

# Техническое задание на REST API сервис

Создать REST API сервис, который будет принимать данные от метеорологического датчика (сенсор). Сенсор измеряет температуру воздуха и определяет, идет дождь или нет. Каждый раз, когда он будет производить измерение, он будет слать HTTP запрос с данными в формате JSON на наш сервер.

## ТЗ к REST API

Спроектировать и реализовать REST API используя стек Hibernate/Spring/SpringMVC.

### Методы API:

- 1) Добавление сенсора - регистрирует новый сенсор в системе. Другими словами, просто добавляет новый сенсор в таблицу сенсоров в БД. У сенсоров есть только одно поле - название. Вы должны помнить о правилах хорошего кода и использовать DTO для входящего объекта - сенсора. Также, вы должны валидировать то, что сенсора с таким названием еще нет в БД. Если сенсор с таким названием есть в БД - возвращать клиенту сообщение с ошибкой. Также, если название сенсора пустое или содержит менее 3 или более 30 символов, клиенту должно возвращаться сообщение с ошибкой.
- 2) Получение сенсоров — возвращает все сенсоры из БД.
- 3) Получение сенсора — возвращает сенсор из БД по идентификатору.
- 4) Добавление измерения - Добавляет новое измерение. Вещественное поле "value" содержит значение температуры воздуха, булево поле "raining" содержит значение true/false в зависимости от того, зарегистрировал ли сенсор дождь или нет. Помимо этого, в этом запросе передается сам объект сенсора, который получил и отправляет эти "измерения". Значения температуры воздуха, дождя должны сохраняться в таблице в БД. Также, в каждой строке этой таблицы должно содержаться название того сенсора, который прислал эти измерения. То есть сущность "Измерение" имеет связь с сущностью "Сенсор". Все поля у измерения должны валидироваться. Значение "value" должно быть не пустым и находиться в диапазоне от -100 до 100. Значение "raining" должно быть не пустым. Значение "sensor" должно быть не пустым. При этом, название сенсора должно валидироваться в БД. Сенсор с таким названием должен быть зарегистрирован в системе. Если такого сенсора нет в БД - выдавать ошибку. Также, не забывайте про DTO. На сервере, у измерения должно выставляться текущее время, оно должно сохраняться в БД.
- 5) Получение измерений - Возвращает все измерения из БД.
- 6) Получение измерения - Возвращает измерение из БД по идентификатору.
- 7) Получение дождливых дней - Возвращает количество дождливых дней из БД

Методы получения сенсоров должны быть доступны без авторизации. Все остальные методы доступны только после авторизации.

Результат работы вместе с API документацией опубликовать на github.