

# QualiJournal 관리자 시스템 인수인계 보고서

## 완료된 작업

- 관리자 패널 UI 개선 (`index.html`): 관리자 패널 화면에 '일일 보고서 생성', '요약/번역', 'Markdown/CSV 내보내기' 버튼을 추가하였고, 각 기능 버튼에 대해 JavaScript `fetch`를 통한 백엔드 API 연동을 구현했습니다.
- 백엔드 API 엔드포인트 구현 (`server_quali.py`): FastAPI 기반으로 새로운 엔드포인트들을 구축했습니다. 주요 엔드포인트는 `/api/report` (일일 보고서 생성), `/api/enrich/keyword` 및 `/api/enrich/selection` (뉴스 카드 요약/번역 기능), 그리고 `/api/export/md` 및 `/api/export/csv` (결과물 내보내기)입니다. 개발 과정에서 중복된 라우트를 제거하고 전반적인 코드 리팩토링을 완료했습니다.
- Cloud Run 서비스 배포 완료: `gcloud` CLI를 통해 최신 소스 코드를 Cloud Run에 배포하였으며, 운영에 필요한 환경변수 (`ADMIN_TOKEN` 등)를 설정했습니다. 현재 관리자 시스템 백엔드와 정적 프론트엔드가 Cloud Run에서 정상적으로 동작하고 있습니다.

## 다음 단계 및 우선순위 작업

- 커스텀 도메인 연결: 현재 Cloud Run이 제공하는 기본 URL로 서비스 중이므로, **standardai.co.kr** 커스텀 도메인으로 접속할 수 있도록 설정해야 합니다. 도메인 관리 업체인 가비아(Gabia)에서 해당 도메인의 DNS 레코드(CNAME 등)를 Cloud Run에 맞게 설정하고, Cloud Run의 도메인 매핑(domain mapping) 절차를 진행해야 합니다.
- 일일 보고서 자동화: 일일 보고서 생성 기능을 예약 작업으로 자동화하는 것이 다음 과제입니다. 현재는 수동 `POST` API 호출로만 트리거되므로, **Cloud Scheduler** 또는 유사한 스케줄링 서비스를 활용하여 정해진 시간마다 `/api/report` 엔드포인트를 호출하도록 구성하고, 정상 동작 여부를 테스트해야 합니다.
- 백업 스크립트 자동화: 수동으로 실행 중인 백업 스크립트(`run_backup_gcs.ps1`)를 자동화할 필요가 있습니다. 이 스크립트는 결과물이나 데이터를 **Google Cloud Storage (GCS)** 또는 로컬에 백업하기 위한 용도로 파악되며, 현재는 운영자가 직접 실행해야 합니다. 추후 CI 파이프라인, Cloud Scheduler, 또는 서버 cron 작업 등을 통해 해당 스크립트(또는 동등한 기능)를 주기적으로 실행하여 **백업 작업을 자동화**하는 방안을 고려해야 합니다.

## 추가 참고 사항

- ADMIN 토큰 관리: 관리자 전용 API에는 인증을 위해 `ADMIN_TOKEN`이 요구됩니다. 이 값은 현재 Cloud Run 환경변수로 설정되어 있으며, 프론트엔드의 JavaScript `fetch` 요청 시 HTTP 헤더 `X-Admin-Token`에 해당 토큰을 포함하도록 구현해야 합니다. (이미 일부 구현이 완료되었다면, 추후 토큰 값 변경 시 프론트엔드와 백엔드 설정을 동기화해야 합니다.)
- 환경설정 파일 최신화: 프로젝트의 환경설정 관련 파일들이 최신 상태로 반영되어 있습니다. 예를 들어, `.env`, `.gcloudignore`, `Dockerfile` 등은 최근 변경사항까지 모두 업데이트되어 있으므로, 개발 환경 구성이나 배포 시 이 최신 버전을 참고하면 됩니다. 특히 `.env`에는 필요한 환경변수들이 정리되어 있고, `Dockerfile`은 Cloud Run 배포 환경에 맞게 작성되어 있습니다.
- 시스템 구성 및 구조: QualiJournal 관리자 시스템은 **FastAPI** 기반 백엔드와 정적 **HTML/JavaScript** 프론트엔드로 구성되어 있습니다. 이 둘은 하나의 Docker 컨테이너로 빌드되어 Cloud Run에서 서비스됩니다. 즉, Cloud Run의 한 서비스 내에서 백엔드 API 서버와 관리자용 프론트엔드 페이지가 함께 동작합니다. 이 구조를 고려하여 개발 및 배포를 진행하면 되며, 새로운 개발자도 이 아키텍처를 이해하고 이어서 작업을 시작할 수 있습니다.

