

# 감정 인식 모델과 GPT를 통한 가족 갈등 해결 AI 스마트 홈캠 “솔버(Solver)”

*CONCAT*

정서윤(팀장), 김소연, 김윤아, 조혜은, 최윤서

# 목차

*Table of Contents*

## 1 프로젝트 개요

| 개요 | 배경 | 기능 | 기대효과 |

## 2 피드백 및 수정 사항

| 피드백 받은 부분 | 수정 & 추가 사항 |

## 4 프로젝트 설계

| 요구사항정의서 | 구성도 | 흐름도 |  
유스케이스 | 알고리즘 명세서 |

## 5 사용한 기술

| 개발환경 | S/W 기술 | H/W 기술 | DB |

## 3 페르소나와 솔버 작동 시나리오

| 페르소나 인물 | 솔버 작동 시나리오 |



# 팀원 역할

## 정서윤

### - 프로젝트 총괄

: notion, github를 이용한 프로젝트 관리, 하드웨어 기기 모색 주문.

### - 하드웨어

: 라즈베리파이 5+ 웹카메라를 이용한 감정, 갈등 인식 솔루션 도출 기기 개발, 라즈베리파이 초반 환경설정, 주행 기기 YOLO 객체 감지 알고리즘 추가.

### - 인공지능 모델 개발

: 이미지 표정 기반, 음성 톤 멜 스펙트로그램 감정 인식 모델 개발.

### - 웹/DB

: 솔루션 기능(저장된 영상, 솔루션 목록/검색/확인), 감정 통계 기능, 대화의 재구성 기능, 메인 페이지 기능(실시간 영상 송출, 대화 내용 확인), 프론트 개발, 배포.

## 김소연

### - 디자인

: 피그마를 활용해 웹 페이지 디자인 및 레이아웃 설계

- 상호작용 가능한 프로토타입을 제작 및 피드백 반영을 통한 디자인 개선

- 프로젝트의 핵심 가치를 담은 로고 제작

- 웹 디자인 스타일 가이드(버튼, 아이콘, 색상 코드 등) 개발팀과 공유

## 김윤아

### - 웹

: 휴지통 기능(영상 삭제/목록 확인/복구/영구 삭제/30일 이후 자동 삭제), 로그인/회원가입 시 네이버 SMTP 서버를 활용한 이메일 인증(인증코드 발송/인증), 로그인/회원가입 시 비밀번호 해싱, 아이디/비밀번호 찾기, CoolSMS API를 활용한 전화번호 인증(인증코드 발송/인증) 및 AWS S3와 연동한 알림 문자 발송 기능, 프론트 개발

### - DB

: Supabase 데이터베이스 연동

### - 인공지능 모델 개발

: 텍스트를 이용한 감정 인식 개발

## 조혜은

### - 웹

: sns 로그인/ 회원가입 기능, 회원정보 수정 기능(일반 로그인 / SNS 로그인), 로그아웃 기능, 회원탈퇴 기능, 감정대화훈련 기능, 해결책 TTS 기능, 프론트 개발

- Naver Clova Speech API를 활용한 STT, 화자분리 개발

## 최윤서

### - 웹

: 로그인 기능, 회원가입 기능(이름/아이디 중복 검사/비밀번호 확인/비밀번호 일치 검사/이메일 유효성 검사), 이용약관 기능, 프론트 개발

### - 하드웨어

: 주행 기기 제작, Lidar 데이터 처리 및 장애물 회피 기능 개발, 라즈베리파이 4+웹 카메라를 이용한 객체 감지 로직 개발, 모터 제어 및 인터페이스 구현

# 1. 프로젝트 개요



“

감정인식모델과 GPT를 활용하여  
가정에서 발생하는  
갈등/감정을 실시간으로 감지하고,  
갈등을 해소할 수 있는 해결책을 제시하는  
AI 기반의 스마트 홈캠 로봇과 웹 서비스

”

가정 내 갈등을 경험하는 가족 구성원 간의  
갈등을 분석하고 해결책을 제공하여  
가족 관계 개선과 정서적 안정에 기여하는 것을 목표

# 1. 프로젝트 배경



## 1. 사회적 문제의 증가

- 청소년의 가족 문제와 관련된 상담 요청이 증가하고 있으며, 이는 가정 내 불화와 갈등이 심각해지고 있음을 나타낸다.
- 가족 구성원 간의 소통 문제와 감정 조절 장애가 증가하면서 이를 해결할 방법이 필요해지고 있음.

## 2. 감정 조절 장애 진료 건수의 증가

- KCD F63.8(감정 조절 장애) 진료 건수는 2015년부터 2020년까지 꾸준히 증가.
- 감정 조절 문제를 겪는 사람들이 많아지면서 관련된 지원과 중재 시스템의 필요성이 더욱 커짐.

## 3. 가족 갈등의 심각성

- 가정 내 문제는 장기적으로 심리적 문제나 더 큰 갈등으로 발전할 수 있으며, 이에 대한 즉각적인 해결책이 중요.
- 가족 내 갈등은 단순히 구성원 개개인에 영향을 미치는 것이 아니라, 사회 전반에 걸쳐 부정적인 영향을 미칠 수 있음.

## 4. AI 기반 솔루션의 확산 가능성

- AI가 다양한 사회적 문제 해결에 사용되고 있는 만큼, 가족 간의 감정적 갈등을 다루는 AI 솔루션도 자연스럽게 채택될 가능성이 있음.
- 법적, 사회적 영역에서의 AI 사용 확대는 가정 내에서도 AI가 갈등을 완화하고 해결책을 제시하는 데 큰 역할을 할 수 있음을 시사.



# 1. 프로젝트 주요 기능

## 1. 실시간 감정 인식

“음성 톤”, “얼굴 표정”, “대화 내용”  
기반 3가지 감정 인식 모델 제작.

## 2. 갈등 상황 파악

웹 카메라 통해 “음성 톤”,  
“얼굴 표정”, “대화 내용” 기반  
멀티 감정 인식 모델 활용한 갈등  
감지.

## 3. GPT를 활용한 맞춤형 갈등 해결 방법 솔루션 제공

생성형 AI인 GPT API를 활용한  
맞춤형 해결책 제공.

## 4. 갈등 해결 방법 생성 알림 문자 서비스 제공

갈등에 대한 해결 방법이 생성되었다는  
문자 알림 서비스 제공.

## 5. 갈등 해결 방법 AI 영상, 참여자별 행동 가이드 제공

연동된 웹 플랫폼을 통해 맞춤형 갈등  
해결 방법 솔루션 영상 & 텍스트 확인.

## 6. 반성 및 학습을 위한 교육 웹 페이지

- (1) 챗봇음성대화훈련
- (2) 지난 갈등의 감정 통계
- (3) 지난 갈등의 대화 재구성

## 7. 객체(사람) 감지, 장애물 회피 주행

사람이 있는 곳으로 이동하여  
모니터링을 잘 수행하기 위하여 주행.



다양한 인공지능 기술과  
이미지, 영상, 오디오 처리 기술을  
활용하여 개발 수행

# 1. 예상 기대 효과



## 가정 내 효과

### → 가정 내 갈등 해결 및 관계 개선

: 구성원 간의 감정을 이해하고 공감할 수 있도록 돋고 더 나은 의사소통 환경 조성.

: **근본적인 문제를 해결** 할 수 있는 방안 제시하여 문제 해결.

### → 가정 내 갈등을 예방하고 건강한 대화 문화를 형성하는 데 큰 기여

### → 시간과 비용 절약

: 상담소 방문이나 전문가 개입 없이도 초기 단계의 갈등을 해결할 수 있어 **시간, 비용 절약 가능**

: 심리적 안정감을 제공하여 외부 도움 없이 **자립적으로 문제 해결할 수 있는 기회** 제공



## 사회적 효과

### → 가정 갈등으로 인한 사회적 비용 절감

: 가정 내 갈등이 심화되어 상담소, 법적 분쟁, 이혼 등의 문제로 이어지는 경우를 줄여 **사회적 비용 절감 가능**.

### → 정서적 건강 증진과 사회 안정성 강화

: 구성원들의 정서적 안정이 가정에만 국한되지 않고 직장, 학교 등 사회적 관계에서도 긍정적 영향.

: **가정에서의 안정적인 관계는 사회 전체의 안정성을 높이는 데 기여.**

### → 교육 및 갈등 관리 문화 확산

: 감정 관리 및 갈등 해결 방식을 제공함으로써, 효과적인 갈등 해결 방법이 사회 전반이 퍼질 수 있음.

: **가정에서 배운 효과적인 의사소통 방식이 학교나 직장 등 다양한 사회적 환경에서 활용될 수 있음.**

## 2. 피드백 주요 반영 사항

### 1. 솔루션 관련 피드백

- 갈등 해결 방법 텍스트를 두세 줄로 짧게 보여지도록 하기
- 영상 또는 가족 사진으로 대체



### → 피드백을 참고한 반영 사항

- AI 영상을 이용한 갈등 해결 방법 영상 자동 생성.
- 명상 또는 음악 영상 재생.
- 솔루션에 대한 갈등 피드백 텍스트를 최대한 시각화 요소로 대체

### 2. 주행기기 관련 피드백

- 기기의 무게중심 문제로 몸체가 흔들리는 문제로 기기 하단에 무게중심을 단단히 하기 위해 무게가 나가는 물체 부착.
- 주행기기의 시작과 끝 명확히.
- 기기 양 옆으로 지지대를 세워 뒤뚱거리는 문제 방지.



### → 피드백을 참고한 반영 사항

- 내부에 폼보드를 추가로 부착하여 흔들리지 않도록 구조 보강.
- 사람이 없으면 일정 시간 동안 계속 감지하다 모터 정지.  
    > 일정 시간 이후 다시 모터 시작.

## 2. 피드백 반영 사항 \_ 갈등 해결 방법(솔루션) \_ 일주일 동안의 개선 사항

### 반영한 기능

#### 1. AI 캐릭터가 설명해주는 갈등 해결 방법 솔루션 영상

- 어떻게 하면 갈등 해결 방법을 시작적이고 덜 피로한 방식으로 보여줄 수 있을지 고민.
- AI 영상 서비스 Vidnoz AI 플랫폼 이용.
- Vidnoz AI에서 개발용 API를 제공하지 않기 때문에 자체적으로 AI 영상 생성을 자동화 하는 로직 개발.
- S3에 있는 사용자 id 해결 방안 텍스트 파일 폴더를 일정 시간 간격으로 탐색  
→ 새로운 파일 감지 시 AI 영상 생성 후 저장.
- 길이가 긴 텍스트로 갈등 해결 방법을 알려주는 것보다 시각적, 청각적으로 보기 쉬운 영상을 사용하여 사용자 경험 향상.

576721

기존 갈등 해결 방법 텍스트  
설명법 →



### SOLUTION

| 슬버가 갈등에 대한 해결 방법을 알려드릴게요.



수정한 AI 캐릭터가  
설명해주는 갈등 해결 방법  
솔루션 비디오 ↓

<https://www.youtube.com/watch?v=OsFLhl05ys&t=1s>

## 2. 피드백 반영 사항 \_ 갈등 해결 방법(솔루션) \_ 일주일 동안의 개선 사항

### 반영한 기능

#### 2. YouTube API를 통한 랜덤화 된 ‘힐링’ 영상/음악 자동 재생

- 유튜브 API를 통해 ‘힐링’, ‘마음 안정’과 관련된 키워드들에 대한 영상을 자동 랜덤 재생.
- 키워드와 관련된 무분별한 영상 서치를 방지하기 위해 인기순 상위 10개 정도의 영상을 랜덤으로 반환.
- 갈등 해결 방법 확인 페이지에 접속하면 영상 자동 재생되어, 사용자의 마음을 평온하게 하는 것을 도움.

