1. 프로젝트 개요
   1. 프로젝트 명
   2. 프로젝트 목적
      1. 운전 안정성, 유지보수 편리, 자동화
   3. 프로젝트 차별성
      1. End2end로 프로젝트 구성, 기존에는 일부분 구현된 프로젝트는 있지만 전체 프로세스 구현은 없음
   4. 팀 구성
   5. 프로젝트 진행 개요 및 기간
2. 프로젝트 목표 달성도
   1. 당초 목표 및 달성 여부
      1. 주요 성과 (실시간 추론 성공, 정확도 개선)
         1. 엣지 디바이스에서 모델 양자화로 리얼타임, 위험도 모델 분석해서 모델링
      2. 위험도 산정 모델 적용으로 인한 관리 효율 및 서비스 품질 향상
      3. 현재 시스템상 노면 장애물 발생시 민원 접수 이후 처리되며 위험도 산정의 기준이 모호함. 이를 위험도 분석 모델을 통해 자동화하여 처리.
3. 프로젝트 개발 결과
   1. 시스템 구성도 및 최종 아키텍처
   2. 개발환경 및 프레임워크
      1. SBC: Orin Nano
      2. AWS
      3. ROS
      4. Ubuntu
      5. Android studio
      6. Free Cad
      7. Flutter, Django, Node.js, Mysql
      8. YOLO
      9. Github, Notion
   3. 주요 기능 시연 결과 스크린샷/영상
      1. 교내 도로 시연 영상
      2. 실제 공도 주행 시연 영상
   4. 데이터 흐름 및 연계 확인 (Edge → Server → Web & App)
   5. 개발 내용
      1. ROMED 기기- SBC
      2. 도비 - Web
      3. 도비 - App
      4. Back
   6. 프로그램 디렉터리 구조
      1. 전체 디렉터리
      2. SBC, Back, Web, App
4. 성능 및 품질 검증 결과
   1. AI 모델 (Detection & Segmentation) 성능결과 - mAP, recall, precision
   2. 위험도 모델 분석
5. 한계점 및 개선 방안
   1. 확인된 제약사항 또는 미해결 과제
      1. 야간/악천후 시 성능 악화
   2. 기술적 개선 제안
      1. AI 모델 고도화
      2. 웹, 앱 기능 고도화
   3. 확장 로드맵
      1. 서비스 배포
      2. 지자체 API 연동
      3. 실시간 교통정보 연동
      4. ROMED 인지 시스템 소형화
6. 결론
   1. 개발 일정
   2. 프로젝트 후기
7. 부록
   1. 데이터셋 자료
   2. 추가 통계 자료 등등
   3. 참고문헌

