포트폴리오

자기 소개

Summary

저는 개발자로, 빠르고 안정적인 서비스를 만들고 운영하는 것을 좋아합니다. 제품 아이디어를 기획하거나 개발하고 운영하는 것 모두 즐기는 편이고, 한 사람이 전체를 맡아야 하는 환경에서도 잘 굴러가도록 만들어 왔습니다. 지금은 주로 Go 기반 백엔드 개발하면서, 클라우드 인프라와 데이터 파이프라인 쪽에 관심을 두고 있습니다.

지금까지 가장 자신 있게 말할 수 있는 것들: 실제로 돌아가는 프로덕션 시스템 만들기, 비용 최적화, 장애 대응, 팀 없이 혼자 끝까지 가는 것. 아직 부족한 것: 대규모 분산 시스템 설계, 보안 전문성, 프론트엔드 깊이. 이 둘은 앞으로 채우고 싶은 영역입니다.

GitHub에 공개된 활동이 거의 없는 것을 보셨다면 일부러 그러한 것입니다. 사내 코드베이스, 개인 서버, 비공개 레포에서 대부분 일합니다.

연락: hello@example.com · linkedin.com/in/sample · github.com/sample-handle · 한국 GMT+9 · 주말 온라인 협업 가능

기술 스택

경력 사항

Company	Entry Date	Leave Date	Years (Position)	Form
네온랩	2021-01	재직중	5년 0개월 (팀 리드)	정규
아키텍	2017-07	2019-12	2년 6개월 (개발자)	정규
하이퍼랩	2016-03	2017-06	1년 4개월 (개발자)	계약
블루웨일	2014-09	2016-02	1년 6개월 (개발자)	정규
스타트업 A	2013-11	2014-08	0년 10개월 (개발자)	정규
프리랜서	2012-06	2013-10	1년 5개월 (개발자)	계약
대학원 조교	2010-09	2012-05	1년 9개월 (조교)	파트타임
대학교	2007-03	2010-08	3년 6개월 (학생)	정규

비밀번호

팀/분기 서비스 관리 및 배포

형식

- 위치: 1Password 볼트 "Ops"
- 구조: 볼트당 1개 항목
- 항목 이름: 서비스 이름 (예: "Auth-API", "Billing-Worker")
- 필드: 이름 + 값

보관 기간

- 활성: 소속 기간 동안
- 퇴사 후 90일: 읽기 전용
- 퇴사 후 180일: 볼트 삭제

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한
- 외부 협력자: 임시 볼트 (Vault-Ext-{Company})

회전 주기

- 정기: 분기마다 (90일)
- 즉시: 퇴사자 발생 시, 유출 의심 시, 공급자 사고 시

사용 도구

- 1Password CLI (`op`) — 스크립트 및 CI 주입용
- 1Password 웹 볼트 — 일반 사용

회전 절차

1. 새 값 생성
2. 1Password 항목 업데이트
3. 의존 서비스 재배포
4. 24시간 모니터링
5. 이전 값 폐기

사고 대응

- 의심되는 항목 즉시 회전
- 감사 로그 검토: 1Password 활동 로그
- 영향 범위 파악: 어떤 서비스가 그 자격 증명을 사용하는지

감사

- 분기마다 볼트 접근 권한 검토
- 퇴사자 체크리스트: 볼트 접근 권한 제거

비용

월간 서버 스냅샷

형식

- 위치: 1Password 볼트 "Ops"
- 항목 이름: `Snapshot-YYYY-MM`
- 필드: 서비스 이름 + 월간 비용 (USD)

보관 기간

- 활성: 1년
- 1년 이후: 삭제

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- 1Password CLI (`op`)
- 1Password 웹 볼트

절차

1. 매월 1일 AWS 콘솔 접속
2. 전월 청구 내역 확인
3. 서비스별 합계 기록
4. 1Password 항목 생성
5. 전월 대비 10% 이상 변동 시 원인 메모

