который применяется в основном для разработки веб-приложений и веб-серверных скриптов. PHP обычно встраивается непосредственно в HTML код, и его скрипты выполняются на сервере, что позволяет генерировать динамический контент на веб-страницах.

Основные характеристики PHP включают в себя:

* Простота использования: PHP имеет синтаксис, который легко читать и понимать, что делает его доступным для новичков.
* Гибкость: PHP поддерживает различные парадигмы программирования, включая процедурное и объектно-ориентированное программирование, что позволяет разработчикам выбирать подход, который лучше всего подходит для их задач.
* Широкая поддержка: PHP является одним из наиболее популярных языков программирования для веб-разработки и имеет обширное сообщество разработчиков, библиотек и фреймворков.
* Интеграция с веб-серверами: PHP легко интегрируется с различными веб-серверами, такими как Apache, Nginx и другие.
* Динамическое создание контента: PHP позволяет создавать динамические веб-страницы, которые могут меняться в зависимости от пользовательского ввода, данных из базы данных и других факторов.
* Базы данных: PHP обладает встроенной поддержкой для работы с различными системами управления базами данных, такими как MySQL, PostgreSQL, SQLite и другими.
* PHP используется для создания широкого спектра веб-приложений, включая блоги, интернет-магазины, корпоративные веб-сайты, социальные сети и многое другое. Он остается одним из наиболее популярных и распространенных языков программирования для веб-разработки благодаря своей простоте, гибкости и широкой поддержке сообщества.

SQL - это специализированный язык программирования, который используется для работы с реляционными базами данных. Он предоставляет набор команд и инструкций для управления данными в базах данных, таких как добавление, удаление, изменение и извлечение данных.

Основные характеристики SQL:

* Декларативный язык: SQL является декларативным языком программирования, что означает, что вы определяете, что вы хотите сделать с данными, а не как это делать. Вы описываете желаемый результат, а не последовательность действий.
* Манипуляция данными: SQL предоставляет команды для манипуляции данными в базе данных. Например, с помощью SQL вы можете создавать новые таблицы, добавлять новые записи, обновлять существующие записи и удалять записи.
* Язык запросов: для извлечения данных из базы данных. Вы можете создавать запросы, чтобы выбирать данные из одной или нескольких таблиц, фильтровать данные, сортировать их и группировать.

Laravel - это бесплатный и с открытым исходным кодом фреймворк для создания веб-приложений на языке программирования PHP. Он предоставляет разработчикам инструменты и структуру для быстрой и эффективной разработки веб-приложений, используя современные практики разработки и популярные шаблоны проектирования.

Основные характеристики Laravel:

* Элегантный синтаксис: Laravel предлагает простой и выразительный синтаксис, который делает код легким для чтения и понимания.
* Мощные функции: Laravel обладает богатым набором функций, включая маршрутизацию, контроллеры, валидацию данных, работу с базой данных, аутентификацию пользователей, кэширование, отправку электронной почты и многое другое.
* Быстрый старт: Laravel предлагает множество готовых компонентов и инструментов, которые позволяют быстро начать работу над проектом, не тратя время на настройку инфраструктуры.

C# - это объектно-ориентированный язык программирования, разработанный Microsoft. Предназначен для создания широкого спектра приложений, включая веб-приложения, мобильные приложения, десктопные приложения, игры и многое другое.

Основные характеристики C#:

* Объектно-ориентированный: C# поддерживает такие принципы ООП, как инкапсуляция, наследование и полиморфизм.
* Типизация: C# является строго типизированным языком, что означает, что типы данных должны быть явно объявлены и проверены во время компиляции.
* Многопоточность: C# поддерживает многопоточное программирование, что позволяет создавать приложения, которые могут выполнять несколько задач одновременно.
* Сборка мусора: C# автоматически управляет памятью, освобождая программиста от необходимости вручную управлять памятью.
* Интеграция с .NET: C# тесно интегрирован с платформой .NET, что позволяет использовать библиотеки классов .NET для создания приложений.
* LINQ: C# поддерживает Language Integrated Query (LINQ), что позволяет писать запросы к данным прямо в коде.
* Асинхронное программирование: C# поддерживает асинхронное программирование с использованием ключевых слов async и await, что упрощает написание отзывчивых приложений.
* Расширяемость: C# позволяет создавать расширения и библиотеки, которые могут быть использованы в других проектах.