### RELAXING

술버와 함께 힐링의 시간을 가져보세요.



## 2. 피드백 반영 사항 \_ 주행 기기 \_ 일주일 동안의 개선 사항

### 반영한 기능

#### 1. 기기 내부와 바퀴 사이 중심 잡기를 위한 품보드 추가 부착

- 품보드를 바닥에서 2~3mm 띄워 부착하여 회전 중 흔들림과 넘어짐 방지.
- 내부에 품보드를 추가로 부착하여 흔들리지 않도록 구조 보강.

#### 2. 일정 시간 객체 미감지 시 탐색 중지 후 모터 재개

- 일정 시간(1분)동안 객체를 찾지 못하면 객체 탐지 로직을 중단하고 모터를 정지. 이후 일정 시간(**15분**)이 지나면 다시 모터 제어를 시작.



```
search_start_time = None # 객체 탐색 시작 시간  
MAX_SEARCH_TIME = 60 # 객체 탐색 최대 시간  
search_terminated = False # 객체 탐색 종료 여부  
rotation_duration = 3 # 회전 시간 (초)
```

```
if not searching:  
    searching = True  
    print("사람 미감지 - 정지 후 회전 시작")  
    set_motor_for_all('stop', 0)  
    time.sleep(2)  
    last_rotation_time = time.time()  
    set_motor_for_all('left', 66)  
  
else:  
    current_time = time.time()  
    elapsed_time = current_time - search_start_time # 탐색 시간 계산  
  
    if elapsed_time > MAX_SEARCH_TIME:  
        print("탐색 시간 초과 - 탐색 종료-대기")  
        print(f"총 검색 시간: ({elapsed_time:.2f}초)")  
        set_motor_for_all('stop', 0) # 모터 정지  
        search_start_time = None # 객체 탐색 시간 초기화  
        search_terminated = True # 객체 탐색 종료 상태 설정  
        searching = False  
        time.sleep(900) # 15분 대기 (900초)  
        print("15분 대기 완료 - 탐색 재개")  
        search_terminated = False # 객체 탐색 종료 상태 초기화  
return
```

## 2. 수정 & 추가 사항 \_ 갈등 해결책 생성 알림 문자 발송 \_ 일주일 동안의 개선 사항

### 추가한 기능

#### 1. 사용자 회원 전화번호 인증 추가

- CoolSMS API를 활용해 인증번호 발송 및 인증 처리
- 인증 완료 시 DB에 인증 상태 업데이트

#### 2. AWS S3 사용자별 새로운 갈등 해결책 업로드 시 알림 문자 발송

- CoolSMS API를 활용해 S3에 있는 사용자(id)별 해결 방안 텍스트 파일 폴더를 일정 시간 간격으로 탐색하여 새로운 파일 감지 시, 전화번호 인증이 완료된 사용자에 한해 알림 문자 발송
- 갈등 해결 방법이 나왔다는 사실을 즉각적으로 알 수 있도록 웹페이지 접속에 대한 사용자 경험 향상

A-Z user_phone	A-Z is_verified
[NULL]	[NULL]
010	Y
010	N
[NULL]	N

→ 기존

→ 인증이 완료된 사용자

→ 인증을 하지 않은 사용자

→ 전화번호 추가 및 인증을 하지 않은 사용자

[Web발신]  
요청하신 SOLVER 인증번호는 700622  
입니다.

(오늘) 오전 1:38  
[Web발신]  
SOLVER 해결책이 업로드 되었습니다.

### 3. 페르소나에 따른 솔버 작동 시나리오 \_ 재정 문제로 다투는 부부



페르소나 1: 박지현  
(아내, 38세, 프리랜서 디자이너)

#### 1. 성격 및 배경

- 책임감이 강하며, 가계부를 철저히 관리하려 노력하지만 예산 초과로 인해 **스트레스**를 받음.
- 프리랜서로 소득이 일정하지 않아 재정적 안정에 대한 걱정이 큼.
- 남편과의 대화에서 합리적인 해결책을 찾으려고 하지만, 감정적으로 격앙되는 경우가 많음.

#### 2. 재정 상황

- 아이의 교육비, 생활비, 저축 문제로 남편과 의견이 충돌.
- 소비 패턴이 절약적이지만 남편의 지출 관리가 부족하다고 느껴 다툼 발생.



페르소나 2: 김성민  
(남편, 40세, 중소기업 팀장)

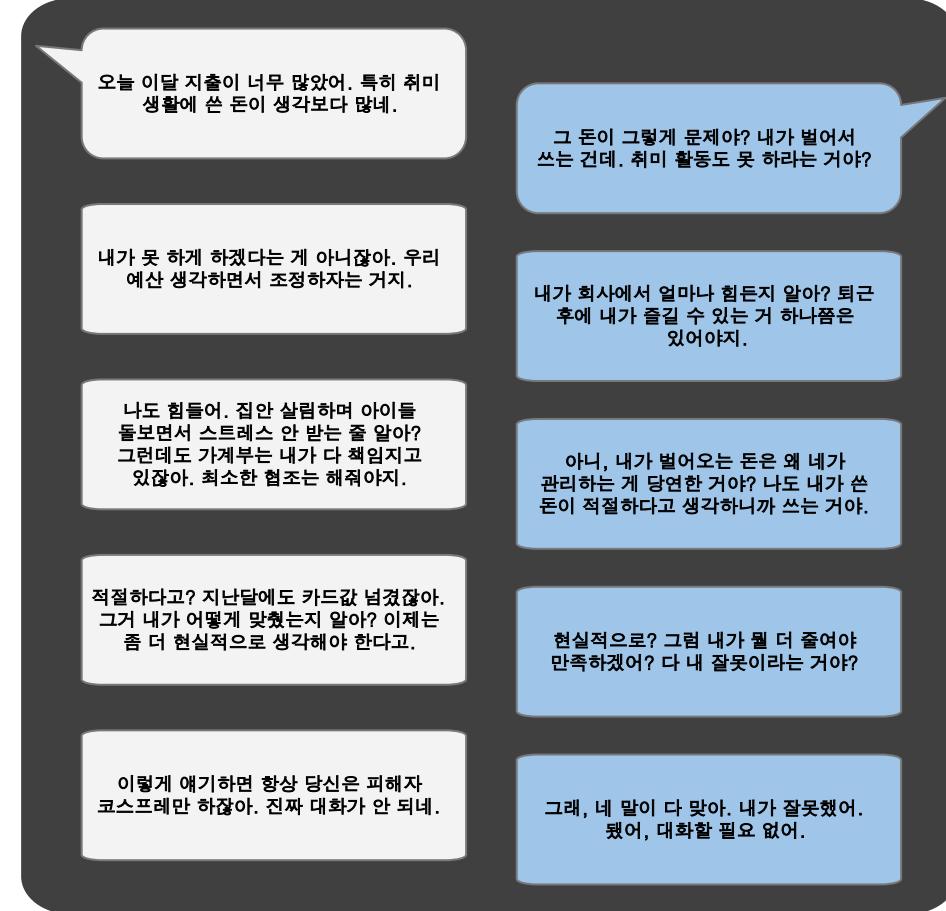
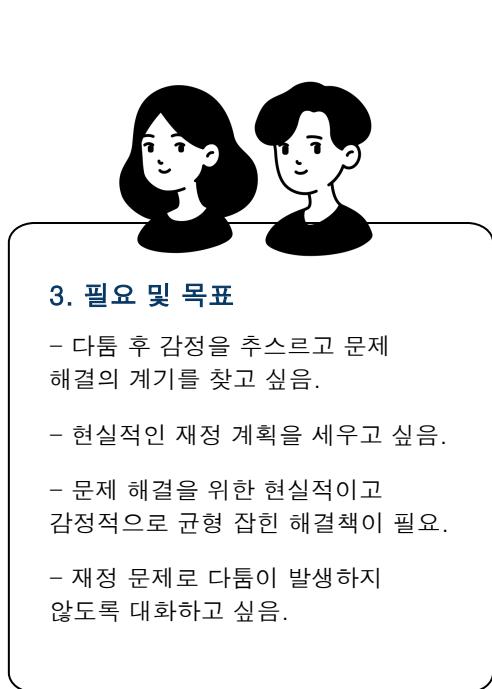
#### 1. 성격 및 배경

- 직장 생활로 인해 피로가 쌓이며, **스트레스를 쇼핑이나 취미 생활로 해소**.
- 가정의 재정 문제에 관심을 갖지만, 아내의 엄격한 지출 관리 방침을 부담 스러워함.
- 대화 중 방어적 태도를 보이거나 감정을 억누르다 폭발하는 경우가 있음.

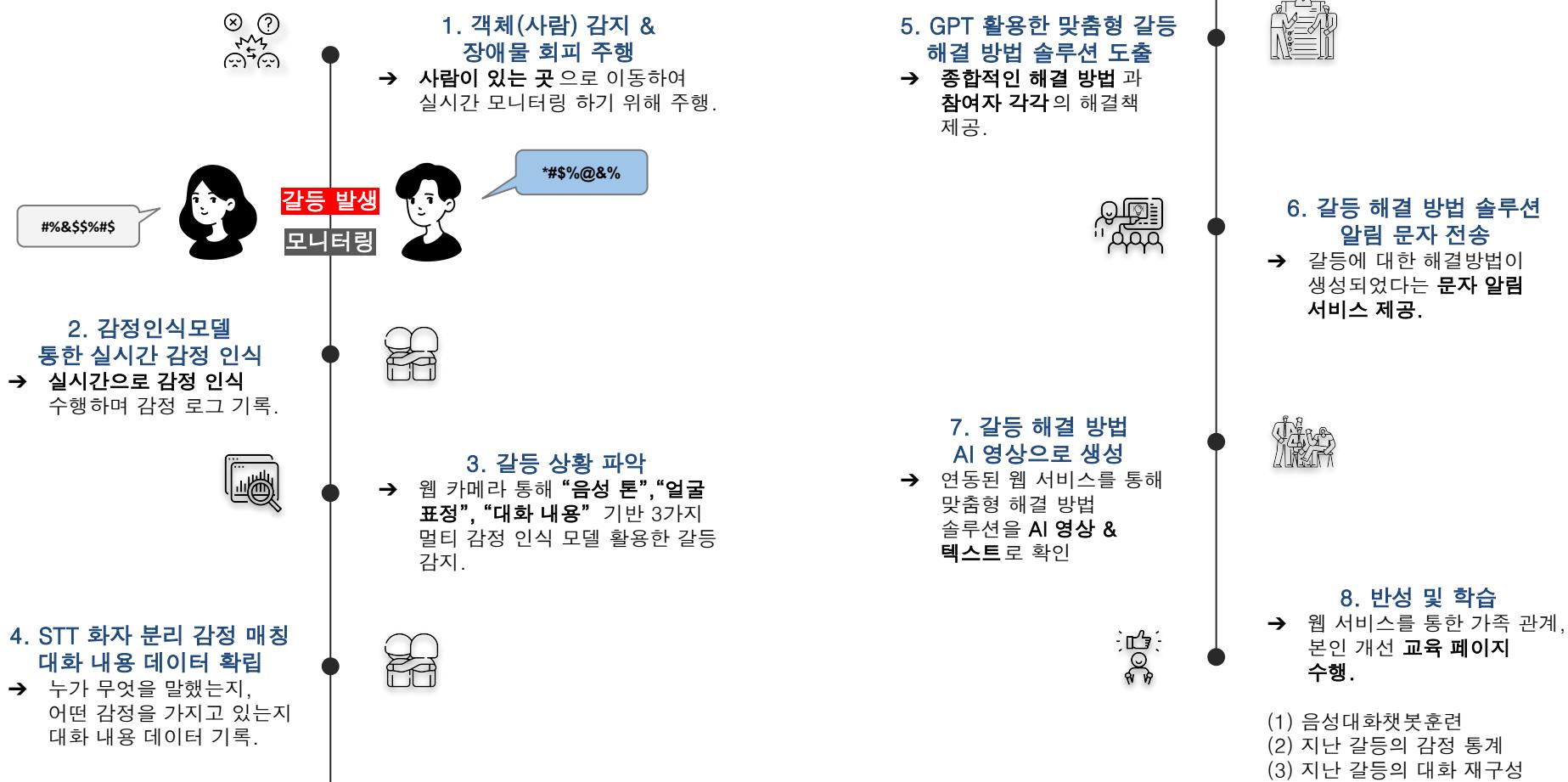
#### 2. 재정 상황

- 월급은 안정적이지만, 개인적인 취미와 가족 외식비 등으로 지출이 많아짐.
- **저축과 소비의 균형 문제**로 아내와 자주 갈등.

