

네이버페이 결제형 독립몰 연동 개발 가이드



결제형 독립몰 연동 개발 가이드

1. 결제형 서비스 소개

1-1. 결제 플랫폼 소개	3
----------------	---

2. 네이버페이 API 개발가이드

2-1. 사전 준비 항목	4
---------------	---

2-2. 인증정보 및 인증방법	5
------------------	---

2-3. API 서버 및 결제 페이지 도메인	6
--------------------------	---

2-4. API 기본 응답 형식	7
-------------------	---

2-5. 간편결제 플랫폼	8
---------------	---

2-6. 정기/반복결제 플랫폼	13
------------------	----

2-7. 부가서비스	20
------------	----

3. 네이버페이 검수 안내

3-1. 네이버페이 체크리스트 사전 점검 안내	21
---------------------------	----

3-2. 검수 체크리스트 항목	23
------------------	----

3-3. 검수 신청 방법 및 소요 기간	24
-----------------------	----

1-1. 결제 플랫폼 소개

네이버페이 간편결제란?

가장 일반적인 결제 타입으로 단건 결제를 의미

이용자는 다양한 네이버페이 가맹점에서 네이버페이 결제수단을 선택하고
최초 등록된 비밀번호 입력만으로 간편하게 결제를 진행할 수 있습니다.

ex) 배달, 쇼핑, 티켓 등 1회성 결제

네이버페이 정기/반복결제란?

최초 1회 결제수단을 등록하고 스케줄링에 의해 주기별 반복 결제를 의미.

이용자는 정기/반복결제를 통해 매번 결제를 진행해야 하는 번거로움 없이 상품 구매 및 서비스를
제공 받으실 수 있습니다.

ex) 정기 배송, 정기 구독, 이용권 정기 구매 등 주기별 반복 결제

2-1. 사전 준비 항목

방화벽

가맹점에서는 아웃바운드 트래픽에 대해 443 포트 허용이 필요하며, 가맹점 정책상 방화벽 설정이 필요할 경우 도메인 단위로 설정을 권장합니다.

인증서 방식

TLS 1.2 프로토콜을 지원하는 클라이언트 라이브러리 사용을 권장합니다.
SSL은 지원하지 않습니다.

API 하위 호환성

- 1) 하위 호환 준수 변경에 대해서는 제공되는 API의 버전 변경 없이 진행됩니다.
- 2) 하위 호환 미 준수 변경에 대해서는 새롭게 버저닝하거나, 신규 API URL 를 통해 제공됩니다.

※ 하위 호환 준수/미 준수 변경 기준 - 네이버페이 개발자센터 참고

2-2. 인증정보 및 인증방법

네이버페이 개발에 필요한 인증정보

네이버페이 가입이 완료되면 개발에 필요한 인증정보를 전달 드립니다.

1) 파트너 ID

API 요청 시 URL에 입력 필요

```
https://dev.apis.naver.com/{파트너 ID}/naverpay/payments/v2.2/apply/payment
```

2) Client ID, Client Secret, Chain ID

API 요청 시 요청 헤더에 입력 필요 (Chain ID는 그룹 타입 가맹점에만 발급)

※ Client Secret 보안상 노출되지 않아야 하며, 클라이언트 바이너리 등에 저장하면 안 됩니다.

외부에 노출된 경우 해당 값에 대한 인증을 무효화하고 새로 발급받아 사용해야 합니다.

```
X-Naver-Client-Id: {Client ID}
```

```
X-Naver-Client-Secret: {Client Secret}
```

```
X-NaverPay-Chain-Id : {Chain ID}
```

2-3. API 서버 및 결제 페이지 도메인

백엔드 API 서버 URL

※ 반드시 HTTPS로 호출해야 합니다.

```
https://{API 도메인}/{파트너 ID}/naverpay/payments/{API 버전}/{API명}
```

- 개발 API 서버 dev.apis.naver.com
- 운영 API 서버 apis.naver.com

결제 페이지

※ NAVER Pay javascript SDK 방식으로 결제창을 호출하실 경우 스크립트 추가만으로 결제창을 생성, 호출할 수 있습니다.

```
https://{네이버페이 서비스 도메인}/payments/{결제 예약 ID}
```

개발용 서비스 서버 도메인

- PC 웹 test-pay.naver.com
- 모바일 웹 test-m.pay.naver.com

운영 서비스 서버 도메인

- PC 웹 pay.naver.com
- 모바일 웹 m.pay.naver.com

2-4. API 기본 응답 형식

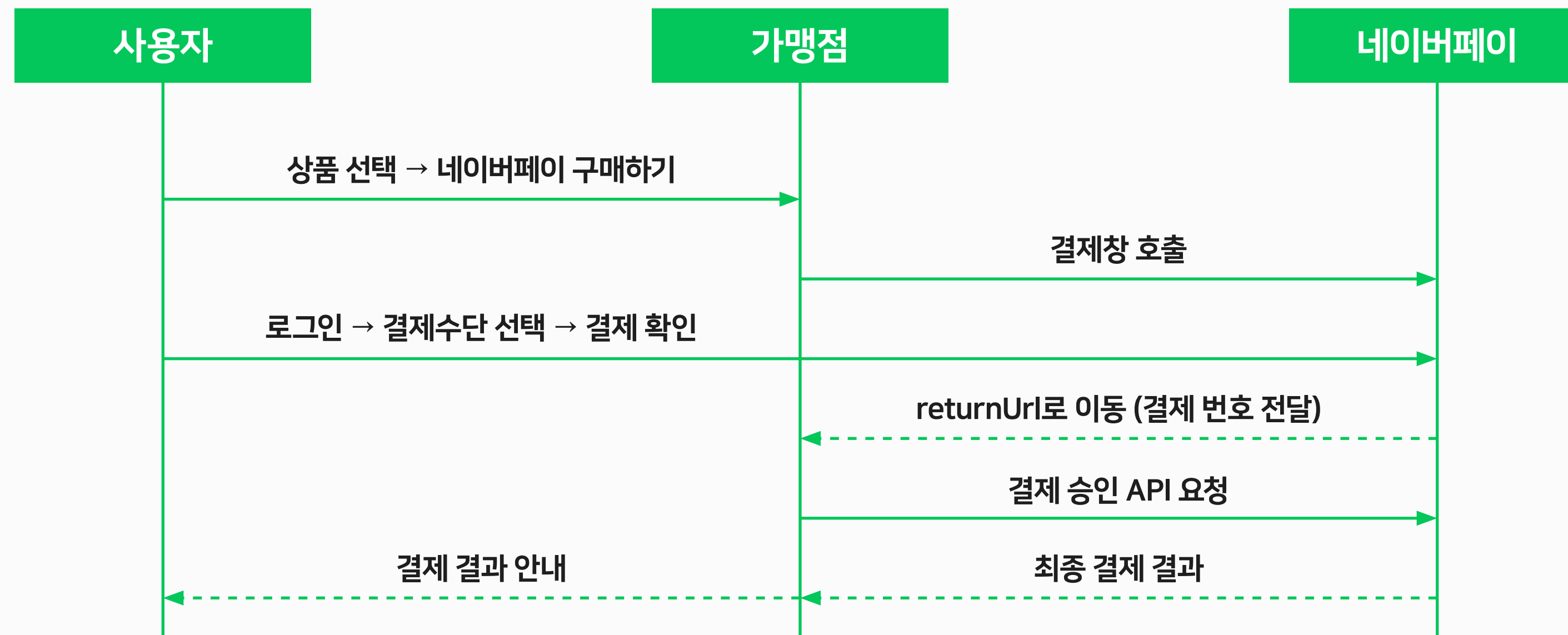
기본 응답 필드

필드명	타입	필수 여부	설명
code	String of Codes	0	결과 코드
message	String	0	상세 메시지
body	Object		응답 데이터

```
Response
HTTP/1.1 200 OK
{
  "code": "Success",
  "message": "detail message(optional)",
  "body": {
    "paymentId": "20170201NP1043587746",
    "detail": {
      "productName": "상품명",
      "merchantId": "loginId",
      "merchantName": "가맹점명",
      "cardNo": "465887*****",
      "admissionYmdt": "20170201151722",
      "payHistId": "20170201NP1043587781",
      "totalPayAmount": 1000,
      "primaryPayAmount": 1000, ...
    }
  }
}
```