프로젝트 완료 보고서

팀 명 :TakeOut

202235220 김민서, 202235257 박주연, 202035126 김성욱, 202235000 이영호

**목차**

1. 프로젝트 개요
   1. 프로젝트 명
   2. 프로젝트 목적
   3. 프로젝트 차별성
   4. 팀 구성 및 업부 분장
2. 프로젝트 목표 달성도
   1. 당초 목표 및 달성 여부
3. 프로젝트 개발 결과
   1. 시스템 구성도 및 최종 아키텍처
   2. 개발환경 및 프레임워크
   3. 주요 기능 시연 결과 스크린샷/영상
   4. 데이터 흐름 및 연계 확인
   5. 개발 내용
   6. 프로그램 디렉터리 구조
      1. 전체 디렉터리
      2. SBC, Back, Web, App
4. 성능 및 품질 검증 결과
   1. AI 모델 성능 결과
   2. 위험도 모델 분석
5. 한계점 및 개선 방안
   1. 확인된 제약사항 또는 미해결 과제
   2. 기술적 개선 제안
   3. 확장 로드맵
6. 결론
   1. 개발 일정
   2. 프로젝트 후기
7. 부록
   1. 데이터셋 자료
   2. 추가 통계 자료 등등
   3. 참고문헌
8. 프로젝트 개요
9. 프로젝트 명
10. 도로 노면 장애 대응을 위한 AI 안전 모니터링 기반 실시간 정보 안내 통합 서비스 시스템
11. 프로젝트 목표
12. 본 프로젝트는 도로 노면 상 발생하는 다양한 장애(포트홀, 균열 등)를 실시간으로 감지하고 이를 운전자에게 직관적으로 알림으로써 주행 안전성을 극대화하고자 한다. 추가적으로 위험도 기반 유지보수 우선순위 설정을 통해 도로 관리자의 유지 및 보수 작업을 효율화·자동화하여 관리 편의를 높이고 궁극적으로 도로 전반의 안전 수준 및 서비스 품질을 향상시키는 것을 목표로 한다.
13. 프로젝트 차별성
14. 앱, 웹 서비스를 제공함으로써 서비스 이용자인 노면 관리자와 운전자에게 통합된 서비스 시스템 제공한다.
15. 도로 관리 등 각 분야의 효율성을 높이고 운전자의 안전 향상을 기대할 수 있다.
16. 웹 서비스에서 노면 불량 신고를 간편하게 관리할 수 있어 처리 효율을 높일 수 있다.
17. 앱, 웹, DB 각 파트 간 연계를 통해 유기적으로 동작하는 프로젝트를 구성하였다.
18. End-to-End
19. 팀 구성 및 업무 분장
    * 1. 팀명: TakeOut
      2. 팀장: 김민서 팀원: 김성욱, 박주연, 이영호
      3. 담당 업무

|  |  |  |  |
| --- | --- | --- | --- |
| 업무 | 담당자 | 세부 업무 | 설명 |
| AI 기반  노면 불량 인식 | 김민서,이영호 | 데이터셋 조사 | 노면 불량 인식을 위한 오픈소스 데이터셋 조사 |
| 박주연, 김성욱 | 객체 인식 모델 조사 | YOLO, Fast-RCNN, Mask-RCNN, Vision Transformer 등 모델 비교 분석 |
| 박주연 | YOLO v11 기반 Object Detection 학습 | YOLO Detection 모델 기반 객체(포트홀 등) 실시간 탐지 |
| 김성욱 | YOLO v11 기반 Segmentation 학습&최적화 | YOLO segmentation 기반 노면 장애(균열, 박리, 요철 등) 실시간 탐지 |
| 노면 장애 위험도 평가 모델 | 김민서, 이영호 | 노면 평가 지표 조사 | 서울시, 도로교통공사 등에서 사용하는 노면 평가 지표 조사 |
| 위험도 평가 모델 모델링 | 자료 조사 결과를 토대로 위험도 모델 모델링 GPS, AI 인식 결과, Tmap API 를 통합하여 위험도 계산 |
| 백앤드 서버 연동 | 백엔드 API 및 Tmap API를 이용하여 엣지 디바이스에서 서버로 업로드 |
| 백엔드 | 이영호 | 서버 구축 | AWS와 Django를 활용하여 서버 구축 |
| 웹 서비스 API | 노면 장애 관리 서비스에 필요한 API 제작 |
| 앱 서비스 API | 네비게이션 서비스에 필요한 API 제작 |
| 유저 관리 | 유저 정보를 통해 서비스 접속 가능한 토큰 발행하고 API별 권한 분류 |
| AWS 환경 구축 | EC2를 사용하여 서버 구축 |
| 웹서비스 | 김민서 | 노면 장애 리스트 확인 및 관리 | 노면 장애 리스트를 DB에서 받아와 리스트화하여 사용자에게 시각화 |
| 노면 장애 알림 발송 | 노면 장애 리스트에서 관리자 판단으로 중요도가 높은 경우 알림 발송 |
| 노면 장애 제보 처리 및 자주 묻는 질문 | 노면 장애 제보 처리 및 자주 질문할 내용들을 예측하여 웹페이지로 구현 |
| 네비게이션 앱 | 김성욱, 박주연 | 지도 API 연동 | Naver Map SDK 를 활용한 지도 생성 |
| 사용자 제공 네비게이션 | 사용자가 설정한 목적지로 경로 안내 |
| 노면 장애 현황 데이터 수신 | DB와 실시간으로 통신하여 진행 경로 상 노면 장애 데이터 수신 |
| 노면 장애 알림 발송 | 경로 주변 노면 장애를 정제하여 사용자에게 최적화된 정보를 네비게이션 상에서 알림 팝업 형태로 안내 |
| Edge Device 환경 구축 | 이영호 | AI 안전 모니터링을 위한 환경 구축 | OS, GPU 드라이버, CUDA, CuDNN 설정 및 사용 프레임워크 세팅 |
| Real-time 환경 구축 | 사전 학습된 모델을 TensorRT를 통해 변환하여 실시간 추론 파이프라인 구성 |
| 3D 모델링 | 박주연 | Edge Device 케이스 | Edge Device 케이스 모델링 |

