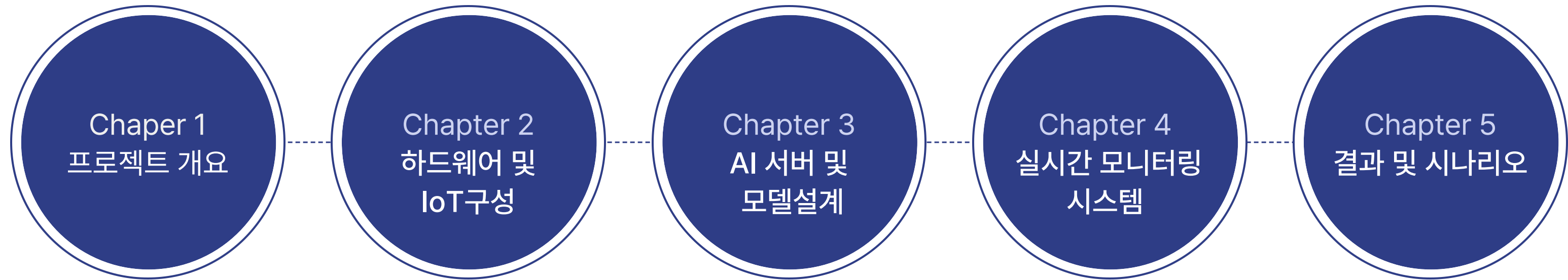
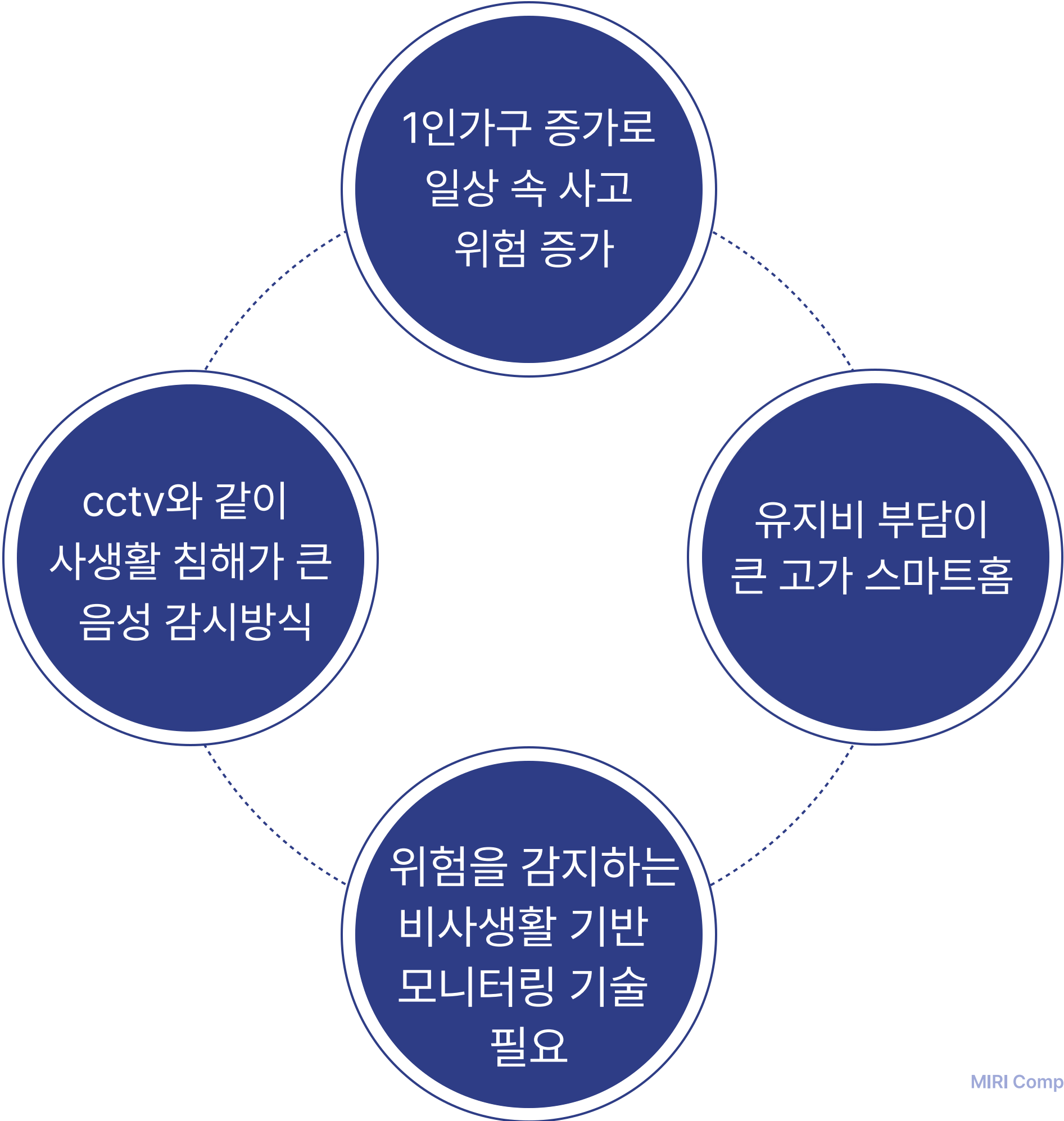


AI Life Solution Challenge

사생활 침해 없는 저비용 1인 가구 위험 알림 AI



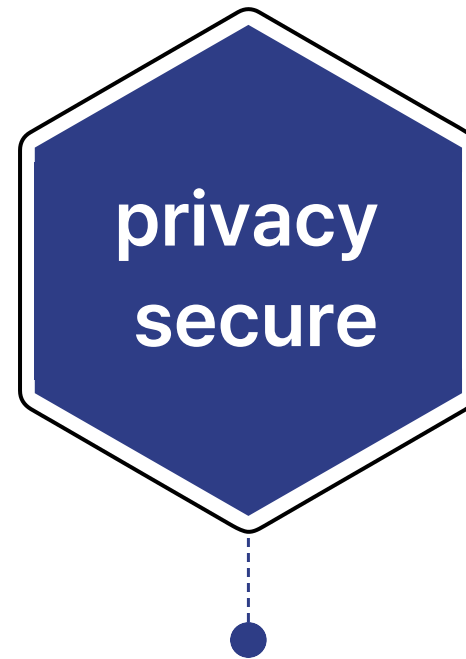
문제배경(Background)



시스템 개발 목적 (Project Goal)



