**ОБЛАСТНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

**«ТОМСКИЙ ЭКОНОМИКО-ПРОМЫШЛЕННЫЙ КОЛЛЕДЖ»**

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ НА РАЗРАБОТКУ API СЕРВИСА**

***ОТЧЁТ ПО УЧЕБНОЙ ПРАКТИКЕ***

*СПЕЦИАЛЬНОСТЬ 09.02.07 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ПРОГРАММИРОВАНИЕ*

Выполнил:

Студент группы 1912с

\_\_\_\_\_\_\_\_\_\_\_\_ / Заикин В.А.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2024 г.

Проверил:

Руководитель УП

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

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2024 г.

Томск 2024

**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ** 3](#_Toc165032721)

[**Информация о проекте и команде разработчиков** 3](#_Toc165032722)

[**Назначение и структура документа** 3](#_Toc165032723)

[**Глоссарий** 3](#_Toc165032724)

[**ОБЩЕЕ ОПИСАНИЕ** 4](#_Toc165032725)

[**Общий взгляд на предлагаемое решение** 4](#_Toc165032726)

[**Ролевая модель** 5](#_Toc165032727)

[**ТЕХНИЧЕСКАЯ РЕАЛИЗАЦИЯ** 9](#_Toc165032728)

[**Концептуальная модель данных** 9](#_Toc165032729)

[**Логическая модель данных** 10](#_Toc165032730)

[**Сущности** 10](#_Toc165032731)

[**Описание API** 14](#_Toc165032732)

[***Формат ответа общих требований*** 14](#_Toc165032733)

[***Авторизация*** 15](#_Toc165032734)

[***Регистрация*** 15](#_Toc165032735)

[***Выход из системы*** 16](#_Toc165032736)

[***Просмотр списка пользователей*** 16](#_Toc165032737)

[***Авторизация пользователя*** 17](#_Toc165032738)

[**ТЕСТИРОВАНИЕ** 17](#_Toc165032739)

[**Авторизация пользователя** 17](#_Toc165032740)

[**РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ** 19](#_Toc165032741)

# **ВВЕДЕНИЕ**

## **Информация о проекте и команде разработчиков**

|  |  |
| --- | --- |
| Название проекта |  |
| Цель проекта |  |
| Задачи проекта | 1. Разработать технического задания API сервиса. 2. Разработать технического задания веб-сайта. 3. Разработать технического задания десктоп приложения. 4. Разработать API сервиса на фреймворке Laravel 10. 5. Разработать веб-сайта на фреймворке Vue 3. 6. Разработать десктоп приложения на C# WPF. 7. Разработать автоматизированные тесты и провести тестирование API сервиса. 8. Разработать автоматизированные тесты и провести тестирование веб-сайта. 9. Разработать автоматизированные тесты и провести тестирование десктоп приложения. |
| Название команды | Команда «Батискаф» |
| Состав команды и роли | Заикин Вадим Анатольевич – Аnalyst, Backend developer, QA.  Шейкина Елизавета Викторовна – Аnalyst, Frontend developer, QA.  Гусейнов Мурад Шахин Оглы – Аnalyst, C# developer, QA. |

