**Руководство программиста для API**

Данное руководство предназначено для разработчиков, которые будут использовать API для взаимодействия с системой управления складом и пользователями. API предоставляет методы для аутентификации пользователей, управления товарами и складами.

**Общие сведения**

API использует стандартные HTTP-методы (GET, POST) для выполнения операций. Все запросы и ответы передаются в формате JSON. Базовый URL для всех запросов: <http://localhost:60461/api>

**Аутентификация и авторизация**

1. **Обычная авторизация**

**Метод: POST**

**URL:** <http://localhost:60461/api/auth/login>

**Тело запроса:**

{

"email": "valeria1739poglazova@gmail.com",

  "password": "tweette"

}

**Ответ:**

**Успешная аутентификация:**

{

    "IDПользователя": 3,

    "ИмяПользователя": "Николай Шишкин",

    "Email": "valeria1739poglazova@gmail.com",

    "ХэшПароля": "tweette",

    "Роль": "Кладовщик",

    "Фото": "Строка двоичных данных (из бд)",

}

**Ошибка аутентификации:** 401 Unauthorized

1. **Запрос токена для 2FA**

**Метод: POST**

**URL:** <http://localhost:60461/api/auth/request-2fa>

**Тело запроса:**

{

"email": "valeria1739poglazova@gmail.com",

}

**Ответ:**

**Успешный запрос:**

{

  "Message": "2FA token generated."

}

**Ошибка:** 400 Bad Request или 500 Internal Server Error

1. **Вход с 2FA**

**Метод: POST**

**URL:** <http://localhost:60461/api/auth/login-2fa>

**Тело запроса:**

{

"email": "valeria1739poglazova@gmail.com",

  "token": "335533"

}

**Ответ:**

**Успешная аутентификация:**

{

    "IDПользователя": 5,

    "ИмяПользователя": "Михаил Булгаков",

    "Email": "valeria1739poglazova@gmail.com",

    "ХэшПароля": "ti9r9et",

    "Роль": "Менеджер по продажам",

    "Фото": "Строка двоичных данных (из бд)"

}

**Ошибка аутентификации:** 401 Unauthorized

**Управление товарами**

1. **Получение списка товаров**

**Метод: GET**

**URL:** <http://localhost:60461/api/Products>

**Ответ:**

Выводит весь список товаров, хранящихся в базе данных.

1. **Добавление нового товара**

**Метод: POST**

**URL:** <http://localhost:60461/api/Products>

**Тело запроса:**

    {

        "IDТовара": 6,

        "Название": "APC",

        "Артикул": "ART463",

        "Штрихкод": "123646789",

        "Категория": "Фрукты",

        "ЕдиницаИзмерения": "кг",

        "Цена": 1.00,

        "СерийныйНомер": "SN-011",

        "МинимальныйЗапас": 500

    }

**Ответ:**

**Успешное добавление:** при успешном запросе, данные, представленные в виде JSON файла, заносятся в базу данных.

**Ошибка:** 400 Bad Request

**Управление складами**

1. **Получение списка складов**

**Метод: GET**

**URL:** <http://localhost:60461/api/Warehouse>

**Ответ:** Выводит весь список складов, хранящихся в базе данных.

**Модели данных**

**Response\_ Products:**

public class Response\_Products

{

public Response\_Products(Товары товары)

{

IDТовара = товары.IDТовара;

Название = товары.Название;

Артикул = товары.Артикул;

Штрихкод = товары.Штрихкод;

Категория = товары.Категория;

ЕдиницаИзмерения = товары.ЕдиницаИзмерения;

Цена = товары.Цена;

СерийныйНомер = товары.СерийныйНомер;

МинимальныйЗапас = товары.МинимальныйЗапас;

}

public int IDТовара { get; set; }

public string Название { get; set; }

public string Артикул { get; set; }

public string Штрихкод { get; set; }

public string Категория { get; set; }

public string ЕдиницаИзмерения { get; set; }

public decimal Цена { get; set; }

public string СерийныйНомер { get; set; }

public Nullable<int> МинимальныйЗапас { get; set; }

}

**Response\_Warehouse:**

public class Response\_Warehouse

{

public Response\_Warehouse(Склады склады)

{

IDСклада = склады.IDСклада;

Название = склады.Название;

Адрес = склады.Адрес;

ТипСклада = склады.ТипСклада;

ЗонаХранения = склады.ЗонаХранения;

IDТовара = склады.IDТовара;

Количество = склады.Количество;

}

public int IDСклада { get; set; }

public string Название { get; set; }

public string Адрес { get; set; }

public string ТипСклада { get; set; }

public string ЗонаХранения { get; set; }

public Nullable<int> IDТовара { get; set; }

public Nullable<int> Количество { get; set; }

}

**Response\_Пользователи:**

public class Response\_Пользователи

{

public Response\_Пользователи(Пользователи юзер)

{

IDПользователя = юзер.IDПользователя;

ИмяПользователя = юзер.ИмяПользователя;

Email = юзер.Email;

ХэшПароля = юзер.ХэшПароля;

Роль = юзер.Роль;

Фото = юзер.Фото;

}

public int IDПользователя { get; set; }

public string ИмяПользователя { get; set; }

public string Email { get; set; }

public string ХэшПароля { get; set; }

public string Роль { get; set; }

public byte[] Фото { get; set; }

}

**LoginRequest:**

public class LoginRequest

{

public string Email { get; set; }

public string Password { get; set; }

}

**Login2FARequest:**

public class Login2FARequest

{

public string Email { get; set; }

public string Token { get; set; }

}