

UnifiPay Direct API Reference — AI 연동 가이드

이 문서 사용법

- 이 문서는 UnifiPay Direct API의 전체 스펙이다. 원본은 `api-reference-direct/index.html` (사람이 보는 화면용 HTML 페이지)이며, 본 문서는 그 안의 API 스펙 사실만 그대로 옮긴 것이다.
- AI 코딩 어시스턴트는 결제 연동 코드를 작성할 때 **이 문서만을 근거**로 삼는다. 이 문서에 없는 내용을 추측하거나 다른 UnifiPay 문서(Classic/PG·Merchant향 `api-reference/index.html` 등)의 내용을 가져와 채우지 않는다.
- Direct는 "결제 링크 발급" 방식이다. 가맹점 서버가 결제 링크를 먼저 발급받아 고객에게 공유하고, 고객이 그 링크에서 직접 결제를 진행하는 흐름이며, Classic과 API 구성이 다르다.
- 본문 중 `apiKey`, `apiSecret`, `X-API-Key`, 서명값, `transactionId` 등은 모두 설명용 예시 값이다. 실제 연동 시에는 UnifiPay에서 발급받은 실제 값으로 치환해야 한다.
- 원본 페이지에 없는 섹션(예: 결제 환불 API, 자동 정기결제 등)은 이 문서에도 없다. Direct는 환불을 API로 제공하지 않고 콘솔에서만 처리한다 (§2.5 참고).

목차

- [개요](#)
- [1. API 연동 다이어그램](#)
- [1.1 결제 다이어그램](#)
- [2. API 상세 소개](#)
- [Base URL](#)
- [2.1 인증 \(Authentication\)](#)
- [2.2 결제 링크 발급 API](#)
- [2.3 결제 상태 확인 API](#)
- [2.4 결제 Webhook](#)
- [2.5 환불 \(콘솔 처리\)](#)
- [2.6 결제 내역 API](#)
- [3. 테스트 가이드](#)

개요

UnifiPay API를 연동하는 개발자를 위한 API 레퍼런스이다. REST API를 통해 스테이블코인 결제 생성부터 정산까지 연동하는 방법을 상세히 설명한다. 연동 프로세스 및 비즈니스 정보는 Guides 페이지를 참고한다.

핵심 특징 3가지 (원문 카드 문구):

특징	설명
빠른 연동	SDK 없이 REST API만으로 결제 연동 완료
HMAC 보안	API 요청 시 HMAC 서명 인증 적용
Webhook 지원	비동기 Webhook으로 실시간 결제 상태 수신

1. API 연동 다이어그램

결제 링크 발급 방식의 가맹점 서버·고객·Unifi Pay 간 데이터 흐름이다.

범례:

- **실선 화살표** = 동기 호출 (Sync)
- **회색 점선 화살표(열린 화살촉)** = 응답 (Response)
- **주황 점선 화살표** = 비동기 Webhook (Async)

액터: Customer (Browser), Your Server (Backend), Unifi Pay (API Server)

1.1 결제 다이어그램

결제 링크 발급부터 Webhook 수신까지의 전체 흐름이다. (SVG 시퀀스 다이어그램을 텍스트 절차로 재서술함)

1. **[동기]** Your Server → Unifi Pay: POST /api/seller/v1/payment/link
헤더: X-API-Key · X-Timestamp · X-Authorization-Hmac
2. **[응답]** Unifi Pay → Your Server: 200 OK — returnUrl, linkId
3. **[동기]** Your Server → Customer: 고객에게 결제 링크 공유 (linkUrl)
4. **[동기]** Customer → Unifi Pay: 결제 링크 접속
(이 시점부터는 "Unifi Pay 결제 링크 페이지" 상에서 진행됨)
5. **[동기]** Customer → Unifi Pay: '결제 진행하기' 클릭 시 결제 요청 생성
transactionId (UUID) · orderId = LINK-{linkId}-{randomstring}
6. **[동기]** Customer → Unifi Pay: 지갑 연결 및 결제 승인 (온체인 전송)
7. **[비동기 Webhook]** Unifi Pay → Your Server: POST {callbackUrl}
callbackUrl 등록 시에만 전송됨. body: status: CONFIRMED / FAILED / CANCELED

