

## 캡스톤디자인 II 중간보고서(표지)

프로젝트명 : 리딩브릿지 - AI 기반 난독증 아동 맞춤형 학습 교재 자동 변환 플랫폼 개발

# 캡스톤 디자인Ⅱ, 중간보고서

Version 1.0

개발 팀원 명(팀리더):최은기  
김지훈  
서동현

대표 연락처:010-2655-4276  
e-mail: deveungi@gmail.com

## 캡스톤 디자인 II 중간보고서 내용

1. 요구사항 정의서에 명세된 기능에 대하여 현재까지 진척된 결과 및 그 내용을 기술하시오.

### 1-1. 소스코드 일부 및 실행환경

시행환경에 대해서는 다양한 운영환경에서 한번에 실행하고 관리할 수 있도록 컨테이너 환경(Docker)를 사용하였다.

각 컨테이너 환경의 구성도는 다음과 같다.

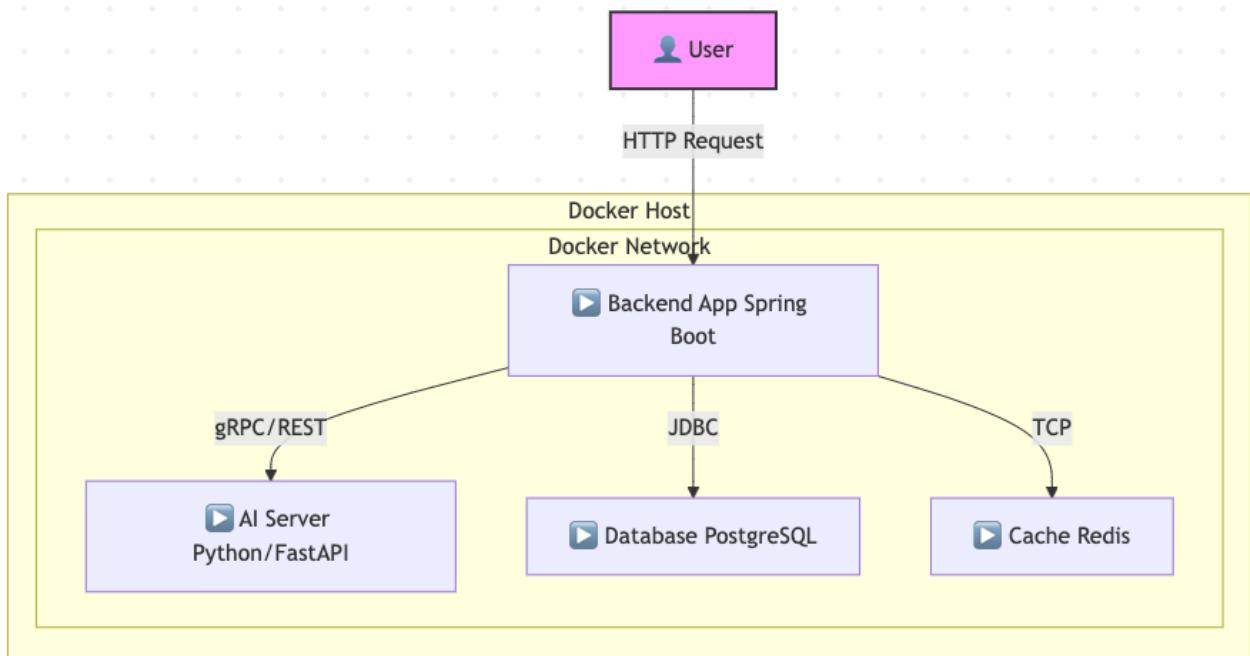


그림 1 컨테이너 서비스 구조도

실행방법은 각 서비스 내부에 정의된 docker-compose.yml 파일을 이용해 컴포즈를 사용하여 오케스트레이션을 수행한다.

```
docker-compose up --build -d
```

프로젝트 전체 구조와 코드스타일을 보여주는 핵심 파일을 선정하였다.

클라이언트 <-> 서버 구조간 엔드포인트를 의미하는 대표적인 모듈의 일부이며, 가독성 문제로 핵심 구현부는 제외하였다.

```

// dyslexia/controller/GuardianController.java

@RestController
@RequestMapping("/guardian")
@RequiredArgsConstructor
public class GuardianController {

    private final GuardianService guardianService;

    @PostMapping("/signup")
    public ResponseEntity<SignUpResponseDto> signUp(@RequestBody GuardianSignUpRequestDto
guardianSignUpRequestDto) {
        // ...
    }

    @GetMapping("/students")
    public ResponseEntity<List<Student>> getStudentsByGuardian(@RequestParam String
guardianId) {
        // ...
    }

    @PostMapping("/assign-textbook")
    public ResponseEntity<Void> assignTextbookToStudent(@RequestBody
TextbookAssignmentRequestDto requestDto) {
        // ...
    }
}

```

그림 2 보호자 인증 및 인가 앤드포인트 정의 - 컨트롤러

```

// dyslexia/service/GuardianService.java

@Service
@RequiredArgsConstructor
@Transactional
public class GuardianService {

    private final GuardianRepository guardianRepository;
    private final StudentRepository studentRepository;
    private final TextbookRepository textbookRepository;
    // ...

    public SignUpResponseDto signUp(GuardianSignUpRequestDto guardianSignUpRequestDto) {
        // ...
    }

    public List<Student> getStudentsByGuardian(String guardianId) {
        // ...
    }

    public void assignTextbookToStudent(TextbookAssignmentRequestDto requestDto) {
        // ...
    }
}

```

그림 3 보호자 및 인증 비즈니스 구현 - 서비스

## 1-2. 시스템 아키텍쳐

본 시스템은 안정성과 역할 분리를 고려하여 **온프레미스(On-premise)** 환경의 물리 서버 2대에 각 기능을 분산 배치하는 **2-Tier 아키텍처**로 설계하였다.

아래 다이어그램은 사용자의 요청이 웹/애플리케이션 서버와 AI/데이터베이스 서버를 거쳐 처리되는 과정과 전체적인 데이터 흐름의 예시이다.