사용 방식

- 분기 검토 시 추세 확인
- 예산 초과 시 알림 발송

인증서

서버/도메인 TLS 스크립트

형식

- 위치: 1Password 볼트 "Ops"
- 항목 이름: `Cert-{domain}`
- 필드: 인증서 (PEM), 개인 키, 만료 일

보관 기간

- 활성: 만료 후 30일
- 30일 이후: 삭제

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- Let's Encrypt
- 1Password CLI (`op`)
- 1Password 웹 볼트

절차

1. 만료 30일 전 알림 수신
2. 갱신 실행
3. 1Password 항목 업데이트
4. 의존 서비스 재배포
5. 만료 일 확인

사용 방식

- 만료 30일 전 알림 수신
- 갱신 실행
- 1Password 항목 업데이트
- 의존 서비스 재배포
- 만료 일 확인

데이터

데이터베이스 백업 스크립트

형식

- 위치: 1Password 볼트 "Ops"
- 항목 이름: `DB-Backup-{service}`
- 필드: 암호화 키, 보관 기간, 마지막 성공 타임스탬프

보관 기간

- 활성: 서비스 운영 기간
- 퇴사 후 90일: 읽기 전용
- 퇴사 후 180일: 볼트 삭제

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- 1Password CLI (`op`)
- 1Password 웹 볼트

절차

1. 매일 자동 실행 (cron)
2. 암호화 키 확인
3. 마지막 성공 타임스탬프 갱신
4. 보관 기간 경과 백업 삭제

사용 방식

- 매일 자동 실행 (cron)
- 암호화 키 확인
- 마지막 성공 타임스탬프 갱신
- 보관 기간 경과 백업 삭제

복구 방식

- 최신 백업 다운로드
- 암호화 키 복호화
- 대상 데이터베이스 복원
- 무결성 검증

사용 방식

- 최신 백업 다운로드
- 암호화 키 복호화
- 대상 데이터베이스 복원
- 무결성 검증

이벤트

이벤트 리스너 및 파이프라인 처리기용 환경 설정 파일

형식

- 위치: 1Password 볼트 "Ops"
- 항목 이름: `Event-Listener-{name}`
- 필드: 구독 URL, 시크릿, 활성 플래그

보관 기간

- 활성: 구독 활성 기간
- 퇴사 후 90일: 읽기 전용
- 퇴사 후 180일: 볼트 삭제

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- Fly.io
- Fly.io Machine Architecture FSM
- 1Password CLI (`op`)

절차

1. 구독 URL 등록
2. 시크릿 발급
3. 활성 플래그 켜기

사용 방식

- 구독 URL 등록
- 시크릿 발급
- 활성 플래그 켜기

복구 방식

- 구독 URL 등록
- 시크릿 발급
- 활성 플래그 켜기

Apple 개발자용 도장 스토리

형식

- 위치: 팀 계정 1명 관리 볼트
- 항목 이름: `Apple-Dev-{role}`
- 필드: 인증서 (.p12), 프로비저닝 프로파일, App ID, 만료 일

보관 기간

- 활성: 퇴사 후 30일 경과 시 폐기
- 폐기된 인증서: 퇴사 후 1년 보관

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- Fastlane Match
- Fastlane Spaceship
- 1Password CLI (`op`)

절차

1. 인증서 발급
2. 프로비저닝 프로파일 갱신
3. 1Password 항목 업데이트
4. 팀 멤버 기기 재동기화

사용 방식

- 인증서 발급
- 프로비저닝 프로파일 갱신
- 1Password 항목 업데이트
- 팀 멤버 기기 재동기화

Apple 앱 배포 인증서 교체 및 Feature Flag 제거

형식

- 위치: 1Password 볼트 "Ops"
- 항목 이름: `Apple-Cert-{YYYYMMDD}`, `Flag-{name}`
- 필드: 인증서 (.p12), 만료 일, 제거된 flag 이름

