

BLE + Mesh Networking

👤 생성자	👤 도경원
📅 작성일	@2025년 4월 22일

BLE (Bluetooth Low Energy) + Mesh Networking = ?

- BLE는 기본적으로 **1:1 통신**을 지원하는 저전력 무선 통신
- 여기에 Mesh 구조를 적용하면 → BLE끼리 연결된 기기들이 **릴레이 방식으로 메시지 전달** 가능

즉, **Bluetooth를 이용해 만든 탈중앙형 오프라인 메시징** ex) 스마트홈, 군사/재난, 차량 통신(자율주행차 간 실시간 정보 공유)

참고앱 분석

1. Bridgefy

- BLE 기반의 오프라인 채팅 앱
- Store-and-Forward 방식으로 메시지 릴레이 (저장하고 있다가 연결되면 보내는)
- **엔드투엔드(E2EE)** 암호화 있음 → 암호화 우리도 필수 일듯 !

2. FireChat

- BLE + WiFi Direct 사용
- Mesh 구조 활용하여 넓은 범위 메시지 전달
- 보안 미비와 수익화 실패로 종료

Flutter에서 Native 연동이 필요한 이유

BLE 자체는 Flutter에서도 쉽게 쓸 수 있지만,

"Mesh Networking"은 BLE 스펙 중에서도 고급 기능에 속하기 때문에 대부분의 기능은 Android/iOS의 ****Native SDK (Java/Kotlin or Swift/Obj-C)****를 통해서만 사용할 수 있음.

1. Flutter에서 BLE 기본 사용은 가능

- 패키지: `flutter_blue_plus`, `flutter_reactive_ble` 등

- 기능: BLE 주변기기 검색, 연결, 데이터 송수신

2. Mesh Networking은 Native SDK 사용이 필요

- Android: Nordic, Silicon Labs, Telink 등에서 제공하는 **BLE Mesh SDK**
- iOS: Apple은 공식 Mesh SDK는 없음 → 커스텀 Native 구현 필요
- → 따라서 Flutter에서 이 SDK를 연동해야 함
-

Nordic BLE Mesh SDK 방식 × Platform Channel (Flutter는 "컨트롤러" 역할만 함)

→ 이 방식 사용할듯

- Flutter는 "컨트롤러" 역할만 함
- 실제 BLE Mesh 통신은 Android 쪽이 주도
- **Platform Channel**이 필수이고, 반복적으로 사용됨

Custom BLE Mesh 방식 ("Flutter 안에서 다 해보겠다!"는 접근)

- 대부분의 BLE 기능을 **Flutter 안에서 처리함** (flutter_blue_plus 같은 플러그인 활용)
- 메시지 송수신, 중계 등 로직도 Flutter로 직접 구현

음성메시지

인터넷 없어서 음성통화는 BLE 대역폭 상 불가능함

- **음성 녹음:** Flutter에서 사용할 수 있는 플러그인 중 record , audio_recorder , flutter_sound 등을 활용해 사용자의 음성을 녹음
- **파일 저장:** 녹음된 음성은 .aac , .m4a , .wav 등의 포맷으로 기기 내 임시 디렉터리에 저장
- **BLE 전송:** BLE의 데이터 전송 한계를 고려하여 음성 파일을 작은 청크(조각) 단위로 분할하고, 순차적으로 전송
- **수신 및 재조립:** 상대 기기는 청크를 수신하여 원래의 음성 파일로 합쳐 저장
- **음성 재생:** 재생은 just_audio 또는 audioplayers 와 같은 Flutter 플러그인을 통해 구현하며, 자동 또는 수동 재생이 가능하도록 선택

기술적 고려사항

BLE는 기본적으로 데이터 전송량이 적기 때문에, 음성 메시지는 **10~15초 이내**의 짧은 길이로 제한하는 것이 안정적인 전달에 유리하다. 음성 데이터를 나눠 전송하는 만큼, **청크 누락**이나 **순서 오류**를 방지하기 위한 재전송 로직도 함께 고려해야 한다. 또한, 초기 단계에서는 암호화 없이 구현하되, 추후에는 Base64 인코딩 및 AES 기반 암호화 등 보안 기능을 확장