* 1. 프로젝트 배경
     1. 최근 5년간 2만 건 이상의 포트홀이 발생하였으며, 고속도로 특성상 포트홀은 대형 교통사고의 주요 원인이 될 수 있다. 또한 차선 마모, 장애물 방치 등 관리되지 않은 노면 상태는 사고 위험을 더욱 증가시킨다. 이러한 상황에서 즉각적인 조치가 필수적이다.
     2. 노면 장애를 실시간으로 확인하고 신속히 대응하기에는 담당 인력이 턱없이 부족한 실정이다. 게다가 복잡한 민원 절차로 인해 미신고되는 불량 사례가 늘어나고 있어, 기존 관리 방식의 한계를 극복할 필요성이 대두된다해결 방안차량에 장착된 카메라와 AI 시스템을 활용한 노면 불량 자동 인식
  2. 기대 효과
     1. 노면 장애 파악에 필요한 인력 및 비용 절감
     2. 신속한 대응으로 노면 불량 관련 사고 빈도 및 피해 감소
     3. 후행 차량에 대한 사전 경고 제공을 통해 안전성 강화
  3. 프로젝트 기능 요약
     1. 노면 불량 감지 : Edge Device를 차량에 장착, 카메라로 노면 불량을 AI로 실시간으로 감지하여 서버로 전송한다. 모델은 YOLO v11 Detection Model, Segmentation Model을 사용하였다.
     2. 서버 관리 : 감지된 정보는 서버로 전송되어 도로 관리용 웹 대시보드 및 내비게이션 앱과 연동
     3. 네비게이션 앱: 운전자는 앱을 통해 실시간으로 노면 장애 정보 알림을 받음
     4. 위험도 산정: 서버에서 인식된 정보와 위치 정보 등을 활용, 수식을 적용하여 위험도를 산정한다.
     5. 인식된 노면 장애 정보를 모니터링 시스템에 반영하여 실시간 관리
     6. 네비게이션 앱을 통한 운전자 대상 노면 장애 정보 실시간 안내
  4. 알림 서비스: 운전자에게 네비게이션 어플을 통하여 노면 불량의 위치와 정보를 안내한다.
     1. 네비게이션 어플 제작
     2. 지도 api
     3. 길찾기 api
     4. 서버와의 통신
     5. 사용자 알림

* 1. 프로젝트 요약

1. 본 프로젝트는 AI 기반 도로 노면 장애 감지 및 위험도 산정 모델을 통해 관리 효율성,안전성을 높였다.
2. 웹 서비스를 통해 도로 노면 장애에 대한 관리 효율성을 높였고 앱 서비스를 통해 운전자에게 도로 위험에 대한 정보를 실시간으로 구현하였다.
3. 실차 검증으로 실생활 적용 가능성을 확인하고 향후 다양한 기능 확장 및 성능 개선 가능성이 남아 있다

1. 프로젝트 진행 개요 및 기간

사진 넣고 프로젝트 진행 기간 표 넣고 각 기간에 대한 설명 넣고 어찌구~~

2.프로젝트 목표 달성도

* 1. 목표 및 달성 여부
  2. AI 모델을 통해 노면 장애 및 위험도 판단 : 달성
  3. 웹페이지를 기반으로 데이터 베이스의 노면 장애 목록 관리 및 다양한 기술 구현: 달성
  4. 네비게이션 앱 기반 노면 장애의 종류 및 위험도와 위치 안내: 달성

1. 주요 성과

1. AI 모델을 통해 노면 장애 및 위험도 판단

1)   AI 모델에 노면 장애 위험도 판단 로직을 적용하였다.