### 3. 페르소나에 따른 솔버 작동 시나리오 \_ 재정 문제로 다투는 부부



### 3. 페르소나에 따른 솔버 작동 시나리오 \_ 재정 문제로 다투는 부부



### 3. 페르소나에 따른 솔버 작동 시나리오 \_ 최종적으로 도출되는 해결 방법

**GUIDE**

참여자별로 도움이 될 행동 가이드를 준비했어요.

참여자 1에 대한 솔루션입니다.

우선, 상대방의 스트레스와 감정적 부담을 인정하는 것이 중요해요.

"당신이 회사에서 얼마나 힘든지 알고 있어요."

나도 그 스트레스를 이해하고, 당신이 무엇을 통해 해소하고 있는지 알고 싶어요"라는 말로 상대방을 이해하고 있다는 점을 표현할 수 있어요.

갈등의 원인은 주로 예산의 불확실성에서 비롯된 것 같아요.

참여자 1의 역할은 가계부 관리를 통해 재정 상황을 명확히 하고, 서로의 지출에 대한 투명성을 높이는 것이라고요.

"우리가 함께 예산을 조정하면, 둘 다 원하는 것을 더 명확히 할 수 있을 거예요"라고 대화를 시작해 보세요.

갈등의 과실 비율을 따질 때, 참여자 1은 상대방의 지출에 대한 부담을 너무 적절적으로 표현하면서 상대방을 방어적으로 만든 부분이 약간 있었어요. 약 30% 정도의 책임이 있다고 볼 수 있어요.

"이번 달에는 어느 부분에서 줄일 수 있을지 함께 생각해보면 좋겠어요."

그리고 당신이 꼭 필요한 것은 무엇인지 듣고 싶어요"라는 말로 대화를 이어가세요.

갈등이 해결된 이후에도 정기적으로 서로의 각점은 체크하고, 예산



3. 참여자별 각각의 행동 가이드 (TTS 제공)

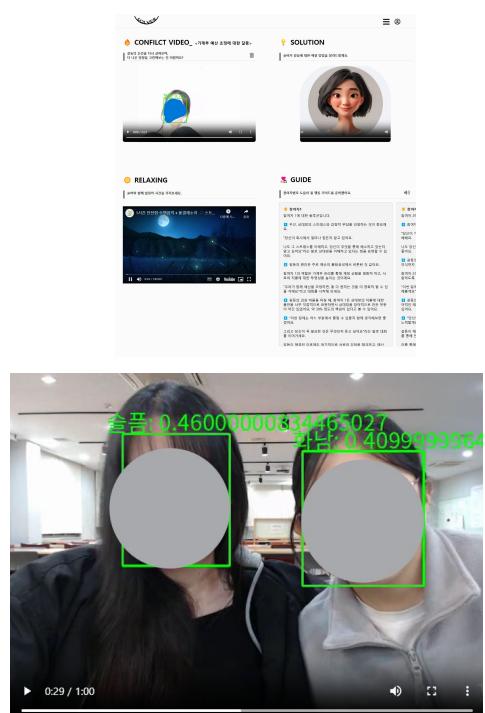
RELAXING

마음이 편안해지는 힐링음악▶▶▶ 속 휴식 음악▶▶▶ 나중에 시▶▶▶ 공유▶▶▶

솔버와 함께 힐링의 시간을 가져보세요.

▶ 0:29 / 1:00

YouTube

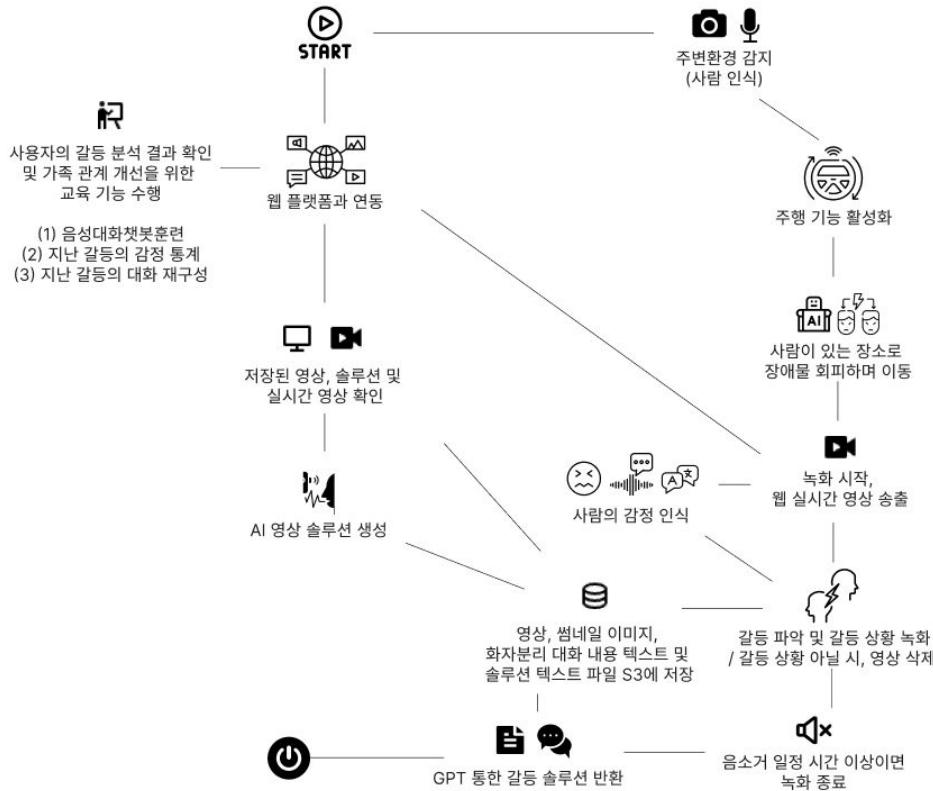


4. 갈등의 순간 다시 살펴보는 용도의 저장된 갈등 영상

# 4. 프로젝트 설계 \_ 요구사항정의서

구분	H/S	서비스(메뉴)	필요기능	기능 설명	
이용자	웹	회원가입	일반 회원가입	사용자는 이메일 인증을 통해 사이트 내 회원가입을 할 수 있다.	
			sns 회원가입	사용자는 sns 인증을 통해 사이트 내 회원가입을 할 수 있다. (카카오, 구글, 네이버)	
		로그인	일반 로그인	사용자는 일반 회원가입을 통해 회원가입한 계정으로 로그인할 수 있다.	
			sns 로그인	사용자는 sns 회원가입을 통해 회원가입한 계정으로 로그인할 수 있다. (카카오, 구글, 네이버)	
		로그아웃	로그아웃	사용자는 로그아웃 할 수 있다.	
		아이디/비밀번호 찾기	아이디/비밀번호 찾기	사용자는 가입한 이메일을 통해 새로운 비밀번호와 기존 아이디를 전송 받을 수 있다.	
		김정 통계	참여한 대화의 김정통계	사용자는 자신이 참여한 대화 중 더 나은 방향으로 재구성을 원하는 대화의 날짜를 클릭하면 해당 대화를 재구성 된 것을 확인할 수 있다.	
		솔루션	참여한 대화의 재구성	사용자는 자신이 참여한 대화에 김정통계를 시각적으로 날짜별로 확인하고 그에 대한 피드백을 확인할 수 있다.	
			저장된 영상 목록	사용자는 사용자별로 저장된 자신이 나온 영상 목록을 확인할 수 있다.	
			저장된 영상 검색	사용자는 사용자별로 저장된 영상을 날짜, 제목 키워드로 검색할 수 있다.	
			저장된 영상 삭제	사용자는 사용자별로 저장된 자신이 나온 영상을 삭제할 수 있다.	
			저장된 영상 상황	사용자는 녹화된 갈등 상황 영상을 확인할 수 있다.	
			저장된 영상, 해결책 확인	사용자는 AI 캐릭터가 설명해주는 갈등 해결 방법 비디오를 확인할 수 있다.	
			음악/영상 자동 재생	사용자는 음악/영상 자동 재생을 통해 마음을 안정시킬 수 있다.	
			갈등 해결 방법 알림 문자	사용자는 갈등에 대한 참여자별 자세한 행동 가이드를 확인할 수 있다.	
		김정 대화 훈련	김정 음성 대화 피드백 챗봇	사용자는 음성으로 챗봇과 대화하며 김정조절대화 훈련을 할 수 있다.	
			회원정보 수정	사용자는 회원정보를 수정할 수 있다.	
			전화번호 인증	사용자는 전화번호를 인증하여 인증 상태를 업데이트할 수 있다.	
			회원 탈퇴	사용자는 회원 탈퇴를 할 수 있다.	
			가족 관계 개선을 위한 팁	사용자는 랜덤으로 표시되는 가족 관계 개선을 위한 팁을 확인할 수 있다.	
		마이페이지	사용자는 삭제된 영상을 휴지통에서 확인할 수 있다.		
			휴지통	사용자는 삭제된 영상을 다시 복구할 수 있다.	
			삭제된 영상 휴지통에서 '비우기'	사용자는 삭제된 영상을 휴지통에서 '비우기' 기능으로 완전히 삭제할 수 있다.	
			삭제된 영상 휴지통에서 '삭제'	사용자는 30일 후 자동 삭제를 통해 영상을 관리할 수 있다.	
			영상 실시간 스트리밍 확인	사용자는 실시간으로 영상을 확인할 수 있다.	
	메인 화면	김정 매칭 화자분리 대화내용 확인	김정 매칭 화자분리 대화내용 확인	사용자는 갈등에 대한 김정이 매칭된 화자분리 대화내용이 실시간으로 반환되어 텍스트로 확인할 수 있다.	
			갈등 해결 피드백 텍스트 정보	사용자는 갈등에 대한 해결 피드백 내용이 실시간으로 반환되어 텍스트로 확인할 수 있다.	
			갈등 상황 실시간 감지	사용자는 갈등 상황을 실시간으로 감지한다.	
		갈등 상황에 대한 해결 방안 제시	갈등 상황에 대한 해결 방안 제시	개발한 김정 분석 모델로 분석된 김정 결과와 Naver Clova Speech API를 통한 화자분리 후 김정 매칭된 대화 내용을 바탕으로 OpenAI API를 통해 갈등 해결을 위한 구체적인 방안을 제시한다.	
		음성인식 및 대화요약	음성인식 및 대화요약	음성인식 기술을 사용하여 대화를 인식하고 분석한다. 대화의 주요 내용을 요약하여 갈등의 핵심 문제를 파악한다.	
	기기		주행 기능	실시간 사람 객체 감지를 통해 주행한다.	
			객체 감지	객체(사람) 감지를 통해 사람이 있는 곳으로 이동한다.	
			장애물 회피	라이다를 이용하여 장애물을 회피하고 사람 방향으로 주행한다.	
	갈등 여부 파악 녹화 시작	갈등 여부 파악 녹화 시작	실시간 영상 김정 분석을 통해 갈등 여부를 파악한 뒤 녹화를 시작한다.		
	웹 플랫폼과 연결	웹 플랫폼과 연결	솔버와 웹 서버를 연동시켜 통신한다. 갈등결과 분석 결과를 확인하고, 저장된 영상과 실시간 모니터링을 할 수 있다.		

# 4. 프로젝트 설계 \_ 시스템 흐름도



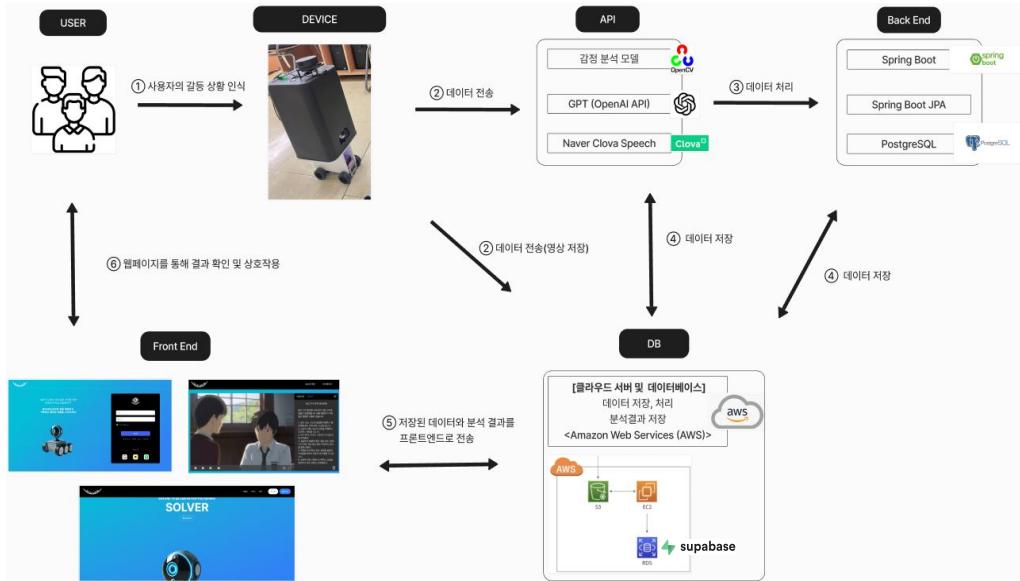
- ① 주변 **객체(사람)** 방향 장애물 회피 **주행**.
- ② 감정인식모델 기반 실시간 **감정 인식**.
- ③ **갈등상황 파악**.  
(표정, 단어, 음성톤 기반 감정 인식 모델 활용)  
→ **갈등 상황 아닐 시**, 영상 삭제.  
→ **갈등 상황 시**, 해결 방법 솔루션 반환.
- ④ 음소거 일정 시간 이상이면 대화가 종료되었다고 판단하여 녹화 종료.
- ⑤ GPT를 통한 **즉각적인 갈등 해결 방법 솔루션** 반환.
- ⑥ Vidnoz AI 이용한 갈등 해결 방법 담은 **AI 영상 생성**.

## 사용자

- ① **연동된 웹**을 통한 홈캠 솔루션, 저장된 영상 확인
- ② **반성 및 학습**을 위한 웹 기능 수행

# 4. 프로젝트 설계 \_ 시스템 구성도

## 자세한 시스템 구성 로직 설명



① 기기는 **모니터링**으로 사용자의 음성 및 영상 데이터를 수집한다. Flask 서버로 웹에 실시간으로 영상을 송출한다. 음성과 영상 데이터는 avi와 wav로 따로 녹화되기 때문에 오디오와 비디오의 길이를 조정하여 mp4로 변환한다.

② **감정 분석 모델**을 통해 실시간으로 변화하는 감정 데이터를 기록한다. 사용자의 음성 데이터는 Naver Clova Speech API를 통해 **STT 화자분리**된 텍스트로 변환된다.

화자분리는 실시간으로 불가능하기 때문에 영상을 먼저 S3에 저장한 후 해당 영상을 서버에 저장하고 화자분리를 수행 후 삭제하는 방식으로 이루어진다.

③ **분석된 감정 데이터**와 **화자분리 결과**를 입 열림 유무에 따라 각 화자와 시간에 맞게 매칭시켜 감정 대화 내용 텍스트 파일을 생성한다.

④ 기록된 영상의 음성 톤, 표정, 텍스트 단어 기반 3가지 감정인식모델을 활용하여 즉각적으로 **갈등 상황 유무를 파악** 한다.

⑤ 결합된 텍스트 파일을 미리 작성된 솔루션 템플릿이 있는 **OpenAI API(GPT)**에게 전송해 분석하고 적절한 해결방안을 반환한다.

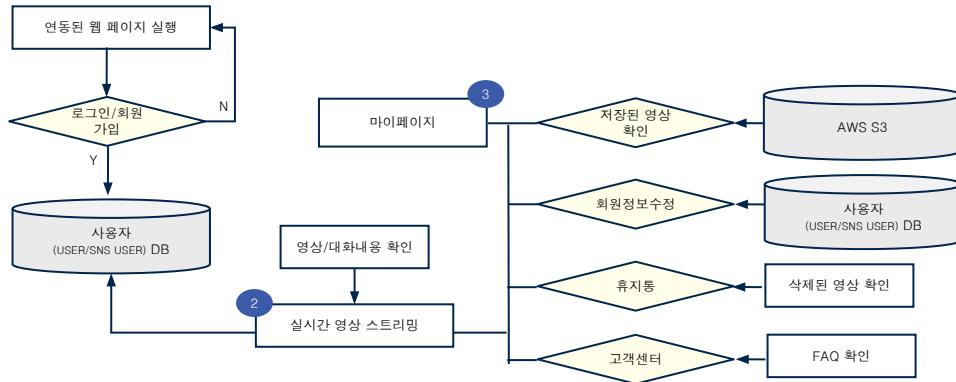
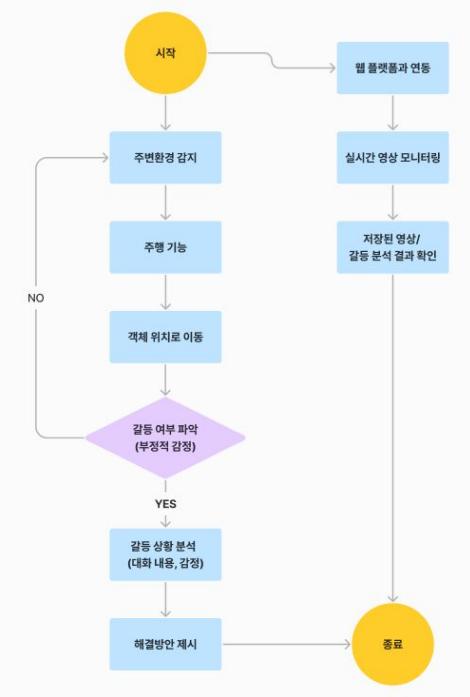
⑥ 생성된 해결 방안과 결합된 텍스트 파일, 영상 데이터를 **각 아이디 별로 자동**으로 생성되는 AWS S3 버킷 폴더에 저장한다.

⑦ Vidnoz AI에서 개발용 API를 제공하지 않기 때문에 자체적으로 **AI 영상 생성을 자동화**하는 로직 개발하여 S3에 있는 사용자 id 해결 방안 텍스트 파일 폴더를 일정 시간 간격으로 탐색하고 **새로운 파일 감지 시 AI 영상 생성** 후 저장한다.

⑧ 저장된 데이터와 분석 결과를 SpringBoot 기반 백엔드를 이용하여 프론트엔드로 전송하여 **데이터를 시각적으로** 표시한다. 저장된 영상과 갈등에 대한 적절한 해결 방안을 보여준다.

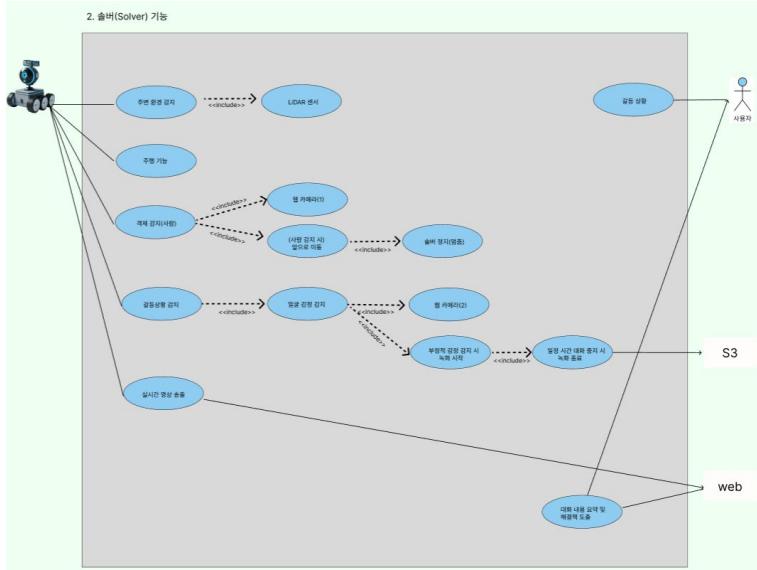
⑨ 사용자는 웹 페이지를 통해 분석 결과와 해결방안, 영상 등을 확인한다. 제공된 해결 방안을 **참고하여 갈등을 해결** 한다.

