

## DRAFT [payment-service] Вызов сервиса (инфо или платёж)

### Цель

Реализовать единый API-метод для вызова сервиса (информационного или платёжного) через Paynet.

Функциональность используется для выполнения проверки реквизитов или создания инвойса на проведения оплаты пользователем на фронте (WEB/MOBILE).

### Описание функционала

#### Общая логика

1. По флагу `is_payment` определяется тип сервиса:

- **Информационный (`is_payment = false`)** — выполняется запрос на **проверку** или получение данных.

В этом случае происходит **синхронный вызов API Paynet** (`/v2/transaction`), полученный ответ кастомизируется согласно таблице `response_field`, после чего возвращается пользователю.

- **Платёжный (`is_payment = true`)** — выполняется **создание и проведение платёжной операции**:

i. Создаётся **инвойс** в `acquiring-service`.

ii. `acquiring-service` инициирует списание средств в `wallet-service` с клиента.

iii. После успешного создания инвойса — создаётся **операция** в таблице `operation`.

iv. Формируется ответ клиенту с дальнейшим ожиданием подтверждения платежа

### Эндпоинт

**Метод:** POST

**URL:** `/{service_id}/execute`

**Авторизация:** Bearer Token

#### Параметры запроса

Параметр	Тип	Описание
<code>service_id</code>	integer	Идентификатор сервиса (из таблицы <code>service</code> )

#### request

```
1 {  
2   "fields": {  
3     "account": "998901234567",  
4     "amount": 10000, // амавнт в мажорных единицах то есть 10 000.00  
5     // сум 00 тийин  
6   },  
7   "currency": "UZS"  
}
```

### response (инфо-сервис)

```
1 {
2     "status": "SUCCESS",
3     "type": "INFO",
4     "serviceId": 5001,
5     "response": [
6         {
7             "label": {
8                 "ru": "ФИО",
9                 "uz": "F.I.Sh.",
10                "en": "Full name"
11            },
12            "value": "Иванов Иван Иванович"
13        },
14        {
15            "label": {
16                "ru": "Задолженность",
17                "uz": "Qarzdorlik",
18                "en": "Debt"
19            },
20            "value": "15000 UZS"
21        }
22    ]
23 }
```

### response (платёжный сервис)

```
1 {
2     "status": "PENDING",
3     "type": "PAYMENT",
4     "serviceId": 5002,
5     "operationId": 120045,
6     "invoiceId": "INV-845390",
7     "response": [
8         {
9             "label": {
10                "ru": "Статус платежа",
11                "uz": "To'lov holati",
12                "en": "Payment status"
13            },
14            "value": "Успешно выполнен"
15        },
16        {
17            "label": {
18                "ru": "Номер операции",
19                "uz": "Operatsiya raqami",
20                "en": "Operation number"
21            },
22            "value": "PAY-20251008-123456"
23        }
24    ]
25 }
26
27 {
28     "type" : "PAYMENT",
29     "id" : "baedfdbf-8ca3-4b27-b2be-b38298d13aab",
30     "currency" : "ETH",
31     "amount" : "0.000105",
32     "meta" : {
33         "paymentAmount" : 5009.63,
34         "paymentCurrency" : "UZS"
35     },
36     "fromAccount" : "84f1514d-55b8-49a8-9e6c-d4c085467883",
37     "feeAmount" : "0.0000105000000000",
38     "status" : "INITIAL",
39     "createdAt" : "2025-10-16T06:18:39.766Z",
40     "updatedAt" : "2025-10-16T06:18:39.766Z",
41     "paymentAmount" : "5009.63",
42 }
```

```
43     "paymentCurrency" : "UZS"
44 }
```

response (Пример ответа при ошибке)

```
1  {
2      "status": "FAILED",
3      "errorCode": "PAYNET_105",
4      "message": {
5          "ru": "Неверный лицевой счёт",
6          "uz": "Hisob raqami noto'g'ri",
7          "en": "Invalid account number"
8      }
9 }
```

## Бизнес-процесс

### Инфо-сервис

1. Получаем данные запроса.
2. Проверяем тип сервиса → `is_payment = false`.
3. Формируем запрос к Paynet API.
4. Получаем ответ.
5. Маппим ответ через `response_field` → только `is_enabled = true`.
6. Возвращаем клиенту кастомизированный ответ.

### Платёжный сервис

1. Проверяем `is_payment = true`.
2. Создаём инвойс в `acquiring-service`.
3. При успешном создании инвойса создаём запись в `operation` со статусом `PENDING`.
4. Отправляем запрос к Paynet API на оплату.
5. По ответу Paynet:
  - Если **ошибка** → обновляем статус операции = `FAILED`, уведомляем `acquiring-service` (триггерим возврат), возвращаем ошибку клиенту на основе `error_code`.
  - Если **успех** → обновляем статус операции = `SUCCESS`, обновляем `userFavorites`, формируем ответ клиенту на основе `response_field`.

## Коды ошибок

Код	Сообщение	Описание
400	Bad Request	Некорректные данные в теле запроса
401	Unauthorized	Неавторизованный пользователь
403	Forbidden	Нет доступа
404	Service Not Found	Указанный сервис не найден
409	Operation Conflict	Повторная попытка выполнения операции

422	Paynet Error	Ошибка на стороне Paynet
500	Internal Server Error	Внутренняя ошибка сервера

#### Критерии приёмки

ID	Имя	Дано	Когда	Тогда
1	Инфо-запрос	Сервис с <code>is_paymen</code> <code>t=false</code>	Отправлен <code>POST</code> <code>/{{servi</code> <code>ce_id}}/</code> <code>execute</code>	Возвращает ся <code>status=</code> <code>SUCCESS ,</code> <code>type=IN</code> <code>F0 ,</code> список ответных полей по <code>respons</code> <code>e_field</code>
2	Платёжный запрос	Сервис с <code>is_paymen</code> <code>t=true</code>	Отправлен <code>POST</code> <code>/{{servi</code> <code>ce_id}}/</code> <code>execute</code>	Создаётся инвойс, операция, Paynet вызывается, ответ возвращает ся пользователю
3	Ошибка Paynet	Paynet возвращает ошибку	Выполнен запрос	Статус операции <code>FAILED ,</code> инициируетс я возврат через <code>acquiri</code> <code>ng-</code> <code>service ,</code> клиенту возвращает ся сообщение об ошибке

4	Успешный платёж	Paynet возвращает успех	Выполнен запрос	Операция <code>SUCCESS</code> , <code>user_favorites</code> обновлён/ создан, возвращён ответ
5	Неавторизованный запрос	Нет токена	Выполнен запрос	Возвращается <code>401 Unauthorized</code>