## **Назначение и структура документа**

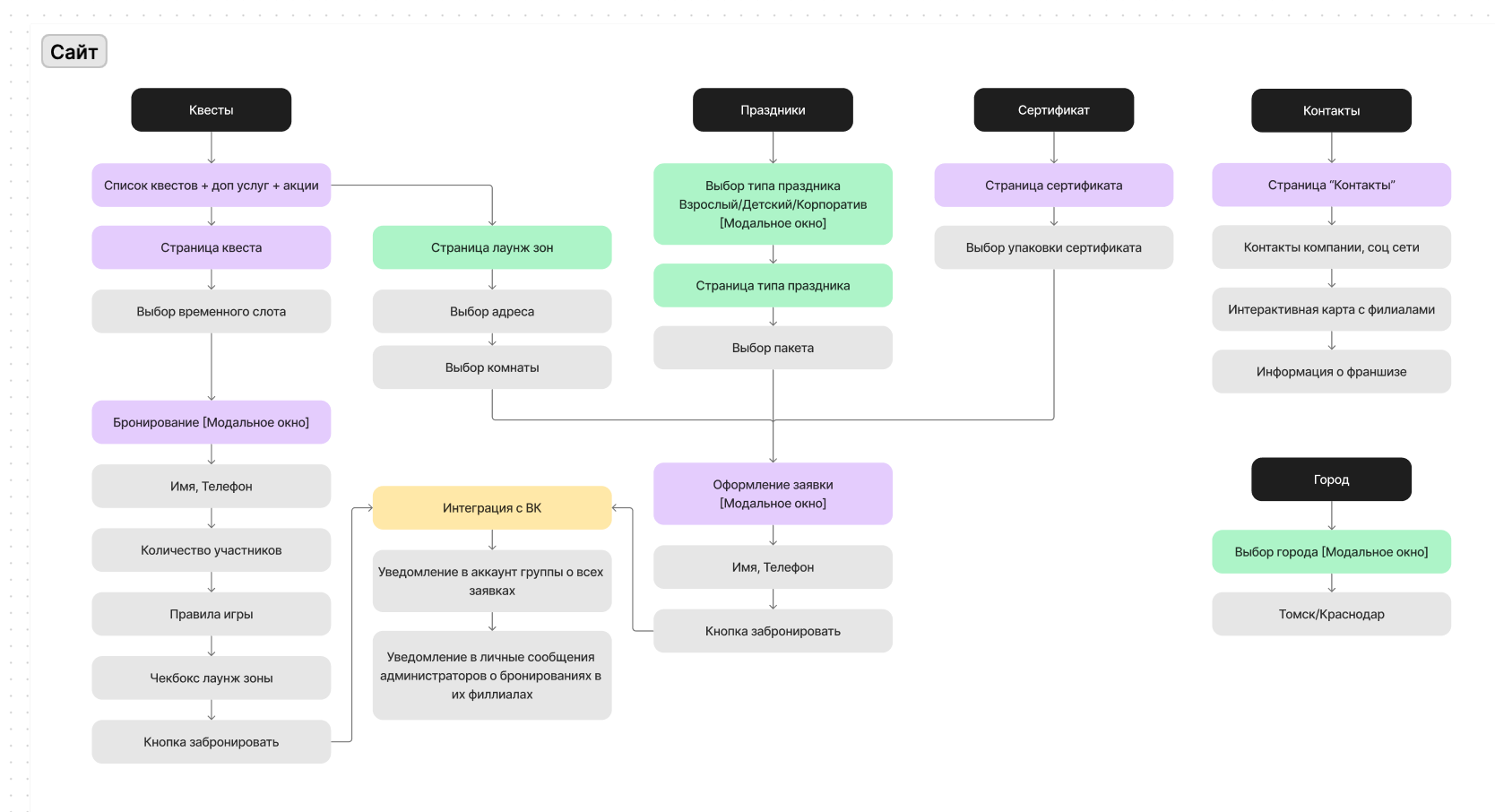
Настоящий документ рассчитан на использование в проектной команде в качестве ключевого документа, определяющего требование по реализации задачи по разработке API сервиса для проекта «…».

## **Глоссарий**

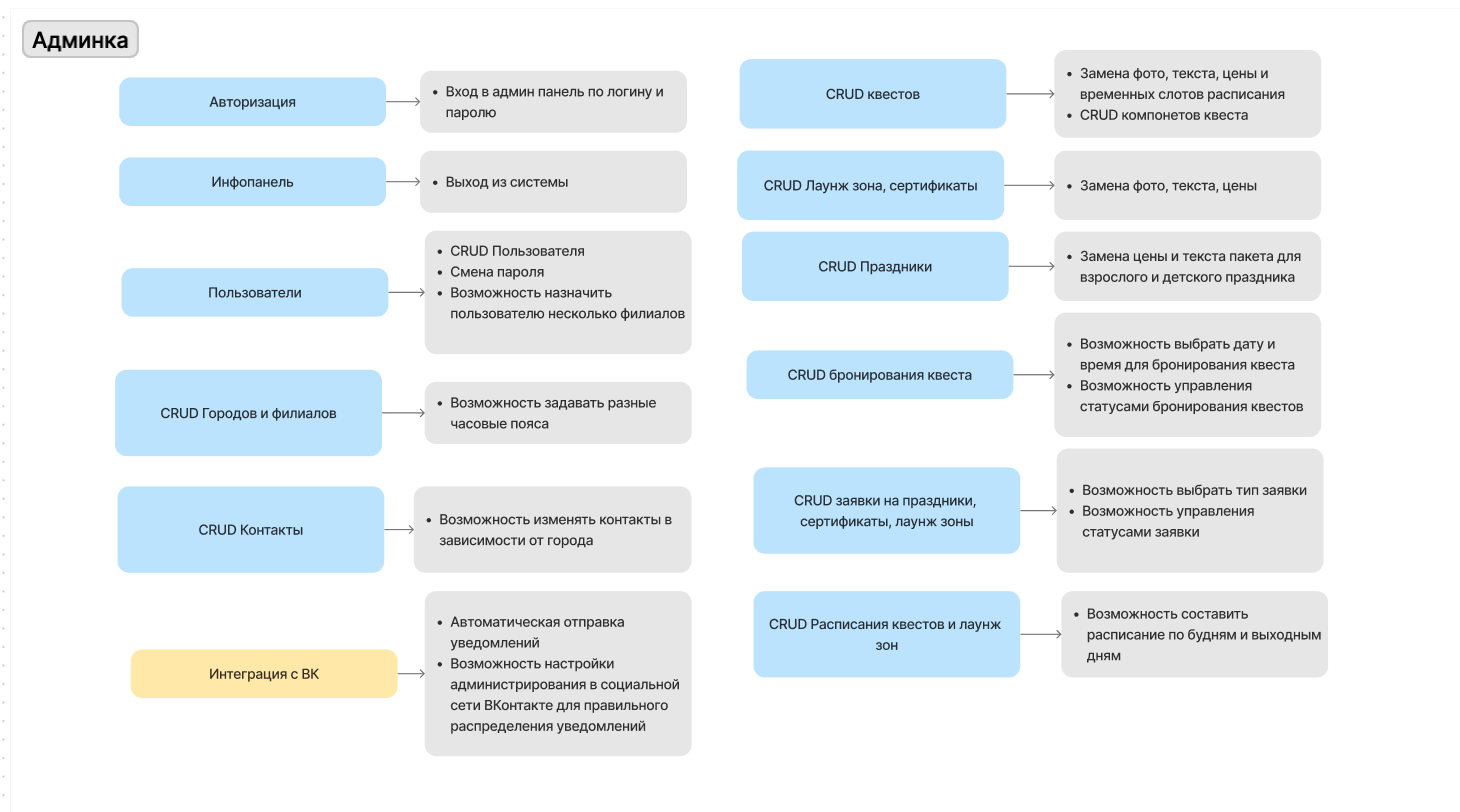
|  |  |
| --- | --- |
| **Понятие** | **Определение понятия** |
| API | API (интерфейс программирования приложений) – это набор определенных правил и соглашений, который используется для взаимодействия между различными программными компонентами. API определяет, как различные части программного обеспечения могут взаимодействовать друг с другом, обычно путем обмена данными и выполнения определенных операций. |
| Postman | Postman – это инструмент для тестирования и разработки API. Он предоставляет удобный интерфейс для отправки HTTP запросов к API, а также для анализа ответов. |
|  |  |
|  |  |
|  |  |

