



김민규님을 응원해보세요!

지금 만족하고 있어요

김민규

안드로이드 개발자



주도적이고 논리적으로 문제를 해결하는 안드로이드 개발자 김민규입니다.

목표를 달성하기 위해 모든 리소스를 투자하며, 그 과정에서 팀원들에게 동기를 부여하고 다양한 피드백을 주고받는 것을 좋아합니다.

맡은 업무는 반드시 해내며, 맡은 업무뿐 아니라 프로젝트를 위한 것이라면 무엇이든 스스로 찾아 해결해냅니다.

교육

선문대학교

대학교(학사) | 컴퓨터공학
2016.03. ~ 2022.02. | 졸업

기술 스택

Kotlin, coroutine, RxJava, Java, Git, Firebase, MVVM, clean-architecture, Jetpack

경력

지넵스

연구원 | 신규사업부

추천해요 🍷

제안하기 ✉

댓글 💬

2022.02. ~ 2024.04. (2년 3개월)

회사의 메인 프로젝트인 차량에 탑재되는 AVN 시스템의 유지보수를 진행하였습니다.

Java 1.8 버전을 사용하는 멀티 모듈 프로젝트이며, VoiceMemo/Setting 보조 지원업무를 수행하며 Java와 안드로이드에 대한 기초 역량을 키울 수 있었습니다.

한국, 중국, 미국, 러시아, 이스라엘 등 해외에 수출되는 차량에 탑재되는 시스템이기 때문에 다양한 UX와 다국어, 좌우반전 레이아웃 등을 경험하였습니다.

주요 업무는 레드마인에 등록되는 이슈 해결로, 신규 개발을 통한 구조 확장보다는 기존 프로젝트에서 발생하는 이슈와 그에 따른 사이드 이펙트에 대한 일/주 단위의 대응이었습니다. 주니어 개발자들은 이슈 책임자로 지정되지 않아 맡은 파트 외에도 다른 여러 파트들의 이슈를 책임자와 함께 트러블슈팅하였습니다.

이후 사내 신규 프로젝트 팀인 모바일 어플리케이션 개발 팀으로 부서를 이동했습니다.

팀원은 AOS 2, 서버 1, 디자인 1, 기획 1 으로 구성되어있으며,

암 환자 동행 서비스, 홈트레이닝 보조 서비스, 자동차 정비소 예약 서비스의 컨슈머/파트너스용 앱을 개발하였습니다.

안드로이드 파트에서 프로젝트 아키텍처 설계, 신규 기술 도입등과 같은 팀 단위의 기술 부채를 감소시키기 위한 프로젝트 세팅을 도맡아 진행하였으며, 개인 기술 부채 감소를 위해 지속적으로 공부하며 프로젝트 구조 변경과 보일러플레이트 코드 감소, 코드의 재사용성/효율성을 지속적으로 증가시켰습니다.

또한 기획-안드로이드-서버 간의 효율적이고 유연한 협업을 위해 방향성을 제시하고 그에 기반한 커뮤니케이션 방식을 수립하였습니다.

자기소개

재직 당시에는 4~5명 규모의 팀에서 Kotlin을 사용하는 안드로이드 파트를 주도하며 Java 기반 앱 1개, Kotlin 기반 앱 3개를 개발하였습니다.

프로젝트에 적용할 MVVM, 멀티 모듈 클린 아키텍처와 디자인 패턴, DI, UI/UX 설계 또는 써드파티 라이브러리를 도입하며 안드로이드 개발을 주도적으로 수행했습니다.

[어려운 문제를 해결한 경험]

- 4달의 기간 동안 Kotlin 기반의 앱 2개(누도카 & 누도카 파트너스)를 요구사항 분석, 기획, UI/UX 설계, 구현, QA, 배포까지 소프트웨어 생명주기 전반에 걸친 몰입 경험이 있습니다.
- 파트너스 앱의 경우, 컨
전 과정을 혼자 도맡아
사항 도출부터 배포까지
댓글 0
내에 완료하였습니다.

3. 또한 두 달 간의 개발 이후에 전체 요구사항의 50% 정도가 변경되는 상황에서 4주간의 주 단위 스프린트를 통해 주요 기능의 50% 이상을 꾸준한 리팩토링과 리뉴얼을 통해 요구사항을 맞추며 성능을 개선하였습니다.