2-5. 간편결제 플랫폼

간편결제 플랫폼 결제 흐름



2-5. 간편결제 플랫폼

결제 / 취소 연동

1) NAVER Pay javascript SDK 결제창 호출

```
<html>
<head></head>
<body>
<input type="button" id="naverPayBtn" value="네이버페이 결제 버튼">
<script src="https://nsp.pay.naver.com/sdk/js/naverpay.min.js"></script>
<script>
  var oPay = Naver.Pay.create({ //네이버페이 객체를 생성합니다.
    "mode": "development", // development(개발환경) or production(운영환경)
    "clientId": "가입완료 후 발급 받은 인증정보 입력"
    //"chainId": "{그룹 타입일 경우 chainId를 넣어주세요}"
    "payType": "normal", // normal(간편결제) or recurrent(정기/반복결제)
    "openType": "page"
  });

  //직접 만드신 네이버페이 결제버튼에 click Event를 할당하세요
  var elNaverPayBtn = document.getElementById("naverPayBtn");

  elNaverPayBtn.addEventListener("click", function() {
    oPay.open({ // 네이버페이 결제 화면을 호출하며, 파라미터에 결제 데이터를 입력합니다.
      "merchantPayKey": "가맹점 주문번호",
      "productName": "대표 상품명",
      "totalPayAmount": 10000,
      "taxScopeAmount": 10000,
      "taxExScopeAmount": 0,
      "productCount": 1,
      "returnUrl": "http(s)://your-site-return-url"
    });
  });
</script>
</body>
</html>
```

2-5. 간편결제 플랫폼

2) 결제 인증 결과 전달

- 사용자가 결제창에서 결제를 진행하면 가맹점이 설정한 returnUrl 또는 onAuthorize callback function 으로 결제 인증 결과가 전달됩니다.
- 성공 응답

필드명	타입	설명
resultCode	String	결제 결과(Success: 성공)
paymentId	String	네이버페이 결제번호. 최대 50바이트

- 실패 응답

필드명	타입	설명
resultCode	String	결제 결과(Fail: 실패)
resultMessage	String	실패 사유 메시지 (사용자 안내 필요)
reserveld	String	결제 예약 ID

- 가맹점은 결제 인증 결과를 확인하여 실패 응답을 받은 경우 실패 사유를 사용자에게 안내하고, 성공 응답인 경우 가맹점의 주문 정보를 생성한 후 네이버페이에 결제 승인을 요청합니다.

