

---

노이로제(**Noise + Zero**)

융복합 PROJECT

제안서

---

1 조

# 목차

## I. 프로젝트 개요

- i. 프로젝트명
- ii. 프로젝트 기간 및 일정
- iii. 프로젝트 제안 배경 및 목적

## II. 제안 사항

- i. 구현 기능
- ii. 활용기술

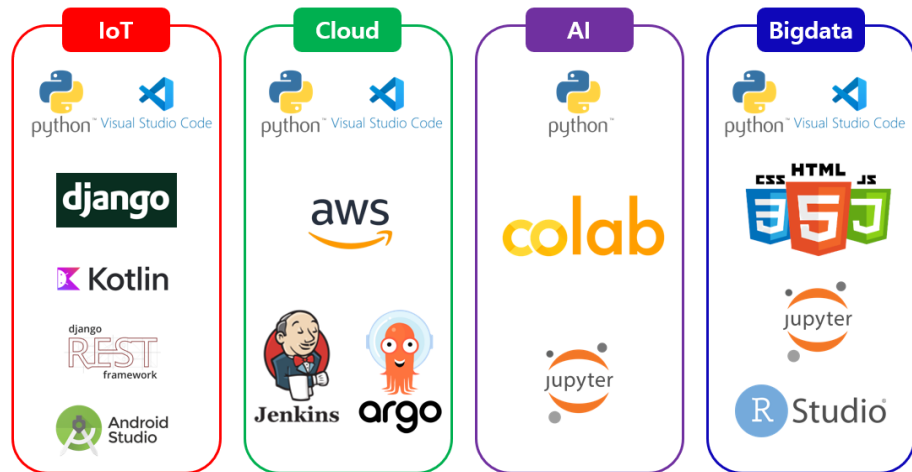
## I. 프로젝트 개요

프로젝트명	충간소음 측정 및 관리, 공유 커뮤니티 서비스
프로젝트 기간	2023년 05월 31일 ~ 2023년 07월 05일
프로젝트 제안 배경 및 목적	최근 증가하는 세종시의 충간소음 민원에 대응하여 빅데이터, <b>IoT, AI</b> , 클라우드 융합기술을 활용한 데이터 기반 관리 서비스를 개발하여 충간소음 문제를 예방하고 해결한다.

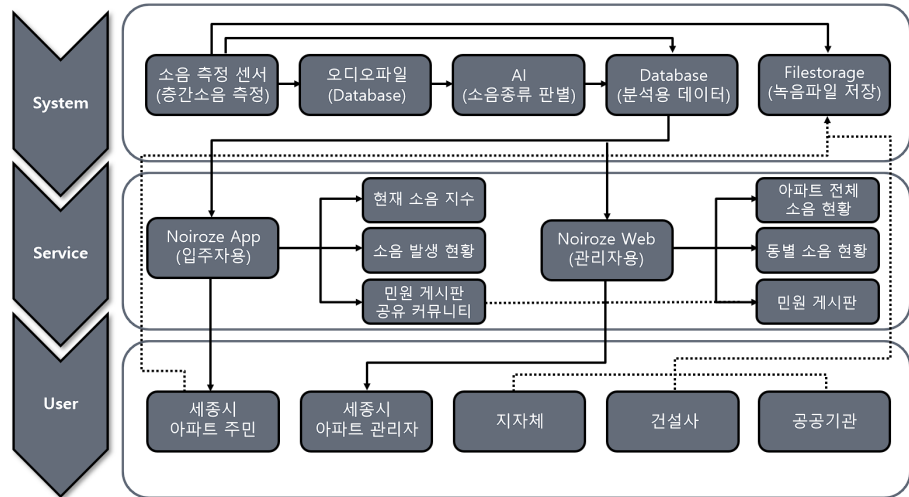
## II. 제안 사항

구현 기능	<div>1. 충간소음 측정(감지)을 위한 <b>IoT</b> 기기 설치 및 데이터 수집</div> <div>2. 대량의 충간소음 데이터를 처리하고 저장하기 위한 빅데이터, 클라우드 기술 적용</div> <div>3. 소음 종류 예측을 위한 <b>AI</b> 알고리즘 개발</div> <div>4. 서비스의 확장성과 유연성을 위해 클라우드 환경 구축</div> <div>5. 세종시의 아파트 거주자들을 대상으로</div> <div>충간소음 측정 및 관리 서비스</div>
-------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

활용 기술



사용 기술	사용언어	통신기술	협업 tool
<b>Framework</b> <ul style="list-style-type: none"> <li>· Django</li> <li>· Android Studio</li> </ul> <b>Cloud</b> <ul style="list-style-type: none"> <li>· AWS EC2</li> <li>· Lambda</li> <li>· S3</li> <li>· RDS</li> <li>· EKS</li> </ul> <b>AI // DataAnalysis</b> <ul style="list-style-type: none"> <li>· Jupyter notebook</li> </ul> <b>Big data</b> <ul style="list-style-type: none"> <li>· R Studio</li> </ul> <b>IDE</b> <ul style="list-style-type: none"> <li>· Arduino</li> <li>· AWS Cloud9</li> <li>· VS Code</li> </ul>	<b>back-end</b> <ul style="list-style-type: none"> <li>· Python</li> </ul> <b>front - end</b> <ul style="list-style-type: none"> <li>· HTML5</li> <li>· CSS</li> <li>· jQuery</li> <li>· JavaScript</li> </ul> <b>Android App</b> <ul style="list-style-type: none"> <li>· Kotlin</li> </ul> <b>Arduino</b> <ul style="list-style-type: none"> <li>· C++</li> </ul>	<ul style="list-style-type: none"> <li>· HTTP</li> <li>· TCP/IP</li> <li>· Serial</li> <li>· REST API</li> </ul>	<ul style="list-style-type: none"> <li>· ZOOM</li> <li>· GitHub</li> <li>· Google Docs</li> <li>· Notion</li> <li>· Slack</li> </ul>



1. IoT센서의 설치 장소에 따라 특정 ID를 부여하고, 해당 장소에서 기준치 이상의 소음 발생 시 감지한 센서 값 (데시벨 측정 등)을 HTTP통신을 이용하여 API서버로 전송
2. API서버로 넘어온 값을 데이터베이스에 저장  
소음발생위치, 종류, 크기 등의 데이터를 AI모델이 분류 및 가공하여 웹 및 앱에서 활용
3. 특정 데이터(소음, 진동)가 기준치를 초과하면  
데이터 및 녹음 파일을 서버로 전송
4. 메인서버 DB에 저장된 데이터를 그래프형태로  
소음 종류, 소음 크기 등을 웹페이지 및 어플리케이션에 구현  
특정 데이터가 범위를 초과하면 소음 지수와 함께  
주요 원인을 보여줌.