# 4. 프로젝트 설계 \_ 기능 흐름도



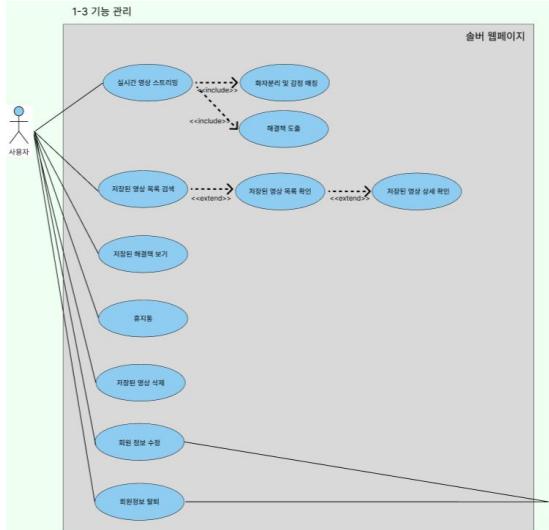
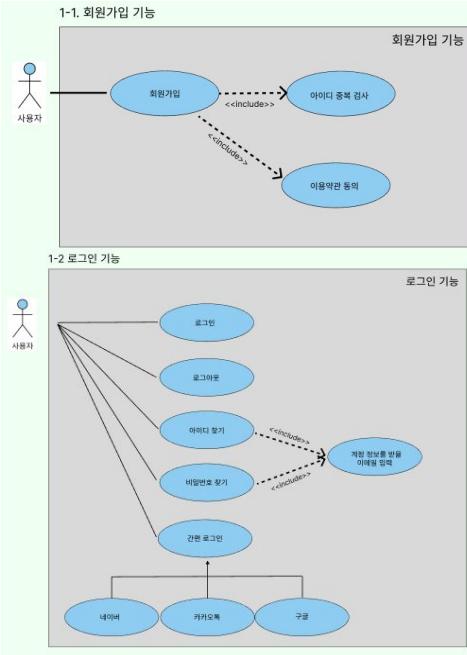
- 1 솔버 웹 페이지를 클릭하여 솔버 기기와 연동된 웹 페이지에 접속한 후 로그인과 회원가입을 수행
- 2 메인페이지에서 실시간 영상 스트리밍 확인
- 3 마이페이지를 선택, 메뉴(저장된 영상 확인, 회원정보수정, 휴지통, 고객센터)를 선택한 후 해당 메뉴 실행

# 4. 프로젝트 설계 \_ 유스케이스 \_ 객체(사람) 감지 및 벽 회피 주행 기기



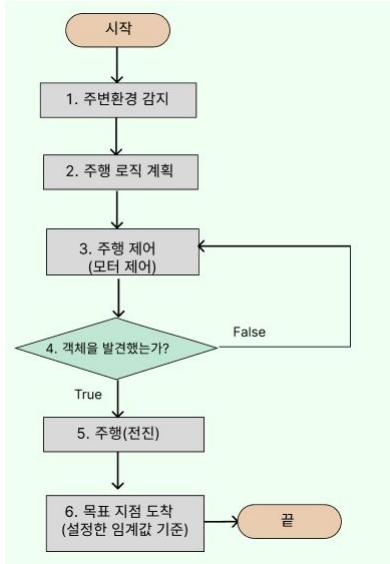
		솔버(solver) 기능에 관한 설명		
항목				
개요	사용자가 솔버를 이용할 수 있다.			
관련 액터	주 액터	사용자, 솔버		
	보조 액터	데이터베이스, 웹 서버		
우선 순위	상	중요도	10	
	하	난이도	솔버 기기를 만들어내고 기능 구현을 할 수 있는 능력이 필요하다.	
선행 조건	솔버 기기를 구매해야한다.			
후행 조건	솔버 기기가 작동될 수 있도록 평지에 두어야하며 wifi가 연결된 상태여야한다.			
시나리오	1. 솔버는 라이다 센서를 통해서 주변 환경을 감지하여 주행할 수 있다. 2. 솔버는 주행 기능을 가지고 있다. 3. 솔버는 객체 감지(사람)기능이 있으며 웹 카메라(1)을 통해 사람 쪽으로 이동한다. 4. 솔버의 갈등 상황 감지는 웹 카메라(2)를 이용하여 얼굴 감정을 감지한다. 5. 사용자의 감정이 부정적이라고 감지하면 녹화를 시작한다. 6. 사용자의 갈등상황이 끝났다고 판단 시 녹화를 종료한다. 7. 녹화 된 영상은 저장 후 S3로 전송한다. 8. 웹은 솔버로부터 실시간 영상을 받고, 대화 내용 요약 및 해결책을 도출한다. 9. 사용자는 실시간 영상의 대화 내용 요약 및 해결책을 웹으로부터 제공받을 수 있다.			
	A1. 사용자의 감정이 부정적으로 판단되지 않는 경우 : 3가지의 모델에서 부정적 감정 여부를 파악하고 부정적인 감정이 과반수 시 갈등 상황이라고 판단하는데 이와 같지 않으면 영상 녹화를 중단한다.			
비기능적 요구사항	성능- 사용자가 갈등상황 중 15초 동안 서로 오고가는 대화가 없으면 솔버는 갈등상황이 종료되었다고 판단하고 녹화를 자동으로 영상 녹화를 종료함.			

# 4. 프로젝트 설계 \_ 유스케이스 \_ 웹 페이지



솔버 웹 사이트에 대한 설명(1-1,1-2,1-3)			
항목	개요	관련 액터	사용자
	주 액터	보조 액터	관리자, 데이터베이스, 웹 서버
우선 순위	상	중요도 낮아도	8
선행 조건	솔버 기기를 구매한 사용자에게 한해 웹 사이트 이용이 가능하다.		
후행 조건	솔버 기기 사용 이력이 있어야 하며 wifi 등 인터넷이 작동되는 환경이여야 한다.		
시나리오	기본 시나리오	1. 사용자는 아이디 중복, 이용약관 필수 등의 항목을 통해서 회원가입을 할 수 있다. 2. 회원을 로그인 할 수 있으며 아이디, 비밀번호 찾기가 가능하다. 간편로그인 3가지 방법(네이버, 카카오, 구글)을 통해서도 가능하다. 3. 사용자는 메인화면에서 웹 카메라(2)를 통해 녹화되는 실시간 영상을 확인할 수 있으며 감정과 화자 분리된 대화 내용의 매칭, 해결 방안을 제공 받을 수 있다. 4. 사용자는 마이페이지에서 이전에 저장된 영상을 볼 수 있으며 영상 삭제도 가능하다. 5. 관리자는 회원 정보 및 탈퇴를 관리한다.	
		A1. 영상 삭제 시 : 삭제한 영상은 휴지통으로 이동하고 휴지통에서는 30일 후 완전히 삭제되지만 영상을 바로 삭제하고자 한다면 휴지통으로 이동해 한번 더 지우면 완전히 삭제된다.	
비기능적 요구사항			
		성능 - 저장된 영상 목록 검색 시, 영상 제목과 날짜를 기반으로 키워드 검색이 가능하다.	

# 4. 프로젝트 설계 \_ 알고리즘 명세서 \_ 주행 기기

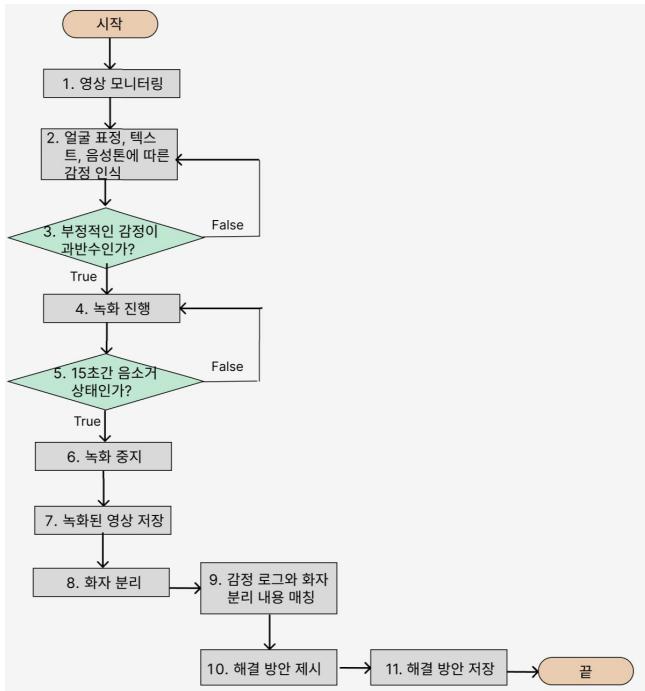


## ● 알고리즘 명 세서

1. LiDAR를 활용하여 주변환경 감지 즉, 장애물이 있는지 여부를 파악한다.
2. 장애물 감지 여부와 객체(사람) 감지 여부에 따른 주행 로직을 결정한다.
3. PWM(Pulse Width Modulation) 신호를 사용하여 모터를 제어한다.
4. YOLO 알고리즘을 통해 객체(사람)를 감지한다.  
\* YOLO 알고리즘 : 이미지를 한 번만 보고 바로 물체를 검출할 수 딥러닝 객체 감지 알고리즘
- 4-1. 객체를 감지하지 못하면 일정 시간 동안 탐지 로직 실행한 후, 객체를 찾지 못하면 모터를 정지, 그 후 15분 대기하고 다시 모터 제어를 한다.
5. 카메라를 통해 객체(사람)를 감지하고 사람이 있는 방향으로 이동한다.
6. 객체(사람) 바운딩 박스 면적이 일정 임계값 이상을 차지하면 모터를 정지 한다.  
(사람이 일정 거리만큼 떨어져 있는 것이 확인되면)

# 4. 프로젝트 설계 \_ 알고리즘 명세서 \_ 감정/갈등 인식 및 해결 방법

## ● 알고리즘 명세서



1. 실시간으로 영상을 모니터링 한다.
2. 얼굴 표정, 텍스트, 멜 스펙트로그램을 기반으로 감정을 종합적으로 분석해 감정을 인식하고 이를 로그로 기록한다.
3. 15초간 음소거 상태인지 판단한다.
4. 15초간 음소거 상태일시 녹화를 중지한다.
5. 녹화된 영상을 해당 사용자 id별로 S3에 저장한다.
6. 얼굴 표정, 텍스트, 멜 스펙트로그램 기반 감정 모델을 통해 부정적인 감정이 과반수인지 판단 후 '갈등 인식' 판단한다.
7. 각각의 모델을 통한 '갈등 인식'이 과반수이면 '갈등 상황' 판단 후 솔루션 도출 로직을 진행한다. 갈등 상황이 아닐 시, 영상을 삭제한다.
8. 녹화된 영상에서 Naver Clova Speech API를 활용해 화자 분리를 수행한다.
9. 기록된 감정 로그와 화자 분리 내용을 매칭한다.
10. 감정 로그와 매칭된 화자 분리 내용을 토대로 GPT OpenAI API를 활용해 갈등에 대한 해결 방안을 제시한다.
11. 제시한 해결 방안을 저장한다.

