

캡스톤디자인 II 중간보고서(표지)

프로젝트명 : AI 챗봇을 활용한 온라인 강의 웹 서비스
캡스톤 디자인II, 중간보고서

Version 1.0

개발 팀원 명(팀리더):박상현
주준하
남승완

대표 연락처:010-5212-8903
e-mail: 20171612@edu.hanbat.ac.kr

캡스톤 디자인 II 중간보고서 내용

소스 코드

```
import './chat.css';
import React, { useState, useRef, useEffect } from 'react';

function Chat({ pdf }) {
  const [socketConnected, setSocketConnected] = useState(false);
  const endOfMessagesRef = useRef(null);
  const [newMessage, setNewMessage] = useState("");
  const [arrivalMessage, setArrivalMessage] = useState(null);
  const [messages, setMessages] = useState([]);

  const ws = useRef(new WebSocket("ws://localhost:9998"));

  useEffect(() => {
    ws.current.onopen = () => {
      console.log("server connect");
      setSocketConnected(true);
    };
    ws.current.onclose = () => {
      console.log("server Disconnect");
    };
    ws.current.onerror = (error) => {
      console.log("server error : " + error);
    };
  }, []);

  useEffect(() => {
    ws.current.onmessage = (message) => {
      const arrivalMessageData = {
        text: message.data,
        time:
          new Date(Date.now()).getHours() +
          ":" +
          new Date(Date.now()).getMinutes(),
        id: "frnd_message",
      };
      setArrivalMessage(arrivalMessageData);
    };
  });
}
```

```

    }, [arrivalMessage]);

    useEffect(() => {
      if (arrivalMessage && arrivalMessage !== "") {
        setMessages((prev) => [...prev, arrivalMessage]);
      }

      scrollToBottom();
    }, [arrivalMessage]);

    const handleSendMessage = (e) => {
      e.preventDefault();
      if (newMessage !== "") {
        const messageData = {
          text: newMessage,
          time:
            new Date(Date.now()).getHours() +
            ":" +
            new Date(Date.now()).getMinutes(),
          id: "my_message",
        };

        if (socketConnected) {
          let sendMessage;
          if (pdf === "") {
            sendMessage = `{"question":
"${messageData.text}"}`;
          } else {
            sendMessage = `{"question": "${messageData.text}",
"file": "${pdf}"}`;
          }
          console.log(sendMessage);
          ws.current.send(sendMessage);
          setMessages((list) => [...list, messageData]);
          setNewMessage("");
          scrollToBottom();
        }
      }
    };

    const scrollToBottom = () => {
      endOfMessagesRef.current.scrollToView({
        behavior: "smooth",
        block: "start",
      });
    };

```

```

    };

    const handleChangeInput = (e) => {
        setNewMessage(e.target.value);
    };

    return (
        <div className="chat_container">
            <div className="chat_box">
                {messages.map((messageContent) => {
                    if (messageContent.id === "my_message") {
                        return (
                            <div
                                className="message
my_message">
                                    <p>
                                        {messageContent.text}
                                    <br />
                                    <span>{messageContent.time}</span>
                                </p>
                            </div>
                        );
                    } else {
                        return (
                            <div
                                className="message
frnd_message">
                                    <p>
                                        {messageContent.text}
                                    <br />
                                    <span>{messageContent.time}</span>
                                </p>
                            </div>
                        );
                    }
                })}
                <div id="end_scroll" ref={endOfMessagesRef} />
            </div>

            <div className="chat_input">
                <form onSubmit={handleSendMessage}>
                    <input
                        type="text"

```

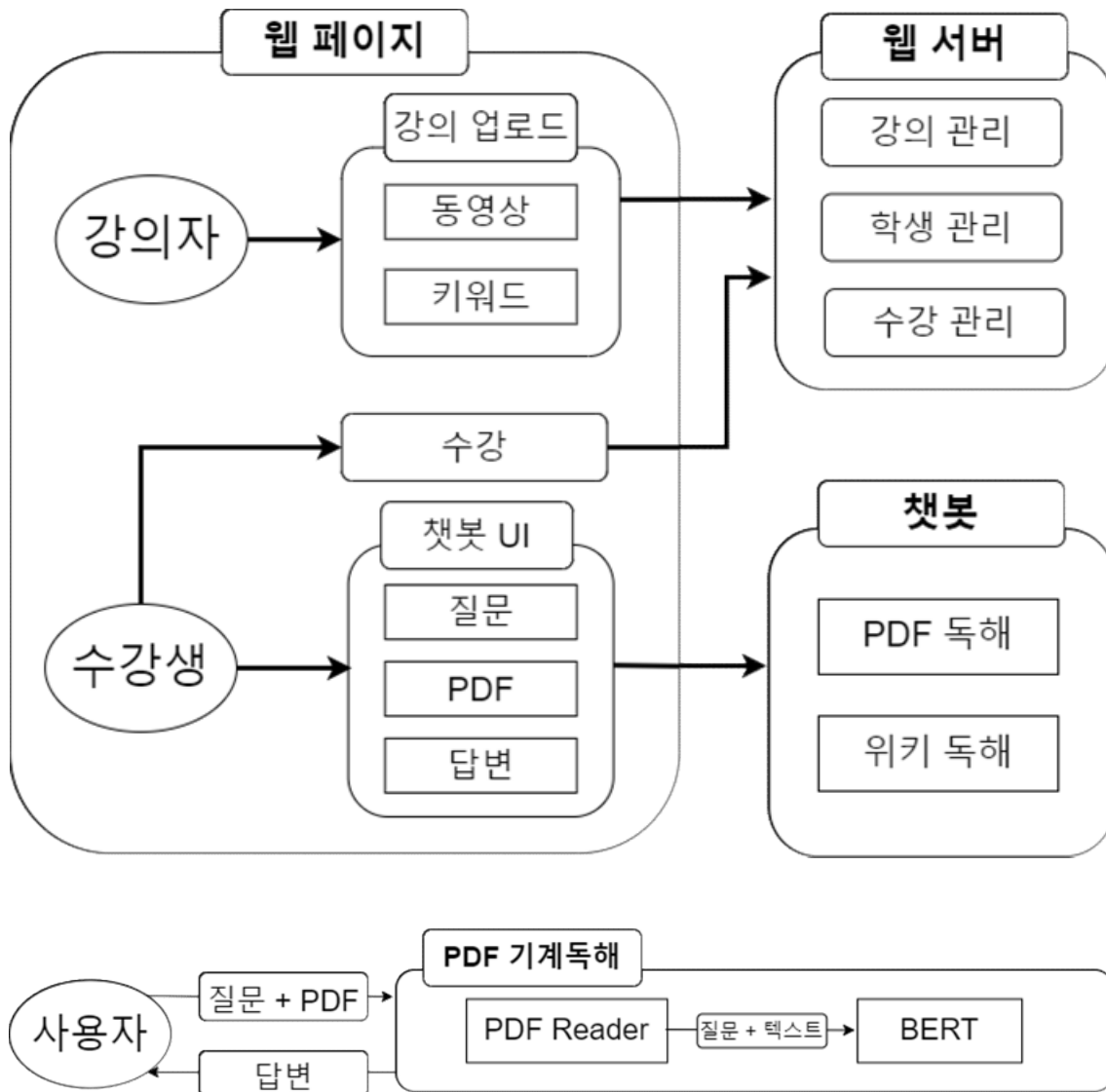
```

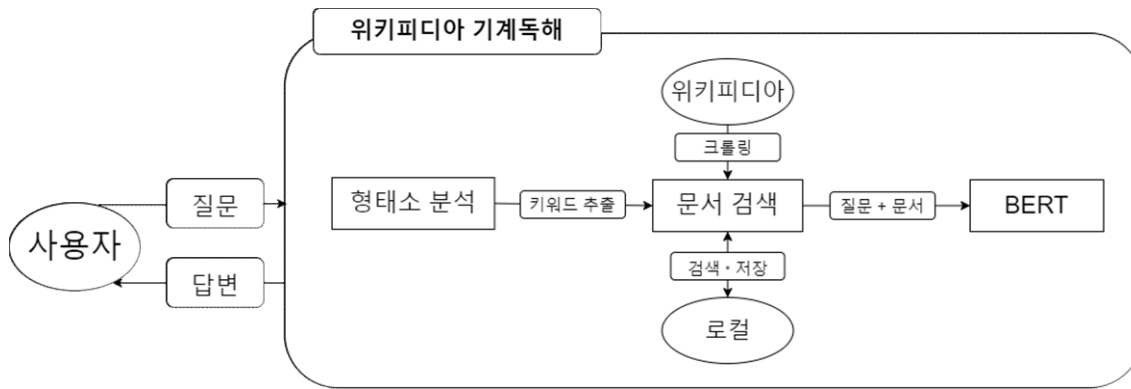
        value={newMessage}
        placeholder="챗봇에게 질문 하세요"
        onChange={handleChangeInput}
    ></input>
    <button
      onClick={handleSendMessage}>&#9658;</button>
  </form>
</div>
</div>
</div>
);
}

export default Chat;
챗팅 웹페이지 소스코드

```

시스템 구성





1. 질의응답 시스템

- 형태소 분석기 : 사용자의 질문을 분석해서 키워드(명사, 고유명사, 외래어 등)를 추출하고 문서 검색기로 전달한다.
- 문서 검색기 : 위키피디아에서 키워드에 대한 정보를 검색하고 문단 단위로 분할하여 언어 모델에 입력한다.
- BERT : 사용자 질문과 강의자료 또는 위키피디아 문서에 대한 임베딩을 생성하고 답변을 예측한다. 베이스 모델로 다국어 BERT 모델을 사용한다.

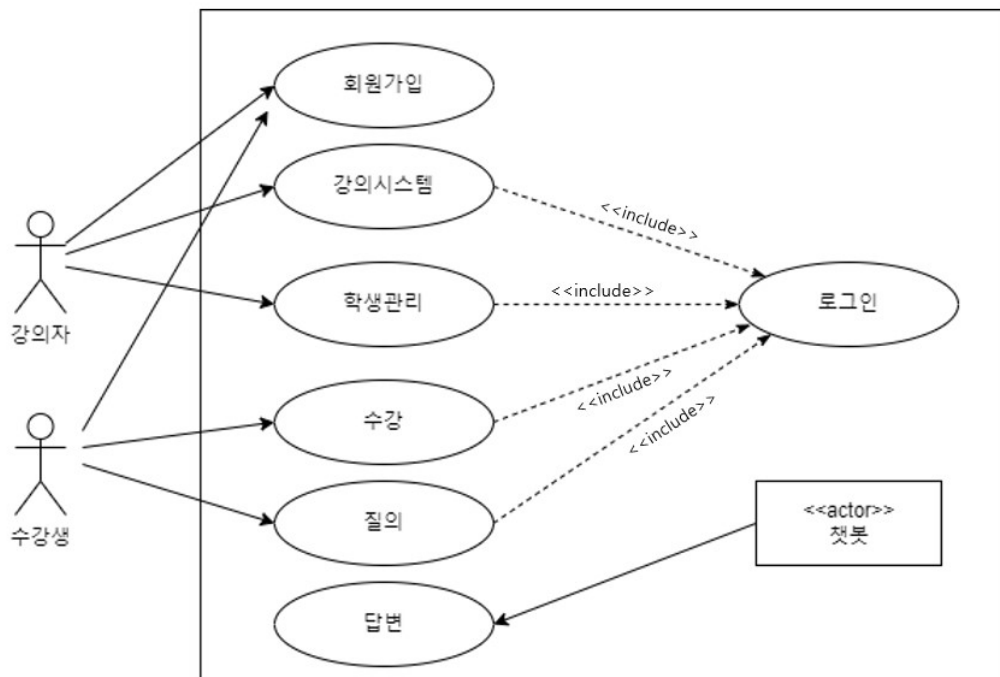
2. 강의 웹 서비스 서버

- 메인 페이지 : 사용자 로그인과 권한을 검증하고 사이트에 등록된 강의와 사용자의 최근 학습 강의를 목록화한다.
- 로그인 관리 : 이메일 검사를 통해 학교 관련자만 회원가입 가능하도록 하며 기본적인 로그인, 자동 로그인을 지원한다.
- 내 강의목록 페이지 : 사용자가 수강하는 전체 강의목록을 확인하고 강의 진도 등 강의에 대한 정보를 표시한다.
- 강의 대시보드 페이지 : 강의 대시보드를 통해 사용자가 등록한 강의에 대한 정보를 확인하는 기능을 제공한다.
- 강의 동영상 페이지 : 동영상 강의를 재생하고 챗봇 UI를 제공한다.
- 챗봇 채팅 : 사용자는 챗봇 UI를 통해서 질의응답을 수행할 수 있다.
- 강의 관리자 페이지 : 강의 관리자는 강의와 관련된 정보를 입력, 변경하거나 동영상을 업로드 할 수 있다.
- 회원 데이터 처리 : 사용자가 회원가입을 할 때 입력하는 id, password를 데이터베이스에

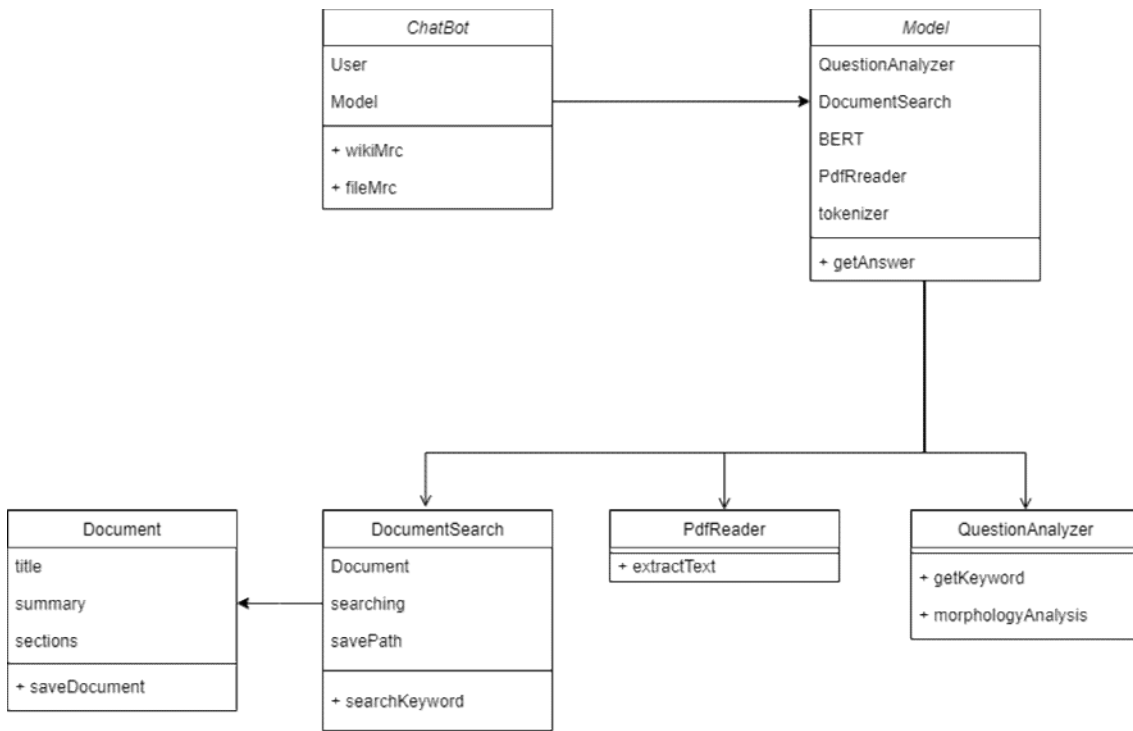
암호화하여 저장하고 가입자의 소속을 분류한다.

- 강사 데이터 처리 : 강사는 자신의 강의를 업로드하고 키워드를 입력하여 질의응답 성능을 올릴 수 있다.

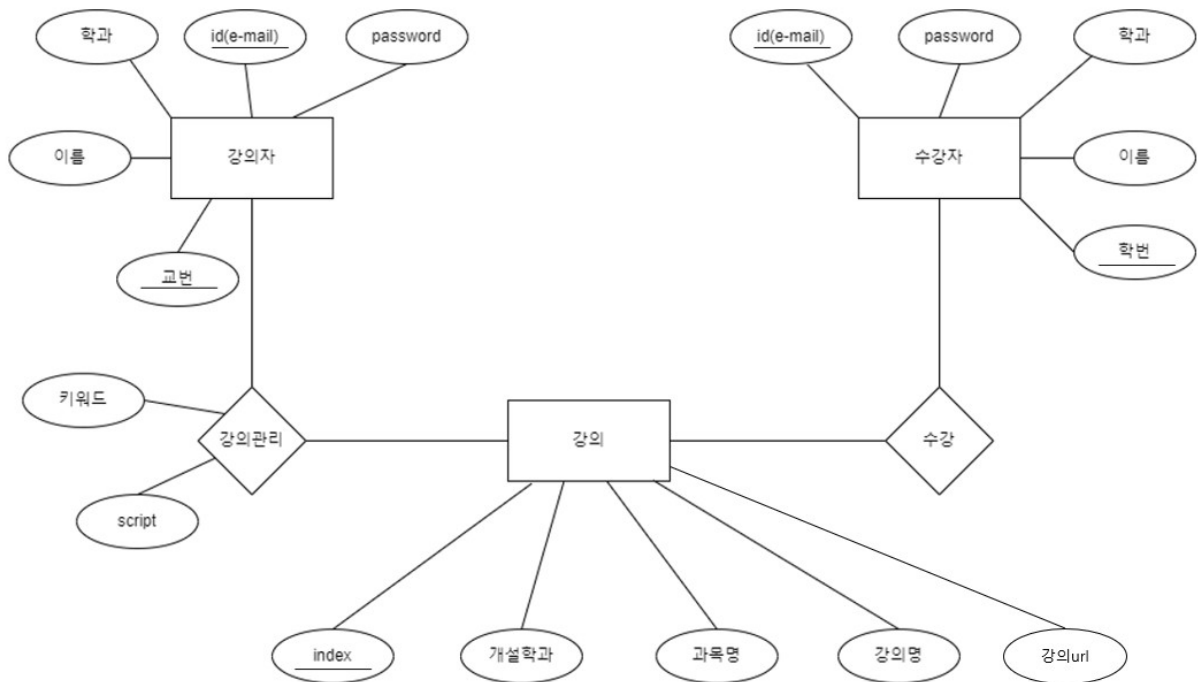
유스 케이스



클래스 다이어그램



E-R 다이어그램



테스트 계획서

NO	대분류	중분류	소분류	테스트 사항	Pass Result	비고
1	Front-end & Back-end	웹 페이지	메인화면	메인화면 접속 시 등록된 강의목록이 정상적으로 출력되는지 확인	디자인한 웹 페이지 동작	
2			로그인	로그인 화면 접속 후 id, pw 입력 시 로그인이 정상적으로 되는지 확인	로그인 성공, 실패	
3			사용자 강의목록	로그인 후 내 강의목록 페이지 접속 시 전체 강의목록이 정확하게 표시되는지 확인	사용자 강의 목록화	
4			강의 대시보드	강의 대시보드 접속 시 사용자가 등록한 강의에 대한 정보가 정상 출력되는지 확인	수강한 강의 정보 표시	
5			강의 관리자 페이지	동영상 업로드와 키워드, 스크립트 입력이 정상적으로 되는지 확인	동영상, 키워드, 스크립트 저장	
6		강의 동영상 페이지	동영상	강사가 업로드한 동영상, 링크 등이 동영상 플레이어에서 정상적으로 동작하는지 확인	동영상 재생	
7			챗봇UI	사용자 질문이 서버로 정상적으로 전달되어 답변을 출력하는지 확인	질문 입력, 답변 출력	

NO	대분류	중분류	소분류	테스트 사항	Pass Result
1	챗봇	질문 분석기	문장 tokenize	문장을 tokenize할 때 검색 가능한 의미 단위로 분할하는지 확인	위키피디아에서 검색 가능한 키워드 생성
2			임베딩 vocab 조회	임베딩 vocab을 조회해서 키워드에 대한 임베딩이 있는지 정상적으로 검색하는지 확인	임베딩 파일이 있는 키워드 추출
3		문서 검색기	동음이의어 처리	동음이의어에 대한 처리를 통해 관련된 키워드를 정확하게 추출하는지 확인	복수의 동음이의어에서 단어 선정
4			위키피디아 검색	키워드를 위키피디아에서 정상적으로 검색을 할 수 있는지 확인	키워드에 대한 위키피디아 문서
5			문단 분할	가져온 문서를 소주제, 문단에 따라 분할하는 지 확인	문서 소주제, 문단 단위로 분리
6		언어 모델	임베딩 길이 제한	토큰의 수가 모델에서 처리 가능한 수를 넘지 않는지 확인	한 번에 모델에 입력되는 토큰 수를 512개 이하로 제한
7			임베딩 생성	언어 모델에서 사용자 질문과 문서에 대한 임베딩을 정상적으로 생성하는지 확인	임베딩 생성
8		기계독해	답변 영역 예측	질문 임베딩, 문서 임베딩을 통해서 답변 영역을 찾을 수 있는지 확인	문서 내 답변 영역의 시작, 끝 토큰 인덱스 예측
9			답변 길이 제한	답변을 챗봇 UI에 출력했을 때 가독성을 유지할 수 있는	답변 길이 제한

				길이인지 확인	
--	--	--	--	---------	--

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

	문제해결을 위해 적용한 방법(또는 기법) 결과, (문제점, 해결방안)
	- 애자일 기법을 통한 유연하고 잦은 회의를 통해 속도감 있게 개발한다. 주 1회 회의를 통한 피드백과 개발
문제점	- 코로나19 감염증으로 인한 컨디션 저하로 인해 개발의 속도가 늦어졌다. 주 1회 회의를 하지 못하고 연락이 뜸해지는 상황. 타 과목 시험 기간으로 인해 캡스톤에 집중할 수 없어 개발의 공백.
해결방안	- 집중이 덜 되더라도 화상회의를 통한 피드백, 잦은 화상회의를 통한 서로 동기부여.

	데이터베이스 설계의 문제
문제점	- 데이터베이스의 설계에서 E/R 다이어그램 작도에서 강의자와 수강자, 강의 Entity set 사이에 루프가 생겨 redundancy의 원인이 될 가능성이 생김,
해결방안	- 집중이 덜 되더라도 화상회의를 통한 피드백, 잦은 화상회의를 통한 서로 동기부여.

문제점	- 따라서 이를 해결하기 위해 강의자와 수강자를 잇는 직접적인 relation을 삭제하고, 강의 Entity set를 사이에 둔 relation으로 강의자와 수강자의 relation을 표현해 루프, redundancy 발생을 예방함.
-----	--

문제점	- BERT 학습에 사용되는 KorQuAD 1.0은 답변 형태가 단답형으로 문장 형식의 답변이 필요한 질문에 답변하기 어려운 문제점이 있기 때문에 이를 개선하기 위해서 추가적인 데이터셋을 구축 중이지만 성능에 영향을 줄 정도의 데이터 수를 구축하는 것에 어려움이 있다.
문제점	- BERT에 입력되는 문서의 길이가 길수록 성능이 떨어지는 문제점이 발생하는데 이를 개선하기 위해서 문서 입력 길이와 관련된 하이퍼파라미터인 stride를 조정하고 있으며, 질문에 대한 답변 일 확률이 높은 n개의 예측 답변을 BERT에 재입력하여 답변을 재예측하는 방법을 고려하고 있다.

프로젝트명 : AI 챗봇을 활용한 온라인 강의 웹 서비스

소프트웨어 요구사항 정의서

Version 1.0

개발 팀원 명(팀리더):박상현
주준하
남승완

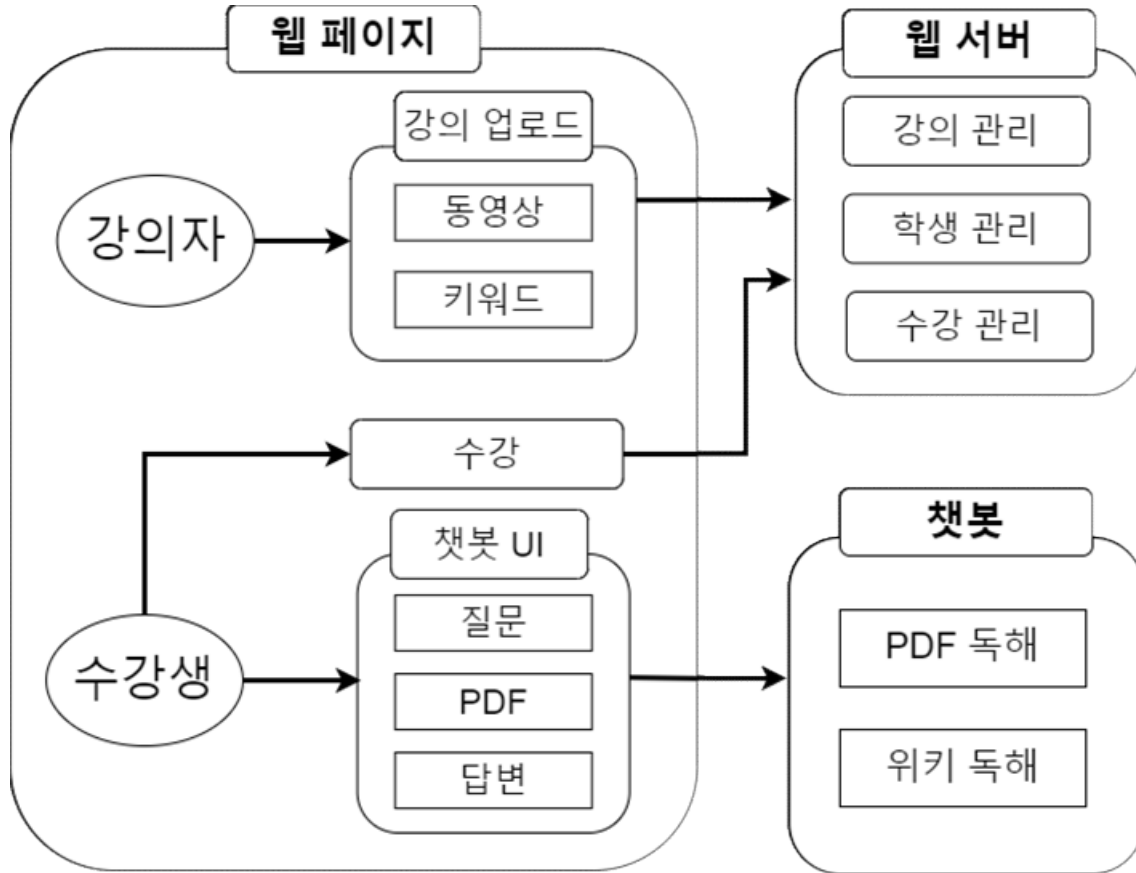
대표 연락처:010-5212-8903
e-mail: 20171612@edu.hanbat.ac.kr

목차

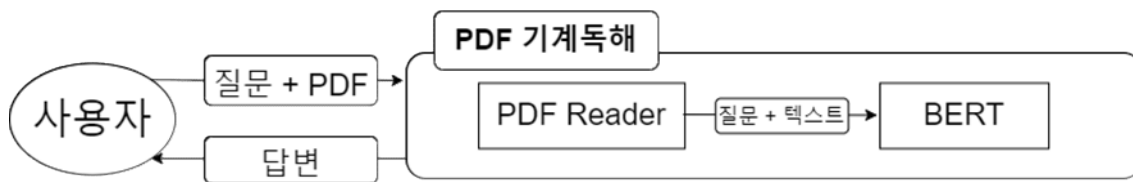
1. 개요
2. 시스템 장비 구성요구사항
3. 기능 요구사항
4. 성능 요구사항
5. 인터페이스 요구사항
6. 데이터 요구사항
7. 테스트 요구사항
8. 보안 요구사항
9. 품질 요구사항
10. 제약 사항
11. 프로젝트 관리 요구사항

1. 시스템 개요

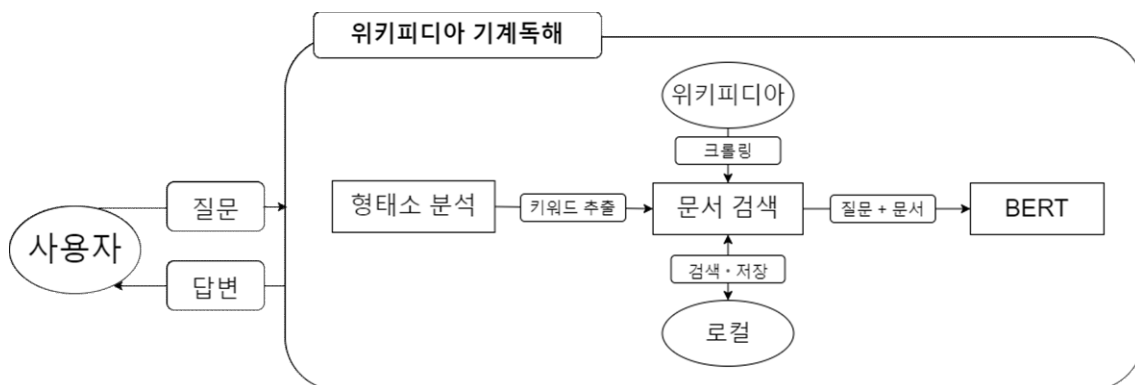
[시스템 기능도]



[강의자료 MRC]



[위키피디아 MRC]



2. 시스템 장비 구성요구사항

요구사항 고유번호		ECR-001		
요구사항 명칭		웹 개발장비		
요구사항 분류		시스템 장비구성 요구사항	응락수준	필수
요구사항 상세설명	정의	웹 어플리케이션		
	세부내용	<ul style="list-style-type: none"> - 장비 품목 : Asus Laptop - 장비 수량 : 1개 - 장비 기능 : 챗봇 기능과 동영상 플레이어. - 장비 성능 및 특징 : 사용자는 챗봇을 이용하여 질의응답 서비스를 이용하며 온라인 강의 수강할 수 있는 UI를 개발. 관리자는 챗봇에 이용할 수 있는 키워드를 제공하고 사용자에게 온라인 강의를 제공할 수 있도록 관리자 페이지 개발 		

요구사항 고유번호		ECR-002		
요구사항 명칭		GPU		
요구사항 분류		시스템 장비구성 요구사항	응락수준	필수
요구사항 상세설명	정의	언어모델(BERT) 학습 및 fine tuning을 위한 GPU		
	세부내용	<ul style="list-style-type: none"> - 장비 품목 : NVIDIA Tesla P100 GPU - 장비 수량 : 1개 - 장비 기능 : 언어모델 학습 및 fine tuning을 위한 GPU - 장비 성능 및 특징: google colab pro에서 제공하는 GPU mezzanine(NVLink): 3584 * 1328/1480 MHz, 9519-10609 GFLOPS, 16G memory 		

요구사항 고유번호		ECR-003		
요구사항 명칭		챗봇 서버		
요구사항 분류		시스템 장비구성 요구사항	응락수준	필수
요구사항 상세설명	정의	질의응답 서비스 제공 서버		
	세부내용	<ul style="list-style-type: none"> - 장비 품목 : 노트북 - 장비 수량 : 1개 - 장비 기능 : 질의응답 서비스 제공 - 장비 성능 및 특징: 질의응답 서비스 제공을 위한 python 패키지 설치 필요(websocket, pytorch, transformers) 		

요구사항 고유번호		ECR-004		
요구사항 명칭		웹 운영 서버		
요구사항 분류		시스템 장비구성 요구사항	응락수준	필수
요구사항 상세설명	정의	- 웹 서비스 제공		
	세부내용	<ul style="list-style-type: none"> - 장비 품목 : - 장비 수량 : 1 - 장비 기능 : 강의동영상 재생 플랫폼, 데이터 로직 처리 - 장비 성능 및 특징: <ol style="list-style-type: none"> 1. 클라이언트로부터 요청받은 로그인 또는 회원가입 정보를 받아서 DB에 저장 또는 로그인에 대해 성공여부 보냄 2. 로그인 여부에 따라 웹서비스 이용 3. 질의응답 모델을 호출하여 사용 가능해야함. 4. 사용자에게 보여지는 프론트부분과 로직을 처리하는 백엔드가 분리되어 존재. 		

요구사항 고유번호		ECR-004		
요구사항 명칭		DataBase		
요구사항 분류		시스템 장비구성 요구사항	응락수준	필수
요구사항 상세설명	정의	- 데이터저장 및 관리		
	세부내용	<ul style="list-style-type: none"> - 장비 품목 : - 장비 수량 : 1 - 장비 기능 : 회원정보 저장, 조회, 수정 - 장비 성능 및 특징: <p>사용자 정보 저장, 요청에 따른 조회 또는 수정</p>		

3. 기능 요구사항

요구사항 고유번호		SFR-001		
요구사항 명칭		형태소 분석기		
요구사항 분류		기능 요구사항	응락수준	필수
요구사항 상세설명	정의	사용자 질문 분석, 주요 키워드 추출		
	세부내용	1. 위키피디아 기계독해에서 활용 2. 사용자 질문 분석 및 키워드(일반, 고유 명사) 추출 3. 핵심적인 단어 몇 가지를 선정 4. 추출한 단어를 문서 검색기로 전달		

요구사항 고유번호		SFR-002		
요구사항 명칭		문서 검색기		
요구사항 분류		기능 요구사항	응락수준	필수
요구사항 상세설명	정의	wikipedia에서 키워드에 대한 정보를 검색		
	세부내용	1. 동음이의어에 의한 복수 검색 키워드에서 필요한 키워드 선정 2. 키워드에 대한 문서 검색 3. 문서를 문단 단위로 분할 4. 문단 단위로 언어모델에 입력		

요구사항 고유번호		SFR-003		
요구사항 명칭		언어모델		
요구사항 분류		기능 요구사항	응락수준	필수
요구사항 상세설명	정의	BERT 기반의 자연어 임베딩 생성 모델		
	세부내용	1. 입력된 자연어를 토큰화하고 임베딩을 생성 2. 모델로 구글의 다국어 BERT 모델을 KorQuAD 1.0과 추가로 제작한 데이터 셋에 fine tuning하여 성능 향상		

요구사항 고유번호		SFR-004		
요구사항 명칭		기계독해		
요구사항 분류		기능 요구사항	응락수준	필수
요구사항 상세설명	정의	사용자 질문에 대한 임베딩과 문단 임베딩을 이용한 답변 영역 예측		
	세부내용	1. FFNN, softmax를 이용한 문단 내 답변의 시작과 종료 영역 결정 2. 시작 임베딩 토큰, 종료 임베딩 토큰의 인덱스를 이용해서 자연어 답변영역 추출		

요구사항 고유번호		SFR-005		
요구사항 명칭		메인 페이지		
요구사항 분류		기능	응락수준	필수
요구사항 상세설명	정의	웹 사이트의 메인 페이지.		
	세부내용	- 사이트에 등록된 강의목록과 최근 학습 강의 확인. - 로그인 비활성화시 nav의 전체강의 버튼을 제외한 모든 버튼 클릭 시 로그인 창으로 이동 - 관리자로 인증되지 않은 일반 사용자는 관리자페이지 비활성화 안내.		

요구사항 고유번호		SFR-006		
요구사항 명칭		로그인 관리		
요구사항 분류		기능	응락수준	선택
요구사항 상세설명	정의	인증된 사용자에게 한하여 로그인.		
	세부내용	<ul style="list-style-type: none"> - 학교계정의 사용자에게 한하여 회원가입 가능. @edu.hanbat.ac.kr로 끝나는 이메일을 통해 인증 - 자동 로그인 사용 체크하면 자동으로 로그인 가능. 쿠키를 통해 관리 쿠키 삭제 시 자동 로그인이 되지 않음. - 로그인을 하지 않고 메인 강의목록 페이지로 이동 가능. - email, password 형식의 로그인 		

요구사항 고유번호		SFR-007		
요구사항 명칭		내 강의목록 페이지		
요구사항 분류		기능	응락수준	필수
요구사항 상세설명	정의	전체 강의목록 확인 가능		
	세부내용	<ul style="list-style-type: none"> - 로그인된 사용자가 등록한 강의 목록확인 - 강의목록 section 클릭 시 강의 창으로 이동. - 강의의 진도율 표시 - 로그인되지 않은 사용자가 접근 시 로그인 페이지로 이동. - 스크롤을 통해 강의 목록 확인. 		

요구사항 고유번호		SFR-008		
요구사항 명칭		강의 대시보드 페이지		
요구사항 분류		기능	응락수준	필수
요구사항 상세설명	정의	강의 대시보드를 통해 강의 학습		
	세부내용	- 로그인된 사용자는 자신이 등록한 강의 대시보드를 확인 할 수 있다.		

요구사항 고유번호		SFR-09		
요구사항 명칭		강의 동영상 페이지		
요구사항 분류		기능	응락수준	필수
요구사항 상세설명	정의	동영상으로 이루어진 강의를 수강할 수 있고 챗봇을 이용할 수 있다.		
	세부내용	<ul style="list-style-type: none"> - 동영상 영역과 챗봇 채팅 영역으로 이루어져 있다. - 챗봇 채팅 영역은 오른쪽 nav를 통해 강의 목차로 바꿀 수 있다. - 챗봇 채팅은 강의 목차로 바뀌어도 기록이 저장된다. - 동영상은 유튜브 플레이어를 통해 재생된다. - 강의 목록은 스크롤을 통해 볼 수 있다. 		

요구사항 고유번호		SFR-010		
요구사항 명칭		챗봇 채팅		
요구사항 분류		기능	응락수준	필수
요구사항 상세설명	정의	챗봇을 통해 사용자는 질의응답을 할 수 있다.		
	세부내용	<ul style="list-style-type: none"> - 텍스트를 통해 챗봇에게 질문을 할 수 있으며 챗봇은 적절한 답변을 전달한다. - 챗봇은 부적절한 사용자의 질문이 있을 시 사용자에게 응답이 어려움을 전달한다. 		

요구사항 고유번호		SFR-011		
요구사항 명칭		강의 관리자 페이지		
요구사항 분류		기능	응락수준	필수
요구사항 상세설명	정의	등록된 강의 관리자 전용 강의 관리.		
	세부내용	<ul style="list-style-type: none"> - 관리자는 동영상 링크 또는 파일을 통해 업로드. - 강의에 관련한 키워드 입력. - 강의 관련 내용을 작성. 		

요구사항 고유번호		SFR-012		
요구사항 명칭		회원 데이터 처리		
요구사항 분류		기능 요구사항	응락수준	필수
요구사항 상세설명	정의	- 회원가입		
	세부내용	<ul style="list-style-type: none"> - 회원가입을 요구하는 사용자에게 사용자가 정한 id와 password를 등록 - 회원은 학부생과 강사로 분류. - DB에 암호화하여 저장 		

요구사항 고유번호		SFR-013		
요구사항 명칭		강사 데이터 처리		
요구사항 분류		기능 요구사항	응락수준	필수
요구사항 상세설명	정의	- 키워드 등록		
	세부내용	<ul style="list-style-type: none"> - 강사로 분류된 회원은 자신의 강의 핵심 주제 키워드를 등록. - 강사로 분류된 회원은 자신의 유튜브 영상을 플로팅하여 한 강의의 목차로 업로드 가능. 		

4. 성능 요구사항

요구사항 고유번호		PER-001		
요구사항 명칭		챗봇 서버 동시 접속 수		
요구사항 분류		성능	응락수준	필수
요구사항 상세 설명	정의	- 챗봇 서버에 대한 동시 접속 제어		
	세부 내용	- 다수 사용자를 위한 멀티 스레드 기반의 웹 소켓 서버 - 일정 시간 미사용 세션 종료		
요구사항 고유번호		PER-002		
요구사항 명칭		질문에 대한 응답 시간		
요구사항 분류		성능	응락수준	필수
요구사항 상세 설명	정의	챗봇 응답 시간		
	세부 내용	- 사용자 질문을 복수의 챗봇에 스케줄링하여 응답 시간 단축 - 문서 검색 시간 : 1초 이내 - 문서 임베딩 생성 시간 : 단문 기준 3초 이내(문서 길이 비례) - 답변 예측 시간 : 2초 이내		
요구사항 고유번호		PER-003		
요구사항 명칭		언어모델 fine tuning		
요구사항 분류		성능	응락수준	필수
요구사항 상세 설명	정의	언어모델을 질의응답 데이터셋으로 fine tuning할 때 소요되는 자원		
	세부 내용	- gpu 메모리는 12GB 이상 필요하고 60,407개 질의응답 쌍(36MB 분량)의 데이터셋을 학습시키는데 5시간 이상 소요(GPU 성능에 따라 차이 존재)		

5. 인터페이스 요구사항

요구사항 고유번호		SIR-001	
요구사항 명칭	웹 구성		
요구사항 분류	인터페이스 요구사항	응락수준	필수
요구사항 상세설명	정의	웹 사이트의 메인 페이지.	
	세부 내용	- nav 영역에는 전체강의버튼, 내강의버튼, 관리자버튼, 최근강의버튼, 내정보버튼으로 이루어짐 - nav 컴포넌트를 재사용. - 로그인 상태가 아닐 때 회원만 이용가능한 항목은 로그인이나 회원가입을 유도. - 강의목록 section 클릭 시 강의 대시보드로 이동.	
주석	•		
요구사항 출처	•		

요구사항 고유번호		SIR-002	
요구사항 명칭	웹 디자인		
요구사항 분류	인터페이스 요구사항	응락수준	필수
요구사항 상세설명	정의	사용자 편의를 위한 웹 디자인	
	세부 내용	- 상하 스크롤을 통해 화면 이동 - 사용자 피로도 최소화와 가독성을 고려한 색상과 글씨 크기를 구성 - 버튼과 텍스트의 구분을 확실하게 하여 사용자가 사용하기 직관적이게 설계	
주석	•		
요구사항 출처	•		

요구사항 고유번호		SIR-003	
요구사항 명칭	메인 페이지		
요구사항 분류	인터페이스 요구사항	응락수준	필수
요구사항 상세설명	정의	웹 사이트의 메인 페이지.	
	세부 내용	- main 영역은 전체강의 목록영역과 최근 학습 강의목록 영역으로 나누어짐 - 전체 강의 목록영역과 최근 학습 강의목록 영역은 최대 3개의 강의를 보여주며 강의 이미지, 제목, 설명으로 하나의 section을 이룬다.	
주석	•		
요구사항 출처	•		

요구사항 고유번호		SIR-004	
요구사항 명칭	강의 대시보드 구성		
요구사항 분류	인터페이스 요구사항	응락수준	필수
요구사항 상세설명	정의	사용자 강의 대시보드 구성	
	세부 내용	- 강의 진도율과 강의 커리큘럼을 확인할 수 있다. - 강의 목차를 클릭하여 수강할 수 있다. - 전에 학습했던 구간을 이어서 수강할 수 있다.	
주석	•		
요구사항 출처	•		

요구사항 고유번호		SIR-005	
요구사항 명칭	강의 업로드		
요구사항 분류	인터페이스	응락수준	필수
요구사항 상세설명	정의	강의 동영상상을 웹 서버 데이터베이스에 저장하기 위한 인터페이스	
	세부 내용	- 동영상 파일 또는 유튜브 동영상 링크 업로드에 대한 인터페이스 - 강의 주제, 핵심 키워드 저장에 대한 인터페이스 - 강의 대본(스크립트) 저장에 대한 인터페이스	
주석	강의 대본 업로드는 선택 사항이나 챗봇의 답변 영역을 강의 내 발화 내용까지 확장할 수 있음		
요구사항 출처	•		

요구사항 고유번호		SIR-006		
요구사항 명칭		PDF 기계독해 인터페이스		
요구사항 분류		인터페이스	응락수준	필수
요구사항 상세 설명	정의	사용자가 공지사항, 강의자료 내용에 대한 질의응답을 수행하기 위한 인터페이스		
	세부 내용	<ul style="list-style-type: none"> - 사용자는 강의 동영상 페이지 상의 챗봇 UI를 통해서 질의응답 수행 - 사용자는 질문과 강의명을 서버로 전달하며, 서버상에서 해당 강의와 연결된 PDF 자료들을 사용해서 기계독해 수행 		
산출정보		사용자 질문에 대한 답변		

요구사항 고유번호		SIR-007		
요구사항 명칭		위키피디아 기계독해 인터페이스		
요구사항 분류		인터페이스	응락수준	필수
요구사항 상세 설명	정의	사용자 질문에 대한 답변을 위키피디아에서 검색하기 위한 인터페이스		
	세부 내용	<ul style="list-style-type: none"> - 사용자는 강의 동영상 페이지 상의 챗봇 UI를 통해서 질의응답 수행 - 사용자는 질문과 강의명을 서버로 전달하며, 서버상에서 해당 강의와 연결된 PDF 자료가 없을 경우 위키피디아 문서에 대해서 기계독해 수행 		
산출정보		사용자 질문에 대한 답변		

요구사항 고유번호		SIR-008		
요구사항 명칭		PDF 리더와 BERT의 연계		
요구사항 분류		인터페이스	응락수준	필수
요구사항 상세 설명	정의	PDF 리더와 BERT의 인터페이스		
	세부 내용	<ul style="list-style-type: none"> - PDF 리더를 통해서 추출한 텍스트를 사용자 질문과 함께 BERT에 입력 - BERT로 입력되는 토큰의 수는 최대 512개로 제한 		
산출정보		PDF 기계독해 예측 결과		

요구사항 고유번호		SIR-009		
요구사항 명칭		형태소 분석기와 문서 검색기의 연계		
요구사항 분류		인터페이스	응락수준	필수
요구사항 상세 설명	정의	형태소 분석기와 문서 검색 기간의 인터페이스		
	세부 내용	<ul style="list-style-type: none"> - 형태소 분석기에서 추출한 사용자 질문에 대한 키워드(명사, 고유명사, 외래어 등)를 문서 검색기에 전달하여 검색 - 검색은 로컬 저장소를 우선 탐색 후 없을 경우 위키피디아에서 키워드를 검색 후 로컬에 저장 		
산출정보		추출 키워드에 대한 위키피디아 문서		

요구사항 고유번호		SIR-010		
요구사항 명칭		문서 검색기와 BERT의 연계		
요구사항 분류		인터페이스	응락수준	필수
요구사항 상세 설명	정의	문서 검색기와 BERT간의 인터페이스		
	세부 내용	<ul style="list-style-type: none"> - 사용자 질문과 위키피디아 문서를 BERT로 전달하기 위한 인터페이스 - 질문과 문서에 대한 임베딩을 생성하고 기계독해를 수행 - BERT로 입력되는 토큰의 수는 최대 512개로 제한 		
산출정보		위키피디아 기계독해 예측 결과		

요구사항 고유번호		SIR-011		
요구사항 명칭		BERT와 챗봇UI의 연계		
요구사항 분류		인터페이스	응락수준	필수
요구사항 상세 설명	정의	BERT와 챗봇UI간의 인터페이스		
	세부 내용	<ul style="list-style-type: none"> - 예측한 답변을 사용자에게 전달하기 위한 인터페이스 - 챗봇 UI에서 답변의 가독성을 위한 길이 조절 또는 출력 형식 변화 필요 		
산출정보		챗봇UI에 답변 출력		

6. 데이터 요구사항

요구사항 고유번호	DAR-001		
요구사항 명칭	초기자료 구축		
요구사항 분류	데이터	응락수준	필수
요구사항 세부내용	<ul style="list-style-type: none"> - 강의 동영상상을 업로드하고 분야에 맞게 분류 - BERT를 fine tuning 질의응답 테스트에 맞춰 fine tuning하기 위해서 질문-답변 쌍의 데이터 구축(KorQuAD로 대체) - 강의자료 형식에 데이터 셋 약 100개 이상 구축 		

요구사항 고유번호	DAR-002		
요구사항 명칭	언어모델 vocab		
요구사항 분류	데이터	응락수준	필수
요구사항 세부내용	<ul style="list-style-type: none"> - 언어모델에서 문장 tokenize 및 인덱스 부여에 사용하는 파일 - 언어모델을 교체하지 않는 한 변동되지 않음. 		

요구사항 고유번호	DAR-003		
요구사항 명칭	데이터베이스 관리		
요구사항 분류	데이터	응락수준	필수
요구사항 세부내용	<ul style="list-style-type: none"> - 테이블은 유저 정보 테이블, 강의명과 사용자 질문의 쌍 테이블로 설정 - redundancy를 최소화해야 함. - null값이 없도록 해야 함. 		

7. 테스트 요구사항

요구사항 고유번호		TER-001		
요구사항 명칭		BERT 성능 테스트		
요구사항 분류		테스트	응락수준	필수
요구사항 세부내용		- 복수의 모델에 대해서 동일한 검증 데이터셋을 준비하고 가장 높은 성능을 보이는 모델을 선택		
요구사항 고유번호		TER-002		
요구사항 명칭		임베딩 길이 테스트		
요구사항 분류		테스트	응락수준	필수
요구사항 세부내용		- 모델에서 한 번에 생성 가능한 토큰의 임베딩 수는 최대 512개로 모델에 입력하기 전에 길이를 적절하게 분할하는 로직이 정상 작동하는지 테스트		
요구사항 고유번호		TER-003		
요구사항 명칭		질문 분석기 성능 테스트		
요구사항 분류		테스트	응락수준	필수
요구사항 세부내용		- 질문 분석기가 키워드를 추출할 때 위키피디아에서 검색 가능한 형태로 추출하는 지에 대한 테스트		
요구사항 고유번호		TER-004		
요구사항 명칭		사용자 웹 페이지와 챗봇 서버 연결 테스트		
요구사항 분류		테스트	응락수준	필수
요구사항 세부내용		- 사용자가 챗봇 UI에 입력하는 데이터를 챗봇 서버로 정확하게 적절한 시간 내에 전달하는지와 다수의 사용자가 동시 접속했을 때 이를 효과적으로 처리할 수 있는지에 대한 성능 테스트		
요구사항 고유번호		TER-005		
요구사항 명칭		사용자 웹 페이지 회원 관리		
요구사항 분류		테스트	응락수준	필수
요구사항 세부내용		- 로그인 상태에 활성화된 페이지가 정상적으로 작동 하는지 테스트		
요구사항 고유번호		TER-006		
요구사항 명칭		강의 플레이어와 챗봇 채팅 테스트		
요구사항 분류		품질(기술관점)	응락수준	필수
요구사항 상세 설명	정의	다른 브라우저와 사용자 장치 환경에서 잘 작동하는지 테스트		
	세부 내용	- 사용자마다 사용하는 브라우저와 장치 환경이 다를 때 오류없이 정상적으로 작동하는지 테스트		

8. 보안 요구사항

요구사항 고유번호	SER-001		
요구사항 분류	응용 및 DB보안		
요구사항 분류	보안	응락수준	필수
요구사항 세부 내용	<ul style="list-style-type: none"> - 일반 사용자는 직접적으로 DB접근을 할 수 없음. - 사용자 개인정보(아이디, 이름, 학번, 비밀번호 등)는 소스코드에 직접 하드코딩 하지 않음. - 해싱을 통해 내부 관리자도 패스워드를 알아볼 수 없게 함. 		

요구사항 고유번호	SER-002		
요구사항 분류	웹페이지 보안		
요구사항 분류	보안	응락수준	필수
요구사항 세부 내용	<ul style="list-style-type: none"> - 편의를 위한 쿠키의 이용에서 쿠키는 사용자 개개인에 따라 구분되어야 함. - 쿠키의 이름에 개인정보가 들어가지 않도록 쿠키명을 무작위 또는 암호화하여 생성해야 함. 		

9. 품질 요구사항

요구사항 고유번호		QUR-001		
요구사항 명칭		챗봇의 답변 정확도		
요구사항 분류		품질(기술관점)	응락수준	필수
요구사항 상세 설명	정의	언어모델 성능과 기계독해기의 성능과 관련된 챗봇의 답변 정확도		
	세부 내용	<ul style="list-style-type: none"> - 답변 정확도가 0.9 이하라면 언어모델과 기계독해기 성능의 향상이 필요 - 사용자가 답변의 내용적 오류를 발견하지 못하거나 오인할 가능성이 있는 답변을 제시하는 경우 서비스 구조의 전반적인 검토가 필요 		
요구사항 고유번호		QUR-002		
요구사항 명칭		챗봇의 답변 속도		
요구사항 분류		품질(기술관점)	응락수준	필수
요구사항 상세 설명	정의	사용자가 질문을 한 시점부터 답변을 받을 때까지 걸리는 시간		
	세부 내용	<ul style="list-style-type: none"> - 답변을 생성할 때 가장 오래 걸리는 케이스는 질문 형태소 분석, 문서 검색, 답변 예측을 모두 포함할 때이며 속도 향상을 위해서 cpu, gpu 등의 성능 향상이 필요 - 복수의 챗봇에 사용자 질문을 스케줄링하여 시간 단축 		
요구사항 고유번호		QUR-003		
요구사항 명칭		챗봇 서버 오류 처리		
요구사항 분류		품질(기술관점)	응락수준	필수
요구사항 상세 설명	정의	챗봇 서버에서 발생하는 다양한 오류에 대한 처리		
	세부 내용	<ul style="list-style-type: none"> - 다수 사용자의 동시 접속에 의한 성능 저하, 오류에 대해서 적절한 오류 발생 메시지를 사용자에게 전달 - 답변 생성 과정에서 발생하는 오류에 대해서 복구가 불가능할 시 사용자에게 적절한 메시지를 전달 		
요구사항 고유번호		QUR-004		
요구사항 명칭		챗봇 UI의 가독성		
요구사항 분류		품질(기술관점)	응락수준	필수
요구사항 상세 설명	정의	챗봇 UI가 출력하는 답변의 가독성		
	세부 내용	<ul style="list-style-type: none"> - 답변이 긴 경우 가독성이 떨어지기 때문에 적절한 길이 조절 또는 출력 형태 수정이 필요 		
요구사항 고유번호		QUR-005		
요구사항 명칭		언어모델의 이식성		
요구사항 분류		품질(기술관점)	응락수준	필수
요구사항 상세 설명	정의	언어모델의 기존 서버와 다른 운영체제에서 동작하는지에 대한 여부		
	세부 내용	<ul style="list-style-type: none"> - 개발 완료 후 웹 서버와 통합 또는 서버 이전을 할 때 새로운 환경에서 정상작동을 하기 위해서 운용 운영체제와 호환되는 파이썬 패키지 설치가 필요 - pytorch, transformers, tensorflow 등 		

10. 제약 사항

요구사항 고유번호	COR-001		
요구사항 명칭	언어모델 변경 제약 사항		
요구사항 분류	제약사항	응락수준	필수
요구사항 세부내용	<ul style="list-style-type: none"> - 언어모델은 기본적으로 BERT 기반의 모델을 사용하며, 성능 향상, 안전성 등의 목적으로 교체할 때 tensorflow로 작성된 모델을 사용한다. 또한 문장 tokenize에 사용하는 방법을 언어모델에서 사용하는 방식과 일치 시켜야한다. 		

요구사항 고유번호	COR-002		
요구사항 명칭	질문 분석기 변경 제약 사항		
요구사항 분류	제약사항	응락수준	필수
요구사항 세부내용	<ul style="list-style-type: none"> - 질문 분석기의 정확도 향상을 위해서 사용되는 알고리즘, 도구 등을 변경할 때 선정되는 키워드의 품사는 명사, 고유명사가 되도록 한다. 		

요구사항 고유번호	COR-003		
요구사항 명칭	웹 클라이언트, 챗봇 서버 간 통신 데이터 제약 사항		
요구사항 분류	제약사항	응락수준	필수
요구사항 세부내용	<ul style="list-style-type: none"> - 클라이언트의 사용자와 챗봇 서버 간에 교환(질문, 답변)하는 데이터는 확장성을 위해서 Json을 사용한다. 		

요구사항 고유번호	COR-004		
요구사항 명칭	시스템 구조 설계		
요구사항 분류	제약사항	응락수준	필수
요구사항 세부내용	<ul style="list-style-type: none"> - 오류 수정, 성능 향상, 안정성 등을 목적으로 기존 모듈을 교체를 유연하게 하기 위해서 모듈 간 인터페이스를 독립적으로 구성하고 데이터 입출력 방식이 변경되지 않도록 설계한다. 		

11. 프로젝트 관리 요구사항

요구사항 고유번호	PMR-001		
요구사항 명칭	품질관리		
요구사항 분류	프로젝트 관리	응락수준	필수
요구사항 세부내용	<ul style="list-style-type: none"> - 웹서비스 개발과 챗봇 모델 개발로 나눈다. - 웹서비스는 UI개발, DB, 서버로 나뉜다. - UI는 프론트엔드가 맡으며 동작의 전반적인 구조를 설계해야 한다. - DB, 서버는 서비스 배포, 아키텍처 전반적인 인프라를 설계해야 한다. - 프로토타입을 공유하여 수정, 보완 사항을 의논한다. - 각 구성원들은 각자 맡은 분야에 대한 설계, 개발, 문서화를 진행하고 주기적인 검토와 회의를 통해서 전체적인 설계 내용을 구현하고 통합한다. 		