<기술적 성취>

1. 아키텍처 설계가 모든 상황에 절대적으로 중요하다고는 생각하지 않지만, 이 전 프로젝트들에서는 그것의 장단점/중요성에 대한 인지가 없었습니다.
2. 그 결과 프로젝트가 진행될수록 구조적 개선의 중요성만을 깨달은 상태로 기술 부채가 쌓여가는 것을 경험하였습니다.
3. 디자인 패턴(MVC, MVVM) 이외에는 프로젝트의 특성/확장성/유지보수성을 염두한 아키텍처 설계를 경험해 보지 못하였습니다. 그 당시에 책을 통해 "클린 아키텍처"에 대해 공부하고 있었고, 이상적이라 생각한 구조였기 때문에 프로젝트에 적용하였습니다.
또한 이를 통해 이상적인 아키텍처에 대해 경험해본다면, 이후 다양한 프로젝트에서 최적의 판단을 하는 데에 도움이 될 것으로 생각하였습니다.
4. 팀에 UI/UX 디자이너가 없었기 때문에 UI/UX에 대한 학습도 진행 중이었습니다.
 - a. 안드로이드 UI/UX 가이드라인을 따를 것
 - i. 단순성 + 일관성 있는 UI를 설계하되, 의미 없는 요소는 전부 제거할 것
 - ii. 상호 작용 비용을 최소화할 것
 - iii. 화면 특성에 따른 인지 부하를 고려할 것
 - iv. 지나친 대조를 지양할 것
 - b. 직관성과 유연성에 포커스를 둘 것

<기술적 성취의 성과>

프로젝트란에 작성된 **누도카 파트너스**와 **누도카**에 대한 경험을 작성하였습니다.

1. 아키텍처
 - a. 아키텍처 설계 단계에 큰 리소스를 투자해서 기술 부채의 바운더리를 크게 감소시켜본 경험이 없었기 때문에, 구조적 개선을 위해 프로젝트의 가장 중요한 특성을 도출해 냈습니다.
 - b. 파트너스-컨슈머 앱 간의 상호 작용이 존재하기 때문에 프로젝트 진행 간에 "파트너스/컨슈머의 변경 사항이 컨슈머/파트너스에 전파되는 현상"이 발생할 가능성을 염두에 두었습니다.
 - c. "서로 다른 앱 간의 변경 사항 전파"는 서비스 특성 상 배제할 수 없다고 생각하여 파트너스/컨슈머 앱 각각에서 확장성/유지보수성을 최우선 가치로 두고, 변경의 전파를 최소한으로 제한할 수 있는 구조로 설계하였습니다.
 - i. 멀티 모듈 구조

댓글 0 다.

- ii. Domain 계층은 독립적인 모듈로, Data/Presentation 계층은 모두 Domain 계층 의존적으로 설계하여 저수준 모듈의 변경 사항이 고수준 모듈로 전파되지 않도록 하였습니다.

2. 의존성 주입 및 확장성 증가

- a. 의존성 주입과 Domain-Data 계층 간의 의존성 역전 원칙 적용을 위해 Hilt를 사용하였습니다.(각 모듈 별로 Hilt module 클래스 사용)

3. UI/UX 개선

- a. 사용자의 액션을 끌어내야 하는 화면에 정적인 콘텐츠만 존재하는 상황
 - i. Lottie 라이브러리로 CTA 버튼 주변에 동적인 요소들을 배치하여 사용자의 포커스를 이끌었습니다.
- b. 차량 수리 리포트 화면에서 발생하는 1-2초 간의 로딩(다수의 요소/이미지가 존재하는 화면)
 - i. Shimmer 라이브러리를 사용한 스켈레톤 로딩을 통해 2초 미만의 지연 시간이 발생하는 요소들의 UX를 개선하였습니다.
- c. 히스토리 백/하이어라키 백 구분
 - i. UI/UX 시나리오 내에서 특정 화면들 간의 전환에 대해 화면 전환 전략을 다르게 구성하였습니다.
ex) 예약 프로세스 중 "주소 검색" 버튼 클릭 시, 외부 개념의 개입이므로 입력 완료 후 화면 전환 시 하이어라키 백 적용
- d. 토스트/스낵바 사용 케이스 구분
 - i. 긍정적 피드백의 경우 스낵바, 부정적 피드백의 경우 토스트를 사용하여 피드백하였으며, 일관된 적용으로 사용자가 피드백의 중요도와 목적에 대해 명확하게 인지할 수 있습니다.

