

코드리뷰 가이드

**** 해당 디렉토리 파일에 'README.md'파일을 참고해주시면 감사하겠습니다 ****

DevCamp 2기 김건형 인턴

안녕하세요. Last Punch팀 팀장을 맡은 iOS 담당 김건형 인턴입니다.

프로젝트 전반을 RxSwift를 사용하여 MVVM으로 작성했으나, 몇몇 View에서는 상황에 따라 스토리보드를 활용하여 MVC로 채택하면서 유동적으로 구현하게 되었습니다.

해당 문서에서는 코드 리뷰를 위한 가이드로 작성되었습니다. 프로젝트 대부분이 Snapkit으로 구현되어 실제 동작 화면을 보시기 쉽지 않을 것 같습니다. [README URL](#)을 통해 **스크린샷**을 확인해 주세요

MVVM으로 작성한 코드는 View(VC)와 ViewModel로 나뉘어집니다.

View(VC)는 아래 4개의 함수를 기본 함수로 하고 있습니다.

- **init()** : 가장 초기에 실행되며, Data를 넘겨 받거나 token을 불러옴
- **bind()** : ViewModel을 바인딩함
- **attribute()** : UI 객체의 속성을 결정함
- **layout()** : UI 객체들의 constraints를 결정함

ViewModel은 ViewModel에서 **Input, Output** 객체를 생성하고 View에서 바인딩하는 방법으로 구현하고 있습니다. 또한, input은 **PublishSubject**, output은 **PublishRelay**로 구성하게 되었습니다.

MVC로 작성한 코드는 View(VC)와 .xib 파일로 나뉘어지며, 스토리보드를 통해 View를 그린 후, **File Owner**에 Custom Class를 적용하는 방법을 채택하게 되었습니다.

MVC를 혼합하여 사용하게 된 이유는 개발 초기에 **RxDataSource**를 다루는 부분에서 미숙한 부분이 있었고, 팀원들과 개발 속도를 맞추기 위해 사용하게 되었습니다. 이 부분은 설명 부분에서 더욱 자세히 다루겠습니다.

네트워크 통신 부분은 전부 **Singleton Class**를 만들어 **Singleton Patten**을 활용하여 처리하게 되었습니다.

Welcome

README URL	https://github.com/njsh4261/SGS_Last_Punch/tree/dev/src/frontend/ios#1-%ED%9A%8C%EC%9B%90%EA%B0%80%EC%9E%85
View(VC)	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/WelcomeView/WelcomeViewController.swift
설명	RxSwift를 배우고 가장 처음 구현한 초기 화면입니다. 로그인 및 회원가입 버튼을 눌러 화면을 이동하게 됩니다. 다순 화면전환 event만 있어 ViewModel은 만들지 않았습니다.
기술 스택	RxSwift, RxCocoa, SnapKit, Then(ui 객체들을 더욱 쉽게 속성을 주입시킬때 사용한 라이브러리)

회원가입

View(VC)	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/WelcomeView/RegisterEmail/RegisterViewController.swift
ViewModel	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/WelcomeView/RegisterEmail/RegisterViewModel.swift
설명	Bind Input으로 Email, code, name, password, retryPassword field value를 받아 ViewModel에서 네트워크처리를 하고 있습니다. Bind Output의 값을 통해 UI 객체들의 Hidden을 관리하며, 단계별로 회원가입을 할 수 있도록 처리 했습니다. 또한, 이메일이 변경된다면, 모든 Filed값을 초기화 하는 부분 또한 처리 했습니다.
기술 스택	RxSwift, RxCocoa, SnapKit, Then, ProgressHUD(slack과 유사한 alert와 activity indicator 애니메이션 라이브러리)

로그인

View(VC)	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/WelcomeView/LoginEmail/LoginViewController.swift
ViewModel	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/WelcomeView/RegisterEmail/RegisterViewModel.swift
설명	가장 기억나는 부분은 구현 후, 처음에 scrollView로 감싸지 않아서 로그인 버튼이 조금 밖에 보이지 않았고, 이 부분을 처리하기 위해, register와 함께 scrollView와 contentsView로 감싸게 되었습니다.
기술 스택	RxSwift, RxCocoa, SnapKit, Then, ProgressHUD, SwiftKeyChainWrapper(UserDefault와 사용법이 같고, 더욱 안전하게 사용자의 정보를 저장하는 라이브러리)

워크스페이스 목록

README URL	https://github.com/njsh4261/SGS_Last_Punch/tree/dev/src/frontend/ios#3-%EC%9B%8C%ED%81%AC%EC%8A%A4%ED%8E%98%EC%9D%B4%EC%8A%A4-%EC%84%A0%ED%83%9D%EC%B6%94%EA%B0%80%EA%B3%A0%EC%9C%A0%EC%A3%BC%EC%86%8C%EB%A1%9C-%EC%A0%91%EC%86%8D
View(VC)	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/WelcomeView/WelcomeViewController.swift
cell	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/WorkspaceListView/WorkspaceListCell.swift
설명	Rx Table의 itemSelect , itemDeslected , didscroll , Pull down/ pagination + Dirve 처리하는 부분을 깊게 배우게 되었습니다.
기술 스택	RxSwift, RxCocoa, SnapKit, Then,ProgressHUD, SwiftKeyChainWrapper

워크스페이스 생성

View(VC) 워크스페이스 이름	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/WorkspaceListView/NewWorkspace/NewWorkspaceFirstViewController.swift
View(VC) 채널 이름	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/WorkspaceListView/NewWorkspace/NewWorkspaceSecondViewController.swift
ViewModel	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/WorkspaceListView/NewWorkspace/NewWorkspaceViewModel.swift
설명	두개의 View(VC)가 같은 ViewModel을 공유하며 사용하고 있습니다. 입력받는 text길이를 알아내기 위해 viewModel에서 계속 구독중이며, 설정한 길이(15)보다 길어질 경우에 extension Reactive로 UITextField를 확장했습니다.
기술 스택	RxSwift, RxCocoa, SnapKit, Then,ProgressHUD

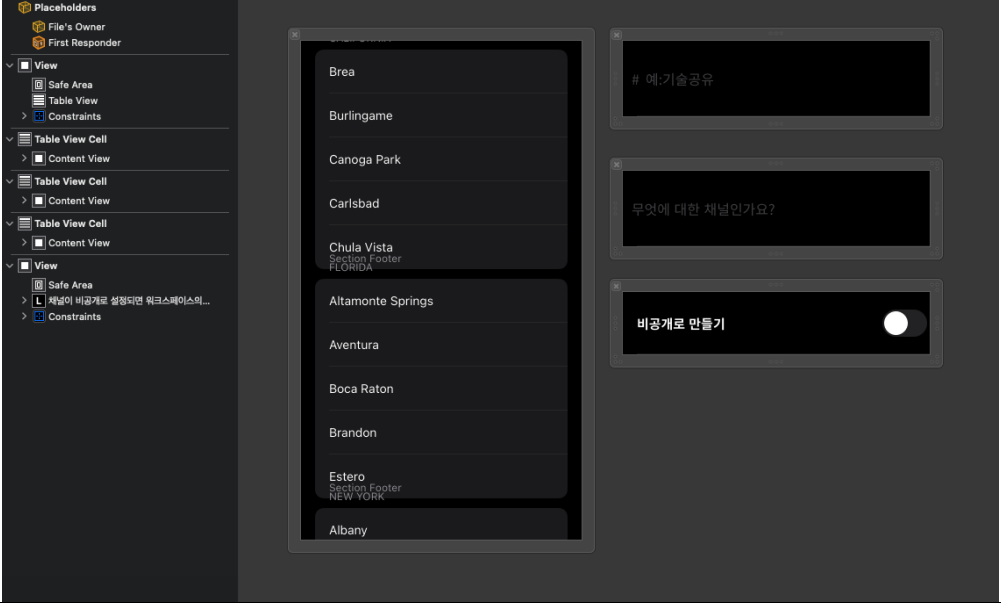
워크스페이스 URL 접속

View(VC)	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/WorkspaceListView/SearchURLWorkspace/SearchURLWorkspaceVeiwModel.swift
ViewModel	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/WorkspaceListView/SearchURLWorkspace/SearchURLWorkspaceViewController.swift

설명	Home으로 입장하는 방법이 3가지(선택, 생성, url)이고, 각자 새 TabBar를 생성하게 되어 이것과 같이 CreateHomeController 이라는 class를 만들어 생성하여 생성 후 present하게 됩니다. 더 좋은 방법은 없는지 고민중에 있습니다.
기술 스택	RxSwift, RxCocoa, SnapKit, Then, ProgressHUD, SwiftKeyChainWrapper

홈탭 및 그룹 채팅

README URL	https://github.com/njsh4261/SGS_Last_Punch/tree/dev/src/frontend/ios#4-%ED%99%88%ED%83%AD-%EB%B0%8F-%EA%B7%B8%EB%A3%B9-%EC%B1%84%ED%8C%85
View(VC)	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/WorkspaceListView/WorkspaceListViewController.swift
ViewModel	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/WorkspaceListView/WorkspaceListCell.swift
Channel Cell	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/Home/Cell/ChannelListCell.swift
Member Cell	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/Home/Cell/MemberListCell.swift
설명	가장 중요한 ViewController 와 ViewModel 이라고 생각합니다. RxDataSources 를 처음 사용해보면서, DispatchGroup 을 통해 workspace 정보, channelList, memberList 를 가져오게 됩니다. 또한, Stomp Websoket 의 Register 후, 모든 채널에 대해 subscribe 를 하게 됩니다. StompWebSoket Class 는 이곳 에서 확인할 수 있습니다.
기술 스택	RxSwift, RxCocoa, SnapKit, Then, ProgressHUD, SwiftKeyChainWrapper, RxDataSource(rx에서 table의 datasource를 다루기 위한 라이브러리)
채널 생성	
View(VC)	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/Home/NewChannel/NewChannelViewController.swift

Xib	
설명	채널 부분은 MVC로 구현하게 되었습니다.
기술 스택	RxSwift, RxCocoa, SnapKit, Then, ProgressHUD
그룹 채팅	
View(VC)	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/ChatView/GroupMessageViewController.swift
ViewModel	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/ChatView/GroupMessageViewModel.swift
설명	MessageKit을 활용하여, 상대방과 대화 메시지를 전송하는 기능입니다. 약 1주일간 message를 직접 구현하기 위해 시간을 쏟았지만, 구현하지 못하게 되어 라이브러리를 사용하게 되었습니다. 1주일간 직접 구현해본 경험을 토대로 빠르게 구현할 수 있었습니다.
기술 스택	RxSwift, RxCocoa, SnapKit, Then, ProgressHUD, SwiftKeyChainWrapper, MessageKit