# 5. 개발환경

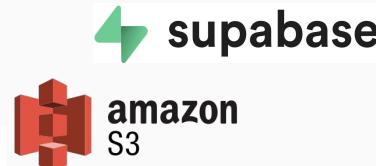
Front-end



API



Back-end



개발 & 협업 도구

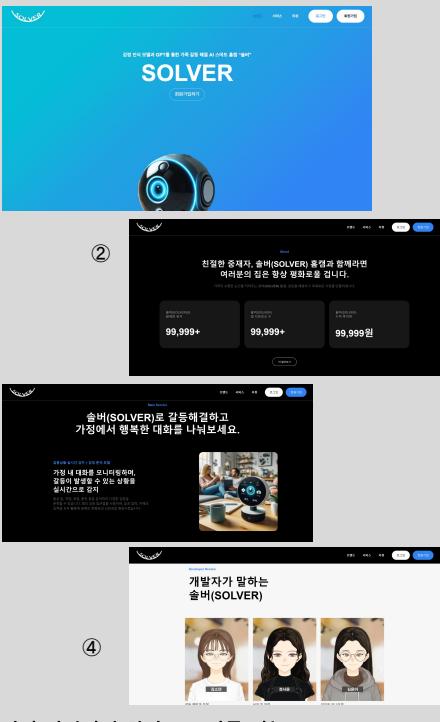


# 5. 주요 적용 기술

감정 분석 및 인식 모델의 제작	<ul style="list-style-type: none"><li>- AI 허브와 Kaggle를 통해 감정 분석을 위한 데이터(음성, 텍스트, 이미지)를 수집하여 Google Colab에서 모델 생성 학습.</li><li>- <b>텍스트 기반 감정 분석</b> : HuggingFace와 KoBERT 모델을 활용하여 텍스트 데이터를 긍정과 부정으로 분류하는 자연어 처리(NLP) 기법을 사용하여 텍스트로 된 대화 내용의 감정을 분석.</li><li>- <b>음성 기반 감정 분석</b> : 멜 스펙트로그램을 사용하여 음성 데이터를 시각화한 후, 음성의 감정을 분석하는 모델.</li><li>- <b>이미지 기반 얼굴 표정 인식</b> : OpenCV와 이미지 데이터를 사용하여 얼굴 표정으로부터 감정을 인식.</li></ul>
음성 인식 및 화자 분리	<ul style="list-style-type: none"><li>- <b>STT (Speech-to-Text)</b> : 음성 데이터를 텍스트로 변환하는 음성 인식 기술을 활용하여 대화 내용을 텍스트로 전환.</li><li>- <b>화자 분리 (Speaker Diarization)</b> : 대화에서 각 발화자를 식별하여 <b>누가 어떤 발화를 했는지</b> 를 구분.</li></ul>
객체 감지 및 장애물 회피	<ul style="list-style-type: none"><li>- <b>컴퓨터 비전 모델</b> : 객체 감지를 위한 YOLO 등과 같은 딥러닝 기반 모델을 사용하여 사람 및 장애물을 인식.</li><li>- <b>센서 기반 제어</b> : Lidar를 사용하여 장애물 감지 및 회피 기능을 구현.</li></ul>
AI 기반 솔루션 도출	<ul style="list-style-type: none"><li>- <b>GPT 모델</b> : OpenAI의 GPT API를 사용하여 갈등 분석 및 해결책 도출을 위한 적절한 텍스트 생성을 수행.</li><li>- <b>맞춤형 템플릿 사용</b> : 사전 정의된 템플릿을 사용 해 맞춤형 해결책을 <b>자동으로 생성하는 로직</b> 을 설계 .</li><li>- <b>AI 캐릭터 솔루션 영상</b>: Vldnoz AI라는 AI 영상 플랫폼에 로그인하여 <b>AI 영상을 자동으로 생성하는 로직</b> 을 구성.</li></ul>
웹 플랫폼 개발	<ul style="list-style-type: none"><li>- <b>백엔드 기술</b> : Spring Boot, Flask 등 백엔드 프레임워크를 사용하여 서버 로직과 데이터 처리 기능을 구현.</li><li>- <b>프론트엔드 기술</b> : html, css, js를 사용하여 사용자 인터페이스를 개발.</li></ul>
데이터 시각화 및 피드백 제공	<ul style="list-style-type: none"><li>- <b>차트 라이브러리</b> : Chart.js 등을 사용하여 감정 분석 결과와 통계 데이터를 <b>시각적으로 표현</b>.</li><li>- <b>TTS (Text-to-Speech)</b>: 해결책을 음성으로 제공하기 위해 TTS 기술을 사용하여 사용자 경험을 강화.</li></ul>
클라우드 및 데이터 저장소	<ul style="list-style-type: none"><li>- <b>AWS S3</b> : 녹화된 영상 및 분석된 데이터를 저장하고 관리하기 위한 클라우드 스토리지. <b>각 id 별로 폴더가 생성</b> .</li><li>- <b>클라우드 데이터베이스</b> : 사용자 정보와 감정 분석 결과를 저장하기 위해 Supabase 등의 데이터베이스 사용.</li></ul>

# 5. S/W 기능 & 기술 – 웹 페이지 \_ 회원 기능

웹 소개 화면 (처음 접속 화면)



①

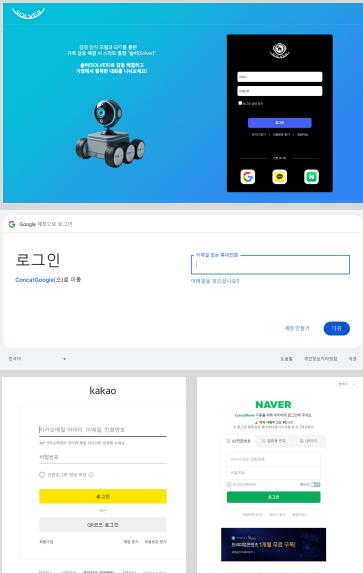
③

④

- 로그인과 회원가입 화면으로 이동 가능

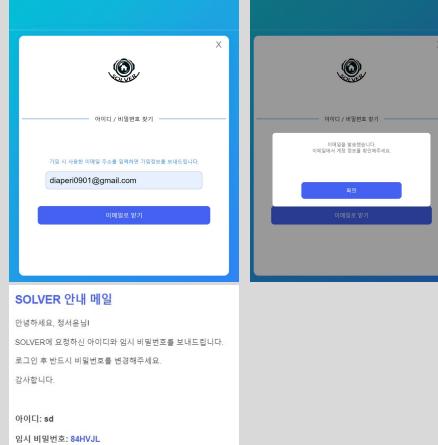
- “솔버” 브랜드의 소개, 서비스, 개발자 리뷰로 구성

로그인 화면 / 간편 로그인 화면



- [간편 로그인]
  - Google, Kakao, Naver Developers에서 API를 쓸 수 있도록 Application을 만든 후 발급받은 Client Id와 Client Secret Key 를 이용해 로그인 API를 가져와서 Google, Kakao, Naver 로그인이 가능하도록 설정

아이디/비밀번호 찾기 화면

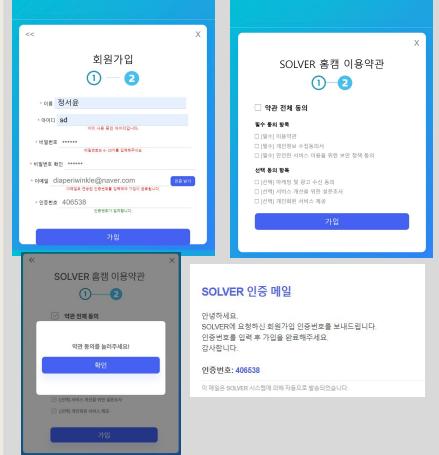


- 사용자의 이메일로 임시 비밀번호를 생성하여 네이버 SMTP 서버를 통해 전송하고, 비밀번호를 업데이트

- bcrypt를 사용하여 비밀번호를 안전하게 암호화

5

회원가입 화면



[회원가입]

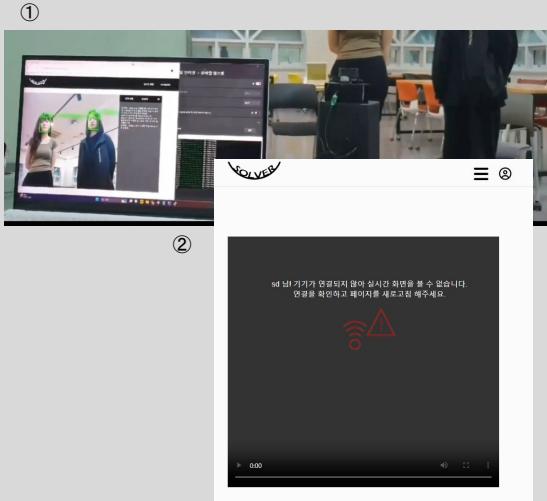
- 사용자가 이름, 아이디(아이디 중복검사), 비밀번호, 비밀번호 확인, 이메일 입력 후 인증 번호를 받는 방식. 이후 이용약관에 동의하면 회원가입 완료.

- 사용자의 이메일로 인증 번호를 네이버 SMTP 서버를 통해 전송

- bcrypt를 사용하여 비밀번호를 안전하게 암호화

# 5. S/W 기능 & 기술 – 웹 페이지 \_ 메인 / 솔루션 확인

메인 화면 – 실시간 화면 모니터링  
(위: 기기 연결 시 / 아래: 기기 연결 X)

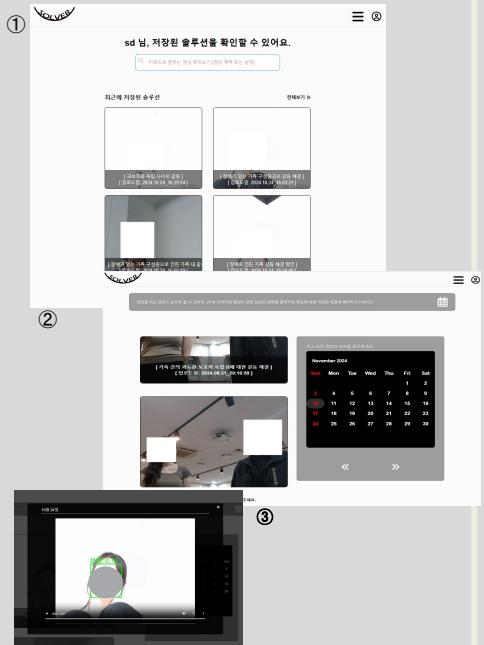


- 라즈베리파이에서 flask 서버를 이용하여 로그인된 사용자 id 값과 라즈베리파이 웹캠으로 녹화되는 영상을 웹에서 실시간 송출.

- 라즈베리파이에서 사용자 id 값에 따라 S3에 자동적으로 풀더 생성.

- (1) 영상,
- (2) 영상에 대한 셈네일 이미지,
- (3) 감정로그 텍스트 파일
- (4) 솔루션 영상,
- (5) 솔루션 텍스트 파일
- (6) 화자분리 감정매칭 대화내용 텍스트 파일 저장.

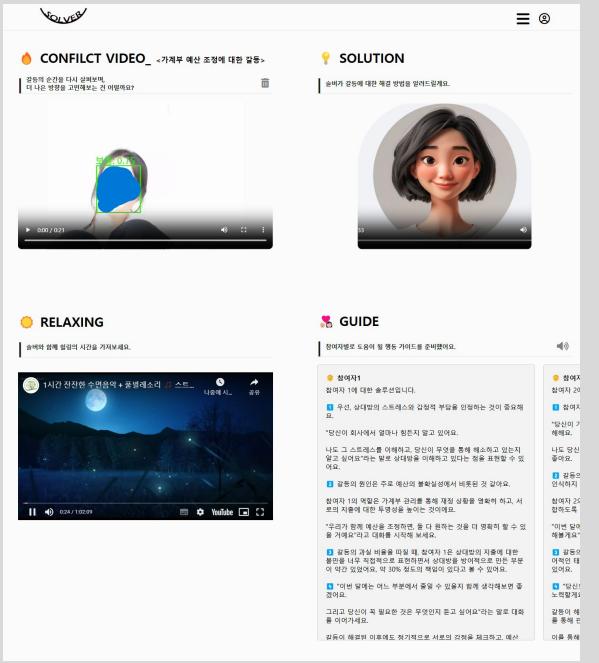
저장된 솔루션 및 영상 확인 목록 화면



- S3에 id 별로 저장된 영상과 셈네일 이미지를 최신 순으로 확인할 수 있음.

- 옆 칼린더를 이용하여 원하는 날짜의 솔루션 영상과 결과를 확인할 수 있음.

솔루션 페이지 화면



- (1) 녹화된 해당 과거 갈등 영상,
- (2) AI 솔루션 영상,
- (3) 마음 안정을 위한 음악/영상 자동 재생,
- (4) 갈등에 대한 참여자별 자세한 행동 가이드 확인 가능.