4. 메모리 누수 개선

- a. Debug 모드에서 LeakCanary를 사용하여 메모리 누수를 감지하고, 해당 문제를 모두 개선하였습니다.
- b. RxKotlin의 CompositeDisposable을 사용하여 각 프래그먼트에서 리스너의 할당/해제에 의한 메모리 누수를 방지하고, 보일러플레이트 코드를 감소시켰습니다.

[동료와의 협업]

애자일한 사고를 가지고 스프린트 단위로 목표를 빠르게 달성하는 것을 좋아합니다.

저는 빠르고 정확한 피드백을 기반으로 동료와 함께 문제를 해결해나가며,

팀원들과 목표를 달성하기 위해 협업하는 과정에서 항상 주도적이고 능동적인 자세를 가진 사람입니다.

정비소 예약 앱의 개발 과

댓글

정비소, 정비 프로세스

등의 부족한 도메인 지식들

그로 인해 프로젝트에 대한 팀원들의 몰입도를 향상시킬 수 있었으며, 프로젝트 초반에는 진행하지 않았던 코드 리뷰 문화를 도입하여 코드의 품질을 높임과 동시에 책임감 있는 코드를 작성할 수 있는 환경을 마련했습니다.

프로젝트

러너비(Runner-be)

사이드 프로젝트

2024.07. ~ 진행 중

런닝 취미를 가진 사용자들에게 새로운 런닝 경험을 제공하고자 진행중인 프로젝트입니다. 기존에 운영되던 서비스의 안드로이드 네이티브 개발을 맡고 있으며, 서비스 리뉴얼을 위해 신규 기능 추가 및 유지 보수를 진행하고 있습니다.

팀은 AOS 1, IOS 1, 서버 1, UI/UX 1, 디자인 2로 구성되어 있습니다.

[주요 기능]

- 러닝 모집
 - 러닝 페이스/성별/연령 등 다양한 조건을 지정하여 참가자를 모집하는 게시글
 - 위치 기반의 모집 글 작성 및 조회
 - 지도에 각 게시글 별 마커 노출
 - 모집 글 내부에서 해당 러닝 참여 및 타 러너 정보 조회
 - 북마크 기능을 통한 모집 글 관리
- 러닝 로그
 - 개인/단체 러닝 별 러닝을 기록하는 기능
 - 스탬프/상세 설명/날씨 입력
- 러닝 톡
 - 참여한 러닝의 참가자들과 대화하는 기능
 - 이미지 전송
 - 현재는 소켓 통신을 사용한 실시간 채팅은 아닙니다
 - UI는 채팅 기능과 동일하게 구현되어 있으나, 내부 로직은 일반적인 게시글/댓글과 동일하게 작성되어 있습니다
- 러닝 기록 관리
 - 주/월 단위로 러닝 기록을 관리할 수 있는 기능
 - 참여한 러닝에 대해 참여자들의 출석을 관리하는 기능
 - 참여한 러닝이 있는 경우 단체 러닝 기록 확인에 개인 러닝 기록 작성

댓글 0

[기술 스택]

- Kotlin 1.8.0
- 단일 모듈 MVVM
- Coroutines + Flow
- Retrofit + Gson
- Android jetpack(databinding, lifecycle, paging3, navigation)
- Hilt
- RxKotlin+RxBinding
- Glide
- Notion, Figma, Zeplin, Slack, GatherTown, Postman

누도카 파트너스

지냅스

2023.12. ~ 2024.03.

누도카 사용자들이 이용하는 정비소(파트너스)에게 예약 관리/리포트 작성/내역 조회/정비소 관리를 제공하는 서비스입니다.

프로젝트 구성원은 AOS 1, 서버 1 입니다.