보충 설명 (원문):

- **Your Server:** apiKey 와 HMAC Signature 를 이용해 Unifi Pay에 결제 링크 발급을 요청한다.
- **결제 링크:** 발급받은 returnUrl 을 고객에게 공유한다. 공유 채널에는 제한이 없다.
- **Customer:** 링크에 접속해 결제를 진행하면 결제 요청이 생성되고, 지갑 연결 및 서명으로 결제가 완료된다.
- **Webhook:** 결제 링크 발급 시 callbackUrl 을 등록한 경우에만 전송된다. 등록하지 않으면 결제 상태 확인 API로 조회해야 한다.

2. API 상세 소개

Base URL

환경	Base URL
Preview	https://app-api-pay.unifi.me
Production	https://api-pay.unifi.me

2.1 인증 (Authentication)

서버 간 API 요청은 기본적으로 HMAC 인증이 필수이다. `X-Authorization-Hmac` 헤더에 서명값을 포함해야 한다.

기본 인증 규칙: 별도 표기가 없는 한 서버 간 API는 모두 HMAC 인증이 기본이다. 인증 방식이 다른 API(예: 무인증 공개 API)는 해당 API에 별도로 표기한다.

HMAC 생성 공식

```
BASE64(HMACSHA256(apiSecret, {HTTP_METHOD}{URI}{X-API-Key}{X-Timestamp}{REQUEST_BODY}))
```

예제 1 — REQUEST_BODY가 있는 경우

엔드포인트: `POST /api/seller/v1/payment/link`

요청 예시:

```
request header:
  X-API-Key:          "b6ef2f7c-3e74-438c-b456-689e821992be"
  X-Timestamp:        1773017787000
  X-Authorization-Hmac: "some-hmac-signature"
request body:
{
  "requestId": "REQ-20260304-0001",
  "storeId": "123",
  "serviceName": "MyShop",
  "itemName": "ITEM-GAME-001",
  "itemPrice": 12.34,
```

```

"orderCurrencyCode": "USD",
"returnUrl": "https://myshop.com/payment/returnUrl",
"callbackUrl": "https://myshop.com/payment/callbackUrl",
"expiresAt": "2026-08-31T00:00:00Z"
}

```

서명 생성:

```

X-Authorization-Hmac = Base64(
  HmacSHA256(
    apiSecret,
    "POST"                                ← HTTP_METHOD
    + "/api/seller/v1/payment/link"       ← URI
    + "b6ef2f7c-3e74-438c-b456-689e821992be" ← X-API-Key
    + "1773017787000"                     ← X-Timestamp
    +
    "{\"requestId\":\"REQ-20260304-0001\",\"storeId\":\"123\",\"serviceName\":\"MyShop\",\"itemName\":\"ITEM-GAME-001\",\"itemPrice\":12.34,\"orderCurrencyCode\":\"USD\",\"returnUrl\":\"https://myshop.com/payment/returnUrl\",\"callbackUrl\":\"https://myshop.com/payment/callbackUrl\",\"expiresAt\":\"2026-08-31T00:00:00Z\"}" ← REQUEST_BODY (공백 없이)
  )
)

```

예제 2 — REQUEST_BODY가 없는 경우

엔드포인트: GET /api/seller/v1/payment/123

요청 예시:

```

request header:
X-API-Key:          "b6ef2f7c-3e74-438c-b456-689e821992be"
X-Timestamp:        1773017787000
X-Authorization-Hmac: "some-hmac-signature"

```

서명 생성:

```

X-Authorization-Hmac = Base64(
  HmacSHA256(
    apiSecret,
    "GET"                                ← HTTP_METHOD
    + "/api/seller/v1/payment/123"       ← URI
    + "b6ef2f7c-3e74-438c-b456-689e821992be" ← X-API-Key
    + "1773017787000"                     ← X-Timestamp
  )
)

```

← REQUEST_BODY 없음 (생략)

```
)  
)
```

요청 헤더 파라미터

파라미터	타입	필수	설명
X-Authorization-Hmac	STRING	필수	생성된 HMAC 서명값. 모든 서버 간 API 요청에 필수.
X-API-Key	STRING	필수	Unifi Pay에서 발급받은 apiKey.
X-Timestamp	STRING	필수	밀리초 단위 Unix 타임스탬프. 현재 시각과 5분 이상 차이 시 요청이 거부됨.

