

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Бэк-энд разработка

Отчет

Домашние работы 3,4

Документирование API средствами swagger и Postman;

Тестирование API средствами Postman

Выполнили:

**Жижилева Арина
Строганова Елизавета**

K3342

**Проверил:
Добряков Д. И.**

Санкт-Петербург

2025 г.

СОДЕРЖАНИЕ

Домашнее задание 3.....	3
Домашнее задание 4.....	6

Домашнее задание 3

Задание:

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

Ход работы:

Создали файл `swagger.ts`



```
1 import swaggerJsdoc from "swagger-jsdoc";
2
3 const options = {
4   definition: {
5     openapi: "3.0.0",
6     info: {
7       title: "Fitness App API",
8       version: "1.0.0",
9       description: "API documentation for the Fitness App backend",
10    },
11    servers: [
12      {
13        url: "http://localhost:3000",
14        description: "Development server",
15      },
16    ],
17  },
18  apis: ["./src/controllers/*.ts", "./src/routes/*.ts"], // Путь к файлам с JSDoc
19 };
20
21 const swaggerSpec : object = swaggerJsdoc(options);
22
23 export default swaggerSpec;
```

В `app.ts` прописали использование `swagger`

```
1  > import ...
13
14  const app : Express = express();
15  app.use(express.json());
16
17  // Swagger UI
18  app.use("/api-docs", swaggerUi.serve, swaggerUi.setup(swaggerSpec));
19
20  // Роуты
21  app.use("/users", userRoutes);
22  app.use("/posts", blogPostRoutes);
23  app.use("/measurements", userMeasurementsProgressRoutes);
24  app.use("/plans", workoutPlanRoutes);
25  app.use("/workouts", workoutRoutes);
26  app.use("/workout-progress", userWorkoutProgressRoutes);
27  app.use("/like", postLikeRoutes);
28  app.use("/comment", postCommentRoutes);
29  app.use("/tag", postTagRoutes);
30
31  export default app;
```

Package.json

```
20    "typeorm": "^0.3.22",
21    "swagger-ui-express": "^5.0.1",
22    "swagger-jsdoc": "^6.2.8"
23  },
24  "devDependencies": {
25    "@types/express": "^5.0.1",
26    "@types/node": "^22.15.3",
27    "@types/swagger-ui-express": "^4.1.6",
28    "@types/swagger-jsdoc": "^6.0.4",
29    "ts-node-dev": "^2.0.0",
30    "typescript": "^5.8.3"
31  }
32 }
```

Заходим на локалхост:

Fitness App API

1.0.0 OAS 3.0

API documentation for the Fitness App backend

Servers

http://localhost:3000 - Development server

BlogPosts

POST	/posts	Create a new blog post	⌵
GET	/posts	Get all blog posts	⌵
GET	/posts/{id}	Get a blog post by ID	⌵
PUT	/posts/{id}	Update a blog post by ID	⌵
DELETE	/posts/{id}	Delete a blog post by ID	⌵