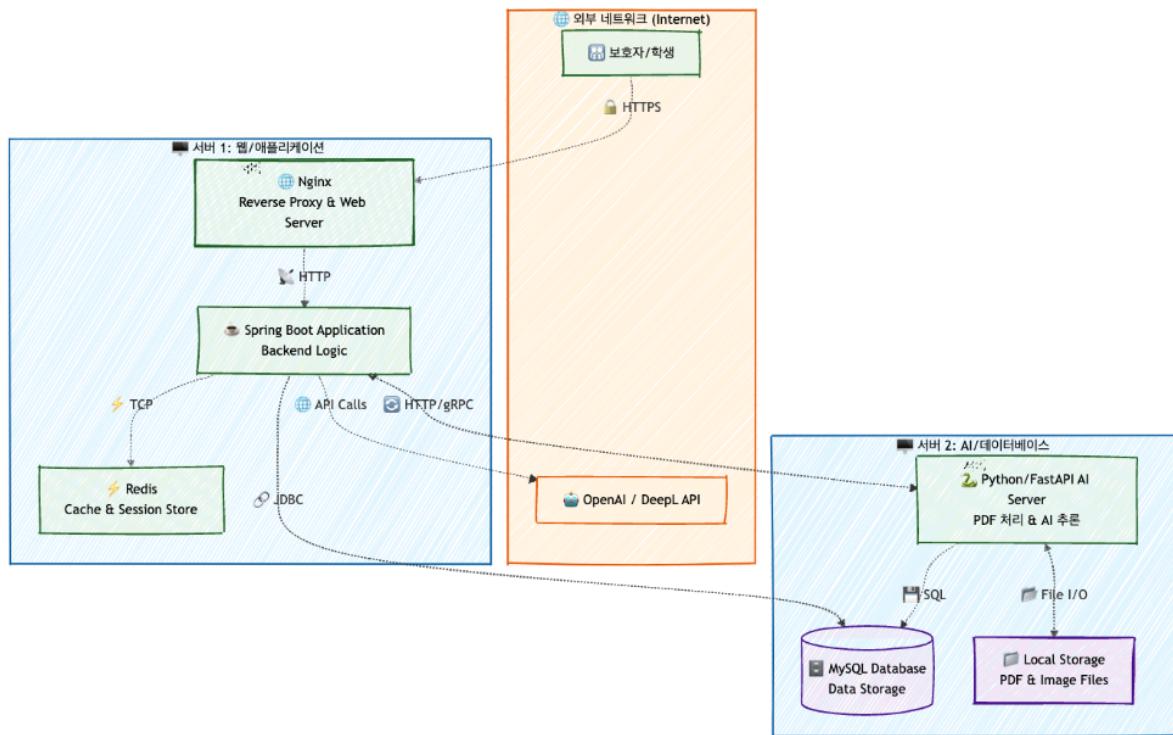


그림 4 시스템 아키텍쳐 구성도

## 1-3. 기능별 상세 요구사항

### 1-3-1. 사용자 관리 기능

ID	명칭	상세 내용	중요도
USER-001	보호자 회원가입	보호자는 아이디, 비밀번호, 이름, 이메일, 전화번호를 입력하여 회원가입을 요청한다.	상
USER-002	학생 회원가입	학생은 아이디, 비밀번호, 이름, 보호자 연동 코드, 관심사를 입력하여 회원가입을 요청한다. 시스템은 보호자 코드의 유효성을 확인하고, 유효한 경우 해당 보호자에게 학생 계정을 종속시켜 생성한다.	상
USER-003	로그인	사용자는 아이디와 비밀번호를 입력하여 로그인을 요청한다.	상
USER-004	보호자-학생 연동	보호자는 자신의 고유 연동 코드를 조회할 수 있다. 학생은 회원가입 시 이 코드를 사용하여 보호자와 계정을 연결한다.	중
USER-005	보호자의 학생 목록 조회	보호자는 자신에게 등록된 모든 학생의 목록을 조회한다.	중

### 1-3-2. 문서 및 교재 관리 기능

ID	명칭	상세 내용	중요도
DOC-001	<b>PDF 문서 업로드</b>	보호자는 학습에 사용할 PDF 파일을 시스템에 업로드하고, 파일은 서버의 지정된 위치에 저장되고, 문서 정보(ID, 제목, 파일 경로 등)가 데이터베이스에 기록된다. 비동기적으로 문서 변환 프로세스가 시작된다.	상
DOC-002	<b>업로드된 문서 목록 조회</b>	보호자는 자신이 업로드한 모든 문서의 목록과 각 문서의 변환 상태(처리 중, 완료, 실패)를 조회한다.	상
DOC-003	<b>문서 변환 상태 실시간 확인</b>	보호자는 문서 변환 작업의 현재 진행 상태(e.g., 텍스트 추출 중, 이미지 분석 중)를 실시간으로 확인한다. Redis에 저장된 현재 작업 상태 정보가 클라이언트로 전송된다.	상
DOC-004	<b>변환 완료된 교재를 학생에게 할당</b>	보호자는 변환이 완료된 교재를 선택하여 특정 학생 또는 여러 학생에게 학습하도록 할당한다.	중

### 1-3-3. 학습 콘텐츠 및 진행 관리 기능

ID	명칭	상세 내용	중요도
LRN-001	<b>할당된 교재 목록 조회</b>	학생은 자신에게 할당된 모든 디지털 교재의 목록(제목, 총 페이지 수, 학습 진행률 등)을 조회한다.	상
LRN-002	<b>교재 페이지 내용 조회</b>	학생은 특정 교재의 특정 페이지를 요청하여, 변환된 학습 콘텐츠(텍스트 블록, 이미지, AI 팁 등)를 조회한다.	상
LRN-003	<b>학습 진도 업데이트</b>	학생이 페이지 학습을 완료하면, 해당 페이지의 학습 완료 상태가 서버에 기록 및 갱신된다.	상
LRN-004	<b>학습 진도 확인 (보호자)</b>	보호자는 학생의 교재별, 페이지별 학습 진행 상황을 모니터링하여 학습 현황을 파악한다.	중

### 1-3-4. AI 보조 기능

ID	명칭	상세 내용	중요도
AI-001	어휘 분석 및 설명 요청	학생이 페이지 내의 특정 단어나 문장을 선택(드래그 등)하고 '어휘 분석' 기능을 요청한다. 시스템은 선택된 텍스트를 AI 모델에 전달하여, 학생의 눈높이에 맞는 쉬운 설명, 예문, 형태소/음운 분석 결과를 생성하여 반환한다.	상
AI-002	텍스트를 이미지로 시각화	학생이 특정 문단이나 문장을 선택하고 '이미지로 보기' 기능을 요청한다. 시스템은 해당 텍스트의 핵심 의미를 요약한 프롬프트를 생성하여 이미지 생성 AI 모델을 호출하고, 결과 이미지를 사용자에게 보여준다.	상
AI-003	페이지 핵심 내용 요약 (AI Tip)	시스템은 페이지 변환 과정에서 해당 페이지의 핵심 내용을 AI를 통해 미리 요약하거나, 학생들이 어려워할 만한 부분에 대한 팁을 생성한다. 생성된 AI 팁은 페이지 콘텐츠와 함께 저장되었다가, 학생이 페이지를 조회할 때 함께 제공된다.	상

### 1-4. 설계 모델

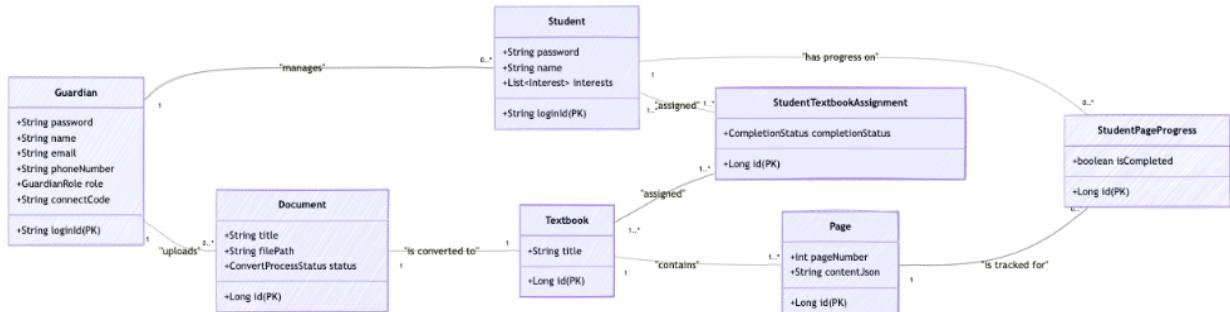


그림 5 도메인 결합 구조 (UML)

다음 클래스 다이어그램은 시스템의 핵심 객체와 그 관계를 정의한다.

- **Guardian (보호자)과 Student (학생):** 시스템의 주요 사용자이다. 한 명의 보호자는 여러 학생을 관리할 수 있는 일대다(1:N) 관계를 가진다.
- **Document (문서)와 Textbook (교재):** 보호자가 업로드한 원본 PDF 파일(Document)은 시스템에서 학습 가능한 디지털 Textbook(교재)으로 변환된다. 이는 일대일(1:1) 관계를 의미한다.
- **Textbook (교재)과 Page (페이지):** 하나의 Textbook은 여러 개의 Page로 구성되어, 일대다(1:N) 관계를 형성한다.
- **Student, Textbook, Assignment:** Student와 Textbook은 StudentTextbookAssignment라는 중간 객체를 통해 다대다(N:M) 관계를 맺는다. 이를 통해 어떤 학생에게 어떤 교재가 할당되었는지를 관리한다.
- **Student, Page, Progress:** StudentPageProgress 객체는 특정 학생(Student)이 특정 페이지(Page)의 학습을 완료했는지 여부를 추적한다.

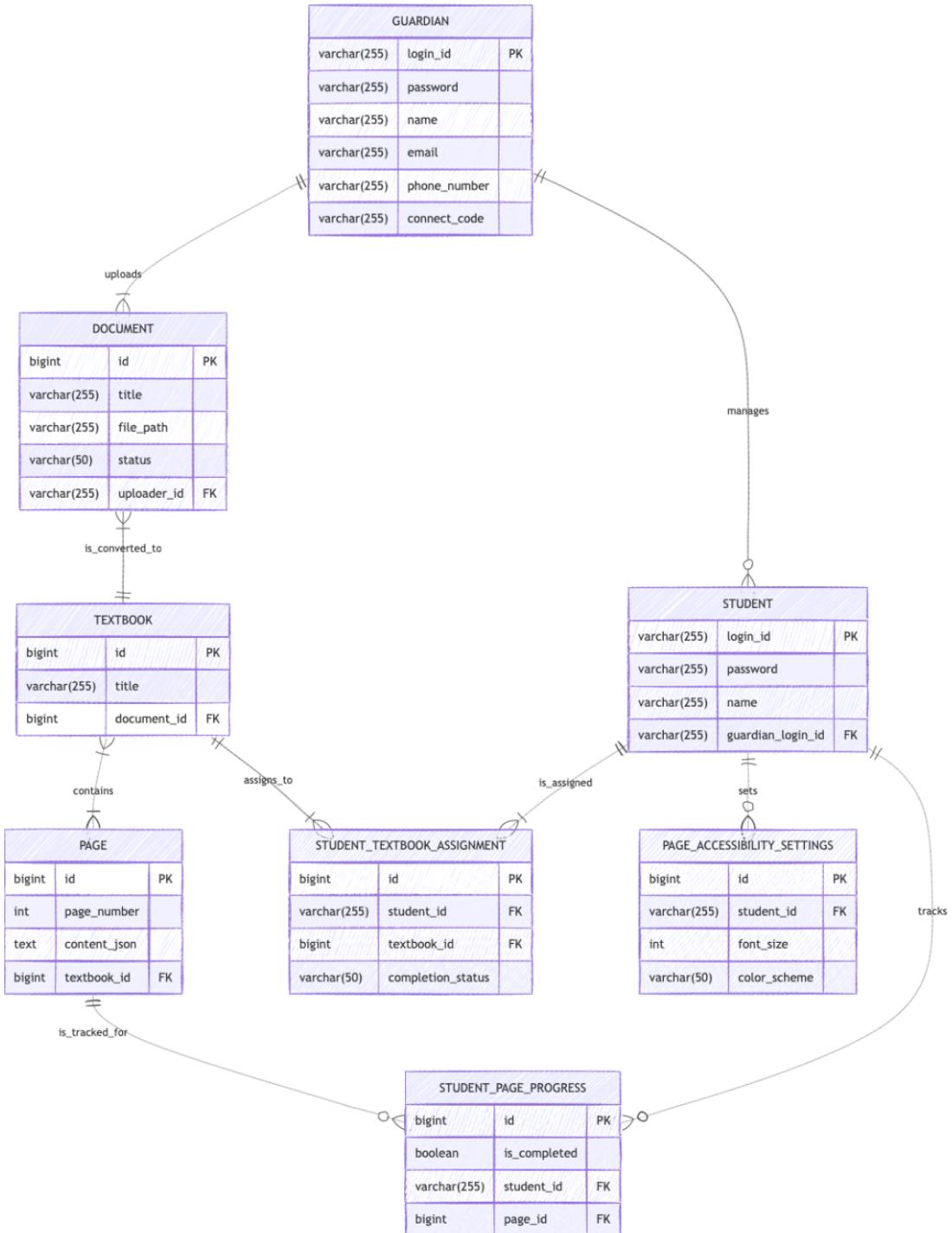


그림 6 데이터베이스 ERD

2. 프로젝트 수행을 위해 적용된 추진전략, 수행 방법의 결과를 작성하고, 만일 적용과정에서 문제점이 도출되었다면 그 문제를 분석하고 해결방안을 기술하시오.

본 프로젝트는 **애자일(Agile)** 방법론을 기반으로 주 단위의 스프린트(Sprint)를 설정하여 진행했습니다. 각 스프린트마다 계획 수립 → 설계 및 개발 → 테스트 → 회고의 과정을 반복하며 변화하는 요구사항에 유연하게 대응하고, 지속적으로 결과물을 통합하고 개선하는 전략을 채택했다.

## 2.1 문제해결을 위해 적용한 기법 및 결과

### 초기 접근 방식의 문제점

프로젝트 초기에는 Spring Boot 애플리케이션 내에 OpenAI API 호출 모듈을 직접 통합하

여 AI 기능을 구하였으며, 이 방식은 빠른 프로토타이핑에는 유리했으나, 다음과 같은 심각한 문제점에 직면하였다.

### 문제점 1: 프롬프트 관리의 복잡성 증가

- AI 기능(어휘 분석, 문장 요약, 이미지 생성 등)이 다양해지면서, Java 소스코드 내에 하드코딩된 프롬프트 문자열들이 산재했다.
- 프롬프트의 미세한 수정이 필요할 때마다 전체 애플리케이션의 코드를 수정하고 재배포해야 했으며, 이는 프롬프트의 버전 관리와 성능 테스트를 매우 어렵게 만들었다.

### 문제점 2: 복합적인 AI 작업 처리의 한계

- 'PDF 내용 기반 질의응답'과 같이 여러 단계의 AI 추론이 필요한 기능을 구현하기 위해 여러 개의 프롬프트를 수동으로 연결(Chaining)하는 로직을 Java로 작성해야했다.
- 이러한 체이닝 로직은 코드의 복잡도를 급격히 높였고, 유지보수 비용과 시간 또한 기하급수적으로 증가시켰다.

## 해결 방안: LangChain 도입 및 AI 서버 분리

이러한 문제를 해결하기 위해, 저희는 **LangChain 프레임워크를 도입하고 Python(FastAPI) 기반의 AI 서버를 별도로 구축하여** 기존 아키텍처를 변경하는 과감한 결정을 내렸습니다.

### 해결 과정:

- AI 로직 분리:** Spring Boot 애플리케이션에서는 비즈니스 로직만 담당하고, 모든 AI 관련 기능(프롬프트 관리, LLM 호출, 체이닝)은 새로운 AI 서버에 위임했다.
- LangChain 적용:** AI 서버에서는 LangChain을 사용하여 복잡한 프롬프트 체인을 표준화된 방식으로 손쉽게 구현했으며, LangChain의 LCEL(LangChain Expression Language)을 통해 여러 AI 모델과 데이터 소스를 파이프라인처럼 연결하여 코드의 가독성과 재사용성을 크게 향상시켰다.
- API 기반 통신:** Spring Boot 서버와 Python AI 서버는 명확하게 정의된 REST API를 통해 통신하도록 설계하여 두 서버 간의 의존성을 최소화했다.

결과적으로 이점은 다음과 같았다.

**프롬프트와 로직의 분리:** 프롬프트를 코드와 분리하여 관리할 수 있게 되었고, AI 기능의 수정 및 확장이 매우 용이해졌다.

- 개발 생산성 향상:** 복잡한 AI 작업을 위한 상용구 코드(Boilerplate) 작성이 줄어들어, 핵심 기능 개발에 더 집중할 수 있게 되었다.
- 성능 및 확장성 확보:** AI 연산이 많은 작업을 별도의 서버로 분리함으로써, 웹 애플리케이션 서버의 부하를 줄이고 전체 시스템의 안정성과 성능을 개선했다.

## 2.2 팀원의 책임 및 역할 수행

### 문제점: 기술 스택 변경으로 인한 역할의 모호성 발생

AI 서버를 분리하고 LangChain이라는 새로운 기술을 도입하면서 팀원 간의 역할 경계가 모호해지는 문제가 발생했다.

기존 백엔드 담당 팀원들은 Python 및 LangChain에 대한 학습 부담을 느꼈고, AI 서버를 담당하게 된 팀원은 기존 Spring Boot 시스템과의 연동 지점을 파악하는 데 어려움을 겪었으며, 이로 인해 초기에는 API 규격 정의를 두고 책임 소재가 불분명해지며 개발이 지연되는 현상이 두드러졌다.

- 해결 방안: API 우선 설계(API-First Design) 및 역할 재정의**
- 본격적인 개발에 앞서, Swagger(OpenAPI)를 사용하여 두 서버 간에 주고받을 API 명세부터 공동으로 확정했습니다.
- 이를 통해 각자의 책임 범위를 명확히 했습니다. Spring Boot 팀은 'API 호출 및 비즈니스 로직 처리'에, AI 팀은 'API 명세에 맞는 기능 구현'에만 집중할 수 있었다.
- 주 1회 코드 리뷰 세션을 통해 서로의 진행 상황을 공유하고 기술적인 장벽을 함께 해결하며 팀워크를 강화했다.

### 2.3 프로젝트 일정 계획

#### 문제점: 예기치 않은 아키텍처 변경으로 인한 일정 지연

프로젝트 초기 계획에는 없던 LangChain 도입 및 별도의 AI 서버를 구축하게 되면서, 팀원들이 새로운 기술에 적응하고 학습할 시간이 필요했다.

Python 기반의 AI 프레임워크와 서버 운영에 대한 기술적인 러닝커브를 예상보다 충분히 고려하지 못하여, 기존 코드를 수정하고 새로운 기능을 개발하는 과정에서 프로젝트 중간에 약 2주간의 일정 지연이 발생하였다.

#### 해결 방안: 집중적인 사전 학습을 통한 기술 내재화 및 일정 재산정

이러한 기술적 장벽을 극복하고 지연된 일정을 만회하기 위해, 저희 팀은 보다 근본적인 해결책이 필요하다고 판단하여, 본격적인 개발 재개에 앞서, 팀원들은 **방학 기간을 활용하여 새로운 기술에 대한 집중적인 사전 학습을 진행하게 되었다.**

먼저 LangChain의 공식 문서를 통해 기본적인 개념과 사용법을 익혔고, 심화 학습을 위해 'RAG 마스터'라는 전문 서적을 구매하여 함께 스터디를 진행하게 되었다.

이 과정을 통해 단순히 기능을 구현하는 것을 넘어, 기술의 내부 동작 원리와 효율적인 활용 방안에 대해 깊이 이해할 수 있었다.

이러한 사전 학습 덕분에 팀원들의 전반적인 기술 숙련도가 향상되었고, 이를 바탕으로 남은 개발 과업에 대한 소요 시간을 더욱 현실적으로 예측할 수 있게 되었으며, 이 예측을 기반으로 전체 프로젝트 일정을 다시 수립하였고, 이후로는 계획에 맞춰 안정적으로 프로젝트를 진행할 수 있었다.

[별첨1]

프로젝트명 : 리딩브릿지 - AI 기반 난독증 아동 맞춤형 학습 교재 자동 변환 플랫폼 개발

# 소프트웨어 요구사항 정의서

Version 1.0

개발 팀원 명(팀리더):최은기  
김지훈  
서동현

대표 연락처:010-2655-4276  
e-mail: deveungi@gmail.com

## 목차

1. 개요
2. 시스템 장비 구성요구사항
3. 기능 요구사항
4. 성능 요구사항
5. 인터페이스 요구사항
6. 데이터 요구사항
7. 테스트 요구사항
8. 보안 요구사항
9. 품질 요구사항
10. 제약 사항
11. 프로젝트 관리 요구사항

## 1. 요구사항 정의서에 사용되는 양식 설명

### 1-1. 사용자 관리 기능

ID	명칭	상세 내용	중요도
USER-001	보호자 회원가입	보호자는 아이디, 비밀번호, 이름, 이메일, 전화번호를 입력하여 회원가입을 요청한다.	상
USER-002	학생 회원가입	학생은 아이디, 비밀번호, 이름, 보호자 연동 코드, 관심사를 입력하여 회원가입을 요청한다. 시스템은 보호자 코드의 유효성을 확인하고, 유효할 경우 해당 보호자에게 학생 계정을 종속시켜 생성한다.	상
USER-003	로그인	사용자는 아이디와 비밀번호를 입력하여 로그인을 요청한다.	상
USER-004	보호자-학생 연동	보호자는 자신의 고유 연동 코드를 조회할 수 있다. 학생은 회원가입 시 이 코드를 사용하여 보호자와 계정을 연결한다.	중
USER-005	보호자의 학생 목록 조회	보호자는 자신에게 등록된 모든 학생의 목록을 조회한다.	중

### 1-2. 문서 및 교재 관리 기능

ID	명칭	상세 내용	중요도
DOC-001	PDF 문서 업로드	보호자는 학습에 사용할 PDF 파일을 시스템에 업로드하고, 파일은 서버의 지정된 위치에 저장되고, 문서 정보(ID, 제목, 파일 경로 등)가 데이터베이스에 기록된다. 비동기적으로 문서 변환 프로세스가 시작된다.	상
DOC-002	업로드된 문서 목록 조회	보호자는 자신이 업로드한 모든 문서의 목록과 각 문서의 변환 상태(처리 중, 완료, 실패)를 조회한다.	상
DOC-003	문서 변환 상태 실시간 확인	보호자는 문서 변환 작업의 현재 진행 상태(e.g., 텍스트 추출 중, 이미지 분석 중)를 실시간으로 확인한다. Redis에 저장된 현재 작업 상태 정보가 클라이언트로 전송된다.	상
DOC-004	변환 완료된 교재를 학생에게 할당	보호자는 변환이 완료된 교재를 선택하여 특정 학생 또는 여러 학생에게 학습하도록 할당한다.	중

### 1-3. 학습 콘텐츠 및 진행 관리 기능

ID	명칭	상세 내용	중요도
DOC-001	<b>PDF 문서 업로드</b>	보호자는 학습에 사용할 PDF 파일을 시스템에 업로드하고, 파일은 서버의 지정된 위치에 저장되고, 문서 정보(ID, 제목, 파일 경로 등)가 데이터베이스에 기록된다. 비동기적으로 문서 변환 프로세스가 시작된다.	상
DOC-002	<b>업로드된 문서 목록 조회</b>	보호자는 자신이 업로드한 모든 문서의 목록과 각 문서의 변환 상태(처리 중, 완료, 실패)를 조회한다.	상
DOC-003	<b>문서 변환 상태 실시간 확인</b>	보호자는 문서 변환 작업의 현재 진행 상태(e.g., 텍스트 추출 중, 이미지 분석 중)를 실시간으로 확인한다. Redis에 저장된 현재 작업 상태 정보가 클라이언트로 전송된다.	상
DOC-004	<b>변환 완료된 교재를 학생에게 할당</b>	보호자는 변환이 완료된 교재를 선택하여 특정 학생 또는 여러 학생에게 학습하도록 할당한다.	중

## 1. 시스템 개요

다음은 해당 프로젝트의 PDF 교안 변환 및 처리 환경에 대해 Spring Boot 애플리케이션 내부 모듈을 활용한 처리 구성 예시이다.

현시점에서 AI 처리 성능 최적화와 운영 비용을 최소화하기 위해 Spring Boot 애플리케이션 내부에서 멀티스레드 처리 방식으로 구성하였으며, 각 처리 모듈은 PDF 파싱, AI 변환, 데이터 처리 운용 목적으로 활용된다.