⚠ 타임스탬프 유효 범위: X-Timestamp 는 현재 서버 시각에서 5분 이내여야 한다. 이 범위를 벗어나면 재전송 공격(Replay Attack) 방지를 위해 요청이 자동으로 거부된다.

서버 timestamp 조회

Unifi Pay 서버의 timestamp를 조회하여 서버 상태를 확인하고, 응답의 timestamp 값을 HMAC 서명에 활용한다.

```
GET /api/v1/server/time
```

응답 필드:

필드	타입	설명
timestamp	INTEGER(INT64)	서버의 timestamp 밀리초 단위 값

정상 응답 예시 (200 OK):

```
{  
  "timestamp": 1773017787000  
}
```

오류 응답 예시 (500 Internal Server Error):

```
{  
  "code": "INTERNAL_SERVER_ERROR",
```

```
"message": ""
}
```

2.2 결제 링크 발급 API

고정된 금액과 상품의 결제 링크를 발급한다. 응답으로 받은 링크를 고객에게 공유하면 별도 연동 없이 결제 페이지로 진입한다.

```
POST /api/seller/v1/payment/link
```

요청 바디 파라미터

파라미터	타입	길이	필수	설명
requestId	STRING	64	필수	발급 요청에 대한 고유 식별자(먹등키). 영문·숫자·하이픈·언더스 코어만 사용. 동일한 requestId 로 재요청하면 중복 발급으로 거부됨.
storeId	STRING	128	필수	요청자가 제공하는 가맹점 ID 정보.
serviceName	STRING	40	필수	링크 화면과 결제창에 노출될 서비스 또는 가맹점 정보.
itemName	STRING	100	필수	판매 상품의 이름(상품명).
itemPrice	DOUBLE	—	필수	판매 상품의 가격. 주문 통화 기준이며 0보다 커야 함.
orderCurrencyCode	STRING	8	필수	ISO 4217을 따르는 주문 통화 코드. USD 또는 JPY 를 지원.
returnUrl	STRING	2048	선택	결제 완료 후 사용자가 복귀할 URL. 지정하지 않으면 복귀 버튼 없이 결제를 진행.
callbackUrl	STRING	2048	선택	결제 결과를 수신할 Webhook URL. 지정하지 않으면 Webhook을 전송하지 않으므로, 결제 상태 확인 API로 결과를 조회해야 함.
expiresAt	STRING	—	필수	결제 링크의 만료 시각. ISO-8601 형식이며 미래 시점이어야 함. 만료 후 링크에 접속하면 만료 오류로 응답.

응답 필드

필드	타입	설명
linkUrl	STRING	발급된 결제 링크 URL. 고객에게 공유한다.

필드	타입	설명
linkId	STRING	링크 식별자. 가맹점이 보관하며, 링크를 만료시킬 때는 콘솔의 결제 링크 목록에서 강제 만료 버튼을 사용한다.

요청 예시 (JSON)

```
{
  "requestId": "REQ-20260304-0001",
  "storeId": "123",
  "serviceName": "MyShop",
  "itemName": "ITEM-GAME-001",
  "itemPrice": 12.34,
  "orderCurrencyCode": "USD",
  "returnUrl": "https://myshop.com/payment/returnUrl",
  "callbackUrl": "https://myshop.com/payment/callbackUrl",
  "expiresAt": "2026-08-31T00:00:00Z"
}
```

응답 예시 (JSON)

```
{
  "linkUrl": "https://unifipay.example.com/pay/links/3f1d0a549c1e4f6a8a5b2f",
  "linkId": "3f1d0a549c1e4f6a8a5b2f"
}
```