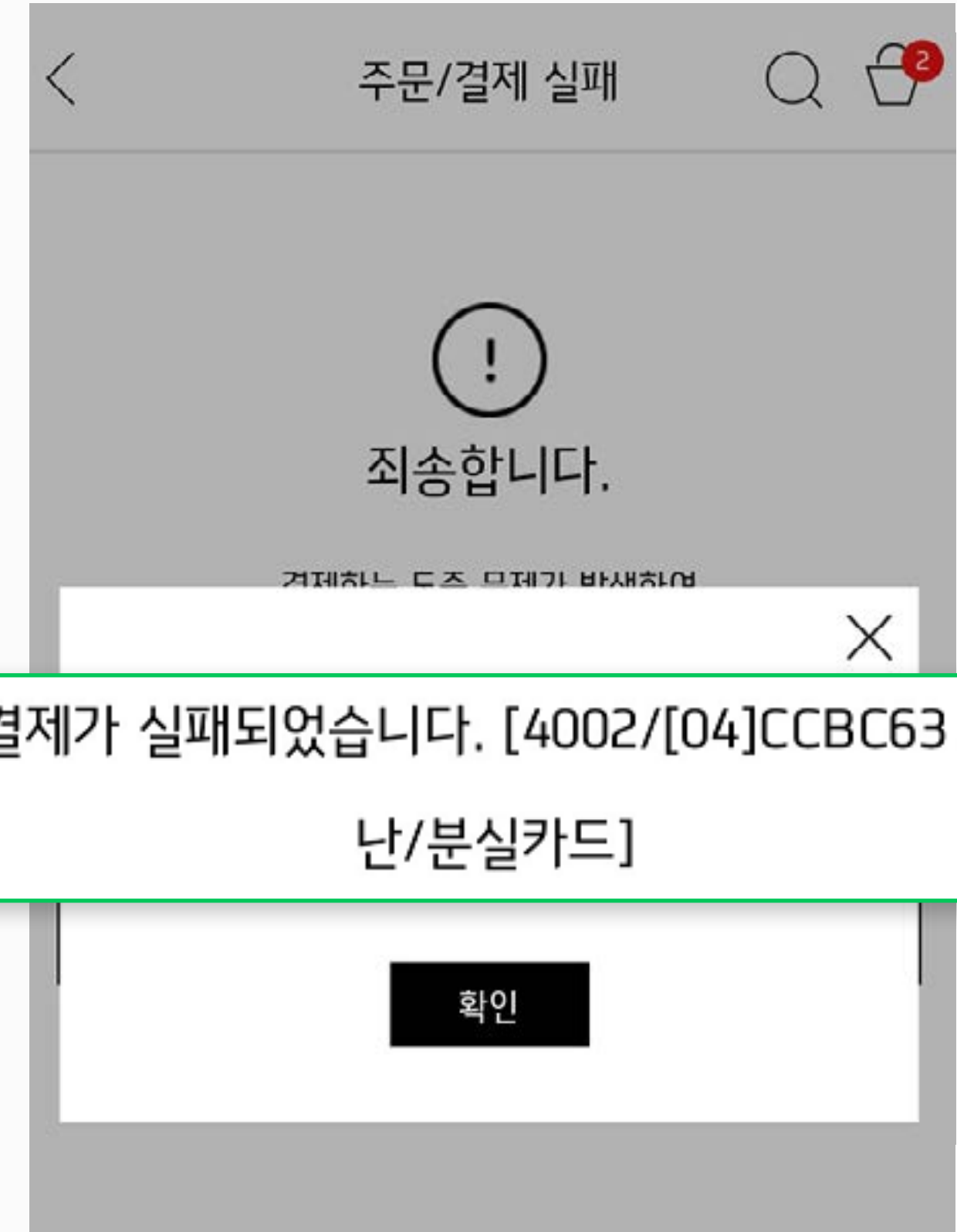
2-5. 간편결제 플랫폼

3) 결제 승인 API

- 결제 인증 결과로 전달 받은 paymentId(네이버페이 결제번호)로 결제 승인을 요청합니다.
- 결제 승인 API 성공 시 실 승인되며, 실패 시 사유를 인지할 수 있도록 사용자 안내가 필요합니다.
- 실패 사유 안내 예시)

[응답]

```
responseBody={"code":"Fail","message":"4002/[04]CCBC63 도난/분실카드"}
```



2-5. 간편결제 플랫폼

4) 결제 취소 API

paymentId(네이버페이 결제번호)로 결제 취소를 요청하며,
cancelAmount(취소 요청 금액)에 따라 총 결제금액에서 전체 취소 또는 부분 취소로 처리됩니다.