노트

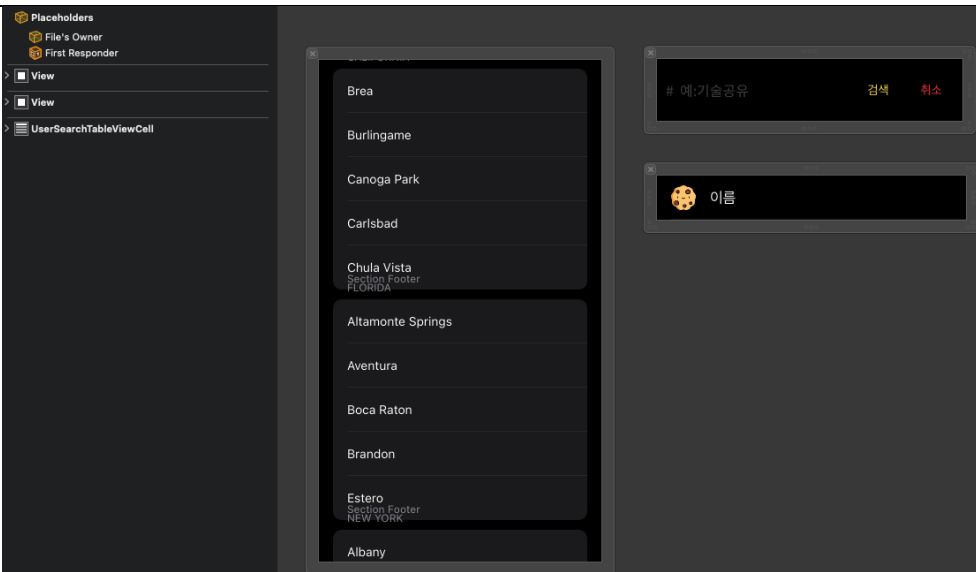
README URL	https://github.com/njsh4261/SGS_Last_Punch/tree/dev/src/frontend/ios#5-%EB%85%B8%ED%8A%B8
View(VC)	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/ChatView/Note/NoteListViewContoller.swift
ViewModel	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/ChatView/Note/NoteListViewModel.swift
Note Cell	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/ChatView/Note/NoteListCell.swift
WebView	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/ChatView/Note/CreateWKWebView.swift
설명	<p>WebKit의 WebView를 통해 Web의 노트 기능을 보여주고 있습니다.</p> <p>처음에는 노트기능을 iOS에서도 만들기 위해 직접구현하고자 하였고, 그 후 라이브러리를 찾게 되었는데, web과 공동작업을 하려면 양쪽의 라이브러리가 같아야 파싱하는 시간을 단축하게 되는데, 맞는 라이브러리가 없었습니다.</p> <p>그래서 web에서 구현하고 iOS에서는 webView로 가져오기로 했습니다.</p> <p>Collection View를 통한 노트 목록을 정의하고, ReusableView로 header를 만들고, GestureRecognizerDelegate를 이용해 note를 삭제합니다.</p>
기술 스택	RxSwift, RxCocoa, SnapKit, Then,ProgressHUD, SwiftKeyChainWrapper, RxDataSource, webKit

DM(Direct Message) 및 1:1 채팅

README URL	https://github.com/njsh4261/SGS_Last_Punch/tree/dev/src/frontend/ios#6-dmdirect-message%ED%83%AD-%EB%B0%8F-11-%EC%B1%84%ED%8C%85
View(VC)	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/DirectMessage/DirectMessageListViewController.swift
ViewModel	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/DirectMessage/DirectMessageListViewModel.swift
DM Cell	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/DirectMessage/DirectMessageListCellView.swift

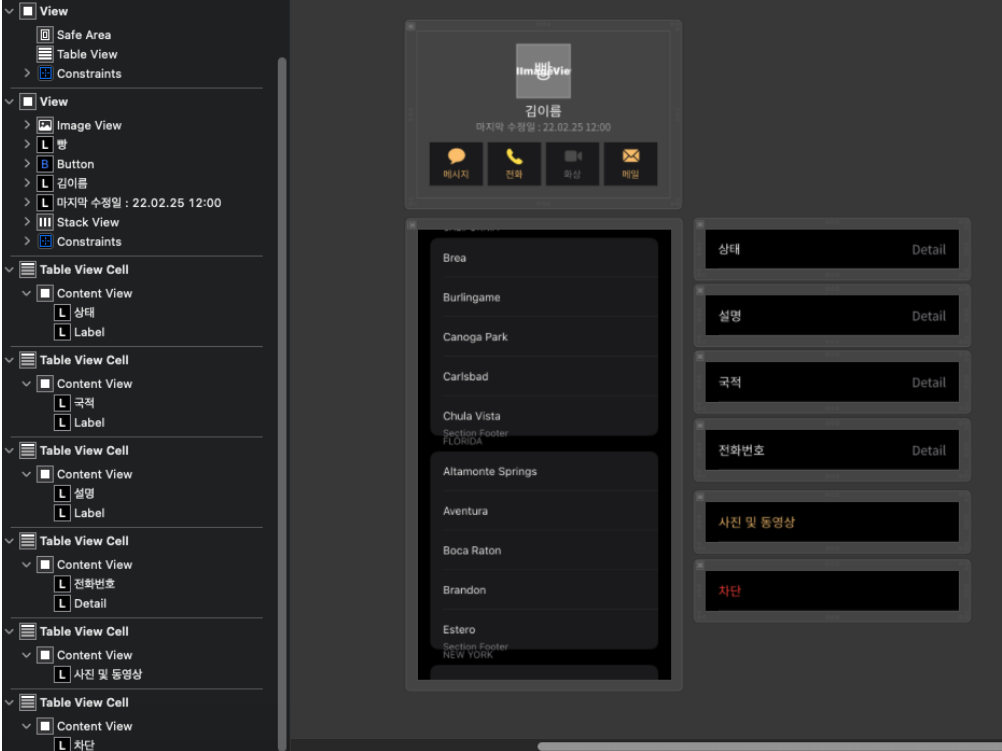
설명	<p>WebKit의 WebView를 통해 Web의 노트 기능을 보여주고 있습니다.</p> <p>처음에는 노트기능을 iOS에서도 만들기 위해 직접구현하고자 하였고, 그 후 라이브러리를 찾게 되었는데, web과 공동작업을 하려면 양쪽의 라이브러리가 같아야 파싱하는 시간을 단축하게 되는데, 맞는 라이브러리가 없었습니다.</p> <p>그래서 web에서 구현하고 iOS에서는 webView로 가져오기로 했습니다.</p> <p>Collection View를 통한 노트 목록을 정의하고, ReusableView로 header를 만들고, GestureRecognizerDelegate를 이용해 note를 삭제합니다.</p>
기술 스택	RxSwift, RxCocoa, SnapKit, Then, ProgressHUD, SwiftKeyChainWrapper, RxDataSource, webKit