## 5. S/W 기능 & 기술 \_ 웹 페이지 \_반성/학습을 위한 교육 페이지

# 5. S/W 기능 & 기술 – 웹 페이지 \_ 마이페이지

**휴지통 화면**

– AWS S3에서 각 사용자별 30일 이상된 파일을 탐색하여 오래된 영상을 자동 삭제

– 휴지통 영상 분류: 영상의 날짜에 따라 주 단위로 파일을 구분하여 관리

– 휴지통 비우기: 사용자가 영상을 일괄적으로 영구 삭제

– 영상 복구: 삭제된 영상을 “저장된 영상”페이지로 복원하는 기능

**회원 정보 수정 화면**

– [일반 로그인 사용자]는 이름, 이메일, 전화번호, 비밀번호 수정이 가능, 아이디는 고유한 값으로 수정 불가

– [SNS 로그인 사용자]는 정보 수정 대신 이름, 이메일, 로그인한 계정을 확인할 수 있도록 함

– 로그아웃과 탈퇴 가능

– CoolSMS API를 사용해 전화번호 인증을 위한 인증 번호 발송 및 인증 상태 업데이트

– 인증이 완료된 사용자에 대해 AWS S3 사용자(id)별 텍스트 파일 폴더를 일정 시간 간격으로 탐색하여 새로운 갈등 해결책 업로드 시 알림 문자 발송

**고객센터 화면**

– 고객센터의 자주 묻는 질문(FAQ)를 제공하여 궁금한 내용을 검색하거나 질문별로 답변을 확인할 수 있도록 구성

– 검색 기능과 페이지네이션을 통해 정보를 쉽게 탐색 가능

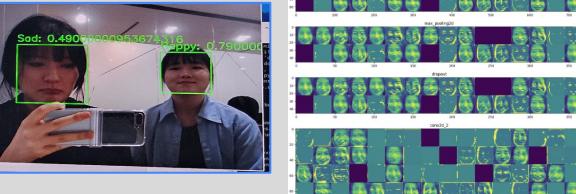
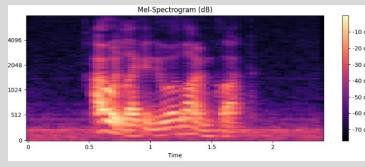
회원정보 수정

휴지통

고객센터

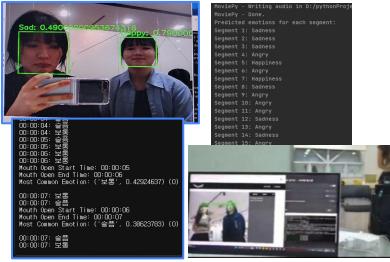
로그아웃

# 5. S/W 기술 – 멀티 감정 인식 모델의 제작 colab

얼굴 표정 이미지 기반 모델	텍스트 단어 기반 모델	음성 툰 멜 스펙트로그램 기반 모델
<p><b>1. 데이터 준비</b> – “kaggle”의 “FER2013 데이터셋” 이용. 각 데이터는 이미지 픽셀 값(48x48 해상도, 흑백 이미지)과 감정 라벨(angry, disgust, fear, happy, sad, surprise, neutral)을 포함.</p> <p><b>2. 데이터 분할 및 전처리</b> – 학습/테스트 데이터 분할: 데이터 재구성: 픽셀 데이터를 48x48 배열로 변환하여 CNN 모델에 입력 가능하도록 4차원 데이터로 조정.</p> <p><b>3. 얼굴 감지 및 특징 추출</b> – 얼굴 검출: OpenCV의 CascadeClassifier를 사용하여 이미지에서 얼굴 영역 검출. – 특징 추출: 검출된 얼굴을 정해진 크기(48x48)로 리사이즈하여 학습 데이터로 활용</p> <p><b>4. 모델 설계</b> – CNN(Convolutional Neural Network) 구조: 3개의 Convolution 레이어와 MaxPooling 레이어를 사용. 마지막 Dense 레이어에 각각의 노드를 배치하여 감정 클래스 예측.</p> <p><b>5. 모델 학습</b> – 학습 데이터에 데이터 증강을 적용하여 모델 학습.</p> 	<p><b>1. 데이터 준비</b> – “AI 허브”的 “감성대화말뭉치”, “한국어 감정 정보가 포함된 단발성 및 연속적 대화 데이터셋”과 “NSMC”的 “네이버 영화 긍부정 데이터셋”을 이용해 훈련데이터 178,000개, 테스트데이터 50,000개로 학습 진행 – 부정 레이블 89,173개 / 긍정 레이블 88,827개</p> <p><b>2. 라이브러리</b> – TensorFlow와 HuggingFace의 transformers 라이브러리 사용 – NumPy와 Pandas로 데이터 처리 – TQDM으로 학습 진행 상황 표시</p> <p><b>3. 데이터 처리</b> – SEQ_LEN = 64로 텍스트 최대 길이를 설정하여 입력 데이터 처리 – 토큰화: 텍스트를 숫자 토큰으로 변환하고 패딩 처리 – 마스크: 패딩된 부분은 0, 실제 텍스트는 1로 표시하여 모델이 패딩을 무시하도록 설정</p> <p><b>4. 데이터 변환</b> – 텍스트 데이터를 토큰화하고, 마스크와 세그먼트 생성 – 출력: 토큰, 마스크, 세그먼트와 레이블을 반환</p> <p><b>5. 모델 정의</b> – KoBERT 모델 (monologg/kobert)을 HuggingFace에서 로드 – 출력: 감정 분석을 위한 sigmoid 활성화 함수가 적용된 1개의 Dense 레이어</p> <p><b>6. 모델 학습</b> – Adam 옵티마이저와 BinaryCrossentropy 손실 함수 사용 – 과적합 방지를 위해 50% 드롭아웃 적용</p>	<p><b>1. 데이터 준비, 데이터 전처리</b> – “AI 허브”的 “감정 분류를 위한 대화 음성 데이터셋” 이용. – CSV 파일과 음성 데이터셋 준비: CSV 파일에 각 음성 파일의 ID와 감정 레이블이 포함되어 있고, 음성 파일은 .wav 형식으로 제공됨. – 감정 클래스 정의: 감정 레이블 정의 (Angry, Disgust, Fear, Happiness, Neutral, Sadness, Surprise).</p> <p><b>2. 멜 스펙트로그램 생성</b> – librosa 라이브러리를 사용하여 오디오 데이터를 멜 스펙트로그램으로 변환</p> <p><b>3. 데이터셋 구축</b> 오디오 파일을 로드하고, 멜 스펙트로그램 생성. 데이터 분할: 학습, 검증, 테스트 데이터로 분리 (60:20:20 비율). ollate_fn 함수로 배치 내 모든 멜 스펙트로그램의 크기를 동일하게 맞춤.</p> <p><b>4. 모델 설계</b> – CNN(Convolutional Neural Network) 구조: 34개의 합성곱 레이어 (각각 64, 128, 256, 512 필터 사용). 감정 클래스 개수(7개)를 출력.</p> <p><b>5. 모델 학습</b> – 가중치 적용된 CrossEntropyLoss: 클래스 불균형을 해결하기 위해 클래스별 가중치를 적용.</p> <p><b>6. 검증 및 테스트</b></p>

# 5. S/W 기술 – 감정/갈등 인식 및 해결 방법 도출 관련

3가지 감정 인식 모델 기반  
감정/갈등 상황 인식  
→ 감정 인식 모델의 개발



- 텍스트, 얼굴 표정 인식, 멜 스펙트로그램 기반 음성 감정 분석 등 제작한 3가지 모델을 동시에 활용해 갈등을 종합적으로 분석.

- 실시간으로 변화하는 감정 로그 기록.

- 각 모델마다 부정적 감정이 과반수이면 '갈등발생' 판단.

- 갈등 상황이 아닌 것으로 판단되면 영상이 삭제되고, 갈등 상황인 것으로 판단되면 최종적으로 솔루션을 도출.

갈등 대화/상황  
감정 매칭 화자 분리 텍스트  
기반 데이터 확립

참여자1: (보통) 오늘 이달 지출이 너무 많았어. 특히 하미 생활에 쓴 돈이 생각보다 많네.  
참여자2: (화남) 그 뜻이 그렇게지? 내가 봄에서 시스 드래. 키미 할증도 꽤 하라는 거야?  
참여자3: (슬픔) 내가 못 하게 하겠다는 게 아니잖아. 우리 예전 살피면서 조절하자는 거지.  
참여자2: (보통) 내가 회사에서 얼마나 할증지 알아? 되면 후에 내가 출출 수 있는 거 하나쯤은 있어야지.  
참여자3: (화남) 난도 날도 둘째어. 짚어 살피면서 아예 물보면서 스트레스 애 받는 줄 알았지? 그런데이도 가게는데 내가 다 책임져야 있잖아. 유통은 혼조는 혼조는 혼조지.  
참여자2: (화남) 아니, 내가 빚어놓은 돈은 옛 얘기 관리하는 게 달라진 거지? 난도 내가 금 윤이 적잖 하다고 생각하니까 죽는 거야.  
참여자3: (화남) 적혀있다고? 지난날에도 카드간 널었잖아. 그게 내가 어떻게 맞겠는지 알아? 이제는 좀 더 현실적으로 생각해야 한다고.  
참여자2: (슬픔) 현실적으로? 그럼 내가 잘 더 좋아야 만족하잖아? 더 내 걸질이라는 거야?  
참여자3: (보통) 이렇게 얘기하면 항상 당신은 미워하지 코스피만 하잖아. 진짜 대화가 안 되네.  
참여자2: (화남) 그래. 네 말이 다 맞아. 내가 잘못했어. 악어. 대화할 필요 없어.

- 갈등 영상만 보고 **사람이 아닌 로봇이** 갈등 파악부터 솔루션 도출까지 자동적으로 할 수 없음. **이미지와 텍스트의 적절한 입력이 필요**.

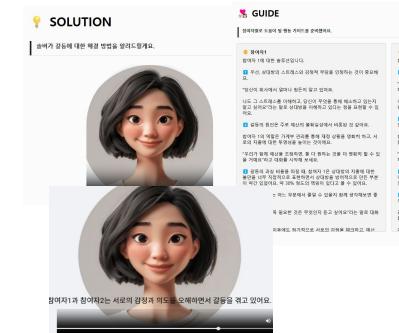
- 갈등 영상을 통해 **stt 화자 분리**를 적용하여 참여자들의 음성이 참여자 각각의 텍스트로 변환.

- **화자 분리된 대화 내용이 어떤 비언어적 감정을 가지고 있는지**, 감정 분석한 결과의 로그를 적용하여 각각의 대화 내용에 매칭.

- 누가 무엇을 말했는지, 어떤 감정을 가지고 있는지 **비언어적 신호**를 텍스트와 함께 데이터로 확립하고 입력되어 각각의 참여자별 솔루션을 도출 가능.

- 또한, 비 언어적 감정 표현과 대화 내용이 일치하지 않는 비꼬는 표현 등을 파악하는데 도움.

AI 영상 생성 및  
갈등 참여자별 객관적인 맞춤형  
갈등 해결 방안 제시(TTS, 영상)



- 감정 매칭 된 화자분리 대화 데이터 활용

- GPT OpenAI API의 적절한 해결책 요구 템플릿을 바탕으로 갈등 분석을 수행.

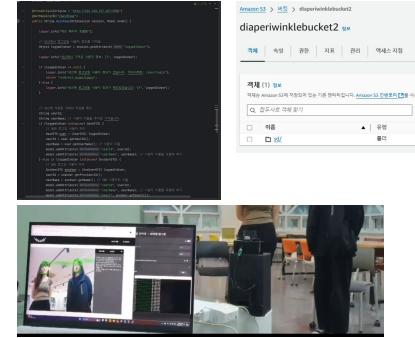
- Vidnoz AI에서 개발용 API를 제공하지 않기 때문에 자체적으로 AI 영상 생성을 자동화 하는 로직 개발.