기여도 : 100%

아키텍처 및 요약 설명 : [https://torch-scallop-](https://torch-scallop-dca.notion.site/f49860335c684ec2b3b8416082777cf2?pvs=4)

[dca.notion.site/f49860335c684ec2b3b8416082777cf2?pvs=4](https://torch-scallop-dca.notion.site/f49860335c684ec2b3b8416082777cf2?pvs=4)

[주요 기능]

- 예약 관리
 - 컨슈머 앱에서 발생한 요청을 정비소가 관리하는 기능
 - 예약 요청/예약 확정/지난 예약 등 조건에 맞춰 필터링
- 정비 현황
 - 예약 상태별 정비 현황을 업데이트할 수 있는 기능
 - 해당 차량의 주행 거리 업데이트
 - 정비 리포트/정비사 코멘트 작성
- 정비 내역
 - 모든 상태의 예약들을 년/월 조건에 따라 조회하는 기능
- 통계/정산
 - 년/월 조건에 따라 해당 기간의 통계를 조회하는 기능
- 정비소 관리
 - 정비소의 영업일/영업 시간/휴무일 지정을 통해 관리하는 기능
 - 정비소 프로필 사진
 - 해당 기능을 통해

댓글 0

지에 노출

[기술 스택]

- Kotlin 1.8.0
- 멀티 모듈 클린아키텍처 + MVVM
- Coroutines + Flow
- Retrofit + Gson
- Android jetpack(databinding, lifecycle, paging3, navigation)
- Hilt
- RxKotlin+RxBinding
- LeakCanary
- Fastlane + app distribution(테스트 앱 배포)
- Glide
- Shimmer
- Lottie

누도카

지냅스

2023.10. ~ 2024.03.

사용자 주변의 차량 정비소를 소개하고 정비 예약을 통해 이용을 돕는 서비스입니다.

프로젝트 구성원은 AOS 2, 서버 1, 기획 및 디자인 1 입니다.

기여도 : 50%

아키텍처 구조도 : <https://torch-scallop-dca.notion.site/eb7e0992bb0d405ab73c08fdb62219fe?pvs=4>

[주요 기능]

- 내 차량 관리
 - Hyphen API를 사용한 차량 조회 후 차량을 등록하는 기능
 - 대표차 지정 및 삭제 기능
- 예약
 - 차량을 정비하기 위한 예약 기능
 - 엔진오일/타이어/수리/점검 각 항목 별 예약 프로세스 제공
- 커뮤니티
 - 지역별/차종 별 게시글을 조회하는 게시판 기능
- 정비 리포트
 - 파트너스가 작성한 정비 리포트들을 조회하는 기능
 - 수리/점검 별로 11
- 마이페이지

댓글 💬

- 내 정보 관리/오픈소스 라이선스 고지/알림 설정 등을 관리하는 기능
- 네이버/카카오 소셜 로그인
- 결제
 - Google billing API를 사용한 구독 결제 기능
 - PortOne API를 사용한 연간 결제 기능

[기술 스택]

- Kotlin 1.8.0
- 멀티 모듈 클린아키텍처 + MVVM
- Coroutines + Flow
- Hilt
- Retrofit + Gson
- Google billing API + Portone API
- Android jetpack(databinding, lifecycle, paging3, navigation)

홈트레이닝 보조 서비스

지냅스

2023.08. ~ 2023.09.

홈트레이닝을 하는 사용자에게 BLE 기기 연동을 통한 운동 자세 보조를 제공하는 서비스의 프로토타입입니다.

프로젝트 구성원은 AOS 1, 하드웨어 1 입니다.

[주요 기능]

- Bluetooth 디바이스 조회 및 연결
- 4개의 BLE 디바이스와의 통신 및 데이터 조회

[기술 스택]

- Kotlin 1.8.0
- 멀티 모듈 MVVM
- RxKotlin + RxBle
- Coroutines + Flow
- Android jetpack(databinding, lifecycle, navigation)

Wecan

지냅스

2022.11. ~ 2023.02.

댓글 💬

암 환자의 병원 방문을 보조하는 동행인을 매칭해주는 서비스입니다.
프로젝트 구성원은 AOS 2, 서버 1, 기획 및 디자인 1 입니다.

[주요 기능]

- 예약
 - 수도권 내 병원 조회 및 동행인을 지정하여 예약하는 기능
- 의료 정보
 - 수도권 내 병원 정보 제공
 - 각 질병 별 전문의 정보 제공

[기술 스택]

- 단일 모듈 MVVM
- Retrofit + Gson
- RxJava + RxBinding
- Glide

자격증

정보처리기사

한국산업인력공단
2021.11.

포트폴리오

URL

[기술 블로그](#)



Powered by Rallit.

댓글 💬