## Используемое программное обеспечение

Draw.io — это бесплатный онлайн инструмент для создания диаграмм и схем. Он предоставляет широкий набор инструментов и элементов, которые позволяют пользователям создавать различные типы диаграмм, такие как организационные диаграммы, блок-схемы, ER-диаграммы, сетевые диаграммы, диаграммы потока процессов и многое другое.

Open Server Panel — это бесплатное программное обеспечение для Windows, которое предоставляет локальную среду разработки для веб-приложений. Он включает в себя веб-сервер Apache, базу данных MySQL, сервер приложений PHP и другие инструменты, необходимые для разработки и тестирования веб-приложений.

PhpStorm — это интегрированная среда разработки (IDE) для языка программирования PHP, разработанная компанией JetBrains. PhpStorm предоставляет разработчикам мощные инструменты для написания кода, отладки, тестирования и поддержки веб-приложений на PHP.

Postman — это популярный инструмент для разработки, тестирования и документирования API. Он предоставляет разработчикам удобный способ отправлять HTTP-запросы к веб-сервисам и анализировать ответы.

Visual Studio — это интегрированная среда разработки (IDE) от Microsoft, предназначенная для создания, отладки и развертывания приложений на различных платформах и языках программирования.

# Глава 1. Техническое задание на разработку информационного продукта

## Описание проекта и задач

**ОБЩЕЕ ОПИСАНИЕ ПРОЕКТА**

Все пользователи системы подразделяются на две группы:

1. Администраторы.
2. Курьеры.

Курьеры получают доступ к функциям только после успешной авторизации.

Функционал не авторизированного пользователя:

1. Авторизация.

Функционал администратора:

1. Регистрация курьеров.
2. Просмотр и подтверждение смен.
3. Считает доход курьера.
4. Добавление, редактирование, удаление бонусов.
5. Просмотр, редактирование, удаление профилей у курьеров.
6. Просмотр доступностей курьеров.
7. Просмотр, добавление, редактирование, удаление штрафов у курьеров.

Функционал курьера:

1. Просмотр, добавление, редактирование, удаление доступностей.
2. Просмотр штрафов.
3. Просмотр, добавление, редактирование, удаление оценки смены.

Функционал авторизированного пользователя:

1. Выход.
2. Просмотр бонусов.
3. Просмотр и редактирование своего профиля.
4. Просмотр дня.

В примерах будет использоваться переменная {{host}} которая обозначает адрес

информационного продукта.

**ОБЩИЕ ТРЕБОВАНИЯ**

Сервер не может или не будет обрабатывать запрос из-за чего-то, что очевидно является ошибкой клиента:

|  |
| --- |
| Неверный формат запроса |
| "code": 400,  "message": "Bad Request" |

Для доступа к ресурсу требуется аутентификация, и клиент не предоставил действительные учетные данные:

|  |
| --- |
| Не авторизован |
| "code": 401,  "message": "Unauthorized" |

Сервер понял запрос, но отказывается его авторизовать. Это может быть связано с отсутствием необходимых прав доступа:

|  |
| --- |
| Доступ запрещен |
| "code": 403,  "message": "Forbidden" |

Сервер не может найти запрошенный ресурс. Это может быть связано с неправильным URL или удалением ресурса:

|  |
| --- |
| Ресурс не найден |
| "code": 404,  "message": "Not Found" |

В случае ошибок, связанных с валидацией данных во всех запросах необходимо возвращать следующее:

|  |
| --- |
| тело ответа |
| "code": <code>,  "message": <message> |

Вместо <code> и <message> необходимо указывать соответствующее значение, определенное в описании ответа на соответствующий запрос.

