**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ**

**Ордена Трудового Красного Знамени**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

Кафедра «Информационных технологий»

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

К Проектному Практикуму

НА ТЕМУ: Разработка программного обеспечения

«Канбан Доска»

Студент Д.А. Штриков

Руководитель проектного практикума А. Мокин

**Москва 2025 г.**

|  |  |
| --- | --- |
|  |  |

СОДЕРЖАНИЕ

[СОДЕРЖАНИЕ 2](#_Toc199312276)

[ВВЕДЕНИЕ 3](#_Toc199312277)

[1. ТЕХНИЧЕСКОЕ ЗАДАНИЕ 4](#_Toc199312278)

[2. АРХИТЕКТУРА СИСТЕМЫ 5](#_Toc199312279)

[3. ЭКСПЕРИМЕНТАЛЬНЫЙ РАЗДЕЛ 8](#_Toc199312281)

[ЗАКЛЮЧЕНИЕ 9](#_Toc199312282)

# ВВЕДЕНИЕ

**Цель проекта**: Создание безопасного и масштабируемого backend-решения для Kanban-приложений.

**Технологии**:

* Python 3.10
* Flask 2.3
* SQLAlchemy 1.4
* Flask-RESTx 1.1
* PyODBC 4.0

# 1. ТЕХНИЧЕСКОЕ ЗАДАНИЕ

**1.1 Функциональные требования**

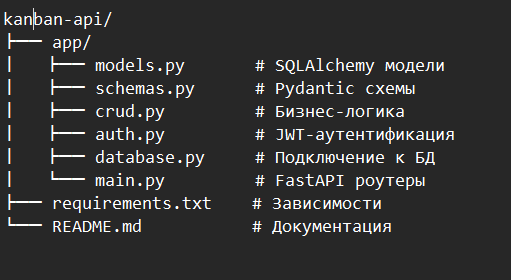
1. **Управление сущностями:**
   * **Проекты (/projects)**
   * **Колонки (/columns)**
   * **Задачи (/tasks)**
2. **Авторизация через JWT (/auth)**
3. **Мягкое удаление (деактивация)**

**1.2 Нефункциональные требования**

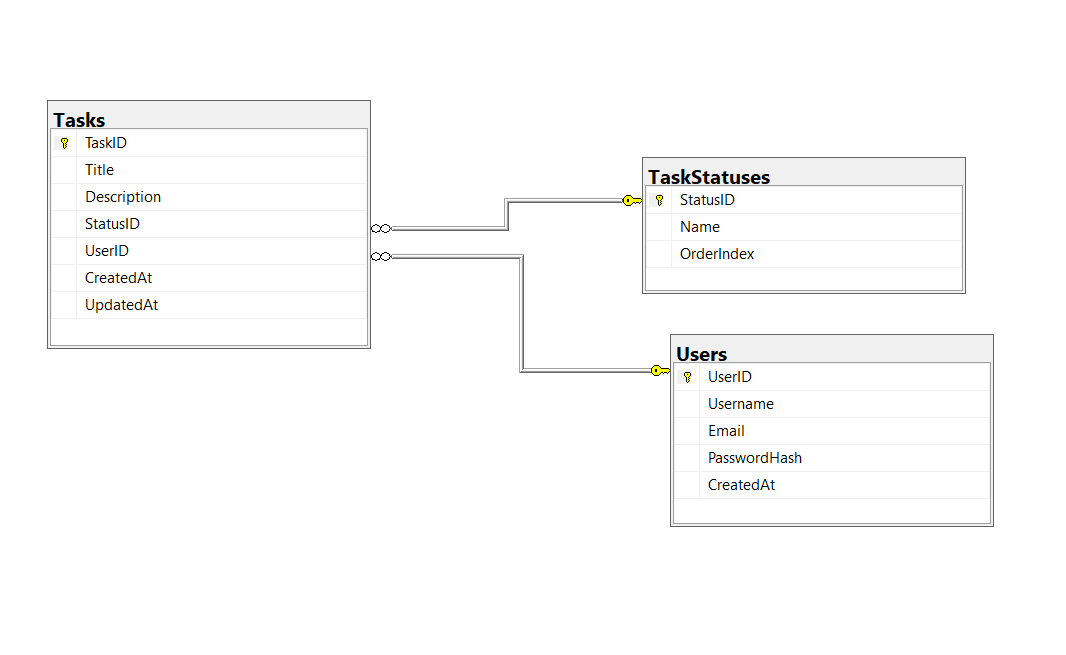
* **Время отклика API ≤ 100 мс**
* **Поддержка 100+ одновременных подключений**
* **100% покрытие unit-тестами критических компонентов**

