Спецификация endpoint'ов для запроса бизнеса по блокировке
платежей
Подготовил: Саитов Э.В.
Казань 2024

1. Спецификация endpoint'ов

Для реализации поставленных бизнесом задач предлагается спецификация, представленная на рисунках 1-4.

```
openapi: 3.0.0
      title: Payment Block API
      version: 1.0.0
     /clients/{clientId}/block:
 6 -
         summary: Заблокировать платежи клиента
8
         - name: clientId
10 -
11
             in: path
12
            required: true
             description: Идентификатор клиента
13
14
         type: string responses:
15
16
            '200':
             description: Платежи клиента успешно заблокированы
18
19 -
             description: Клиент не найден
     /clients/{clientId}/unblock:
23 -
         summary: Разблокировать платежи клиента
24
25 -
          name: clientId
             in: path
28
             description: Идентификатор клиента
29
               type: string
          responses:
32 -
            '200':
33 -
34
             description: Платежи клиента успешно разблокированы
35
              description: Клиент не найден
36
```

Рисунок 1 – Спецификация endpoint'ов (1)

```
/clients/{clientId}/status:
38
39 -
        get:
          summary: Проверить статус блокировки клиента
40
41
           - name: clientId
42 -
              in: path
              required: true
description: Идентификатор клиента
44
                type: string
          responses: '200':
48 -
49 -
50
              description: Статус клиента получен
51 -
                application/json:
54
                     type: object
55 -
                     properties:
56
57
                         type: boolean
58
                         description: Указывает, заблокированы ли платежи клиента
59
60
                         type: string
                         description: Причина блокировки (если применимо)
             '404':
63
               description: Клиент не найден
64
```

Рисунок 2 – Спецификация endpoint'ов (2)

```
/clients/{clientId}/block-reason:
66
           summary: Установить причину блокировки для клиента
68
           parameters:
             name: clientId
 70
               in: path
               description: Идентификатор клиента
 73 -
               schema:
           type: string requestBody: required: true
 74
 75
 76
                application/json:
 78 -
 79
                    type: object
80
                    properties:
81
82
                        type: string
83
                         description: Причина блокировки (например, мошенничество, неверные
                           реквизиты)
85 -
           responses:
              '200':
86
87
               description: Причина блокировки установлена успешно
              '404':
88
89
               description: Клиент не найден
90
    components:
91
       schemas:
ClientStatus:
           type: object
94
96
             type: string
blocked:
97
98
99
                type: boolean
100
              reason:
101
                type: string
```

Рисунок 3 – Спецификация endpoint'ов (3)

Payment Block API OAS 3.0

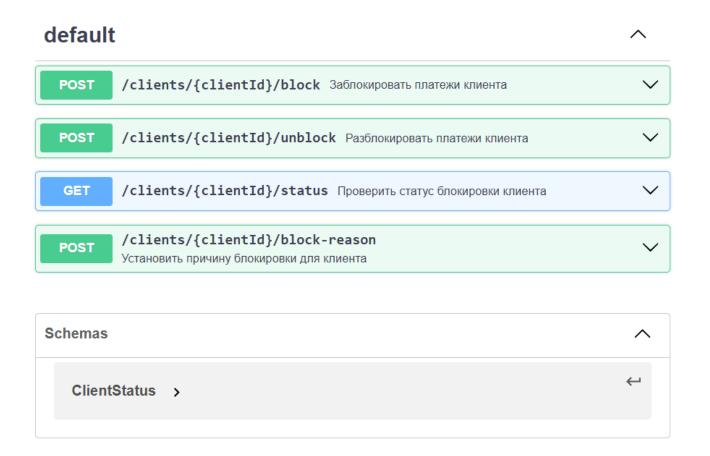


Рисунок 4 – Спецификация endpoint'ов (4)

2. Структура хранения информации в БД

Предлагаемая структура базы данных содержит следующие таблицы: clients (клиенты) с атрибутами:

- client_id (UUID, PRIMARY KEY) уникальный идентификатор клиента;
 - name (VARCHAR) имя клиента;
 - created_at (TIMESTAMP) дата и время создания записи;
- updated_at (TIMESTAMP) дата и время последнего обновления записи.

payment blocks (блокировки платежей) с атрибутами:

- block_id (UUID, PRIMARY KEY) уникальный идентификатор блокировки;
- client_id (UUID, FOREIGN KEY) идентификатор клиента, которому принадлежит блокировка;
- blocked (BOOLEAN) статус блокировки (TRUE заблокирован, FALSE разблокирован);
- reason (VARCHAR) причина блокировки (например, мошенничество, неверные реквизиты);
- created_at (TIMESTAMP) дата и время создания записи о блокировке;
- updated_at (TIMESTAMP) дата и время последнего обновления записи о блокировке.

Помимо этого, предлагаются следующие варианты запросов к базе данных, написанные на языке SQL:

Для блокировки клиента:

INSERT INTO payment_blocks (client_id, blocked,
reason, created_at, updated_at)
VALUES ('<client_id>', TRUE, '<reason>', NOW(),
NOW());

Для разблокировки клиента:

UPDATE payment_blocks
SET blocked = FALSE, updated_at = NOW()
WHERE client id = '<client id>';

Для проверки статуса блокировки: SELECT blocked, reason FROM payment_blocks WHERE client_id = '<cli>id>';

Для установки причины блокировки:

UPDATE payment_blocks

SET reason = '<new_reason>', updated_at = NOW()

WHERE client_id = '<client_id>' AND blocked = TRUE;