

# 자기소개서

## 의미 있는 프로젝트나 활동

### "보안과 안정성, 두 마리 토끼를 잡은 핀테크 앱 고도화 프로젝트"

가장 의미 있었던 경험은 최근 3년 9개월간 재직한 핀테크 스타트업에서 해외 송금 앱의 보안 취약점 점검 대응 및 아키텍처 고도화를 주도했던 프로젝트입니다.

금융 서비스의 특성상 사용자의 자산을 보호하는 것이 최우선 과제였습니다. 2023년과 2025년, 두 차례에 걸쳐 외부 보안원의 취약점 점검을 수행하며 앱의 보안 수준을 한 단계 끌어올려야 했습니다. 저는 보안 솔루션을 적용하고 코드를 수정하는 실무를 전담하며, 단순히 취약점을 막는 것을 넘어 앱의 전반적인 구조를 되돌아보는 계기로 삼았습니다.

이 과정에서 기존의 레거시 코드가 유지보수와 보안 대응에 걸림돌이 된다는 것을 파악하고, 팀 내 협업을 통해 클린 아키텍처(Clean Architecture) 도입과 모듈화 작업을 병행했습니다. 결과적으로 보안 점검을 완벽하게 통과하여 서비스의 신뢰도를 높였을 뿐만 아니라, 코드의 결합도를 낮춰 이후 신규 기능 개발 속도를 높이는 유의미한 변화를 끌어낼 수 있었습니다. 이 경험을 통해 '안정적인 서비스'란 견고한 코드 베이스 위에서 만들어진다는 확신을 갖게 되었습니다.

## 효율성 개선 경험

### "런타임 에러 제로에 도전하다: Koin에서 Hilt로의 마이그레이션"

반복적인 런타임 이슈를 근본적으로 해결하기 위해 의존성 주입(DI) 라이브러리를 교체하여 개발 효율성을 개선한 경험이 있습니다.

기존 프로젝트는 Koin을 사용하고 있었는데, 컴파일 타임에 에러를 잡지 못해 앱 실행 중에 크래시가 발생하는(Runtime Error) 문제가 간헐적으로 발생했습니다. 이는 개발자가 실수를 인지하는 시점을 늦추고, QA 비용을 증가시키는 비효율적인 원인이었습니다.

저는 이를 해결하기 위해 팀에 컴파일 타임에 의존성 검증이 가능한 Hilt로의 전환을 제안했습니다. 이미 운영 중인 서비스의 핵심 라이브러리를 교체하는 것은 리스크가 큰 작업이었지만, 점진적인 마이그레이션 전략을 수립하여 진행했습니다.

마이그레이션 완료 후, 의존성 관련 에러를 빌드 시점에 사전에 차단할 수 있게 되어 앱의 안정성(Stability)이 크게 향상되었습니다. 또한, 안드로이드 표준 라이브러리인 Jetpack 컴포넌트와의 호환성이 좋아져 보일러플레이트 코드가 줄어들었고, 결과적으로 팀 전체의 개발 생산성을 높이는 성과를 거두었습니다.

## 문제 해결 경험

## "사용자 이탈을 막는 UX 설계: 환율 변동 에러 대응"

업무 중 마주친 가장 난감했던 문제는 해외 송금 신청 과정에서 실시간 환율 변동으로 인해 발생하는 서버 에러였습니다.

송금 신청 도중 환율이 급격히 변하면 서버에서 에러를 반환했는데, 기존 로직은 단순히 "에러가 발생했습니다"라는 메시지만 띄우고 프로세스를 중단시켰습니다. 이는 사용자의 잘못이 아님에도 불구하고 처음부터 다시 송금을 진행해야 하는 불편을 초래했고, 이탈률을 높이는 원인이었습니다.

저는 이 문제를 해결하기 위해 에러 발생 시 플로우를 전면 재설계했습니다. 단순히 에러를 띄우는 대신, "환율이 변동되었습니다. 변경된 환율로 계속 진행하시겠습니까?"와 같이 사용자가 상황을 인지하고 다음 액션을 선택할 수 있도록 유도하는 기능을 개발했습니다.

서버 에러라는 기술적인 문제를 사용자 경험(UX) 관점에서 재해석하여 해결책을 제시했고, 그 결과 송금 실패율을 낮추고 고객 불만을 줄이는 데 기여할 수 있었습니다. 이는 기술적 구현만큼이나 사용자의 맥락을 이해하는 것이 중요함을 깨달은 계기였습니다.

## 인간관계 경험

### "개발자의 언어가 아닌, 기획자의 언어로 소통하기"

다양한 부서와 협업하며 비즈니스 요구사항을 기술로 실현하는 과정에서 소통의 중요성을 깊이 배웠습니다

한번은 마케팅 팀에서 고객 유입 경로를 분석하고 싶다는 요청이 들어왔습니다. 초기 요구사항은 추상적이었지만, 저는 개발자의 관점에서 "안 됩니다"라고 말하기보다 "어떤 데이터가 필요한지"를 먼저 물었습니다. 마케팅 팀의 니즈가 '이벤트별 가입 전환율'을 보는 것임을 파악한 후, 회원가입 시 가입 경로를 선택하는 페이지를 신설하고, 해당 데이터를 이벤트 로그로 남겨 정확한 지표를 추적할 수 있도록 개발했습니다.

또한, CS 팀으로부터 특정 국가 송금 시 수취인 정보 수정이 불편하다는 피드백을 들었을 때도, 이를 단순 민원으로 넘기지 않고 즉시 개선 작업에 착수했습니다.

이처럼 타 부서의 고충과 목표를 경청하고 이를 기술적으로 해결해 주었을 때, 동료들로부터 "함께 일하고 싶은 개발자"라는 신뢰를 얻을 수 있었습니다. 이러한 협업 경험은 제가 기술적 고집보다는 팀의 목표를 최우선으로 하는 개발자로 성장하는 밑거름이 되었습니다.