저비용 센서
(FSR·PIR·Reed)만으로
실시간 행동 패턴 파악



영상, 음성을 사용하지
않아 사생활이 보호됨



누구나 설치 가능한
저비용·고확장성
안전 시스템 구축

전체 시스템 architecture

센서노드
ESP32

- 압력(FSR)
- 문감지(Reed)
- 움직임(PIR)
- ESP-NOW를 통해
허브에 실시간 전송

허브
ESP32

- 각 노드 데이터 수신
- FAST API 서버로 데이터 전달
- blynk 통한 기본 모니터링

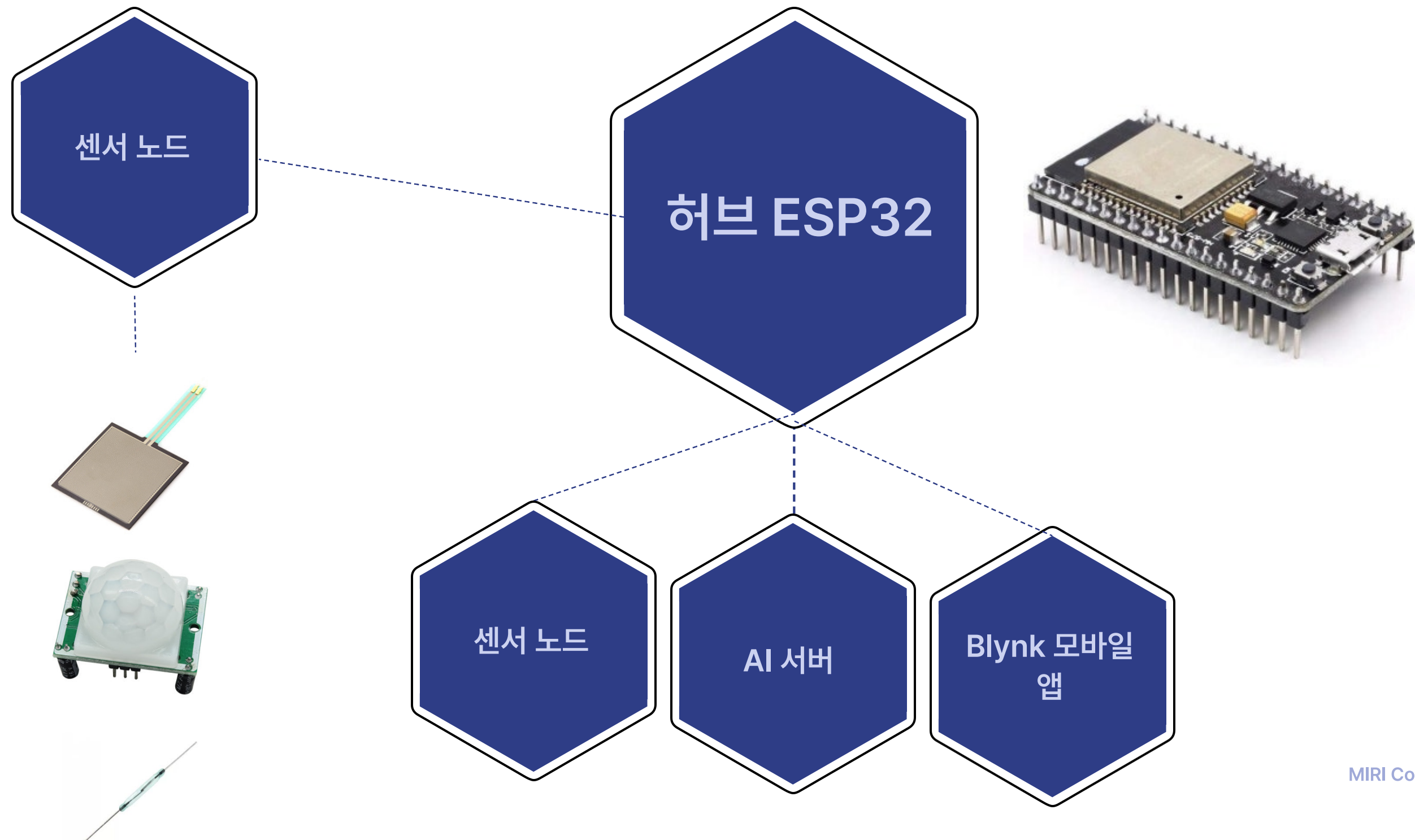
AI 서버

- 실시간 데이터 기반으로
Normal/Danger분류
- 위험점수(risk score)계산

대시보드
streamlit

- 실시간 센서 변화 그래프
- 현재 상태 표시 및
데이터 로그 시각화

하드웨어 및 IoT 구성 (Sensor & ESP32 Network)



03 | AI 서버 및 모델 설계 (AI Model & FastAPI)

1) AI 학습 데이터 (7일 생활 패턴 기반)

PIR(거실 움직임), FSR(침대/욕실/식탁/현관 압력), 리드스위치(욕실문·현관문)
센서별 7일·초 단위 시계열 데이터 → 단일 테이블로 통합

Feature Engineering 적용

시간 특징: hour, weekday, 전/후 30초 변화량
관계 특징: PIR 부재 + 욕실문 닫힘, 침대 지속 압력 등

라벨 생성 방식

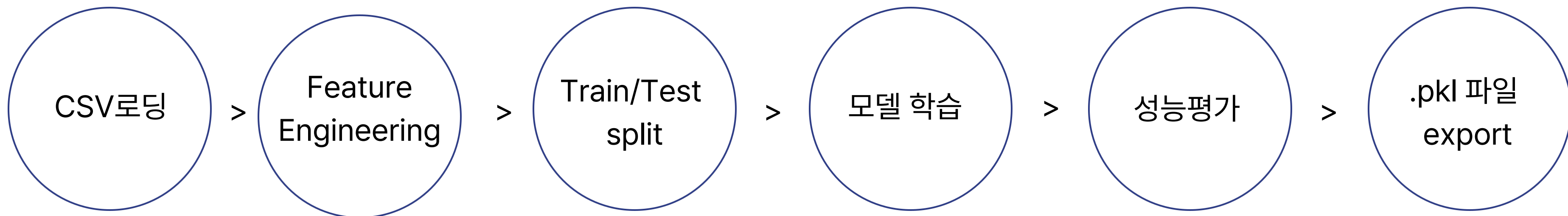
정상 생활 패턴: 반복적, 짧은 센서 변화
위험 패턴: 욕실 장시간 체류, 무움직임 지속 등

03 | AI 서버 및 모델 설계 (AI Model & FastAPI)