## Специфические требования к API

## Функционал неавторизированного пользователя

Аутентификация

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/auth/login  **Method:** POST  **Headers:**  "Content-Type: application/json"  "Accept: application/json"  **Body:**  {  "login": "ladyaev-na",  "password": "qweQWE123"  } | **Response code:** 200  **Response body:**  **{**  "name": "Никита",  "surname": "Ладяев",  "patronymic": "Александрович",  "login": "ladyaev-na",  "password": "qweQwe123",  "token": "token"  **}** |

Входные параметры:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Параметр** | **Тип данных** | **Обязательность** | **Описание** | **Варианты значений** |
| login | varchar(64) | Да | Логин | ladyaev-na |
| password | varchar(64) | да | Пароль | qweQWE123 |

Проверки:

|  |  |
| --- | --- |
| **Параметр** | **Условия** |
| login | Максимальная длина 64 символа.  Если больше 64, то сервис возвращает ответ с кодом 400 и описанием «Максимальная длина 64 символов». |
| password |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 400 | Максимальная длина 64 |

Выход

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/auth/logout  **Method:** POST  **Headers:**  "Content-Type": application/json"  "Authorization": Bearer<token>  **Body:**  {  "token": "token"  } | **Response code:** 200  **Response body:**  **{**  "message": "Вы вышли из системы"  **}** |

Входные параметры:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Параметр** | **Тип данных** | **Обязательность** | **Описание** | **Варианты значений** |
| token | varchar(255) | Да | Токен | eyJhbGciOiJIUzI1NiIsInR5cCI6I  kpXVCJ9 |

Проверки:

|  |  |
| --- | --- |
| **Параметр** | **Условия** |
| token | Максимальная длина 255 символа.  Если больше 255, то сервис возвращает ответ с кодом 400 и описанием «Максимальная длина 255 символов». |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 400 | Максимальная длина 255 |
| 401 | Не авторизован |

Регистрация

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/auth/register  **Method:** POST  **Headers:**  "Content-Type": application/json"  "Accept": application/json  **Body:**  {  "name": "Никита",  "surname": "Ладяев",  "patronymic": "Александрович",  "login": "ladyaev-na",  "password": "qweQWE123"  } | **Response code:** 201  **Response body:**  **{**  "message":"Регистрация успешна",  "user": {  "name": "Никита",  "surname": "Ладяев",  "patronymic":"Александрович",  "login": "ladyaev-na",  "password": "qweQWE123"  },  {  "token": "eyJhbGciOiJIUzI1N  iIsInR5cCI6IkpXVCJ9"  **}** |

Входные параметры:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Параметр** | **Тип данных** | **Обязательность** | **Описание** | **Варианты значений** |
| name | varchar(64) | Да | Имя | Никита |
| surname | varchar(64) | Да | Фамилия | Ладяев |
| patronymic | varchar(255) | Нет | Отчество | Александрович |
| login | varchar(64) | Да | Логин | Логин |
| password | varchar(64) | Да | Пароль | Пароль |

Проверки:

|  |  |  |
| --- | --- | --- |
| **Параметр** | **Условия** | |
| patronymic |  | Максимальная длина 255 символа.  Если больше 255, то сервис возвращает ответ с кодом 400 и описанием «Максимальная длина 255 символов». |
| name |  | Максимальная длина 64 символа.  Если больше 64, то сервис возвращает ответ с кодом 400 и описанием «Максимальная длина 64 символов». |
| surname |  |
| password | Поле должно включать в себя цифры и буквы двух регистров. |
| login | Поле должно быть уникальным.  Если логин повториться, то сервис возвращает ответ с кодом 400 и описанием «Логин уже существует». |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 400 | Максимальная длина 255. |
| Максимальная длина 64. |
| 401 | Не авторизован. |
| 403 | Доступ запрещен |

