МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
 ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНО УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«Санкт-Петербургский национальный исследовательский университет  
информационных технологий, механики и оптики»

Факультет информационных технологий и программирования

Кафедра информационных систем

Лабораторная работа № 4

Создание контроллеров страниц и спецификации

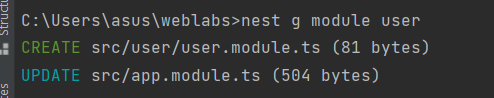
**Выполнил студент группы № М33081:**   
Куксенко Кристина

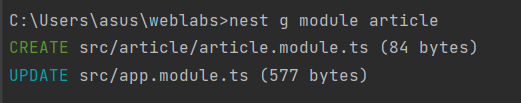
**Проверила:**

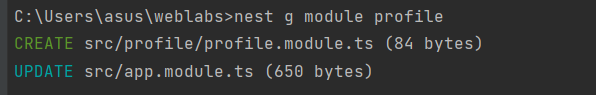
Приискалов Роман Андреевич

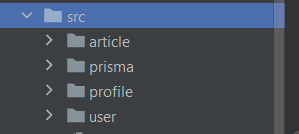
САНКТ-ПЕТЕРБУРГ  
2022

Используем следующие команды, чтобы сгенерировать модули:

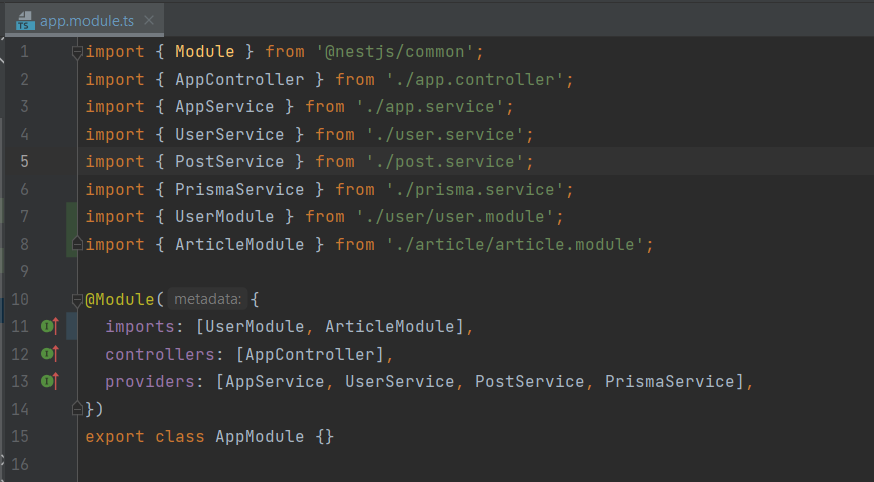




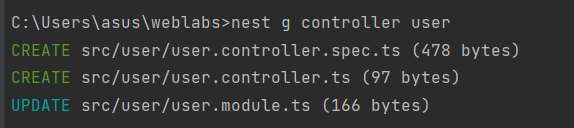


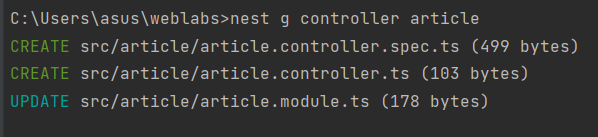


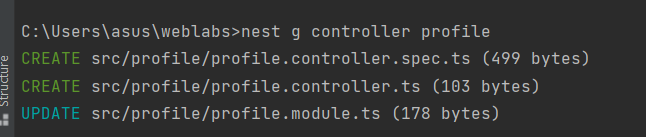
Также, они добавились в app.module.ts:



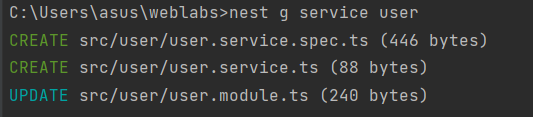
Генерируем для модулей контроллеры:

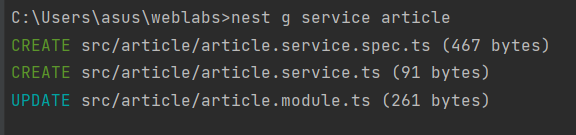


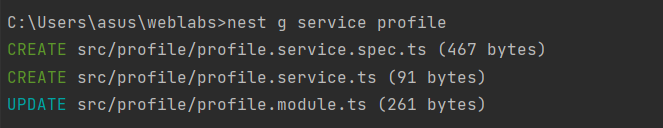




И сервисы:





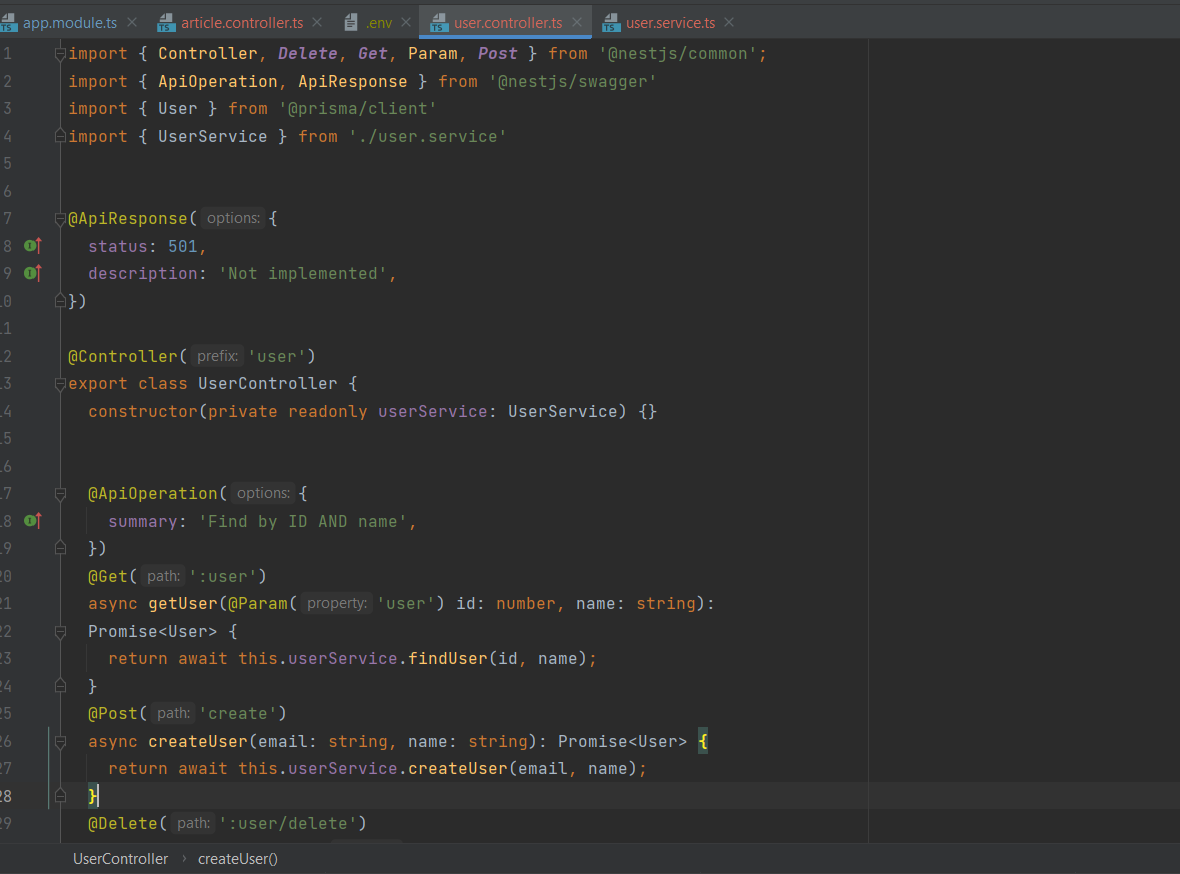


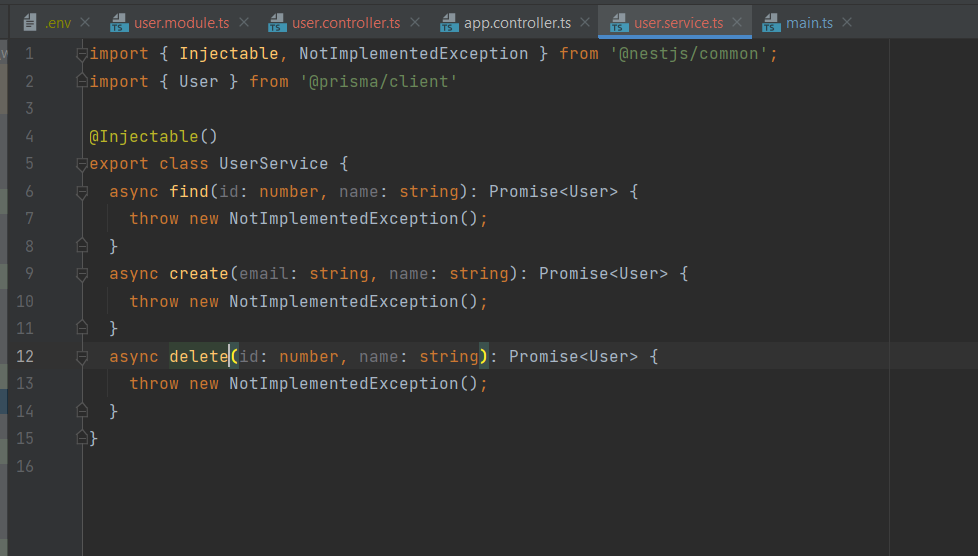
Чтобы настроить автоматическую генерацию OpenAPI спецификации, устанавливаем модуль swagger:

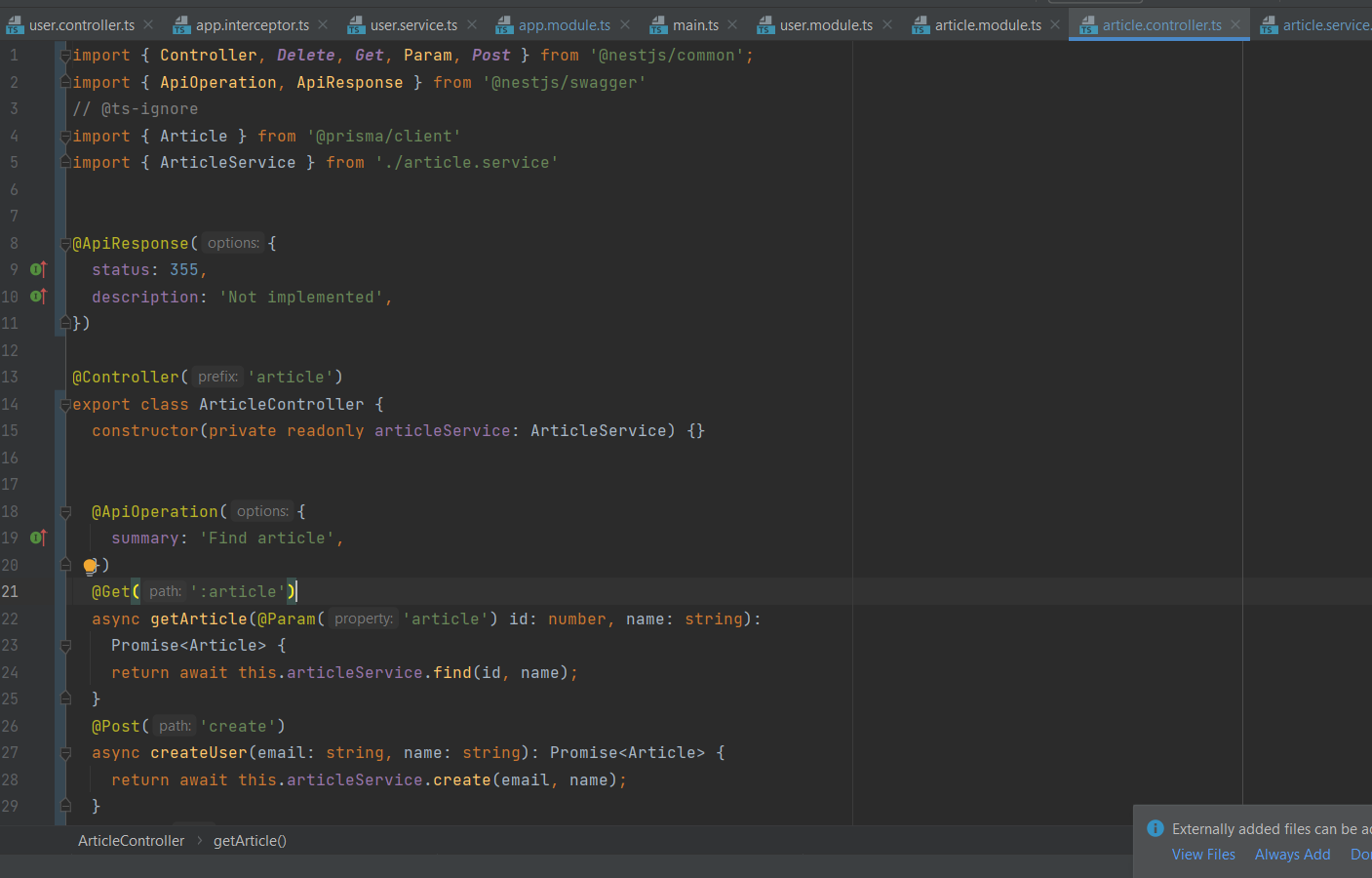


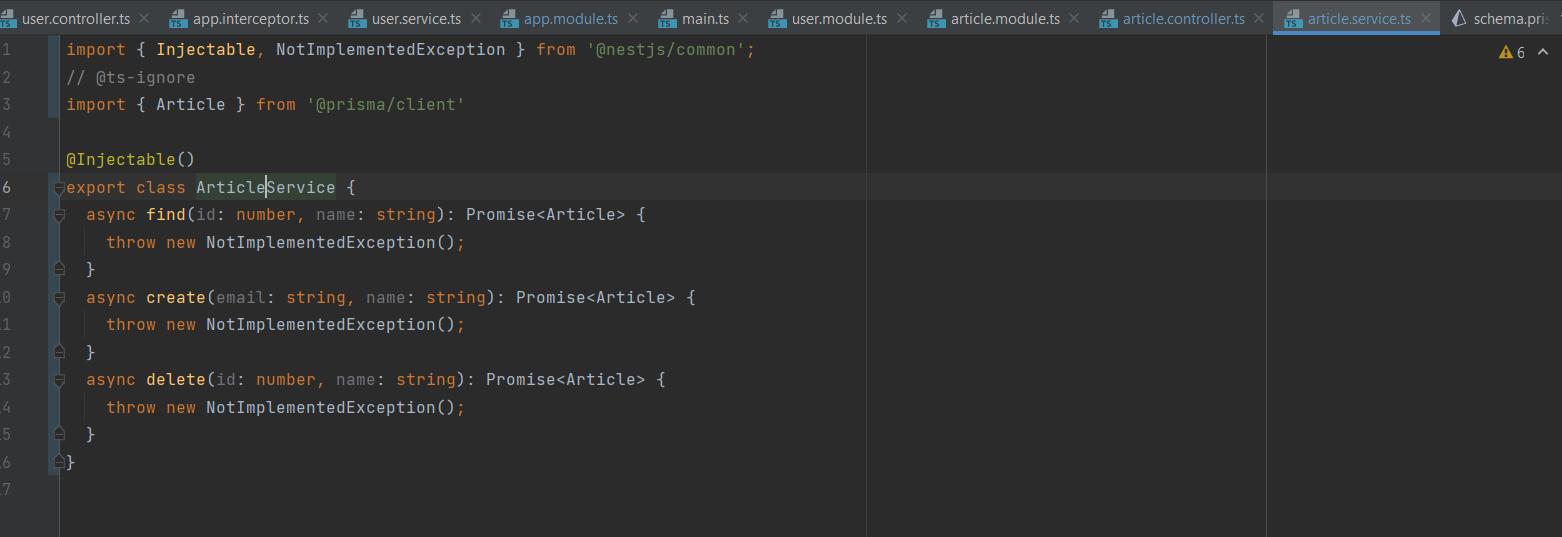
Исключения типа NotImplementedException будут обработаны Nest’ом  
автоматически и данный метод отработает согласно RFC и вернёт ответ HTTP  
501 Not Implemented.

Прописываем в файлах контроллера и сервиса для каждой модели операции, которые можно будет с ними выполнять:

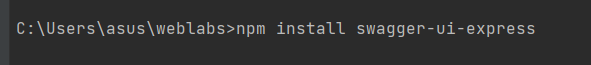








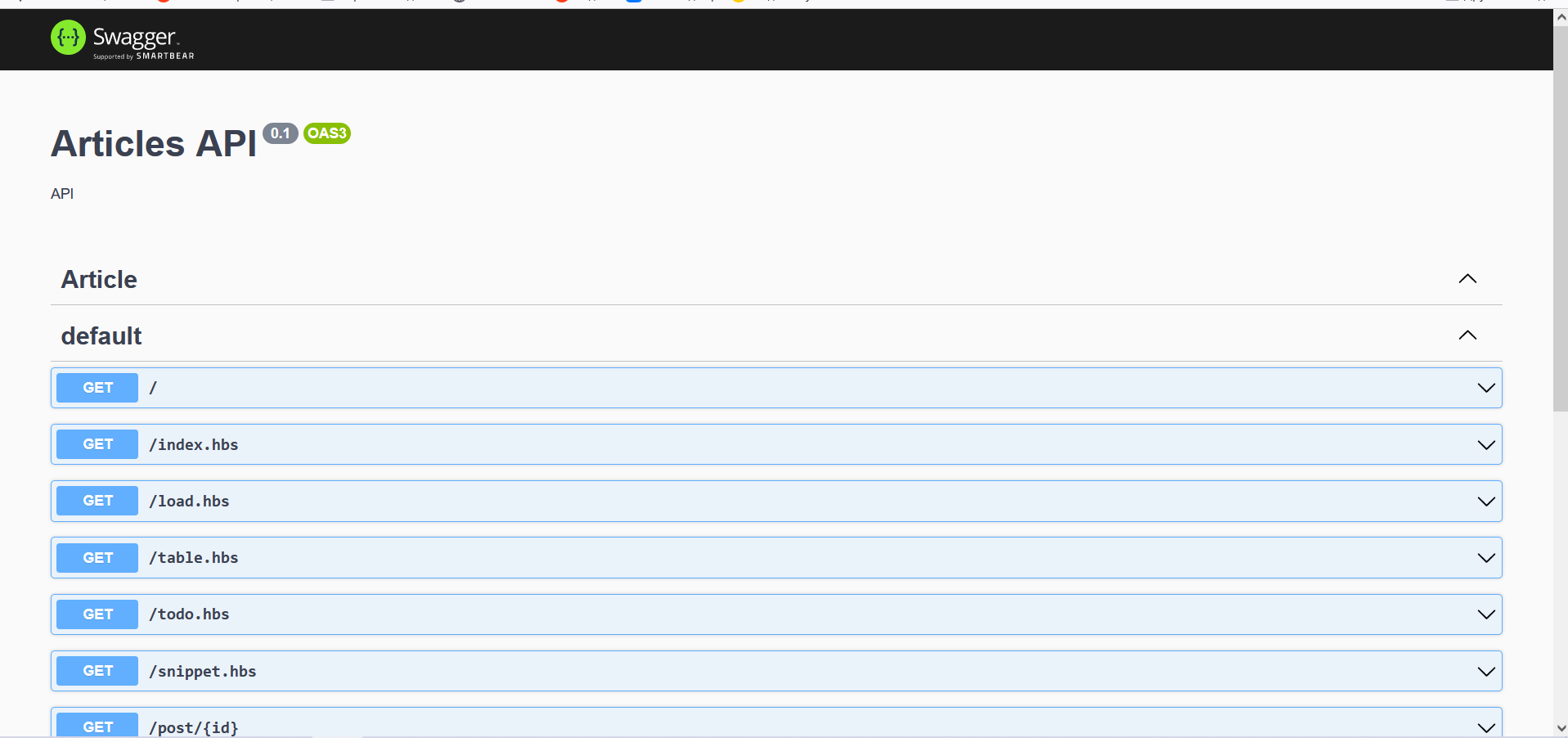
Теперь устанавливаем swagger-ui:

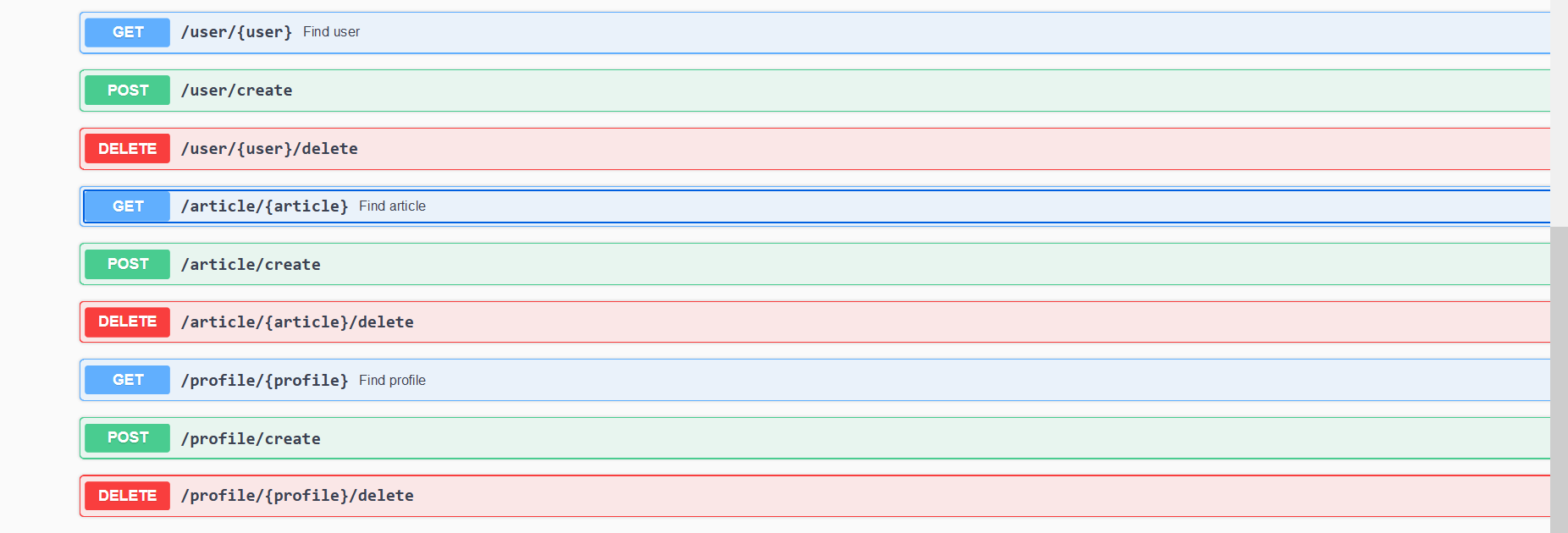


Заходим на наш сайтик, в конце дописываем /api и перезагружаем:

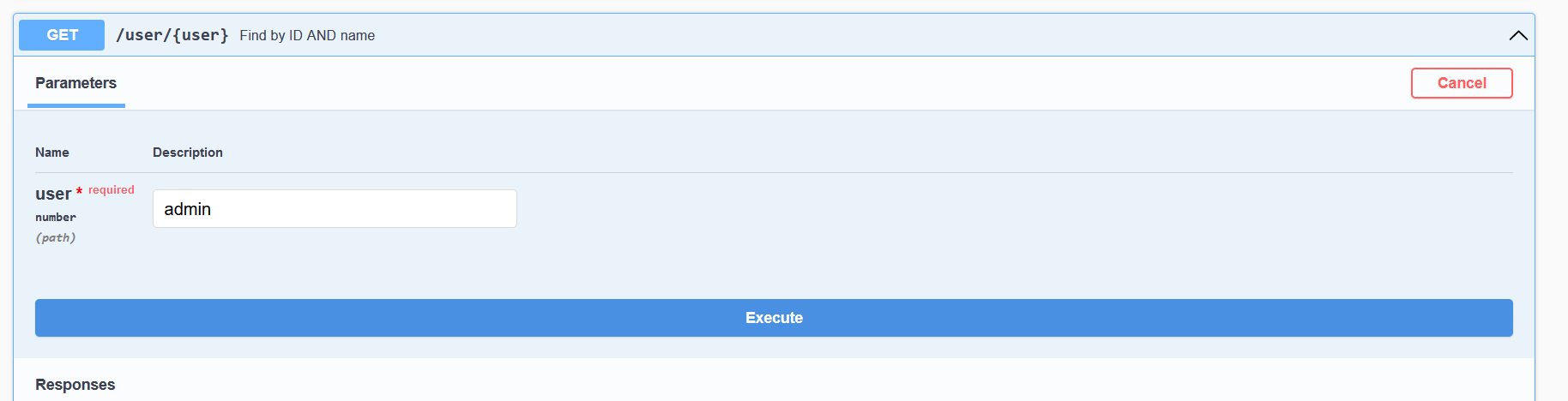


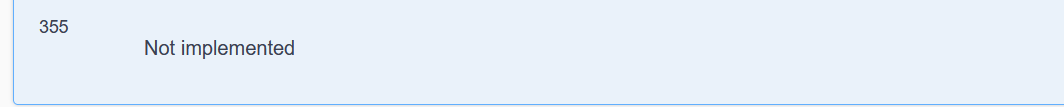
Вот что у нас получилось:





Пример метода получения профиля пользователя:



Такой код, потому что я сама установила такое число в коде:  
  
  


Пример метода поста профиля пользователя:

