

AI 논술 첨삭 웹 애플리케이션 요구사항 정의서

1. 프로젝트 개요

본 문서는 Django 프레임워크 기반의 'AI 논술 첨삭 웹 애플리케이션' 개발에 대한 기능적 및 비기능적 요구사항을 정의한다. 본 애플리케이션은 사용자가 논술 답안 이미지를 업로드하면, RAG(검색 증강 생성) 기술을 활용한 AI가 대학별 기출문제의 채점 기준과 모범 답안을 바탕으로 심층적인 첨삭 피드백과 예상 점수를 제공하는 것을 목표로 한다.

2. 기능적 요구사항 (Functional Requirements)

- 사용자가 시스템을 통해 무엇을 할 수 있는가에 대한 정의

ID	기능 그룹	요구사항 상세	우선순위
FR-01	사용자 인증	사용자는 아이디와 비밀번호를 입력하여 회원가입을 할 수 있다.	High
FR-02		아이디가 이미 존재할 경우, 적절한 오류 메시지를 표시한다.	High
FR-03		회원가입 시 입력한 두 개의 비밀번호가 일치하지 않으면 오류 메시지를 표시한다.	High
FR-04		사용자는 아이디와 비밀번호로 로그인할 수 있다.	High
FR-05		로그인 실패 시 (예: 비밀번호 불일치), "아이디 또는 비밀번호가 잘못되었습니다."와 같은 오류 메시지를 표시한다.	High
FR-06		로그인한 사용자는 로그아웃할 수 있다.	High
FR-07	문제 열람	사용자는 대학, 연도, 문항을 드롭다운 메뉴에서 선택하여 문제지를 조회할 수 있다.	High
FR-08		선택된 문제지는 고화질 이미지 형태로, 페이지를 넘길 수 있는 캐러셀 뷰어로 제공된다.	Medium

FR-09		문제 열람 페이지에는 사용자가 시간을 측정할 수 있는 타이머(시작/일시정지/초기화) 기능이 제공된다.	Medium
FR-10	답안 제출 및 첨삭	사용자는 채점받을 문제(대학, 연도, 문항)를 선택할 수 있다.	High
FR-11		사용자는 직접 작성한 답안 이미지를 업로드할 수 있다.	High
FR-12		'AI 첨삭 실행' 버튼을 누르면, 시스템은 업로드된 이미지에서 OCR 기술을 통해 텍스트를 추출한다.	High
FR-13		시스템은 추출된 텍스트와 문제 정보를 바탕으로 AI 첨삭을 실행하고, 그 결과를 데이터베이스에 저장한다.	High
FR-14	결과 조회	첨삭 완료 후, 사용자는 상세 결과 페이지로 자동 이동한다.	High
FR-15		결과 페이지에는 학생 답안(OCR), 모범 답안, AI 코멘트(총평, 잘한 점, 아쉬운 점)가 표시된다.	High
FR-16		AI 코멘트 중 '수정 제안' 항목은 원문과 수정문을 비교하여 변경된 부분이 시각적으로 강조되어 표시된다.	Medium
FR-17		결과 페이지에는 AI 챗봇 기능이 제공되어, 사용자가 첨삭 결과에 대해 추가 질문을 할 수 있다.	High
FR-18	첨삭 히스토리	사용자는 '나의 첨삭 히스토리' 페이지에서 자신이 제출했던 모든 첨삭 기록 목록을 최신순으로 볼 수 있다.	High
FR-19		히스토리 목록에서 특정 항목을 클릭하면, 해당 첨삭의 상세 결과 페이지로 이동한다.	High

3. 비기능적 요구사항 (Non-Functional Requirements)

- 시스템이 어떻게 작동해야 하는가에 대한 품질 및 제약 조건 정의

ID	구분	요구사항 상세	우선순위
NFR-01	보안	답안 제출, 첨삭 결과 조회, 히스토리 조회 등 모든 핵심 기능은 반드시 로그인이 된 사용자만 접근할 수 있어야 한다.	High

NFR-02		사용자의 비밀번호는 데이터베이스에 반드시 암호화된 상태로 저장되어야 한다.	High
NFR-03		사용자는 자신의 검색 기록만 조회할 수 있어야 하며, 다른 사용자의 기록에는 접근할 수 없어야 한다.	High
NFR-04		OpenAI API 키와 같은 민감한 정보는 소스 코드에 직접 노출되지 않고, 환경 변수(.env)를 통해 관리되어야 한다.	High
NFR-05	성능	AI 검색 요청 후, 사용자는 30초 이내에 결과를 확인할 수 있어야 한다. (네트워크 및 API 상태에 따라 변동 가능)	Medium
NFR-06		AI 챗봇의 응답 시간은 사용자 질문 후 10초 이내를 목표로 한다.	Medium
NFR-07		페이지 로딩 시, 불필요한 AI 모델(OCR, Grader) 로딩을 지연시켜 초기 서버 구동 속도를 최적화해야 한다 (싱글턴 패턴 적용).	High
NFR-08	사용성	모든 페이지는 데스크톱 및 모바일 환경에서 적절하게 표시되는 반응형 웹 디자인을 따라야 한다.	Medium
NFR-09		AI 챗봇이 답변을 생성하는 동안, 사용자에게 '입력 중'임을 시각적으로 알려주는 로딩 인디케이터를 표시해야 한다.	Low
NFR-10	유지보수	새로운 대학의 기출문제를 추가할 때, 코드 수정 없이 PDF 및 JSON 파일 추가와 자동화 스크립트 실행만으로 데이터가 업데이트될 수 있어야 한다.	High
NFR-11		Django 뷰 로직은 기능(main, auth, api)에 따라 여러 파일로 분리하여 코드의 가독성과 관리 용이성을 높여야 한다.	Medium