

Renux 회의록 1021-1111

Renux 회의록 1021

일시	10월 21일 18:00	팀장	조준용
장소	다향관 C3	팀원	신원철, 육심호

간단한 쿼리를 머신러닝으로 처리하도록 할 수 있는지 토의

제안발표 피드백 보완

- 학생회랑 실시간 채팅은 학생회 측에서 시간 여유가 없을 수도 있는데, 그러면 실시간성이 조금 떨어지는 것 아닐까요?
⇒
- AI 챗봇을 앱으로 발전할 생각이 있으신가요?
⇒ 네
- 재순위(reranking) 기법에 어떤 방식을 적용할 생각이신가요?
⇒
- 개인 메시지를 저장하는 과정에서 민감한 개인 정보 보호 방법이 무엇인지
⇒ 챗봇 시작시 개인정보를 입력하지 말라는 경고문을 출력한다. → 개인정보 입력하면 본인이 손해이므로 일반적으로 하지 않을 것
- 맞춤화 채팅을 위해서 지난 기록들을 저장해둔다면 참조할 데이터가 많아지면 지연 시간이 생길텐데 어떻게 해결할 건지
⇒ 이는
- 학생회 같은 숨겨진 정보는 학교 공지에 안 올라오는 정보라 크롤링이 불가능할 것 같은데 이러한 정보는 어떻게 수집,저장되는지
⇒

Renux 회의록 1023

일시	10월 23일 18:00	팀장	조준용
장소	비대면 - 웹엑스	팀원	신원철, 육심호

1. 백엔드 - 샘플 서버 배포 테스트

- docker 리눅스(ubuntu)에 nginx 설치 후 배포
- 포트포워딩까지 완료하여 외부 접속 테스트 성공

2. 프론트 - 리액트 공부 시작

3. RAG - 임베딩 모델

- 후보군 선정(Hugging face)
 - upskyy/bge-m3-korean
 - nlpai-lab/KURE-v1
 - jinaai/jina-embeddings-v3

차주 목표

- 조준용 : 리프레쉬 토큰
- 신원철 : 임베딩 모델 성능 비교
- 육심호 : 리액트 실습 및 UI 구현

Renux 회의록 1028

일시	10월 28일 18:00	팀장	조준용
장소	다향관 C3	팀원	신원철, 육심호

보고서 구성 방향

1. 작성 방식 변경

- 문장형 서술 방식으로 작성한다.

2. 내용 구성 순서

- 1번(배경):** 전체적인 상황 및 문제 인식의 배경 중심으로 간결하게 서술
- 2번(문제점):** 구체적인 문제 상황과 현 시스템의 한계를 명확히 기술
- 3번(목표):** 우리가 해결하고자 하는 핵심 목표를 명확히 제시
- 서비스 범위 및 운영원리는 뒤로 이동하고, 기대효과는 학생 입장, 직원 입장 등 이해관계자 관점으로 구체적으로 기술한다.

세부 구성 지침

- 문제 상황의 핵심 키워드를 먼저 제시하고, 필요성을 논리적으로 서술한다.
- 우리가 추구하는 서비스 범위를 명확히 정의하고, 이를 기준으로 유사 서비스(선행 기술 3~4개) 와 비교 분석한다.
- 해외 대학 챗봇 사례도 함께 조사하여 비교 분석할 것.
 - 가능한 한 가장 완성도 높은 서비스 사례를 선정.
 - 각 대학별 구체적 기능 설명 포함.
 - 참고문헌, 출처 표기로 신뢰성 확보.
- 선행기술 분석 표 하단에 사례 요약 추가.

개발 목표 및 기술 구성

- 개발 목표는 한 줄 요약(핵심 요구사항 중심)으로 제시.
- 이후에 개발 기술 및 구성 내용을 구체적으로 기술.
- 유즈케이스 다이어그램을 참고하여 다음 예시 구조로 작성.
 - 사용자(학생, 게스트) → 채팅방 관리 / 질의응답 이용**

- 관리자(학생회, 직원) → 데이터 관리 / 회원 관리
 - 시스템 다이어그램(컴포넌트 구조도) 포함.
 - 개발 목표 중 세부 실행 내용은 '개발 내용' 파트로 이동 (분량 조정 목적).
 - 기술 스택은 마지막 부분에 그림 형태로 정리.
-

추진 일정

- 간트차트 형태로 추진 일정 시각화.
- 주요 단계별 일정 구분.

Renux 회의록 1030

일시	10월 30일 18:00	팀장	조준용
장소	다향관 C3	팀원	신원철, 육심호

RAG - Embedding 모델

- 3가지 비교해서 'KURE-v1' 모델을 선정.
→ 임베딩을 진행.
 1. 질문이 들어왔을 때, 본문 내용까지 유사도를 측정 →
질문이 간단하면 본문과 조금 유사해도 장학 공지를 궁금했는데 일반 공지에 있는 장학 내용을 가져오는 등의 오류가 있어. →
이거를 해결하기 위해 공지 제목으로만 유사도 검증.
- LLM 모델은 OPEN AI API를 이용하는 것으로 결정.

백엔드 & 프론트엔드

- React library를 이용하여 구조 개편.
- Build → 결과물을 서버 폴더에 적용됨을 확인.

차주 계획

- AI
 - RAG 성능 개선
- 프론트
 - CSS 수정
 - 채팅방 생성 요청 보내기 구현
 - 채팅 전송 구현
- 백엔드
 - 채팅방 생성 구현
 - 채팅 수신 구현

Renux 회의록 1101

일시	11월 1일 18:00	팀장	조준용
장소	다향관 C3	팀원	신원철, 육심호

채팅 웹 서버와 IIm 서버 분리

- 임시 IIm 서버 구현 성공
- 요청 응답 확인

채팅방 생성 기능 구현

차주 계획

- 프론트 - 채팅 전송 기능 구현
- 백엔드 - 받은 채팅 내용을 IIm 서버에 전송하여 대담을 받은 후 클라이언트로 응답
- AI - 기존 구현 모델 기반으로 IIm

Renux 회의록 1104

일시	11월 4일 18:00	팀장	조준용
장소	다향관 C3	팀원	신원철, 육심호

중간보고서 작성

유스케이스, 시스템 블, 플로우 차트 등 이미지 개선 및 추가

선행 서비스 비교 이미지 캡처

Renux 회의록 1106

일시	11월 6일 18:00	팀장	조준용
장소	다향관 C3	팀원	신원철, 육심호

중간 보고서 구조 수정

과거 OSS 프로젝트 보고서 참조

피드백 반영하여 문장형으로 변경

Renux 회의록 1111

일시	11월 11일 18:00	팀장	조준용
장소	다향관 C3	팀원	신원철, 육심호

docker-compose.yml 파일로 컨테이너 자동화 성공

프론트 작업이 더욱 수월해질 것으로 전망됨

중간보고서 마무리

발표자료 제작 시작

최종 검토