**프로젝트 진행 보고서(2주차)**

**\* 프로젝트를 위해서 진행한 내용을 간단하게 정리**

1. 기능 구현 목록

|  |
| --- |
| - Firebase 연동  - firebase\_auth를 이용해서 사용자 등록/인증 기능 구현  - 로그인 페이지 UI/UX 구현 (login.dart) |

⦁screens/login.dart

|  |
| --- |
| 소스코드가 매우 긴 관계로 링크로 대체합니다…  https://github.com/SoonMinKwun/history\_of\_studying/blob/main/flutter/meonji/lib/screens/login.dart |

⦁screens/weather\_screen.dart

|  |
| --- |
| …  …  …  import 'package:firebase\_auth/firebase\_auth.dart'; // 사용자 등록/인증 관련  …  …  …  class \_WeatherScreenState extends State<WeatherScreen> {  …  …  …  final \_authentification =        FirebaseAuth.instance; // firebase auth 인스턴스 생성 (변하지 않는 private)    User? loggedUser; // 로그인된 유저  // 유저 정보 불러오기    void getCurrentUser() {      try {        final user =            \_authentification.currentUser; // \_authentification에서 가져온 유저 이름을 삽입        // 로그인 성공시        if (user != null) {          loggedUser = user;          print(loggedUser!.email);        }        // 로그인 실패시      } catch (e) {        print(e);      }    }  …  …  …  @override  void initState() {  …  …  …  getCurrentUser(); // 초기화 될때마다 유저 정보 불러오기  }  …  …  …  @override  Widget build(BuildContext context) {  …  …  …  IconButton(              // 로그아웃 버튼              icon: Icon(                Icons.exit\_to\_app,              ),              onPressed: () {                \_authentification.signOut(); // 로그아웃  Navigator.pushReplacement(                  context,                  MaterialPageRoute(                    builder: (BuildContext context) =>                        LoginSignupScreen(), // 로그인 페이지로 이동                  ),                );              },              iconSize: 40.0,            )  …  …  …  } |

위에 있는 코드는

<https://github.com/SoonMinKwun/history_of_studying/tree/main/flutter/meonji> 에서도 볼 수 있습니다.

- 관련 기술

material : <https://api.flutter.dev/flutter/material/material-library.html>

Firebase : <https://firebase.google.com/>

Google\_fonts : <https://pub.dev/packages/google_fonts>

firebase\_auth : <https://pub.dev/packages/firebase_auth>