2)   실제 차량을 이용해 노면 장애 데이터를 전송하는 기능을 적용 및 성공하였다.

2. 웹페이지를 기반으로 데이터 베이스의 노면 장애 목록 관리 및 다양한 기술 구현

1)   DB와 웹페이지 간 통신으로 노면 장애 리스트를 업데이트하는 기능을 적용하는 것에 성공하였다.

2)   회원 가입 및 로그인, 알림 데이터 전송, 디자인 수정 및 완성을 완료하였다.

3.네비게이션 앱 기반 노면 장애의 종류 및 위험도와 위치 안내

1)   네이버 맵 호출 및 일정 간격마다 DB에서 노면 장애 데이터 현황을 호 출하도록 하였다.

2)   운전자에게 팝업으로 노면 장애 정보를 안내하는 기능 구현에 성공하였다.

4. 실제 차량에 카메라 설치 및 기능 테스트

1)   실제 차량에 카메라를 설치하여 AI 모델의 정상 작동 여부를 확인하였다.

2)   AI 모델과 앱, DB 간 통신을 통해 운전자가 포트홀이나 노면 장애를 팝업 알림을 통해 미리 인지하고 적절히 대처할 수 있음을 확인하였다.

1. 효과
   1. 관리 효율성 증대: 도로 관리자와 노면 관리자는 노면 장애 위치와 이미지를 실시간으로 확인하고, 처리 여부(반영, 반려, 삭제)를 신속하게 결정할 수 있다. 이를 통해 기존에 복잡했던 절차를 간소화하고, 관리 효율을 크게 향상시킬 수 있다.
   2. 안정성 강화: 관리자가 장애 정보를 파악하고 즉각적인 조치를 취함으로써 잠재적인 사고를 예방할 수 있다. 위험도가 높은 장애물일수록 우선적으로 처리되며, 전반적인 도로 안정성을 개선시킨다.
   3. 운전자 편의성 향상: 운전자는 실시간 안내를 통해 경로 상 노면 장애와 상세 정보를 사전에 인지를 할 수 있다. 이를 통해 장애물 회피나 적절한 속도 조절을 통해 적절한 대응이 가능해져, 노면 장애에 대한 사고 위험을 효율적으로 감소시킨다.

1. 프로젝트 개발 결과
   1. 시스템 구성도 및 최종 아키텍처

텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 폰트, 로고이(가) 표시된 사진

자동 생성된 설명

1. 개발 환경 및 실행 환경
   1. 개발 환경

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 기기 | 운영체제 | 프레임워크 | 사용 언어 |
| SBC | Jetson Orin Nano | Ubuntu 20.04 | - | - |
| Sensor | Ublox zed-f9p GPS Module, Logitech C920 Camera | - | ROS Message Protocol | - |
| AI Model | Jetson Orin Nano | Ubuntu 20.04 | YOLOv11 Seg,  YOLOv5 Detection | Python |
| Back | Host PC | Ubuntu 20.04 | MySQL, AWS, Django | Python |
| Web | Guest PC | - | Node.js, React | JaverScript |
| App | Guest PC | Android | Flutter | Dart |

1. 주요 기능 시연 결과 스크린샷/영상
2. 데이터 흐름 및 연계 여부
   1. 데이터 흐름 다이어그램 넣고 웹페이지, 네비게이션 앱 각 기능 당 데이터 어떻게 연계되었고 어찌고저찌고~

