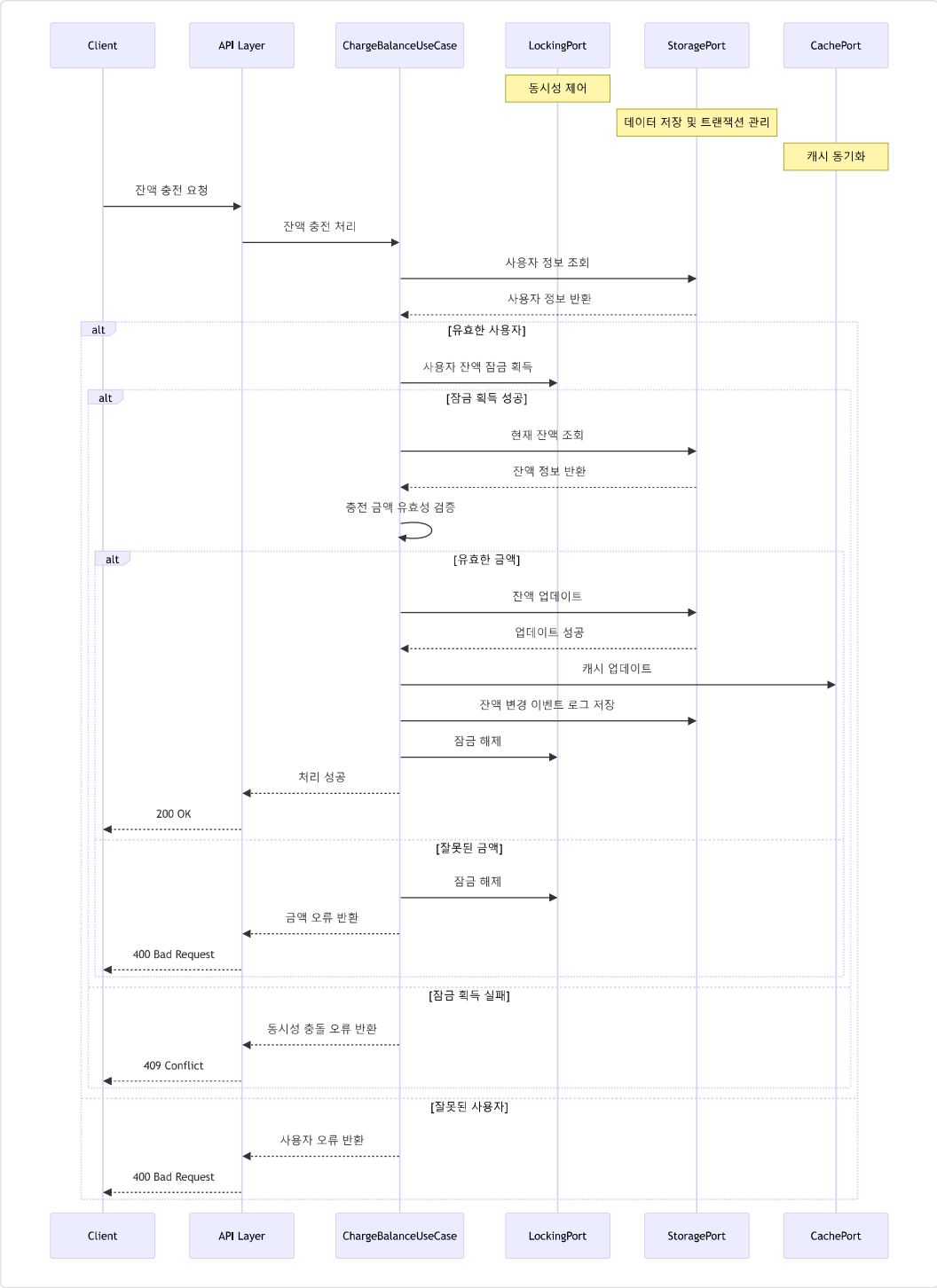


E-Commerce API 시퀀스 다이어그램

1. 잔액 관리 API

1.1 잔액 충전 (POST /balance/charge)