사용자 검색

View(VC)	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/DirectMessage/UserSearch/UserSearchViewController.swift
Xib	
설명	Snack 사용자중 like문을 통해 검색 후, 정확히 email과 일치 하는 사용자를 추가시켰습니다.

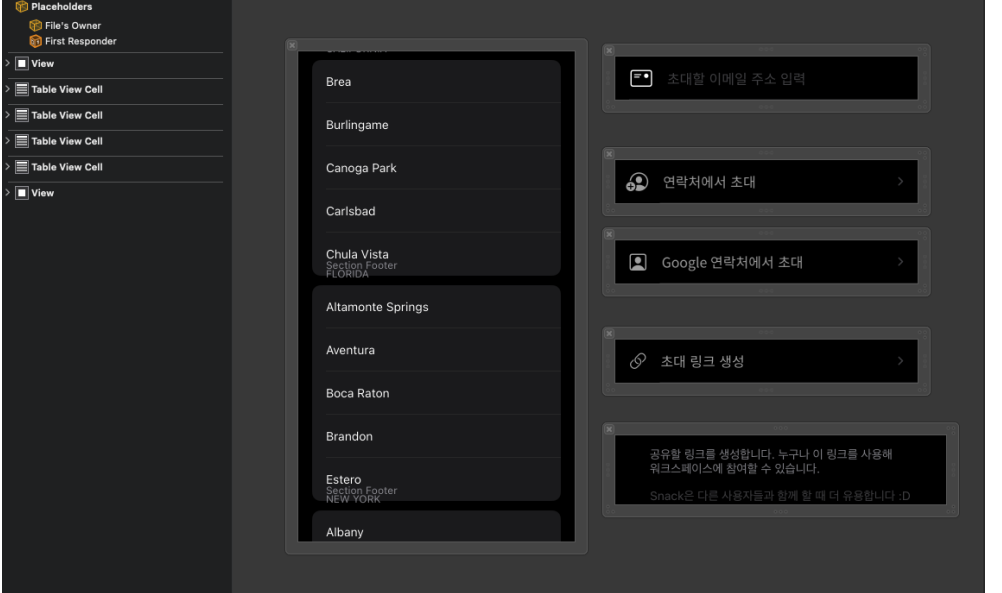
프로필 보기

View(VC)	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/ChatView/Profile/ProfileViewController.swift
ViewModel	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/ChatView/Profile/ProfileViewModel.swift

Xib	
-----	--

사용자 추가

View(VC)	https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/DirectMessage/UserInvitation/UserInvitationViewController.swift
--------------------------	---

Xib	
-----	--

[나(설정) 탭] - MVC

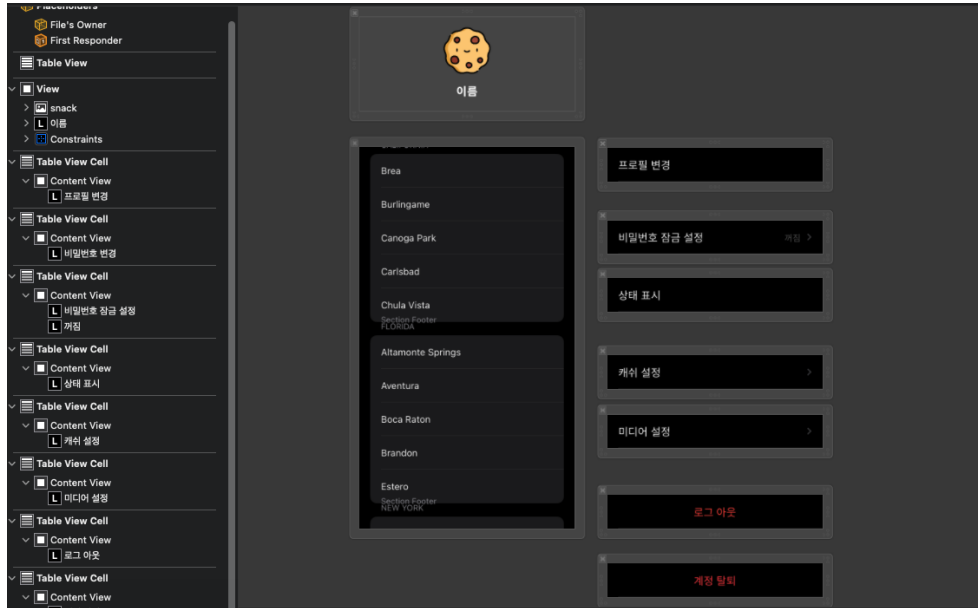
README URL :

https://github.com/njsh4261/SGS_Last_Punch/tree/dev/src/frontend/ios#7-%EB%82%98%EC%84%A4%EC%A0%95%ED%83%AD

