

사용자 이미지 기반 AI 챗봇 테스트 계획서

9조 허정우 김보배

주제: 사용자 이미지 기반 AI 챗봇

- 주요 기능들
 - 사용자가 자연어로 묻고 답하기
 - 이미지 업로드
 - 이미지 삭제

테스트 계획

- 테스트 요소: 주요 기능들
- 테스트 방법
 - 단위 테스트
 - 시스템 테스트
- 테스트 평가 기준
 - 단위 테스트 - 각 모듈이 정상적으로 동작하여 올바른 결과가 나오는지, 예외 처리가 가능한지 평가
 - 시스템 테스트 - 주요 기능 요구사항을 만족하는지 입력 값에 따른 출력 결과로 검증

단위 테스트

- 파이프라인을 구성하는 함수 별로 테스트
- 개발 환경에서 진행
- 각 함수의 input에 따라 적절한 값이나 결과가 나와야 한다
- 테스트용 데이터 사용
- 데이터 베이스는 미리 구축되어 있다

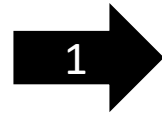
주요 파이프라인 3가지

1. 업로드한 이미지로부터 정보를 추출하여 그래프에 저장
 - 이미지에서 정보 추출
 - 정보를 그래프 데이터베이스 쿼리로 변환
 - 그래프 데이터베이스에 저장
2. 질의를 가지고 그래프를 검색하여 RAG방식으로 응답 생성
 - 자연어 질의 의도 파악
 - (RAG가 필요하다면) 데이터 베이스 쿼리 생성
 - 그래프 데이터 베이스 검색으로 정보 추출
 - LLM이 응답 생성
3. 삭제한 이미지에 대해 그래프에서도 삭제
 - 그래프 데이터베이스 쿼리 생성
 - 그래프 데이터베이스에서 정보 삭제

1. 업로드한 이미지로부터 정보를 추출하여 그래프에 저장

1. 이미지에서 정보 추출

- 데이터: 테스트용 이미지
- 결과: 메타데이터, 객체, 객체사이 관계 등의 정보가 생성



- Image_id: 1234
- metadata
 - 2025년 3월 23일 오후 12:36
 - 대전 유성구 문화원로
- 크림우동
- 그릇
- 크림우동이 그릇에 담김

1. 업로드한 이미지로부터 정보를 추출하여 그래프에 저장

2. 정보를 그래프 데이터베이스 쿼리로 변환

- 데이터: Json 형식의 정보 데이터
- 결과: 바로 실행가능한 쿼리

```
CREATE (img:Image {id: "1234"});  
CREATE (udon:Object {label: "크림우동"});  
CREATE (time:Time {datetime: "2025-03-23T12:36:00"});  
CREATE (img)-[:CONTAINS]->(udon);  
CREATE (img)-[:TAKEN_AT_TIME]->(time);  
...
```



- Image_id: 1234
- metadata
 - 2025년 3월 23일 오후 12:36
 - 대전 유성구 문화원로
- 크림우동
- 그릇
- 크림우동이 그릇에 담김

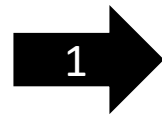


1. 업로드한 이미지로부터 정보를 추출하여 그래프에 저장

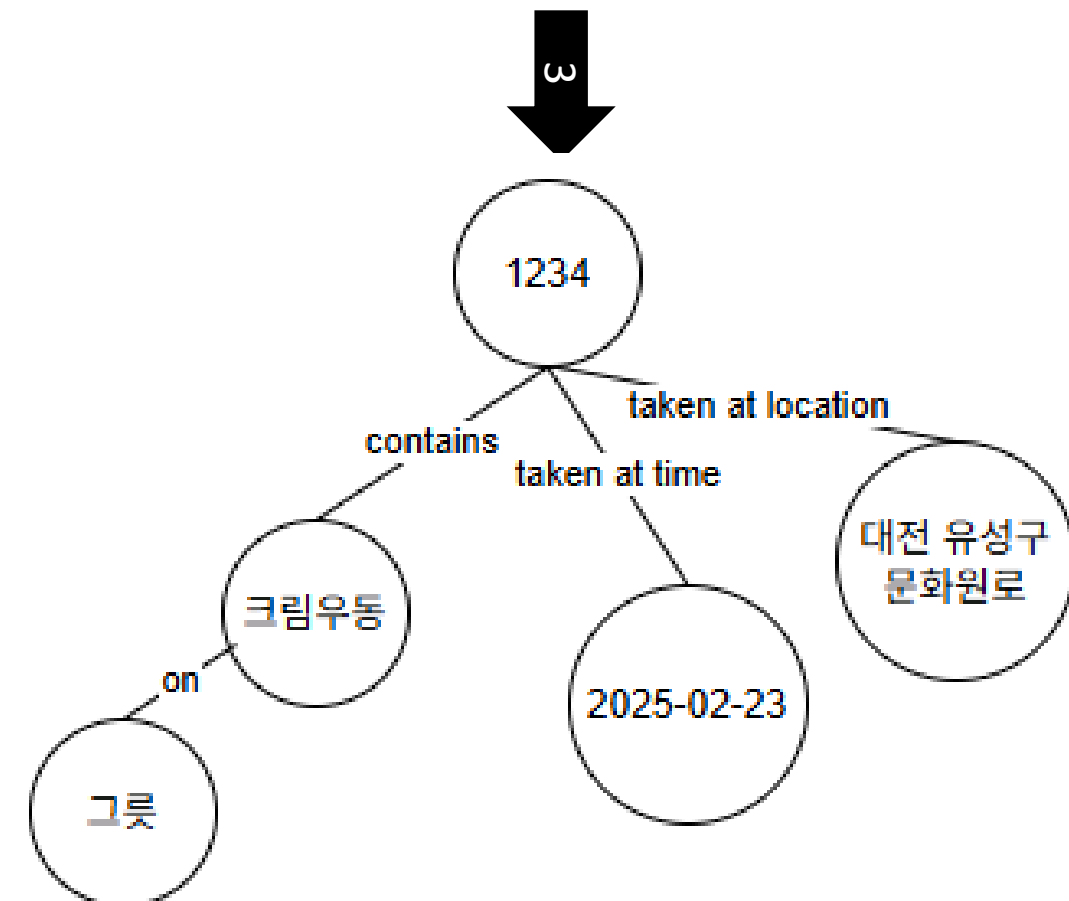
3. 그래프 데이터베이스에 저장

- 데이터: CREATE 쿼리
- 결과: 그래프 데이터베이스에 노드와 엣지가 생성

```
CREATE (img:Image {id: "1234"});  
CREATE (udon:Object {label: "크림우동"});  
CREATE (time:Time {datetime: "2025-03-23T12:36:00"});  
CREATE (img)-[:CONTAINS]->(udon);  
CREATE (img)-[:TAKEN_AT_TIME]->(time);  
...
```



- Image_id: 1234
- metadata
 - 2025년 3월 23일 오후 12:36
 - 대전 유성구 문화원로
- 크림우동
- 그릇
- 크림우동이 그릇에 담김



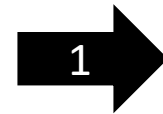
2. 질의를 가지고 그래프를 검색하여 RAG로 응답 생성

1. 자연어 질의 의도 파악

- 데이터: 자연어 질의
- 결과: RAG가 필요한지 판단

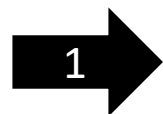
데이터베이스에 테스트용
정보가 있다고 가정하고 진행

질문: 내가 크림우동을 먹었는데 언제
어디서 먹었는지 기억이 안나



RAG가 필요함

질문: 오늘 날씨 좋다



RAG가 필요하지 않음

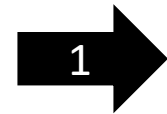
2. 질의를 가지고 그래프를 검색하여 RAG로 응답 생성

2. 데이터 베이스 쿼리를 생성

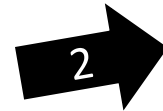
- 데이터: 자연어 질의
- 결과: 실행 가능한 데이터 베이스 MATCH 쿼리 출력

```
MATCH (img:Image)-[:CONTAINS]->(udon:Object {label: "크림우동"})  
OPTIONAL MATCH (img)-[:TAKEN_AT_TIME]->(time:Time)  
OPTIONAL MATCH (img)-[:TAKEN_AT_LOCATION]->(loc:Location)  
RETURN img.id, time.datetime AS when, loc.address AS where
```