PDF 파싱 모듈 - 업로드된 PDF 파일의 텍스트, 이미지, 표 등 구조적 요소 추출 및 메타데이터 분석을 통한 콘텐츠 기본 구조 파악 지원

AI 변환 모듈 - LangChain 기반 RAG 파이프라인을 통해 난독증 친화적 콘텐츠 자동 변환 및 문장 길이 최적화, 어휘 난이도 조정 처리 지원

데이터 처리 모듈 - PGVector 벡터 데이터베이스, ElastiCache 캐싱, 변환 결과 저장 및 실시간 사용자 피드백 수집을 위한 데이터 파이프라인 운영 지원

이러한 모듈 기반 처리 아키텍처를 통해 대용량 PDF 교안의 효율적 처리와 함께 난독증 아동의 인지적 특성에 맞춘 고품질 맞춤형 학습 콘텐츠 생성을 실현한다.

## 기술 구조 PDF 기반 교안 생성 프로세스

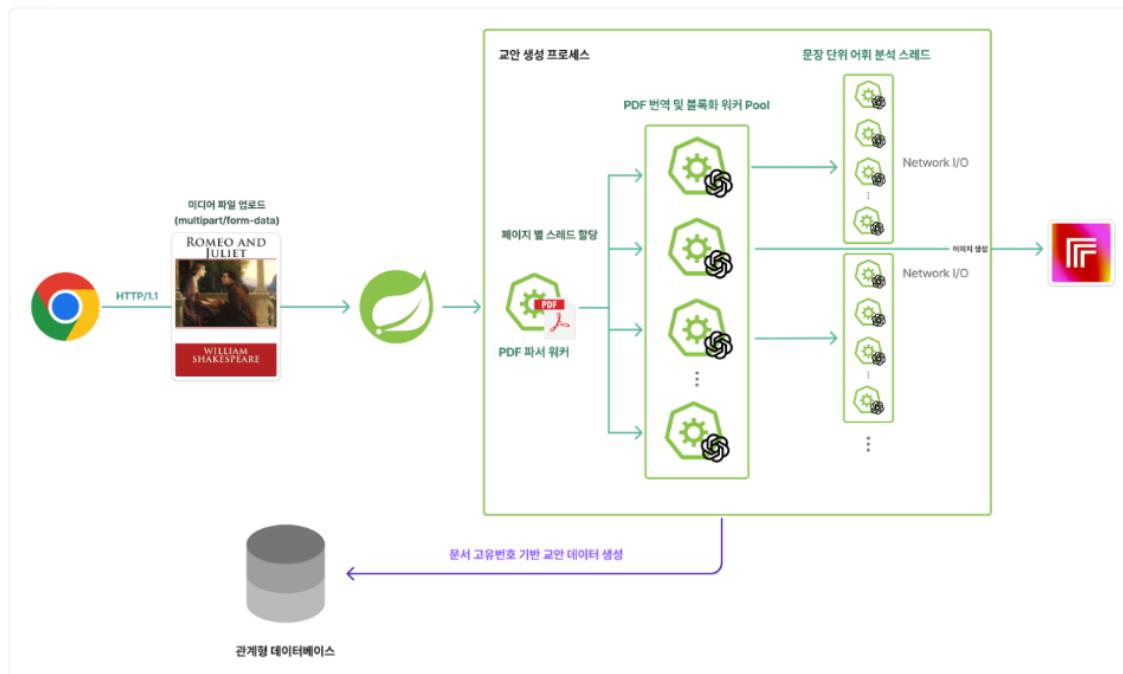


그림 7 난독 생성교안 생성에 대한 개요

## 2. 시스템 장비 구성요구사항

요구사항 고유번호		CLAWS-001		
요구사항 명칭		H/W: 애플리케이션 서버		
요구사항 분류		시스템 장비구성 요구사항	응락수준	필수
요구사항 상세설명	정의	<b>시스템 통합 운영 서버</b> - 웹/WAS, AI, DB 등 모든 서비스의 기반이 되는 가상 서버 역할을 수행		
	세부 내용	<b>CPU:</b> 8 vCPU 이상 <b>Memory:</b> 32GB 이상 <b>Disk:</b> 100 GB 이상 (NVMe SSD) (AWS EC2 t3.2xlarge 또는 동급 사양)		
요구사항 고유번호		CLAWS-002		
요구사항 명칭		S/W: 운영체제		
요구사항 분류		시스템 장비구성 요구사항	응락수준	필수
요구사항 상세설명	정의	<b>서버 운영 및 관리</b> 모든 소프트웨어 스택이 설치되고 실행되는 기본 환경을 제공		
	세부 내용	Ubuntu 22.04 LTS (64bit)		
요구사항 고유번호		CLAWS-003		
요구사항 명칭		S/W: 웹 서버		
요구사항 분류		시스템 장비구성 요구사항	응락수준	필수
요구사항 상세설명	정의	<b>요청 분배 및 SSL 처리</b> 사용자 요청을 수신하여 내부 애플리케이션으로 전달하는 리버스 프록시 및 정적 파일 서빙 기능을 담당		
	세부 내용	Nginx 1.18.x 이상		

요구사항 고유번호		CLAWS-004		
요구사항 명칭		S/W: 웹 애플리케이션 서버(WAS)		
요구사항 분류		시스템 장비구성 요구사항	응락수준	필수
요구사항 상세설명	정의	<b>핵심 비즈니스 로직 처리</b> 회원 관리, 학습 관리 등 시스템의 주요 비즈니스 로직을 실행		
	세부 내용	Spring Boot 3.x (Java 17 기반)		

요구사항 고유번호		CLAWS-005		
요구사항 명칭		S/W: AI 서버		
요구사항 분류		시스템 장비구성 요구사항	응락수준	필수
요구사항 상세설명	정의	AI 연산 및 추론 PDF 문서 처리, LangChain 기반의 AI 모델 호출 등 자원 소모가 많은 AI 관련 작업을 담당		
	세부 내용	Python 3.10+, FastAPI		

요구사항 고유번호		CLAWS-006		
요구사항 명칭		S/W: 데이터베이스		
요구사항 분류		시스템 장비구성 요구사항	응락수준	필수
요구사항 상세설명	정의	데이터 저장 및 관리 사용자 정보, 학습 데이터 등 모든 정형 데이터를 영구적으로 저장 및 관리		
	세부 내용	PostgreSQL 8.0 이상		

요구사항 고유번호		CLAWS-007		
요구사항 명칭		S/W: 인메모리 캐시		
요구사항 분류		시스템 장비구성 요구사항	응락수준	필수
요구사항 상세설명	정의	성능 향상 및 상태 관리 세션 정보, PDF 변환 상태 등 실시간 데이터 처리 및 캐싱을 통해 시스템 응답 속도 향상		
	세부 내용	Redis 7.0 이상		

요구사항 고유번호		CLAWS-008		
요구사항 명칭		S/W: 배포/런타임		
요구사항 분류		시스템 장비구성 요구사항	응락수준	필수

요구사항 상세설명	정의	서비스 컨테이너화 및 관리 각 소프트웨어 구성 요소를 격리된 컨테이너 환경에서 실행하고, 이를 통해 배포 및 관리 용이
	세부 내용	Docker, Docker Compose

### 3. 기능 요구사항

요구사항 고유번호	SFR-RB-001		
요구사항 명칭	사용자 인증 및 회원가입		
요구사항 분류	기능	응락수준	필수
요구사항 상세설명	정의	보호자/학생 구분 기반 회원가입 및 카카오톡 OAuth2 로그인 기능 제공	
	세부 내용	<ul style="list-style-type: none"> <li>- 카카오톡 OAuth2 기반 소셜 로그인</li> <li>- 역할별(보호자/학생) 회원가입 플로우</li> <li>- JWT 기반 인증 토큰 발급 및 재발급</li> <li>- 사용자 인증 상태 관리 (Refresh 토큰 및 만료처리 포함)</li> <li>- 학생 초대 링크 생성/연결 기능</li> </ul>	

