

한성대학교

컴퓨터공학부

2024- 2 SW프리캡스톤디자인

최종보고서

EduCraft

팀원: 유현서, 박진영

담당 교수: 김성동

<1>프로젝트 계획서 - 9팀

1. 프로젝트 개요

1.1 프로젝트 이름

- EduCraft

1.2 문제정의문

- EduCraft는 학습자가 능동적으로 학습할 수 있도록, LLM을 활용해 객관식 및 주관식 문제를 자동 생성하고 즉각적인 피드백을 제공하는 학습 보조 시스템입니다.

1.3 프로젝트 배경

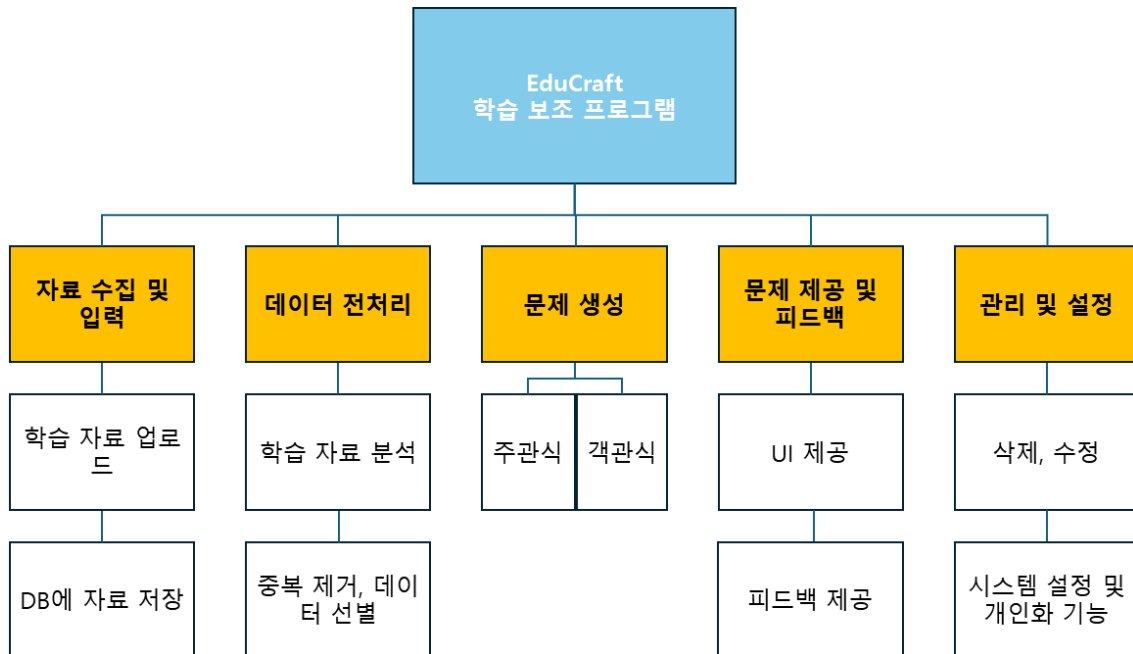
- 현재 학생들은 스스로 학습 내용을 점검하거나 문제를 만들 수 있는 자원이 부족하여, 자기주도적인 학습이 어렵습니다.

- 특히 비대면 학습 상황에서는 효과적인 피드백과 문제 생성 도구가 없어서 학습자가 학습 성과를 적시에 확인하거나 능동적으로 학습을 이어가는 데 한계가 있습니다.

- 이를 해결하기 위해, LLM 기반의 문제 생성 및 피드백 시스템을 도입하여 학생들이 학습 내용을 쉽게 점검하고, 주도적으로 학습을 진행할 수 있도록 돕는 것이 목표입니다.

2. 기능 구조도 (Functional Decomposition Diagram)

2.1 기능 구조도



2.2 기능 설명

2.2.1 자료 수집 및 입력

📁 사용자가 학습 자료를 업로드하거나 URL로 제공할 수 있는 기능

📁 DB에 자료를 저장하는 기능

2.2.2 데이터 전처리

📁 학습 자료를 분석하고 텍스트 정제 및 구조화하는 기능

📁 중복 자료를 제거하고 유효한 데이터를 선별하는 기능

2.2.3 문제 생성

📁 LLM을 이용하여 학습 자료로부터 문제를 생성하는 기능(객관식, 주관식)

2.2.4 문제 제공 및 피드백

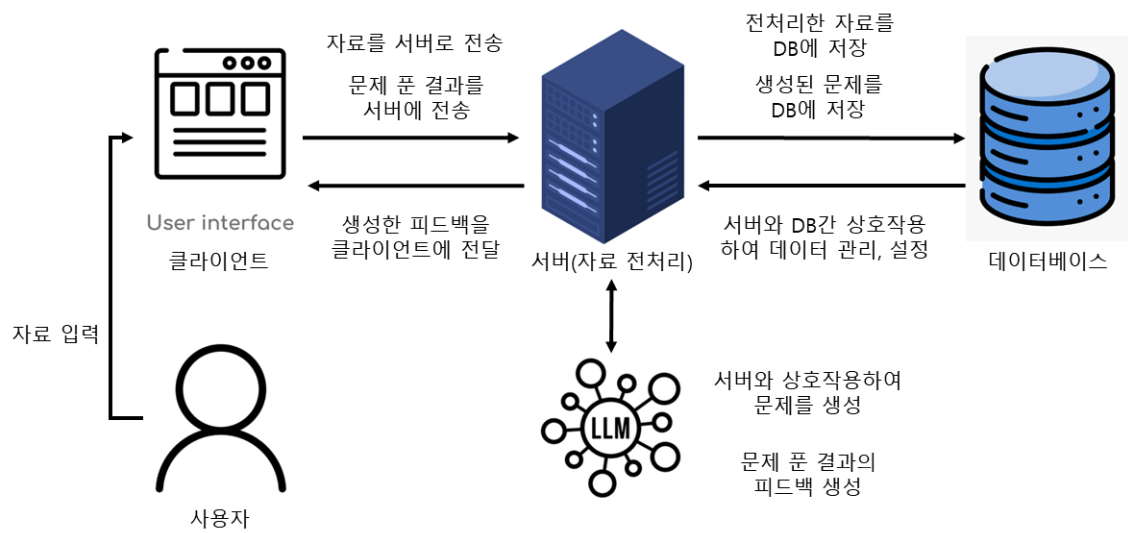
- ☞ 사용자가 문제를 풀 수 있는 UI 제공
- ☞ 사용자가 문제 풀이 결과에 따라 피드백 제공
- ☞ 피드백을 바탕으로 문제의 난이도를 조정하거나 새로운 문제 생성

2.2.5 관리 및 설정

- ☞ 학습 자료와 문제를 관리하는 기능 (삭제, 수정 등)
- ☞ 학습자의 성과 분석 기능
- ☞ 시스템 설정 및 개인화 기능

3. 시스템 구조도

3.1 시스템 구조도



3.2 시스템 구성 요소 설명

구성요소	설명	개발 환경	구현 방법
클라이언트	웹 애플리케이션으로 구현되며, 서버와 통신하여 데이터를 주고 받습니다. 문제 풀이 및 결과를 시각적으로 제공하는 것이 목표입니다.	HTML, CSS, JavaScript (React.js 혹은 Vue.js 와 같은 프레임워크)	클라이언트는 REST API를 통해 서버와 통신하며, 학습 자료 입력, 문제 풀이 인터페이스, 피드백 화면을 제공합니다.
서버	클라이언트의 요청을 받아 자료를 처리하고, 문제 생성, 데이터 전처리 및 피드백 제공 등의 핵심 기능을 수행합니다.	Node.js (Express.js 프레임워크) 또는 Python (Flask/Django)	서버는 클라이언트에서 전송된 데이터를 처리하고, 데이터베이스와 LLM으로 요청을 전달하며, 결과를 다시 클라이언트에 반환합니다.
데이터베이스	문제 생성에 필요한 자료를 저장하고, 생성된 문제와 사용자 응답 데이터를 기록합니다.	MySQL 또는 PostgreSQL	서버와 연결된 데이터베이스는 학습 자료, 문제, 사용자 정보를 저장합니다.
LLM	학습 자료로부터 객관식 및 주관식 문제를 생성하고, 사용자 응답에 대해 분석하고 피드백을 제공합니다.	Okestro API를 사용	서버는 LLM에 학습 자료를 전송해 문제를 생성하거나 사용자 응답에 대한 피드백을 요청합니다.

4. 일정 계획

주차	마일스톤/작업	설명
5주차	프로젝트 계획 수립 및 역할 분담	프로젝트 개요 작성, 기능 구조 및 요구 사항 정의, 기술 스택 선택, 시스템 구조도 설계
6주차	자료 수집 및 데이터 전처리 설계	데이터 수집 관련 모듈 설계(인터페이스 포함), 데이터 전처리 방법론 및 코드 설계, 데이터 베이스 구조 설계 및 구축, 데이터 입력 및 검증 기능 개발
7~9주차	문제 생성 알고리즘 개발	LLM과의 연동 부분 개발, 객관식/주관식 문제 생성 알고리즘 설계 및 구현, 데이터베이스와 문제 생성 모듈 연동
10주차	문제 제공 및 피드백 기능 개발	UI 설계 및 문제 제공 화면 개발, 피드백 알고리즘 설계, 서버-클라이언트 간 데이터 전송 모듈 개발, UI에서 문제 풀기 및 피드백 제공 기능 구현
11~13주차	테스트 및 디버깅	자신이 담당한 모듈의 테스트 및 디버깅, 통합 테스트 진행 및 최종 디버깅, 테스트 중 발견된 문제 해결 및 성능 최적화 작업
14주차	발표	기능 구현 및 성과 정리, 시스템 구조 및 기술적 상세 설명 작성, 시연 준비 및 발표 자료 제작
15주차	최종 보고서 작성	개요 및 설계 설명

5. 역할 분담

5.1 유현서 – LLM 모델 통합 및 문제 생성 알고리즘 개발, 프론트엔드

- 📁 LLM을 활용한 학습 자료 분석

- 📁 객관식 및 주관식 문제 생성 알고리즘 개발

5.2 박진영 – 백엔드 및 서버 관리

- 📁 서버 개발 및 관리

- 📁 데이터 처리 및 API 설계

- 📁 사용자 요청 처리 및 문제 생성 결과 반환

<2>SW 상세 설계서

1. 서론

1-1. 프로젝트 개요

- EduCraft는 학습자가 능동적으로 학습할 수 있도록, LLM을 활용해 객관식 및 주관식 문제를 자동 생성하고 즉각적인 피드백을 제공하는 학습 보조 시스템입니다.

1-2. 문제 정의

- 현재 학생들은 스스로 학습 내용을 점검하거나 문제를 만들 수 있는 자원이 부족하여, 자기주도적인 학습이 어렵습니다.

- 특히 비대면 학습 상황에서는 효과적인 피드백과 문제 생성 도구가 없어서 학습자가 학습 성과를 적시에 확인하거나 능동적으로 학습을 이어가는 데 한계가 있습니다.

- 이를 해결하기 위해, LLM 기반의 문제 생성 및 피드백 시스템을 도입하여 학생들이 학습 내용을 쉽게 점검하고, 주도적으로 학습을 진행할 수 있도록 돕는 것이 목표입니다.

2. 요구사항 분석

2-1. 기능 요구사항

2-1-1. 자료 수집 및 입력

- ☞ 학습자가 사용할 학습 자료를 입력할 수 있어야 한다.
- ☞ 다양한 포맷(텍스트, PDF 등)의 자료를 지원해야 한다.

2-1-2. 데이터 전처리

- ☞ 입력된 학습 자료를 문제 생성에 적합한 형태로 전처리해야 한다.
- ☞ 불필요한 데이터를 제거하고, 텍스트를 구조화된 형식으로 변환해야 한다.

2-1-3. 문제 생성

- ☞ LLM을 이용해 학습 자료를 기반으로 객관식 및 주관식 문제를 자동으로 생성해야 한다.
- ☞ 문제 생성 시 난이도, 문제 유형 등을 선택할 수 있는 옵션을 제공해야 한다.

2-1-4. 피드백 제공

- ☞ 학습자가 문제를 풀고 나면 즉각적으로 정답 여부와 함께 피드백을 제공해야 한다.
- ☞ 피드백에는 정답 설명과 추가 학습 자료가 포함될 수 있어야 한다.

2-1-5. 사용자 관리 및 설정

- ☞ 학습자는 계정을 생성하고, 학습 진행 상황을 저장할 수 있어야 한다.
- ☞ 학습자 맞춤 설정(예 : 선호하는 문제 유형, 난이도 등)을 지원해야 한다.

2-2. 비기능 요구사항

2-2-1. 성능

- ☞ 문제 생성 시간은 3초 이내로 완료되어야 한다.
- ☞ 서버는 동시 접속자 100명 이상을 지원할 수 있어야 한다.

2-2-2. 확장성

- ☞ 시스템은 증가하는 데이터와 사용자를 처리할 수 있도록 확장 가능한 구조로 설계되어야 한다.
- ☞ 새로운 학습 자료 유형이나 문제 유형을 쉽게 추가할 수 있어야 한다.

2-2-3. 보안

- ☞ 사용자 데이터는 안전하게 보호되어야 하며, 권한에 따른 접근 제어가 필요하다.
- ☞ 데이터 전송 시 암호화된 통신 방식을 사용해야 한다.