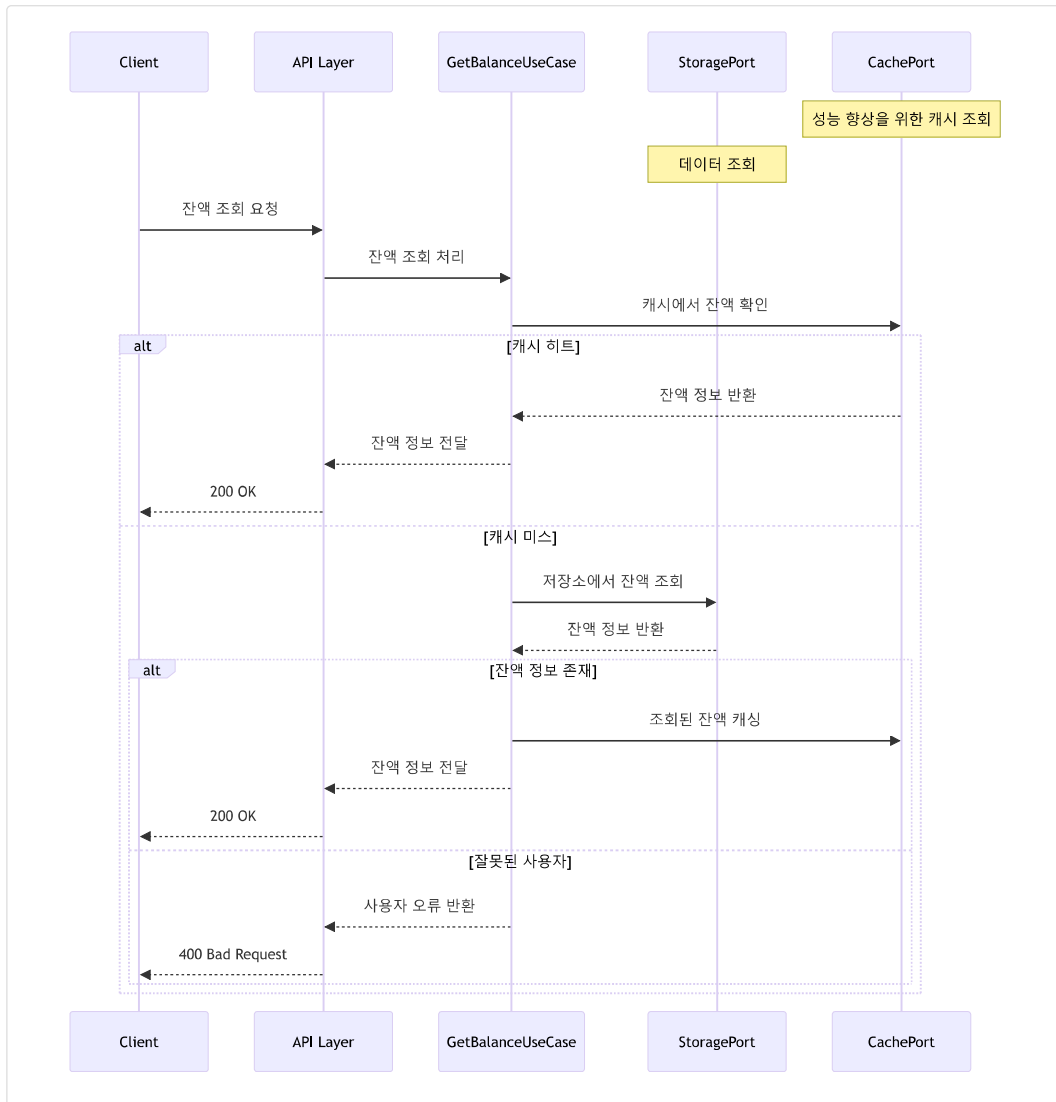


설명: 사용자 ID와 충전 금액을 받아 잔액을 업데이트함. 'LockingPort'로 동시성 제어, 'StoragePort'로 잔액 업데이트 및 이벤트 로그 저장, 'CachePort'로 캐시 동기화. 멍등성은 'IdempotencyKey'로 처리.

예외:

- ERR_BALANCE_INVALID_USER (400): 유효하지 않은 사용자 ID.
- ERR_BALANCE_INVALID_AMOUNT (400): 충전 금액 범위 초과.
- ERR_BALANCE_CONCURRENCY_CONFLICT (409): 동시 충전 요청 충돌.

1.2 잔액 조회 (GET /balance/{userId})



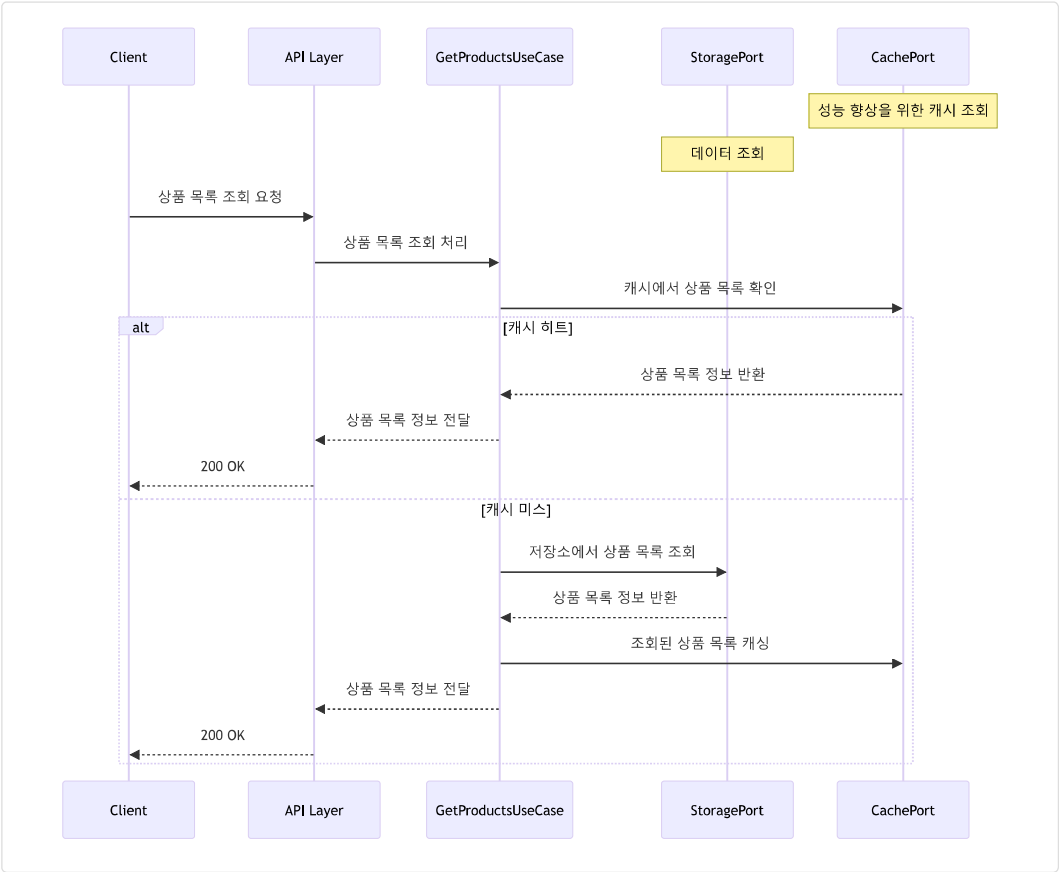
설명: 사용자 ID로 잔액을 조회함. 'CachePort'로 캐시 조회, 캐시 미스 시 'StoragePort'에서 데이터 조회.

