

# 윤용운 포트폴리오

## Work Experience

### 포토위젯

전세계 4300만 회원 보유, iOS 전용 위젯을 활용한 폰 꾸미기 앱

근무 기간: 2023.05 ~ 2023.08

iOS 14.0+ UIKit SwiftUI Combine Swift Concurrency Instruments TCA MVVM-C

#### 1. 마이 페이지 리뉴얼

UIKit SwiftUI Combine Swift Concurrency Instruments TCA



#### Problem

- 포토 위젯 앱은 기존 구조(Uikit 기반 MVVM-C)와 팀에서 선택한 구조(SwiftUI 기반 TCA)가 공존
  - 마이 페이지는 기존 구조 사이에 들어가는 화면이었기 때문에 두 구조간 연동 필요
- 불필요한 계산으로 인한 앱 성능 저하
- 상태 값이 변경되더라도 뷰가 렌더링 되지 않는 문제
  - 위젯을 수정하더라도 반영되지 않음
- 무분별한 글로벌 객체 참조로 발생하는 메모리 누수

#### Action

- TCA 구조에서 제어할 수 없는 로직을 분리하여 별도로 관리
- 데이터 변경 시 정확한 비교 통해 렌더링 되도록 수정(화면 로딩 속도 60% 개선)
- Instruments(Leaks)를 사용하여 누수가 발생하는 코드 격리

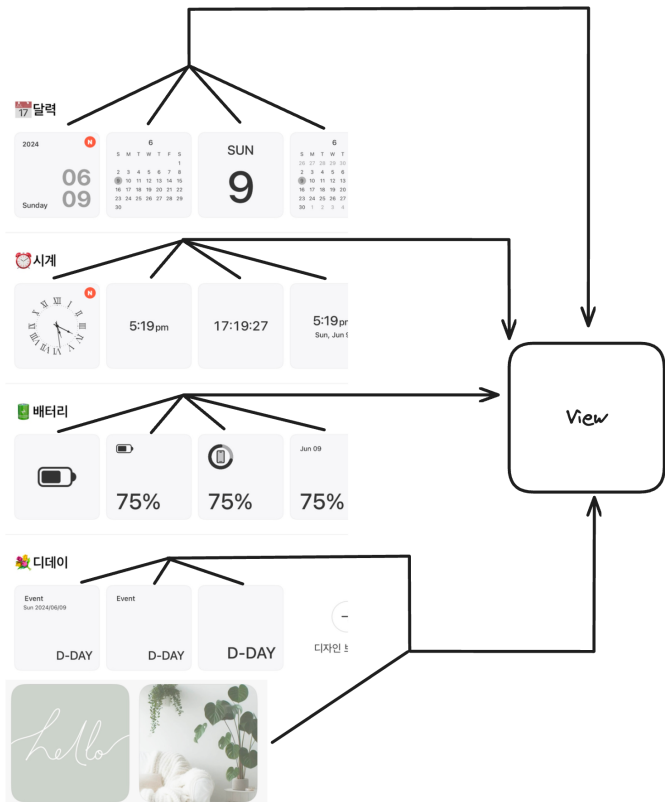
#### Result

- 예측 가능한 코드베이스로 개선되어 개발 생산성 향상 및 유지 보수성 증가
- 여러 화면에서 의도한 대로 표시되지 않던 버그를 수정

- 마이페이지에서 발생하는 메모리 누수 제거

## 2. 다양한 도메인에서 재활용할 수 있는 뷰 구현

SwiftUI



### Problem

- 여러 타입의 위젯에서 동일하게 변경해야 하는 데이터를 일관되게 제어하지 못함  
ex) App Theme(Light Mode, Dark Mode) 변경에 따른 색상 변경

### Action

- 위젯 상태를 관리하는 미들웨어 추가  
ex) 색을 변경하는 미들웨어, 폰트를 변경하는 미들웨어

### Result

- 분리된 인터페이스를 호출하는 대신, 하나의 인터페이스를 통해서 뷰를 사용할 수 있도록 제공
- 분산되어 있던 상태 관리 로직을 한곳에서 관리
- 기존 버그 수정하고 추후 발생 가능성을 낮춤

## 3. 네비게이션 모듈 설계

UIKit

SwiftUI

Combine

TCA

### Problem

- UIKit과 SwiftUI가 혼용되는 개발 환경
  - SwiftUI의 네비게이션 링크를 사용했을 때, UIKit에서 정상적으로 동작하지 않는 문제
  - 잘못된 네비게이션 컨트롤러 접근으로 인한 의도치 않은 화면 노출

### Challenge

- 신규 피쳐 개발의 생산성을 올리고 유지보수 측면에서 통합한 모듈의 필요
- 레거시 코드와 신규 코드(TCA) 모두 호환되어야 함

### Action

- SwiftUI에서도 네비게이션 컨트롤러를 사용하도록 구현
  - UINavigationController 제작 (상태 기반 커스텀 네비게이션 컨트롤러 제작)
- 최소 버전 16.0+에서 사용하는 API를 14.0+에서 사용할 수 있도록 수정(TCA 라이브러리 API)

- UINavigationController 이해를 도울 수 있는 예제 프로젝트 제공

## Result

- 레거시 코드를 수정하지 않고 내비게이션 버그 해결
- 신규 피쳐 개발 속도 향상
- 상태 기반의 내비게이션 라우팅 구현
- 예제 프로젝트를 제공으로 팀원들의 이해도 향상

# 키네마스터

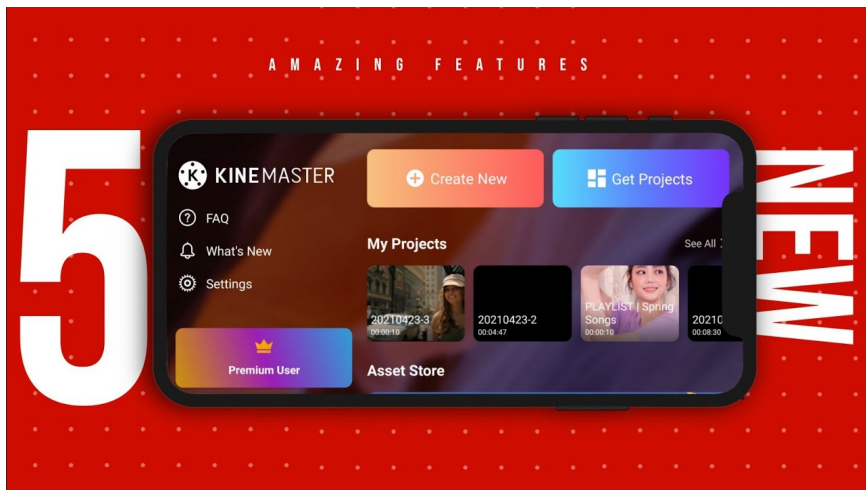
전세계 6억 다운로드, 18개국어를 제공하는 글로벌 동영상 편집 앱

근무 기간: 2020.10 ~ 2022.10

iOS 14.0+ UIKit SwiftUI Combine Swift Concurrency Instruments MVC VIVM MVVM

## 1. 대규모 개편에서 사용될 주요 화면 개발

UIKit Combine VIVM



## Problem

- 유사한 뷰 구조를 가진 화면들이 다양하게 존재(중복 코드)
- 레거시 코드의 책임 분리가 명확하지 않음
- 나라별로 다르게 노출되는 콘텐츠 존재
- Objective-C 코드와 호환

## Challenge

- 아랍권 유저 수의 증가로 인해, RTL(아랍어) 지역을 위한 UI 구현

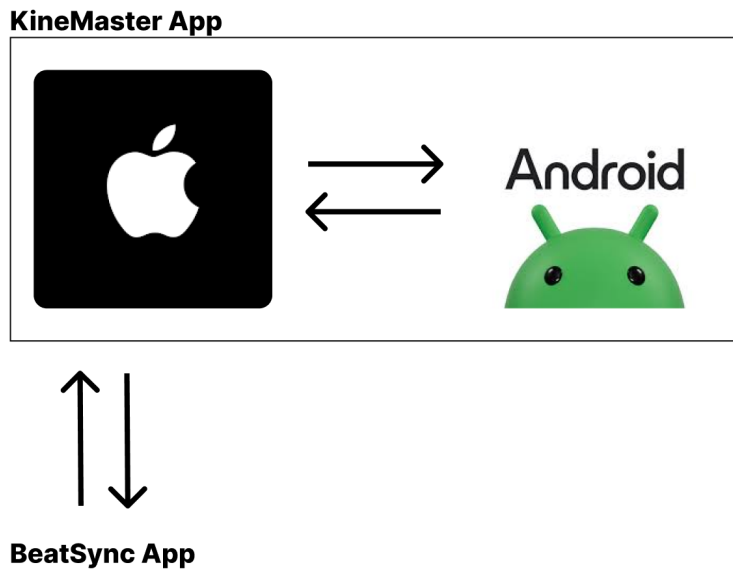
## Action

- 유사한 뷰 구조를 가진 다른 화면들(홈, 설정, 폰트 선택)을 재활용 가능하게 설계
- 리덕스를 참고한 단방향 아키텍처를 구현및 사이드 이펙트(Combine으로 구현) 관리
- 새롭게 추가된 기능의 주요 로직은 테스트 코드를 작성

