

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Бэк-энд разработка

Отчет

Домашняя работа №3

Выполнил:

Гнеушев Владислав
К3339

Проверил:

Добряков Д. И.

Санкт-Петербург

2025 г.

Задача

1. Реализовать автодокументирование средствами swagger;
2. Реализовать документацию API средствами Postman.

Ход работы

1. Добавлена документация для всех эндпоинтов при помощи библиотеки JSDoc

```
const router = express.Router();
const companyController = new CompanyController();

/**
 * @openapi
 * /api/companies:
 *   get:
 *     tags:
 *       - Companies
 *     summary: Get all companies
 *     responses:
 *       200:
 *         description: A list of companies.
 *         content:
 *           application/json:
 *             schema:
 *               type: array
 *               items:
 *                 $ref: '#/components/schemas/CompanyOutput'
 *       500:
 *         $ref: '#/components/responses/InternalServerError'
 */
router.get('/', async (req: Request, res: Response, next: NextFunction) => {
  try {
    const companies = await companyController.getAll();
```

Рисунок 1 — пример документации для эндпоинта

Для каждого эндпоинта были описаны формат принимаемых и отдаваемых данных (при помощи ссылки *ref* на соответствующий компонент), а также возможные коды ответов.

2. Зарегистрирован эндпоинт для просмотра swagger-документации

```
import swaggerJsdoc from 'swagger-jsdoc';
import swaggerUi from 'swagger-ui-express';
import { Express } from 'express';
import { settings } from './config/settings';

const options = {
  definition: {
    openapi: '3.0.0',
    info: {
      title: 'Headhunter Competitor API',
      version: '1.0.0',
      description: 'API documentation',
      contact: {
        name: 'Gneushev Vladislav',
        email: 'vladgneushev@gmail.com'
      }
    },
    servers: [
      {
        url: `${settings.app.PROTOCOL}://localhost:${settings.app.PORT}`,
        description: 'Development server'
      }
    ],
    externalDocs: {
      description: "swagger.json",
      url: "/swagger.json"
    },
  },
  apis: ['./src/routes/**/*.ts', './src/controllers/**/*.ts'],
};

const swaggerSpec = swaggerJsdoc(options);

export const setupSwagger = (app: Express) => {
  app.use('/docs', swaggerUi.serve, swaggerUi.setup(swaggerSpec));
  app.get('/swagger.json', (req, res) => {
    res.setHeader('Content-Type', 'application/json');
    res.send(swaggerSpec);
  });
  console.log(`Swagger UI available at /docs, generated by swagger-jsdoc`);
};
```

Рисунок 2 — Регистрация swagger в глобальном роутере

Были использованы библиотеки *swagger-jsdoc* и *swagger-ui-express*

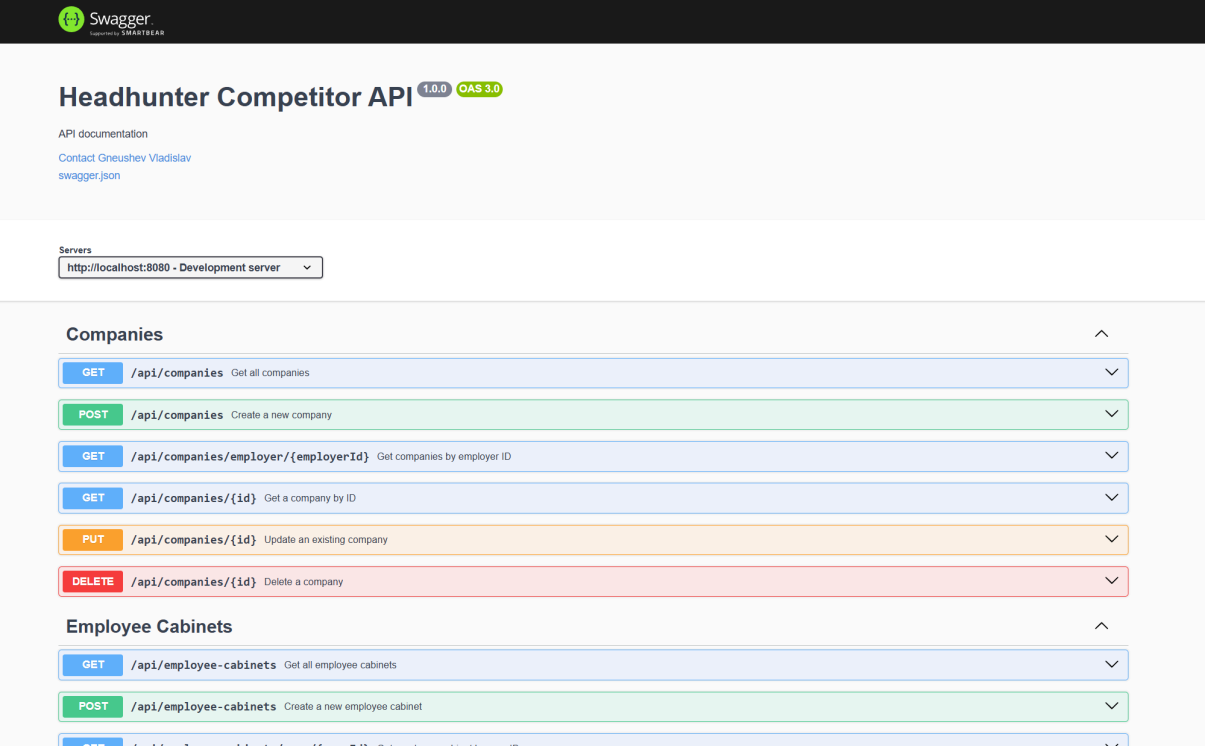


Рисунок 3 — Страница с документацией

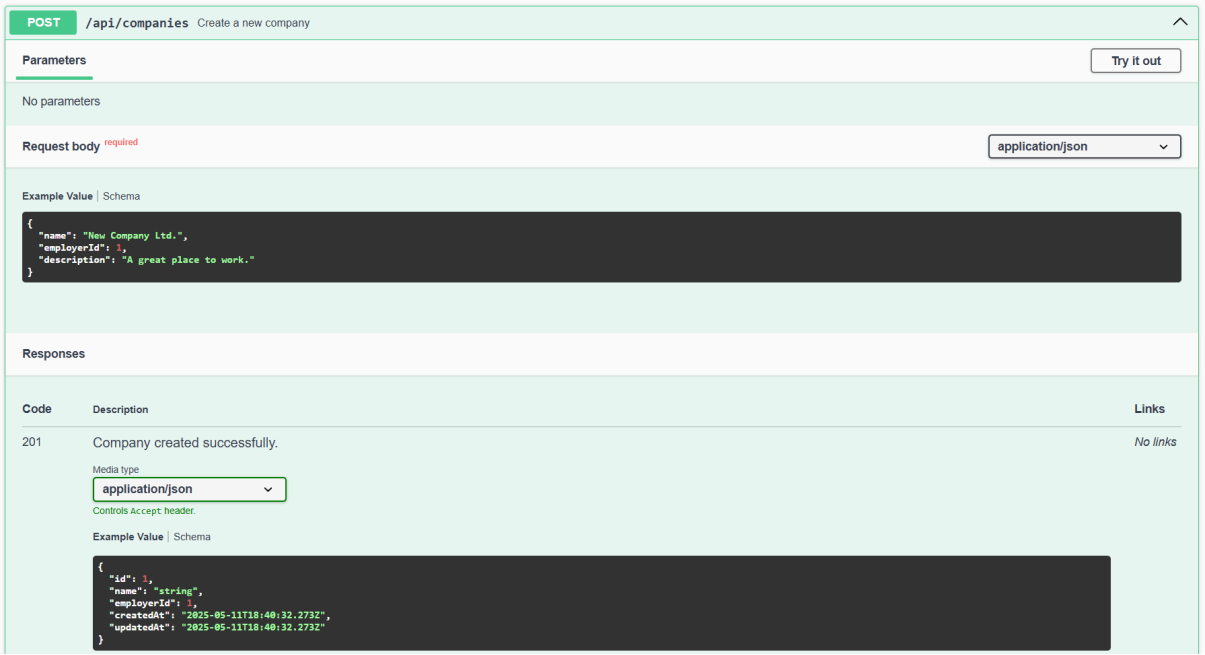


Рисунок 4 — Подробная информация об эндпоинте: принимаемые данные и возможные ответы

3. Реализована документация в Postman

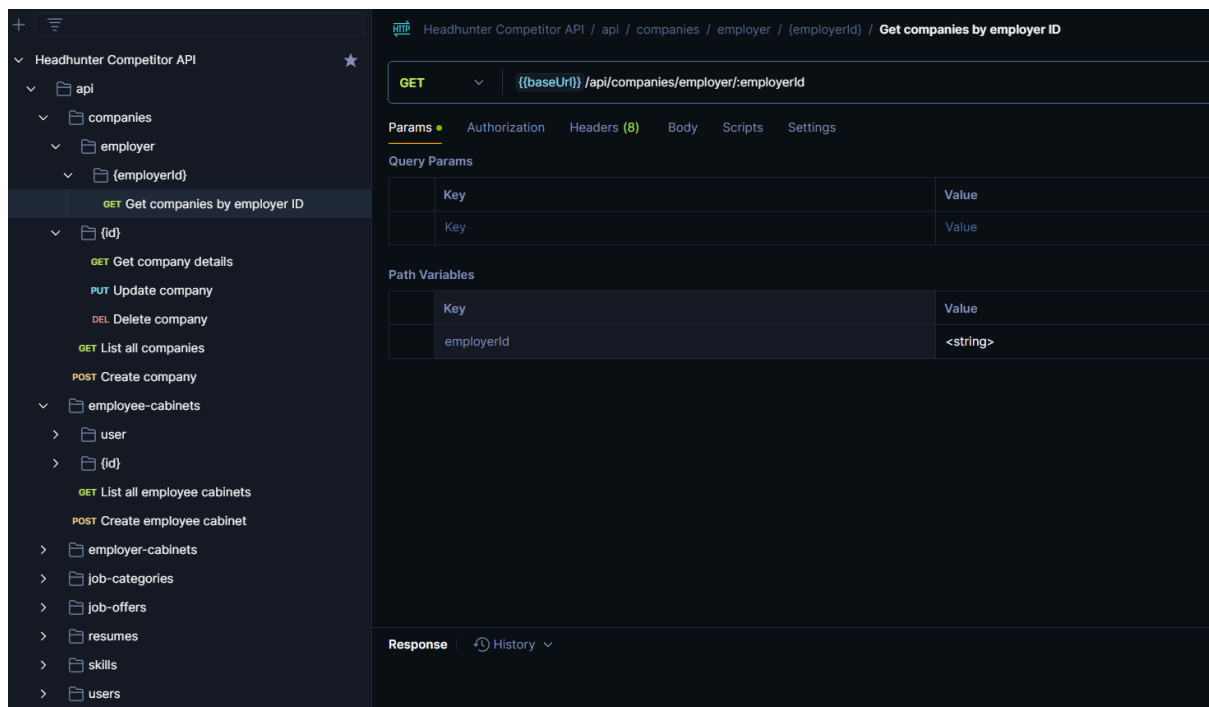


Рисунок 4 — Документация в Postman

Вывод

В данной домашней работе в проект была добавлена автодокументация эндпоинтов при помощи библиотеки JSDoc, а также составлена документация в Postman. При расширении проекта автодокументация позволит описывать эндпоинты прямо в коде при помощи комментариев, без необходимости дополнительно вести документацию при помощи специализированных инструментов.