

# 1. 프로젝트 한 줄 요약

7 주간 커플의 '지속성'과 '갈등 중재'를 목표로 한 웹 서비스 **아이컨택(AI Contact)** 을 설계·구현했다. Spring Boot-React-AWS 기반으로 AI 네컷만화 생성·S3 저장·DB 연동을 엔드투엔드로 완료했고, 부모 얼굴 특징→아기 이미지 생성 품질을 **퓨샷 프롬프팅**으로 개선했다

---

## 2. 프로젝트 개요

- 프로젝트명: 아이컨택(AI Contact)
  - 기간 / 투입 시간: 2025.07.07 ~ 2025.08.18 (7 주)
  - 팀 구성(역할): 총 6 명, 백엔드 4 명, 프론트엔드 2 명
    - (본인) 백엔드 & AI 연동 & 인프라: DALL·E 파이프라인, S3/RDS, JWT 인증, 엔티티/스키마 설계, 배포 파이프라인 연계
  - 사용 기술: Spring Boot 3, Java 17, React, MySQL(Amazon RDS), AWS S3, JPA(Hibernate), JWT, CI/CD, Gradle
  - 프로젝트 목표(한 문장): *개인화된 콘텐츠와 AI 중재를 통해 커플의 관계 지속성을 높이는 서비스 구축*
- 

## 3. 주요 성과와 하이라이트

- 네컷만화 자동 생성 E2E 구축: GPT 시나리오 → DALL·E 이미지 생성 → S3 업로드 → 메타데이터 DB 저장(테이블 comic\_strips: 커플 ID, 생성자 ID, S3 URL/Key, 제목, 생성시각) → 조회 API 까지 구현.
  - 아기 이미지 품질 향상: 기존 추상 프롬프트 대비 **퓨샷 프롬프팅 + 속성 열거**(눈동자 색, 얼굴형, 피부톤 등) 도입으로 **일관성·디테일 개선**([전/후 스크린샷 첨부]).
  - 인증/권한 안정화: JWT 기반 로그인/토큰 발급, Spring Security 예외 대응 및 DTO 계층 정리로 직렬화 이슈 제거.
  - 데이터 모델링 & 리포지터리 정비: ComicEntity/ComicRepository/서비스 계층 정립, S3 키 규칙(폴더링·UUID)으로 운영 안정성 확보.
- 

## 4. 도전 과제와 해결 과정

## 문제 A | 아기 이미지가 부모 특징을 제대로 반영하지 못함

- **원인:** 프롬프트가 추상적이며 속성 누락(눈·코·입·피부톤·헤어·얼굴형 등) + 샘플 일관성 부족.
  - **해결:** **퓨샷 프롬프팅**으로 속성 템플릿화(예: *눈동자 색/모양/눈두덩, 눈썹 굵기/각도, 코 길이/넓이, 입술 두께/윤곽, 피부톤, 얼굴형, 헤어 길이/컬/앞머리, 귀 크기, 광대/턱선*), 부모별 특징 벡터화 → 프롬프트 삽입; 전/후 품질 샘플 세트로 내부 리뷰.
  - **결과:** 디테일 반영률 상승,
- 

## 5. Lessons Learned

- **프롬프트 엔지니어링:** 퓨샷 + 속성 열거가 생성 품질에 미치는 영향이 큼; 샘플 세트 관리와 실패 케이스 아카이빙이 재현성에 중요.
  - **백엔드 아키텍처:** 파일 스토리지-DB의 **정합성 보장**은 키 설계와 트랜잭션 경계 설계에서 시작됨.
  - **JPA 실무 팁:** 엔티티 직접 노출 대신 **DTO 변환**으로 직렬화·순환참조·보안 리스크를 예방.
  - **협업:** PR/MR 템플릿·체크리스트·라벨링을 표준화하니 리뷰 품질과 속도가 개선.
  - **운영 관점:** 로그 기반 **지표화(실패율, 재시도, 평균 응답)** 없이는 품질 판단이 어렵다.
- 

## 6. 다음에 시도하고 싶은 것

- **AI 중재자 고도화:** 대화 요약·감정 분석·룰 기반 중재 시나리오 + *리트리벌*로 개인 맥락 반영.
  - **PWA & 푸시:** 리텐션 강화를 위한 브라우저 푸시·오프라인 모드·홈 화면 설치.
  - **개인정보 보호:** 프라이버시 by design, 민감 정보 분리 저장·권한/감사 로그 강화.
- 

## 7. 마무리 한 줄

이번 프로젝트는 **개인화와 신뢰성**을 동시에 잡기 위한 도전이었다. 다음 프로젝트에서는 **더 빠른 실행과 더 견고한 운영 기준**으로 사용자에게 ‘계속 쓰고 싶은 커플 웹앱’을 완성하겠다.