

SK네트웍스 Family AI 과정 19기

모델링 및 평가 테스트 계획 및 결과 보고서

산출물 단계	모델링 및 평가
평가 산출물	테스트 계획 및 결과 보고서
작성 시작일	2026-01-12
제출일	2026-01-15
깃허브 경로	https://github.com/Poli-Cheetah/poli-cheetah
작성 팀원	오하원

1. 테스트 개요

- 테스트 기간: 2026-01-13 ~ 2026-01-14
- 목표: 본 문서는 시스템 테스트를 위한 계획과 테스트 결과를 보고합니다. 테스트의 목적은 시스템의 기능적 및 비기능적 요구 사항을 충족하는지 확인하고, 성능과 안정성을 검증하는 것입니다.

2. 테스트 목적

테스트의 주요 목적은 다음과 같습니다:

- 기능성 검증: 시스템이 요구된 기능을 제대로 수행하는지 확인합니다.
- 성능 검증: 시스템의 응답 시간, 처리 속도 등 성능 관련 요구 사항을 만족하는지 확인합니다.
- 안정성 및 에러 검증: 시스템이 예상치 못한 오류나 예외 상황에서 적절히 동작하는지 검증합니다.

3. 테스트 환경

3.1 하드웨어 환경

- 서버: HP ProBook 450 15.6 inch G10 Notebook PC
- CPU: 13th Gen Intel(R) Core(TM) i5-1340P(1.90 GHz)
- RAM: 32.0GB

- SSD: 475 GB (사용 가능 공간: 183 GB)

3.2 소프트웨어 환경

- 운영 체제: Windows 11 Pro 24H2(OS 빌드 26100.7462)
- 데이터베이스: AWS Lightsail
- 웹 서버: uvicorn 0.40.0
- 프로그래밍 언어: Python 3.11
- 테스팅 도구: JUnit, Selenium, Apache JMeter (~~후후 예정~~)

3.3 네트워크 환경

- 테스트 서버: 개발용 네트워크 내에서 테스트 진행

3.4 테스트 도구 및 소프트웨어

- ~~Selenium: UI 테스트 자동화~~
- ~~JUnit: 단위 테스트~~
- ~~JMeter: 성능 테스트~~

4. 테스트 케이스

4.1 기능 테스트 케이스

테스트 케이스 ID	테스트 항목	테스트 설명	예상 결과	실제 결과	비고
TC_001	로그인 가능	사용자 이메일과 비밀번호를 입력하여 로그인 시도	로그인 성공 후 대시보드 화면으로 이동		미구현
TC_002	회원가입 가능	모든 필수 필드를 입력하고 회원가입 버튼 클릭	회원가입 완료 후 확인 이메일 발송		미구현
TC_003	문서 업로드	사용자가 markdown 문서를 업로드할 수 있어야 한다.	문서 업로드 완료 출력	실패	재구현 중
TC_004	문서 섹션화	사용자가 전달한 문서를 분할한 후 유사한 문장들끼리 하나하의 섹션으로 만든다. 이미 비슷한 섹션이 존재하면 해당 섹션 안에 포함시킨다.	문서는 의미적으로 유사한 문장 기준으로 자동 분할되며 기존에 유사한 섹션이 있으면 해당 섹션에 포함된다.		미구현
TC_005	섹션 색인	새로 생성된 섹션의	생성된 각 섹션에는		미구현

	생성	요약내용을 분석하여 분류하기에 알맞은 색인을 생성한다.	요약 내용을 기반으로 한 분류용 색인이 정확히 생성된다.		
TC_006	섹션 original texts 저장	문서를 섹션화 했어도 바뀌지 않은 문서 원본을 복원할 수 있다.	섹션화 이후에도 원본 문서 데이터는 손실 없이 저장되어 언제든 복원 가능하다.	성공	구현
TC_007	문서 생성	사용자가 로컬에 새 문서를 생성할 수 있다.	로컬 저장소에 새 문서가 생성된다.	성공(보완필요)	구현
TC_008	문서 수정 (로컬)	사용자가 로컬에 있는 문서를 수정할 수 있다.	사용자가 수정한 단일 로컬 문서의 내용만 변경되며, 저장 후 해당 문서 조회 시 즉시 반영된다.	성공	구현
TC_009	패널 사이즈 조절	사용자가 마우스 드래그 앤 드롭으로 3개의 패널별 사이즈를 조절할 수 있다.	각 패널은 최소 조절 가능 크기가 정해져 있다. 최소 크기까지 마우스로 리사이징이 가능하다.	성공	구현
TC_010	문서 수정 (클라우드)	사용자가 클라우드에 있는 문서를 수정할 수 있다	사용자가 수정한 클라우드 문서의 섹션 변경 사항이 동일 섹션으로 연결된 다른 문서들에도 일괄 반영되어 저장되고, 각 문서 조회 시 즉시 확인된다.	실패	재구현 중
TC_011	문서 수정 되돌리기	수정 중이던 문서를 다시 처음 상태로 되돌릴 수 있다.	처음 문서에 아무 내용이 없었으면 빈 내용으로 돌아가고, 내용이 있었으면 그 내용으로 되돌린다.	성공	구현
TC_012	섹션 동기화	문서 수정할 시 해당 섹션에도 변경사항을 동기화한다.	문서 수정 내용은 관련 섹션에 자동으로 동기화되어 구조와 내용이 일관되게 유지된다.		API구현/프론트 미구현
TC_013	문서 목록 조회	클라우드에 있는 문서 목록을 볼 수 있다.	사용자는 접근 권한이 있는 클라우드 문서 목록을 정상적으로 조회할 수 있다.	성공	구현
TC_014	문서 상세 조회	사용자가 클라우드에 있는 문서의 내용을 볼 수 있다.	사용자는 선택한 문서의 전체 내용을 최신 상태로 확인할 수 있다.	성공	구현

TC_015	병합 제안	병합이 필요하거나 가능한 섹션들을 사용자에게 보여주며 병합을 할 것인지 제안한다.	중복되거나 의미적으로 유사한 섹션들이 병합 후보로 식별되어 사용자에게 목록 형태로 제안된다.		미구현
TC_016	병합	사용자가 병합하길 희망하는 섹션들을 병합한다.	사용자가 선택한 섹션들이 하나의 섹션으로 정상적으로 병합되고 내용 손실 없이 반영된다.		API구현/ 프론트 미구현

5. 테스트 결과 및 결론

본 테스트는 시스템의 기능성을 검증하고, 정상적인 작동을 확인하는 데 중점을 두었습니다. 기능 테스트를 통해 시스템이 요구 사항을 충족하는지 검토하였고, 향후 수정 및 구현이 필요한 부분에 대한 정보를 제공했습니다.

필수적인 기능 테스트가 선제적으로 완료된 후 성능 테스트를 통해 최적화 및 알고리즘 개선이 필요한 부분이 어디인지 파악하고자 합니다.