퀄리뉴스를 전면적으로 리팩터링하여 안정적이고 확장 가능한 시스템으로 구축하려면, 명확한 프레임워크 설계와 체계적인 워크플로우가 필요합니다. 아래는 수집부터 발행까지의 전체 흐름을 포괄하는 개발 가이드입니다.

---

1. 아키텍처 프레임워크

1.1 모듈 계층 구조

퀄리뉴스는 각 기능을 모듈화하여 유지보수성과 확장성을 확보해야 합니다. 기본 계층은 다음과 같습니다.

데이터 소스 계층: 공식 뉴스, 학술 논문, 표준/정책 문서, 커뮤니티/포럼 등 다양한 출처를 정의한 소스 목록 파일(JSON). 설정 파일에는 각 소스의 URL, 유형, 신뢰도 가중치 등을 포함합니다.

수집 계층 (Collector): 각 유형별 크롤러가 소스로부터 자료를 수집합니다. 뉴스 RSS/HTML 파서, 학술 API, 표준 문서 스크래퍼, 커뮤니티 파서 등으로 구성됩니다.

전처리 계층 (Preprocessing): 수집된 자료의 메타데이터를 정제하고 중복을 제거하며, 품질 게이트(QG)와 팩트체크(FC)를 수행해 신뢰도가 낮은 자료를 걸러냅니다.

평가 계층 (Evaluator): editor_rules.json에 정의된 도메인·키워드 가중치, 최신성, 이슈 중요도 등을 적용하여 기사 가치 점수를 계산합니다.

요약·번역 계층: 중요한 문장을 추출하고, 한국어 요약과 번역을 생성합니다. 번역 API는 선택적으로 사용하며, Glossary를 적용합니다【1002_2퀄리뉴스의 기술적목표.txt†L39-L47】.

저장 계층 (Storage): 전처리와 평가를 마친 자료를 데이터베이스 또는 JSON 파일에 저장합니다. 키워드 중심의 색인을 유지해 조회 효율성을 높입니다.

편집 및 승인 계층 (Admin UI): 편집자가 자료를 검토, 승인, 고정(핀)하고 코멘트를 추가할 수 있는 UI를 제공합니다.

발행 계층 (Publisher): 승인된 자료를 섹션별·키워드별로 묶어 HTML/Markdown/JSON 형태의 뉴스 페이지로 생성하고 배포합니다.

1.2 데이터 흐름

소스 로딩: feeds/official_sources.json, community_sources.json 등에서 소스를 읽습니다.

자동 수집: Collector 모듈이 지정 주기마다 소스를 순회하며 자료를 가져옵니다.

전처리 및 평가: QG/FC로 품질을 걸러내고, 점수 계산 후 저장 계층에 기록합니다.

요약·번역: 필요 시 영문 자료에 대해 요약 및 번역을 수행하여 한국어 정보를 제공합니다.

UI에서 검토: 편집자가 승인하고 한줄 코멘트를 작성합니다.

발행: 선정된 자료를 섹션 또는 키워드별로 정렬하고 HTML/MD/JSON을 생성하여 배포합니다.

---

2. 워크플로우 개발 단계

2.1 소스 정의 및 초기 설정

소스 파일 작성: official_sources.json, community_sources.json, paper_sources.json 등을 작성하여 URL, RSS, 도메인 신뢰도 등 메타정보를 등록합니다.

config.json 설정: 품질 기준(QG), 팩트체크 기준(FC), 스코어링 가중치, 섹션 순서 등을 설정합니다.

editor_rules.json 설계: 도메인별 기본 점수, 핵심 키워드 가중치, 이슈 중요도 등을 정의합니다.

2.2 수집 모듈 개발

뉴스 크롤러: RSS 파서가 기본이며, RSS가 없는 경우 BeautifulSoup으로 HTML 목록에서 기사 링크를 추출합니다.

