

KTDS MS AI 개발역량 향상 과정 MVP 과제 발표

KT STB 개발 지원 Agent

발표자: 김수희



1. 프로젝트 개요

해결하고자 하는 문제

KT STB 개발 시 전용 규격 및 IPTV 가이드 등 방대한 문서로 인한 개발 속도 저하

- 수백 페이지 문서 검색의 어려움
- 문서 이해와 코드 적용까지의 높은 시간 비용

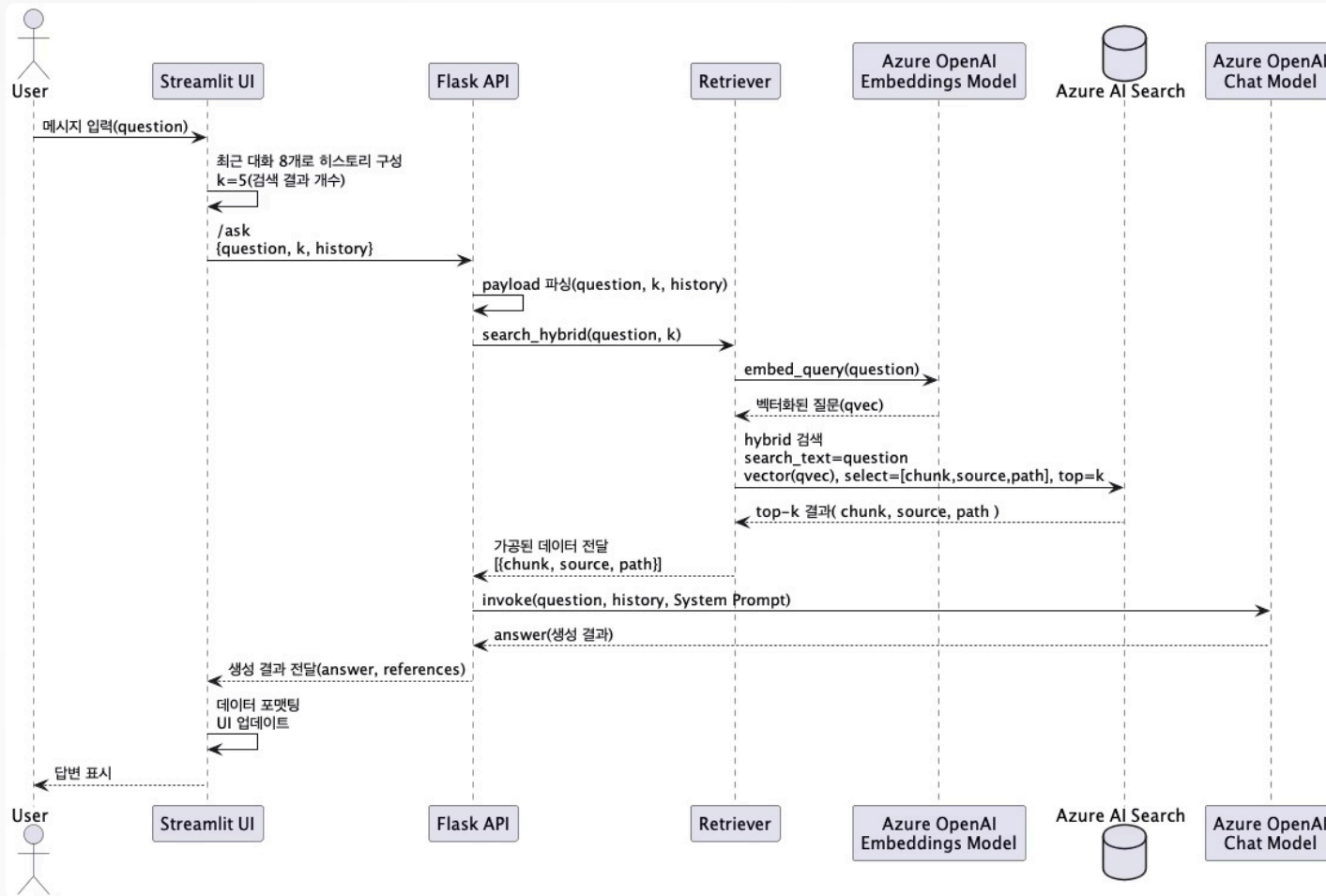
제안하는 솔루션

Azure OpenAI + Azure AI Search 기반 RAG 개발 지원 에이전트

- 자연어 질문으로 관련 규격 검색
- 정확한 요약과 샘플 코드 제공
- 출처 문서 링크로 즉시 접근

대상 사용자: KT STB 앱 개발자, PM, 아키텍트 등 관련 실무자

2. 시스템 아키텍처



Streamlit UI

1

세션 관리와 히스토리 유지, API 질의 전송 및 답변 렌더링

Flask API

2

REST API 수신 처리 및 App Service 관리 담당

Retriever & Search

3

쿼리 임베딩 후 하이브리드 검색으로 관련 문서 조회

Azure OpenAI

4

임베딩 모델과 Chat 모델로 의미 기반 검색 및 답변 생성

3. 핵심 기술 특징



문서 청킹 & 메타데이터 보존

인덱서/스킬셋 기반 페이지 분할
후 source, path 메타데이터 유지
→ 근거(출처) 추적이 용이함



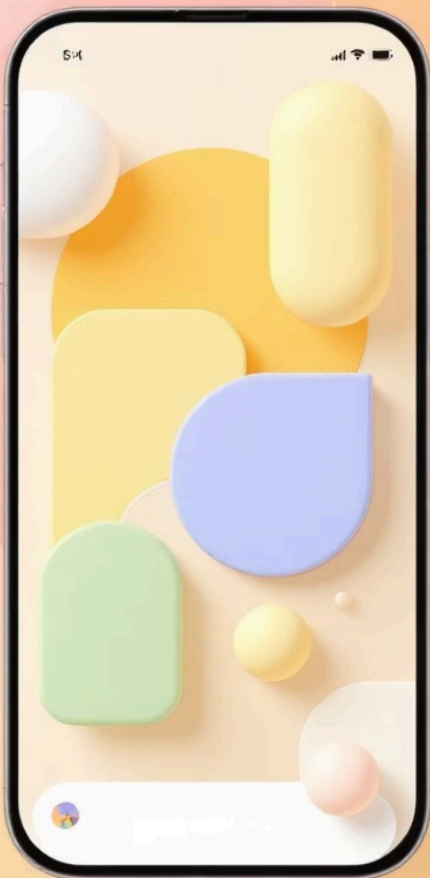
대화 맥락 유지

최근 8개 질의를 히스토리로 전달
하여 LLM이 맥락 기반 답변 생성
→ 후속 질문 정확도 대폭 향상



모듈형 아키텍처

Flask API ↔ Retriever ↔ LLM 호출을 모듈화하여 UI 레이어 분리
→ 다양한 UI 프레임워크로 확장 가능



4. 라이브 데모

Streamlit UI

직관적이고 반응형 인터페이스로 빠른
프로토타이핑에 최적화

데모 체험하기

Gradio UI

머신러닝 모델 시연에 특화된 사용자
친화적 인터페이스

데모 체험하기

📄 두 가지 UI 옵션을 제공하여 사용자 선호도와 용도에 따른 선택 가능

5. 향후 개선 및 확장 계획

이벤트 기반 자동 인덱싱

PDF 업로드 → Azure Functions(Blob Trigger) → 자동 청킹/임베딩 → AI Search 업데이트

- 신규/개정 문서 실시간 반영
- 수동 관리 작업 최소화

배포 자동화 (Azure CLI)

인프라와 앱 설정을 스크립트화하여 일관된 배포 환경 구축

- 개발-스테이징-운영 원클릭 배포
- 초기 환경 구성 시간 대폭 절감

플러그형 멀티 UI

백엔드 REST API 고정, UI 레이어 모듈화로 다양한 채널 확장

- Teams Bot, React, 사내 포털 연동
- 사용자 경험 최적화된 UI 선택