1. 개발 내용
   1. 노면 장애 인식 모델
   2. 프론트엔드
      1. 사이트맵

텍스트, 스크린샷, 직사각형, 폰트이(가) 표시된 사진

자동 생성된 설명

2. Node.js 구성

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 폰트, 텍스트, 스크린샷, 그래픽이(가) 표시된 사진  자동 생성된 설명 |  | 폰트, 텍스트, 스크린샷, 번호이(가) 표시된 사진  자동 생성된 설명 | 폰트, 텍스트, 스크린샷, 그래픽이(가) 표시된 사진  자동 생성된 설명 | 폰트, 텍스트, 스크린샷, 라인이(가) 표시된 사진  자동 생성된 설명 |
| 폰트, 텍스트, 그래픽, 스크린샷이(가) 표시된 사진  자동 생성된 설명 | 폰트, 텍스트, 스크린샷, 번호이(가) 표시된 사진  자동 생성된 설명 | 폰트, 텍스트, 스크린샷, 그래픽이(가) 표시된 사진  자동 생성된 설명 | 폰트, 스크린샷, 그래픽, 텍스트이(가) 표시된 사진  자동 생성된 설명 |

3. 웹페이지 화면

HomePage : 전체 지도 및 노면 지도 위치 표시

아이콘 클릭 시 노면 장애 상세 정보 출력

상세 정보의 아이콘 클릭 시 실제 노면 장애 사진 시각화

목록 hover 시 소목록 시각화

대목록 클릭 시 : 노면 장애별 관리. ManagePage로 연결. 노면 장애 목록 표로 시각화, 관리 기능 버튼 상호작용 가능

검색 버튼 클릭 시 : 팝업 시각화. 검색 기준, 정렬 기준 및 검색 기준에 따른 필터링 조건 적용 가능

시간 검색 기준일 경우 : 달력이 시각화되어 날짜를 선택 가능

지역 검색 기준일 경우 : 시/도를 선택하면 해당하는 시/도의 구를 선택 가능

다음과 같이 시/도를 선택할 수 있다.

시/도에서 서울특별시를 선택하고 구를 선택하지 않았을 경우 : 서울특별시에 해당하는 데이터가 필터링됨

노면 장애 행을 클릭할 경우 메인 페이지의 상호 작용과 같이 노면 장애 상세 정보 팝업이 나오며, 아이콘을 클릭할 경우 메인 화면에서의 상호작용과 동일한 사진 확인 가능

노면 장애별 관리 hover 시 나타나는 소목록 중 노면 균열을 선택했을 경우 해당되는 카테고리의 데이터만 필터링

표 오른쪽 상단의 편집 버튼을 클릭할 경우 노면 데이터 목록 관리가 가능한 상태로 삭제 버튼이 나타남

1, 6번째 행의 삭제 버튼을 클릭한 결과 데이터베이스에서 해당 노면 장애 삭제됨

대목록에서 알림 전송을 클릭할 경우 알림 발송 페이지 AletsPage로 이동. ManagePage와 동일하게 각 행을 클릭하면 노면 장애 상세정보 확인 가능, 검색 필터링과 장애 종류별 필터링 가능. 알림 발송 표시를 띄우는 기능 구현

발송 버튼 클릭 시 행 번호와 알림이 전송되었다는 알림이 상단에 나타남

대목록에서 민원 접수 내역을 클릭할 경우 민원 관리 페이지 ComplaintsPage로 이동. ManagePage와 동일하게 각 행을 클릭하면 노면 장애 상세 정보 확인 가능, 검색 필터링과 장애 종류별 필터링 가능. 민원 접수 내역에서 조치를 할 수 있는 기능 구현

반영 버튼을 눌렀을 경우 처리 여부가 처리로 변화한 것을 확인 가능

반려 버튼을 눌렀을 경우 처리 여부가 반려로 변화한 것을 확인 가능. 노면 장애별 관리 페이지 표에서 해당 행이 사라짐

대목록에서 Q&A를 클릭할 경우 예상되는 질문 목록 페이지 FAQPage로 연결됨. 각 행을 클릭하면 답변을 확인할 수 있음.

작은 삼각형을 누르면 답변 확인이 가능

ㅁㄴㅇㄹ

ㅁㄴㅇㄹ

ㅁㄴㅇㄹ

ㅁㄴㅇㄹ

ㅁㄴㅇㄹ

ㅁㄴㅇㄹ

상세 기능

|  |  |  |
| --- | --- | --- |
| 도비 Web | HomePage | 전체 노면 장애 위치를 지도에 시각화하여 전반적인 파악을 도우며, 아이콘을 클릭할 경우 해당 노면 장애의 상세 정보를 확인할 수 있다. |
| ManagePage | - 노면 장애 리스트를 확인할 수 있다.  - 각 리스트를 클릭하면 MapPopup으로 이어져 지도 상에서 노면 상의 위치와 상세 정보를 아이콘을 통해 알 수 있다.  - 관리자 권한으로 편집 버튼을 통해 노면 장애 데이터를 데이터베이스 상에서 삭제할 수 있다. |
| AlertsPage | - 노면 장애 리스트를 확인할 수 있다.  - 각 리스트를 클릭하면 MapPopup으로 이어져 지도 상에서 노면 상의 위치와 상세 정보를 아이콘을 통해 알 수 있다.  - 관리자 권한으로 알림 버튼을 통해 노면 장애 데이터의 정보를 알림으로 보낼 수 있다. |
| ComplaintsPage | - 노면 장애 리스트를 확인할 수 있다.  - 각 리스트를 클릭하면 MapPopup으로 이어져 지도 상에서 노면 상의 위치와 상세 정보를 아이콘을 통해 알 수 있다.  -  관리자 권한으로 반영 혹은 반려 버튼을 통해 ManagePage에서 노면 장애 데이터를 시각화할지, 미처리 또는 처리 상태로 변경할지 결정할 수 있다. |
| FaqPage | - 자주 질문할 만한 내용과 답변을 정리해둔 페이지이다.  - 작은 화살표를 누를 경우 숨겨져 있던 답변이 시각화된다. |
| LoginPage | - 관리자 로그인 기능이 구현되어 있다.  - 로그인 시 해당 유저의 권한이 관리자 권한으로 변경된다.  - 회원가입 버튼을 누르면 SignupPage에 접근할 수 있다. |
| SignupPage | - 관리자 회원가입이 가능하다.  - 유저의 정보를 데이터베이스로 전송한다. |
| SearchPopupPage | - 기간 범위 혹은 지역 단위로 필터링을 적용할 수 있다. |
| MapPopupPage | - 노면 장애 리스트를 클릭하였을 때 나오는 팝업으로, 노면 장애 사진과 자세한 정보를 확인할 수 있다. |