보관 기간

- 활성: 교체 후 90일 경과 시 폐기
- 폐기된 인증서: 1년 보관
- 제거된 flag: 영구 보관

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- Fastlane Match
- Fastlane Spaceship
- 1Password CLI (`op`)
- 1Password 웹 볼트

절차

1. 인증서 발급
2. 프로비저닝 프로파일 갱신
3. 1Password 항목 업데이트
4. 팀 멤버 기기 재동기화
5. 만료 일 확인
6. 코드에서 flag 제거
7. 배포 후 24시간 모니터링
8. 이전 인증서 폐기

사용 방식

- 인증서 발급
- 프로비저닝 프로파일 갱신
- 1Password 항목 업데이트
- 팀 멤버 기기 재동기화
- 만료 일 확인
- 코드에서 flag 제거
- 배포 후 24시간 모니터링
- 이전 인증서 폐기

Fumel 스크립트

형식

- 위치: 팀 볼트 "Ops"
- 항목 이름: `Fumel-{name}`
- 필드: 인증서 (.crt), 개인 키 (.key), 만료 일, 구독 URL

보관 기간

- 활성: 퇴사 후 30일 경과 시 폐기
- 폐기된 인증서: 퇴사 후 1년 보관

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- Fly.io
- Fly.io Machine Architecture FSM
- 1Password CLI (`op`)

절차

1. 인증서 발급
2. 개인 키 생성
3. 구독 URL 등록
4. 만료 일 기록

사용 방식

- 인증서 발급
- 개인 키 생성
- 구독 URL 등록
- 만료 일 기록

이벤트 이터레이터 개인 Tuple 저장 및 TTAB 간격 확인

형식

- 위치: 팀 볼트 "Ops"
- 항목 이름: `Tuple-{stream}`
- 필드: 인증서 (.crt), 개인 키 (.key), 만료 일, 구독 URL

보관 기간

- 활성: 퇴사 후 30일 경과 시 폐기
- 폐기된 인증서: 퇴사 후 1년 보관

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- Fly.io
- Fly.io Machine Architecture FSM
- 1Password CLI (`op`)

절차

1. 인증서 발급
2. 개인 키 생성
3. 구독 URL 등록
4. 만료 일 기록

사용 방식

- 인증서 발급
- 개인 키 생성
- 구독 URL 등록
- 만료 일 기록

Analytics 모듈 그래 및 작동

형식

- 위치: Analytics 운영 볼륨 접근 권한 보유 팀 멤버
- 경로: `/var/analytics/{name}.db`
- 모드: 0600, 소유자 `analytics`
- 백업: 없음

보관 기간

- 활성: 퇴사 후 30일 경과 시 폐기
- 폐기된 인증서: 퇴사 후 1년 보관

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- Fly.io
- Fly.io Machine Architecture FSM
- 1Password CLI (`op`)

절차

1. 인증서 발급
2. 개인 키 생성
3. 구독 URL 등록
4. 만료 일 기록

사용 방식

- 인증서 발급
- 개인 키 생성
- 구독 URL 등록
- 만료 일 기록

Analytics 모듈 그래 및 작동

형식

- 위치: Analytics 운영 볼륨 접근 권한 보유 팀 멤버
- 경로: `/var/analytics/{name}.db`
- 모드: 0600, 소유자 `analytics`
- 백업: 없음

보관 기간

- 활성: 퇴사 후 30일 경과 시 폐기
- 폐기된 인증서: 퇴사 후 1년 보관

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- Fly.io
- Fly.io Machine Architecture FSM
- 1Password CLI (`op`)

절차

1. 인증서 발급
2. 개인 키 생성
3. 구독 URL 등록
4. 만료 일 기록

사용 방식

- 인증서 발급
- 개인 키 생성
- 구독 URL 등록
- 만료 일 기록

cocoapods -v SPM 이전

형식

- 위치: 팀 볼트 "Ops"
- 항목 이름: `Xcode-Tuist-{name}` 또는 `Xcode-SPM-{name}`
- 필드: 인증서 (.p12), 만료 일, 팀 ID