예외:

- ERR_BALANCE_INVALID_USER (400): 유효하지 않은 사용자 ID.

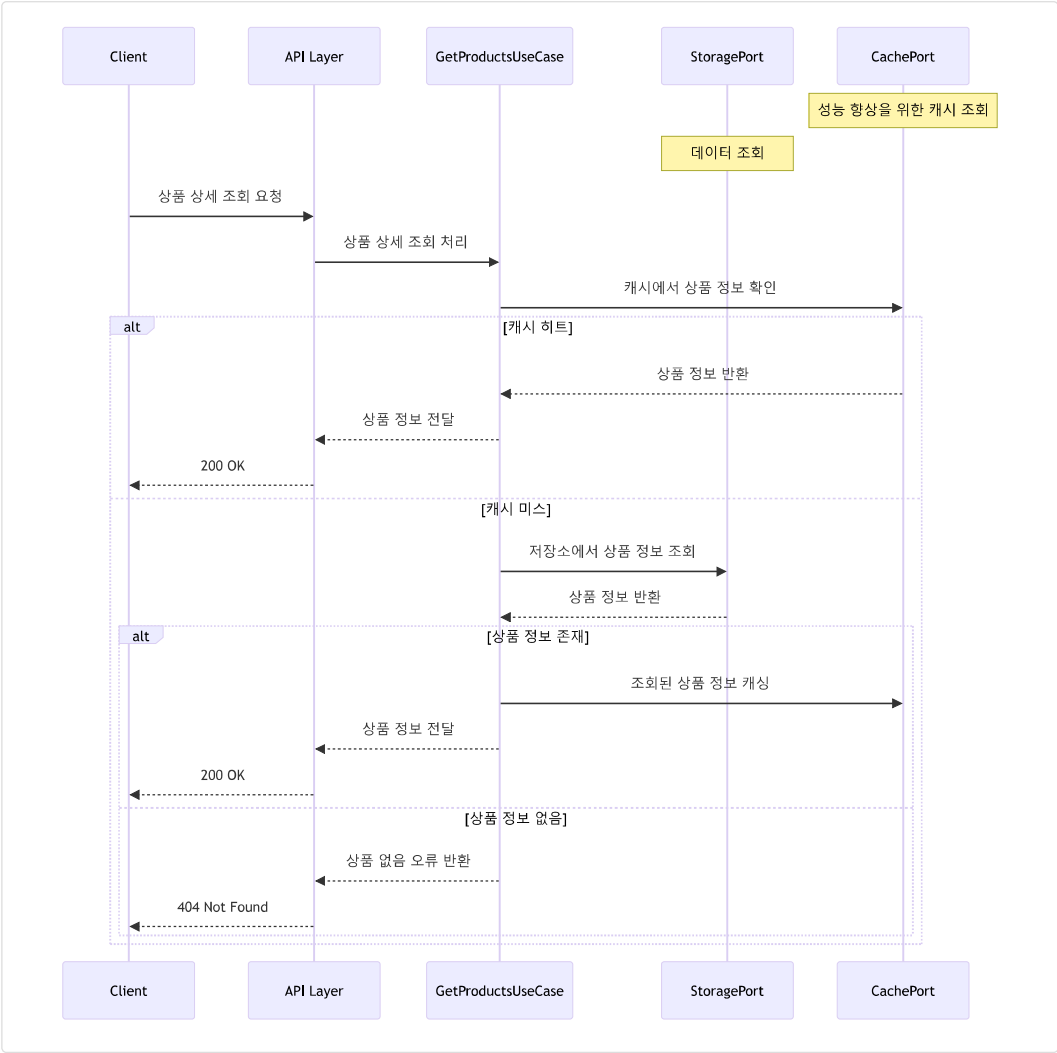
2. 상품 관련 API

2.1 상품 목록 조회 (GET /product/list)



설명: 상품 목록을 조회함. 'CachePort'로 캐시 조회, 캐시 미스 시 'StoragePort'에서 데이터 조회.

2.2 단일 상품 상세 조회 (GET /product/{productId})