 **참고:** 응답의 linkUrl 을 고객에게 공유하면 별도 연동 없이 결제가 진행된다. requestId 는 멍든키이므로 동일한 값으로 재요청하면 중복 발급이 거부된다.

오류 응답

HTTP 상태	오류 코드	설명
400	BAD_REQUEST	유효하지 않은 입력 값 (형식 오류, 필수 값 누락, 과거 시점의 expiresAt 등)
401	UNAUTHORIZED	유효하지 않은 HMAC
403	FORBIDDEN	허용되지 않은 IP 혹은 Path
503	MAINTENANCE	점검 중
406	PAYMENT_INVALID_REQUEST	동일한 requestId로 재요청한 경우 또는 callbackUrl이 허용 목록에 없는 경우

HTTP 상태	오류 코드	설명
406	PAYMENT_UNSUPPORTED_POLICY	지원하지 않는 주문 통화 또는 금액 정책 (orderCurrencyCode, itemPrice)
406	PAYMENT_STORE_INACTIVE	요청한 storeId가 ACTIVE 상태가 아닐 때
406	PAYMENT_WALLET_NOT_REGISTERED	정산 지갑 주소가 등록되지 않은 경우
500	INTERNAL_SERVER_ERROR	서버 내부 오류

2.3 결제 상태 확인 API

생성된 결제 건의 현재 상태를 확인한다.

```
GET /api/seller/v1/payment/{transactionId}
```

응답 필드

필드	타입	설명
transactionId	STRING	Unifi Pay에서 생성한 거래 번호.
orderId	STRING	가맹점 거래에 대한 ID 정보.
storeId	STRING	요청자가 제공하는 가맹점 ID 정보.
serviceName	STRING	결제창에 노출될 서비스 또는 가맹점 정보.
status	STRING	아래 enum 참고
failType	STRING	status 가 FAILED 일 때 설정. 아래 enum 참고
items	Array	결제 상품 목록. 최대 1건
items[].name	STRING	판매 상품의 식별 정보.
items[].price	DOUBLE	판매 상품의 가격.
orderCurrencyCode	STRING	ISO 4217를 따르는 주문 통화 코드. 현재 "USD" 로 고정.
orderAmount	DOUBLE	주문한 금액.
payCurrencyCode	STRING	결제한 스테이블 코인 종류.
payAmount	DOUBLE	결제된 금액.
blockchainTxId	STRING	블록체인 트랜잭션 해시. 결제가 최종 성공(CONFIRMED)했거나, 블록체인 네트워크에 전파된 후 실패(FAILED)한 경우에 반환.
blockchainNetworkFee	DOUBLE	블록체인 네트워크 수수료. 결제 성공(CONFIRMED) 및 네트워크 전파 후 실패(FAILED) 시점에 실제로 소모된 수수료 금액이 반환됨. (Unifi가 전액 대납하여 사용자 부담은 없음.)

status enum:

값	설명
CREATED	결제 생성 완료
PENDING	결제 진행 중
CONFIRMED	결제 성공 (종결)
FAILED	결제 실패 (종결)
CANCELED	결제 취소 (종결)

failType enum (status가 FAILED일 때 설정):

값	설명
WALLET_PURCHASE_BLOCKCHAIN_FAILED	unifi wallet에서 blockchain 출고 과정 중 실패
WALLET_PURCHASE_INSUFFICIENT_BALANCE	사용자 지갑의 잔고 부족
PAYMENT_PURCHASE_BLACKLISTED	사용자의 지갑이 payment 블랙리스트에 포함되어 실패
UNKNOWN	결제 과정 중 알 수 없는 오류

응답 예시

```
{
  "transactionId": "<uuid>",
  "orderId": "<orderId>",
  "storeId": "<storeId>",
  "serviceName": "<serviceName>",
  "items": [
    {
      "name": "ITEM-GAME-001",
      "price": 50000
    }
  ],
  "status": "CONFIRMED",
  "failType": null,
  "countryCode": "KOR",
  "payAmount": 49000,
  "payCurrencyCode": "USDT",
  "orderAmount": 50000,
  "orderCurrencyCode": "USD",
  "blockchainTxId":
    "0x4fce0d72ff7f4b9f4ba0cf58d30119924bea3811d85a830c762e1d87b3e1ee17",
}
```

```
"blockchainNetworkFee": 0.00253778
}
```

(참고: 응답 예시에는 위 "응답 필드" 표에 명시되지 않은 `countryCode` 필드가 실제로 포함되어 있다. 원문 그대로 옮김.)

