



# 조립 가설기자재 품질평가 관리 시스템

## 설계 목적

- 건설 현장에서 사용되는 가설기자재(간단히 말해 건축재료)를 웹으로 관리하고자 함.
- 사용자 간의 의사소통을 체계적으로 제작

## 기본 구성

- 사용자는 크게 일반사용자, 시스템 관리자로 나뉨
  - 일반사용자는 가설기자재 품질관련 문서만 접근가능
  - 시스템관리자는 일반관리자 권한 + 웹 데이터 관리를 맡음
- 이후 사용자는 4종류의 소분류로 나뉨
  - 일반관리자
  - 현장대리인
  - 일반 건설사업관리기술인
  - 총괄 건설사업관리기술인
- 시스템 관리자는 기본정보 문서를 관리
  - 현장 관리
  - 설치위치명 등록
  - 문서 등록
  - 사용자 등록/삭제
- 사용되는 문서 4종류는 다음과 같음
  - 구조 안전성 검토
  - 자재 공급원

- 품질검사
- 설치작업 체크리스트
- 이 외의 검색기능인 결과보고, 공지사항, 법률정보가 존재

## 요구사항 분석

### 최종 설계서

[https://s3-us-west-2.amazonaws.com/secure.notion-static.com/855033d4-1b43-42f0-9b69-fe205bb980fe/화면설계서\\_220304.zip](https://s3-us-west-2.amazonaws.com/secure.notion-static.com/855033d4-1b43-42f0-9b69-fe205bb980fe/화면설계서_220304.zip)

- 해당 문서를 확인하고 전체적인 흐름을 파악
- 대부분의 내용이 현재와 같음

### 일부 수정본

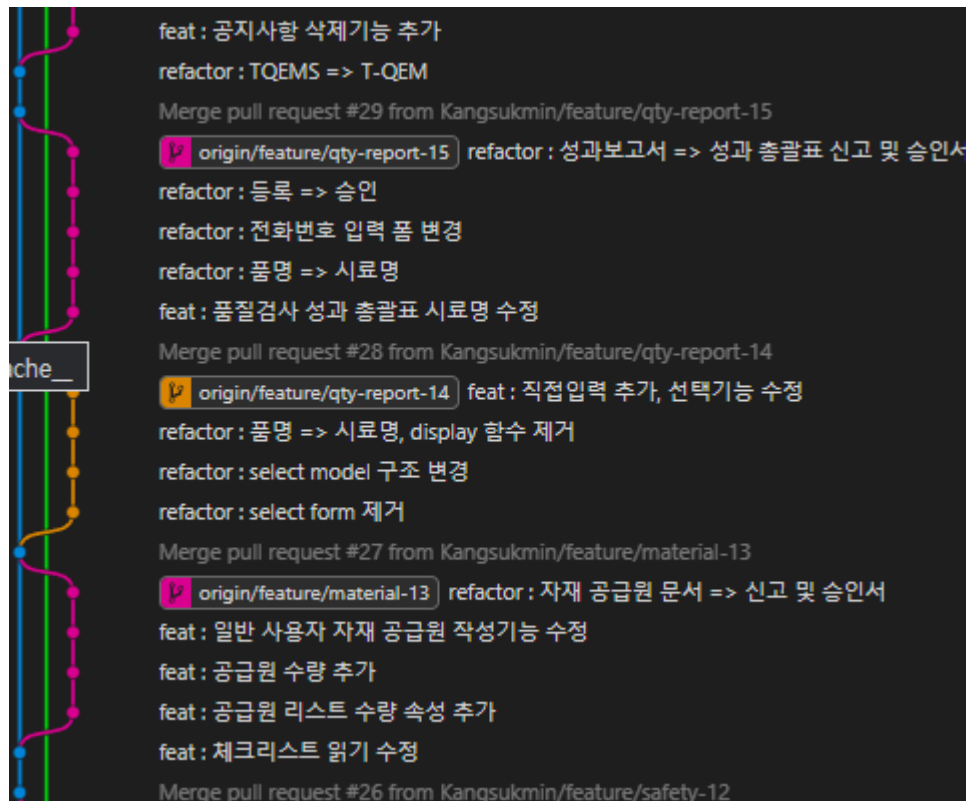
<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/3ea239e3-2bd9-4fb4-ac4d-57452ba44e5a/220406.zip>