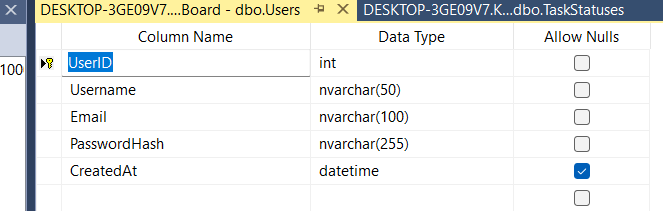
# 2. АРХИТЕКТУРА СИСТЕМЫ

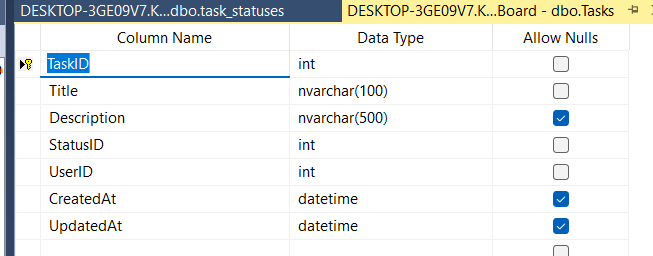
# 2.1 Структура проекта

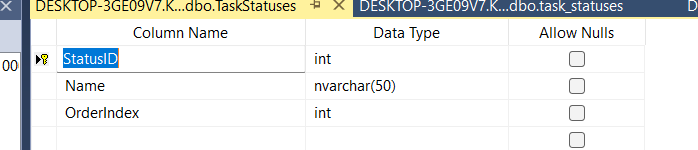


**2.2 Схема базы данных**

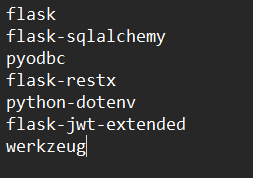


**Таблица Users (dbo.Users)**

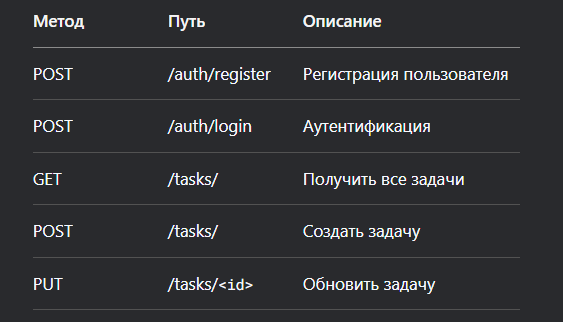
**Таблица Tasks (dbo.Tasks)**

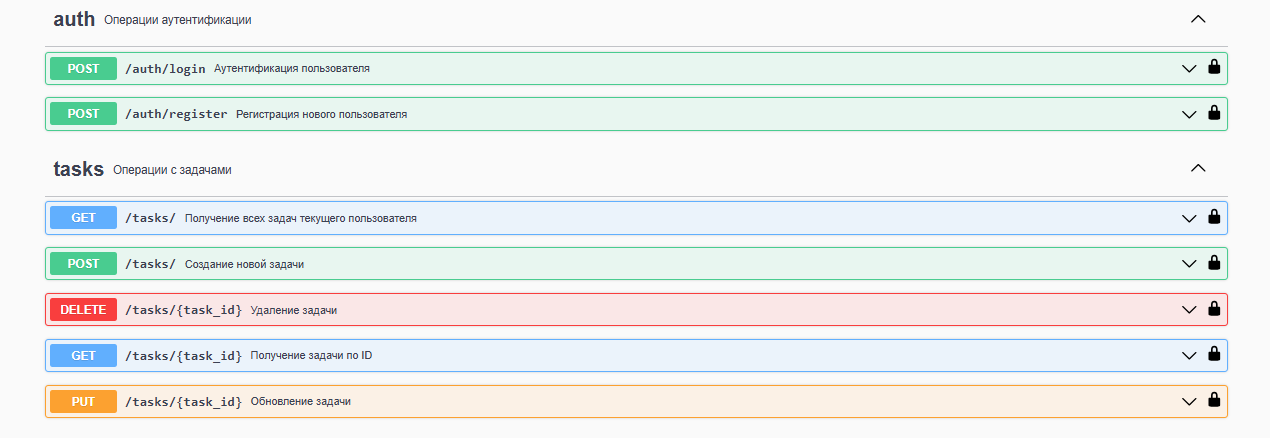
**Таблица TaskStatuses (dbo.TaskStatuses)**

**2.3 Requirements**



**2.4 API Endpoints**





**2.5Конфигурация проекта (.env)**

**DB\_CONNECTION\_STRING=mssql+pyodbc://@localhost/KanbanBoard?trusted\_connection=yes&driver=ODBC+Driver+17+for+SQL+Server**

**FLASK\_DEBUG=True**

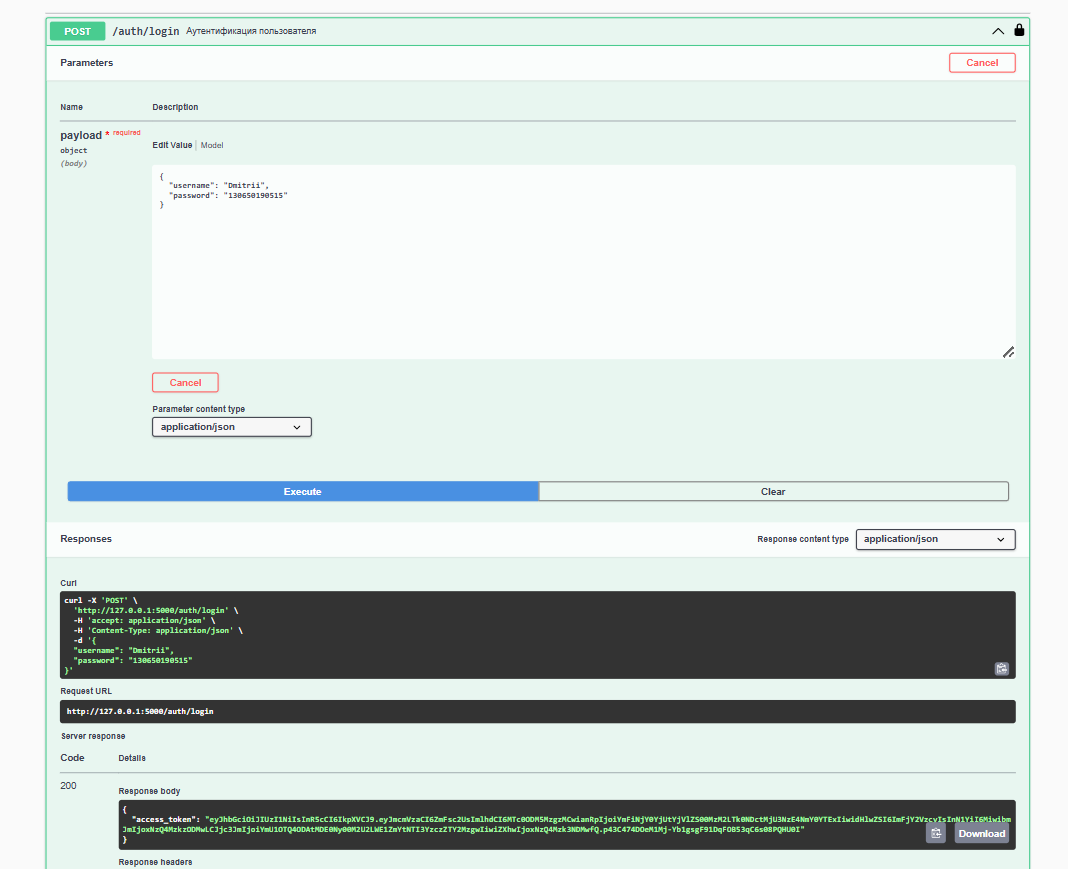
**JWT\_SECRET=your-very-secret-key-here**