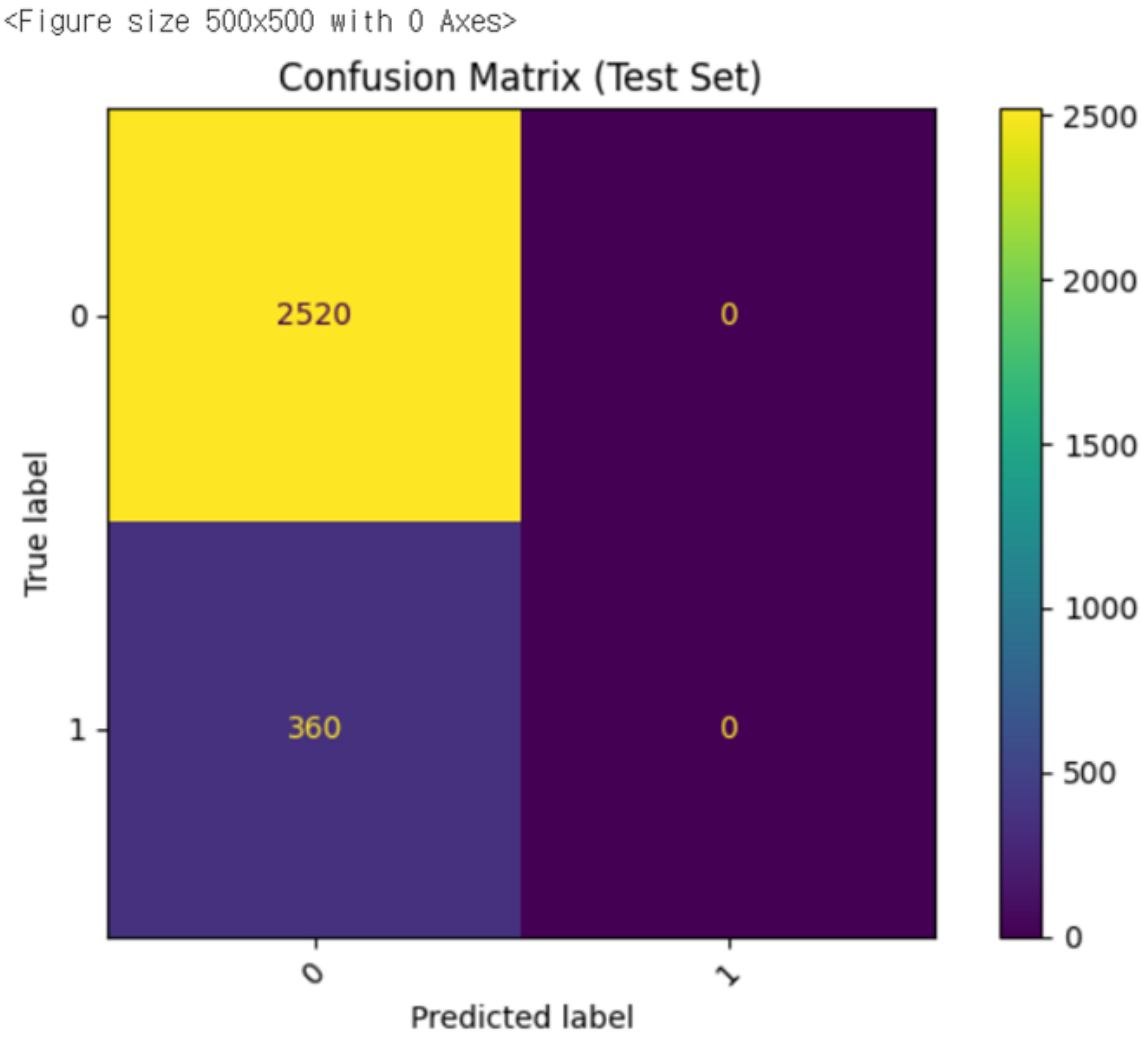
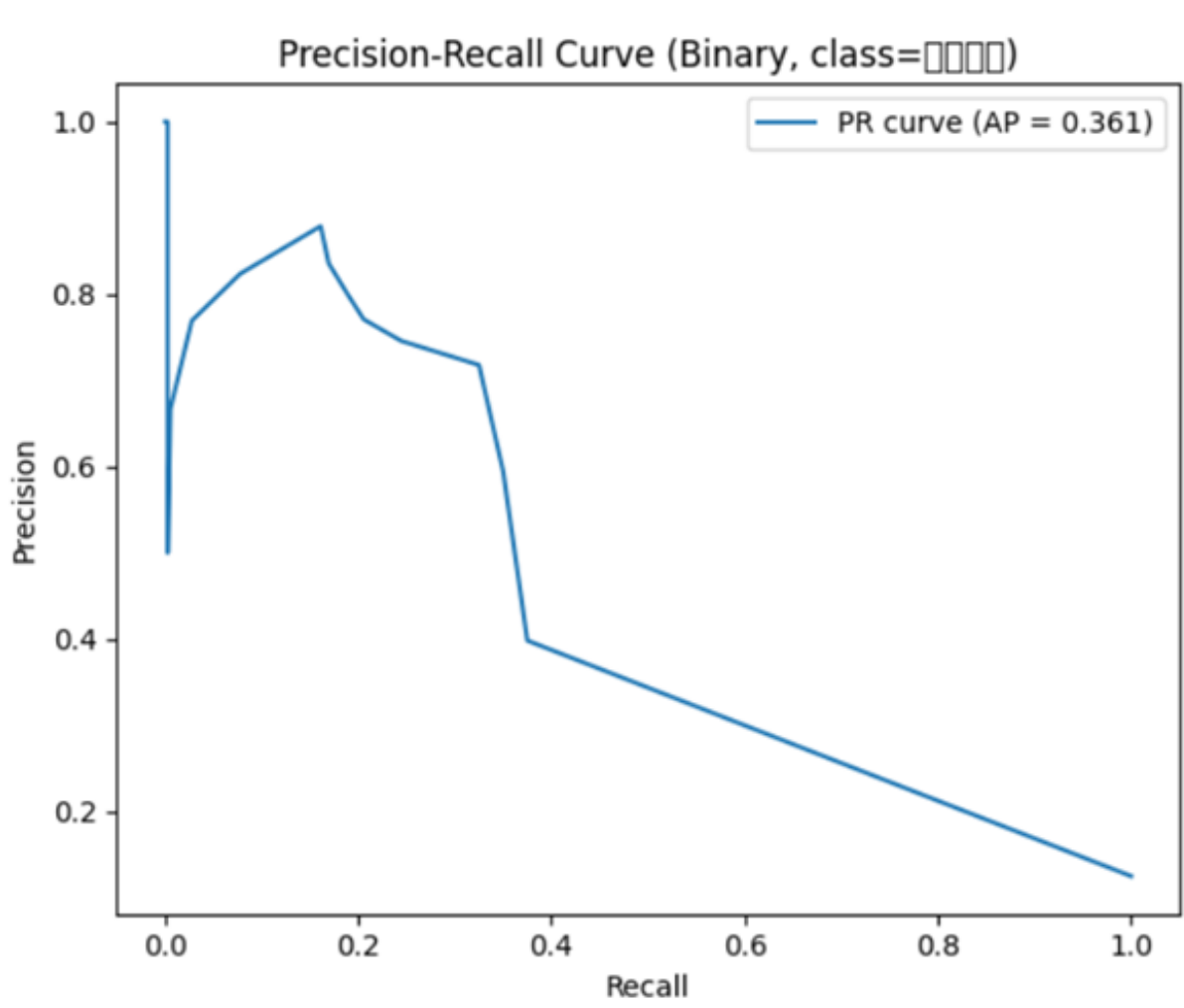
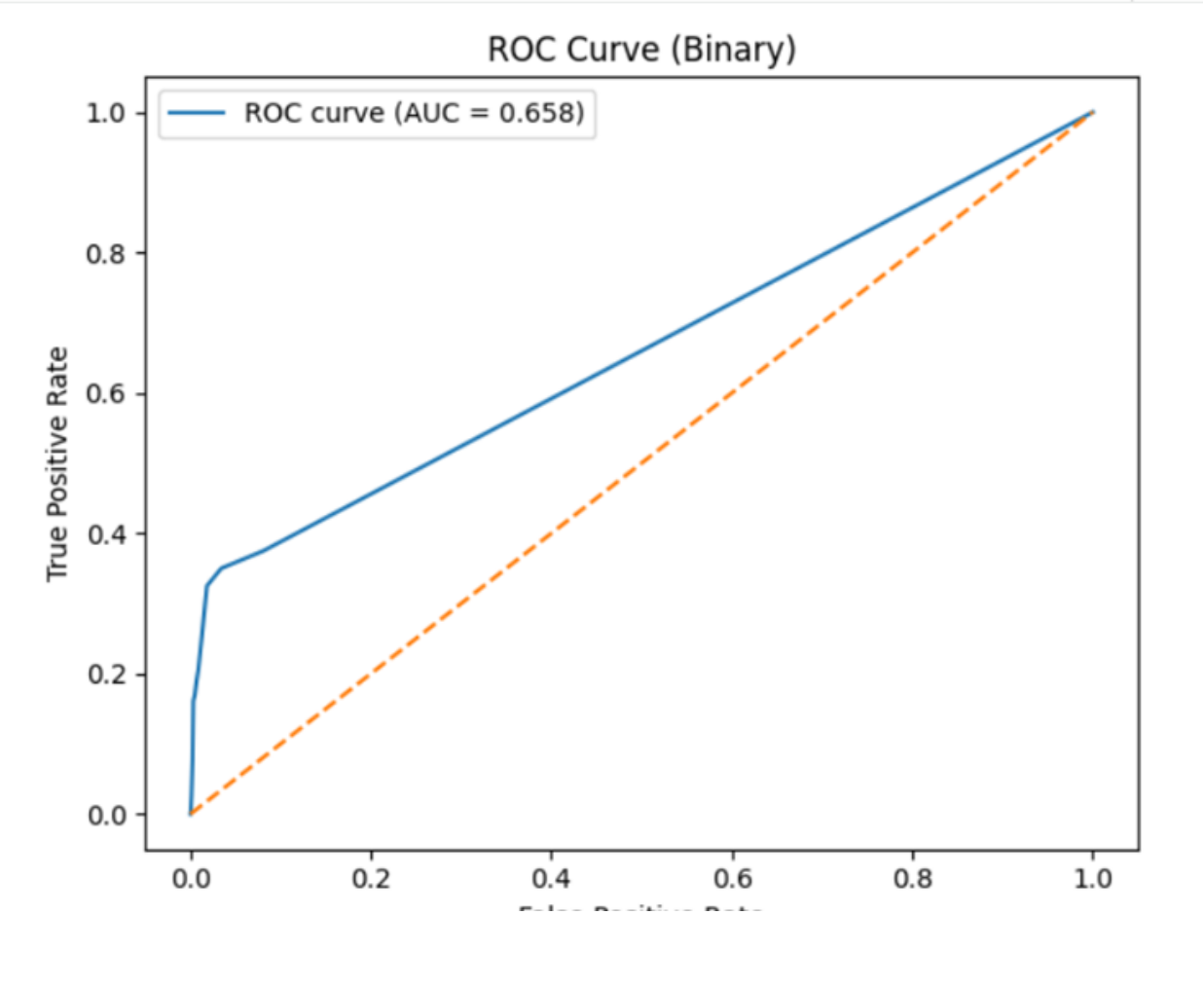
2) 모델 구성(AI_life_pattern.ipynb)