논문/표준/정책 크롤러: 각 기관의 API 또는 사이트 검색 기능을 활용해 키워드를 기반으로 자료를 수집합니다.

커뮤니티 크롤러: Reddit, 포럼 등에서 키워드 검색 및 서브레딧 별 수집을 수행합니다.

중복 제거: 기존 자료와 비교해 동일 링크나 유사 제목을 가진 자료를 제거합니다.

2.3 평가 및 필터링

품질 게이트: 기사 길이, 문장 수, 링크 수 등을 기준으로 PASS/HOLD/REJECT를 판별합니다. 표준 문서나 요약본에 적합한 별도 기준을 두어 짧은 문서를 무조건 제외하지 않도록 합니다【1003_쿼리목표 (1).txt†L1-L30】.

팩트체크: 숫자·연도·표준 코드·근거 링크를 탐지하고 점수를 계산합니다.

스코어링: 도메인 신뢰도, 키워드 적합도, 최신성, 이슈 중요도에 따라 점수를 합산합니다.

결과 저장: 각 자료는 { id, type, score, qg_status, fc_status, metadata } 형태로 DB/파일에 저장합니다.

2.4 요약·번역 모듈

중요 문장 추출: _extract_main_text_from_html로 본문을 정제한 후 상위 2~3개 문장을 선택합니다.

한국어 요약 생성: OpenAI GPT-4o, DeepL, Google Translate 등을 활용해 추출한 문장을 기반으로 자연스러운 한국어 요약을 생성합니다【1002_2쿼리뉴스의 기술적목표.txt†L39-L47】.

Glossary 적용: 전문 용어와 표준 번호를 사전에 정의하여 용어 번역의 일관성을 유지합니다.

2.5 UI 및 승인 프로세스

대시보드 디자인: 목록 뷰(테이블/카드), 자료 상세 모달, 승인·보류·제외 버튼, 요약 및 품질/팩트 점수 표시 등 핵심 기능을 포함합니다.

키워드 모드 지원: 키워드별로 수집 결과를 조회하고, 히스토리 뷰나 연대기 필터링을 제공합니다.

편집자 협업: 여러 편집자가 동시 작업할 경우를 대비해 상태 동기화와 권한 관리를 설계합니다.

2.6 발행 및 배포

페이지 생성: 승인된 자료를 섹션 및 키워드별로 그룹화한 HTML/Markdown/JSON 파일을 생성합니다.

스케줄링: cron 또는 APScheduler를 사용해 매일 정해진 시간에 자동으로 발행합니다.

백업 및 로그: 발행 결과와 로그를 백업 디렉터리에 저장하여 장애 발생 시 복구할 수 있도록 합니다.

2.7 테스트 및 배포

단위 테스트: 각 모듈(수집, 평가, 요약, 발행)에 대해 유닛 테스트를 작성합니다.
통합 테스트: 키워드 입력부터 발행까지 전체 파이프라인 테스트를 자동화합니다.
배포: 도커 컨테이너 또는 가상환경에서 실행하도록 Dockerfile과 CI/CD 파이프라인을 구축합니다.

---

3. 향후 확장 및 고려사항

검색 최적화: Elasticsearch/OpenSearch를 도입하여 키워드 검색과 시간순 정렬을 효율화합니다.
AI 보조 기능: 기사 추천, 트렌드 예측, 키워드 추천 등의 AI 기능을 장기적으로 검토합니다.
독자 인터페이스: 독자용 사이트를 분리해 뉴스 페이지, 키워드 아카이브, 검색 기능 등을 제공할 수 있습니다.
피드백 루프: 독자의 클릭·조회 데이터를 활용해 스코어링 알고리즘과 큐레이션 품질을 지속적으로 개선합니다.

이 프레임워크와 워크플로우 가이드에 따라 개발을 진행하면, 퀄리뉴스가 키워드 중심의 역사 큐레이션과 정기 뉴스 발행을 모두 안정적이고 확장 가능한 구조로 운영할 수 있을 것입니다.