|  |  |  |
| --- | --- | --- |
| ROMED | usb\_cam | 카메라에서 v4l2 라이브러리를 사용하여 이미지를 읽고 이를 Sensor\_msg/Image 타입의 ros topic으로 publish함 |
| gps\_data | Ublox zed-f9p gps에서 위경도 데이터를 받아 Sensor\_msg/NavSatFix 타입의 ros topic으로 publsih함 |
| detectsegment | YOLO detecion, segmentation 모델을 사용하여 이미지에서 노면 장애를 검출하여, p\_project/DetectSeg 메시지 타입의 ros topic으로 publsih함 |
| risk\_analyst\_node | 노면 장애 검출 결과와 gps 데이터를 이용하여 Tmap api를 통해 추가적인 정보를 받아 이를 토대로 노면 장애의 위험도 분석 후 이를 서버에 업로드 |
| API 서버 | road\_monitoring\_project | 서버의 url 설정, 권한 설정 등 기본 설정을 저장 |
| road\_info | 노면 장애 정보를 관리, Post, Get, Put, Delete 기능을 지원하고 필터 기능, 주행 경로에 있는 장애 정보 반환 |
| users | 회원가입, 로그인 기능 및 유저 토큰 정보 관리 |
| utils | 기능 구현을 위한 함수 |
| media | 업로드된 노면 장애 사진을 저장 |
| 도비 Web | HomePage | 전체 노면 장애 위치를 지도에 시각화하여 전반적인 파악을 도우며, 아이콘을 클릭할 경우 해당 노면 장애의 상세 정보를 확인할 수 있다. |
| ManagePage | - 노면 장애 리스트를 확인할 수 있다.  - 각 리스트를 클릭하면 MapPopup으로 이어져 지도 상에서 노면 상의 위치와 상세 정보를 아이콘을 통해 알 수 있다.  - 관리자 권한으로 편집 버튼을 통해 노면 장애 데이터를 데이터베이스 상에서 삭제할 수 있다. |
| AlertsPage | - 노면 장애 리스트를 확인할 수 있다.  - 각 리스트를 클릭하면 MapPopup으로 이어져 지도 상에서 노면 상의 위치와 상세 정보를 아이콘을 통해 알 수 있다.  - 관리자 권한으로 알림 버튼을 통해 노면 장애 데이터의 정보를 알림으로 보낼 수 있다. |
| ComplaintsPage | - 노면 장애 리스트를 확인할 수 있다.  - 각 리스트를 클릭하면 MapPopup으로 이어져 지도 상에서 노면 상의 위치와 상세 정보를 아이콘을 통해 알 수 있다.  -  관리자 권한으로 반영 혹은 반려 버튼을 통해 ManagePage에서 노면 장애 데이터를 시각화할지, 미처리 또는 처리 상태로 변경할지 결정할 수 있다. |
| FaqPage | - 자주 질문할 만한 내용과 답변을 정리해둔 페이지이다.  - 작은 화살표를 누를 경우 숨겨져 있던 답변이 시각화된다. |
| LoginPage | - 관리자 로그인 기능이 구현되어 있다.  - 로그인 시 해당 유저의 권한이 관리자 권한으로 변경된다.  - 회원가입 버튼을 누르면 SignupPage에 접근할 수 있다. |
| SignupPage | - 관리자 회원가입이 가능하다.  - 유저의 정보를 데이터베이스로 전송한다. |
| SearchPopupPage | - 기간 범위 혹은 지역 단위로 필터링을 적용할 수 있다. |
| MapPopupPage | - 노면 장애 리스트를 클릭하였을 때 나오는 팝업으로, 노면 장애 사진과 자세한 정보를 확인할 수 있다. |
| 도비 App | Map\_screen | - Naver Map SDK 를 활용해 현재 위치, 목적지 설정, 경로 표시  - Naver Direction API로 경로 계산 및 실시간 위치 추적으로 거리 및 시간 갱신  - 100m 간격 경로 좌표 추출해 서버 전송 |
| Notification\_service | - FlutterLocalNotificationPlugin 초기화 및 Android Notification Channel 설정  - 지정된 payload 를 포함에 특정 알림 표시  - High priority로 설정한 알림을 표시해 사용자가 즉각 인지 가능 |
| Server | - 응답을 JSON으로 파싱, 위험도 레벨에 따라 가장 높은 위험 장애물 식별  -고위험 장애물 시 “감속”메시지, 저위험 장애물 시 “유의” 메시지를 알림 및 TTS 로 안내 |
| MyTTSService | - Andorid Native 환경에서 TTS를 구현하는 실제 코드  - Text 를 받아서 백그라운드에서 음성 안내 |
| MainActivity | -Flutter와 Android Native를 채널로 연동하는 기능.  - Background TTS & Notification을 위한 코드 |

1. 웹 디자인
2. 백엔드
3. 네비게이션 앱
4. 프로그램 디렉터리 구조

map\_screen.dart : 앱 화면 실행

Notification\_service.dart : TTS 데이터 전송 및 알림 팝업

Server.dart : 서버로부터의 데이터 송수신 및 전처리

MyTTSService : Android Native와 연동해 TTS 기능 실제 기능 구현

1. 실행시 로드 화면
2. 실행 화면
3. 목적지 지정 및 경로 안내
4. 네이버 지도 API 연동 및 경로 안내 시작

노면 장애물 알림

네이버맵을 통한 네비게이션 화면

1. 성능 및 품질 검증 결과
   1. AI 모델 성능 결과

기존 pre-trained model인 yolo v5 모델을 최근에 나온 yolo v11 모델로 학습시켜 유의미한 성능 향상 달성

* + 1. Result of YOLO v5 Segmentation Confusion Matrix
    2. Result of YOLO v11 Segmentation Confusion Matrix

