**Цель:** Реализовать функциональность редактирования заказа в клиент-серверном приложении "Частная пекарня", которая позволит пользователям изменять существующие заказы, добавлять новые товары, изменять количество товаров и удалять товары из заказа.

**Функциональные требования:**

1. **Аутентификация и авторизация:**
   * Пользователь должен быть аутентифицированным для выполнения действий с заказами.
   * Проверка прав пользователя на изменение заказа (например, только владелец заказа может его редактировать).
2. **Редактирование заказа:**
   * Пользователь может изменить статус заказа (например, из "новый" в "подтвержден").
   * Пользователь может добавлять новые товары в заказ.
   * Пользователь может изменять количество уже добавленных товаров.
   * Пользователь может удалять товары из заказа.
   * Система должна проверять наличие товаров на складе перед подтверждением изменений.
3. **Валидация данных:**
   * Все вводимые данные должны проходить проверку на корректность (например, количество товаров не может быть отрицательным).
   * Проверка корректности идентификаторов заказа и товаров.
4. **Обновление данных в базе данных:**
   * Обновление соответствующих записей в таблицах Заказы и Заказанные\_товары.
   * Обновление суммы заказа при изменении товаров или их количества.

**Описание API метода:**

**Метод:** PUT /заказы/{id}

**Описание:** Редактирование существующего заказа.

**Параметры пути:**

* id: Идентификатор редактируемого заказа.

**Тело запроса (JSON):**

json

{

"статус": "подтвержден",

"товары": [

{

"ключ\_товара": 1,

"количество": 2

},

{

"ключ\_товара": 3,

"количество": 1

}

]

}

**Ответы:**

* **200 OK:** Заказ успешно обновлен.
* **400 Bad Request:** Некорректные данные запроса (например, отрицательное количество товаров).
* **401 Unauthorized:** Пользователь не авторизован.
* **403 Forbidden:** Пользователь не имеет прав на редактирование заказа.
* **404 Not Found:** Заказ или товары не найдены.
* **500 Internal Server Error:** Ошибка сервера при обработке запроса.

**Пример запроса:**

http

PUT /заказы/1

Content-Type: application/json

Authorization: Bearer {токен}

{

"статус": "подтвержден",

"товары": [

{

"ключ\_товара": 1,

"количество": 2

},

{

"ключ\_товара": 3,

"количество": 1

}

]

}

**Пример ответа:**

json

{

"сообщение": "Заказ успешно обновлен",

"заказ": {

"идентификатор": 1,

"статус": "подтвержден",

"дата\_заказа": "2025-01-01",

"товары": [

{

"ключ\_товара": 1,

"название": "Булка с изюмом",

"количество": 2,

"стоимость": 50.00

},

{

"ключ\_товара": 3,

"название": "Торт Наполеон",

"количество": 1,

"стоимость": 200.00

}

],

"общая\_стоимость": 300.00

}

}

**Последовательности :**

plaintext

Клиент -> Сервер: PUT /заказы/{id}

Сервер -> База данных: UPDATE Заказы SET статус = 'подтвержден' WHERE идентификатор = {id}

Сервер -> База данных: DELETE FROM Заказанные\_товары WHERE ключ\_заказа = {id}

Сервер -> База данных: INSERT INTO Заказанные\_товары (ключ\_заказа, ключ\_товара, количество) VALUES ...

База данных -> Сервер: Подтверждение обновления

Сервер -> Клиент: Ответ (подтверждение обновления заказа)