## Result

- 테스트 코드 작성으로 인한 리팩토링 안정성 확보(신규 기능 테스트 커버리지 80% 달성)
- KineMaster 앱 최초로 RTL 지역을 위한 UI를 제공

## 2. 앱 간 통신 데이터 모델링



### Problem

- 데이터를 주고받는 여러 앱(키네마스터, 비트싱크 등)이 존재하며, 앱들은 다른 팀에 의해 개발
  - 앱 간 주고받는 데이터를 잘못사용해서 동일한 동영상(결과값)을 만들지 못하는 문제

### Action

- 키네마스터(iOS) - 비트싱크(iOS)간, iOS - Android간 데이터를 주고 받는 과정에서 다르게 처리되는 값 조사

### Result

- iOS와 Android 두 플랫폼에서 동일한 데이터를 표준화된 방식으로 사용하도록 문서화
- 앱 간 통신에서 발생하는 여러 버그 해결

## 3. SPM을 사용한 모듈 분리

### Problem

- 모든 코드를 메인 앱 타겟에서 관리하여 클린 빌드 시간이 10분 이상 소모되는 문제 발생

### Action

- 빌드 시간을 단축시키기 위해 새롭게 개발되는 기능과 공통적으로 사용하는 도메인을 분리

### Result

- 타겟 별로 빌드 가능하도록 변경(분리된 기능 기준 빌드 시간 80% 이상 감소)
- 프리뷰 기능을 활용한 디자이너와의 커뮤니케이션 비용 감소

## 4. 사내 발표

작업도중 발견한 문제 해결한 과정, 새롭게 학습한 기술등 다양한 주제로 발표

### Swift Concurrency

Swift Concurrency

Swift 5.5에 새롭게 도입된 Swift Concurrency를 새로운 서비스에 적용하기 위해, WWDC22 관련 세션을 정리 후 팀원들에게 공유.

### Result

- 직관적이고 간결한 비동기 코드 작성
- 지속적인 피드백으로 팀원들이 프로젝트에 적용할 수 있도록 가이드 제공

### What is MVVM

MVVM VIM

### Problem

- 개발자 마다 다른 MVVM 구현 방식

- 무분별한 싱글톤 사용으로 인한 데이터 경합 발생
- 예측하기 어려운 코드

### Result

- 안티 패턴을 문서화하여 다른 팀과도 공유
- 개선 방안과 대체제를 논의

## Custom Redux | [Blog](#)

[Instruments](#) [MVC](#) [VIVM](#) [MVVM](#)

- 시니어 분의 추천으로 기존 레거시 코드의 예측 불가능한 문제를 리덕스 패턴으로 해결한 후, 정보 공유
- 클로저 주입에서 발생하는 메모리 누수 해결 방법 공유 | [Blog](#)

### Result

- 팀원 전원이 앱 구조를 논의할 수 있는 자리를 만듦
- 키네마스터 앱에서 사용하는 구조에 대해 얼라인을 맞춤

## Community

### Let'Swift 2022

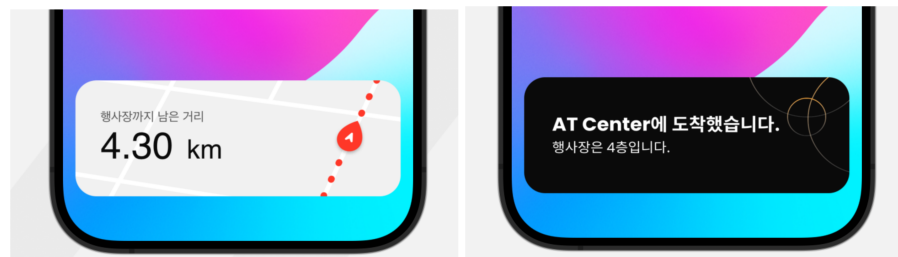
Conference Organizer | [GitHub](#)