요구사항 고유번호	SFR-RB-002		
요구사항 명칭	PDF 교안 업로드 및 AI 변환 기능		
요구사항 분류	기능	응락수준	필수
요구사항 상세설명	정의	PDF 교안을 난독증 친화적 디지털 콘텐츠로 자동 변환하는 기능 제공	
	세부 내용	<ul style="list-style-type: none"> <li>- PDF 파일 업로드 및 텍스트/이미지 추출</li> <li>- LangChain 기반 RAG 파이프라인 처리</li> <li>- 문장 길이 최적화 (한국어 기준 15단어 내외)</li> <li>- 난독증 친화 폰트 및 시각적 요소 적용</li> <li>- 변환 진행률 실시간 모니터링 및 완료 알림</li> </ul>	

<b>요구사항 고유번호</b>		SFR-RB-003		
<b>요구사항 명칭</b>		맞춤형 읽기 환경 제공		
<b>요구사항 분류</b>		기능	<b>응락수준</b>	필수
<b>요구사항 상세설명</b>	정의	난독증 학생을 위한 개인화된 읽기 환경 설정 기능		
	세부 내용	<ul style="list-style-type: none"> <li>- 글자 크기, 줄 간격, 자간 자유 조절</li> <li>- 다양한 배경색 옵션 (밝은, 세피아, 연한 파랑, 어두운 테마)</li> <li>- TTS(Text-to-Speech) 음성 읽기 지원</li> <li>- 어휘 하이라이트 및 단어 클릭 시 발음 재생</li> <li>- 실시간 읽기 진행률 표시</li> </ul>		

<b>요구사항 고유번호</b>		SFR-RB-004		
<b>요구사항 명칭</b>		교안 스토어 및 관리 기능		
<b>요구사항 분류</b>		기능	<b>응락수준</b>	필수
<b>요구사항 상세설명</b>	정의	전문가 제작 교안 구매/대여 및 보유 교안 관리 기능		
	세부 내용	<ul style="list-style-type: none"> <li>- 학년별, 과목별, 가격별 필터 검색</li> <li>- 교안 평점 및 리뷰 시스템</li> <li>- 구매/대여/무료 다운로드 기능</li> <li>- 보유 교안 목록 관리 및 학생별 할당</li> <li>- 교안 사용 통계 및 진행률 추적</li> </ul>		

<b>요구사항 고유번호</b>		SFR-RB-005		
<b>요구사항 명칭</b>		실시간 학습 모니터링		
<b>요구사항 분류</b>		기능	<b>응락수준</b>	필수
<b>요구사항 상세설명</b>	정의	학생의 학습 활동을 실시간으로 추적하고 분석하는 기능		
	세부 내용	<ul style="list-style-type: none"> <li>- 학습 진도, 이해도, 집중도 실시간 추적</li> <li>- 보호자 대시보드를 통한 학습 현황 시각화</li> <li>- 주간/월간 학습 통계 및 성취 현황 제공</li> <li>- 도움이 필요한 순간 자동 감지 및 알림</li> <li>- 학습 패턴 분석 및 개선 제안</li> </ul>		

요구사항 고유번호		SFR-RB-006		
요구사항 명칭		동기부여 및 성취 관리 시스템		
요구사항 분류	기능	응락수준	필수	
요구사항 상세설명	정의	학습 동기 증진을 위한 칭찬 및 성취 시스템 기능		
	세부 내용	<ul style="list-style-type: none"> <li>- 보호자가 제공하는 칭찬 스티커 시스템</li> <li>- 학습 성취에 따른 뱃지 및 보상 제공</li> <li>- 완료한 페이지 수, 총 학습 시간 통계</li> <li>- 주간 성취 현황 대시보드</li> <li>- 음운 분해 및 쓰기 연습 결과 추적</li> </ul>		

요구사항 고유번호		SFR-RB-007		
요구사항 명칭		알림 및 시스템 관리 기능		
요구사항 분류	기능	응락수준	필수	
요구사항 상세설명	정의	시스템 알림 및 관리자 기능 제공		
	세부 내용	<ul style="list-style-type: none"> <li>- 학습 진행 상황 알림 (칭찬 메시지 수신 등)</li> <li>- 교안 변환 완료 알림</li> <li>- 시스템 사용량 및 성능 모니터링</li> <li>- 사용자 활동 로그 분석</li> <li>- 서비스 상태 점검 및 오류 알림</li> </ul>		

#### 4. 성능 요구사항

<b>요구사항 고유번호</b>	PER-001				
<b>요구사항 명칭</b>	AI 교안 변환 처리 시간				
<b>요구사항 분류</b>	성능	응락수준	필수		
<b>요구사항 상세설명</b>	정의	데이터 전처리 및 AI 응답 생성			
	세부 내용	<ul style="list-style-type: none"> <li>- 10페이지 분량의 텍스트 위주의 표준 PDF 파일을 업로드 완료 후 60초 이내에 변환이 완료되어야 한다.</li> <li>- 이미지가 다수 포함되어있거나 레이아웃이 복잡한 10페이지 분량의 PDF 파일을 업로드 완료 후 90초 이내에 변환이 완료되어야 한다.</li> </ul>			
<b>요구사항 고유번호</b>	PER-002				
<b>요구사항 명칭</b>	AI 교안 변환 성공률				
<b>요구사항 분류</b>	성능	응락수준	필수		
<b>요구사항 상세설명</b>	정의	데이터 전처리 및 AI 응답 생성			
	세부 내용	<ul style="list-style-type: none"> <li>- 표준 PDF 파일에 대해 95% 이상의 변환 성공률을 보장해야 한다.</li> <li>- 변환 실패 시, 시스템은 원인을 분석하여 사용자에게 명확한 실패 사유를 안내하고 재시도를 유도하거나 고객 지원팀에 문의하도록 안내해야 한다.</li> </ul>			
<b>요구사항 고유번호</b>	PER-003				
<b>요구사항 명칭</b>	동시 사용자 처리 용량				
<b>요구사항 분류</b>	성능	응락수준	필수		
<b>요구사항 상세설명</b>	정의	클라이언트 뷰어 및 교안 생성			
	세부 내용	<ul style="list-style-type: none"> <li>- 최소 1,000명의 동시 접속 사용자를 지원 없이 처리할 수 있어야 한다.</li> <li>- 실시간 트래픽을 고려하여 20%의 사용자는 실시간 학습 모니터링 기능을 사용 중인 상태로 가정하며, 이로 인해 발생하는 대량의 WebSocket 연결 부하를 안정적으로 감당할 수 있어야 한다.</li> </ul>			

요구사항 고유번호		PER-004		
요구사항 명칭		클라이언트 렌더링 성능		
요구사항 분류		성능	응락수준	필수
요구사항 상세설명	정의	클라이언트 뷰어의 화면 로딩 및 전환		
	세부 내용	<ul style="list-style-type: none"> <li>- 학생용 교안 뷰어의 첫 페이지 로딩은 일반적인 네트워크 환경(LTE)에서 2초 이내에 완료되어야 한다.</li> <li>- 교안 내에서 다음 페이지로 넘어가는 전환은 500ms 이내에 시각적으로 완료되어야 한다.</li> </ul> <p>이 요구사항은 단순한 기술적 목표를 넘어선다. 1에서 언급된 바와 같이, 난독증 아동은 '작업 기억'의 제한과 '시각적 처리'의 부담을 겪는다. 느린 페이지 로딩이나 버벅이는 화면 전환은 학생의 집중력을 분산시키고, 읽기 과정에 불필요한 인지적 스트레스를 가중시킨다. 이는 학습에 대한 좌절감과 거부감으로 이어져 앱의 핵심 가치를 훼손할 수 있으므로, 매우 엄격한 성능 목표를 설정하고 클라이언트 측 렌더링 최적화 및 데이터 사전 로딩(Pre-loading) 기술을 적용하여 이를 달성해야 한다.</p>		

