

작업자 안전 위험 빅데이터 /AI 분석 웹 서비스 : 스마트 위치, 밴드 센서 기반의 위험 행동 인식

프로젝트 소개

배경

건설 현장에서의 낙상 등 안전 사고를 미연에 방지하기 위해 이미 일어난 사고 데이터를 수집하여 사고가 발생할 가능성이 높은 상황을 예측하여 조치를 취할 수 있도록 AI 프로그램을 개발하고자 한다.

목표

- 스마트 밴드 또는 스마트 헬멧으로 데이터를 수집.
- 수집된 생체 데이터를 통한 작업자 상태 관리.
- 근로자의 자이로스코프 센서 데이터, 맥박을 통한 행동 특성을 분석.
- 위험 예측을 활용해서 안전관리자에게 위험관리자 위치 및 상태를 알림.

요구 사항

프로젝트 요구 사항

순서	id	요구 사항
1	wsp01	근로자의 생체, 상태 데이터를 분석하여 관리자의 화면에 정보 구현 및 알림
2	wsp02	특정 건설 현장에 투입된 작업자 목록을 표출
3	wsp03	지도창에 작업자를 아이콘으로 표출하고 해당 아이콘을 클릭하면 작업자 신상 정보 구현
3-1	wsp0301	특정 작업자를 선택하면 작업체의 생체 신호, 자이로 센싱 데이터를 그래프로 표출
3-2	wsp0302	낙상, 추락 등의 위험 작업자 위치를 지도창에서 표출하고 특정 작업자를 선택하면 해당 작업의 위험 데이터와 그래프를 화면에 표출

서비스 요구 사항

순서	id	요구 사항
1	wss01	특정 건설 현장에 투입된 작업자 목록을 표출
1-1	wss0101	각 작업자를 클릭하면 상세 정보 표출
1-2	wss0102	작업자 검색 기능
1-3	wss0103	작업자 추가 기능
1-4	wss0104	작업자 삭제 기능
1-5	wss0105	작업자 상태 필터 기능
2	wss02	지도 아이콘별 번호 부여 (식별 가능)

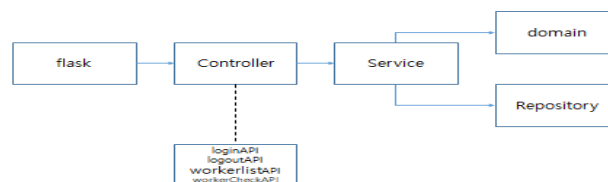
분석

데이터

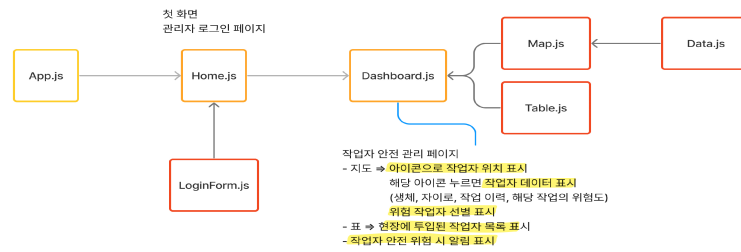
- 데이터 특징 : 연속형 데이터
- 데이터 형태 : csv
- 데이터 내용 : 작업자별 자이로스코프 및 가속도 데이터, 맥박
- 데이터 전처리 : numpy, pandas, sklearn
- 데이터 특징에 관한 시각화 : t-sen
- 분석 방법 : 머신러닝 - xgboost, 로지스틱회귀 / 딥러닝 - cnn, lstm
- 데이터 출처 : kaggle data(...), github data(...), 강남엔인코뉴스 수집 데이터

웹 개발

- 언어 및 프레임워크 : Java, JavaScript, Node.js, Python, SpringBoot, React, Flask
- 개발 환경 : Eclipse, VSCode, PyCharm
- 라이브러리 : axios, react-router-dom, jpa
- BE

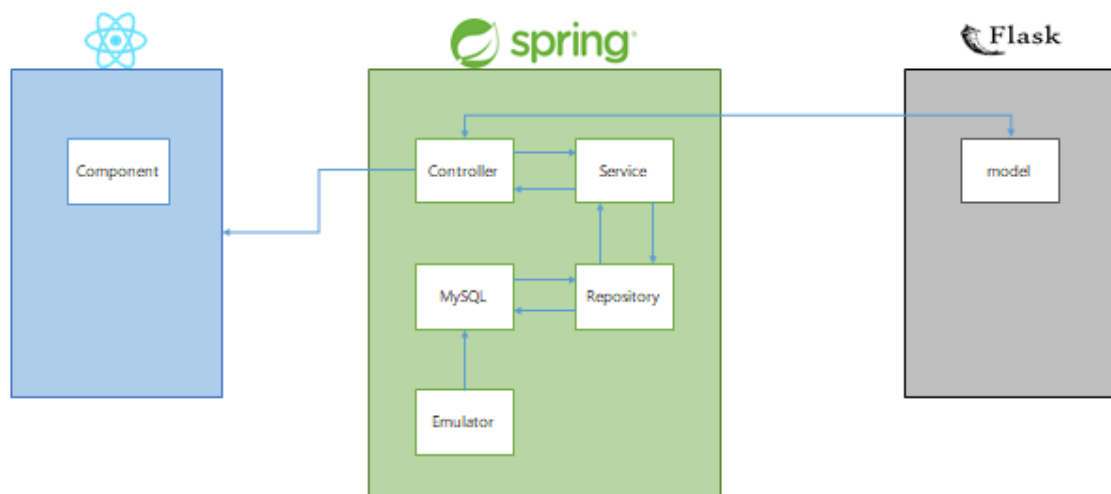


- FE



설계

시스템 구성도



DA

1. 데이터 수집 및 전처리

- 건설 현장에서 근로자의 자이로스코프 센서 데이터, 맥박 등의 데이터를 수집한다.
- 수집된 데이터를 전처리하여 이상치나 결측치 등을 제거하고, 필요한 경우 데이터를 정규화하는 등의 과정을 수행하여 데이터를 가공한다.

2. 가설 설정

- 건설 현장에서는 근로자가 반복되는 작업을 하게 되는데, 이 과정에서 근로자의 자세나

움직임이 불규칙해지면 사고 발생 가능성이 높아질 것이다.

- B. 따라서, 근로자의 움직임 데이터와 맥박 변화를 사용하여 이상 감지 알고리즘을 구현하고, 일정 이상의 불규칙한 패턴이 발견될 경우 사고가 발생할 가능성이 높다는 가설을 설정한다.

3. 모델링

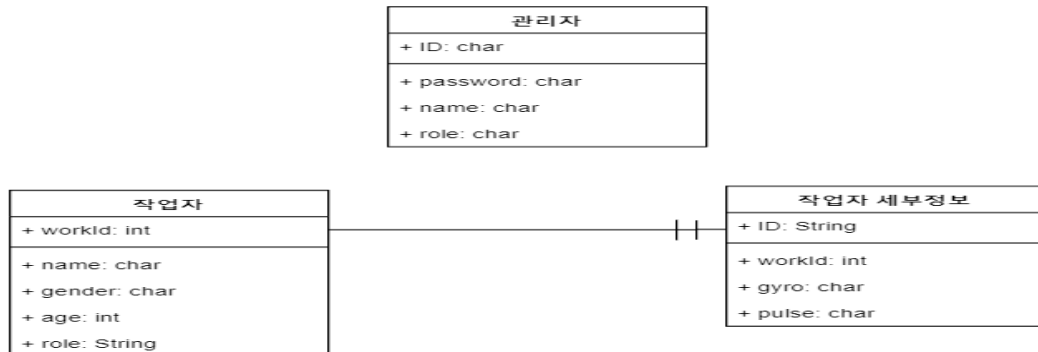
- A. 머신러닝/딥러닝 기반의 이상 감지 알고리즘을 구현하여 근로자의 상태를 모니터링 하고, 일정 이상의 불규칙한 패턴이 발견될 경우 즉시 경고를 발생시켜 사고를 예방하는 프로그램을 개발한다.

4. 모델 평가 및 보완

- A. 머신러닝/딥러닝 모델의 성능을 평가하기 위해 근로자의 실제 사고 데이터와 모델이 예측한 사고 데이터를 비교하여 모델의 정확도를 평가하고, 모델의 성능을 향상시키기 위해 추가적인 데이터나 특성을 수집하여 모델을 보완한다.

BE

- 데이터베이스 설계

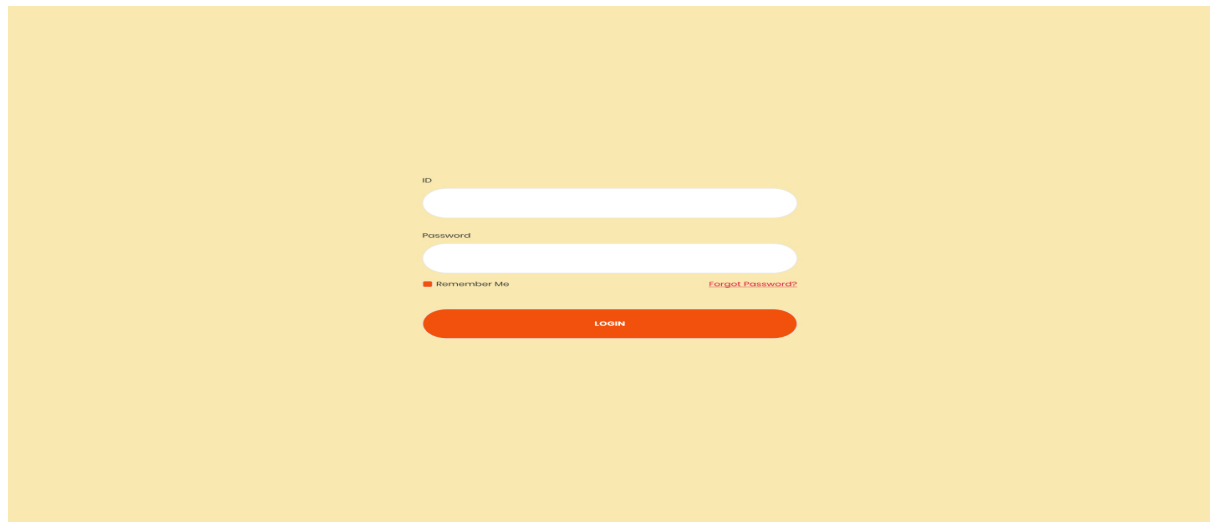


- REST API 설계

Index	Method	URI	Description
1	POST	/login	로그인
2	PUT	/logout	로그아웃
3	GET	/workerList	작업자 조회
4	POST	/workeradd	작업자 추가
5	DELETE	/workerdelete	작업자 삭제

FE

- Home 화면



A login form on a yellow background. It includes an ID input field, a Password input field, a "Remember Me" checkbox, a "Forgot Password?" link, and a "LOGIN" button.

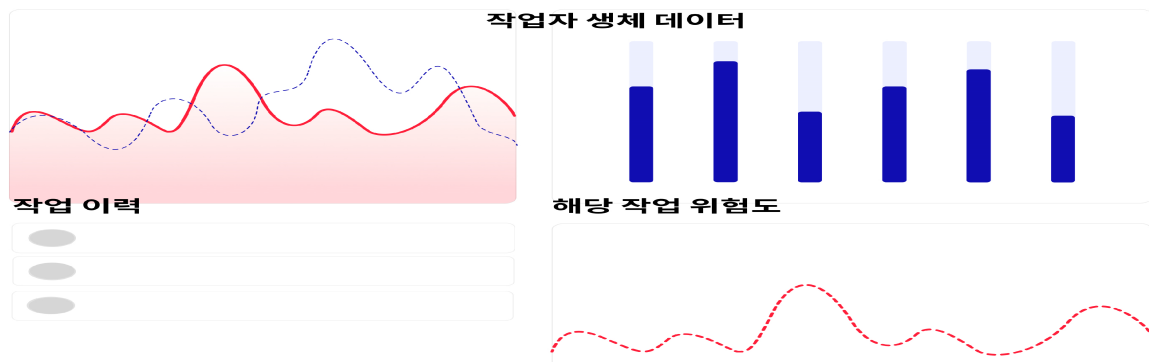
- Dashboard 화면



A dashboard screen with an orange header. The header contains a "KEEP ME" button, a toggle switch, and a "관리자" (Admin) label. The main content area is divided into two sections: "건설 현장 지도" (Construction Site Map) on the left, showing a map of a university campus with various buildings and a stadium, and "투입된 작업자 목록" (List of Input Workers) on the right, which is a table with a search bar and a list of workers.

- 상세 화면

아이콘을 누르면 표시되는 창



역할 분담

- 구민지 (팀장) : FE1, BE2
- 김단우 : DA1, FE2
- 양철민 : BE1, DA2

일정

	1주	2주	3주	4주	5주	6주	7주	8주	9주	10주
DA	D1	D2		D3		D4	D5	D6	D7	
BE	B1	B2		B3		B4	B5	B6	B7	
FE	F1	F2		F3		F4	F5	F6	F7	

DA

- D1 : 계획
- D2 : 데이터 전처리 & 머신러닝 기반 이상 감지 알고리즘 구현
- D3 : 모델 개선
- D4 : 플라스크를 활용한 API 작성
- D5 : 신경망 기반 이상 감지 알고리즘 구현 (딥러닝, 텐서플로)
- D6 : 모델 및 주요 서비스 개선
- D7 : 테스트, 보완 및 발표

BE

- B1 : 계획
- B2 : 모델 호출해서 REST API 생성
- B3 : 스키마 기타 API 확정
- B4 : 1차 구현, 개발 환경 구축, 로그인 시큐리티, 관리자 테이블, 각 서버 데이터 연결

- B5 : 테스트 및 피드백

- B6 : 2차 구현- 작업자 테이블, 각 서버의 요구·요청에 따른 API 작성, 데이터 저장

- B7 : 테스트, 보완 및 발표

FE

- F1 : 계획

- F2 : UI/UX 설계 및 시각화

- F3 : UI/UX 확정

- F4 : 1차 구현 - 컴포넌트별 구현, 디자인 레이아웃 구현

- F5 : 테스트 및 피드백

- F6 : 2차 구현 - BE와 연결, 페이지 구현 및 완성

- F7 : 테스트, 보완 및 발표

프로젝트 구현 환경

DA

- Python : 3.10.10

- xgboost : 1.7.5

- sklearn : 1.2.2

- tensorflow : 2.10.0

- Pycharm : 2023.1.1

BE

- Springboot : 3.0.6

- Java : 17

FE

- React : 18.2.0

- react-router-dom : 6.11.0

- axios : 1.4.0

· Node : 18.13.0

· VSCode : 1.77.3