(1) AI 캐릭터가 설명하는  
갈등에 대한 종합적인 갈등 해결 방법

(2) 참여자별 갈등에 대한 실질적, 구체적 행동 계획 제안

(2-1) 감정 분석에 따른 **참여자별 공감**,  
(2-2) 갈등의 원인 분석,  
(2-3) 참여자별 **파실(잘못)** 비율 판단과 설명,  
(2-4) 갈등에 대한 참여자별 **세부적인 솔루션**,

웹페이지 □ 라즈베리파이  
연결을 통한 개인화 된  
실시간 화면 송출  
(홈캠의 역할/flask 서버)



- 로그인된 **사용자 id** 값을 가져와  
라즈베리파이 웹캠으로 녹화되는 영상 웹에서  
송출.  
(flask 서버)

- 사용자 id 값에 따라 S3에 자동적으로 풀더  
생성. 영상 및 솔루션 저장.

# 5. H/W 기술 \_ 사용된 부품

기능	감정/갈등 인식, 분석 및 해결책 반환을 위한 장치	객체 감지 장애물 회피 주행
부품 사진		
부품 항목	로지텍 HD 웹캠 C920, 라즈베리파이5	라이다 센서 RPLIDAR A1M8-R6, 라즈베리파이4, 라즈베리파이 RC카 쉴드, DC 모터, 보조배터리, 웹카메라
설명	<ul style="list-style-type: none"><li>➤ 저장된 모델을 통한 감정/갈등 인식/분석, 영상 송출, 영상 처리, 화자 분리 stt, gpt api를 통한 해결책 도출</li><li>→ <b>갈등 해결 스마트 흡캠의 역할</b></li></ul>	<ul style="list-style-type: none"><li>➤ 장애물을 회피하며 객체(사람)이 있는 곳으로 주행 (라이다 센서와 웹카메라를 이용하여 <b>장애물, 객체 인식</b>)</li></ul>

# 5. H/W 기능 & 기술 – 주행 기기

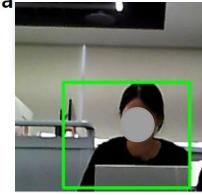
## 객체 감지(사람) 주행

- YOLO 모델을 사용하여 카메라에서 실시간으로 객체(사람)를 감지.  
→ cv2.VideoCapture(0)을 사용하여 카메라에서 실시간으로 프레임을 가져옴.
- 객체 감지 시 객체의 바운딩 박스를 확인하고, 이를 기반으로 PWM(Pulse Width Modulation) 신호를 사용하여 모터를 제어해 객체가 있는 방향으로 주행.
- 사람과 일정 거리 가까워지면 주행 기기 정지.  
바운딩 박스 면적의 임계값(CLOSE\_DISTANCE\_THRESHOLD)을 설정하여 면적이 임계값을 초과할 경우 모터를 정지하고 모터제어를 종료. 만약 사람이 일정 거리 이상 떨어져 있다면 기기는 사람과의 거리를 유지하기 위해 전진.
- 객체 미감지 시 회전 및 탐색을 통해 객체를 찾음.  
- 1분간 회전하며 객체 탐색 후 미감지 시 모터 정지, 15분 대기 후 다시 모터 제어.

```
# YOLO 처리 및 모터 제어  
camera = cv2.VideoCapture(0)  
camera.set(cv2.CAP_PROP_FRAME_WIDTH, 320)  
camera.set(cv2.CAP_PROP_FRAME_HEIGHT, 240)
```

```
# YOLO 설정  
yolo_cfg = "/home/pi/pythonFiles/data/yolov3.cfg"  
yolo_weights = "/home/pi/pythonFiles/data/yolov3.weights"  
yolo_names = "/home/pi/pythonFiles/data/coco.names"
```

Object Detection Camera



```
search_start_time = None # 객체 탐색 시작 시간  
MAX_SEARCH_TIME = 60 # 객체 탐색 최대 시간  
search_terminated = False # 객체 탐색 종료 여부  
rotation_duration = 3 # 회전 시간 (초)
```

```
if elapsed_time > MAX_SEARCH_TIME:  
    print("탐색 시간 초과 - 탐색 종료-대기")  
    print(f"총 검색 시간: {elapsed_time:.2f}초")  
    set_motor_for_all('stop', 0) # 모터 정지  
    search_start_time = None # 객체 탐색 시간 초기화  
    search_terminated = True # 객체 탐색 종료 상태 설정  
    searching = False  
    time.sleep(900) # 15분 대기 (900초)  
    print("15분 대기 완료 - 탐색 재개")  
    search_terminated = False # 객체 탐색 종료 상태 초기화  
return
```

```
CLOSE_DISTANCE_THRESHOLD = 38400
```

```
if motor_control_started:  
    if detected_area >= CLOSE_DISTANCE_THRESHOLD:  
        stop_motors()
```

```
if target_box:  
    if max_area >= CLOSE_DISTANCE_THRESHOLD:  
        set_motor_for_all('stop', 0)  
        motor_control_started = False # 모터 제어 완전 종료
```

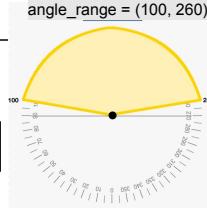
# 5. H/W 기능 & 기술 – 주행 기기

## 장애물 회피 주행

- 기기 상단의 LiDAR 센서를 통해 장애물을 감지.
- LiDAR 감지 각도 범위를 기기의 앞쪽 100도~260도사이로 설정하여 해당 영역을 감지하도록 조정.
- check\_obstacle() 함수: lidar\_data에서 거리(distance)와 각도(angle)를 포함한 장애물 정보를 확인.
- 장애물 감지 거리: 250mm 이하에 있을 경우 장애물로 인식.  
장애물 회피 동작 실행: 회전 후 장애물 감지여부(True, False) 반환 후 True일 경우 과정 반복.
- LiDAR센서 시각화 화면: 작은 흰색원으로 라이다 센서의 위치를 표현, 즉 기기의 위치.



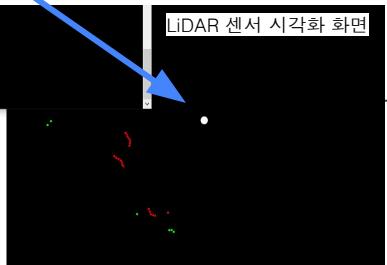
```
lidar_data = []
angle_range = (100, 260)
```



```
# 장애물 감지 함수 정의
def check_obstacle():
    global lidar_data
    obstacle_detected = False # 장애물 감지 여부 초기화
    for data in lidar_data:
        distance = data['distance']
        if distance < 250: # 장애물 거리 기준(250mm)
            obstacle_detected = True # 장애물 감지
            print("장애물 감지됨:", distance, "mm")
            break # 장애물 감지 시 루프 종료
    if not obstacle_detected:
        print("장애물 미감지") # 장애물 미감지 시 출력
    return obstacle_detected # 장애물 감지 상태 반환
```

장애물 감지됨 : 247.75 mm  
모터 제어 상태 : 장애물 감지 여부 = True  
장애물 감지 - 원쪽으로 회전  
장애물 감지됨 : 248.5 mm  
모터 제어 상태 : 장애물 감지  
장애물 감지 - 원쪽으로 회전  
장애물 감지됨 : 247.5 mm  
모터 제어 상태 : 장애물 감지 여부 = True  
장애물 미감지

Object Detection Camera



fetch()로 0.1초마다 /data에서 LiDAR 데이터를 가져와 업데이트.

LiDAR 센서로 측정된 거리가 1000mm이하인 경우 감지  
->1000mm이하: 초록색, 더 가까운 장애물: 빨간색

## 5. H/W 기능 & 기술 – H/W 주행 시연 영상



<https://youtube.com/shorts/qxnCn2g-weo?feature=share>

## 5. DB & AWS S3

## - Supabase - 회원 관리

Table Editor		solver Free		solver		Enable branching	
schema: public		Filter		Sort		Insert	
+ New table						Auth policy role postgres	
Search tables... <input type="text"/>							
<b>sns_user</b>							
<input checked="" type="checkbox"/> id <input checked="" type="checkbox"/> email <input checked="" type="checkbox"/> name <input checked="" type="checkbox"/> provider <input checked="" type="checkbox"/> provider_id							
1	1	@gmail.com	조혜은	google	118294216562414		
2	2	@gmail.com	조혜은	kakao	366858		
3	3	@naver.com	조혜은	naver	hEgFG-EmAachgW5DTzMdmgcU_ru-M43Afj1DYQ_I29s		
+ New table							
Search tables... <input type="text"/>							
<b>sns_user</b>							
<input checked="" type="checkbox"/> use... <input checked="" type="checkbox"/> i... <input checked="" type="checkbox"/> is_secession bpchar						enroll_date timestamp	
2	N					2024-10-27 18:09:33.639000	
18	N					2024-11-17 22:12:51.987000	
<b>user_table</b>							
<input checked="" type="checkbox"/> is_secession <input checked="" type="checkbox"/> user_no <input checked="" type="checkbox"/> enroll_date <input checked="" type="checkbox"/> update_date <input checked="" type="checkbox"/> a2_user_id_ <input checked="" type="checkbox"/> a2_user_name_ <input checked="" type="checkbox"/> a2_user_email_ <input checked="" type="checkbox"/> a2_user_pw_ <input checked="" type="checkbox"/> a2_user_phone_ <input checked="" type="checkbox"/> a2_is_verified_							
1	N	2	2024-10-27 18:09:33.639	2024-10-27 18:09:33.639	sd	inaverc_r.\$2a\$10\$am7m79Yw	[NULL]
2	N	18	2024-11-17 22:12:51.987	2024-11-19 15:53:31.881	222	@gmail.c.\$2a\$10\$Ocd/a3Z8	010

## - AWS S3 -

### 사진 & 영상 & 텍스트 파일 사용자(id)별 관리

Amazon S3 > 버킷

▶ 개정 소스넷 - 24시간마다 업데이트 [모든 AWS 리전]

[Storage Lens 대시보드 보기](#)

Storage Lens은 스토리지 사용량 및 활용 추세에 대한 가시성을 제공합니다. [자세히 알아보기]

---

법용 버킷 | 디렉터리 버킷

법용 버킷 (2) 정보 [모든 AWS 리전]

[\[삭제\]](#) ARN 복사 | 버이 있음 | 삭제 | 버킷 만들기

버킷은 S3에 저장되는 데이터의 체계이니입니다.

검색창: 이름으로 버킷 찾기

< | > | ⌂

---

이름	AWS 리전	IAM Access Analyzer	생성 날짜
diaperiwinklebucket1	아시아 태평양(서울)	ap-southeast-2	2024. 5. 29. pm 대한 분석기 보기
diaperiwinklebucket2	아시아 태평양(서울)	ap-northeast-2	2024. 8. 21. pm 대한 분석기 보기

Amazon S3 > 버킷 > diaperiwinklebucket2

diaperiwinklebucket2 정보

작성 | 속성 | 권한 | 지표 | 관리 | 액세스 지침

---

작성 (3) 정보

[\[삭제\]](#) S3 URL 복사 | URL 복사 | 다운로드 | 업로드

작성 Amazon S3에 저장되어 있는 기본 엔터티입니다. Amazon S3 인증トリ리를 사용하여 액세스를 원하는 경우 하위 명시적으로 권한을 부여해야 합니다. [자세히 알아보기]

검색창: 접두사로 작성 찾기

---

이름	유형	마지막 수정
55/	폴더	-
done/	폴더	-
sd/	폴더	-

감사합니다