## 5. 인터페이스 요구사항

요구사항 고유번호		INT-001		
요구사항 명칭		사용자가 시스템을 사용하기 위한 웹 기반의 그래픽 사용자 인터페이스 (GUI)		
요구사항 분류		인터페이스	응락수준	필수
요구사항 세부내용	정의	Spring Boot WAS와 Python AI 서버 간의 데이터 송수신 인터페이스		
	세부내용	<ul style="list-style-type: none"> <li>- 시스템의 모든 기능은 웹 브라우저를 통해 시각적으로 제공되어야 한다.</li> <li>- 주요 웹 브라우저(Chrome, Safari 등)의 최신 버전과 호환되어야 한다.</li> </ul>		
기타 고려 사항		-		
산출정보		HTML, CSS, JS 기반의 웹 페이지		
요구사항 출처		UI/UX 설계 프로토타입		
요구사항 고유번호		INT-002		
요구사항 명칭		내부 시스템 인터페이스 (WAS ↔ AI 서버)		
요구사항 분류		인터페이스	응락수준	필수
요구사항 세부내용	정의	Spring Boot WAS와 Python AI 서버 간의 데이터 송수신 인터페이스		
	세부내용	<ul style="list-style-type: none"> <li>- 통신은 RESTful API(JSON 형식)를 기반으로 한다.</li> <li>- API 명세는 Swagger(OpenAPI 3.0)를 통해 명확하게 정의 및 관리되어야 한다.</li> </ul>		
기타 고려 사항		-		
산출정보		Swagger(OpenAPI) API 문서		
요구사항 출처		시스템 아키텍처 설계서		

요구사항 고유번호		INT-003		
요구사항 명칭		외부 AI API 연동 인터페이스		
요구사항 분류		인터페이스	응락수준	필수
요구사항 세부내용	정의	외부 AI 서비스(OpenAI, DeepL 등)와 연동하기 위한 인터페이스		
	세부내용	<ul style="list-style-type: none"> <li>- 각 서비스 제공사의 공식 API 규격을 준수하여 HTTPS로 통신한다.</li> <li>- API Key 등 인증 정보는 안전하게 관리되어야 한다.</li> <li>- 외부 API의 응답 지연 및 오류에 대한 예외 처리가 포함되어야 한다.</li> </ul>		
기타 고려 사항		-		
산출정보		Swagger(OpenAPI) API 문서		
요구사항 출처		기능 요구사항 정의서		

## 6. 데이터 요구사항

요구사항 고유번호	DAR-001		
요구사항 명칭	데이터 식별 및 유형 정의		
요구사항 분류	데이터	응락수준	필수
요구사항 세부내용	<p>리딩브릿지 서비스에서 사용되는 주요 데이터는 다음과 같이 정의됨</p> <ul style="list-style-type: none"> <li>- 사용자 데이터 (이메일, 닉네임, 가입일, 역할, 인증정보 등)</li> <li>- 교안 데이터 (원본 PDF 파일, 생성된 교안)</li> <li>- 학습 데이터(페이지별 체류 시간, TTS 사용 횟수 및 구간 등)</li> <li>- 상호작용 데이터(학생의 쓰기 연습 결과물)</li> </ul>		
요구사항 고유번호	DAR-002		
요구사항 명칭	초기 자료 구축 및 테스트 데이터 준비		
요구사항 분류	데이터	응락수준	필수
요구사항 세부내용	<p>개발 및 운영 전, 아래와 같은 목적을 위한 초기 데이터 구축이 필 요함</p> <ul style="list-style-type: none"> <li>- 테스트용 계정 더미 데이터 생성</li> <li>- 다양한 유형의 샘플 교안 데이터 생성</li> <li>- 관리자/운영자용 테스트 로그 구축</li> <li>- 1,000건 이상의 대규모 모의 학습 로그 데이터 생성</li> </ul>		
요구사항 고유번호	DAR-003		
요구사항 명칭	데이터 품질 관리 및 보존 정책		
요구사항 분류	데이터	응락수준	필수
요구사항 세부내용	<p>주요 데이터는 아래 기준에 따라 품질 관리 및 백업 정책을 마련해야 함</p> <ul style="list-style-type: none"> <li>- 사용자 계정 정보: 회원 탈퇴 요청 시 지체 없이 파기</li> <li>- 학습 로그 데이터: 개인식별 정보를 제거한 후 3년간 보관</li> <li>- 시스템 설정값 및 운영정보는 1년 이상 보관</li> <li>- AI 교안 성생 결과는 생성모델 개선을 위한 학습 데이터로 분리 저장</li> </ul>		

## 7. 테스트 요구사항

요구사항 고유번호	TER-001		
요구사항 명칭	테스트 방안 및 절차		
요구사항 분류	테스트	응락수준	필수
요구사항 세부내용	<p>시스템은 다음의 순서로 테스트를 진행하여 전체 기능의 정상 동작과 성능 적합 여부를 검증해야 한다.</p> <p>① 단위 테스트 (Unit Test):</p> <ul style="list-style-type: none"> <li>- Spring, FastAPI 기반 백엔드 서비스 단위 테스트</li> <li>- JUnit, Pytest 활용하여 REST API 및 핵심 로직 정상 작동 확인</li> </ul> <p>② 통합 테스트 (Integration Test):</p> <ul style="list-style-type: none"> <li>- 프론트엔드(Next.js)와 백엔드(Spring, FastAPI) 간 API 통신 테스트</li> <li>- 데이터베이스(PostgreSQL) 연동 상태 확인</li> <li>- Redis 캐시 확인</li> </ul> <p>③ 부하 테스트 (Stress / Load Test):</p> <ul style="list-style-type: none"> <li>- 평균 동시 접속자 수 250명을 기준으로 JMeter를 통해 시나리오 기반 테스트 수행</li> <li>- 로그인, PDF 업로드, 교안 생성, 학습 세션 등 주요 흐름 시뮬레이션</li> <li>- 서버 응답 시간, 에러율, CPU/RAM 사용량 모니터링 포함</li> </ul> <p>모든 테스트는 CI 환경(GitHub Actions)과 병행하여 반복 수행 가능하도록 스크립트화 되어야 함</p>		

## 8. 보안 요구사항

요구사항 고유번호	SER-001		
요구사항 명칭	보안지침 준수		
요구사항 분류	보안	응락수준	필수
요구사항 세부내용	<p>본 시스템은 KISA, 행정안전부, OWASP 등 보안 가이드라인에 따라 개발 · 배포 · 운영되어야 함</p> <p>로그인/인증/세션 처리/에러 메시지 등에 보안 기준을 적용하고, 주요 개인정보 및 로그는 저장 전 암호화/마스킹 처리</p> <p>보안 요구사항은 개발 초기부터 반영되며, GitHub Actions 통한 배포 과정에서도 정적 코드 보안 검사(예: StyleGuide, BiomeJS) 등을 수행해야 함</p>		

요구사항 고유번호	SER-002		
요구사항 명칭	인증 및 계정 보안		
요구사항 분류	보안	응락수준	필수
요구사항 세부내용	<p>사용자 인증은 JWT 기반 인증 체계를 사용함</p> <p>Access Token은 만료 주기를 짧게(예: 15분), Refresh Token은 HttpOnly + Secure 쿠키에 저장하여 탈취 위험 최소화</p> <p>Refresh Token 탈취 시 재사용 방지를 위해 서버 저장소에 토큰 재발행 이력 및 폐기 처리 구조 구현</p> <p>계정 정보는 해시(SHA256 이상)로 암호화하며, 탈퇴 시 논리 삭제 정책을 따른다</p>		

요구사항 고유번호	SER-003		
요구사항 명칭	애플리케이션 계층 보안		
요구사항 분류	보안	응락수준	필수
요구사항 세부내용	<p>모든 API 요청은 서버와 클라이언트 양쪽에서 유효성 검증을 거치며, 허용된 입력 외의 데이터는 즉시 차단</p> <p>JPA를 사용한 ORM 기반 쿼리 생성을 통해 SQL Injection 위험 제거</p> <p>입력값의 길이 제한, 허용된 문자 이외 차단, 잘못된 요청에 대한 403/422 응답 분기 처리 포함</p> <p>클라이언트 단에서 전송되는 모든 요청은 CSRF 위험 방지를 위해</p>		