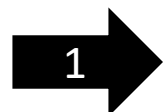
질문: 내가 크림우동을 먹었는데 언제
어디서 먹었는지 기억이 안나



RAG가 필요함



질문: 오늘 날씨 좋다



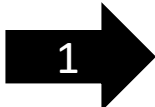
RAG가 필요하지 않음

2. 질의를 가지고 그래프를 검색하여 RAG로 응답 생성

3. 그래프 데이터 검색으로 정보 추출

- 데이터: MATCH 쿼리
- 결과: 검색 결과

질문: 내가 크림우동을 먹었는데 언제 어디서 먹었는지 기억이 안나



RAG가 필요함

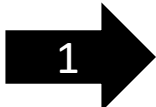


```
MATCH (img:Image)-[:CONTAINS]->(udon:Object {label: "크림우동"})
OPTIONAL MATCH (img)-[:TAKEN_AT_TIME]->(time:Time)
OPTIONAL MATCH (img)-[:TAKEN_AT_LOCATION]->(loc:Location)
RETURN img.id, time.datetime AS when, loc.address AS where
```



img.id	when	where
img_1234	2025-03-23T12:36:00	대전 유성구 문화원로

질문: 오늘 날씨 좋다



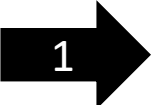
RAG가 필요하지 않음

2. 질의를 가지고 그래프를 검색하여 RAG로 응답 생성

4. LLM이 응답 생성

- 데이터: 검색된 결과
- 결과: 자연어 응답 출력

질문: 내가 크림우동을 먹었는데 언제 어디서 먹었는지 기억이 안나



RAG가 필요함



```
MATCH (img:Image)-[:CONTAINS]->(udon:Object {label: "크림우동"})
OPTIONAL MATCH (img)-[:TAKEN_AT_TIME]->(time:Time)
OPTIONAL MATCH (img)-[:TAKEN_AT_LOCATION]->(loc:Location)
RETURN img.id, time.datetime AS when, loc.address AS where
```

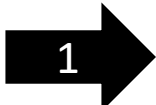


img.id	when	where
img_1234	2025-03-23T12:36:00	대전 유성구 문화원로



2025년 3월 23일에 유성구 문화원로에서 먹었어요!

질문: 오늘 날씨 좋다



RAG가 필요하지 않음



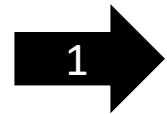
날씨가 좋네요~

3. 삭제한 이미지에 대해 그래프에서도 삭제

1. 그래프 데이터베이스 쿼리 생성

- 데이터: 삭제한 이미지의 아이디
- 결과: 바로 실행 가능한 Delete 쿼리로 출력

Id: 1234 삭제함



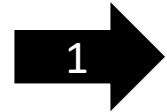
```
MATCH (img:Image {id: "1234"})-[r]->(n)
DETACH DELETE img, n
```