- 데이터 스케일링(표준화)
- Feature set 30+ 생성(센서조합, 패턴 기반)
- RandomForest/ 딥러닝 모델 비교 후 최적 모델 선정
- Grid Search 로 파라미터 튜닝
- 최종 모델 및 스케일러를 pkl 파일로 저장 →FastAPI 서버에 탑재

전체 파이프라인



3) 모델 성능



-ROC AUC = 0.658

-완전 랜덤(0.5) 보다 의미있게 높은 분류 성능

-PR AUC = 0.361

-위험 클래스 데이터 적은 상황에서도 일정수준의 탐지능력 확보

-정상 클래스(0)는 매우 안정적으로 예측

-위험 클래스(1)는 일부 미탐지 발생
이는 데이터 수 부족에서 기인

03 | AI 서버 및 모델 설계 (AI Model & FastAPI)

4) FastAPI 기반 실시간 AI 추론 서버 구조

노트북에서
FastAPI 서버 실행

- 학습된 .pkl 모델+스케일러 로드
 - SensorInput 스키마로
입력 Json 검증

/predict 엔드포인트

- 허브 ESP32 에서 HTTP POST로
센서 상태 전송
- 동일한 Feature Scaling 적용 후
위험여부 예측
- status(NORMAL/DANGER), risk_score(0-1) 반환

로그&대시보드 연동

- 예측 결과를 realtime_log.csv에
기록
- Streamlit 대시보드가 이를 읽어
실시간 상태, 그래프 표시

03 | AI 서버 및 모델 설계 (AI Model & FastAPI)

5) 생활 패턴 기반 위험 탐지 시나리오-----대표적인 3개의 예시

욕실 사고 위험 시나리오

- 욕실 문(Read_bathroom) 닫힘 상태 지속
- 동시에 PIR living 움직임 없음+FSR_bathroom 변화 거의 없음
- 모델이 욕실 장기 체류 패턴으로 인식하여 높은 risk_score 출력