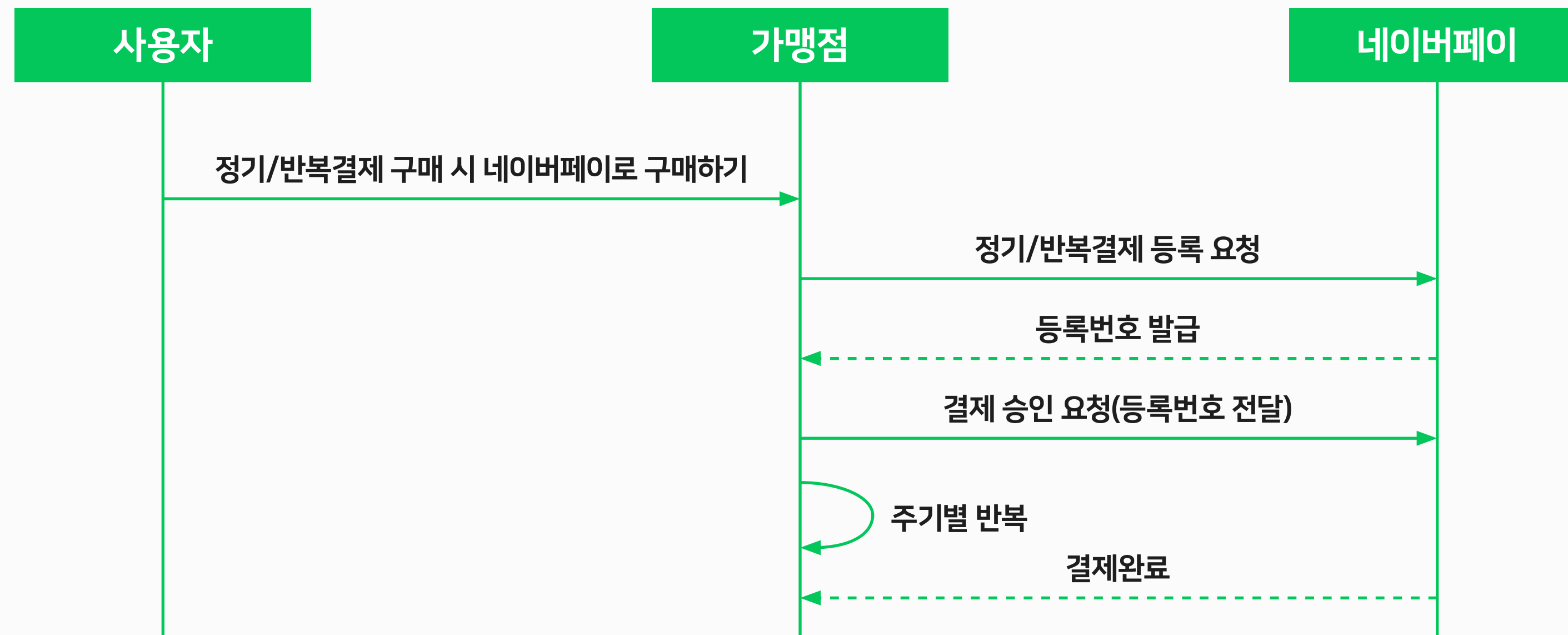
※ 응답 결과 코드 예외 사항

네이버페이는 카드, 계좌, 포인트 복합 결제를 지원하는 특성상
취소 처리 과정 원천사(카드/은행)의 실패 시 취소실패(CancelNotComplete) 상태로 분류합니다.
CancelNotComplete 응답 코드를 받으신 경우 취소 성공과 동일하게 주문 처리가 필요합니다.

코드 값	타입	설명
CancelNotComplete	String	네이버페이 취소 진행 중 상태로 취소처리가 완료되지 않았으나 네이버페이 자동 재시도를 통해 취소 완료 예정