2-2-4. 사용성

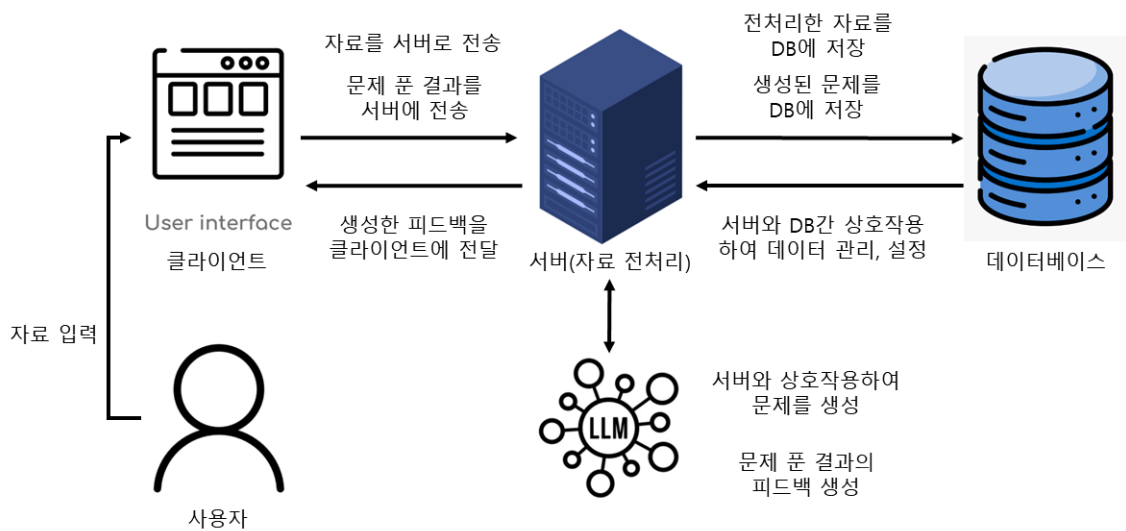
- ☞ 사용자 인터페이스는 직관적이고 사용하기 쉬워야 한다.
- ☞ 모바일, 태블릿, 데스크톱 등 다양한 기기에서 원활하게 작동해야 한다.

2-2-5. 신뢰성

- ☞ 시스템은 고가용성을 유지해야 하며, 다운타임을 최소화해야 한다.
- ☞ 오류 발생 시 자동 복구 기능을 제공하여 사용자 경험에 영향을 최소화해야 한다.

3. 시스템 아키텍처 설계

3-1. 시스템 구조도



3-2. 모듈 간 관계

3-2-1. 클라이언트 <-> 서버

📖 사용자가 클라이언트에서 학습 자료를 입력하거나 문제를 요청하면, 클라이언트는 이를 서버로 전송합니다. 서버는 클라이언트 요청에 따라 문제를 생성하거나 피드백을 제공합니다.

📖 데이터 흐름 :

학습 자료 입력 -> 서버 전송(POST)

문제 생성 요청 -> 서버 전송(GET/POST)

피드백 요청 -> 서버 전송(GET)

결과 수신 -> 클라이언트 출력

3-2-2. 서버 <-> 데이터베이스

☞ 서버는 학습자가 입력한 데이터를 데이터베이스에 저장하고, 문제 생성 및 피드백 제공을 위해 데이터를 읽어옵니다.

☞ 데이터 흐름 :

학습 자료 저장 (INSERT/UPDATE)

생성된 문제 저장 및 조회 (SELECT/INSERT)

사용자 진행 상황 저장 (UPDATE)

사용자 정보 조회 및 업데이트(SELECT/UPDATE)

3-2-3. 서버 <-> LLM (문제 생성 모듈)

☞ 서버는 학습 자료를 LLM에 전달하여, 이를 기반으로 문제를 생성합니다. 생성된 문제는 서버로 다시 전달되어 클라이언트에 제공됩니다.

☞ 데이터 흐름 :

학습 자료 -> LLM 전달 (POST)

LLM에서 문제 생성 -> 서버로 결과 반환 (JSON 데이터 형식)

3-2-4. 클라이언트 <-> 데이터베이스 (간접적인 연결)

☞ 클라이언트는 직접 데이터베이스와 연결되지 않으며, 서버를 통해 데이터베이스에 접근합니다. 사용자는 서버에 요청을 보내면 서버가 데이터베이스에서 필요한 데이터를 조회하여 클라이언트에 전달합니다.

4. 모듈 설계

4-1. 클라이언트 모듈

4-1-1. 기능

- 사용자 인터페이스(UI) 제공
- 학습 자료 입력
- 문제 생성 요청
- 문제 풀이 및 피드백 제공

4-1-2. 데이터

- 사용자 입력 학습 자료
- 서버로부터 받은 문제 및 피드백

4-1-3. 주요 컴포넌트

- UI/UX : 웹 또는 모바일 기반의 인터페이스 설계
- API 호출 : 서버로 데이터를 전송 및 결과 수신 (REST API)
- 데이터 표시 : 생성된 문제 및 피드백을 사용자에게 출력

4-2. 서버 모듈

4-2-1. 기능

- 클라이언트 요청 처리

📁 데이터베이스 및 LLM과 통신

📁 문제 생성 및 피드백 처리

4-2-2. 데이터

- 📖 학습 자료 및 문제 정보
- 📖 사용자 정보 및 학습 기록

4-2-3. 주요 컴포넌트

- 📖 요청 처리기 : 클라이언트의 요청을 받아서 적절히 분기 처리 (GET, POST 등)
- 📖 데이터 처리기 : 학습 자료를 전처리하여 LLM으로 전달
- 📖 LLM 연동기 : LLM과 통신하여 문제 생성 및 결과를 수신
- 📖 API 설계 : 클라이언트와의 통신을 위한 REST API 설계

4-3. 데이터베이스 모듈

4-3-1. 기능

- 📖 학습 자료, 문제, 사용자 정보 저장 및 관리
- 📖 서버에서 필요할 때 데이터를 읽고 쓰기 가능

4-3-2. 데이터

- 📖 학습 자료 및 문제 데이터
- 📖 사용자 정보 및 학습 기록

4-3-3. 주요 컴포넌트

- 📖 학습 자료 테이블 : 입력된 자료 저장
- 📖 문제 테이블 : 생성된 문제 저장
- 📖 사용자 테이블 : 사용자 계정 정보 및 학습 진행 기록 저장

📁 질의 처리기 : 서버로부터 오는 요청에 따라 데이터 조회/수정/삭제

4-4. LLM 모듈

4-4-1. 기능

📁 학습 자료를 분석하여 문제 생성

📁 서버로부터 받은 학습 자료를 기반으로 문제 유형 결정

4-4-2. 데이터

📁 학습 자료 입력

📁 생성된 문제 데이터

4-4-3. 주요 컴포넌트

📁 문제 생성기 : LLM을 통해 학습 자료에서 객관식 및 주관식 문제 생성

📁 문제 전달기 : 생성된 문제 데이터를 서버에 반환

📁 피드백 분석기 : 문제 풀이 후 피드백 생성

5. 데이터베이스 설계

5-1. 데이터베이스 요구사항

- 📁 학습 자료와 생성된 문제를 저장
- 📁 사용자 정보를 관리하고 학습 진행 상태를 기록
- 📁 문제 풀이 및 피드백 기록 저장

5-2. 주요 테이블 설계

5-2-1. User 테이블 (사용자 정보)