3. 삭제한 이미지에 대해 그래프에서도 삭제

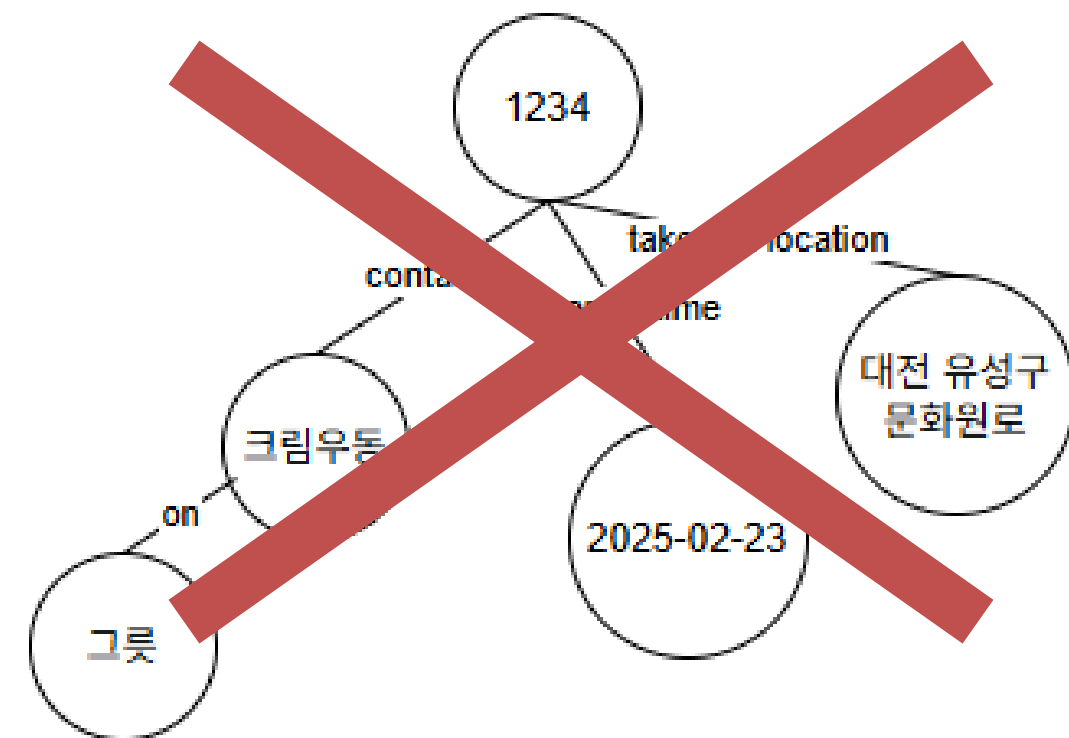
2. 그래프 데이터베이스에서 정보 삭제

- 데이터: Delete 쿼리
- 결과: 데이터베이스 상에서 사진과 관련된 노드와 엣지가 삭제

Id: 1234 삭제함



```
MATCH (img:Image {id: "1234"})-[r]->(n)  
DETACH DELETE img, n
```



시스템 테스트

- 주요 기능에 대해 테스트
 - 사용자가 이미지를 업로드하면 그래프에 저장하는 기능 (Upload)
 - 질의를 입력 받고 응답을 생성하는 기능 (Match)
 - 사용자가 이미지를 삭제하면 그래프에서 정보도 삭제하는 기능(Delete)
- 개발 완료 후 진행
- 입력 데이터 별 예상 결과를 명시하고 비교하는 방식으로 진행

1. 이미지를 업로드 하면 그래프에 저장하는 기능

1. 정상적인 이미지 파일 → 그래프에 노드와 엣지가 추가
2. 이미지 파일은 맞지만 아무 정보가 없는 배경이거나 심하게 흔들림
→ 오류 응답 생성
3. 올바른 이미지 파일 형식이 아님
→ 잘못된 이미지라는 오류 메시지

2. 질의를 입력 받고 응답을 생성하는 기능

1. 데이터 베이스에 존재하는 데이터에 대한 질의 → 정상적인 응답 생성
2. 데이터 베이스에 없는 데이터에 대한 질의(날짜 질문을 했는데 없거나 장소가 없거나) → 정보를 찾을 수 없다는 응답 생성
3. RAG가 필요하지 않은 일반 질의 → 정상적인 응답 생성
4. 질문이 이상한 경우 → 질의를 다시 입력하라는 응답 생성

3. 사용자가 이미지를 삭제하면 그래프에서 정보도 삭제하는 기능

1. 데이터 베이스에 존재하는 이미지 ID → 관련 노드와 엣지 삭제
2. 데이터 베이스에 없는 이미지 ID → 오류로 처리

감사합니다