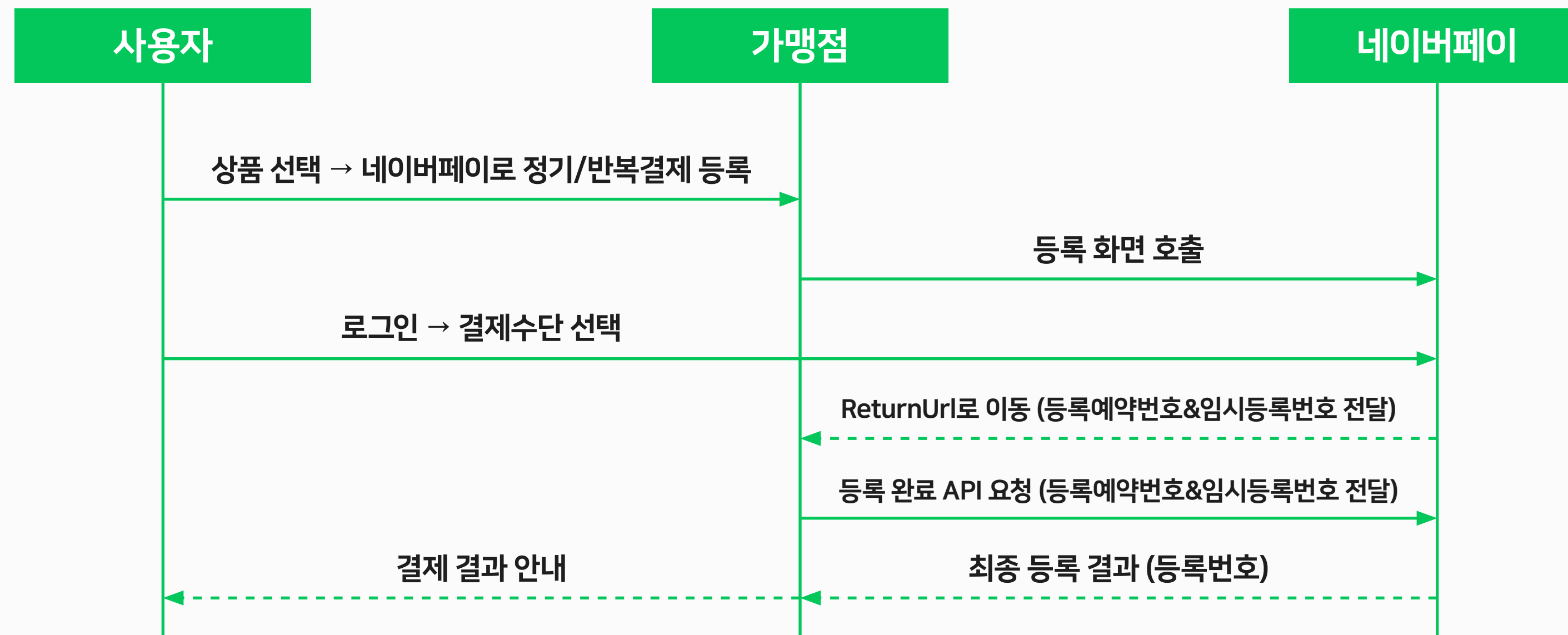
2-6. 정기/반복결제 플랫폼

정기/반복결제 개요



2-6. 정기/반복결제 플랫폼

정기/반복결제 등록 흐름



2-6. 정기/반복결제 플랫폼

결제수단 등록 연동

1) NAVER Pay javascript SDK 등록창 호출

```
<html>
<head></head>
<body>
<input type="button" id="naverPayBtn" value="네이버페이 정기/반복결제 버튼">
<script src="https://nsp.pay.naver.com/sdk/js/naverpay.min.js"></script>
<script>
  var oPay = Naver.Pay.create({ //네이버페이 객체를 생성합니다.
    "mode" : "development", // development(개발환경) or production(운영환경)
    "clientId": "가입완료 후 발급 받은 인증정보 입력"
    //"chainId" : "{그룹 타입일 경우 chainId를 넣어주세요}"
    "payType": "recurrent" // normal(간편결제) or recurrent(정기/반복결제)
  });

  //직접 만드신 네이버페이 결제버튼에 click Event를 할당하세요
  var elNaverPayBtn = document.getElementById("naverPayBtn");

  elNaverPayBtn.addEventListener("click", function() {
    oPay.open({ // 네이버페이 결제수단 등록 화면을 호출하며, 파라미터에 결제 데이터를 입력합니다.
      "actionType": "NEW", // NEW(신규 등록) or CHANGE(결제수단 변경)
      "targetRecurrentId": "현재 사용하는 recurrentId", // actionType이 "CHANGE"면 현재 사용하고 있는 recurrentId 전달
      "productCode": "상품 코드",
      "productName": "대표 상품명",
      "totalPayAmount": 10000,
      "returnUrl": "http(s)://your-site-return-url"
    });
  });
</script>
</body>
</html>
```

2-6. 정기/반복결제 플랫폼

2) 등록 인증 결과 전달

- 사용자가 등록창에서 등록을 진행하면 가맹점이 설정한 returnUrl 또는 onAuthorize callback function 으로 등록 인증 결과가 전달됩니다.
- 성공 응답

필드명	타입	설명
resultCode	String	등록 결과(Success: 성공)
reserveld	String	등록 예약 번호
tempReceiptId	String	임시 등록 번호

- 실패 응답

필드명	타입	설명
resultCode	String	등록 결과(Fail: 실패)
resultMessage	String	실패 사유 메시지 (사용자 안내 필요)
reserveld	String	등록 예약 번호

- 가맹점은 등록 인증 결과를 확인하여 실패 응답을 받은 경우 실패 사유를 사용자에게 안내하고, 성공 응답인 경우 가맹점의 주문 정보를 생성한 후 네이버페이에 등록 완료를 요청합니다.

2-6. 정기/반복결제 플랫폼

3) 등록 완료 API

- 정기/반복결제 최종 등록 완료를 요청합니다.
- 가맹점 returnUrl에 등록 인증 결과로 받은 reserveld(등록 예약 번호)와 tempReceiptId(임시 접수 번호)로 등록 완료를 요청합니다.

4) 등록 해지 API

- recurrentId(등록 번호)로 정기/반복결제 등록 해지를 요청합니다.
- 결제가 더 이상 발생하지 않는 시점에 recurrentId(등록 번호)를 해지해 주셔야 합니다.

5) 등록 내역 조회 API

- 정기/반복결제 등록 내역을 조회하며 페이징 형태로 제공됩니다.
- 등록 상태와 등록/해지된 날짜, 결제수단 등이 응답됩니다.
- 검색 조건의 전체 데이터를 조회하기 위해서는 totalPageCount(전체 페이지 개수)를 확인하신 뒤 pageNumber(조회하고자하는 페이지번호)를 설정하여 반복 호출해 주시면 됩니다.

2-6. 정기/반복결제 플랫폼

결제 / 취소 연동

1) 결제 예약 API

- 결제 주기에 맞춰 결제 예약/승인 API 요청이 필요 합니다.
- 등록 완료 시 전달 받은 recurrentId(등록 번호)로 결제 예약을 요청합니다.