View(VC) :

https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/Settings/SettingsViewController.swift

Xib

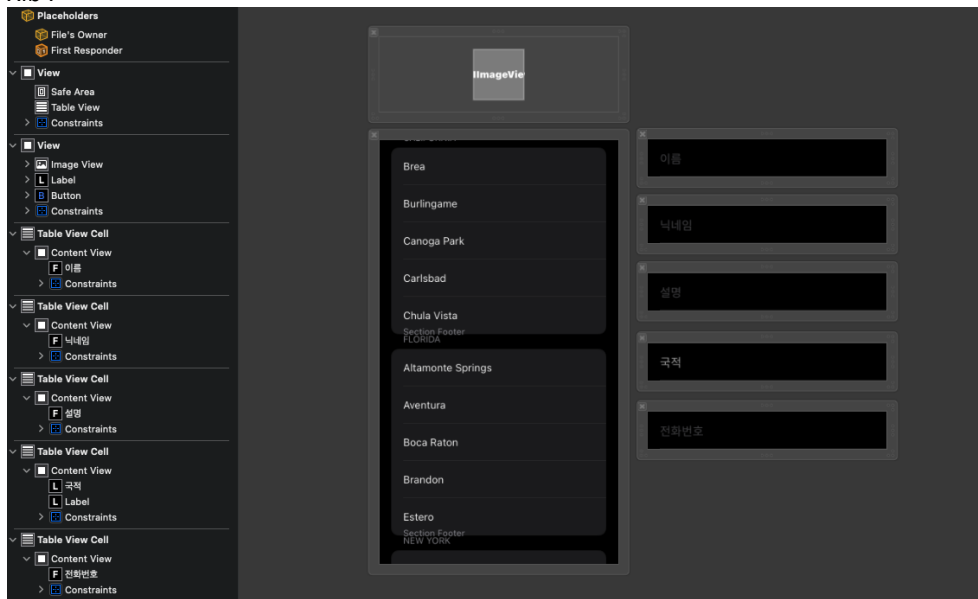


[프로필 변경]

View(VC) :

https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/Settings/SettingsViewController.swift

Xib :

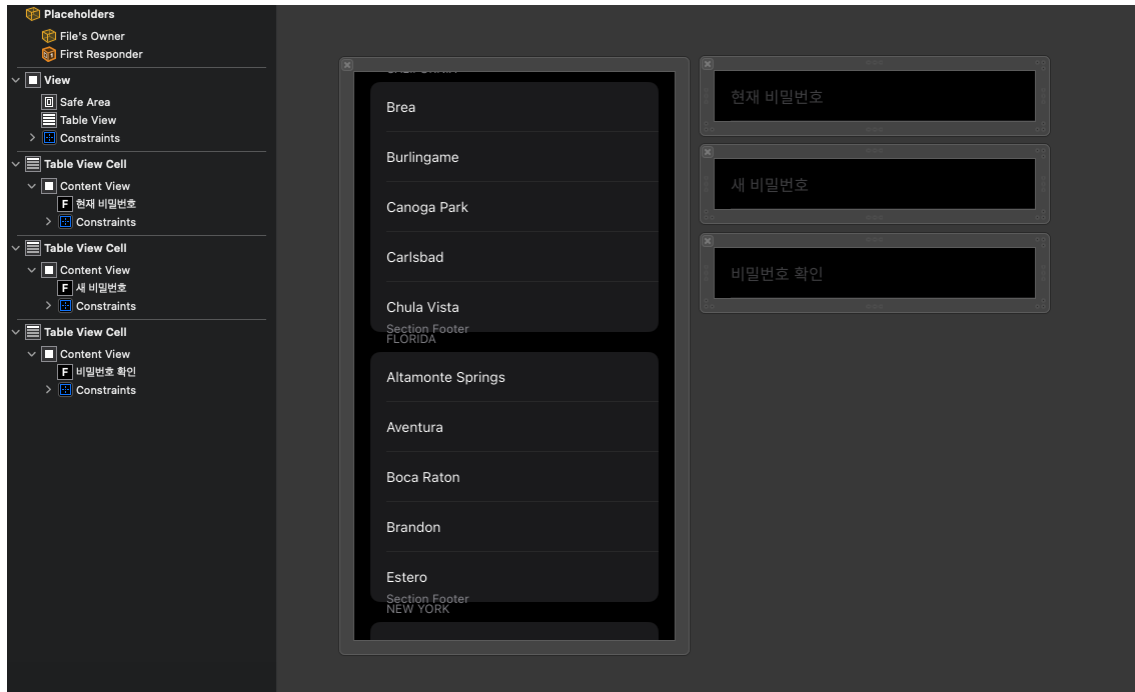


[비밀번호 변경]

View(VC) :

https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/Settings/Password/PasswordView.swift

Xib :