침대에서의 실신패턴

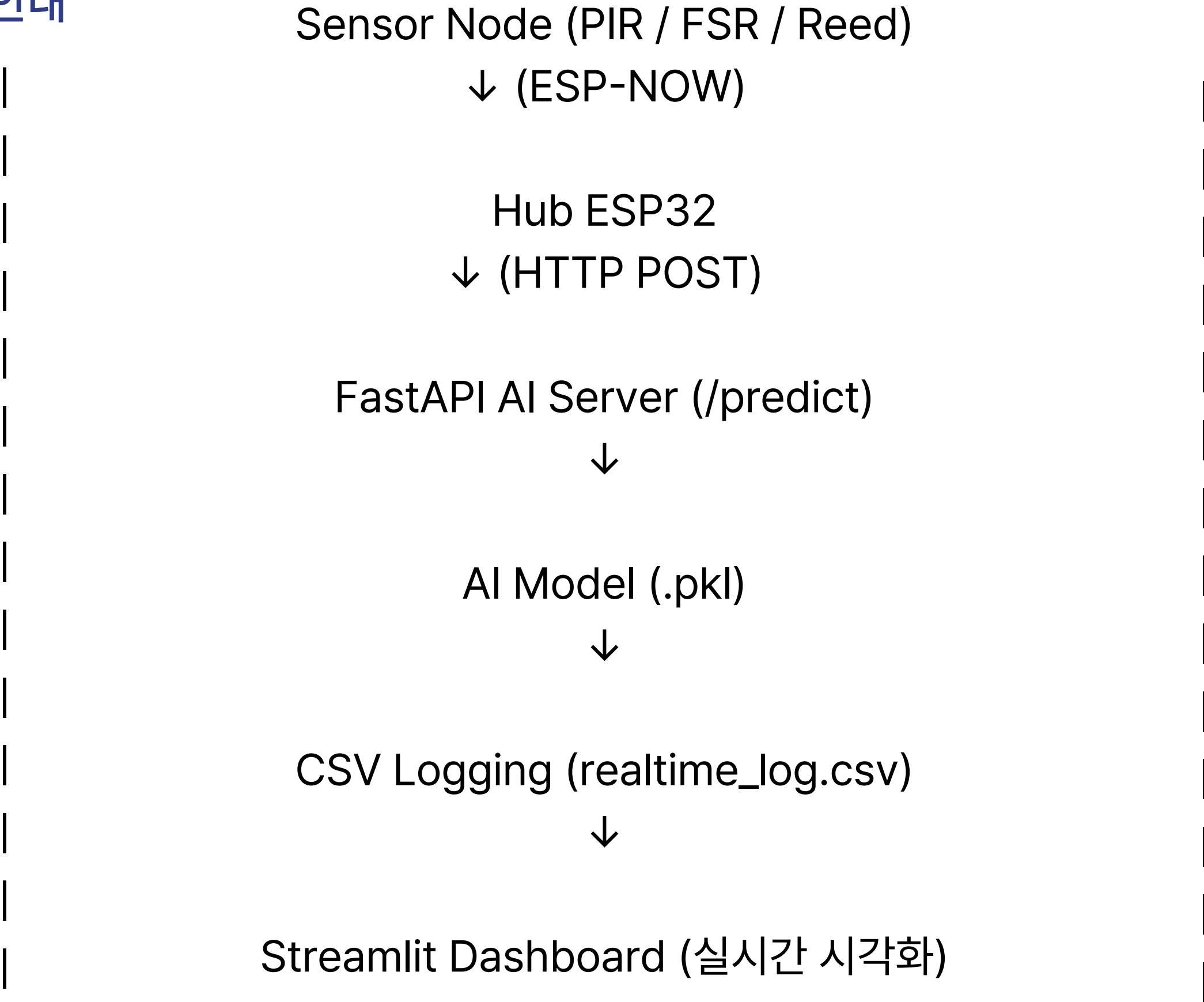
- FSR_bed 압력 값이 오랫동안 유지
- 거실 PIR, 현관 센서 변화 거의 없음
- 수면, 휴식 패턴을 벗어난 비정상정지 상태로 판단→ 위험 알람 후보

귀가, 외출 패턴 인지

- Reed_entrance+FSR_entrance 변화
- 시간대(weekday/hour) 정보와 함께 분석
- 사용자의 귀가, 외출 규칙성을 학습하여 이상 시간대 움직임 시 추가 위험 반영

04 | 실시간 모니터링 시스템

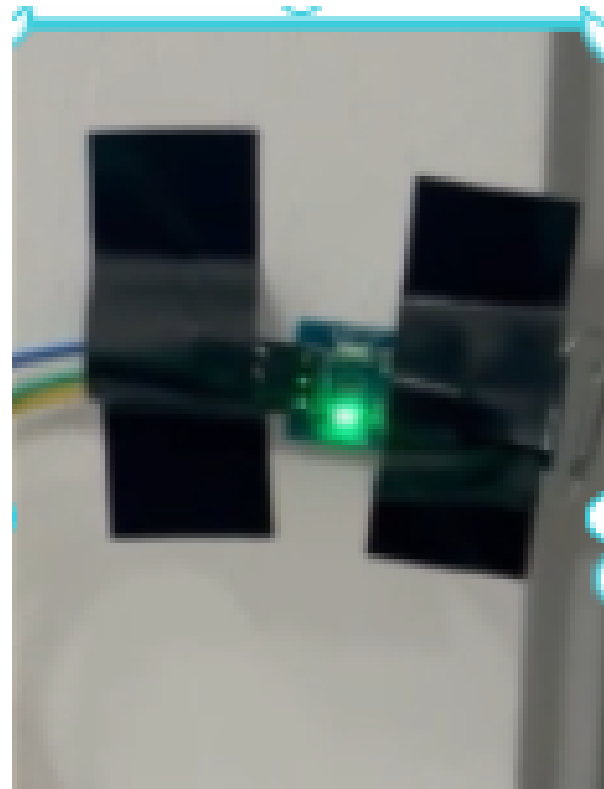
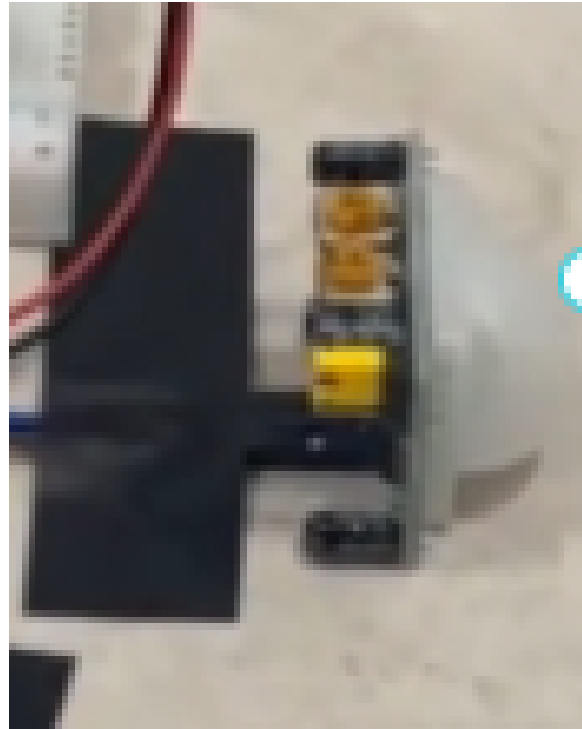
1) 시스템 흐름도 구성안내



