Got My Favorite

모바일앱프로그래밍 7팀. 2016116811, 황찬웅. 2016110253, 박재화

# **Code**

## SignInViewController.swift

* Line 20 – 31

SignIn을 위한 작은 View의 Top Contraint를 길고 짧게함으로써 view를 띄웠다 없앴다 하며 animated함수를 통해 좀 더 생동감 있게 드러나게 함.

* Line 46 - 52

특정 조건 하에 다른 스토리보드의 viewController로 넘어가기. 위해 self.present()함수를 사용.

* Line 54 – 68

로그인이 유효하지 않을 시, 사용자에게 직관적으로 보일 수 있도록 빨간색이 되며 좌우로 흔들리는 애니메이션.

* Line 83 – 86

키보드를 화면에 띄우지 않는 함수.[1]

## SignUpViewController.swift

SignInViewController.swift에서 사용된 기술들과 크게 다르지 않다.

* Line 22-27

Alpha값을 이용해서 button을 toggle시킴.

## MatchingViewController.swift

로그인 이후 메인화면이라 할 수 있는 부분.

* Line 24-33

매칭 버튼을 눌렀을 때의 이벤트 처리함수. StoryBoard 이동을 구현

* Line 34-40

정했던 취향을 변경하여 칭호를 변경할 수 있는 버튼 이벤트. 앞선 ColorViewController로 View전환

* Line 43-54

ViewDidLoad. 로그인한 사용자의 정보를 Display

## SocketIOManager

서버 통신을 위한 Socket.io swift package manager를 사용한 클래스

* Line 15-25 : 지정된 서버와 연결 및 서버의 데이터 가져오기
* Line 29-44 : 서버 연결, 연결 해제, 메세지 주고 받기 함수

## customtableViewCell.swift

Tableview 내부의 아이템을 정의.

## QutionaryViewControllerr.swift

질문지 작성 viewController에 대응하는 클래스.

* Line 19 - 32

O나 X버튼 입력시, 입력했던 질문과 그 답을 list에 추가한다.

* Line 42 - 74

TableView의 함수를 사용하기 위해 extension함. List의 추가와 삭제에 대한 함수가 구현됨.

## UserModel.swift

사용자 관련 정보를 담은 구조체. Id, pw, 이름 등등을 저장하며 이 앱에 이미 회원가입이 된 유저인지, 또는 회원가입이 유저로 추가해주는 함수가 구현되어있다.

* Line 17-18

User의 main칭호와 sub칭호를 저장.

* Line 19-20

User가 입력한 설문지 질문과 각 답을 저장하는 배열.

* Line 25-28

회원가입 시 유저에 추가하는 함수.

* Line 30-37

로그인 시 유저인지 확인하는 함수

## colorViweController.swift

사용자의 취향에 맞는 색감을 고르도록 한다. 색감에 따른 칭호는 3가지가 있으며 칭호는 랜덤으로 정해진다.

* Line 15 : 색감에 따른 칭호 dictionary
* Line 32 : 칭호를 랜덤으로 부여, segue를 통해 정해진 칭호를 넘겨준다.
* Line 41-51 : arc4random 강제 extension

## mediaViweController.swift

사용자의 취향에 맞는 미디어 컨텐츠의 카테고리를 고르게 하는 것. 총 8가지 카테고리중 2가지를 고를 수 있고 각 카테고리에서 9개중 1가지를 고를 수 있다.

* Line 20-29 : 컨텐츠 선택 수에 따른 DONE 버튼 설정. 2가지를 골라야만 활성화 된다.
* Line 36-47 : MediaThemeViewController, FoodViewController로 segue를 넘겨줌
* Line 55-57 : 버튼 클릭 불가 함수

## MediaThemeViweController.swift

사용자가 고른 카테고리에서 취향에 맞는 컨텐츠를 고르게 하는 것. 컨텐츠에 따른 칭호가 main과 sub로 나뉘어 진다.

* Line 29-44 : 테마에 따른 칭호 dictionary
* Line 71-79 : 고른 테마 수에 따라 main과 sub 칭호 중 한가지를 설정

## foodViweController.swift

사용자의 취향에 맞는 음식을 고른다. 음식명이 칭호로 적용된다. 2가지 음식을 고르며 순서에 따라 메인과 서브가 나뉜다.

* Line 45-47 : 메인 음식을 설정
* Line 47-51 : 서브 음식을 설정
* Line 51-57 : 메인스토리보드로 이동

## QuestionViweController.swift

매칭된 사용자끼리 서로의 질문지를 풀게 하는 뷰. O,X에 관계없이 상대방과 일치하면 초록색이 늘어나 점수가 올라가고 불일치하면 빨간색이 늘어나고 감점 된다.

* Line 61-67 : 상대방의 질문지 로드
* Line 70-80 : 상대방의 정답과 나의 정답의 일치여부 판단
* Line 86-100 : 상대방의 정답과 나의 정답이 일치했을 때 UI 변경 및 점수 획득
* Line 101-115 : 상대방의 정답과 나의 정답이 불일치했을 때 UI 변경 및 점수 감점
* Line 117-134 : 모든 칸이 한가지 색으로 통일 되었을 경우 시각적 효과 부여
* Line 136-160 : 질문이 끝났을 때 뷰 이동과 점수 계산

## ScoreViweController.swift

질문을 주고받은 후의 결과를 보여주고 상대에게 연락처나 메일 주소 등을 보낼지 선택하는 뷰

# **발전 가능성**

* firebase 서비스 api를 활용하여 다양한 사용자를 서버를 통해 관리 할 수 있음
* 1대1 소켓 통신이 아닌 서버 api를 이용해서 진짜 랜덤매칭을 구현할 수 있음
* 정보의 저장.교환에 서버를 사용하여 커뮤니티성을 높일 수 있음
* timer event를 통해 매칭시간을 시각화 할 수 있음

##### References

1. <https://nol-a.tistory.com/5>
2. <https://medium.com/@twih1203/swift-ios-%ED%99%94%EB%A9%B4%EC%A0%84%ED%99%98%ED%95%98%EA%B8%B0-5e5998679d3a>
3. <https://github.com/socketio/socket.io-client-swift>