2) 결제 승인 API

- recurrentId(등록번호)와 결제 예약 시 전달 받은 paymentId(결제번호)로 결제 승인을 요청합니다.
- 결제 승인 API 성공 시 실 승인되며, 실패 시 사유를 인지할 수 있도록 사용자 안내가 필요합니다.
- 실패된 경우 가맹점 정책에 맞춰 재 요청 혹은 등록 해지 해주셔야 합니다.

2-6. 정기/반복결제 플랫폼

3) 결제 취소 API

paymentId(네이버페이 결제번호)로 결제 취소를 요청하며,
cancelAmount(취소 요청 금액)에 따라 총 결제금액에서 전체 취소 또는 부분 취소로 처리됩니다.

※ 응답 결과 코드 예외 사항

네이버페이는 카드, 계좌, 포인트 복합 결제를 지원하는 특성상
취소 처리 과정 원천사(카드/은행)의 실패 시 취소실패(CancelNotComplete) 상태로 분류합니다.
CancelNotComplete 응답 코드를 받으신 경우 취소 성공과 동일하게 주문 처리가 필요합니다.

코드 값	타입	설명
CancelNotComplete	String	네이버페이 취소 진행 중 상태로 취소처리가 완료되지 않았으나 네이버페이 자동 재시도를 통해 취소 완료 예정

2-7. 부가서비스 (간편결제, 정기/반복결제 공통)

부가서비스 연동

1) 결제내역조회 API

결제내역, 정산 데이터 조회가 가능하며, 가맹점 정산 대사 작업 시 활용하실 수 있습니다.

2) 거래완료 API

가맹점 정산 / 포인트 적립 기준이 "거래완료" 경우 연동이 필요한 API입니다.
API 호출 시 가맹점 정산 처리 및 사용자 포인트 적립이 진행됩니다.

3) 포인트 적립 요청 API

가맹점 "결제일 기준 정산", 포인트 적립 기준이 "수동 적립" 경우 연동이 필요한 API입니다.
API 호출 시 사용자 포인트 적립이 진행됩니다.

4) 현금영수증 발행대상 금액조회 API

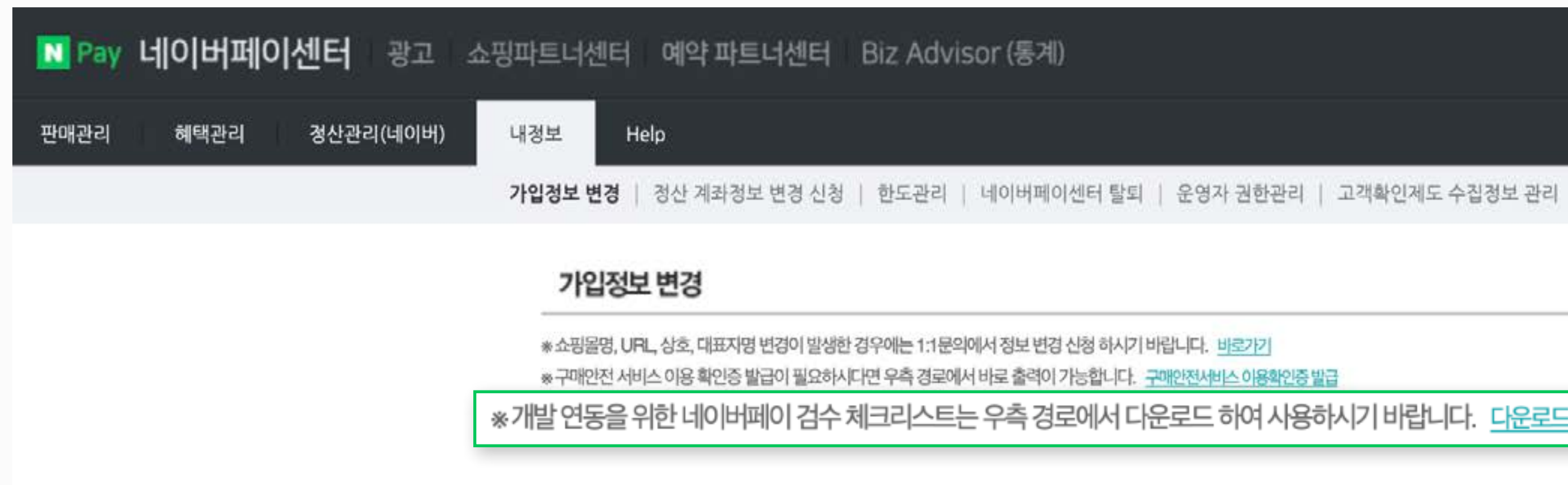
현금영수증을 가맹점에서 직접 발급해야 할 경우 연동이 필요한 API입니다.
네이버페이 결제금액 중 현금영수증 발행대상 금액 조회 용도로 제공되며,
신용카드 결제금액을 제외한 계좌, 네이버페이 포인트(충전성 포인트)가 해당됩니다.