- 📁 사용자 계정 정보 및 학습 진행 상태를 저장

📁 필드 :

user_id(PK) : 사용자 고유 ID

username : 사용자 이름

email : 이메일 주소

password: 암호화된 비밀번호

created_at : 계정 생성일

progress : 학습 진행 상태 (문제 풀이 기록 등)

5-2-2. StudyMaterial 테이블 (학습 자료)

- 📁 설명 : 사용자가 입력한 학습 자료를 저장

📁 필드 :

material_id(PK) : 학습 자료 고유 ID

user_id(FK) : 자료를 입력한 사용자 ID (User 테이블 참조)

title : 학습 자료 제목

content : 학습 자료 내용 (텍스트, 문서 등)

created_at : 자료 입력일

5-2-3. Question 테이블 (생성된 문제)

📁 LLM을 통해 생성된 문제를 저장

📁 필드 :

question_id(PK) : 문제 고유 ID

material_id(FK) : 문제 생성에 사용된 학습 자료 ID (StudyMaterial 테이블 참조)

question_type : 문제 유형 (객관식, 주관식)

question_content : 문제 내용

created_at : 문제 생성일

5-2-4. Answer 테이블 (문제 풀이 기록)

📁 사용자가 문제를 풀이한 결과 및 피드백을 저장

📁 필드 :

answer_id(PK) : 답변 고유 ID

user_id(FK) : 문제를 풀이한 사용자 ID (User 테이블 참조)

question_id(FK) : 풀이한 문제 ID (Question 테이블 참조)

user_answer : 사용자가 입력한 답변

feedback : 자동 생성된 피드백

created_at : 답변 제출일

5-3. 테이블 간 관계

5-3-1. User <-> StudyMaterial

☞ 사용자가 입력한 학습 자료를 저장하고, 학습 자료를 기반으로 문제를 생성

5-3-2. StudyMaterial <-> Question

☞ 학습 자료에서 생성된 문제를 기록

5-3-3. User <-> Answer <-> Question

☞ 사용자가 문제를 풀이하고, 그 결과와 피드백을 저장

5-4. 정규화 및 확장성

☞ 정규화 : 각 테이블은 1차, 2차, 3차 정규형에 맞춰 설계되어 중복 데이터를 최소화하고 데이터 무결성을 보장.

☞ 확장성 : 이후 추가 기능이나 데이터를 고려해 테이블 구조를 쉽게 확장할 수 있도록 설계.

6. API 설계

6-1. API 요구사항

- 📖 학습 자료를 입력, 수정, 삭제하는 기능
- 📖 학습 자료 기반으로 문제를 생성하는 기능
- 📖 문제 풀이 및 피드백 제공 기능
- 📖 사용자 관리 (회원가입, 로그인, 정보 수정)

6-2. 주요 API 엔드포인트 설계

6-2-1. 사용자 관련 API

- 📖 POST /users/signup : 사용자 회원가입

Request Body : username, email, password

Response : 성공 여부, 사용자 ID

- 📖 POST /users/login : 사용자 로그인

Request Body : email, password

Response : 인증 토큰 (JWT) 또는 세션 정보

- 📖 PUT /users/{user_id} : 사용자 정보 수정

Request Body : username, password

Response : 성공 여부

6-2-2. 학습 자료 관련 API

- 📖 POST /materials : 학습 자료 업로드

Request Body : user_id, title, content

Response : 성공 여부, 자료 ID

📄 GET /materials/{material_id} : 특정 학습 자료 조회

Response : material_id, title, content, created_at

📄 PUT /materials/{material_id} : 학습 자료 수정

Request Body : title, content

Response : 성공 여부

📄 DELETE /materials/{material_id} : 학습 자료 삭제

Response : 성공 여부

6-2-3. 문제 생성 및 피드백 관련 API

📄 POST /questions/generate : 문제 자동 생성

Request Body : material_id, question_type (객관식/주관식)

Response : 생성된 문제 목록 (질문 ID, 질문 내용)

📄 POST /answers : 문제 풀이 결과 제출 및 피드백 요청

Request Body : user_id, question_id, user_answer

Response : 피드백 정보 (자동 생성된 피드백)

6-3. API 동작 예시

6-3-1. 회원가입

📄 클라이언트에서 /users/signup 에 POST 요청을 보내면, 서버에서 사용자 정보를 저장하고 성공 여부를 응답합니다.

POST /users/signup

{

"username": "JohnDoe",

```
"email": "johndoe@example.com",  
"password": "securePassword123"  
}
```

Response:

```
{  
"status": "success",  
"user_id": 1  
}
```

6-3-2. 학습 자료 업로드

📁 클라이언트에서 /materials 에 POST 요청을 보내면, 서버가 학습 자료를 저장하고 자료 ID를 반환합니다.

POST /materials

```
{  
"user_id": 1,  
"title": "Math Study Notes",  
"content": "Pythagoras theorem explanation..."  
}
```

Response:

```
{  
"status": "success",  
"material_id": 101  
}
```

```
}
```

6-3-3. 문제 생성 요청

☞ 클라이언트가 /questions/generate에 POST 요청을 보내면, LLM을 통해 문제를 생성하고, 서버가 해당 문제 목록을 반환합니다.

POST /questions/generate

```
{  
  "material_id": 101,  
  "question_type": "multiple_choice"  
}
```

Response:

```
{  
  "status": "success",  
  "questions": [  
    {  
      "question_id": 301,  
      "content": "What is the hypotenuse of a right triangle?"  
    },  
    ...  
  ]  
}
```

6-3-4. 문제 풀이 및 피드백 제공

☞ 사용자가 문제 풀이 결과를 서버에 전송하면, 서버는 해당 답변에 대한 피드백을 제공합니다.

POST /answers

```
{  
  "user_id": 1,  
  "question_id": 301,  
  "user_answer": "5"  
}
```

Response:

```
{  
  "status": "success",  
  "feedback": "Correct! The hypotenuse is indeed 5."  
}
```

6-4. 비기능적 요구사항

🔒 보안 : 사용자 인증은 JWT나 OAuth를 사용해 처리하고, 모든 API는 HTTPS를 통해 암호화된 채널에서 통신.

🚀 성능 : 다수의 동시 요청을 처리할 수 있도록 서버 성능을 최적화.

🔧 확장성 : 향후 문제 유형 추가 및 새로운 기능을 쉽게 구현할 수 있도록 API를 확장 가능하게 설계.

🛠️ 유지보수성 : 코드 및 API 문서화 철저, 버전 관리 필요.

7. UI/UX 설계

7-1. 목표 설정

- ☞ 사용자 친화적인 인터페이스 : 사용자들이 프로그램을 쉽게 이해하고 사용할 수 있도록 직관적이고 간결한 디자인 제공.
- ☞ 효율적인 작업 흐름 : 학습 자료 업로드, 문제 생성, 피드백 확인 과정이 매끄럽고 간편하도록 구성.
- ☞ 반응형 디자인 : 다양한 화면 크기(모바일, 태블릿, 데스크톱)에 적합한 UI 구현.

7-2. UI/UX 구성 요소

7-2-1. 로그인 및 회원가입 화면

☞ 구성

- ① 사용자 이름, 이메일, 비밀번호 입력 필드
- ② 로그인/회원가입 버튼
- ③ 비밀번호 찾기 링크

☞ UX 개선 포인트

- ① 입력 오류 시 즉각적인 피드백 제공
- ② 소셜 로그인 옵션 추가 (Google, Facebook 등)

7-2-2. 대시보드 화면

☞ 구성

- ① 최근 학습 자료 리스트
- ② 새 문제 생성 버튼
- ③ 피드백 요청 버튼

☞ UX 개선 포인트

- ① 중요 알림 및 공지사항을 상단에 배치
- ② 개인 학습 진척도 표시 (진행률 바)

7-2-3. 학습 자료 업로드 화면

구성

- ① 제목 및 내용 입력 필드
- ② 파일 업로드 옵션
- ③ 업로드 버튼

UX 개선 포인트

- ① 입력 예시 제공 및 필요한 필드 강조
- ② 업로드 후 성공 메시지 또는 오류 메시지 제공

7-2-4. 문제 생성 화면

구성

- ① 생성할 문제 유형 선택 (객관식, 주관식)
- ② 문제 내용 및 옵션 입력 필드
- ③ 문제 생성 버튼

UX 개선 포인트

- ① 문제 유형별 필드 자동 표시/숨김 기능
- ② 생성된 문제 미리 보기 기능 추가

7-2-5. 문제 풀이 화면

구성

- ① 문제 내용 및 선택지 표시
- ② 답변 제출 버튼

③ 즉각적인 피드백 표시 영역

☞ UX 개선 포인트

- ① 답변 제출 후 피드백을 즉시 확인할 수 있도록 배치
- ② 이전 문제 및 다음 문제 탐색 버튼 추가

7-3. 색상 및 폰트

7-3-1. 색상 팔레트

☞ 기본 색상 : 신뢰와 안정성을 주는 파란색 계열

☞ 강조 색상 : 주의와 흥미를 유도하는 주황색 또는 노란색

7-3-2. 폰트 선택

☞ 가독성이 높은 sans-serif 폰트 사용 (예: Arial, Roboto)

☞ 제목 및 본문에 적절한 폰트 크기 설정 (제목 > 본문)

7-4. 와이어프레임 설계

☞ 각 화면의 레이아웃을 시각적으로 표현한 와이어프레임을 설계합니다.

☞ 도구 : Figma, Adobe XD, Sketch 등의 프로토타입 도구 사용 가능.

☞ 주요 화면의 흐름과 구성 요소를 시각화하여 팀원과 공유 및 피드백을 받을 수 있습니다.

7-5. 사용자 테스트

☞ 초기 디자인에 대한 사용자 피드백을 받고, 사용자의 요구와 기대에 맞게 UI/UX를 조정합니다.

☞ A/B 테스트: 두 가지 버전을 비교하여 더 나은 성능을 가진 디자인을 선택합니다.

8. 테스트 계획

8-1. 테스트 목표

- ☞ 학습 보조 프로그램의 기능이 요구사항에 맞게 동작하는지 확인.
- ☞ 각 모듈 간의 상호작용이 원활한지 확인.
- ☞ 사용자의 학습 경험을 방해하는 오류나 성능 저하를 발견하고 수정.

8-2. 테스트 범위

- ☞ 기능 테스트 : 문제 생성, 자료 입력, 피드백 제공 등 주요 기능이 제대로 동작하는지 테스트.
- ☞ 사용자 인터페이스(UI) 테스트 : 화면 전환, 버튼 클릭, 반응 속도 등이 사용자의 기대에 맞게 동작하는지 확인
- ☞ 성능 테스트 : 학습 자료의 양이 많아졌을 때 시스템의 성능이 저하되는지 여부 확인.
- ☞ 보안 테스트 : 사용자 데이터의 안전성을 확인하고, 악의적인 접근에 대해 취약한 부분이 있는지 검토.
- ☞ 호환성 테스트 : 다양한 기기와 브라우저 환경에서 정상적으로 동작하는지 확인 (예 : 데스크톱, 모바일, 크롬, 사파리)

8-3. 테스트 종류

8-3-1. 단위 테스트 (Unit Testing)

- ☞ 목적 : 각각의 모듈이나 기능이 독립적으로 정상 동작하는지 확인.
- ☞ 테스트 도구 : Jest, Mocha (JavaScript), Catch2 (C++) 등.
- ☞ 예시 : 문제 생성 알고리즘이 올바르게 작동하는지 테스트.

8-3-2. 통합 테스트 (Integration Testing)

📖 목적 : 여러 모듈이 함께 작동할 때 발생하는 오류를 발견.

📖 테스트 도구 : Postman (API 통합 테스트), Selenium (UI 통합 테스트).

📖 예시 : LLM 모델에서 생성된 문제가 DB에 제대로 저장되고, 사용자에게 피드백이 정상적으로 반환되는지 확인.

8-3-3. 시스템 테스트 (System Testing)

📖 목적 : 전체 시스템이 요구사항에 맞게 동작하는지 확인.

📖 예시 : 학습자가 자료를 업로드하고 문제를 푼 후, 피드백을 정확히 받는지 시스템 전반을 테스트.

8-3-4. 성능 테스트 (Performance Testing)

📖 목적 : 다수의 사용자가 동시에 접속하거나, 많은 양의 데이터를 처리할 때 시스템이 견딜 수 있는지 확인.

📖 예시 : 대량의 학습 자료를 업로드하거나, 수백 명의 사용자가 동시에 문제를 생성할 때 시스템 반응 시간 측정.

8-3-5. 사용자 수용 테스트 (UAT)

📖 목적 : 실제 사용자들이 프로그램을 사용할 때의 경험을 평가

📖 예시 : 학생들에게 시스템을 사용해보게 하고, 그들의 피드백을 기반으로 최종 개선.

8-4. 테스트 일정

📖 단위 테스트 : 매주 진행 (각 팀원은 자신의 모듈 테스트).

📖 11주차 : 통합 테스트 및 성능 테스트.

📖 12주차 : 시스템 테스트 및 사용자 수용 테스트.

📖 13주차 : 사용자 피드백 반영 및 최종 버그 수정.

8-5. 테스트 결과 보고서

📖 테스트 결과를 정리하여 어떤 테스트에서 문제가 발생했는지, 해결 방법은 무엇인지 기록.

📖 테스트 로그 작성 : 오류 발생 시 로그 파일을 통해 어떤 상황에서 오류가 발생했는지 기록.

📖 테스트 완료 기준 : 모든 주요 기능 테스트에서 오류가 발견되지 않고, 성능 테스트와 보안 테스트에서 기준을 충족할 때 완료.

9. 보안 및 성능 고려 사항

9-1. 보안 고려 사항

9-1-1. 데이터 보호

☞ 암호화 : 사용자의 개인 정보 및 학습 자료는 저장 및 전송 시 암호화하여 보호. 예를 들어, AES (Advanced Encryption Standard)를 사용하여 데이터베이스에 저장된 정보를 암호화.

☞ SSL/TLS : 클라이언트와 서버 간의 데이터 전송 시 SSL/TLS 프로토콜을 사용하여 데이터의 무결성과 기밀성 보장.

9-1-2. 접근 제어

☞ 사용자 인증 : 사용자 가입 및 로그인 과정에서 강력한 비밀번호 정책을 적용하고, 2단계 인증(2FA)을 도입하여 계정 보안 강화.

☞ 권한 관리 : 역할 기반 접근 제어(RBAC)를 통해 사용자의 권한을 엄격하게 관리. 예를 들어, 관리자와 일반 사용자의 접근 권한을 구분.

9-1-3. 데이터 유출 방지

☞ 입력 검증 : 사용자로부터 입력받는 모든 데이터에 대해 철저한 검증을 수행하여 SQL 인젝션, 크로스 사이트 스크립팅(XSS) 등 공격으로부터 시스템을 보호.

☞ 로그 및 모니터링 : 시스템 로그를 통해 의심스러운 활동을 감지하고, 정기적으로 보안 점검을 수행하여 취약점을 신속히 발견하고 대응.

9-1-4. 서버 보안

☞ 방화벽 설정 : 서버에 접근하는 네트워크 트래픽을 모니터링하고 불법적인 접근을 차단하기 위해 방화벽을 적절히 설정.

☞ 소프트웨어 업데이트 : 운영 체제 및 모든 소프트웨어의 최신 보안 패치를 적용하여 알려진 취약점을 제거.

9-2. 성능 고려 사항

9-2-1. 응답 시간

☞ 최적화 : 데이터베이스 쿼리 및 알고리즘을 최적화하여 응답 시간을 최소화. 예를 들어, 인덱스 사용 및 쿼리 성능 개선.

☞ 캐싱 : 자주 조회되는 데이터를 캐싱하여 서버 부하를 줄이고, 응답 속도를 향상. 예를 들어, Redis 또는 Memcached와 같은 인메모리 데이터 저장소를 활용.

9-2-2. 부하 분산

☞ 로드 밸런서: 사용자의 요청을 여러 서버에 분산시켜 서버의 부하를 균등하게 관리하고, 시스템의 가용성을 높임.

☞ 수평 확장성: 사용자 수가 증가할 때 서버를 추가하여 성능을 향상시키는 구조 설계.

9-2-3. 자원 관리

☞ 메모리 및 CPU 사용량 모니터링: 성능 저하를 예방하기 위해 서버의 메모리 및 CPU 사용량을 모니터링하고, 필요 시 자원을 추가.

☞ 비동기 처리: 사용자 요청 처리 시 비동기 처리를 도입하여, 동시에 여러 작업을 수행하여 성능을 개선.

9-2-4. 테스트 및 모니터링

☞ 부하 테스트: 성능을 미리 예측하고 병목 현상을 발견하기 위해 부하 테스트를 수행.

☞ 모니터링 도구: Prometheus, Grafana와 같은 도구를 사용하여 시스템의 성능 지표를 모니터링하고, 문제 발생 시 즉각적으로 대응.

10. 개발 환경 및 도구

10-1. 개발 환경

10-1-1. 운영 체제

📁 개발 환경: Windows 11, Ubuntu 22.04 LTS

📁 서버 환경: Ubuntu 22.04 LTS (클라우드 호스팅 서비스 이용)

10-1-2. 프로그래밍 언어 및 프레임워크

📁 백엔드:

언어: JavaScript

프레임워크: NodeJS

📁 프론트엔드:

언어: JavaScript, TypeScript

프레임워크: React.js (UI/UX 구현)

📁 LLM 통합:

언어: JavaScript

API: OpenAI GPT-4 API

10-1-3. 데이터베이스


📁 RDBMS: MySQL

10-1-5. 클라우드 서비스


📁 AWS (Amazon Web Services): EC2 (서버 호스팅), RDS (데이터베이스), S3 (파일 저장)

10-2. 개발 도구


10-2-1. 통합 개발 환경 (IDE) 및 에디터

 VS Code: 주요 코딩 작업

10-2-2. 테스트 및 디버깅 도구

 Postman: API 테스트

10-2-4. 데이터 관리 및 분석 도구

 MySQL WorkBench: 데이터베이스 관리 및 쿼리 실행


<3> 최종 발표 자료



EduCraft

LLM기반 학습 보조 프로그램 - 9팀

유현서 박진영



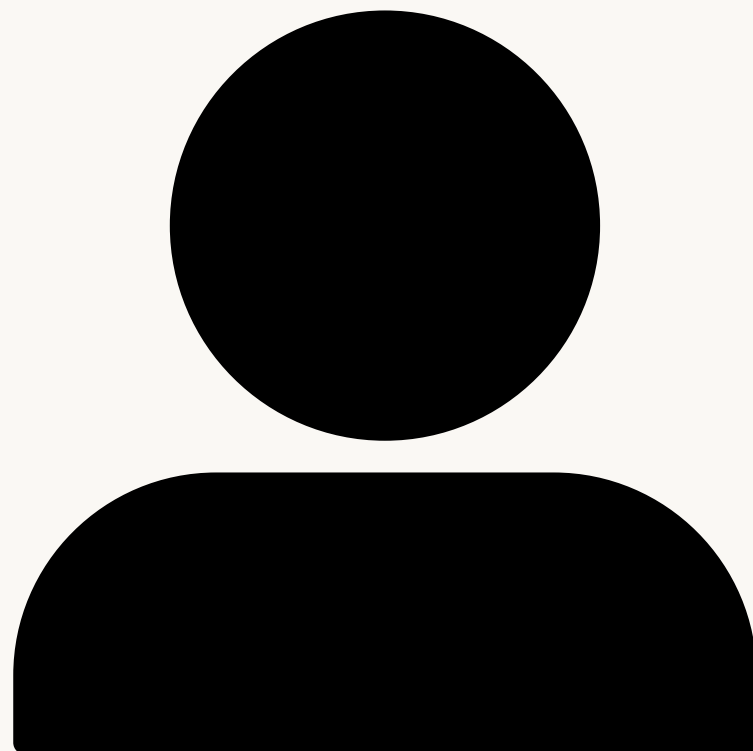
Contents.

01 팀원 역할

02 시스템 구조도

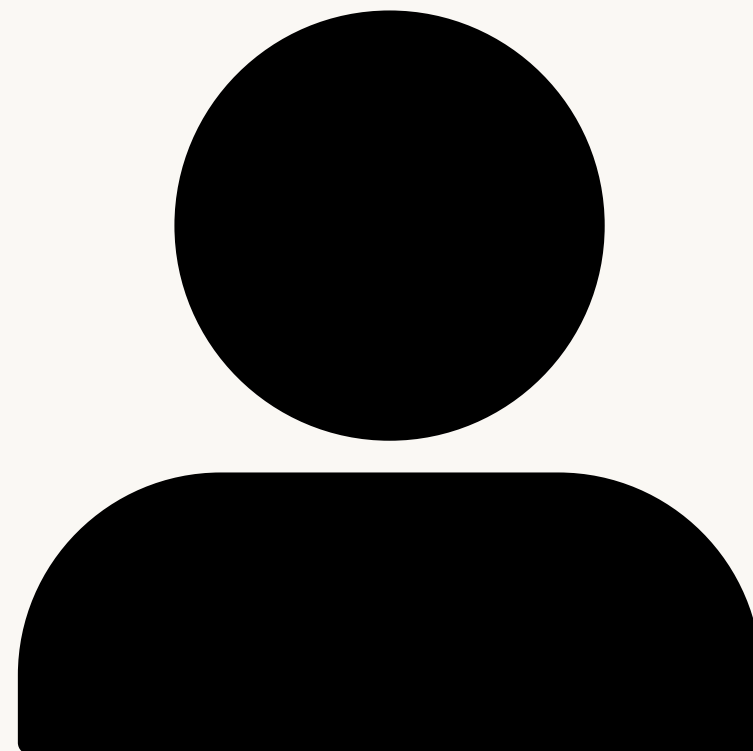
03 구현된 기능들

역할 분담



유현서

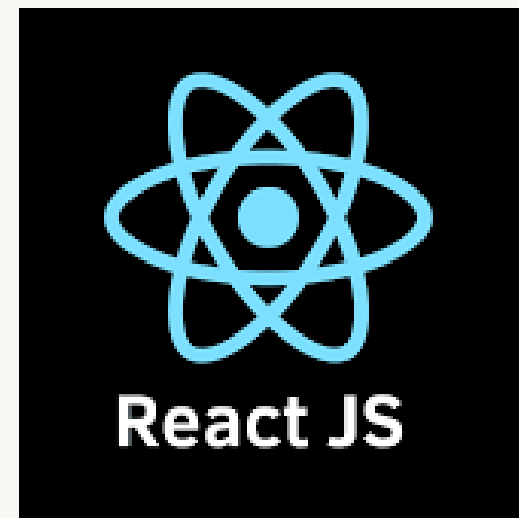
문제 생성 알고리즘 개발
서버 개발 및 관리
사용자 UI 설계



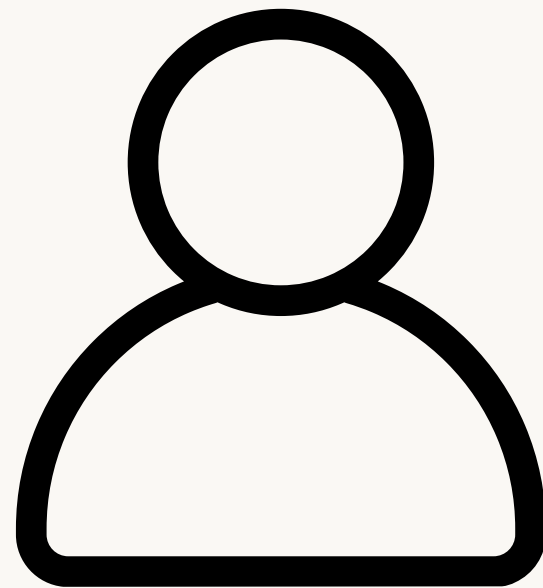
박진영

프론트엔드 개발 및 API 설계

시스템 구조도



UI

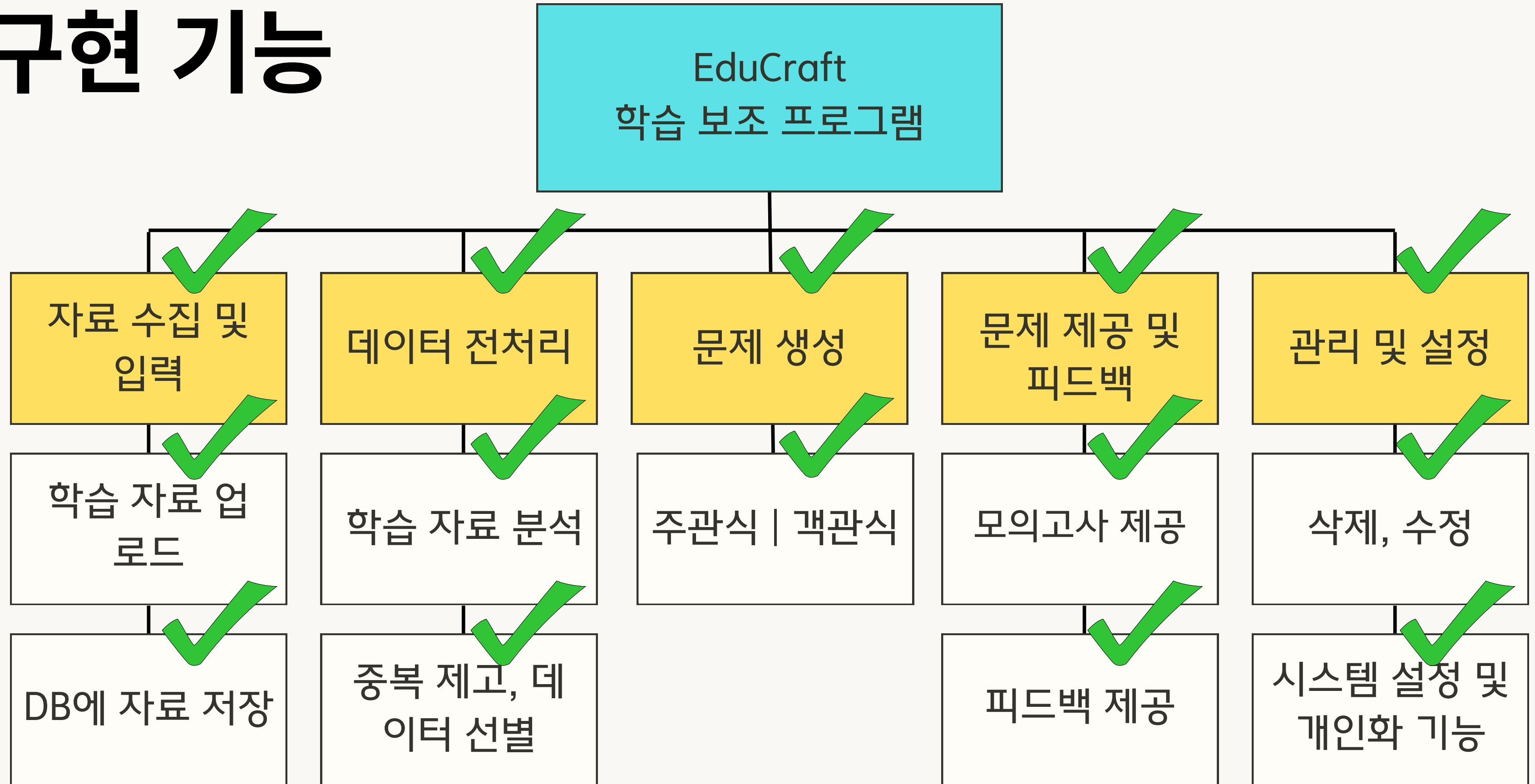


문제, 피드백 요청 및 제공



DB에 자료, 문제 저장 및 관리

구현 기능



자료 화면 & 문제 생성

북마크

모의고사

자료 1 문제 만들기

파일 경로:

객관식 문제 수:
5

주관식 문제 수:
5

난이도:
중간

문제 생성 및 저장

EduCraft

자료 모음

북마크

모의고사

설정

업로드된 자료 목록

자료 1

자료 1번입니다

자료 정보 수정

문제 풀기

문제 만들기

자료 2

자료 2번입니다

자료 정보 수정

문제 풀기

문제 만들기

자료 3

자료 3번입니다

자료 정보 수정

문제 풀기

문제 만들기

자료 업로드

문제 풀기 및 북마크

EduCraft

자료1 - 문제 풀기

타이머: 0:16

문제: 프로그램에서 반환 값을 제시하고 답변을 받는 방식의 함수는 무엇인가?

답안:

답 확인하기

문제 삭제

북마크

문제: 유현서 학생이 2024년 2학기에 컴퓨터프로그래밍 수업에서 작성해야하는 마지막 프로그램의 주제는 무엇인가?

답안:

답 확인하기

문제 삭제

북마크 해제

문제: 작성한 프로그램의 기능 중, 문자열 안에 포함된 특정한 문자의 개수를 세는 함수는 어떻게 명명되었는가?

답안:

답 확인하기

문제 삭제

북마크

달기

EduCraft

자료 모음

북마크

모의고사

설정

북마크된 문제

다음 중 2024년 2학기 컴퓨터프로그래밍 12주차 Programming 과제에서 작성하는 프로그램의 기능을 올바르게 나열한 것은?

1. 문자열 역순 출력, 문자열 공백 삭제, 지정 문자 삭제

2. 문자열 빈도 계산, 알파벳 개수 출력, 특수문자 개수 출력

3. 사용자 질문에 대한 긍정/부정 반환, 단어 개수 계산, 비밀번호 입력 시 "*" 문자 표시

4. 모두

정답: 정답 확인하기

해설: 해설 보기

북마크 삭제

유현서 학생이 2024년 2학기에 컴퓨터프로그래밍 수업에서 작성해야하는 마지막 프로그램의 주제는 무엇인가?

정답: 정답 확인하기

해설: 해설 보기

북마크 삭제

모의고사 제공

모의고사 A: 모의고사 결과

제출일	걸린 시간	총 문제	맞은 문제 수	객관식	주관식	어려움	보통	쉬움	정답률
2024. 11. 27. 오전 10:07:57	0:0	12	0	0	0	0	0	0	0%
2024. 11. 27. 오전 10:28:49	0:1	12	0	0	0	0	0	0	0%
2024. 11. 27. 오후 1:07:19	0:1	12	0	0	0	0	0	0	0%
2024. 11. 27. 오후 1:12:06	0:1	12	0	0	0	0	0	0	0%
2024. 11. 27. 오후 1:15:06	0:1	12	0	0	0	0	0	0	0%
2024. 11. 27. 오후 1:19:13	0:12	12	2	2	0	0	0	2	17%

333: 모의고사 풀기

문제 상태

- ☐ 1번
- ☐ 2번
- ☐ 3번
- ☐ 4번
- ☐ 5번
- ☐ 6번
- ☐ 7번
- ☐ 8번
- ☐ 9번
- ☐ 10번
- ☐ 11번
- ☐ 12번
- ☐ 13번
- ☐ 14번
- ☐ 15번

Q1. 유현서가 2024년 2학기 컴퓨터프로그래밍 12주차 Programming 과제에, 아닌가?

- ☐ 1번: 문자열을 역순으로 출력
- ☐ 2번: 특정한 문자의 개수 세기
- ☐ 3번: 단어의 개수를 계산
- ☐ 4번: 문자열에서 숫자만 삭제

Q2. 유현서가 작성한 프로그램 중에서 긍정/부정 을 판단하는 함수에서 긍정을 의미하는 문자열로 가정한 것은 무엇인가?

정답:

Q3. 2024년 2학기 컴퓨터프로그래밍 12주차 Programming 과제에서 작성해야 하는 프로그램 중 문자열에서 사용자

취소

제출하기

자료 모음

북마크

모의고사

설정

모의고사 목록

333

333

모의고사 수정

모의고사 풀기

기록 보기

222

222

모의고사 수정

모의고사 풀기

기록 보기

모의고사 A

사용자 설정 화면

사용자 설정 화면에서는 현재 만들어져 있는 문제 수와 정답률을 확인할 수 있으며, 모바일 환경에서의 구동을 고려한 반응형 디자인을 제공하고 있습니다

EduCraft

자료 모음

북마크

모의고사

설정

설정



프로필 사진 변경

사용자 이름:

사용자 이메일:

	만든 문제 수	풀 문제 수	정답률
객관식	10개	10개	100%
주관식	23개	18개	96%

비밀번호 변경

로그아웃

계정 탈퇴



프로필 사진 변경

사용자 이름:

사용자 이메일:

	만든 문제 수	풀 문제 수	정답률
객관식	10개	10개	100%
주관식	23개	18개	96%

비밀번호 변경

로그아웃

계정 탈퇴

EduCraft

자료 모음

북마크

모의고사

프로필

업로드된 자료 목록

자료 1

자료 1번입니다

자료 정보 수정

문제 풀기

문제 만들기

자료 2

자료 2번입니다

자료 정보 수정

문제 풀기

문제 만들기

자료 3

자료 업로드

Thank you.

감사합니다