04 | 실시간 모니터링 시스템

2) 센서 Node 구조 & 허브 ESP32역할

센서 Node 구조



각 센서는 독립 ESP32 Node로 구성
0/1 신호로 단순화

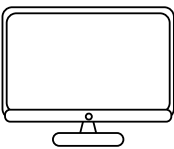
허브 ESP32



모든 Node 데이터를 수신 후
상태 변수(PIR/FSR/Reed)로 통합 관리
최신 상태를 FastAPI /predict에 JSON으로 전송

04 | 실시간 모니터링 시스템

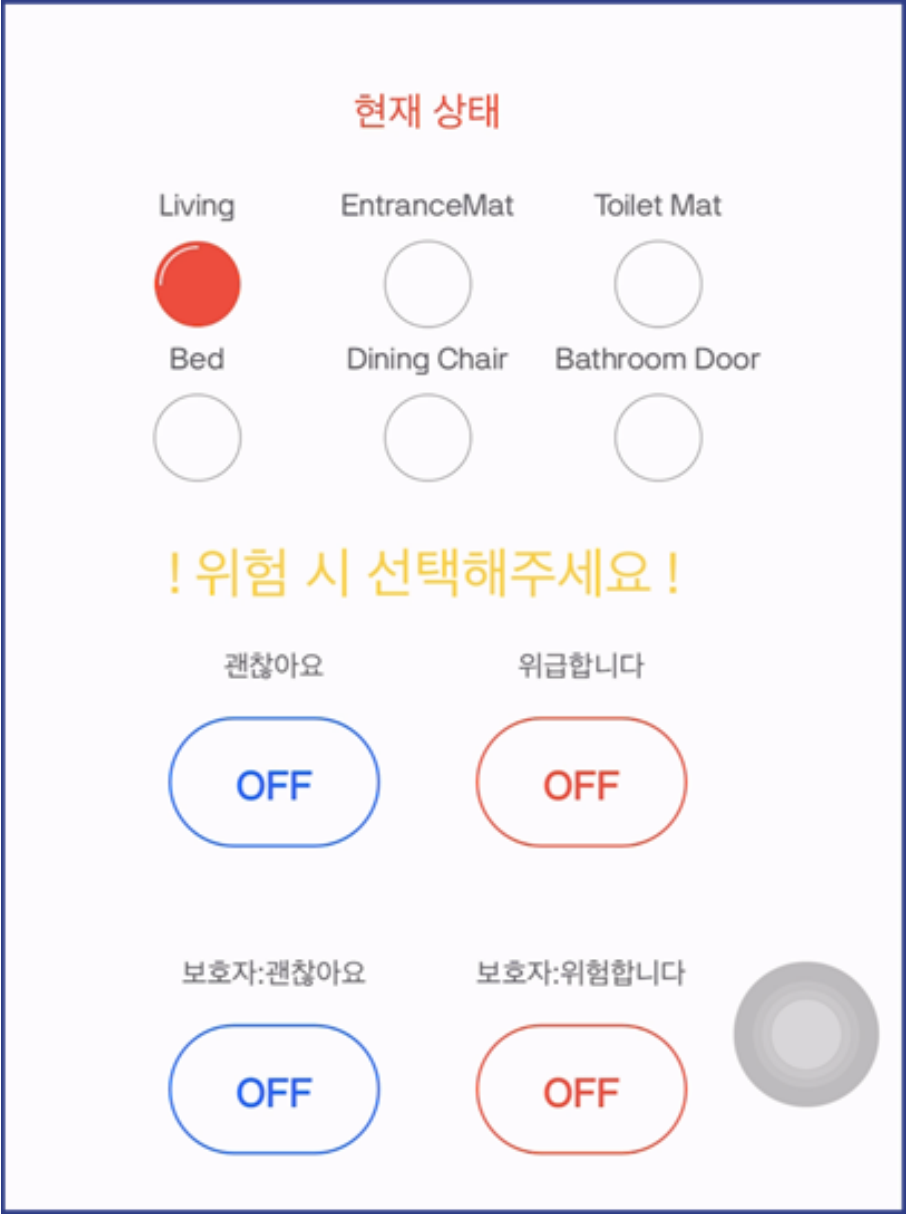
3) 실시간 모니터링 UI(사용자앱+ 대시보드 통합)