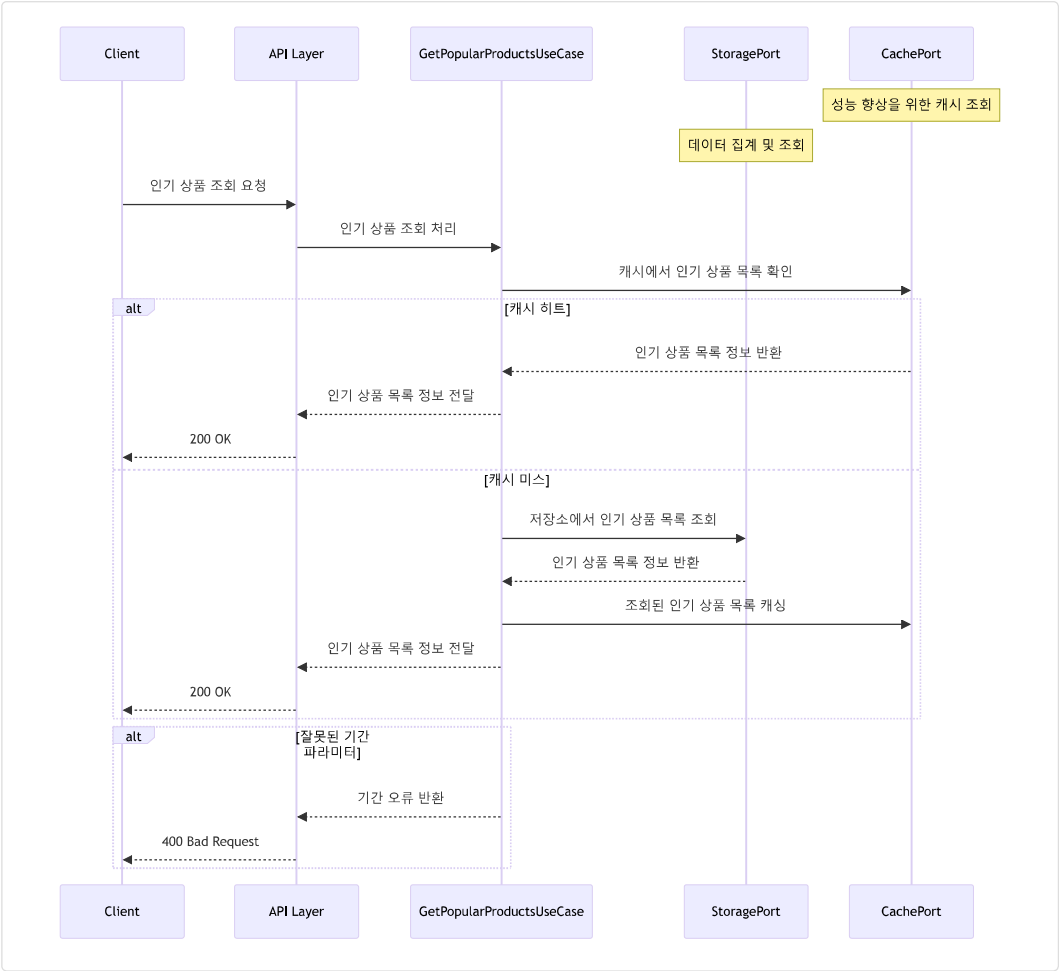


설명: 단일 상품의 상세 정보를 조회함. 'CachePort'로 캐시 조회, 캐시 미스 시 'StoragePort'에서 데이터 조회.

예외:

- ERR_PRODUCT_NOT_FOUND (404): 존재하지 않는 상품 ID.

2.3 인기 판매 상품 조회 (GET /product/popular)



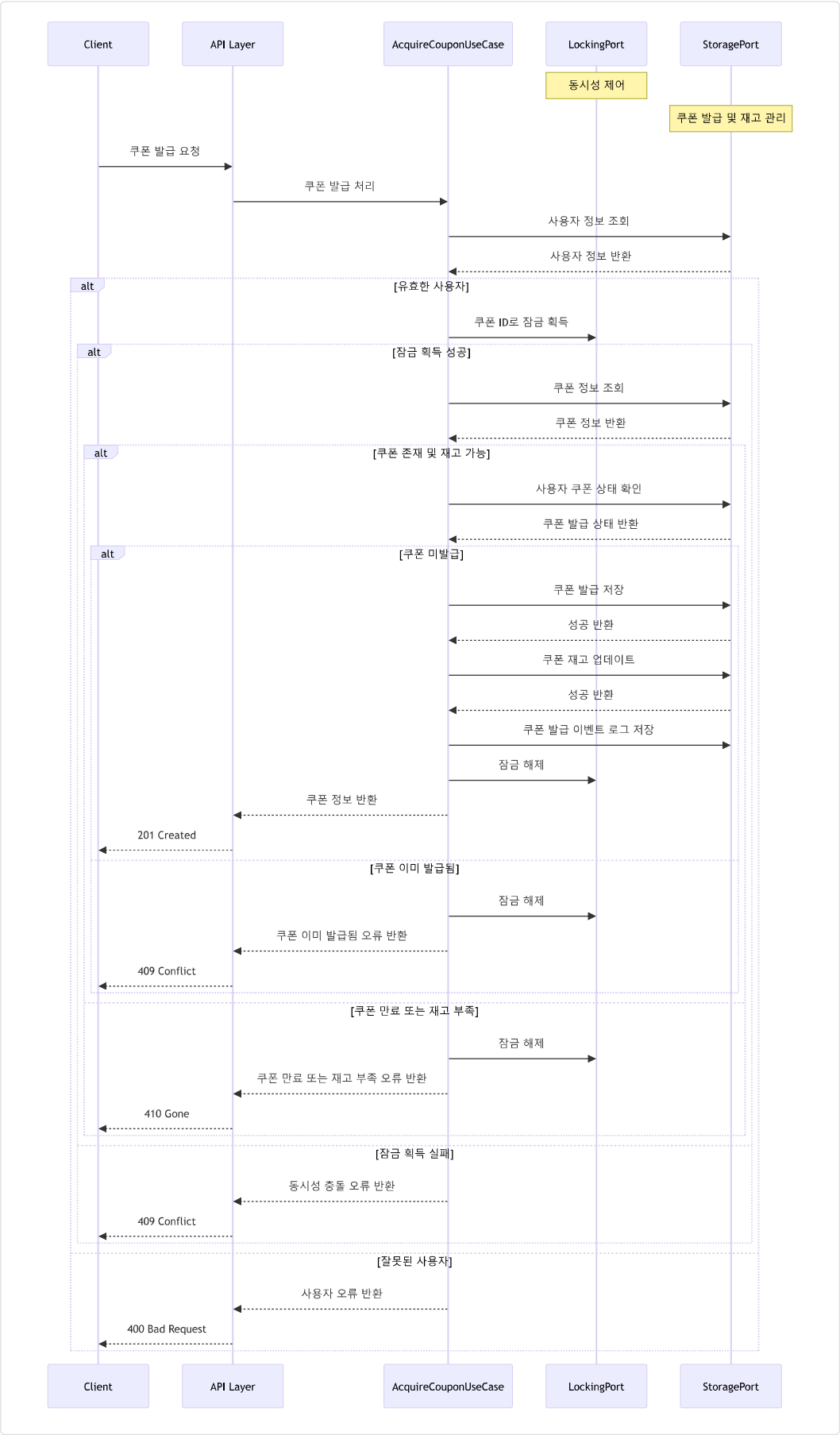
설명: 최근 3일간 상위 5개 상품을 조회함. `CachePort`로 캐시 조회, 캐시 미스 시 `StoragePort`에서 주문 데이터를 집계.