PostComments

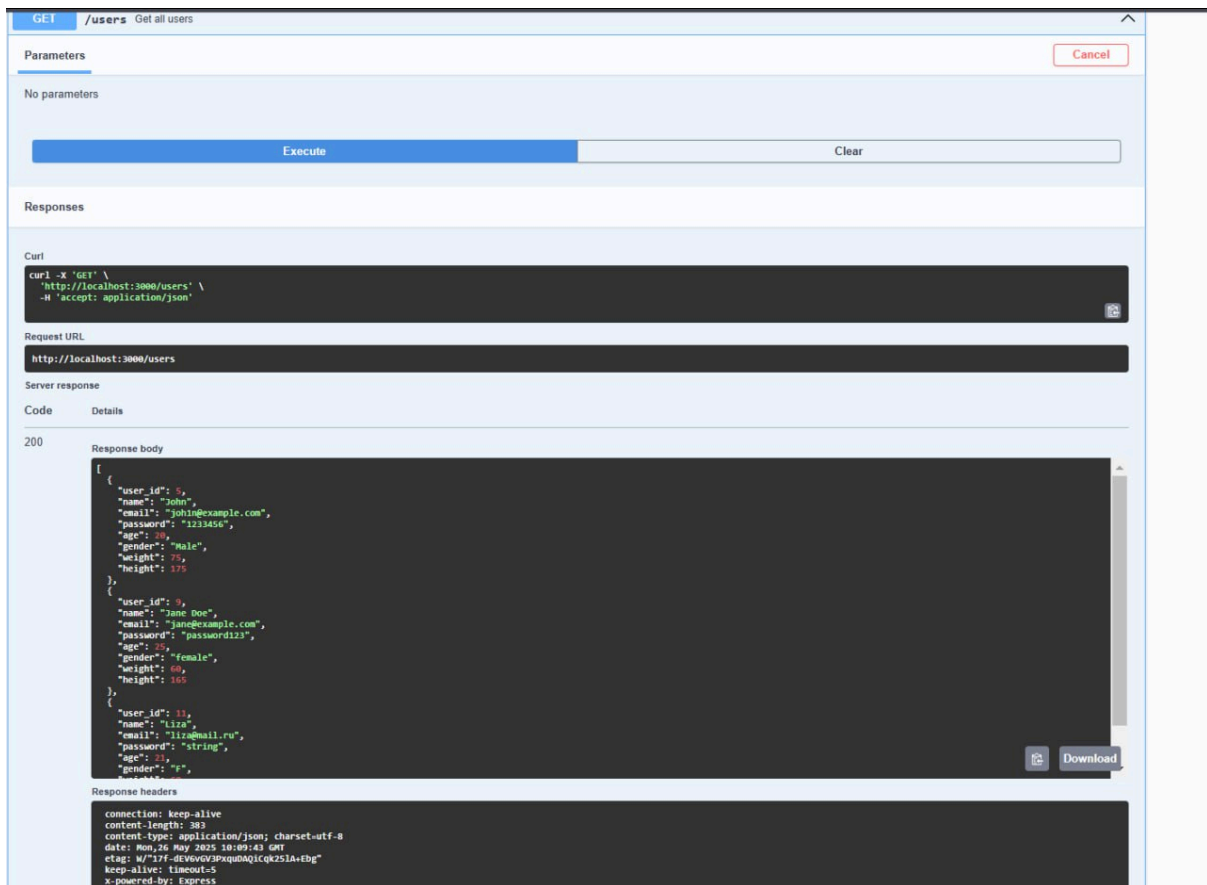
--	--	--	--

Workouts

POST	/workouts	Create a new workout	⌵
GET	/workouts	Get all workouts	⌵
GET	/workouts/{id}	Get a workout by ID	⌵
PUT	/workouts/{id}	Update a workout by ID	⌵
DELETE	/workouts/{id}	Delete a workout by ID	⌵

WorkoutPlans

POST	/plans	Create a new workout plan	⌵
GET	/plans	Get all workout plans	⌵
GET	/plans/{id}	Get a workout plan by ID	⌵
PUT	/plans/{id}	Update a workout plan by ID	⌵
DELETE	/plans/{id}	Delete a workout plan by ID	⌵



Документация Postman:

GET Get All Users

[Open request →](#)

`http://localhost:3000/users`

Retrieve a list of all users.

Example

Success ▾

Request

cURL



```
curl --location 'http://localhost:3000/users'
```

Response

Body

Headers (0)

200 OK

json



[

Домашнее задание 4

Задание:

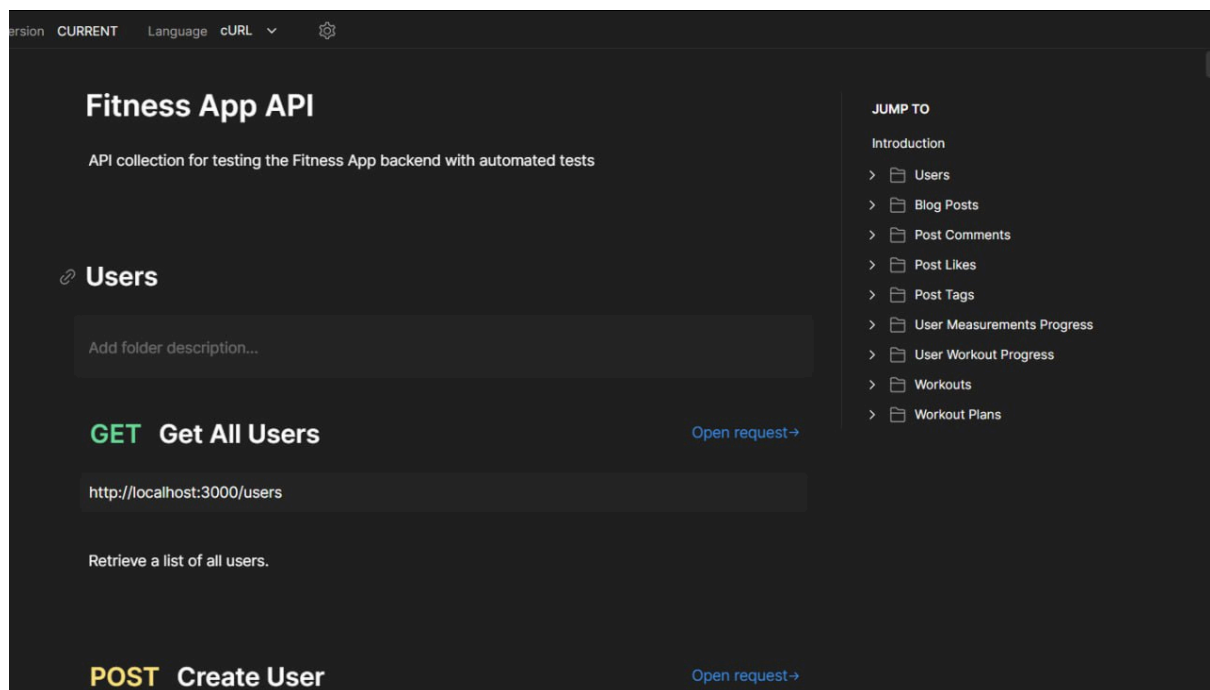
1. реализовать тестирование API средствами Postman;
2. написать тесты внутри Postman.

Ход работы:

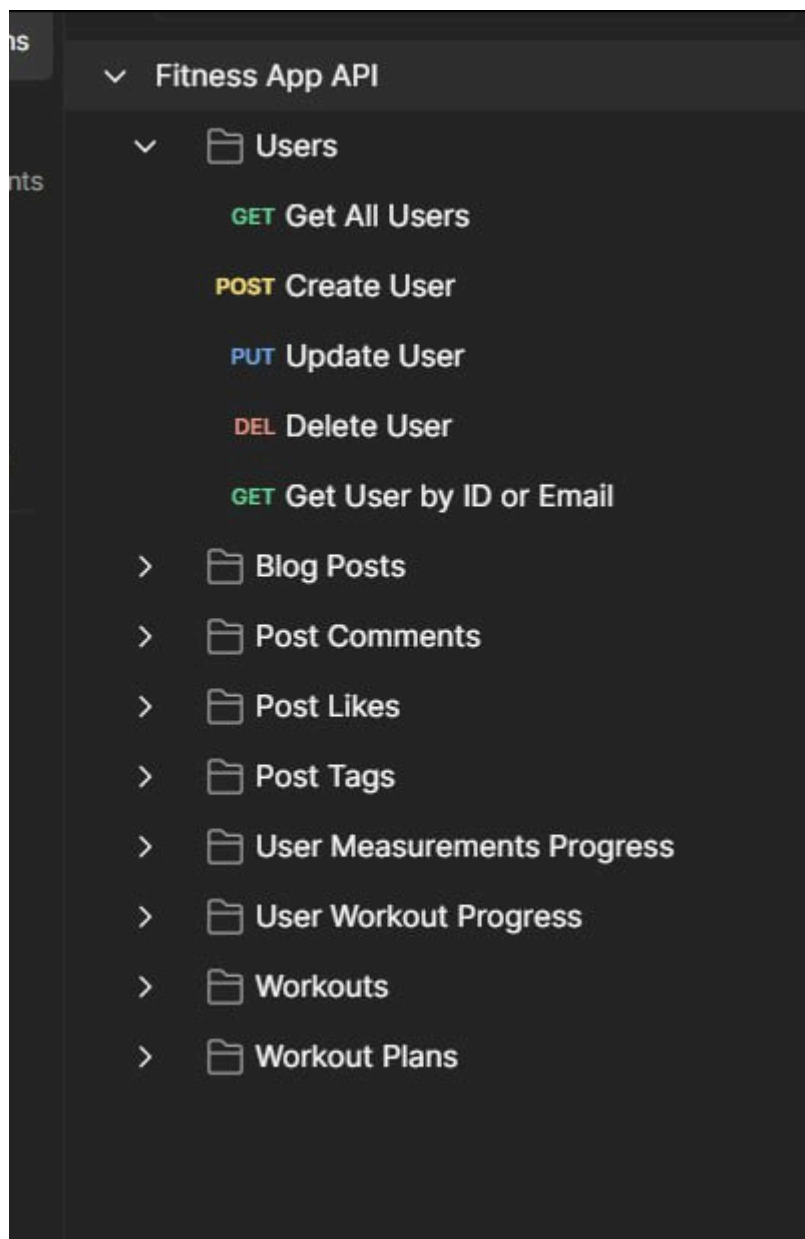
Ссылка на тесты в Postman:

<https://elizavetastroganova.postman.co/workspace/Elizaveta-Stroganova's-Workspace~f60d26e4-c440-4d67-8e7e-bfe5fed1f9fc/collection/44523262-8d6a65da-8f48-4954-8dae-0d098b8256c5/overview?action=share&creator=44523262&active-environment=44523262-88215e28-c7ed-49ab-a05e-10724940fdd3>

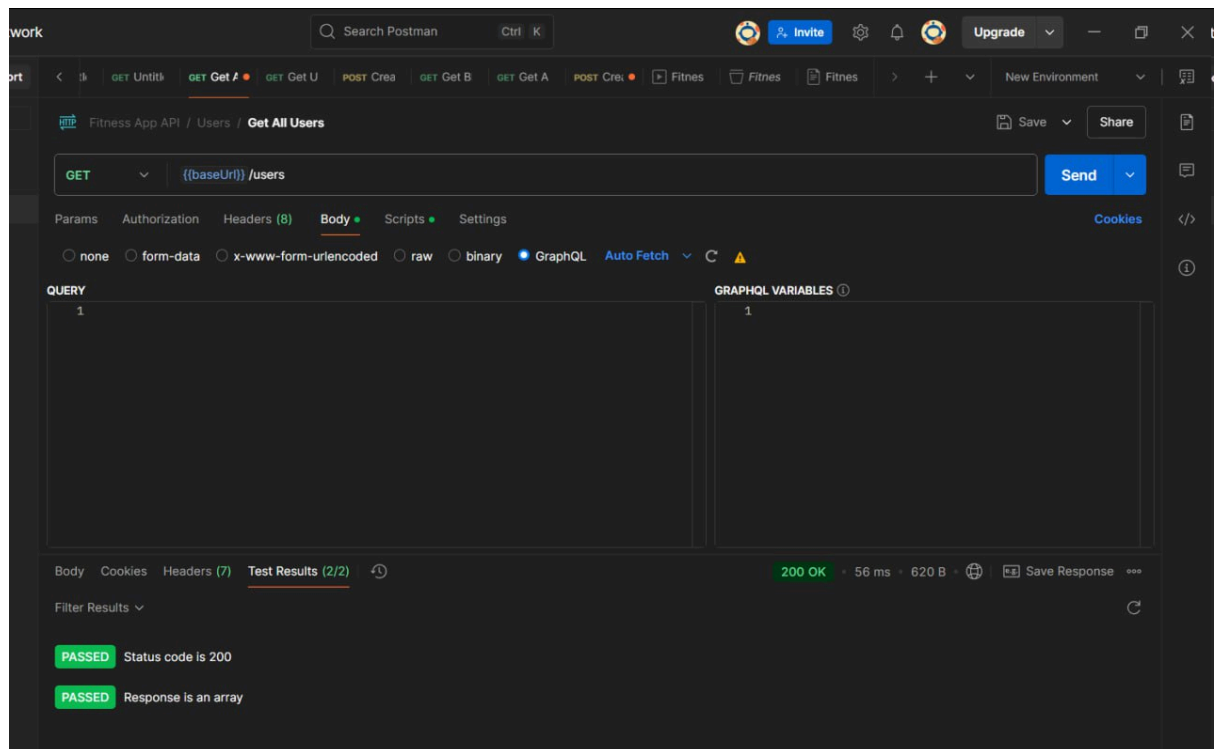
Главная:



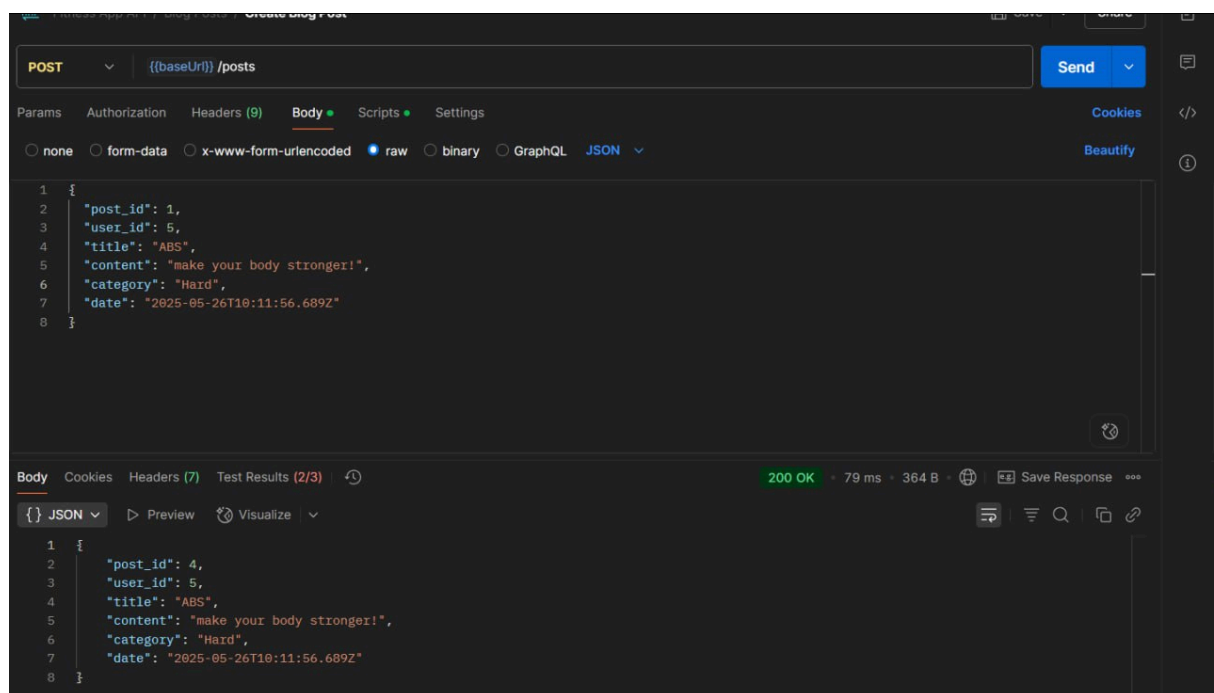
Содержание:



Скрипт get_users с тестами



Скрипт пост блогпост с тестами



Fitness App API / Blog Posts / Create Blog Post

SaveShare

POST{{baseUrl}} /postsSend

ParamsAuthorizationHeaders (9)BodyScriptsSettings

noneform-datax-www-form-urlencodedrawbinaryGraphQLJSON

Beautify

```
1 {
2   "post_id": 1,
3   "user_id": 5,
4   "title": "ABS",
5   "content": "make your body stronger!",
6   "category": "Hard",
7   "date": "2025-05-26T10:11:56.689Z"
8 }
```

BodyCookiesHeaders (7)Test Results (2/3)

200 OK79 ms364 BSave Response

Filter Results

PASSED

Status code is 200

PASSED

Response has post_id