Streamlit Dashboard

실시간 데이터 로그 기반 시각화로 자동갱신
AI 추론 결과 및 위험확률 실시간 모니터링
Human-in-the-loop 기반

Blynk(즉각 대응)+Streamlit(정밀 분석)
사용자 편의와 AI 고도화를 동시에 잡는
투트랙(Two-Track) 관제시스템

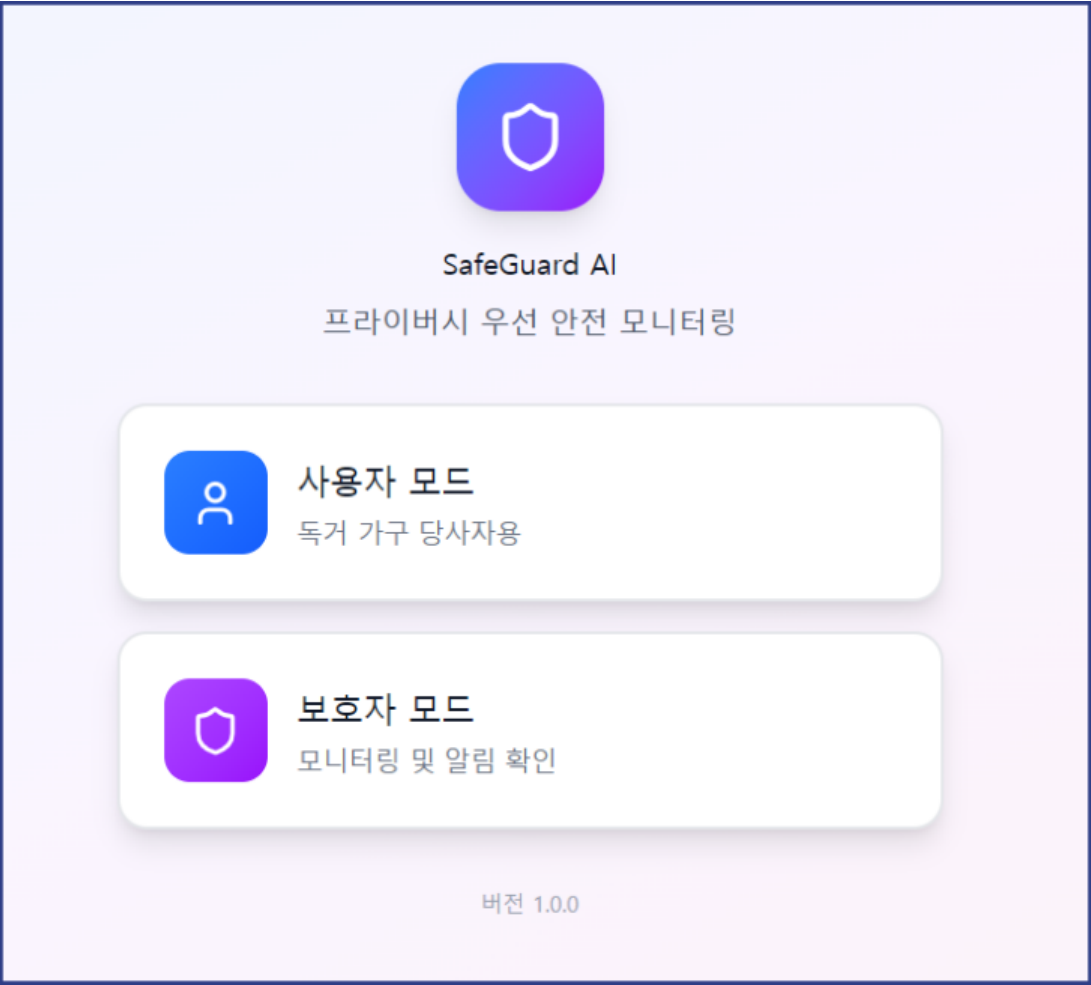


Blynk 앱

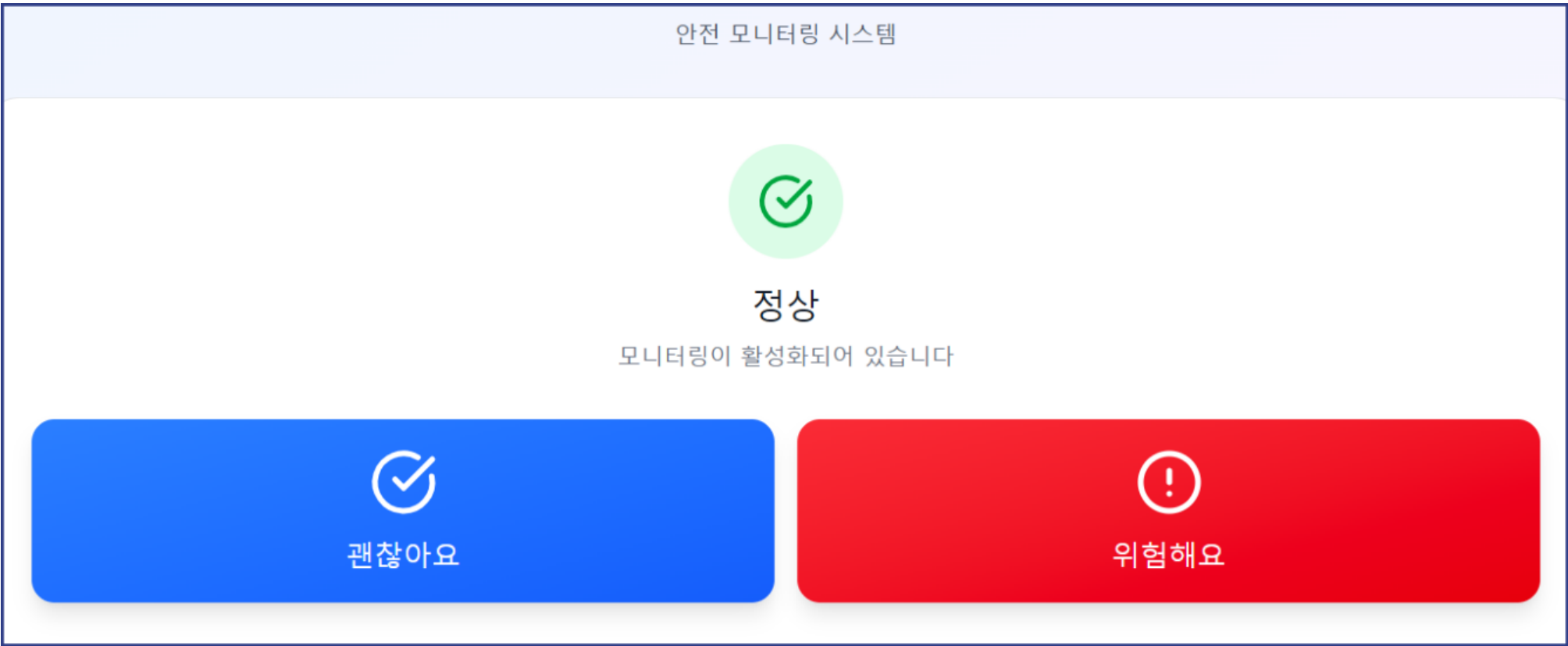
End-User용 모바일 인터페이스 구현
IoT 엣지 디바이스와 저지연(low-latency) 데이터 연동

04 | 실시간 모니터링 시스템

4) Figma 기반 상용 서비스 디자인

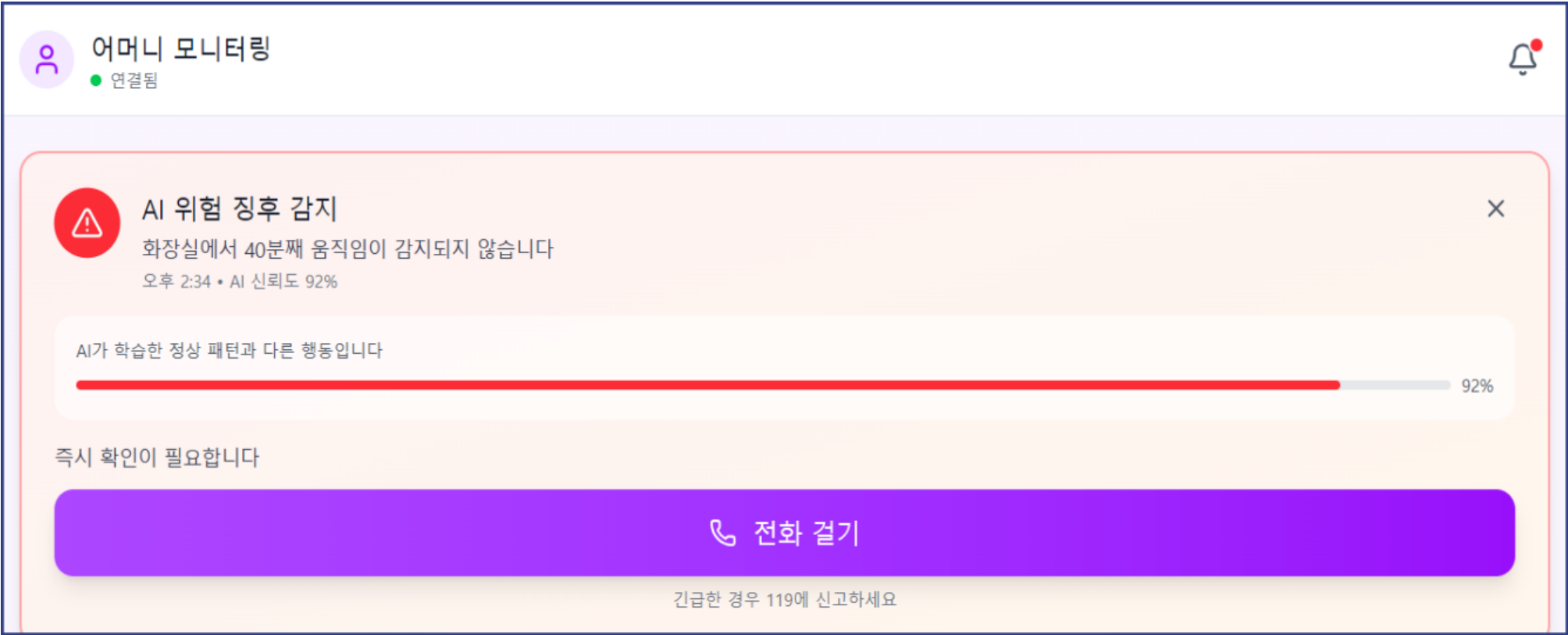


상용 서비스 수준의 UX까지 포함해
실제 돌봄 서비스로 확장 가능한 방향을 설계




사용자모드

보호자 모드





정상/비정상 패턴 실제 테스트 사진

<<

 AI 모델 관리

AI가 잘못 판단했을 때, 아래 버튼으로 신고하면 AI가 더 똑똑해집니다.