상태 안내:

- **CONFIRMED**: 결제가 성공적으로 완료되어 확정된 상태.
- **FAILED**: 결제가 실패한 상태.

오류 응답

HTTP 상태	오류 코드	설명
400	BAD_REQUEST	유효하지 않은 입력 값
401	UNAUTHORIZED	유효하지 않은 HMAC일 경우
403	FORBIDDEN	허용되지 않은 IP 혹은 Path 일 경우
503	MAINTENANCE	점검 중일 경우
406	PAYMENT_TRANSACTION_NOT_FOUND	유효하지 않은 transactionId
500	INTERNAL_SERVER_ERROR	서버 내부 오류

2.4 결제 Webhook

생성한 결제 건에 대한 결제 완료 여부를 비동기로 수신한다.

POST <가맹점 callbackUrl>

요청 바디 필드

필드	타입	필수	설명
transactionId	STRING	필수	Unifi Pay에서 생성한 거래 번호.
orderId	STRING	필수	Unifi Pay에서 생성한 주문 id. 결제 링크를 통한 거래는 <code>LINK-{linkId}-{임의 문자열}</code> 형식으로 발급되며, 고객이 결제 링크에서 결제를 진행한 시점에 결정됨.
status	STRING	필수	결제 처리 결과. <code>CONFIRMED</code> / <code>FAILED</code> / <code>CANCELED</code> .
type	STRING		거래 유형. 환불을 API로 제공하지 않으므로 항상 <code>PURCHASE</code> .

필드	타입	필수	설명
		필수	

요청 예시 (JSON)

```
{
  "transactionId": "<uuid>",
  "orderId": "LINK-3f1d0a549c1e4f6a8a5b2f-a1b2c3",
  "status": "CONFIRMED",
  "type": "PURCHASE"
}
```

 **응답 필수 안내:** 가맹점 서버는 Webhook 수신 후 반드시 HTTP 200 OK 응답 코드를 반환해야 한다.

 **상세 조회:** Webhook은 최소한의 정보만 전달한다. 결제 상세 정보는 Webhook 수신 후 [2.3 결제 상태 확인 API](#)를 호출하여 조회한다.

재시도 정책

가맹점 서버에서 5xx 오류 또는 응답이 없을 경우, 아래 정책에 따라 재전송된다.

순서	재시도 시점
1차 재시도	3초 후
2차 재시도	30초 후
3차 재시도	10분 후
4차 재시도	1시간 후
5차 재시도	3시간 후

2.5 환불 (콘솔 처리)

환불은 API로 제공되지 않는다. 지갑 서명이 필요한 처리이므로 콘솔에서만 실행할 수 있다.

처리 방법: 판매자가 Unifi Pay 콘솔의 **환불 관리** 메뉴에서 본인 명의의 Unifi Wallet으로 직접 서명하여 실행한다.

연동 시 참고 사항

항목	설명
환불 Webhook	제공하지 않는다. 환불 처리 결과는 Webhook으로 전송되지 않으므로 결제 내역 API로 조회해야 한다.
환불 내역 조회	아래 2.6 결제 내역 API 에서 <code>paymentType=REFUND</code> 로 조회한다.
원결제 연결	환불 건의 <code>originTransactionId</code> 가 원결제의 <code>transactionId</code> 를 가리킨다.

💡 **참고:** 환불 지급·환불 금액 기준·주소 및 네트워크 조건 등 환불 정책은 이용 가이드 "5. 환불(송금 보조 기능)" ([../guides/index.html#refund](#))을 참고한다.