# **ОБЩЕЕ ОПИСАНИЕ**

## **Общий взгляд на предлагаемое решение**



(вставить свою концепцию)



(вставить свою концепцию)

## **Ролевая модель**

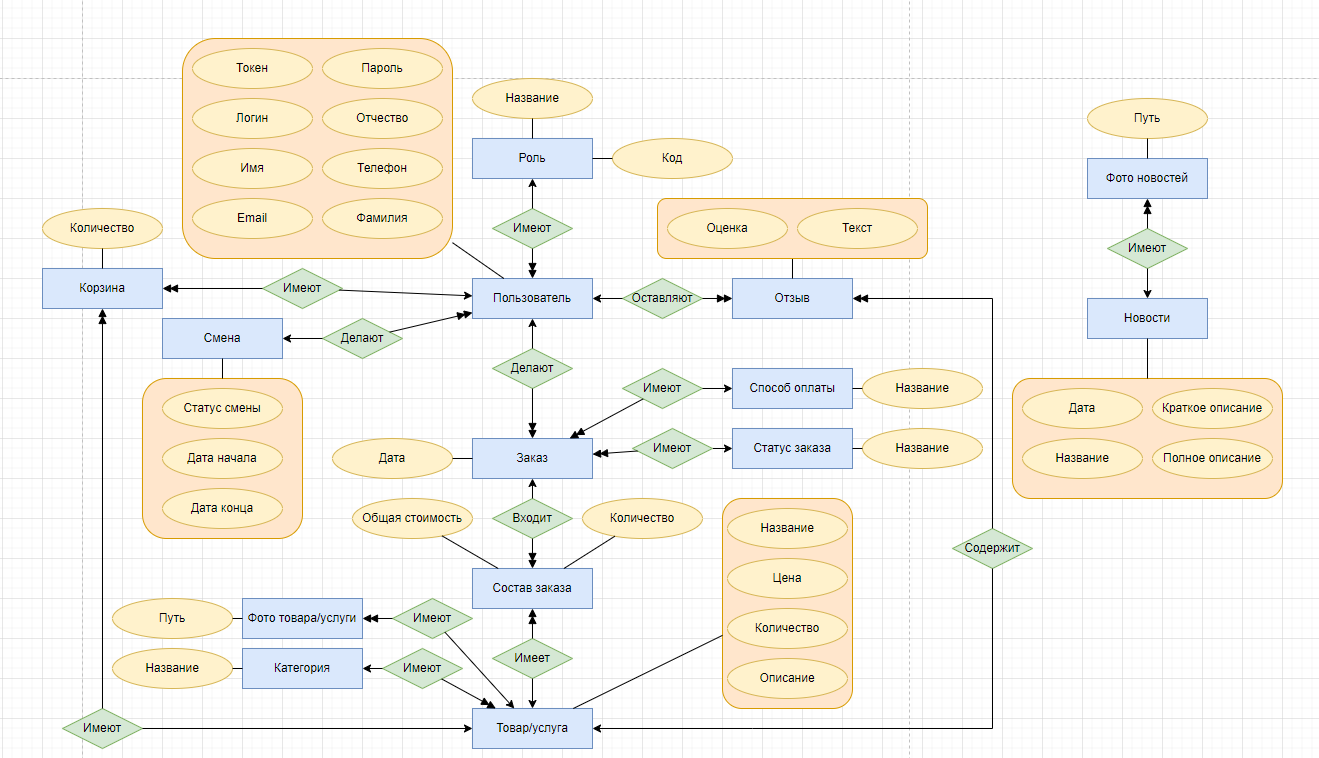
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Раздел** | **Фичи** | **Админ** | **Менеджер** | **Сотрудник** | **Клиент** |
| Инфопанель | ✅Выход из системы | да | да | да | да |
| Пользователи | ✅Просмотр списка пользователей | да | да | нет | нет |
| ✅Просмотр своего профиля | да | да | да | да |
| ✅Регистрация | да | да | да | да |
| ✅Создание пользователя с нужной ролью | да | нет | нет | нет |
| ✅Редактирование пользователя | да | нет | нет | нет |
| ✅Изменение пароля пользователя | да | да (только себя) | да (только себя) | да (только себя) |
| ✅Удаление пользователя | да | нет | нет | нет |
| ✅Просмотреть конкретного пользователя | да | да | нет | нет |
| Роли | ✅Создание роли | да | нет | нет | нет |
| ✅Удаление роли | да | нет | нет | нет |
| ✅Редактирование роли | да | нет | нет | нет |
| Смена | ✅Добавление на смену | да (только сотрудников) | да (только сотрудников) | нет | нет |
| ✅Удаление со смены | да (только сотрудников) | да (только сотрудников) | нет | нет |
| ✅Просмотр смен | да | да | да | нет |
| ✅Просмотр конкретной смены | да | да | да | нет |
| ✅Создание смены | да | да | нет | нет |
| ✅Удаление смены | да | да | нет | нет |
| ✅Редактирование смены | да | да | нет | нет |
| Корзина | ✅Добавление товара в корзину | да | да | да (только в свою) | да (только в свою) |
| ✅Редактирование корзины | да | да | да (только свою) | да (только свою) |
| ✅Удаление товара из корзины | да | да | да (только из своей) | да (только из своей) |
| Просмотр корзины другого пользователя | да | да | нт | нет |
| ✅Просмотр своей корзины | да | да | да | да |
| Новости | ✅Просмотр новостей | да | да | да | да |
| ✅Просмотр конкретной новости | да | да | да | да |
| ✅Редактирование новости | да | нет | нет | нет |
| ✅Удаление новостей | да | нет | нет | нет |
| ✅Создание новостей | да | нет | нет | нет |
| Фото новостей | ✅Просмотр всех фото конкретной новости | да | да | да | да |
| ✅Добавление фото | да | нет | нет | нет |
| ✅Удаление фото | да | нет | нет | нет |
| ✅Редактирование фото | да | нет | нет | нет |
| Фото товаров | ✅Просмотр всех фото конкретного товара | да | да | да | да |