예외:

- `ERR_POPULAR_INVALID_FILTER (400)`: 잘못된 기간 파라미터.

3. 쿠폰 관련 API

3.1 쿠폰 발급 요청 (POST /coupon/acquire)

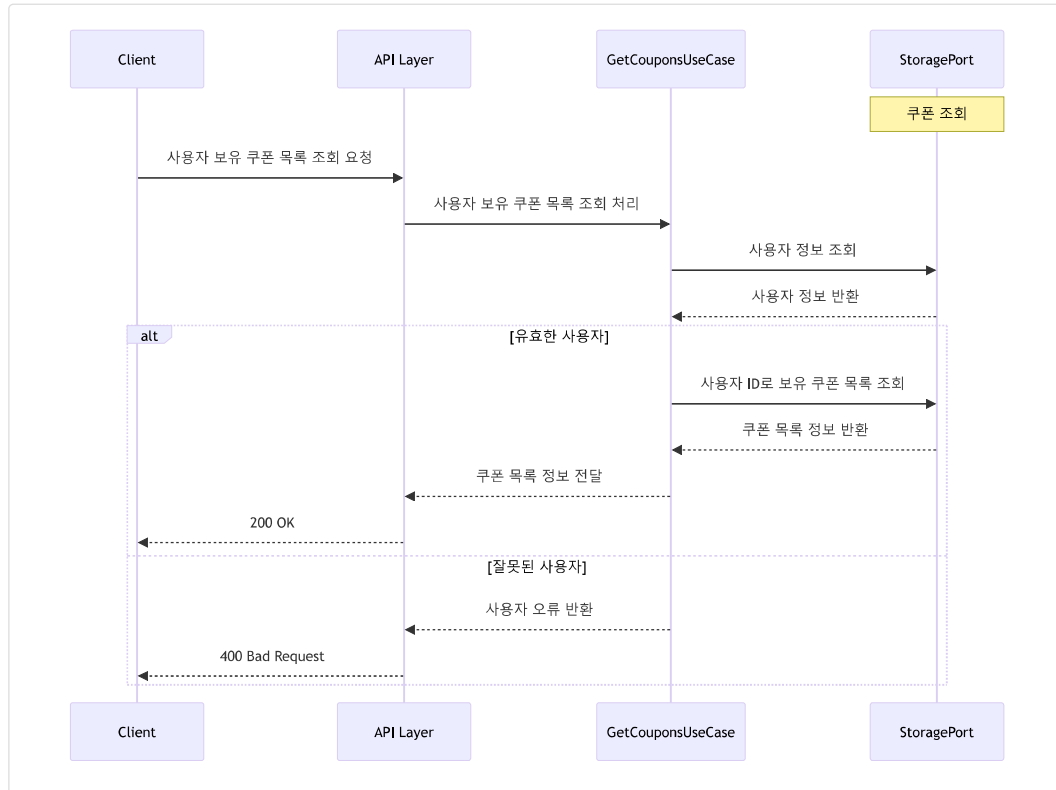


설명: 사용자 ID와 쿠폰 ID를 받아 선택순 쿠폰을 발급함. 'LockingPort'로 동시성 제어, 'StoragePort'로 쿠폰 발급 및 재고 관리. 멍등성은 'IdempotencyKey'로 처리.

예외:

- ERR_BALANCE_INVALID_USER (400): 유효하지 않은 사용자 ID.
- ERR_COUPON_EXPIRED (410): 쿠폰 만료.
- ERR_COUPON_OUT_OF_STOCK (410): 쿠폰 재고 소진.
- ERR_COUPON_ALREADY_ACQUIRED (409): 사용자 이미 쿠폰 발급받음.
- ERR_COUPON_CONCURRENCY_CONFLICT (409): 동시 발급 요청 충돌.

3.2 보유 쿠폰 조회 (GET /coupon/list/{userId})

