



# '20세기 청년들' 프로젝트 계획서

## 1. 프로젝트 개요

### 1.1 프로젝트 이름

자동차 불량 이미지 데이터를 이용한 자동차 외관 하자 자동 검사 프로그램

### 1.2 프로젝트 설명

- 프로젝트 목표
  1. ai허브에서 제공하는 자동차 불량 이미지를 이용하여 인공지능 모델을 학습시켜 입력된 이미지 상의 차량의 하자 여부를 판별하는 모델 제작
  2. 모델을 구축 후 웹캠으로 실제 자동차 촬영하여 하자를 판별하는 시스템을 구축
- 주제 선정 이유
  1. 센서(웹캠)를 통하여 이미지를 입력받아 인공지능에 접목시켜 분류하는 기술은 스마트 팩토리 구현에 중요한 요소임
  2. 자동차는 대한민국을 대표하는 산업 중 하나로 그 중 검사공정에서 유용하게 사용 가능

### 1.3 주요 기술 스택

- 사용언어: 파이썬
- 프레임워크: YOLO(이미지 데이터 학습 알고리즘)
- 라이브러리: numpy(데이터 전처리), openCV(테스트 데이터 수집), Matplotlib(시각화), scikit-learn(성능 측정), PIL(이미지 불러오기)

## 2. 팀 구성 및 역할 분담

### 2.1 팀 소개

- 팀원 4명이 20세기에 태어났다는 공통점이 있음
- 취업을 위하여 열심히 살고 있음

## 2.2 역할 분담

이름	역할	주요 업무	
유승태	팀장	PM, 모델 학습 및 구축, 성능평가 지표 구현, 발표	
김다운	데이터 관리, 모델 구축	데이터 수집(다운로드, feature 선정), 모델 학습 및 구축	
한용찬	모델 및 서비스 시스템 구축	데이터 입/출력 구현, 모델 학습 및 구축, openAI 구현	
신예지	선행연구, 데이터 전처리, 모델 구축	선행연구, 데이터 전처리(json → csv) , 모델 학습 및 구축	

## 3. 프로젝트 세부 계획

### 3.1 일정 및 마일스톤

마일스톤	목표 날짜	세부 설명
데이터 수집 및 전처리	~ 2025-01-17	- ai허브에서 다운로드(500GB → 10GB 축소) 및 Feature 선정(4개) - 라벨데이터 csv 파일로 전처리 및 저장(json → csv) - Numpy, PIL을 통하여 이미지 불러오기
분류 모델 구축	~ 2025-01-22	- CNN 알고리즘 기반 모델 구축 및 학습 진행 - 모델 학습 방법 고도화(정규화, BN 등) - sklearn 기반 AUC score를 통한 모델 성능 평가 - 모델 구축 시간에 따라 이미지 생성 AI를 통한 추가 학습 진행
openCV 기반 프로그램 시연	~ 2025-01-23	- OpenCV(웹캠)를 통하여 실제 차량 이미지 입력 및 하자 여부 판별 - 하자 있을 시 사각형으로 부위 표시
발표 준비	~ 2025-01-24	- PPT 작성 및 발표 준비

### 3.2 개발 프로세스 및 팀 룰

#### • 개발 프로세스

1. 데이터 수집 및 전처리 후 각자 로컬에서 Jupyter notebook 환경에서 개발
2. 각자 분담된 내용에 대하여 git branch 생성 후 push하여 관리
3. 개발 완료 후 main branch에 merge하여 시스템 완성

- 팀 룰

1. 하루하루 진행된 사항을 노션 페이지에 기록

### The work done

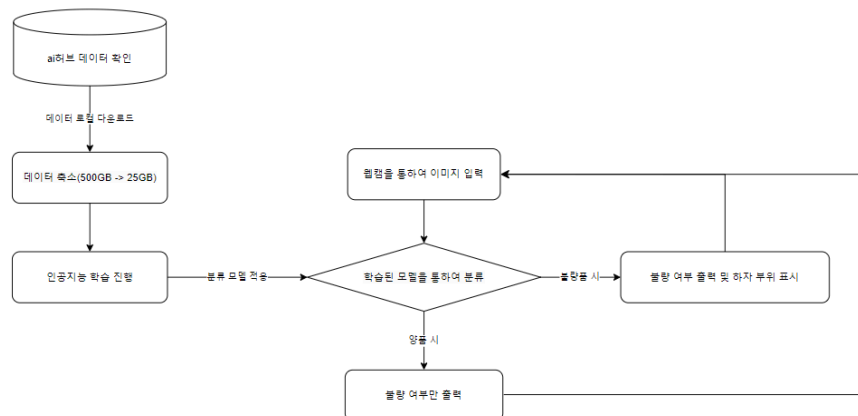
2. commit message는 feat(기능 추가), fix(버그수정) 등 직관적인 단어로 작성

## 4. 기능 명세서

### 4.1 핵심 기능 리스트

기능 이름	설명	우선순위 (High/Medium/Low)	담당자
검사 대상 입력	캠을 통하여 검사하려는 자동차의 이미지를 가져온다.	High	한용찬, 신예지
하자 여부 검사	모델을 통하여 하자 여부 검사	High	전 인원
검사 결과 출력	검사 여부 출력	High	유승태, 김다운
하자 위치 출력	불량 시 손상 부위 표시	Medium	전 인원

### 4.2 플로우 차트



## 5. 기타

### 5.1 참고 자료 및 링크

AI허브: <https://aihub.or.kr/aihubdata/data/dwld.do?currMenu=115&topMenu=100>

## 5.2 부가 정보

git 주소: [https://github.com/smartfactory-project-2/smartfactory\\_project\\_2](https://github.com/smartfactory-project-2/smartfactory_project_2)