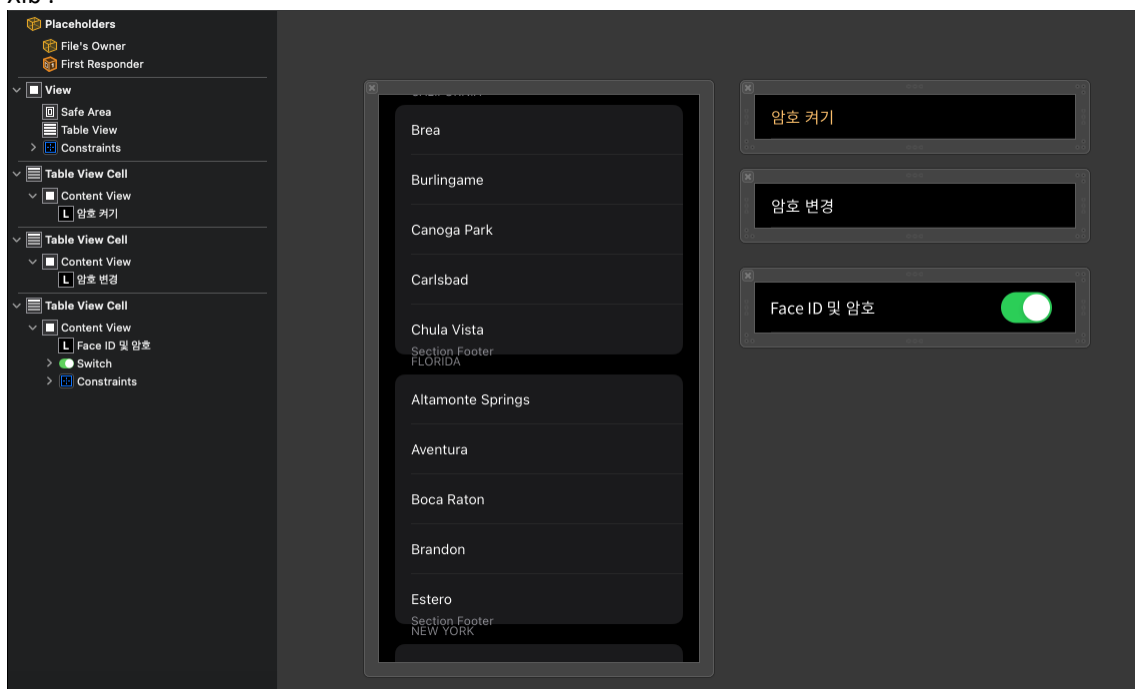


[암호(패스코드) 변경]

View(VC) :

https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/Settings/Passcode/PasscodeView.swift

Xib :

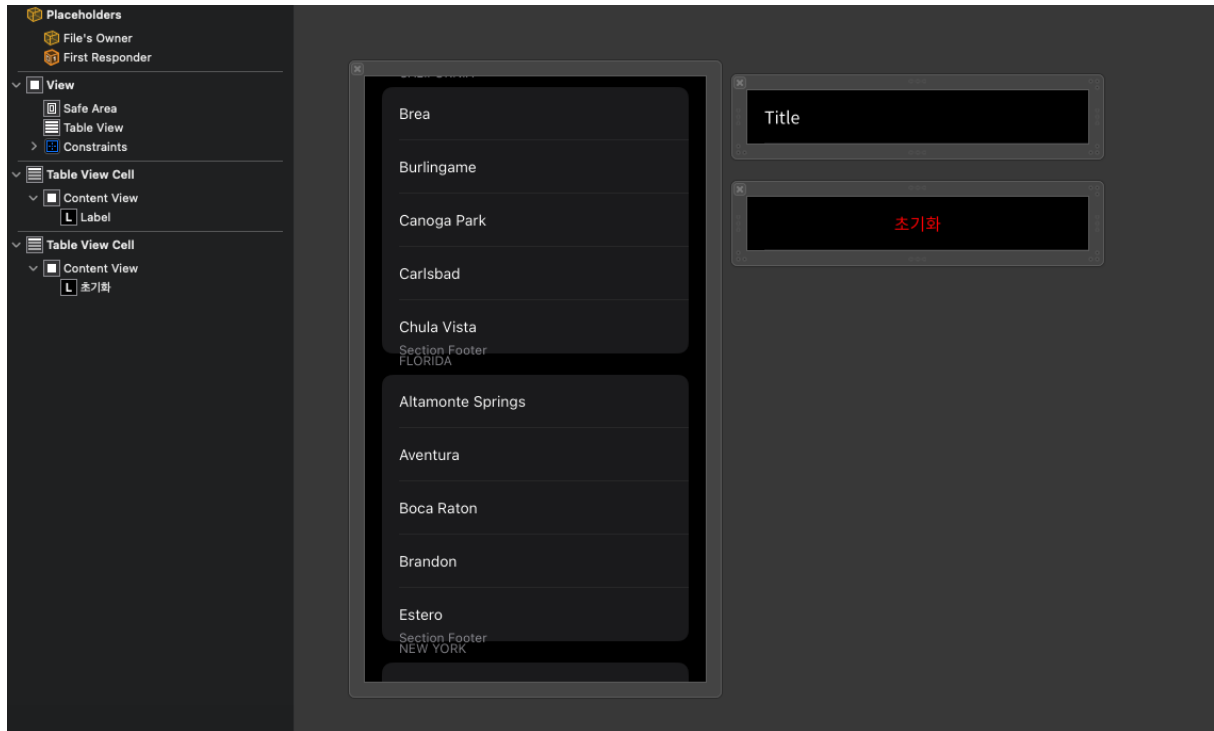


[상태 변경]

View(VC) :

https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/Settings/Status/StatusView.swift

Xib :