2.6 결제 내역 API

결제, 환불 내역을 확인한다.

```
GET /api/seller/v1/payment/settlement/transaction
```

요청 쿼리 파라미터

파라미터	타입	필수	설명
<code>from</code>	STRING	조건부	조회 시작일 (<code>capturedAt</code> 기준, ISO 8601 UTC. 예시 <code>2026-03-01T00:00:00Z</code>). <code>transactionId/settlementId</code> 미지정 시 필수이며, <code>to</code> 보다 이전 시점 설정 필수.
<code>to</code>	STRING	조건부	조회 종료일 (<code>capturedAt</code> 기준, ISO 8601 UTC. 예시 <code>2026-03-31T23:59:59Z</code>). <code>transactionId/settlementId</code> 미지정 시 필수이며, 조회 기간은 최대 90일까지 허용.
<code>paymentType</code>	STRING	선택	결제 타입 필터 (<code>PURCHASE</code> / <code>REFUND</code>). 미지정 시 전체.
<code>transactionId</code>	STRING	선택	Unifi Pay에서 생성한 거래 번호. <code>from/to/settlementId</code> 미지정 시 필수.
<code>settlementId</code>	STRING	선택	Unifi Pay 지급 명세서 ID 정보. <code>from/to/transactionId</code> 미지정 시 필수.
<code>page</code>	INTEGER	선택	페이지 번호 (0부터 시작, 기본값: 0)
<code>size</code>	INTEGER	선택	페이지 크기 (기본값: 20)

응답 필드 (200 OK)

필드 이름	타입	필수	설명
content	List	필수	결제 데이터 목록
content[].partnerCorpName	STRING	필수	법인명 (EN)
content[].storeId	STRING	선택	가맹점 ID 정보
content[].orderId	STRING	필수	거래에 대한 ID 정보
content[].itemName	STRING	필수	상품의 식별 정보
content[].serviceName	STRING	선택	서비스 또는 가맹점명 (EN)
content[].orderCountryCode	STRING	필수	구매자 국가 코드
content[].orderCurrencyCode	STRING	필수	ISO 4217를 따르는 주문 통화 코드
content[].orderAmount	DOUBLE	필수	주문 금액
content[].createdAt	INSTANT	필수	주문 결제 요청 일시
content[].transactionId	STRING	필수	Unifi Pay에서 생성한 거래 번호.
content[].blockchainTxId	STRING	필수	KaiaScan TxID
content[].paymentType	STRING	필수	결제: PURCHASE / 환불: REFUND
content[].originTransactionId	STRING	선택	환불 시 원 거래 번호 (REFUND인 경우 필수, PURCHASE인 경우 Null)
content[].status	STRING	필수	CONFIRMED 고정 (CONFIRMED건만 조회)
content[].finalizedAt	INSTANT	필수	주문 결제 최종 상태 일시
content[].capturedAt	INSTANT	필수	주문 결제 완료 일시

필드 이름	타입	필수	설명
<code>content[].failType</code>	STRING	선택	주문 결제 실패 사유
<code>content[].payCurrencyCode</code>	STRING	필수	실 결제 통화 USDT
<code>content[].payAmount</code>	DOUBLE	필수	실 결제 금액
<code>content[].paymentExchangeRate</code>	DOUBLE	필수	결제 시점 환율. 결제 통화와 주문 통화가 다를 때 적용되며, 같으면 1. (기준 통화: <code>payCurrencyCode</code>)
<code>content[].buyerWalletAddress</code>	STRING	필수	구매자 결제 지갑 주소
<code>content[].variableFeeRate</code>	DOUBLE	선택	결제 수수료율 (%). Unifi Pay 지급 명세서 번호 생성 전인 경우 Null
<code>content[].settlementId</code>	STRING	선택	Unifi Pay 지급 명세서 ID 정보. 생성 전인 경우 Null
<code>content[].settlementCurrencyCode</code>	STRING	선택	정산 통화 코드 (USDT, JPYC). 정산 전인 경우 Null
<code>content[].settlementExchangeRate</code>	DOUBLE	선택	정산 시점 환율. 정산 전인 경우 Null
<code>totalElements</code>	LONG	필수	전체 건수
<code>totalPages</code>	INTEGER	필수	전체 페이지수
<code>pageNumber</code>	INTEGER	필수	현재 페이지 번호
<code>first</code>	BOOLEAN	필수	첫 페이지 여부
<code>last</code>	BOOLEAN	필수	마지막 페이지 여부