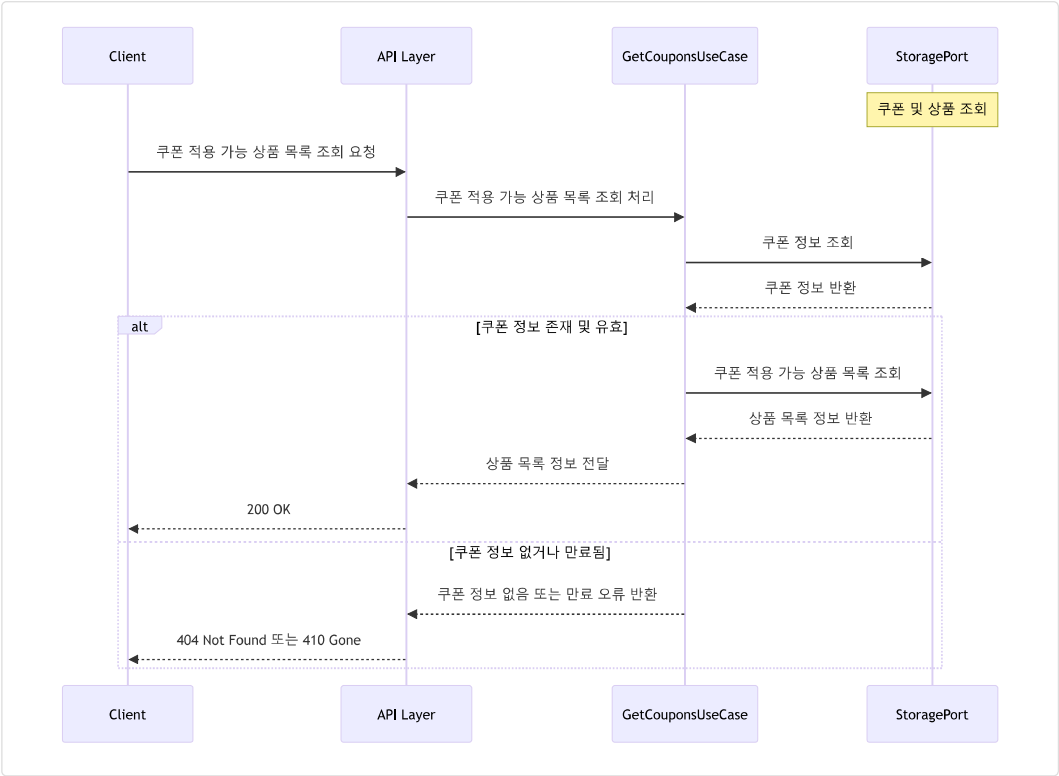


설명: 사용자 ID로 보유한 유효 쿠폰 목록을 조회함. 'StoragePort'에서 데이터 조회.

예외:

- ERR_BALANCE_INVALID_USER (400): 유효하지 않은 사용자 ID.

3.3 쿠폰 적용 가능 상품 조회 (GET /coupon/{couponId}/applicable)



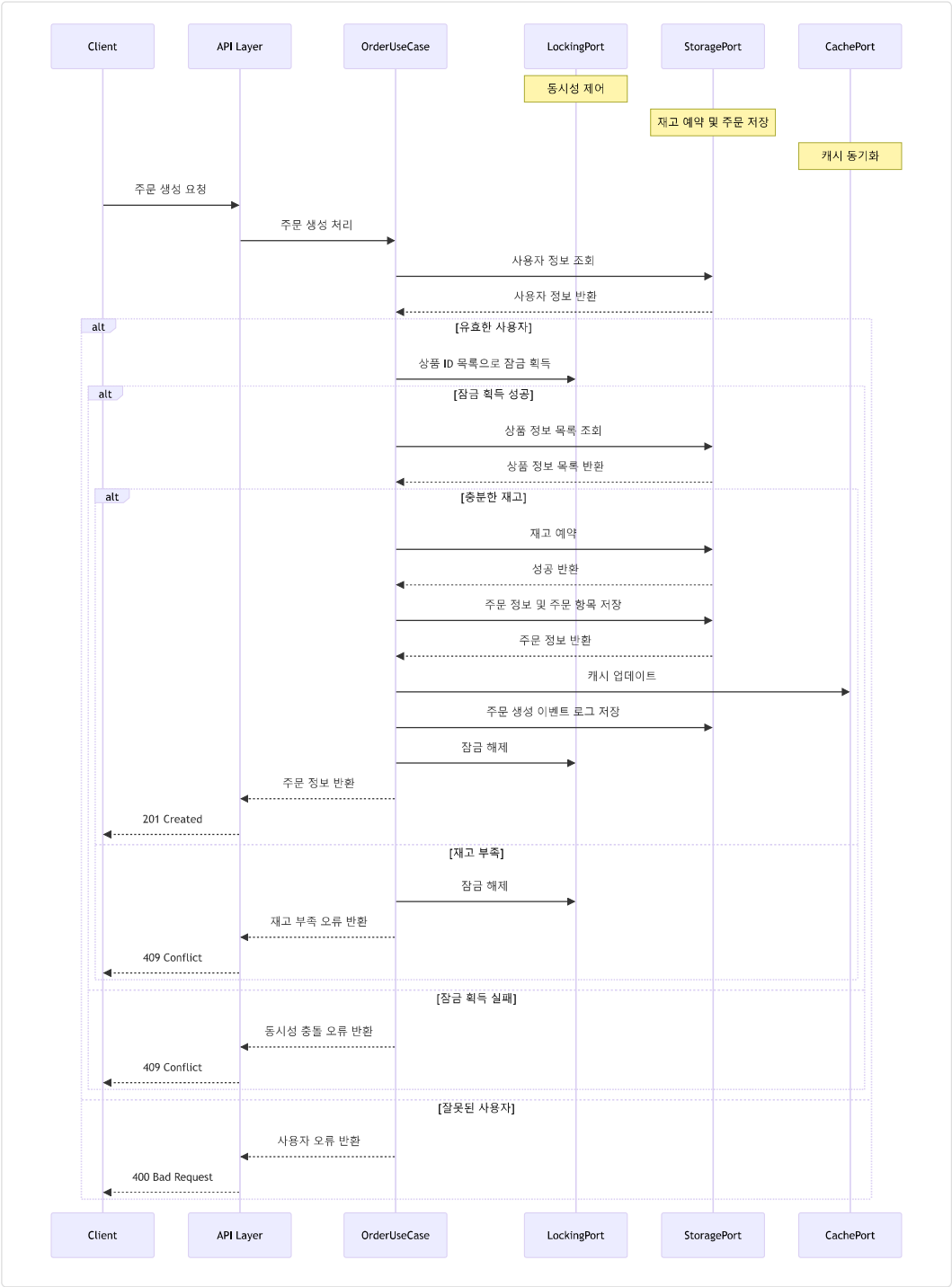
설명: 쿠폰 ID로 적용 가능한 상품 목록을 조회함. 'StoragePort'에서 쿠폰 및 상품 데이터 조회.