2022.11.30 | 양재 aT 센터

활동기간: 2022.05 ~ 2022.12

### 행사장까지 남은 거리를 보여주는 LiveActivity 구현

[SwiftUI](#), [Swift Concurrency](#)



- iOS 16에서 공개된 LiveActivity를 사용하여 실시간으로 행사장까지의 거리를 보여주는 기능 구현

### Challenge

[Swift 6.0](#)

- 복잡한 로직 간소화
- 데이터 경합으로 부터 안전하게 수정 (Swift 6.0)

## Side Projects & Personal Experience

### 플랜즈

편리한 약속잡기, 관리 서비스 앱

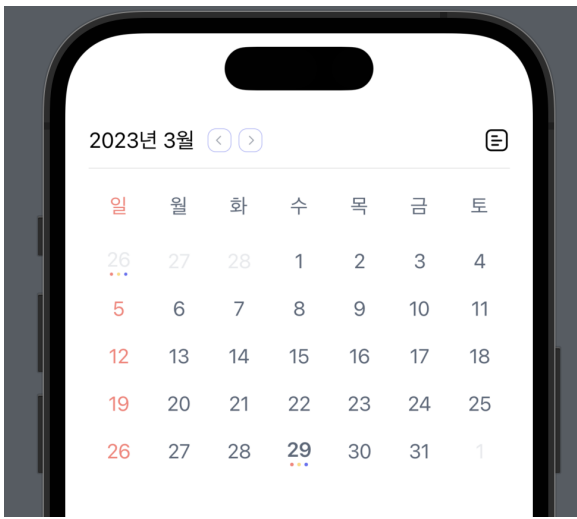
iOS 개발(팀장) | [GitHub](#)

활동 기간: 2022.09 ~ 2023.03

[SwiftUI](#) [TCA](#)

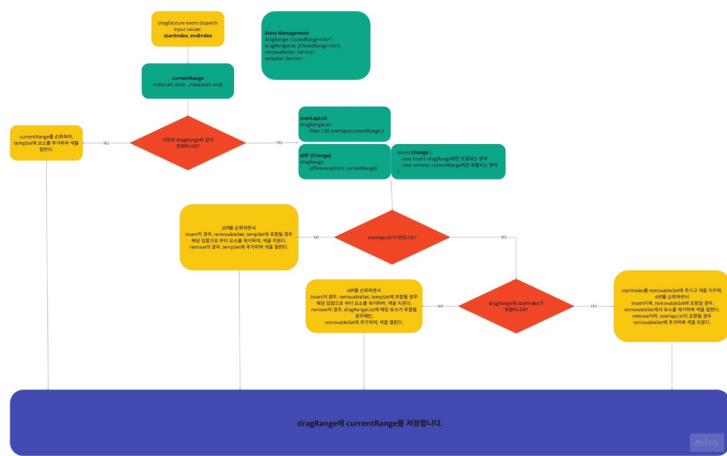
## 1. 커스텀 달력 구현

SwiftUI



- 기본 달력 컴포넌트의 기능과 디자인 요구사항에 맞지 않는 문제 발생
  - Apple 앨범 앱과 동일한 제스처 구현

## 2. 달력 날짜 정책 정의



클라이언트(iOS, Android)에서 사용되는 제스처 관련 정책 정의

- 달력 날짜 Tech spec
- 달력 제스처 Tech spec

## 코드 리뷰어

## iOS 부트 캠프

2021.04 - 2023.06

부트 캠프(Code Squad) 수료생을 대상으로 코드 리뷰어로 활동

- 3개 기수 진행 50여명의 교육생 코드 리뷰

다른 사람과 지식 공유하는 걸 즐기기 때문에, 새롭게 개발 공부를 시작하시는 분들과 소통하며 저에게도 열정을 다시 불태우는 계기가 되었습니다.

제가 과거에 했던 실수들을 보다 쉽게 이해할 수 있게 전달하려고 노력했습니다.

# Education

---

## Code Squad

iOS 부트 캠프

2020.01 - 2020.06

### 코드 스쿼드 iOS 마스터즈코스

- 미션 기반의 코드 리뷰 중심
- 페어 프로그래밍, 팀 프로젝트를 통한 협업 경험
- 다양한 구조 경험(MVC, MVVM)
- 테스트 코드 경험

Swift에서 발생할 수 있는 문제들(메모리 누수, 컨스트레인트 충돌)을 해결하는 방법을 학습했습니다.

구조에 대한 이해도를 높이고 객체 지향적인 사고를 할 수 있게 되었으며, 협업의 중요성도 알게 되었습니다.