응답 예시 — 성공

```
{
  "content": [
    {
      "partnerCorpName": "ABC Corporation",
      "storeId": "store_001",
```

```

    "orderId": "ORD-20260301-0001",
    "itemName": "Premium Subscription",
    "serviceName": "ABC Service",
    "orderCountryCode": "KR",
    "orderCurrencyCode": "USD",
    "orderAmount": 50000,
    "createdAt": "2026-03-01T10:00:00Z",
    "transactionId": "txn_abc123def456",
    "blockchainTxId": "0xabcdef1234567890abcdef1234567890abcdef12",
    "paymentType": "PURCHASE",
    "originTransactionId": null,
    "status": "CONFIRMED",
    "finalizedAt": "2026-03-01T10:01:30Z",
    "capturedAt": "2026-03-01T10:01:25Z",
    "failType": null,
    "payCurrencyCode": "USDT",
    "payAmount": 10.87,
    "paymentExchangeRate": 1.0,
    "buyerWalletAddress": "0x1234567890abcdef1234567890abcdef12345678",
    "variableFeeRate": 1.5,
    "settlementId": "stl_xyz789",
    "settlementCurrencyCode": "USDT",
    "settlementExchangeRate": 1.0
  }
],
"totalElements": 1,
"totalPages": 1,
"pageNumber": 0,
"first": true,
"last": true
}

```

⚠ 원본 소스 불일치 주의: 원본 HTML 페이지에는 위 화면 표시용 JSON(위 예시, `orderCurrencyCode: "USD", orderAmount: 50000`)과, 복사 버튼에 내장된 JSON(`orderCurrencyCode: "KRW", orderAmount: 15000.0`)이 서로 다른 값으로 존재한다. 이는 원본 페이지 자체의 불일치이며, 어느 쪽이 정확한 예시인지는 이 문서만으로 확인할 수 없다. 실제 연동 시 통화·금액 필드의 정확한 동작은 UnifiPay 기술 지원(pay-support@unifi.me)에 확인 후 진행을 권장한다.

응답 예시 — 실패

```
{
  "code": "PAYMENT_INVALID_REQUEST",
  "message": "Date range must not exceed 90 days"
}
```

오류 타입 코드

HTTP 상태	오류 코드	설명
400	BAD_REQUEST	필수 파라미터 누락 또는 타입 불일치
404	NOT_FOUND	파트너 정보를 찾을 수 없음
406	PAYMENT_INVALID_REQUEST	유효하지 않은 요청 (날짜 형식 오류, 필수 조건 미충족, 90일 범위 초과 등)
500	INTERNAL_SERVER_ERROR	서버 내부 오류

3. 테스트 가이드

Preview 환경에서 Unifi Pay 결제 연동을 테스트하는 방법이다.

1. Unifi Pay 테스트는 Preview 환경에서 진행할 수 있다.
2. Unifi Wallet은 Preview 환경에서 Production 기준의 Google 계정으로 생성 및 연동을 진행한다. (Preview 환경에서 생성한 Unifi Wallet은 Production 환경과 별도로 관리된다.) Preview Unifi Wallet: <https://app.unifi.me/>
3. Preview 환경의 결제는 Kaia Blockchain의 테스트넷 환경인 Kairos 기준으로 진행된다.
4. 아래 가이드를 참고하여 Kairos 테스트넷에서 테스트용 USDT를 확보한 후 결제 테스트를 진행한다.

테스트 환경 구성 절차 (SVG 다이어그램 텍스트 재서술, "PREVIEW (Kairos Testnet)" 영역 내 3단계):

1. **STEP 1 — Unifi Wallet 생성:** Preview 환경의 Unifi Wallet 생성 및 연동
2. **STEP 2 — 테스트 USDT 확보:** Kaia Faucet에서 Kairos USDT 수령
3. **STEP 3 — 결제 테스트 진행:** Preview 환경에서 API 연동 테스트

Preview 환경 정보:

항목	값
환경	Preview 환경 / Kairos Testnet (테스트넷)
설명	Preview 환경에서는 카이아 블록체인의 테스트넷인 Kairos 체인이 연동되며, 테스트 자금은 Kaia Faucet 서비스를 통해 확보할 수 있다.

항목	값
Kaia Faucet	https://www.kaia.io/ko/faucet
Base URL	https://app-api-pay.unifi.me