

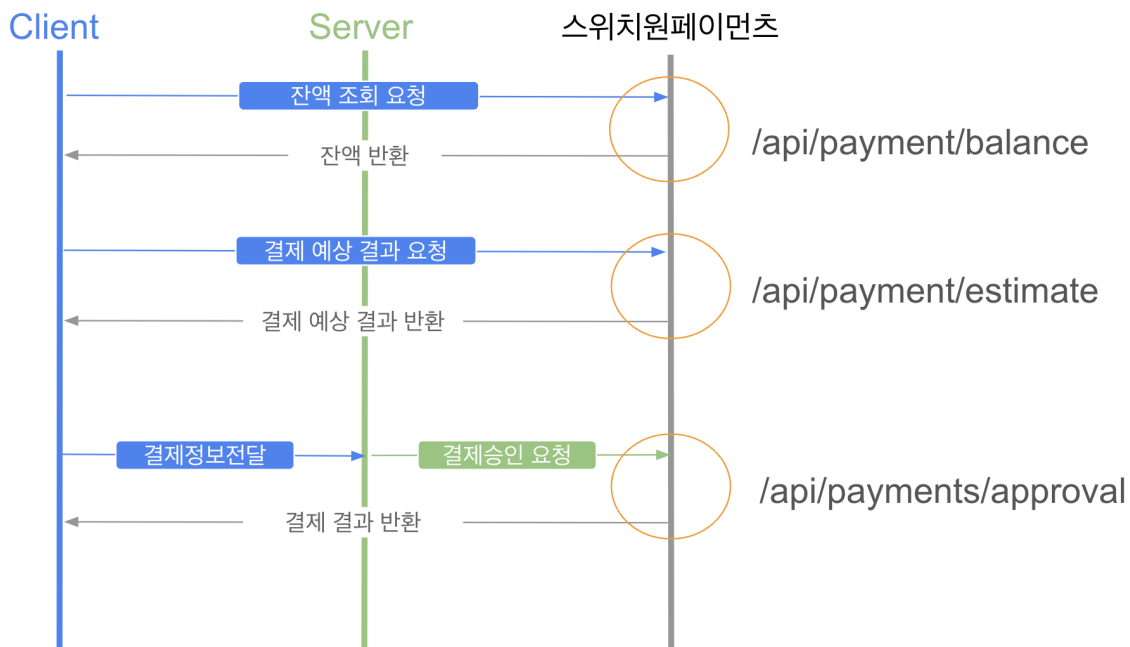
스위치원 백엔드 개발자 채용 과제: 간단한 결제 시스템 백엔드 구현

목적

이 과제의 목적은 간단한 결제 시스템의 백엔드 구현을 통해 지원자의 **API** 개발, 데이터 모델링, 비즈니스 로직 처리 및 테스트 작성 능력을 평가하는 것입니다.

요구사항

- **API 구현:** 아래 정의된 세 개의 **RESTful API** 엔드포인트를 구현하세요.



1. 잔액 조회: /api/payment/balance
2. 결제 예상 결과 조회: /api/payment/estimate
3. 결제 승인 요청: /api/payments/approval

- **Request & Response:** 각 API의 요청과 응답 형식은 아래에 명시된 대로 구현해야 합니다.
- **데이터베이스 연동:** 이 시스템은 사용자와 결제 정보를 저장할 수 있는 데이터베이스(H2)와 연동되어야 합니다. 관련 스키마 설계 및 구현이 필요합니다.
- **단위 테스트:** 각 API에 대한 단위 테스트를 작성하여 기능이 올바르게 동작하는지 검증하세요.

API 상세

1. 잔액 조회

Endpoint	GET /api/payment/balance/{userId}
Request	{}
Response	{ "userId": "12345", "balance": 1000.00, "currency": "USD" }

2. 결제 예상 결과 조회

Endpoint	POST /api/payment/estimate
Request	{ "amount": 150.00, "currency": "USD", "destination": "merchantId123", "userId": "12345" }
Response	{ "estimatedTotal": 155.00, "fees": 5.00, "currency": "USD" }

3. 결제 승인 요청

Endpoint	POST /api/payments/approval
Request	{ "userId": "12345", "amount": 200.00, "currency": "USD", }

	<pre>"merchantId": "merchantId123", "paymentMethod": "creditCard", "paymentDetails": { "cardNumber": "1234-5678-9123-4567", "expiryDate": "12/24", "cvv": "123" } }</pre>
Response	<pre>{ "paymentId": "paymentId12345", "status": "approved", "amount": 200.00, "currency": "USD", "timestamp": "2023-01-23T10:00:00Z" }</pre>

평가 기준

- 코드 품질: 가독성, 유지보수 용이성, 코딩 규약 준수
- 데이터 모델링: 데이터베이스 설계의 적절성 및 효율성
- 로직 구현: 비즈니스 요구사항에 대한 정확한 이해와 구현
- 테스트 케이스: 충분한 커버리지와 적절한 테스트 작성

제출 방법

- 소스 코드를 **GitHub** 리포지토리에 업로드하고, 해당 링크를 제출하세요.

제출 기한

과제는 일주일 내에 완료하여 제출해야 합니다.

이 과제를 통해 지원자의 종합적인 백엔드 개발 능력을 평가하고자 합니다. 성공적인 결과를 기대합니다!