

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Дисциплина:** Бэкенд разработка

Отчет

Домашнее задание №4

Выполнил:

Фамилия Имя  
Тишалович Леонид

Группа  
К33392

Проверил:  
Добряков Д. И.

Санкт-Петербург

2024 г.

**Задача:** Документирование API из ЛР2 средствами Swagger

**Ход выполнения:**

В ходе выполнения домашнего задания была сделана документация с помощью Swagger.

```
const swaggerOptions = {
  swaggerDefinition: {
    info: {
      title: "Shop App",
      description: "Shop API Info",
      contact: {
        name: "Leonid Tishalovich",
      },
      servers: "http://localhost:5000",
    },
  },
  apis: ["src/routes/*.ts"],
};

const swaggerDocs = swaggerJsDoc(swaggerOptions);
app.use("/api-docs", swaggerUi.serve, swaggerUi.setup(swaggerDocs));
```

Рисунок 1 - Конфигурация документации в app.ts

Ниже приведён пример с добавлением документации к authRoutes.

```

import express from "express";
import { registerValidation } from "../validations/auth";
import UserController from "../controllers/UserControllers";

const router = express.Router();

/**
 * @swagger
 * /auth/login:
 *   post:
 *     summary: Log in a user
 *     tags: [Auth]
 *     requestBody:
 *       required: true
 *       content:
 *         application/json:
 *           schema:
 *             type: object
 *             properties:
 *               email:
 *                 type: string
 *                 example: user@example.com
 *               password:
 *                 type: string
 *                 example: password123
 *     responses:
 *       200:
 *         description: Successfully logged in
 *         content:
 *           application/json:
 *             schema:
 *               type: object
 *               properties:
 *                 token:
 *                   type: string
 *       401:
 *         description: Invalid credentials
 */
router.post("/login", UserController.login);

/**
 * @swagger
 * /auth/register:
 *   post:
 *     summary: Register a new user
 *     tags: [Auth]
 *     requestBody:
 *       required: true
 *       content:
 *         application/json:
 *           schema:
 *             type: object
 *             properties:
 *               name:
 *                 type: string
 *                 example: John Doe
 *               email:
 *                 type: string
 *                 example: john@example.com
 *               password:
 *                 type: string
 *                 example: password123
 *     responses:
 *       201:
 *         description: Successfully registered
 *         content:
 *           application/json:
 *             schema:
 *               type: object
 *               properties:
 *                 id:
 *                   type: integer
 *                 name:
 *                   type: string
 *                 email:
 *                   type: string
 *       400:
 *         description: Validation error
 */
router.post("/register", registerValidation, UserController.register);

export default router;

```

Рисунок 2&3 - Добавление документации к authRoutes

Ниже приведена документация сделанная с помощью Swagger

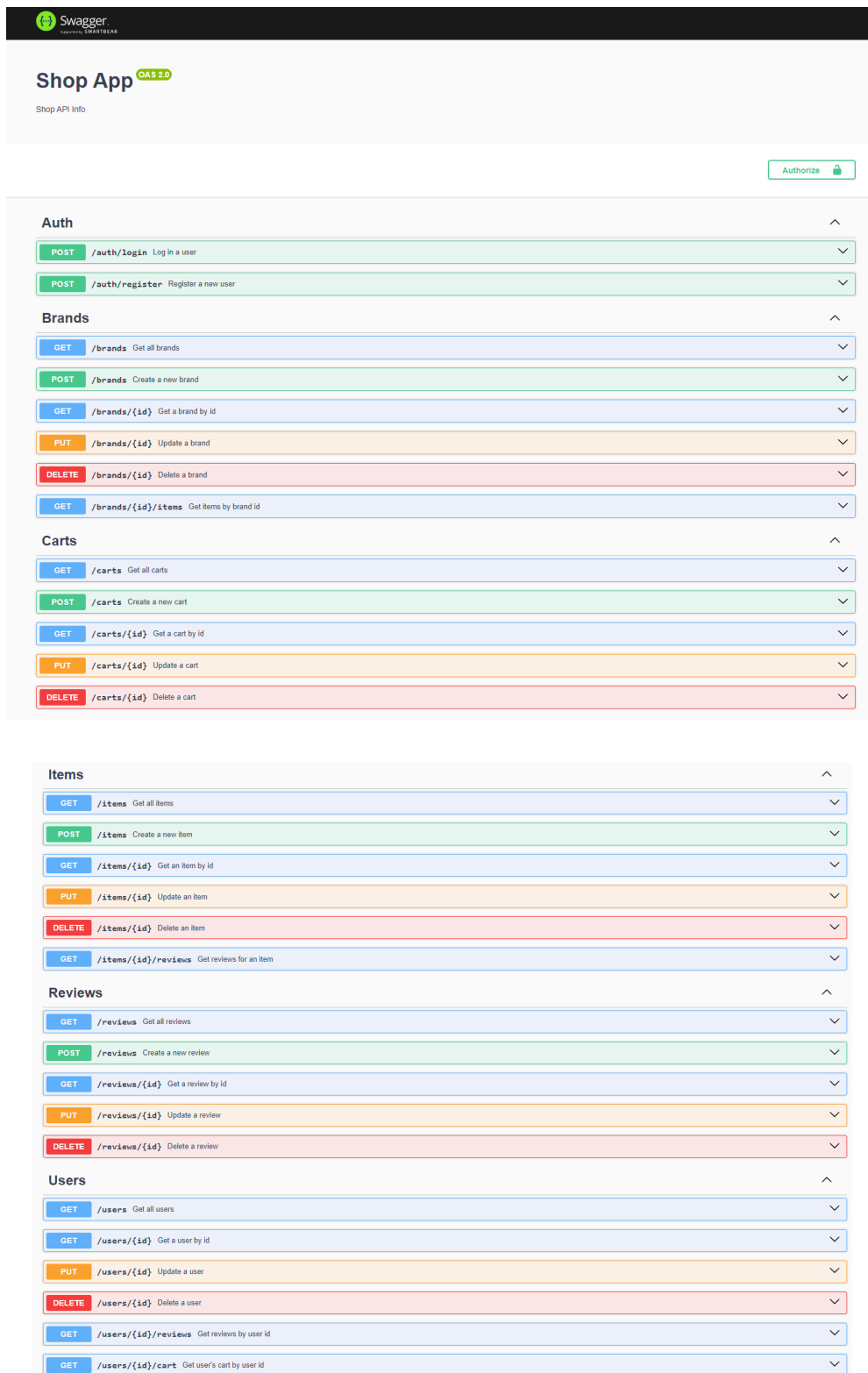


Рисунок 3&4 - Документация в Swagger

**Вывод:** в ходе выполнения данного домашнего задания были получены навыки создания документации средствами Swagger