

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Бэк-энд разработка

Отчет

Практическая работа

Выполнил:

Андронов Денис

Группа

К3342

Проверил:

Добряков Д. И.

Санкт-Петербург

2026 г.

Задача

ДЗ2: Проектирование и технический дизайн API

Срок: 18.02.2026

Задание:

1. Опишите все функциональные и нефункциональные требования к вашему бэкенд-приложению
2. Выберите формат реализации API, которого вы будете придерживаться: REST API или Backend For Frontend
3. Спроектируйте все эндпоинты вашего API в формате OpenAPI-схемы с учётом реализованной диаграммы БД
4. Опишите тело каждого запроса и тело каждого ответа, продумайте возможные ошибки, возвращаемые из API

Ход работы

Функциональные требования

1. Приложение для бронирования столов в ресторанах
 - о Вход
 - о Регистрация
 - о Личный кабинет пользователя
 - о Поиск ресторанов с фильтрацией по кухне, расположению и цене
 - о Страница с информацией о ресторане (меню, фото, отзывы)
 - о История бронирований пользователя

Система должна:

1. Позволять регистрацию пользователей
2. Позволять авторизацию
3. Предоставлять личный кабинет
4. Позволять искать рестораны по:

городу

кухне

ценовой категории

5. Позволять просматривать:

меню

фото

отзывы

6. Позволять создавать бронирования
7. Позволять просматривать историю бронирований
8. Позволять оставлять отзывы

Нефункциональные требования:

1. Использование REST API
2. Формат данных - JSON
3. Поддержка PostgreSQL
4. Валидация входных данных
5. Масштабируемость

Создал OpenAPI-схему с помощью typespec. Эндпоинты в swagger:

RestOratorAPI 0.0.0 OAS 3.1

default ^

`POST /auth/login` ▾

`POST /auth/register` ▾

`POST /reservations` ▾

`GET /reservations/user/{userId}` ▾

`GET /restaurants` ▾

`GET /restaurants/{id}` ▾

`POST /reviews` ▾

`GET /reviews/restaurant/{restaurantId}` ▾

`GET /users/{id}` ▾

Вывод

В результате данной работы я спроектировал схему API.