예외:

- ERR_COUPON_NOT_FOUND (404): 존재하지 않는 쿠폰 ID.
- ERR_COUPON_EXPIRED (410): 쿠폰 만료.

4. 주문 및 결제 API

4.1 주문 생성 (POST /order)

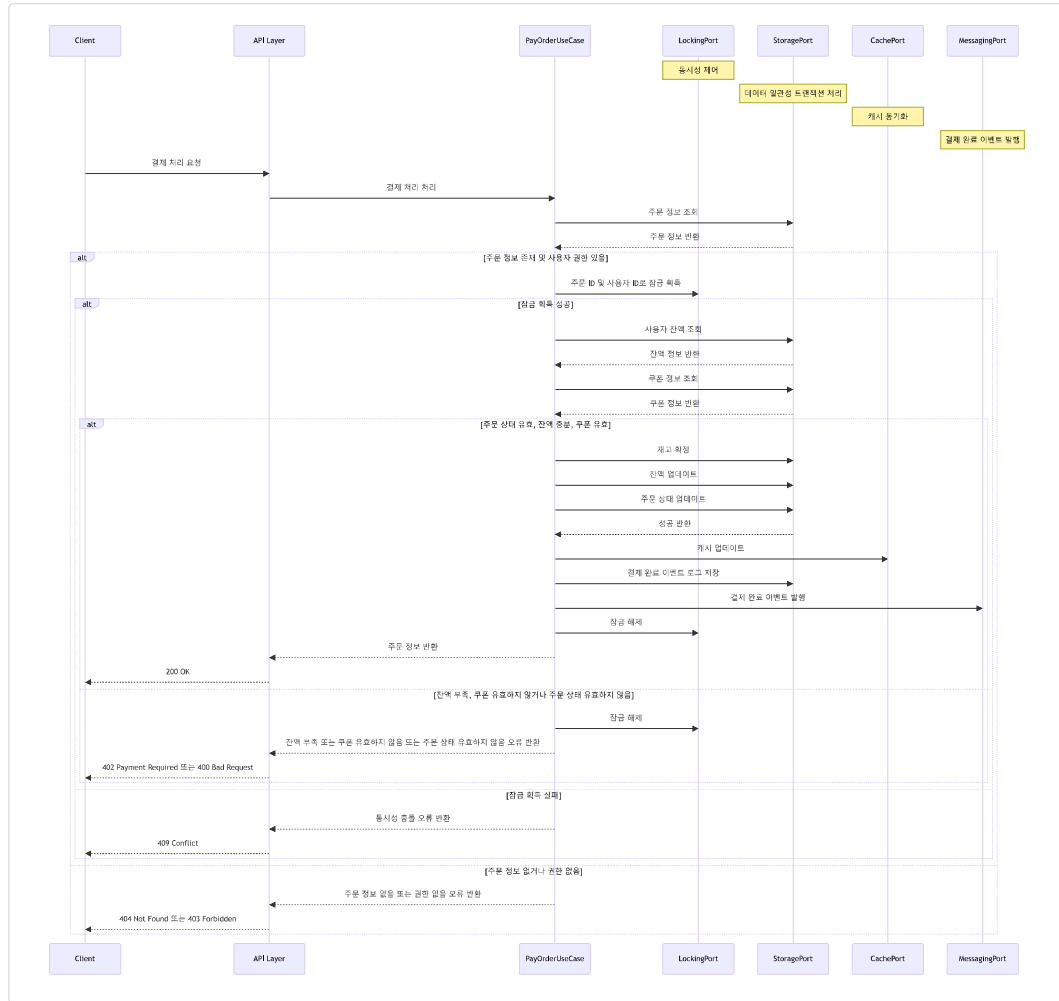


설명: 사용자 ID와 상품 목록(상품 ID, 수량)을 받아 주문을 생성하고, 재고를 임시 예약('reservedStock' 증가)함. 'LockingPort'로 상품 별 동시성 제어, 'StoragePort'로 주문 저장 및 재고 예약, 'CachePort'로 캐시 동기화. 멍등성은 'IdempotencyKey'로 처리.

예외:

- ERR_ORDER_INVALID_USER (400): 유효하지 않은 사용자 ID.
- ERR_ORDER_OUT_OF_STOCK (409): 재고 부족.
- ERR_ORDER_CONCURRENCY_CONFLICT (409): 동시성 충돌.

4.2 결제 처리 (POST /order/{orderId}/pay)

