

Практика 13: REST-сервис

Юрий Литвинов

y.litvinov@spbu.ru

1. Задание на практику

В командах по два человека спроектировать и реализовать REST-сервис для «открытой части» магазина книг, спроектированного двумя неделями ранее¹. Проектируем пока только «открытую» часть сервиса, не требующую аутентификации и, соответственно, не изменяющую состояния сервиса (так что требуется реализовать только GET-запросы).

Должен быть поддержан запрос информации о книгах, включая отзывы (пользовательские и редакторские) и поиск по книгам. Представьте, что этим API будут пользоваться сторонние сервисы (например, чужие магазины) или клиент, показывающий пользователю интересующие его новинки.

При этом, для простоты, слой доступа к данным должен возвращать захадркоженные данные, работать с реальной СУБД не надо. Бизнес-логика должна быть честно реализована, то есть поиск по критериям должен работать «честно», хоть и вручную (без SQL и подобных штук).

Для тестирования сервиса используйте Swagger — при этом сервис должен предоставлять метаданные о себе в формате OpenAPI. Также GET-запросы должны работать и просто из адресной строки браузера.

В конце практики выложите код в репозиторий — этот сервис нам потом ещё потребуется.

Язык реализации и используемая библиотека не важны.

На что ещё стоит обратить внимание:

- на аккуратное проектирование доменных классов — книг, отзывов разных видов, поисковых запросов (паттерн «Спецификация?»);
- на аккуратный дизайн API в соответствии с принципами REST — только коллекции и ресурсы, никаких методов типа `get_book_review`;
- кстати, аккуратный REST API подразумевает фильтрацию на стороне сервера — самое место для функциональности поиска;
- помимо доменных классов, вся «обвязка» должна быть по канонам гексагональной архитектуры — доменная модель отдельно, средства доставки отдельно; в идеале ещё доменные сервисы (для поиска, например) и даже элементы чистой архитектуры, хотя для этой задачи чистая архитектура может быть овердизайном;

¹ Напоминаем, ТЗ: <https://goo.gl/94LyFc>.

- если успеете, нужна также пагинация (чтобы возвращался не весь каталог, а порциями по N книг) и HATEOAS (то есть в каждом ответе также URL с тем, что ещё можно сделать с сервисом).