| ✅Добавление фото | да | нет | нет | нет |
| ✅Удаление фото | да | нет | нет | нет |
| ✅Изменение фото | да | нет | нет | нет |
| Отзывы | ✅Просмотр отзывов | да | да | да | да |
| ✅Просмотр отзывов у конкретного товара | да | да | да | да |
| ✅Создание отзыва | да (только на купленный товар) | да (только на купленный товар) | да (только на купленный товар) | да (только на купленный товар) |
| ✅Редактирование отзыва | да | нет | нет | нет |
| ✅Удаление отзыва | да | нет | нет | нет |
| Способы оплаты | ✅Просмотр способов оплаты | да | да | да | да |
| ✅Редактирование способов оплаты | да | нет | нет | нет |
| ✅Создание способа оплаты | да | нет | нет | нет |
| ✅Удаление способа оплаты | да | нет | нет | нет |
| Статус заказов | ✅Просмотр статуса заказов | да | да | да | да |
| ✅Создание статуса заказа | да | нет | нет | нет |
| ✅Редактирование статуса заказа | да | нет | нет | нет |
| ✅Удаление статуса заказа | да | нет | нет | нет |
| Категории | ✅Просмотр категорий | да | да | да | да |
| ✅Создание категорий | да | нет | нет | нет |
| ✅Редактирование категорий | да | нет | нет | нет |
| ✅Удаление категорий | да | нет | нет | нет |
| Состав заказа | ✅Просмотр конкретного состава заказа |  |  |  |  |
| ✅Просмотр составов заказа по идентификатору заказа |  |  |  |  |
| ✅Редактирование состава заказа |  |  |  |  |
| ✅Удаление состава заказа |  |  |  |  |
| Заказы | ✅Оформление заказа | да | да | да | да |
| ✅Редактирование заказа | да | нет | нет | нет |
| ✅Удаление заказа | да | нет | нет | нет |
| ✅Просмотр заказов | да | да | да (только своих) | да (только своих) |
| ✅Прикрепление сотрудника к заказу | да | да | да (только себя) | нет |
| ✅Открепление сотрудника от заказа | да | да | да (только себя) | нет |
| Отчёты | ✅Просмотр всех заказов по конкретному товару и общей выручки за всё время, а также количеством заказов для данного товара и количество купленного товара | да | нет | нет | нет |
| ✅Просмотр всех заказов и общей выручки, заказов за конкретный ГГГГ.ММ.ДД | да | нет | нет | нет |
| ✅Просмотр всех заказов и общей выручки за период от ГГГГ.ММ.ДД до ГГГГ.ММ.ДД | да | нет | нет | нет |
| ✅Просмотр всех заказов по конкретному товару и общей выручки за период ГГГГ.ММ.ДД до ГГГГ.ММ.ДД, а также количеством заказов для данного товара и количество купленного товара | да | нет | нет | нет |

# **ТЕХНИЧЕСКАЯ РЕАЛИЗАЦИЯ**

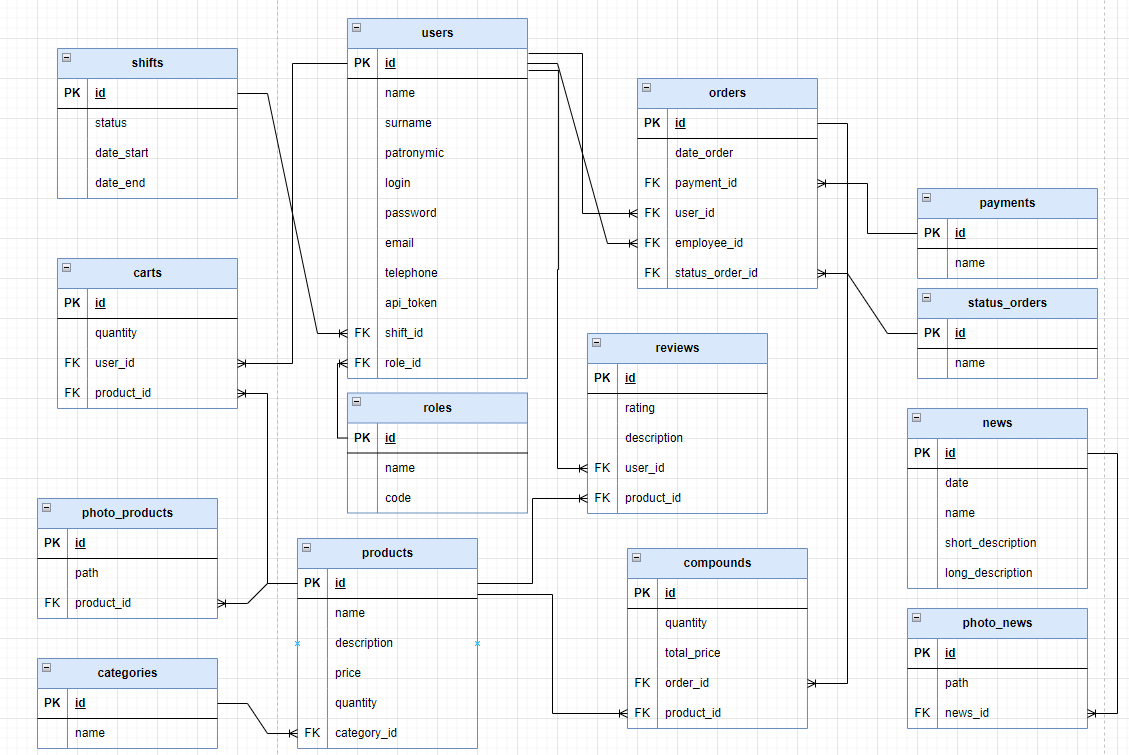
## **Концептуальная модель данных**

Описание концептуальной модели:

1. Пользователь: содержит информацию о пользователе, которые имеют доступ к системе внутри компании. Может содержать поля, такие как имя, фамилия, логин, пароль, телефон, почта, токен.
2. Роль: содержит информацию о роли пользователя. Может содержать поле название.
3. Смена: содержит информацию о дате начала и конца смены. Может содержать поля, такие как дата начала, дата конца, статус смены.
4. Отзыв: содержит информацию о тексте отзывы, а также его оценки. Может содержать поля, такие как текст, оценка.
5. Новости: содержит информацию о название новости, дате публикации, а также описание. Может содержать поля, такие как описание, дата, название.
6. Фото: содержит информацию о пути до картинки. Может содержать поле путь.
7. Категория: содержит информацию о название категории. Может содержать поле название.
8. Товар/Услуга: содержит информацию о названии, описании, цены и количества.
9. Способ оплаты: содержит информацию о наименовании способа оплаты.
10. Заказ: содержит информацию о дате заказа.
11. Состав заказа: содержит информацию об общей стоимости, количества.
12. Статус: содержит информацию о названии статуса.