	동일 도메인 내 쿠키 설정 정책 적용
--	----------------------

## 9. 품질 요구사항

요구사항 고유번호	QUR-001		
요구사항 명칭	시스템 신뢰성 및 고장 복구 시간		
요구사항 분류	품질	응락수준	필수
요구사항 세부내용	서비스는 지정된 조건(250명 동시접속)에서 95% 이상의 가용률을 보장해야 함  주요 서비스에 장애 발생 시, 모니터링 시스템이 이를 자동으로 감지하고 자동 복구 기능을 통해 5분 이내에 서비스가 자동으로 재시작되어 정상 상태로 복구되어야 함  치명적 장애가 발생한 경우, 전체 서비스 중단 시간은 월 3시간 이내로 제한		
요구사항 고유번호	QUR-002		
요구사항 명칭	서비스 사용성		
요구사항 분류	품질	응락수준	필수
요구사항 세부내용	IT 기기 사용에 익숙하지 않은 사용자도 별도의 메뉴얼이나 교육 없이 10분 이내에 회원가입, PDF 교안 업로드 등 핵심 기능을 스스로 완료할 수 있어야 한다. 이를 위해 직관적인 UI와 명확한 안내 메시지를 제공해야 함  학생의 학습 몰입을 방해하지 않도록 최소한의 UI를 제공해야 하며, 학습 중에는 불필요한 UI 요소를 숨겨 콘텐츠에만 집중할 수 있는 몰입 모드를 제공해야 함		

요구사항 고유번호	QUR-003		
요구사항 명칭	서비스 확장성		
요구사항 분류	품질	응락수준	필수
요구사항 세부내용	<p>시스템은 마이크로서비스 아키텍처를 기반으로, 특정 기능(AI변환, 실시간 모니터링)에 부하가 집중될 경우 해당 서비스만 독립적으로 인스턴스 수를 늘려 확장할 수 있는 구조여야 한다.</p>		

요구사항 고유번호	QUR-004		
요구사항 명칭	유지보수성		
요구사항 분류	품질	응락수준	필수
요구사항 세부내용	<p>코드는 기능별, 책임별로 명확하게 모듈화되어 작성되어야 하며, 새로운 기능을 추가하거나 기존 기능을 변경할 때 다른 부분에 미치는 영향을 최소화해야 한다.</p> <p>모든 주요 함수와 클래스에는 그 목적과 동작 방식을 설명하는 적절한 주석과 문서가 포함되어야 하며, API 명세는 항상 최신 상태로 유지되어야 한다.</p>		

## 10. 제약 사항

요구사항 고유번호	COR-001		
요구사항 명칭	개발 언어 및 프레임워크 지정		
요구사항 분류	시스템 개발 제약사항	응락수준	필수
요구사항 세부내용	<ul style="list-style-type: none"> <li>- 백엔드(WAS)는 Java 17 및 Spring Boot 3.x를 기반으로 개발한다.</li> <li>- AI 서버는 Python 3.10 이상 및 FastAPI, LangChain 프레임워크를 사용하여 개발한다.</li> </ul>		

요구사항 고유번호	COR-002		
요구사항 명칭	단일 서버 아키텍처		
요구사항 분류	설계 및 구현 제약사항	응락수준	필수
요구사항 세부내용	<ul style="list-style-type: none"> <li>- 시스템은 AWS EC2 단일 인스턴스 내에서 웹/WAS, AI, DB 등 모든 서비스가 함께 운영되는 통합 배포(Monolithic Deployment) 구조를 따라야 한다.</li> </ul>		

요구사항 고유번호	COR-003		
요구사항 명칭	API 문서 관리		
요구사항 분류	설계 및 구현 제약사항	응락수준	필수
요구사항 세부내용	<ul style="list-style-type: none"> <li>- 서버 간 통신을 위한 모든 API는 Swagger(OpenAPI 3.0)를 사용하여 명세를 정의하고, 이를 기준으로 개발 및 테스트를 진행해야 한다.</li> </ul>		

요구사항 고유번호	COR-004		
요구사항 명칭	컨테이너 기반 배포		
요구사항 분류	설계 및 구현 제약사항	응락수준	필수
요구사항 세부내용	<ul style="list-style-type: none"> <li>- 모든 애플리케이션은 Docker 컨테이너 환경에서 실행되어야 하며, Docker Compose를 통해 서비스 환경을 일관성 있게 관리해야 한다.</li> </ul>		

요구사항 고유번호	COR-005		
요구사항 명칭	사용자 접근성 준수		
요구사항 분류	업무 제약사항	응락수준	필수
요구사항 세부내용	<p>- 주 사용자가 난독증 아동인 점을 고려하여, 모든 UI는 <b>높은 가독성, 직관적인 조작, 간결한 레이아웃</b>을 최우선으로 설계 및 구현해야 한다.</p>		

요구사항 고유번호	COR-006		
요구사항 명칭	개인정보보호 규정 준수		
요구사항 분류	표준 제약사항	응락수준	필수
요구사항 세부내용	<p>- 학생 및 보호자의 개인정보를 수집하고 처리하는 과정은 <b>개인정보 보호법</b> 등 관련 법규를 반드시 준수해야 하며, 데이터베이스에 저장되는 민감 정보는 암호화하여 관리해야 한다.</p>		

## 11. 프로젝트 관리 요구사항

요구사항 고유번호	PMR-001		
요구사항 명칭	프로젝트 수행 조직 구성 및 역할 정의		
요구사항 분류	프로젝트 관리	응락수준	필수
요구사항 세부내용	<ul style="list-style-type: none"> <li>- <b>프로젝트 리더(PL)</b>: 프로젝트 총괄, 일정 및 리스크 관리, 최종 의사 결정</li> <li>- <b>백엔드 개발자</b>: Spring Boot 기반 서버 개발, DB 설계 및 API 구현</li> <li>- <b>AI 개발자</b>: Python 기반 AI 서버 개발, LangChain 모델링 및 프롬프트 엔지니어링</li> <li>- <b>프론트엔드 개발자</b>: 사용자 인터페이스(UI) 개발 및 서버 연동</li> </ul>		
요구사항 고유번호	PMR-002		
요구사항 명칭	프로젝트 일정 및 과업 관리		
요구사항 분류	프로젝트 관리	응락수준	필수
요구사항 세부내용	<ul style="list-style-type: none"> <li>- 모든 개발 과업은 <b>Jira</b>를 사용하여 작업 분할 구조(WBS)에 따라 에픽(Epic), 스토리(Story), 태스크(Task) 단위로 등록하고 추적한다.</li> <li>- 2주 단위의 스프린트(Sprint)를 기반으로 개발을 진행하며, 각 스프린트마다 계획, 리뷰, 회고를 통해 진행 상황을 점검하고 개선한다.</li> </ul>		
요구사항 고유번호	PMR-003		
요구사항 명칭	프로젝트 산출물 및 소통 관리		
요구사항 분류	프로젝트 관리	응락수준	필수
요구사항 세부내용	<ul style="list-style-type: none"> <li>- 모든 프로젝트 관련 문서(요구사항 정의서, 설계서, 회의록, 테스트 케이스 등)는 <b>Notion</b>을 통해 체계적으로 작성하고 버전 관리한다.</li> <li>- 팀 내 주요 공지사항 및 기술 공유 자료는 Notion을 통해 투명하게 공유하여 원활한 소통과 지식 공유를 촉진한다.</li> </ul>		
요구사항 고유번호	PMR-004		
요구사항 명칭	형상 관리		
요구사항 분류	프로젝트 관리	응락수준	필수
요구사항 세부내용	<ul style="list-style-type: none"> <li>- 모든 소스코드는 <b>Git</b>을 사용하여 버전 관리하며, <b>GitHub</b>를 중앙 원격 저장소로 사용한다.</li> <li>- 기능 개발, 버그 수정 등 모든 코드 변경은 별도의 브랜치에서 작업 후 코드 리뷰를 거쳐 메인 브랜치에 병합하는 것을 원칙으로 한다.</li> </ul>		