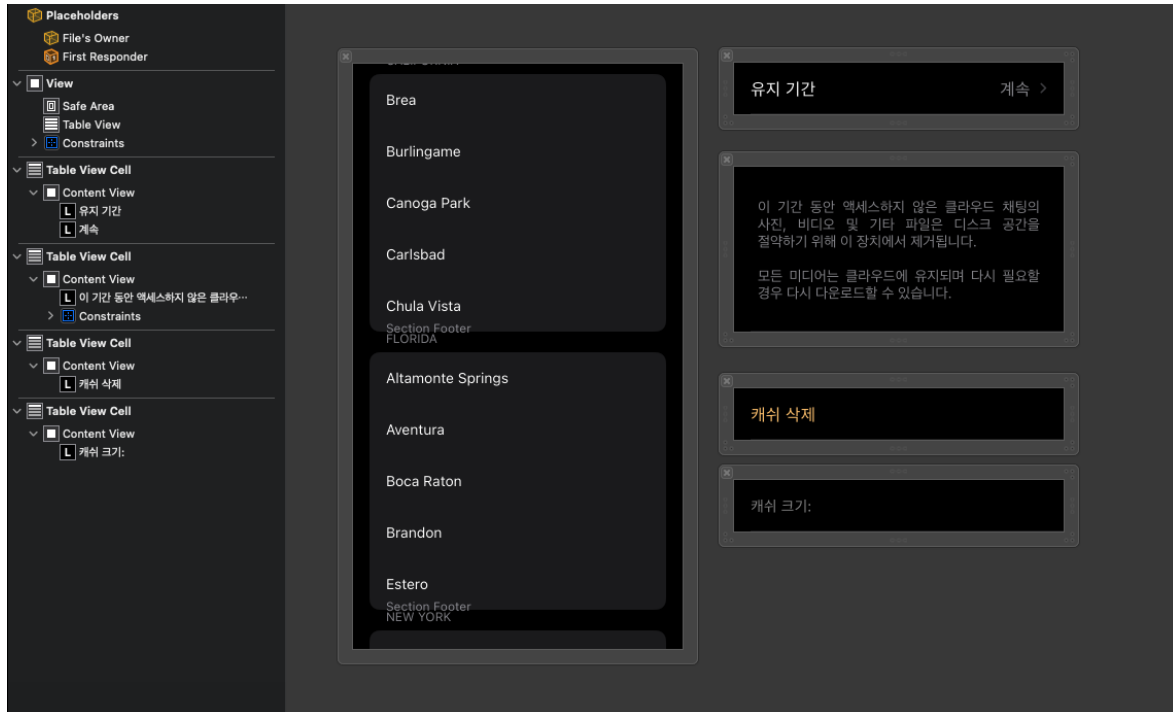


[캐쉬 설정]

View(VC) :

https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/Settings/Cache/CacheView.swift

Xib :

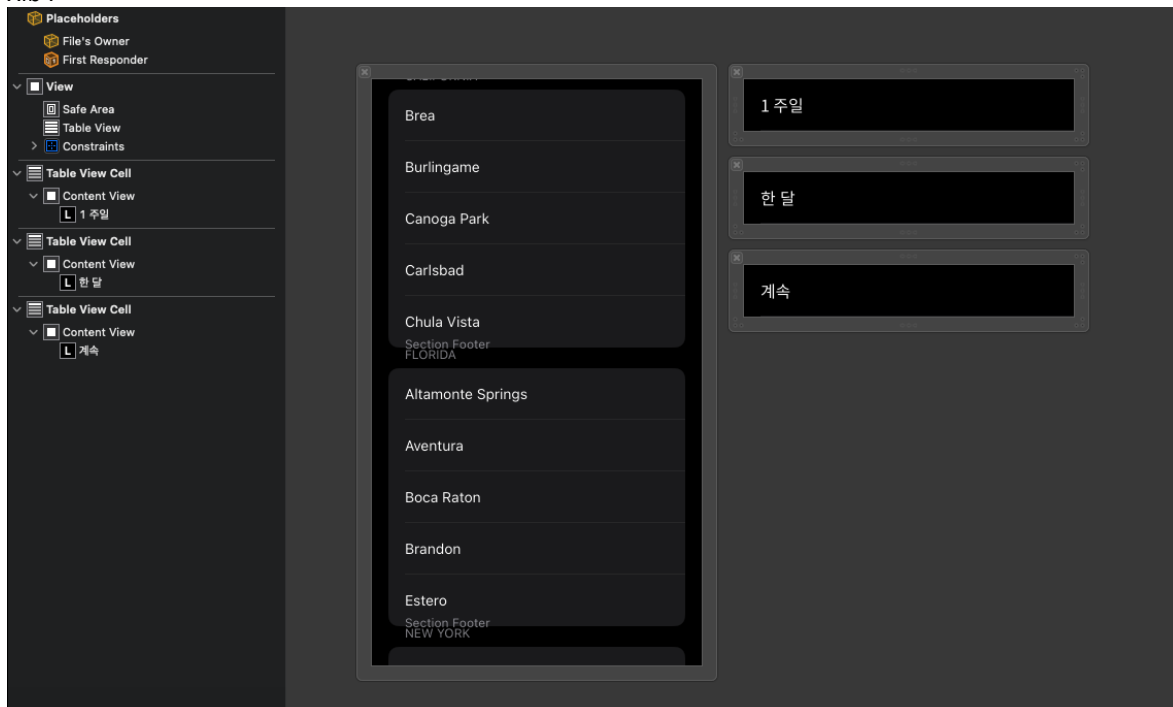


[캐쉬 설정 - 기간 설정]

View(VC) :

https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/Settings/Cache/KeepMedia/KeepMediaView.swift

Xib :