## **Логическая модель данных**



## **Сущности**

**roles (роли)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Ограничение** | **Описание** |
| id | biginteger | PK  AUTO\_INCREMENT | Идентификатор роли |
| name | varchar (64) | NOT NULL | Имя роли |
| code | varchar (64) | NOT NULL  UNIQUE | Код роли |

**users (пользователи)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Ограничение** | **Описание** |
| id | biginteger | PK  AUTO\_INCREMENT | Идентификатор пользователя |
| name | varchar(128) | NOT NULL | Имя пользователя |
| surname | varchar(128) | NOT NULL | Фамилия пользователя |
| patronymic | varchar(128) | NULL | Отчество пользователя |
| login | varchar(128) | NOT NULL  UNIQUE | Логин пользователя |
| password | varchar(128) | NOT NULL | Пароль пользователя |
| email | varchar(128) | NOT NULL  UNIQUE | Электронная почта пользователя |
| telephone | biginteger | NOT NULL  UNIQUE | Номер телефона пользователя |
| api\_token | varchar(255) | NULL  UNIQUE | Токен пользователя |
| shift\_id | biginteger | FK  NULL | Идентификатор смены |
| role\_id | biginteger | FK  NOT NULL | Идентификатор роли |

**products (товары)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Ограничение** | **Описание** |
| id | biginteger | PK  AUTO\_INCREMENT | Идентификатор товара |
| name | varchar(64) | NOT NULL | Название товара |
| description | text | NULL | Описание товара |
| price | decimal (15, 2) | NOT NULL | Цена товара |
| quantity | int | NOT NULL | Количество товара |
| category\_id | biginteger | FK  NOT NULL | Идентификатор категории |

**carts (корзина)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Ограничение** | **Описание** |
| id | biginteger | PK  AUTO\_INCREMENT | Идентификатор корзины |
| quantity | int | NOT NULL | Количество товара в корзине |
| user\_id | biginteger | NOT NULL | Идентификатор пользователя |
| product\_id | biginteger | NOT NULL | Идентификатор товара |

**shifts (смены)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Ограничение** | **Описание** |
| id | biginteger | PK  AUTO\_INCREMENT | Идентификатор смены |
| status | varchar(64) | NOT NULL | Статус смены |
| date\_start | datetime | NOT NULL | Начало смены |
| date\_end | datetime | NOT NULL | Конец смены |

**photo\_products (фотографии товаров)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Ограничение** | **Описание** |
| id | biginteger | PK  AUTO\_INCREMENT | Идентификатор фотографий товаров |
| path | varchar(255) | NOT NULL | Путь до фотографии |
| product\_id | biginteger | FK  NOT NULL | Идентификатор товара |

**categories (категории)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Ограничение** | **Описание** |
| id | biginteger | PK  AUTO\_INCREMENT | Идентификатор категорий |
| name | varchar(64) | NOT NULL | Название категории |

**orders (заказы)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Ограничение** | **Описание** |
| id | biginteger | PK  AUTO\_INCREMENT | Идентификатор заказа |
| date\_order | datetime | NOT NULL | Дата оформления заказа |
| payment\_id | biginteger | FK  NOT NULL | Идентификатор способа оплаты |
| user\_id | biginteger | FK  NOT NULL | Идентификатор пользователя |
| employee\_id | biginteger | FK  NULL | Идентификатор сотрудника |
| status\_order\_id | biginteger | FK  NOT NULL | Идентификатор статуса заказа |

**reviews (отзывы)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Ограничение** | **Описание** |
| id | biginteger | PK  AUTO\_INCREMENT | Идентификатор отзыва |
| rating | int | NOT NULL | Оценка отзыва |
| description | varchar(255) | NOT NULL | Описание отзыва |
| user\_id | biginteger | FK  NOT NULL | Идентификатор пользователя |
| product\_id | biginteger | FK  NOT NULL | Идентификатор товара |

**compounds (состав заказов)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Ограничение** | **Описание** |
| id | biginteger | PK  AUTO\_INCREMENT | Идентификатор состава заказа |
| quantity | int | NOT NULL | Количество товаров |
| total\_price | decimal (15, 2) | NOT NULL | Итоговая цена |
| order\_id | biginteger | FK  NOT NULL | Идентификатор заказа |
| product\_id | biginteger | FK  NOT NULL | Идентификатор товара |

**payments (способы оплаты)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Ограничение** | **Описание** |
| id | biginteger | PK | Идентификатор |
| name | varchar(64) | NOT NULL  AUTO\_INCREMENT | Название способа оплаты |

**status\_orders (статусы заказов)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Ограничение** | **Описание** |
| id | biginteger | PK  AUTO\_INCREMENT | Идентификатор статуса заказа |
| name | varchar(64) | NOT NULL | Название статуса заказа |

**news (новости)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Ограничение** | **Описание** |
| id | biginteger | PK  AUTO\_INCREMENT | Идентификатор новости |
| date | datetime | NOT NULL | Дата публикации новости |
| name | varchar(64) | NOT NULL | Название новости |
| short\_description | varchar(255) | NOT NULL | Короткое описание новости |
| long\_description | text | NOT NULL | Длинное описание новости |