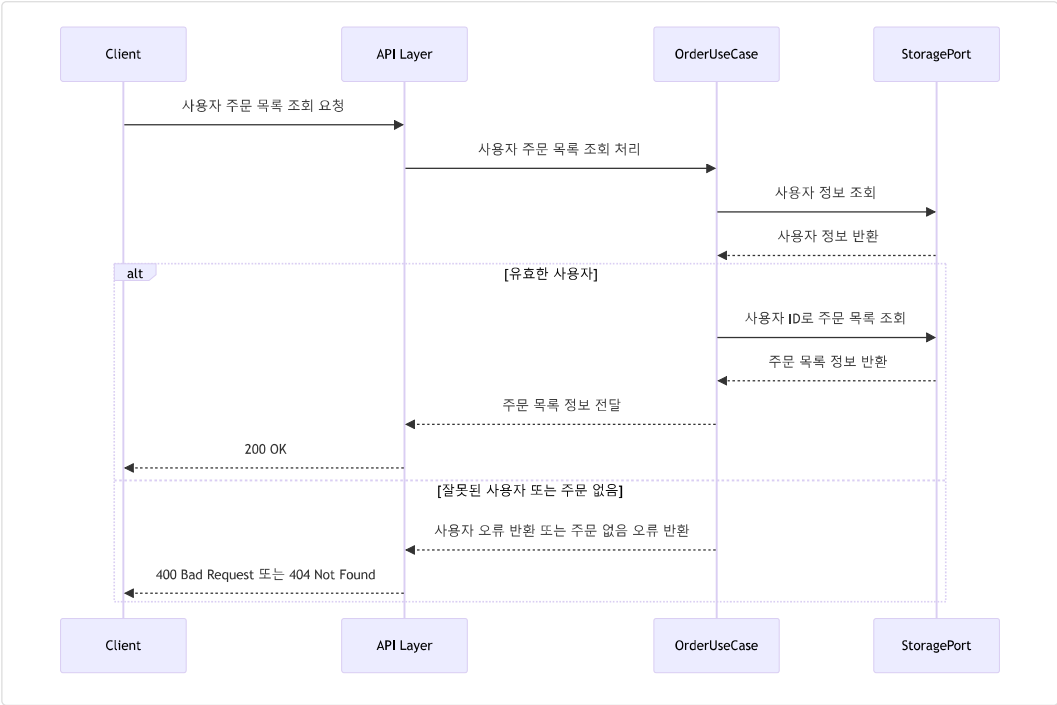


설명: 주문 ID, 사용자 ID, 쿠폰 ID를 받아 결제를 처리함. 'LockingPort'로 주문 및 사용자별 동시성 제어, 'StoragePort'로 잔액 차감, 재고 확정('reservedStock' 감소, 'stock' 감소), 주문 상태 업데이트, 'CachePort'로 캐시 동기화, 'MessagingPort'로 결제 완료 이벤트를 외부 시스템에 전송. 멍등성은 'IdempotencyKey'로 처리.

예외:

- ERR_PAYMENT_INSUFFICIENT_BALANCE (402): 잔액 부족.
- ERR_PAYMENT_INVALID_COUPON (400): 유효하지 않은 쿠폰 ID.
- ERR_PAYMENT_INVALID_ORDER_STATUS (400): 결제 불가 주문 상태.
- ERR_PAYMENT_ORDER_NOT_FOUND (404): 주문이 존재하지 않음.
- ERR_ORDER_UNAUTHORIZED (403): 주문에 대한 접근 권한 없음.
- ERR_PAYMENT_CONCURRENCY_CONFLICT (409): 동시성 충돌.

4.3 주문 목록 조회 (GET /user/{userId}/order/list)

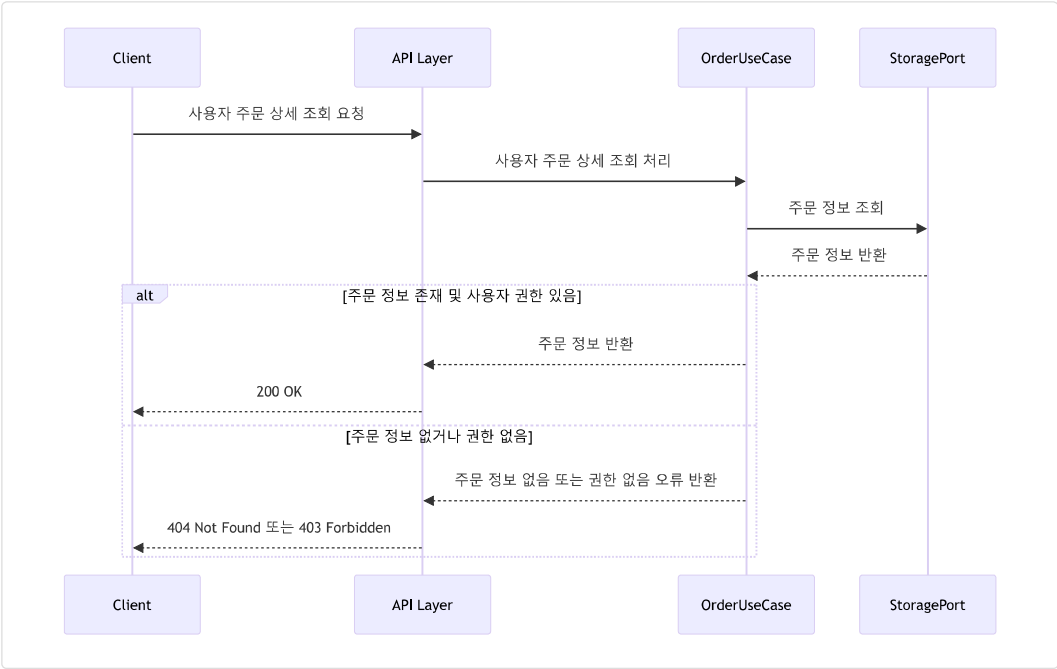


설명: 사용자 ID로 주문 목록을 조회함. 'StoragePort'에서 데이터를 가져옴.

예외:

- ERR_ORDER_INVALID_USER (400): 유효하지 않은 사용자.

4.4 단일 주문 상세 조회 (GET /user/{userId}/order/{orderId})



설명: 특정 주문의 상세 정보를 조회함. 'StoragePort'에서 주문 및 관련 항목을 가져오며, 사용자 권한 확인 포함.

예외:

- ERR_ORDER_NOT_FOUND (404): 주문이 존재하지 않음.
- ERR_ORDER_UNAUTHORIZED (403): 주문이 사용자 소유가 아님.