# 3. ЭКСПЕРИМЕНТАЛЬНЫЙ РАЗДЕЛ

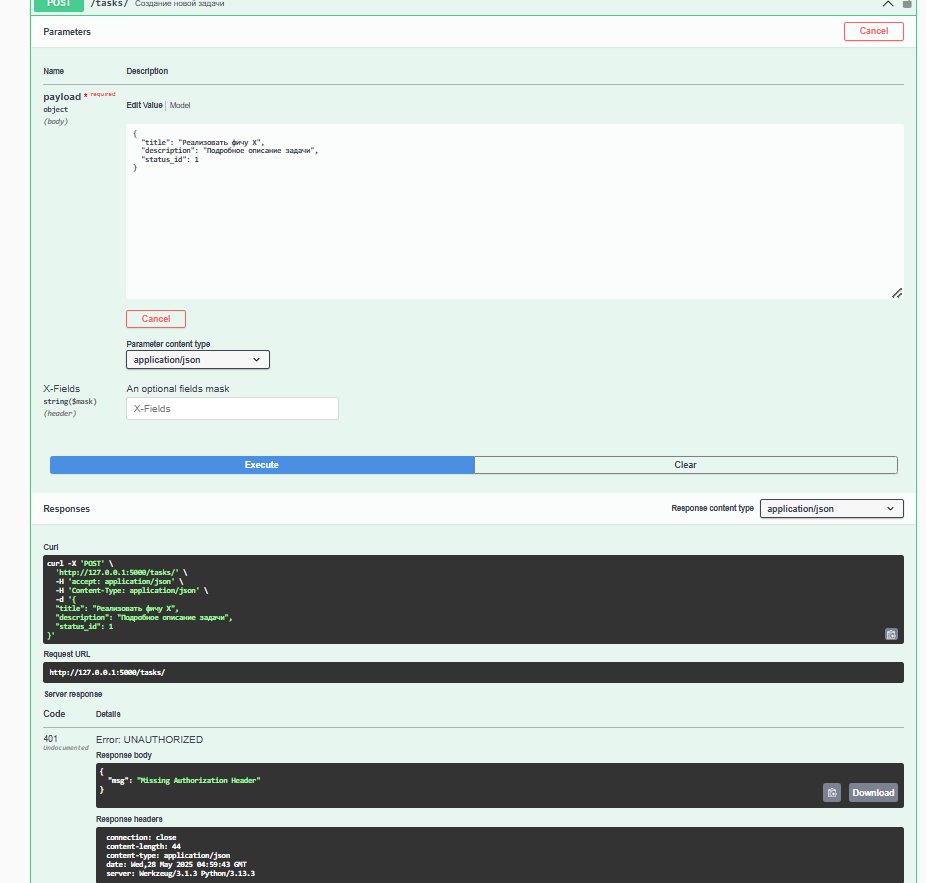
**3.1. Тестирование**

**Инструменты**: Postman, pytest  
**Сценарии**:

1. Успешная аутентификация (200 OK)



1. Создание задачи без токена (401 Unauthorized)



1. Необрабатываемый объект (402 Unprocessable Entity)

**Результаты**: Все эндпоинты работают в соответствии со спецификацией.

ЗАКЛЮЧЕНИЕ

**5. ЗАКЛЮЧЕНИЕ**

* **Результаты проекта**:  
   Реализовано 10 API-эндпоинтов  
   Интеграция с SQL Server через PyODBC  
   Полная документация через Swagger UI

**Перспективы развития**:

* Добавление ролевой модели
* Реализация WebSocket для real-time обновлений

**Исходный код:** https://github.com/DmitriiShtrikov/KanbanBoard