**photo\_news (фотографии новостей)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Атрибут** | **Тип** | **Ограничение** | **Описание** |
| id | biginteger | PK  AUTO\_INCREMENT | Идентификатор |
| path | varchar(255) | NOT NULL | Путь до фотографии |
| news\_id | biginteger | FK  NOT NULL | Идентификатор новости |

## **Описание API**

### ***Формат ответа общих требований***

1. Попытка отправить некорретный запрос

Status: 400

Content-Type: application/json

Body:

|  |
| --- |
| {      "success": **false**,      "code": 400,      "message": "Некорректный запрос"  } |

1. Попытка доступа гостя к защищенным авторизацией функциям системы.

Status: 401

Content-Type: application/json

Body:

|  |
| --- |
| {      "success": **false**,      "code": 401,      "message": "Ошибка авторизации"  } |

1. Попытка доступа авторизованного пользователя к функциям недоступным для него.

Status: 403

Content-Type: application/json

Body:

|  |
| --- |
| {      "success": **false**,      "code": 403,      "message": "Доступ запрещён"  } |

1. Попытка получить несуществующий ресурс.

Status: 404

Content-Type: application/json

Body:

|  |
| --- |
| {      "success": **false**,      "code": 404,      "message": "Не найдено"  } |

1. Попытка отправить неприемлимый запрос

Status: 406

Content-Type: application/json

Body:

|  |
| --- |
| {      "success": **false**,      "code": 406,      "message": "Неприемлимо"  } |

1. В случае ошибок связанных с валидацией данных.

Status: 422

Content-Type: application/json

Body:

|  |
| --- |
| {      "success": **false**,      "code": 422,      "message": {          "key1": [              "Error message 1."          ],          "keyN": [              "Error message N."          ]      }  } |

### ***Авторизация***

Метод позволяет получить токен и роль авторизированного пользователя.

Параметры запроса:

* login – обязательное поле, строка, от 5 до 128 символов;
* password – обязательное поле, строка, от 5 до 255 символов;

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host}}/login  **Method:** POST  **Headers:** Content-Type: application/json  **Body:**  {      "login": "ivan",      "password": "ivan"  } | Успех  **Status**: 200  **Content-Type:** application/json  **Body:**  {      "api\_token": "string",      "role": "string"  }  Ошибки валидации полей  **Формат ответа из общих требований**  Ошибка авторизации  **Формат ответа из общих требований** |

### ***Регистрация***

Метод позволяет пользователю создать аккаунт.

Параметры запроса:

* name – обязательное поле, строка, до 128 символов;
* surname - обязательное поле, строка, до 128 символов;
* patronymic – необязательное поле, строка, до 128 символов;
* login - обязательное поле, строка, уникальное, до 128 символов;
* password – обязательное поле, строка, от 5 до 255 символов, иметь минимум 1 заглавную букву, 1 строковую букву, 1 цифру, 1 специальный символ;
* telephone – обязательно поле, числовой, уникальный, от 1 до 20 цифр;
* email - обязательное поле, строка, до 128 символов;

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host}}/register  **Method:** POST  **Headers:** Content-Type: application/json  **Body:**  {  "surname": "Usertest",  "name": "Usertest",  "login": "Usertest",  "password": "Usertest1!",  "telephone": "77777",  "email": "Usertest@mail.ru"  } | Успех  **Status**: 200  **Content-Type:** application/json  **Body:**  {      "api\_token": "string",      "role": "string"  }  Ошибки валидации полей  **Формат ответа из общих требований** |

### ***Выход из системы***

Метод позволяет выйти из системы.

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host}}/logout  **Method:** GET  **Headers:** Authorization: Bearer {{api\_token}}  **Body:**  **{**  **}** | Успех  **Status**: 204  **Content-Type:** application/json  **Body:**  **{**  **}**  Ошибка авторизации  **Формат ответа из общих требований** |

### ***Просмотр списка пользователей***

Метод позволяет посмотреть список пользователей.

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host}}/users  **Method:** GET  **Headers:** Authorization: Bearer {{api\_token}}  **Body:** | Успех  **Status**: 200  **Content-Type:** application/json  **Body:**  **{**      "data": [          {              "id": 1,              "name": "**string**",              "surname": "**string**",              "patronymic": "**string**",              "login": "**string**",              "password": "**string**",              "email": "**string**",              "telephone": integer,              "api\_token": "**string**",              "role\_id": 1,              "shift\_id": **null**,              "created\_at": **null**,              "updated\_at": **null**,              "shifts": "**string**",              "role": "**string**"          },  {  },  ]  **}**  Ошибки валидации полей  **Формат ответа из общих требований** |

### ***Просмотр категорий товаров***

Метод позволяет получить категории.