## Функционал авторизированного пользователя

Просмотр всех бонусов

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/bonuses  **Method:** GET  **Headers:**  "Content-Type": application/json"  "Authorization": Bearer<token> | **Response code:** 200  **Response body:**  **{**  "title": "Приведи друга",  "description": "При пополнение нашей команды ты получаешь бонус",  "price": 5000  **},**  **{**  "title": "Плохая погода",  "description": "При плохой погоде ты будешь получать надбавку за заказ",  "price": 10  **},**  **{**  "title": "Добрый вечер",  "description": "Выходи на доставку с 18 до 23 вечера и получай надбавку за заказ",  "price": 5  **}** |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 401 | Не авторизован |
| 404 | Ресурс не найден |

Запрос для просмотра конкретного бонуса.

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/bonuses/{id}  **Method:** GET  **Headers:**  "Content-Type": application/json"  "Authorization": Bearer<token> | **Response code:** 200  **Response body:**  **{**  "title": "Приведи друга",  "description": "При пополнение нашей команды ты получаешь бонус",  "price": 5000  **}** |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 401 | Не авторизован |
| 404 | Ресурс не найден |

Просмотр профиля

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/profile/{id}  **Method:** GET  **Headers:**  "Content-Type": application/json"  "Authorization": Bearer<token> | **Response code:** 200  **Response body:**  **{**  "name": "Никита",  "surname": "Ладяев",  "patronymic":"Александрович",  "login": "ladyaev-na",  "password": "qweQWE123"  **}** |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 401 | Не авторизован |
| 404 | Ресурс не найден |

Редактирование профиля

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/profile/{id}  **Method:** POST  **Headers:**  "Content-Type": application/json"  "Accept": application/json  **Body:**  {  "name": "Алексей"  } | **Response code:** 200  **Response body:**  **{**  "message":"Редактирование сохранено",  "user": {  "name": "Никита",  "surname": "Ладяев",  "patronymic":"Александрович",  "login": "ladyaev-na",  "password": "qweQWE123"  } |

Входные параметры:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Параметр** | **Тип данных** | **Обязательность** | **Описание** | **Варианты значений** |
| name | varchar(64) | Нет | Имя | Никита |
| surname | varchar(64) | Нет | Фамилия | Ладяев |
| patronymic | varchar(255) | Нет | Отчество | Александрович |
| login | varchar(64) | Нет | Логин | Логин |
| password | varchar(64) | Нет | Пароль | Пароль |

Проверки:

|  |  |  |
| --- | --- | --- |
| **Параметр** | **Условия** | |
| patronymic |  | Максимальная длина 255 символа.  Если больше 255, то сервис возвращает ответ с кодом 400 и описанием «Максимальная длина 255 символов». |
| name |  | Максимальная длина 64 символа.  Если больше 64, то сервис возвращает ответ с кодом 400 и описанием «Максимальная длина 64 символов». |
| surname |  |
| password | Поле должно включать в себя цифры и буквы двух регистров. |
| login | Поле должно быть уникальным.  Если логин повториться, то сервис возвращает ответ с кодом 400 и описанием «Логин уже существует». |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 400 | Максимальная длина 255. |
| Максимальная длина 64. |
| 401 | Не авторизован. |
| 403 | Доступ запрещен |

Просмотр штрафов

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/penalties  **Method:** GET  **Headers:**  "Content-Type": application/json"  "Authorization": Bearer<token> | **Response code:** 200  **Response body:**  **{**  "description": "Отстранение"  **},**  **{**  "description": "Минус час от смены"  **}** |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 401 | Не авторизован |
| 404 | Ресурс не найден |

Просмотр штрафа

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/penalties/{id}  **Method:** GET  **Headers:**  "Content-Type": application/json"  "Authorization": Bearer<token> | **Response code:** 200  **Response body:**  **{**  "description": "Отстранение"  **}** |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 401 | Не авторизован |
| 404 | Ресурс не найден |

## Функционал администратора

Регистрация курьера

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/register  **Method:** POST  **Headers:**  "Content-Type": application/json"  "Accept": application/json  **Body:**  {  "name": "Иван",  "surname": "Иванов",  "patronymic":"Иванович",  "login": "ivanov-ii",  "password": "Ivan1234%",  "id\_role": 1  } | **Response code:** 200  **Response body:**  **{**  "message": "Курьер зарегистрирован",  "user": {  "name": "Иван",  "surname": "Иванов",  "patronymic": "Иванович",  "login": "ivanov-ii",  "password": "Ivan1234%",  "id\_role": 1  } |

Просмотр смен

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/workshifts  **Method:** GET  **Headers:**  "Content-Type": application/json"  "Accept": application/json | **Response code:** 200  **Response body:**  **{**  "day": "12-09-2024",  "startChange": 07:00,  "endChange": 12:00,  "id\_user": 1  } |

Просмотр смен в дне

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/workshifts/day/{id}  **Method:** GET  **Headers:**  "Content-Type": application/json"  "Accept": application/json | **Response code:** 200  **Response body:**  **{**  "day": "12-09-2024",  "startChange": 07:00,  "endChange": 12:00,  "id\_user": 1  } |

Добавление бонуса

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/bonuses/create  **Method:** POST  **Headers:**  "Content-Type": application/json"  "Accept": application/json  **Body:**  {  "title": "Приведи товарища",  "description": "При пополнение нашей команды ты получаешь бонус",  "price": 10000  } | **Response code:** 200  **Response body:**  {  "message": "Бонус создан",  "title": "Приведи товарища",  "description": "При пополнение нашей команды ты получаешь бонус",  "price": 10000  } |

Редактирование бонуса

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/bonuses/{id}/edit  **Method:** POST  **Headers:**  "Content-Type": application/json"  "Accept": application/json  **Body:**  {  "title": "Приведи друга",  "description": "При пополнение нашей команды ты получаешь приятный бонус",  "price": 7500  } | **Response code:** 200  **Response body:**  {  "message": "Бонус изменён",  "title": "Приведи друга",  "description": "При пополнение нашей команды ты получаешь приятный бонус",  "price": 7500  } |

Удаление бонуса

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/bonuses/{id}  **Method:** DELETE  **Headers:**  "Content-Type": application/json"  "Accept": application/json | **Response code:** 204  **Response body:**  **{**  "message": "Бонус удалён"  **}** |

Просмотр профиля курьера

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/profile/{id}  **Method:** GET  **Headers:**  "Content-Type": application/json"  "Accept": application/json | **Response code:** 200  **Response body:**  **{**  "user": {  "name": "Никита",  "surname": "Ладяев",  "patronymic": "Александрович",  "login": "ladyaev-na",  "password": "qweQWE123",  "id\_status": 1  } |

Редактирование профиля курьера

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/profile/{id}/edit  **Method:** POST  **Headers:**  "Content-Type": application/json"  "Accept": application/json  **Body:**  {  "name": "Человек",  } | **Response code:** 200  **Response body:**  {  "message": "Изменения сохранены",  "user": {  "name": "Человек",  "surname": "Ладяев",  "patronymic": "Александрович",  "login": "ladyaev-na",  "password": "qweQWE123",  "id\_status": 1  } |

Удаление профиля курьера

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/profile/{id}  **Method:** DELETE  **Headers:**  "Content-Type": application/json"  "Accept": application/json | **Response code:** 204  **Response body:**  **{**  "message": "Курьер удалён"  **}** |

Просмотр доступности у курьера

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/day/{id}/availability/{id}  **Method:** GET  **Headers:**  "Content-Type": application/json"  "Authorization": Bearer<token> | **Response code:** 200  **Response body:**  **{**  "user": {  "name": "Человек",  "surname": "Ладяев",  "patronymic": "Александрович",  "login": "ladyaev-na",  "password": "qweQWE123",  "id\_status": 1  },  "availability": {  {  "startChange": 07:00,  "endChange": 12:00,  },  {  "startChange": 13:00,  "endChange": 18:00,  }  **}** |

Редактирование профиля курьера

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/profile/{id}/edit  **Method:** POST  **Headers:**  "Content-Type": application/json"  "Accept": application/json  **Body:**  {  "name": "Человек",  } | **Response code:** 200  **Response body:**  {  "message": "Изменения сохранены",  "user": {  "name": "Человек",  "surname": "Ладяев",  "patronymic": "Александрович",  "login": "ladyaev-na",  "password": "qweQWE123",  "id\_status": 1  } |

Просмотр штрафов у курьера

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/profile/{id}/penalties  **Method:** GET  **Headers:**  "Content-Type": application/json"  "Authorization": Bearer<token> | **Response code:** 200  **Response body:**  **{**  "description": "Отстранение"  **},**  **{**  "description": "Минус час от смены"  **}** |

Просмотр конкретного штрафа у курьера

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/profile/{id}/penalties/{id}  **Method:** GET  **Headers:**  "Content-Type": application/json"  "Authorization": Bearer<token> | **Response code:** 200  **Response body:**  **{**  "description": "Отстранение"  **}** |

Создание штрафа у курьера

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/profile/{id}/penalties/create  **Method:** POST  **Headers:**  "Content-Type": application/json"  "Accept": application/json  **Body:**  {  "description": "Отстранение"  } | **Response code:** 200  **Response body:**  {  "message": "Штраф добавлен",  "data": {  "description": " Отстранение "  }  } |

Редактирование штрафа у курьера

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/profile{id}/penalties{id}/edit  **Method:** POST  **Headers:**  "Content-Type": application/json"  "Accept": application/json  **Body:**  {  "description": "Наказан"  } | **Response code:** 200  **Response body:**  {  "message": "Изменения сохранены",  "data": {  "description": "Наказан"  }  } |

Удаление штрафа у курьера

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/profile{id}/penalties{id}/  **Method:** DELETE  **Headers:**  "Content-Type": application/json"  "Accept": application/json | **Response code:** 204  **Response body:**  **{**  "message": "Штраф удалён"  **}** |

## Функционал курьера

Просмотр дня

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/day/{id}/user/{id}  **Method:** GET  **Headers:**  "Content-Type": application/json"  "Authorization": Bearer<token> | **Response code:** 200  **Response body:**  **{**  "роль": "Никита",  "интервал": "Ладяев",  "длительность": "5 часов",  "доставлено": 25,  "оценка": 5  **}** |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 401 | Не авторизован |
| 404 | Ресурс не найден |

Добавление доступности

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/availability  **Method:** POST  **Headers:**  "Content-Type": application/json"  "Accept": application/json  **Body:**  {  "startChange": 07:00,  "endChange": 12:00,  "date": 2024-01-01  } | **Response code:** 201  **Response body:**  **{**  "message":"Интервал добавлен",  "data": {  "startChange": 07:00,  "endChange": 12:00,  "date": 2024-01-01  } |

Входные параметры:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Параметр** | **Тип данных** | **Обязательность** | **Описание** | **Варианты значений** |
| startChange | time | Да | Начало смены | 07:00 |
| endChange | time | Да | Конец смены | 12:00 |
| date | date | Да | Дата смены | 2024-01-01 |

Проверки:

|  |  |
| --- | --- |
| **Параметр** | **Условия** |
| startChange | Поле должно включать в себя время (H:i:s). |
| endChange |
| date | Поле должно включать в себя дату (Y-m-d). |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 400 | Тип поле должно соответствовать времени |
| Тип поле должно соответствовать дате. |
| 401 | Не авторизован. |
| 403 | Доступ запрещен |

Просмотр доступности

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/availability/{id}  **Method:** GET  **Headers:**  "Content-Type": application/json"  "Authorization": Bearer<token> | **Response code:** 200  **Response body:**  **{**  "startChange": 07:00,  "endChange": 12:00,  "date": 2024-01-01  **}** |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 401 | Не авторизован |
| 404 | Ресурс не найден |

Редактирование доступности

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/availability/{id}  **Method:** POST  **Headers:**  "Content-Type": application/json"  "Accept": application/json  **Body:**  {  "endChange": 15:00  } | **Response code:** 200  **Response body:**  **{**  "message":"Интервал редактирован",  "data": {  "startChange": 07:00,  "endChange": 15:00,  "date": 2024-01-01  } |

Входные параметры:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Параметр** | **Тип данных** | **Обязательность** | **Описание** | **Варианты значений** |
| startChange | time | Нет | Начало смены | 07:00 |
| endChange | time | Нет | Конец смены | 12:00 |
| date | date | Да | Дата смены | 2024-01-01 |

Проверки:

|  |  |
| --- | --- |
| **Параметр** | **Условия** |
| startChange | Поле должно включать в себя время (H:i:s). |
| endChange |
| date | Поле должно включать в себя дату (Y-m-d). |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 400 | Тип поле должно соответствовать времени |
| Тип поле должно соответствовать дате. |
| 401 | Не авторизован. |
| 403 | Доступ запрещен |

Удаление доступности

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/availability/{id}  **Method:** DELETE  **Headers:**  "Content-Type": application/json"  "Accept": application/json | **Response code:** 204  **Response body:**  **{**  "message":"Интервал удален"  **}** |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 401 | Не авторизован. |
| 403 | Доступ запрещен |

Просмотр оценки смены

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/estimation/{id}  **Method:** GET  **Headers:**  "Content-Type": application/json"  "Authorization": Bearer<token> | **Response code:** 200  **Response body:**  **{**  "оценка": 5  **}** |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 401 | Не авторизован |
| 404 | Ресурс не найден |

Добавление оценки смены

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/estimation  **Method:** POST  **Headers:**  "Content-Type": application/json"  "Accept": application/json  **Body:**  {  "оценка": 5  } | **Response code:** 201  **Response body:**  **{**  "startChange": 07:00,  "endChange": 12:00,  "date": 2024-01-01  "оценка": 5  } |

Входные параметры:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Параметр** | **Тип данных** | **Обязательность** | **Описание** | **Варианты значений** |
| estimation | int | Да | Оценка смены | 5 |

Проверки:

|  |  |
| --- | --- |
| **Параметр** | **Условия** |
| estimation | Поле должно включать в себя целое число от 1 до 5 (включительно). |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 400 | Тип поле должно соответствовать целому числу |
| 401 | Не авторизован. |
| 403 | Доступ запрещен |

Редактирование оценки смены

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/estimation/{id}  **Method:** POST  **Headers:**  "Content-Type": application/json"  "Accept": application/json  **Body:**  {  "оценка": 3  } | **Response code:** 200  **Response body:**  **{**  "startChange": 07:00,  "endChange": 12:00,  "date": 2024-01-01  "оценка": 3  } |

Входные параметры:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Параметр** | **Тип данных** | **Обязательность** | **Описание** | **Варианты значений** |
| estimation | int | Да | Оценка смены | 5 |

Проверки:

|  |  |
| --- | --- |
| **Параметр** | **Условия** |
| estimation | Поле должно включать в себя целое число от 1 до 5 (включительно). |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 400 | Тип поле должно соответствовать целому числу |
| 401 | Не авторизован. |
| 403 | Доступ запрещен |

Удаление оценки смены

|  |  |
| --- | --- |
| **ЗАПРОС** | **ОТВЕТ** |
| **URL:** {host}/estimation/{id}  **Method:** POST  **Headers:**  "Content-Type": application/json"  "Accept": application/json | **Response code:** 200  **Response body:**  **{**  "message":"Оценка удалена"  } |

Ответ с ошибками:

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 401 | Не авторизован. |
| 403 | Доступ запрещен |