보관 기간

- 활성: 퇴사 후 30일 경과 시 폐기
- 폐기된 인증서: 퇴사 후 1년 보관

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- Tuist
- Xcode
- 1Password CLI (`op`)

절차

1. 인증서 발급
2. 만료 일 기록

사용 방식

- 인증서 발급
- 만료 일 기록

앱 다운로드 및 업데이트

형식

- 위치: Apple 팀 계정 관리 볼트 또는 개인 키
- 항목 이름: `Apple-Dev-{app}`
- 필드: 인증서 (.p12), 만료 일, 팀 ID

보관 기간

- 활성: 퇴사 후 30일 경과 시 폐기
- 폐기된 인증서: 퇴사 후 1년 보관

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- Fastlane (Performance, Crashlytics, Bigfoot)
- Xcode

절차

1. Fastlane 실행 (Xcode 빌드 및 업로드)
   - Testing, Debugging, Firebase 등 다양한 실행 옵션 제공
2. 인증서 만료 일 기록

사용 방식

- Fastlane 실행 (Xcode 빌드 및 업로드)
   - Testing, Debugging, Firebase 등 다양한 실행 옵션 제공
- 인증서 만료 일 기록

동기화 엔진

형식

- 위치: 팀 1명 관리 볼트 / 개인 키
- 항목 이름: `Sync-{name}`
- 필드: 인증서 (.crt), 개인 키 (.key), 만료 일

보관 기간

- 활성: 퇴사 후 30일 경과 시 폐기
- 폐기된 인증서: 퇴사 후 1년 보관

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- Fly.io
- Fly.io Machine Architecture FSM
- 1Password CLI (`op`)

절차

1. 인증서 발급
2. 개인 키 생성
3. 만료 일 기록

사용 방식

- 인증서 발급
- 개인 키 생성
- 만료 일 기록

Swagger — Swift 파일 전환 작업용

형식

- 위치: Linux 관리 볼트 / 개인 키
- 항목 이름: `Swagger-{name}`
- 필드: 인증서 (.crt), 개인 키 (.key), 만료 일

보관 기간

- 활성: 퇴사 후 30일 경과 시 폐기
- 폐기된 인증서: 퇴사 후 1년 보관

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- Swift, SwiftNIO, SwiftJWT, Vapor, Hummingbird, Lambda
- Linux

절차

1. 인증서 발급
2. 개인 키 생성
3. 만료 일 기록

사용 방식

- 인증서 발급
- 개인 키 생성
- 만료 일 기록

홈 자동화 서비스

형식

- 위치: 홈 볼트 관리자 / 개인 키
- 항목 이름: `Home-{service}`
- 필드: 인증서 (.crt), 개인 키 (.key), 만료 일

보관 기간

- 활성: 퇴사 후 30일 경과 시 폐기
- 폐기된 인증서: 퇴사 후 1년 보관

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- Home Assistant, Mosquitto, Zigbee2MQTT, Node-RED, Grafana, Prometheus, Caddy, Tailscale, Docker Compose
- Raspberry Pi

절차

1. 인증서 발급
2. 개인 키 생성
3. 만료 일 기록

사용 방식

- 인증서 발급
- 개인 키 생성
- 만료 일 기록

Apple CI/CD 스토리

형식

- 위치: CI/CD 관리 볼트 / 개인 키
- 항목 이름: `Apple-CI-{app}`
- 필드: 인증서 (.p12), 만료 일, 팀 ID

보관 기간

- 활성: 퇴사 후 30일 경과 시 폐기
- 폐기된 인증서: 퇴사 후 1년 보관

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- Fastlane (Match, Gym, Pilot, Deliver, Sigh, Scan, Snapshot, Cert)
- GitHub Actions

절차

1. 인증서 발급
2. 만료 일 기록

사용 방식

- 인증서 발급
- 만료 일 기록

홈 자동화 서비스 추가

형식

- 위치: 홈 볼트 관리자 / 개인 키
- 항목 이름: `Home-{service}`
- 필드: 인증서 (.crt), 개인 키 (.key), 만료 일

보관 기간

- 활성: 퇴사 후 30일 경과 시 폐기
- 폐기된 인증서: 퇴사 후 1년 보관

공유 방식

- 1:1 전달: 1Password 항목 공유
- 팀 전체: 볼트 접근 권한

사용 도구

- Home Assistant, Mosquitto, Zigbee2MQTT, Node-RED, Grafana, Prometheus, Caddy, Tailscale, Docker Compose
- Raspberry Pi

절차

1. 인증서 발급
2. 개인 키 생성
3. 만료 일 기록

사용 방식

- 인증서 발급
- 개인 키 생성
- 만료 일 기록

- https://medium.com/musinsa-tech/%EB%AC%B4%EC%8B%A0%EC%82%AC-%EB%AA%A8%EB%B0%94%EC%9D%BC-%EC%95%B1-%EB%A6%AC%EB%89%B4%EC%96%BC-%EB%B0%8F-%EB%B0%B0%ED%8F%AC-%EC%9E%90%EB%8F%99%ED%99%94-2-%EC%9D%B4%EC%A0%84-%EB%B0%8F-%EA%B0%9C%EC%84%A0%EB%90%9C-%EB%B0%B0%ED%8F%AC-%EC%9E%90%EB%8F%99%ED%99%94
- https://medium.com/musinsa-tech/%EB%AC%B4%EC%8B%A0%EC%82%AC-%EB%AA%A8%EB%B0%94%EC%9D%BC-%EC%95%B1-%EB%A6%AC%EB%89%B4%EC%96%BC-%EB%B0%8F-%EB%B0%B0%ED%8F%AC-%EC%9E%90%EB%8F%99%ED%99%94-1-%EB%B0%B0%EA%B2%BD-%EB%B0%8F-%EB%AC%B8%EC%A0%9C%EC%A0%90

유명앱에 아키텍처하면 뭐가 떠올라지나 개인

개요

- 특히 서울대학교 하기 동안 본 인턴쉽을 하게 되었
- 2020 Android 그리고 그 이후
- iOS Android 그리고 그 이후 그
- 개발 그리고 개발 그리고 개발

주요 기술

- Firebase, Google Analytics

인턴 생활 동안 개발 방법

- 서울대학교 하기 동안 개발하는 제대로 하려면 공부
- Google Analytics 공부 그리고 개발 공부
- Firebase 개발
- 그리고 Firebase, GCS, Google Analytics 개발
  - 2020 년 그리고 이번에 공부하면 좋을 것들 그리고
  - 그리고 개발
  - Firebase Functions 그리고 그리고 공부
  - Firebase 그리고 그리고 그리고 개발

인턴 중 가장 기억에 남는 일

- 서울대학교 공부 개발 일하기
- 그리고 Firebase 개발 그리고 그리고
- 그리고 그리고 그리고 그리고 그리고

개발자 아키텍처

- 그리고 그리고 그리고 그리고 그리고 그리고 그리고 개발
- 그리고 그리고 그리고 그리고 그리고 그리고 그리고 그리고
- 그리고 그리고 그리고 그리고 그리고 그리고 그리고 그리고

카닥스의 개발

카닥스의
카닥스의
카닥스의

고객의 사용법이 다른
1588-3535

- APE 보험을 장기보기 위해 상품별 페이지 정리 및 다양한 컴포넌트 형태의 데이터 다운로드 및 연동 작업
- QR Reader를 통해 복권 당첨번호 입력간 확인 기능
- 복권 관련한 정보를 고객으로 제공을 위해 Push 서비스
- 현재 지역 기록을 수신 및 입력간 이용
- 다양한 형태의 소셜 공유 고민

현대라이프 상담시장

개인

- 목표 : 현대라이프 상담시장 모바일서비스를 기존서비스인 WIFI
- 규모 : Android, IOS 1, Back-end 3, web 2, design 1
- IOS 개인 개발자 (김하 진영 - 김도 1)

주요 기술

- AFNetworking, Realm, Firbase
- SVProgressHUD, ZBarSDK, SDWebImage, SDVension
- MDM, Appiron, INISAFE, MFaceone, MTranseSy

주요 성과

- 보안 이슈(보안) 다수 시정
- iPhone/iPad 서비스 국가 지정 (국가가 스토어에도 구현 후 지원)
- Enterprise 개발을 시행한 앱 공유
- 공인인증서 이해
- 웹 + 네이티브 하이브리 방식의 autoupdate 기능
- 프로젝트 다른 개발자가 개발 개발 후 나가 이어서 작업 시작
- Firebase Crashlytics 이용하여 앱 크래시 확인
- 실시간 다이얼로 위해 CMC는 장도 및 로드 클래 클릭 후가 지원

CJ One Card

개인

- 목표 : CJ One 카드 개인(개발대기) 마이앱
- 규모 : Android 1, IOS 1, Flut 장 기획가 1, 디자이너 2
- 개발 : iOS 개인 개발

주요 기술

- Objective-c, WebView, Deeplink, APNs
- Server-Driven UI, Firebase, Mobile application Accessibility

주요 성과

- 앱 심사에 장기간을 위한 UI/UX, 가이드 장구 구성, 프리뷰 지원
- 서비스(보안) 홈 크리틀 위한 CI/LON 시 적용
- 디자인 가이드 장기를 통한 공통 위젯을 제공
- 지원하는 Firebase Analytics 연동을 위한 컨테이션 정의
- 개발 중 발생하는 이슈를 해결해 사용
- 에스에이드, 서드 파티에 종료하게 작업
- Voice Over 기능 개인

아주캐피탈

개인

- 목표 : 유저 서비스 마이그레이션 및 나의 직원용 앱 개발
- 규모 : Android 1, IOS 1, web 2, back-end 2
- 개발 : iOS 개인 개발

주요 기술

- Sarna, NAppProtect, MDM, NFilter, SmartABFramework
- KMINI, MagicSE, MagicSign, MagicMS

주요 성과

- 아주캐피탈 리뉴얼과 마이그레이션 작업을 잘 개발
- 개발중인 상태 **Alternative** 개발을 통해 상태 개발
- 전자의 장이었을 위해 Alternative 사용중을 놓음
- 해외개발 경우를 위한 App <-> Web간의 원만히 일정해
- 운영중 장치가 다양한 보안 상황 사용 경험
- 보이스피시 이용하기 장치 해제하기 상황에서도 일정하기 개발 실시

thePay / 99Pay

개인

- 목표 : 유선 서비스 개발자 해외연변 서비스 제공
- 규모 : Android 1, IOS 1, Back-end 1
- 개발 : iOS 개인 개발자

주요 기술

- Objective-c, WebView, APNs, AFNetworking, LocalCable

주요 성과

- 1.3.5 인터 지원을 위해, 화면 노출 시점에 문구 삽입
- 다양하고 단한 정해진 지정 조건의 기능(최초로 exposure 한 인터 삽입)을 사용
- when -> String.localizable 변경하는 스크립트 생성
- String.localizable -> enum 지원 변경하는 스크립트 생성
- 규칙들을 통한(상황에서 인지) 동일성 생성
- 하단 문제 해결 (공고 지원)

엘리시안

개인

- 목표 : 엘리시안 강촌/제주 예약 및 실시간 스키장 확인 기능 제공
- 규모 : Android 1, web 1, Back-end 1
- 개발 : iOS 개인 개발자

주요 기술

- Objective-c, WebView, APNs, AFNetworking

주요 성과

- CCTV를 통해 스키장 현재 상태 확인 기능
- 제소 예약의 경우 지우 삭제 기능
- 네이티브에 정해 방법도 정하여 APNS 푸시 지원

유니텍

MFCC(Multi Function Control console)

개인

- 목표 : 산공도 이어 인트롤 콘솔 제작
- 규모 : 3인 개발
- 개발 : 남두 인터페이스 데이터 연결

주요 기술

- MFC, c++

주요 성과

- 외국 인력을 위한 사용자인터 기능 제공
- 유저, 방법, 데이터, 이력 속도 입력 후 장비 원격에 표시
- 연결 인터 장치 입력 방식 속도 확인 기능
- 인터페이스 장치 및 실시간 연결 속도 확인
- 인터 라인/장치간 스트레스 (장기)을 여건 고구
- 특정 이력 입력장치의 기능 지원 (입력방향 이어 사항)
- 실제 이력장치와 원격 입력/이력 테스트
- 장비한 위치와 원격 입력/이력의 기타 추가 기능 지원
- 관련 노드

강원대학교 전산원

강원대학교 App

개인

- 목표 : 강원대학교 전산원의 학교정보 / 대표 앱 제작(모바일 정보 제공, 식당 예노 정보 제공, 셔틀버스 정보 제공 등)
- 규모 : Android 1, IOS 1, Back-end 1
- 개발 : Android 개인 개발

주요 기술

- Android Java, XML환경

주요 성과

- 개인에 의한 단결로도
- Native 개발 및 유지보수 , 후후 Flex Mobile로 변경 시 마이스터드는
- 위치정보, 길찾기, VPP 연동, API
- 전산원에서 DAEMON 데이터 제공 지원
- 뉴스 탭 등

리딩 경험

Client 팀장

개인

- 팀원 : Android 3, iOS 3, web 4
- 기간 : 2024.10 ~ 2024.12

주요 성과

- web/IHS 퍼블릭용과 개선
- web/KR CJICG 사용량 개선
- Jira Cloud 이용률

App팀 팀장

개인

- 팀원 : Android 3, iOS 3
- 기간 : 2023.09 ~ 2024.10

주요 성과

- 2024년 하반기
  - 서드 + 인큐베 TBD 도입, 배포절차 도입, Jira Cloud 이용률
  - 서드 + 인큐베 외부플랫폼 도입, 해외 서비 결정
  - 인큐베 + 인큐베 Tuist 적용
  - 서드 + 인큐베 + 인큐베 모듈 분리(TUMA)
  - 관련 블로그
- 2024년 상반기
  - 서드 + 인큐베 SDK cocoapods -> SPM, Swift UI 적용, TBD 적용
  - 인큐베 팀원 : TDD 적용
  - 개발팀 공유 : 테크스택 적용, 배포 공유 이슈화, 리리뷰
- 2023년 하반기
  - 프로젝트 개선 : 브랜치 룰 개선, CI/CD 개선
  - 아키텍처 개선 : MVVM(MVVM 기반), 클린아키텍처
  - 모듈화 : 개발 분리, 기능별 배포
  - 데이터 수집 강화: GA 도입, CI/CD 데이터 수집
  - 관련 블로그

개인

- 월 1회 1:1 진행
- 개인별 목표 설정 및 평가 진행
- TIL(Today Learned) 공유 진행
- 매달의 스터디 모임 리딩 진행
  - 스크럼 진행과 회의 진행

iOS 파트 매니저

개인

- 팀원 : iOS 3
- 기간 : 2022.09 ~ 2023.09

주요 성과

- Roadmap의 작업을 통한 개인 인덱싱도 공유
- 키스톤 브랜드 플랫폼 iOS Flow 개발
- CI/CD 등 빌드/테스트, 배포 자동화 방법 주도
- VIPP 배포 자동화