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host}}/categories  **Method:** GET  **Headers:** Content-Type: application/json  **Body:**  {  } | Успех  **Status**: 200  **Content-Type:** application/json  **Body:**  {      "data": [          {              "id": 1,              "name": "Кресты",              "created\_at": **null**,              "updated\_at": **null**          },          {              "id": 2,              "name": "Гробы",              "created\_at": **null**,              "updated\_at": **null**          },  {  },  ]  }  Попытка получить несуществующий ресурс  **Формат ответа из общих требований** |

### ***Просмотр товаров определённой категории***

Метод позволяет получить товары определённой категории.

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host}}/category/{id}  **Method:** GET  **Headers:** Content-Type: application/json  **Body:**  {  } | Успех  **Status**: 200  **Content-Type:** application/json  **Body:**  {      "data": [          {              "id": 5,              "name": "Крест дубовый с домиком",              "description": **null**,              "price": "8000.00",              "quantity": 9999,              "category\_id": 1,              "created\_at": **null**,              "updated\_at": **null**          },  **{**  **},**  **]**  **}**  Попытка получить несуществующий ресурс  **Формат ответа из общих требований** |

### ***Просмотр конкретного товара***

Метод позволяет посмотреть информацию о конкретном товаре.

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host}}/product/{id}  **Method:** GET  **Headers:** Content-Type: application/json  **Body:**  {  } | Успех  **Status**: 200  **Content-Type:** application/json  **Body:**  {      "data": {          "id": 1,          "name": "Гроб простой без постели",          "description": "Материал: Сосновая доска\\r\\nОтделка гроба: Ткань бордового цвета",          "price": "6500.00",          "quantity": 9999,          "category\_id": 2,          "created\_at": **null**,          "updated\_at": **null**      }  }  Попытка получить несуществующий ресурс  **Формат ответа из общих требований** |

### ***Просмотр всех товаров***

Метод позволяет получить все товары.

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host}}/products  **Method:** GET  **Headers:** Content-Type: application/json  **Body:**  {  } | Успех  **Status**: 200  **Content-Type:** application/json  **Body:**  {      "data": [          {              "id": 1,              "name": "Гроб простой без постели",              "description": "Материал: Сосновая доска\\r\\nОтделка гроба: Ткань бордового цвета",              "price": "6500.00",              "quantity": 9999,              "category\_id": 2,              "created\_at": **null**,              "updated\_at": **null**          },  {  },  ]  }  Попытка получить несуществующий ресурс  **Формат ответа из общих требований** |

### ***Просмотр отзывов у товара***

Метод позволяет получить отзывы у конкретного товара.

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host}}/product/{id}/review  **Method:** GET  **Headers:** Content-Type: application/json  **Body:**  {  } | Успех  **Status**: 200  **Content-Type:** application/json  **Body:**  {      "data": [          {              "id": 2,              "rating": 4,              "description": "Никому не желаю потерять близкого человека. Зато крест с домиком, а ты нет.",              "user\_id": 7,              "product\_id": 5,              "created\_at": **null**,              "updated\_at": **null**          }  {  },  ]  }  Попытка получить несуществующий ресурс  **Формат ответа из общих требований** |

### ***Просмотр всех отзывов***

Метод позволяет получить все отзывы.

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host}}/reviews  **Method:** GET  **Headers:** Content-Type: application/json  **Body:**  {  } | Успех  **Status**: 200  **Content-Type:** application/json  **Body:**  {      "data": [          {              "id": 1,              "rating": 5,              "description": "Очень красивый венок",              "user\_id": 6,              "product\_id": 9,              "created\_at": **null**,              "updated\_at": **null**          },  {  },  ]  }  Попытка получить несуществующий ресурс  **Формат ответа из общих требований** |

### ***Просмотр всех новостей***

Метод позволяет получить категории.

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host}}/categories  **Method:** GET  **Headers:** Content-Type: application/json  **Body:**  {  } | Успех  **Status**: 200  **Content-Type:** application/json  **Body:**  {      "data": [          {              "id": 1,              "name": "Кресты",              "created\_at": **null**,              "updated\_at": **null**          },          {              "id": 2,              "name": "Гробы",              "created\_at": **null**,              "updated\_at": **null**          },  {  },  ]  }  Попытка получить несуществующий ресурс  **Формат ответа из общих требований** |

### ***Просмотр категорий товаров***

Метод позволяет получить категории.

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host}}/categories  **Method:** GET  **Headers:** Content-Type: application/json  **Body:**  {  } | Успех  **Status**: 200  **Content-Type:** application/json  **Body:**  {      "data": [          {              "id": 1,              "name": "Кресты",              "created\_at": **null**,              "updated\_at": **null**          },          {              "id": 2,              "name": "Гробы",              "created\_at": **null**,              "updated\_at": **null**          },  {  },  ]  }  Попытка получить несуществующий ресурс  **Формат ответа из общих требований** |

# **ТЕСТИРОВАНИЕ**

## **Авторизация пользователя**

# **РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ**