

부족하거나 추가 개선 가능한 부분

키워드 뉴스·역사 큐레이션 기능 통합 - 첨부된 기획 문서들에는 하루 한 개 키워드를 선정해 연관된 표준/트렌드를 큐레이션하는 기능과, 역사적 변천을 보여주는 "역사 큐레이션형 쉼표 뉴스" 포맷이 언급됩니다. 현재 코드에는 이 전용 워크플로우가 아직 구현돼 있지 않습니다. 추후 계획대로라면:

engine_core.py에 키워드 히스토리 분석 플러그인과 표준/특허 타임라인 데이터 연동 모듈을 추가하고,

orchestrator.py에서 --collect-keyword-history와 같은 새 작업을 지원하며,

UI에 “오늘의 키워드”를 입력/추천하고 관련 뉴스·역사 콘텐츠를 보여주는 뷰를 만들어야 합니다.

이는 첨부된 트렌드 보고서(ESA/NASA/IPC, EEWorld)에서 제시하는 키워드 인기 추이와도 연동해, 추천 알고리즘을 고도화할 수 있습니다.

트렌드/키워드 데이터 자동 반영 - Reddit 및 EEWorld 트렌드 보고서는 산업 표준과 키워드의 변화 추이를 보여줍니다. 이를 주기적으로 분석해 editor_rules.json의 weights와 keyword_synonyms.json을 자동 갱신하는 기능이 필요합니다. 예를 들어 특정 표준 번호나 기관이 각광받는 시기에 키워드 가중치를 높이는 식으로 콘텐츠 선택 정확도를 높일 수 있습니다.

보고서·내보내기 UI - 주요 파일 패치 계획에서 언급된 보고서 생성·내보내기 기능과 카드 요약/번역 내보내기를 UI에 구현해야 합니다. 현재 UI에는 KPI 표시와 게이트 조정은 있지만, PDF/MD/HTML 요약 보고서를 생성하여 공유하는 기능은 없습니다. 이 부분을 개선하면 관리자나 외부 사용자와 쉽게 정보를 공유할 수 있습니다.

테스트/로그 개선 - orchestrator와 engine_core에는 단계별 콜백과 JSON 로그 출력이 있지만, UI와 서버에서 이를 시각적으로 확인할 수 있는 로그 뷰어나 필터 기능을 강화할 필요가 있습니다. 예를 들어 selected_articles.json과 selected_community.json이 병합되는 과정을 자세히 보여주면 디버깅이 수월해집니다.

완전한 비동기화 - 현재 engine_core는 ThreadPoolExecutor를 통해 일부 병렬 처리를 구현하지만, asyncio/aiohttp 기반으로 완전한 비동기화를 적용하면 네트워크 I/O가 많은 크롤링과 팩트체크 단계에서 더 큰 성능 향상을 얻을 수 있습니다.

국제화 지원 - UI 텍스트와 로그는 대부분 한글로 돼 있으나, 향후 글로벌 사용자나 영문 기사 수요가 늘어날 경우 다국어 인터페이스와 설정 옵션이 필요합니다.

결론

현재 쉐리저널 시스템은 주요 파일을 대폭 리팩터링하면서 비동기 작업, 플러그인 구조, UI 개선 등 기획 문서의 핵심 요구 사항을 상당 부분 반영했습니다. 서버(API)·엔진·오케스트레이터·UI 사이의 연동도 잘 작동하며, 작업 진행 상황과 KPI를 실시간으로 모니터링할 수 있습니다. 다만, 기획 문서에서 제시한 **키워드 기반 역사 큐레이션**과 **트렌드 데이터 반영** 같은 장기 로드맵 기능은 아직 미구현 상태입니다. 앞으로는 키워드/역사 큐레이션 워크플로우와 보고서 생성 기능을 추가하고, 트렌드 분석 결과를 가중치·시놉시스에 반영하는 자동화, 완전한 비동기화를 추진하면 더욱 안정적이고 확장 가능한 시스템이 될 것입니다.