 피드백 반영하여 재학습 시작



AI 생활패턴 위험 감지 모니터링

현재 상태: **NORMAL**

감지 시간: 2025-12-02 01:59:15

판단이 틀렸나요?

네, 이건 '정상'입니다

네, 이건 '위험'입니다

PIR (거실)

0

Bath 압력


273.93220...

Bed 압력

0.0

위험 확률

21.3%



욕실/침대/거실 센서 값이 일상 패턴대로 변화
PIR 움직임 존재 → 정상 활동
위험도 낮음(예: 20%)
모델이 정상 패턴을 안정적으로 학습한 증거

위험 감지 사례



case2) 침대 실신 위험
FSR_bed = 지속 고정값
PIR_living = 0
→ AI 판단: 비정상 무반응

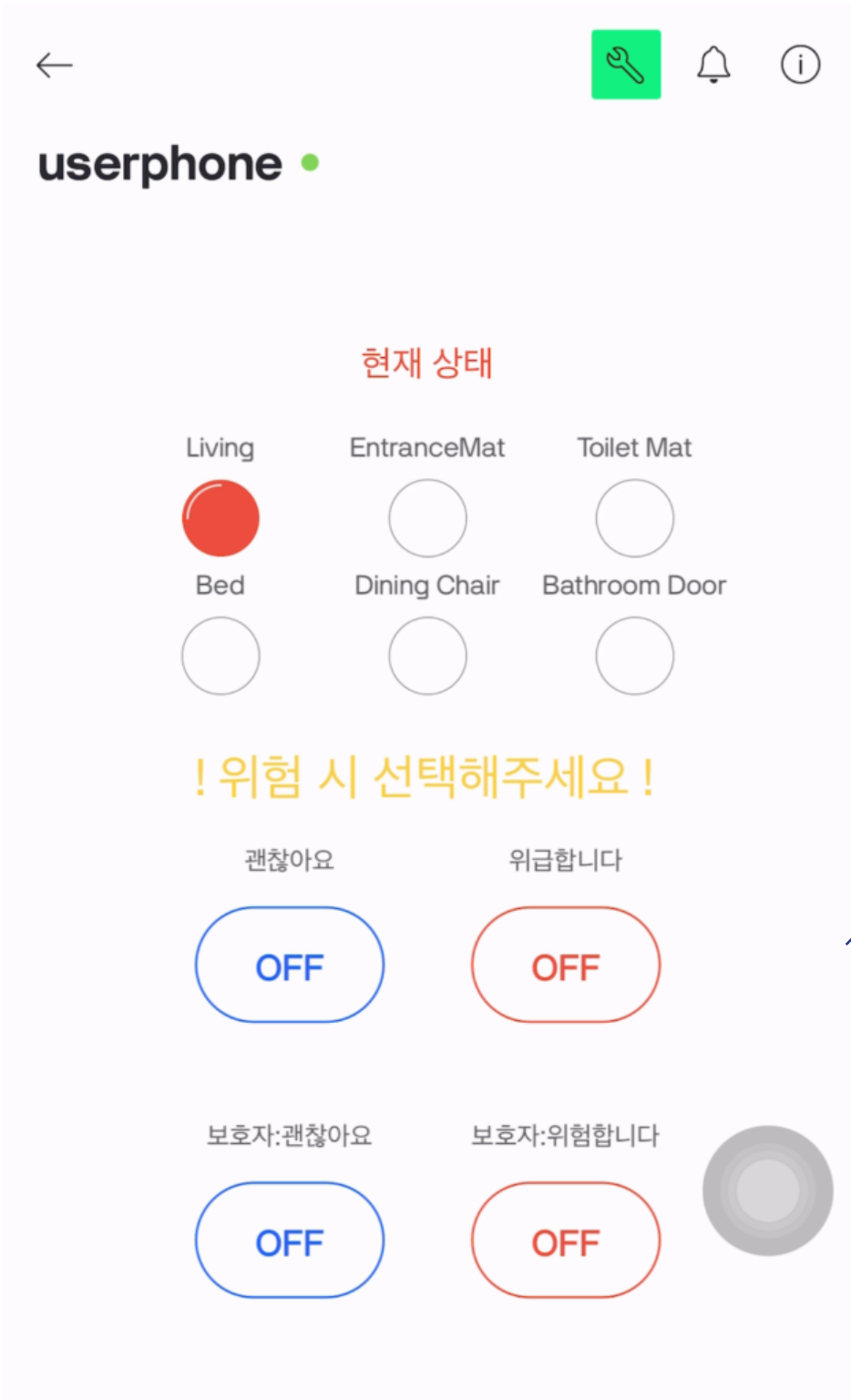


Case1) 욕실 사고 위험
Door_bathroom = 1(닫힘 지속)
PIR_living = 0
FSR_bathroom = 유지
→ AI 판단: 장기 체류 위험(DANGER)



case3) 외출/귀가 패턴 이상
Reed_entrance 움직임 있음
시간대가 평소와 다름
→ AI 판단: 생활 패턴 이상

실제 사용자 앱(Blynk)에서의 반응



실제 센서 값이 스마트폰에서 즉시 반영됨

위험 상황 시 사용자가 “괜찮아요/위험해요” 선택

피드백 기반 AI 개선 가능(Human-in-the-loop)

시스템 장점 및 한계

장점

저비용 센서만으로
고위험 상황을 조기에
탐지할 수 있는 기술
적 기반 확보

프라이버시 보호
카메라 없이 위험 감지
→ 거부감↓ 수용성↑

고확장성 IoT 구조
(ESP32 + Blynk)

한계점

실제 생활 데이터를
장기간 수집하기
어려워 시나리오 기반
synthetic data 설계
가 필요

센서 종류·민감도가 달
라 발생하는 노이즈·동
기화 차이를 조정하는
과정이 복잡

FSR 센서 간 민감도 차
이로 인해 threshold
기반 분류가 어려워,
scaling 재조정이 반복
적으로 요구됨

감사합니다