3-1. 네이버페이 체크리스트 사전 점검 안내

네이버페이 API 연동 개발을 마치셨다면 아래 경로에서 연동 체크리스트를 다운로드 받아 개발에 누락된 내용이 없는지 확인하실 수 있습니다.

네이버페이센터(<https://admin.pay.naver.com/>) > 내 정보 > 가입정보 변경

- 간편결제만 제공하는 경우, 간편결제연동 체크리스트 1종 다운로드 후 확인
- 간편결제 + 정기결제를 제공하는 경우, 간편결제 연동 체크리스트 + 정기결제 연동 체크리스트 2종 다운로드 후 확인



3-1. 네이버페이 체크리스트 사전 점검 안내

체크리스트 항목에는 API 호출, 네이버페이 화면, 결제로직 취약성, BI 사용 가이드 정상 여부를 종합적으로 검토해 주셔야 합니다.

체크리스트 각 시트에 가맹점에서 개발 및 테스트 시 확인하신 내용을 작성해주시면 네이버페이 연동 검수 시점에 참고 자료로 활용하고 있습니다.

시트 구분	설명
검수정보	검수 신청을 위한 필수 입력 항목 작성해 주신 검수 환경, API 연동 현황, 사전준비 항목 기준 검수 진행
간편/정기결제 API	API 플랫폼 연동에 대한 체크리스트 확인 및 결과 입력
결제/등록화면	네이버페이 결제(등록)화면 연동에 대한 체크리스트 확인 및 결과 입력
결제로직 취약성	보안가이드에 대한 체크리스트 확인 및 결과 입력

3-2. 검수 체크리스트 항목

네이버페이 체크리스트 항목을 기반으로 연동 검수를 진행합니다.

카테고리	검수 항목
API 연동	<ul style="list-style-type: none">• 결제 연동 방식과 API 요청 시 호출 방식• 입점 기준에 맞춘 연동 대상 API 확인• 호출 시점 및 요청 파라미터 수정 가이드• 결제수단 별 정상 결제, 취소 여부
결제/등록 화면	<ul style="list-style-type: none">• 다양한 사용자 환경에 결제/등록 화면이 정상적으로 노출되는지 구현 형태 확인• 실패 응답에 대한 사용자 안내• 네이버페이 결제 제휴 카드, 계좌 등록• 가맹점 앱에서 네이버페이 인터페이스 가림 현상 없는지 확인• 네이버페이 디자인 사용가이드에 따라 공식 로고, 표기 규정 등이 부합하고 있는지 확인• 가맹점 주문 ↔ 네이버페이 결제 상태 매칭
결제로직 취약성	<ul style="list-style-type: none">• 결제금액/수량 변조 유무를 체크하는지 확인• 최종 결제 결과 안내 전 검증 여부 확인

3-3. 검수 신청 방법 및 소요 기간

검수 신청 방법

체크리스트 기입하신 후 아래 메일로 검수 요청하여 주시기 바랍니다.

dl_npay_techsupport@navercorp.com

소요 기간

작성하신 체크리스트 내용 및 가맹점 검수 환경 세팅을 확인하여 이상 없을 시 네이버페이 연동 검수를 시작하며, 전체 검수 완료까지는 평균 15영업일 소요됩니다.

다만 가맹점 사정에 따라 검수 기간에 차이가 발생할 수 있으므로, 오픈 일자는 검수 완료 후 확정하여 주시기 바랍니다. 호스팅사 통한 연동은 검수 절차 없이 후속 심사 완료 후 오픈 가능합니다. 평균 1주 소요됩니다.

E.O.D

기타 궁금한 점은 네이버페이 가맹점 고객센터 톡톡이나
1:1 문의하기로 연락 주시면 친절히 답변 드리겠습니다.