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/2dc7ead2-42df-44bc-a0ca-9be701471a8a/220414.zip>

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/6580fd5b-4459-4c6d-b2cd-dcc904b7f786/220425.zip>

- 몇 가지 내용에 대한 수정본이 담겨있음

## 코드 관리



- 코드는 전부 Git으로 관리됨
- 롤백이 필요한 경우 git의 버전관리를 통하여 롤백가능

## 사용된 기술

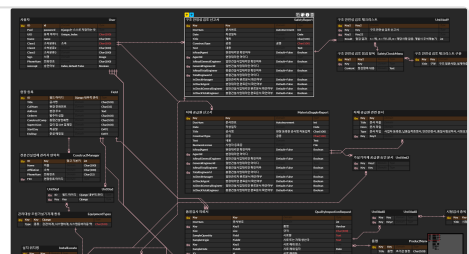
- Django 풀스택
- Poetry 패키지 관리자

## 디비 구조

Copy of inha\_new

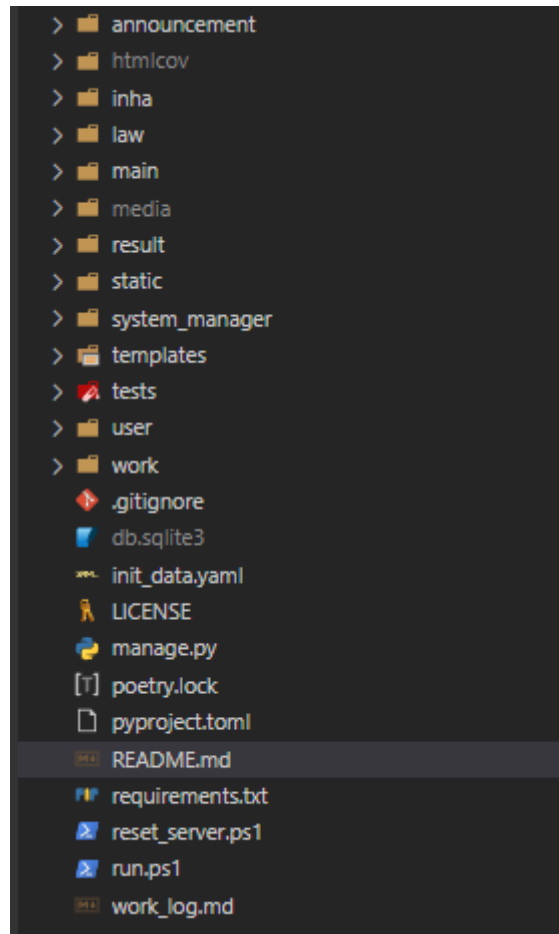
Draw ERD with your team members. All states are shared in real time. And it's FREE. Database modeling tool.

ERD <https://www.erdcloud.com/d/Cm5eGQ8ux8qrx6pyg>



- ERD 클라우드를 이용해 제작
- 우측 상단의 ERD 복사를 통해 사용하길 권장

# 코드 구조



- announcement : 공지사항
- inha : 프로젝트 세팅
- law : 법령정보
- result : 결과보고
- system\_manager : 기본정보 등록
- user : 로그인/회원가입 관련
- work : 업무
  - safety : 구조 안전성 검토
  - material : 자재 반입
  - quality\_request : 품질검사 의뢰
  - quality\_report : 품질검사 성과보고

- before\_install : 설치작업 전 체크리스트
- install : 설치작업 중 체크리스트


그 외 초기 작업은 코드내 [Readme.md](#) 참고

## 기초지식

- 도메인 주소
  - 아이피 주소 형식의 URL을 네이밍 주소로 변환
  - 현재 내도메인.한국 을 사용하며 유료신청을 원할경우 가비아로 대체
- Poetry

### 파이썬 패키지 관리툴 poetry 소개

poetry는 파이썬 의존성 관리 툴입니다. 단순히 의존성 관리만 잘하는 툴이라면 추천할 이유가 별로 없을 것입니다. poetry는 단순히 잘 한다 정도를 넘어서서 매우 잘 지원 해줍니다. poetry.lock 을 사용해서 프로젝트의 의존성을 다른 환경에서도 동일하게 유지할 수 있고, 각 환경에서의 자동완성도 꽤나 잘 지원합니다.

 <https://blog.gyus.me/2020/introduce-poetry/>

- 파이썬 패키지 관리자