Yolov5 Confusion Matrix Confusion Matrix

장애물 모델

1. 한계점 및 개선 방안
   1. 제약사항 및 미해결 과제
      1. AI 모델
         1. 야간/악천후 상황에서 성능 저하
         2. 특정 장애 유형에 대한 클래스 부족
      2. 백엔드
      3. 웹페이지
      4. 네비게이션 앱
         1. 대체 경로 안내 기능 한계
         2. 알림 연동의 제한적 구현

1. 기술적 개선 제안
   1. AI 모델
      1. 데이터 다양성 확보 및 재학습
      2. 다중 센서 융합 모델 도입(LiDAR, Rader)
   2. 백엔드
   3. 네비게이션 앱
      1. UI/UX 개선: 사용자 테스트를 통한 피드백 수집, 직관적 아이콘&색상 사용, 반응형 레이아웃 접근성 향상 기능 추가로 운전 중 조작 편의 성 강화
      2. 대체 경로 안내 기능 추가: 실시간 교통 정보 연계로 우회 경로 안내, 머신러닝 기반 경로 점수화로 상황별 최적 경로 추천 고도화
   4. 웹페이지

1)   디자인적인 요소 : 디자인을 css 개선으로 더 세련되고 현대적인

디자인을 적용하여 개선한다.

2) 네비게이션 앱 알림 연동 : 알림 전송 페이지에서 관리자가 전송

버튼을 누를 경우 네비게이션 앱에서 해당 노면 장애에 대한 정보를  팝업으로 띄우는 기능을 구현한다.

1. 확장 로드맵
   1. 서비스 배포
      1. 클라우드 인프라 확장
   2. 지자체 API 연동
2. 지자체별 도로 상태 API, 공사 일정, 교통량 데이터 연계로 더 풍부한 정정보제공
3. 위험도 분석 결과를 바탕으로 특정 구간에 대한 규제 강화(속도 제한,차선 변경 유도)
   1. 실시간 교통정보 연동
      1. 혼잡도, 사고, 공사 등 교통 정보 실시간 반영으로 운전자에게 최적 경로 재안내
      2. 단순 최단 거리보다 안전도, 혼잡도, 이동 시간등 다중 요인을 고려한 경로 점수화 시스템 도입
   2. ROMED 인지 시스템 소형화
      1. 하드웨어 경량화 및 에너지 효율 개선
      2. 양산 가능한 부품 표준화
      3. 카메라,라이다, 레이더등 다양한 센서를 플러그인 형태로 추가/교체 가능하게 하여 특정 도로 환경에 맞추어 유연한 커스터마이징 지원
4. 결론
   1. 개발 일정

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

* 1. 프로젝트 후기

1. 부록
   1. 데이터셋 자료
      * 1. 고해상도 도로노면 이미지 .(2024.10). AI-HUB. <https://www.aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&dataSetSn=71781>
        2. 도로 장애물 / 표면 인지 영상 수도권. (2023.05). AI-HUB. <https://aihub.or.kr/aihubdata/data/view.do?currMenu=&topMenu=&aihubDataSe=data&dataSetSn=179>
        3. 지자체 도로 정비 AI 학습용 데이터. (2022.07). AI-HUB. <https://aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&aihubDataSe=data&dataSetSn=557>
        4. 부산광역시 항만도로 컨테이너 차량에 의한 노면 파손 이미지 데이터. (2023.12). AI-HUB. <https://www.aihub.or.kr/aihubdata/data/view.do?currMenu=&topMenu=&aihubDataSe=data&dataSetSn=71554>
   2. 참고 문헌
      1. 김재중, 도명식, 김재진, 오주삼. (2020). 노면특성을 고려한 드론 영상 기반의 도로위험도 분석. 제 82회 대한교통학회 학술발표회 발표집
      2. 정상섬, 이광우, 박현도, 김정환. (2018). 도로포장 파손 위험도 예측 및 예측 시스템. 특허 출원 번호 10-2016-0109569
      3. 장윤호, 민경무, , 김. (2015). 도로의 노면 상태 감지시스템 및 방법. 특허 출원 번호 10-2013-0052409