

НИУ ИТМО

Факультет программной инженерии и компьютерных технологий

Отчет по лабораторной работе №1

по дисциплине Сервис-ориентированная архитектура

Студент группы № Р34151

Шипулин Павел Андреевич

Студент группы № Р34151

Горинов Даниил Андреевич

Преподаватель

Цопа Евгений Алексеевич

Санкт-Петербург

2024

Задание (вариант 5190)

Разработать спецификацию в формате OpenAPI для набора веб-сервисов, реализующего следующую функциональность:

Первый веб-сервис должен осуществлять управление коллекцией объектов. В коллекции необходимо хранить объекты класса **City**, описание которого приведено ниже:

```
public class City {
    private Integer id; //Поле не может быть null, Значение поля должно быть
    больше 0, Значение этого поля должно быть уникальным, Значение этого поля
    должно генерироваться автоматически
    private String name; //Поле не может быть null, Строка не может быть пу-
    стой
    private Coordinates coordinates; //Поле не может быть null
    private java.time.ZonedDateTime creationDate; //Поле не может быть null,
    Значение этого поля должно генерироваться автоматически
    private Double area; //Значение поля должно быть больше 0, Поле не может
    быть null
    private Long population; //Значение поля должно быть больше 0, Поле не
    может быть null
    private long metersAboveSeaLevel;
    private java.util.Date establishmentDate;
    private int telephoneCode; //Значение поля должно быть больше 0, Макси-
    мальное значение поля: 100000
    private Climate climate; //Поле может быть null
    private Human governor; //Поле может быть null
}
public class Coordinates {
    private Long x; //Поле не может быть null
    private Float y; //Поле не может быть null
}
public class Human {
    private String name; //Поле не может быть null, Строка не может быть пу-
    стой
    private int age; //Значение поля должно быть больше 0
    private int height; //Значение поля должно быть больше 0
}
public enum Climate {
    TROPICAL_SAVANNA,
    HUMIDCONTINENTAL,
    SUBARCTIC,
    POLAR_ICECAP,
    DESERT;
}
```

Веб-сервис должен удовлетворять следующим требованиям:

- API, реализуемый сервисом, должен соответствовать рекомендациям подхода RESTful.
- Необходимо реализовать следующий базовый набор операций с объектами коллекции: добавление нового элемента, получение элемента по ИД, обновление элемента, удаление элемента, получение массива элементов.
- Операция, выполняемая над объектом коллекции, должна определяться методом HTTP-запроса.
- Операция получения массива элементов должна поддерживать возможность сортировки и фильтрации по любой комбинации полей класса, а также возможность страничного вывода результатов выборки с указанием размера и порядкового номера выводимой страницы.
- Все параметры, необходимые для выполнения операции, должны передаваться в URL запроса.
- Информация об объектах коллекции должна передаваться в формате **xml**.
- В случае передачи сервису данных, нарушающих заданные на уровне класса ограничения целостности, сервис должен возвращать код ответа http, соответствующий произошедшей ошибке.

Помимо базового набора, веб-сервис должен поддерживать следующие операции над объектами коллекции:

- Удалить все объекты, значение поля `governor` которого эквивалентно заданному.
- Сгруппировать объекты по значению поля `id`, вернуть количество элементов в каждой группе.

- Вернуть количество объектов, значение поля `climate` которых больше заданного.

Эти операции должны размещаться на отдельных URL.

Второй веб-сервис должен располагаться на URL `/genocide`, и реализовывать ряд дополнительных операций, связанных с вызовом API первого сервиса:

- `/count/{id1}/{id2}/{id3}` : рассчитать суммарное население трёх городов с заданными `id`
- `/move-to-poorest/{id}` : переселить всё население города с заданным `id` в город с наименьшим уровнем жизни

Эти операции также должны размещаться на отдельных URL.

Для разработанной спецификации необходимо сгенерировать интерактивную веб-документацию с помощью Swagger UI. Документация должна содержать описание всех REST API обоих сервисов с текстовым описанием функциональности каждой операции. Созданную веб-документацию необходимо развернуть на сервере `helios`.

Исходный код

https://github.com/PashcalE2/SOA/tree/main/SOA_lab1

Вывод

В ходе выполнения лабораторной работы была разработана спецификация OpenAPI с использованием двух веб-сервисов.