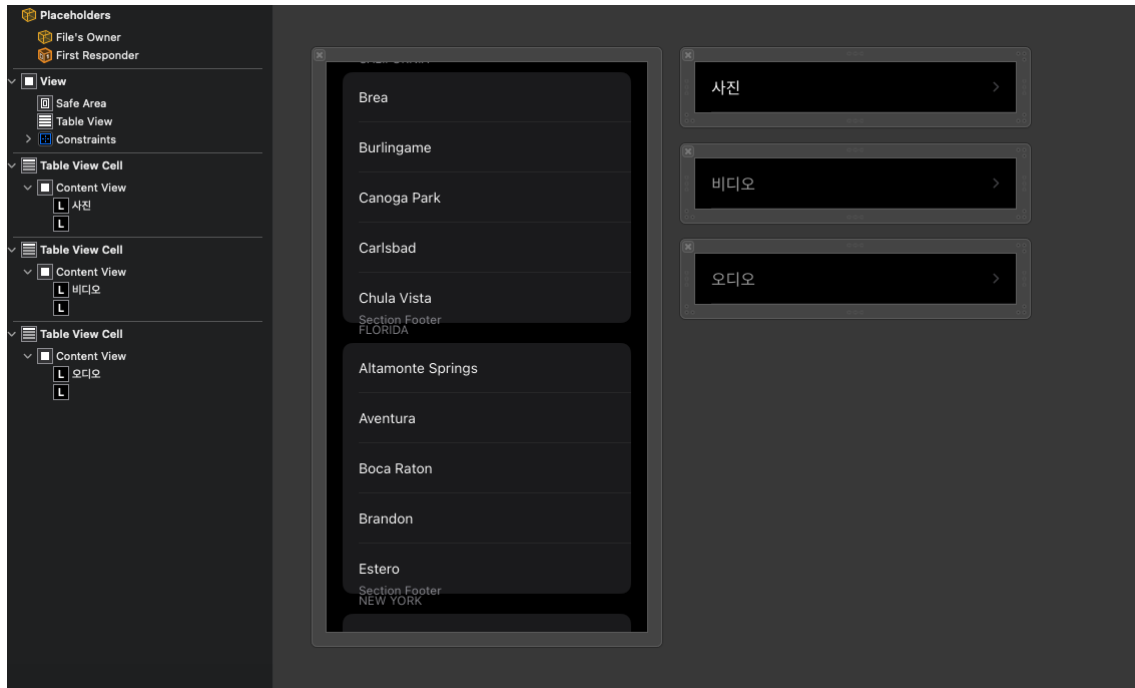


[미디어 관리]

View(VC) :

https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/Settings/Media/MeidaView.swift

Xib :

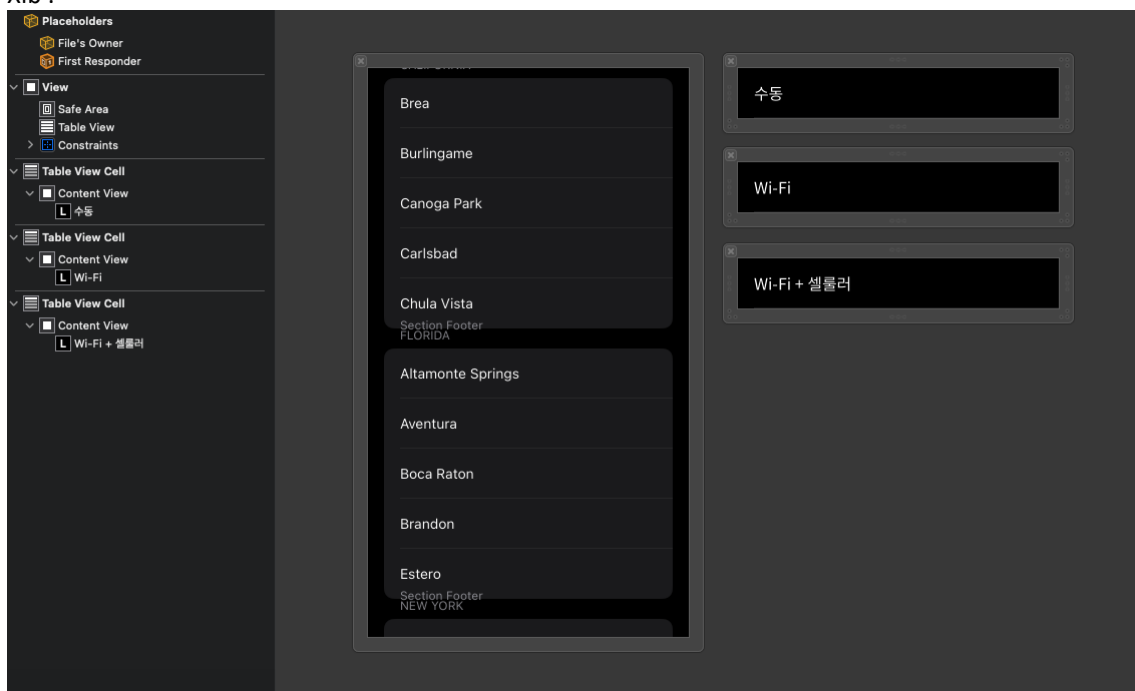


[미디어 관리 - 네트워크 설정]

View(VC) :

https://github.com/njsh4261/SGS_Last_Punch/blob/dev/src/frontend/ios/Snack/ViewControllers/HomeView/Tabs/Settings/Media/NetworkView/NetworkView.swift

Xib :



끝까지 봐주셔서 감사합니다 :D