# 📋 작은 실패 갤러리 (개발 인수인계 캔버스)

## 1. 프로젝트 개요

* **앱명**: 작은 실패 갤러리
* **목표**: 사용자가 자신의 작은 실패를 기록/공유하고, 교훈을 나눌 수 있는 커뮤니티 앱
* **플랫폼**: React Native (Expo) + Firebase (Firestore, Authentication, Storage 예정)
* **플랜**: Firebase Spark(무료) 플랜으로 시작 → Storage(사진 업로드)는 Blaze 전환 이후 구현

## 2. 개발 환경

* **운영체제**: Windows
* **런타임**: Node.js, npm, npx
* **프로젝트 초기화**:
* npx create-expo-app@latest . -t blank-typescript  
  npm i expo-router firebase expo-image-picker
* **추가 패키지**:
  + 타입 정의: @types/react, @types/react-native
* **실행 명령어**:
* npx expo start -c

## 3. Firebase 설정

* **프로젝트 ID**: small-failure-pilot
* **Region**: Firestore → asia-northeast3 (서울)
* **firebaseConfig 주요 값**:
  + projectId: small-failure-pilot
  + storageBucket: small-failure-pilot.appspot.com (Storage Blaze 업그레이드 필요)
* **사용 서비스**: Firestore, Authentication(익명 로그인), Storage(보류 중)

## 4. 구현 현황

### 4.1 기본 구조

* **라우팅**: expo-router
  + / → 글 목록
  + /new → 새 글 쓰기
  + /post/[id] → 글 상세 보기
* **데이터 모델** (Firestore)
  + posts 컬렉션: { title, body, lessons, tags, authorId, createdAt, likeCount, commentCount, … }
  + comments 컬렉션: { postId, body, authorId, createdAt }

### 4.2 기능 현황

* **글 작성**: 제목/본문/배운 점/태그 작성 → Firestore에 저장
* **익명 로그인**: ensureAnonSignIn()으로 글쓰기/댓글/공감 전에 보장
* **글 목록**:
  + 최신순 정렬 (createdAt desc)
  + 작성 시각(KST) 표시
  + 제목/본문 요약 표시
* **글 상세**:
  + 제목, 본문, 배운 점, 태그 배지, 작성 시간 표시
  + 댓글 목록(작성 시간 포함)
  + 공감/신고 버튼
  + 글 작성자 본인만 삭제 버튼 표시
* **공감 (좋아요)**:
  + robust 토글 구현: 같은 사용자가 다시 누르면 취소
  + posts/{id}/likes/{uid} 서브컬렉션으로 관리
* **댓글**:
  + 실시간 구독
  + 빈 문자열 방지 + 글자수 제한
  + 작성 시각 표시
* **삭제**:
  + 글 작성자만 가능
  + 글 삭제 시 댓글도 함께 삭제
* **실패 붙이기 (신규 기능)**:
  + 다른 사람의 실패담에 유사하거나 도움이 될만한 나의 실패를 **“붙이기”** 가능.
  + 붙여진 글은 원글과 연결되어 **연간 실패 모음**으로 확장 노출됨.
  + 사용자는 관련 경험을 쉽게 탐색하고, 실패 경험의 맥락을 넓게 이해할 수 있음.

## 5. Firestore Rules (최신 버전)

rules\_version = '2';  
service cloud.firestore {  
 match /databases/{database}/documents {  
  
 match /posts/{postId} {  
 allow read: if true;  
  
 allow create: if request.auth != null;  
  
 // 작성자만 글 수정/삭제  
 allow update, delete: if request.auth != null  
 && resource.data.authorId == request.auth.uid;  
  
 // likeCount/commentCount 업데이트는 로그인 사용자 누구나 가능  
 allow update: if request.auth != null  
 && request.resource.data.diff(resource.data).changedKeys().hasOnly(['likeCount','commentCount']);  
 }  
  
 match /comments/{commentId} {  
 allow read: if true;  
  
 allow create: if request.auth != null  
 && request.resource.data.postId is string  
 && request.resource.data.body is string;  
  
 // 댓글 작성자 또는 해당 글의 작성자만 삭제 가능  
 allow delete: if request.auth != null && (  
 resource.data.authorId == request.auth.uid ||  
 get(/databases/$(database)/documents/posts/$(resource.data.postId)).data.authorId == request.auth.uid  
 );  
 }  
 }  
}

## 6. 문제와 해결

* **문제: 사진 업로드 실패 (CORS/권한 에러)**  
  → Storage는 Blaze 플랜 필요, 현재 Spark라 보류
* **문제: 글 삭제 실패 (권한 부족)**  
  → 규칙 보완(댓글 삭제 시 글 작성자도 허용)으로 해결
* **문제: 빈 댓글 제출 시 에러 처리 없음**  
  → UI + 서버 모두 가드 추가 (Alert 표시)
* **문제: 공감 버튼 에러**  
  → robust 토글 + 규칙 보완으로 안정화

## 7. 앞으로의 작업 (로드맵)

### 1단계 (완료 ✅)

* 글 작성/목록/상세
* 작성 시간 표시
* 댓글 & 공감 기본 로직
* 글 삭제 (작성자 전용)

### 2단계 (예정 🚩)

* **검색/태그 필터**: Firestore 쿼리(where("tags","array-contains",tag))
* **공감 표시 개선**: 내가 공감했는지 여부에 따라 버튼 라벨 변경
* **댓글 삭제/수정**: 댓글 작성자 권한 추가
* **실패 붙이기**: UI/UX 설계 및 Firestore 구조 확장 (원글 ↔ 붙이기 연결)

### 3단계 (Blaze 전환 후) 💳

* Storage 활성화 → 이미지 업로드 구현
* 글/댓글에 이미지 첨부 및 표시
* 이미지 삭제 로직(Storage Rules 포함)

### 4단계 (추가 확장) 🌱

* 사용자 프로필(선택적)
* 알림(Push Notifications)
* 모더레이션 대시보드(신고글 관리)

## 8. 타겟 페르소나

### 🎯 페르소나 1: 대학생/취업 준비생

* **배경**: 20대 초반, 학업과 진로에 대한 불안이 큰 상황. 실패 경험을 나누고 싶지만 SNS에서는 두렵다.
* **목표**: 작은 실패를 안전하게 기록·공유하고, 동료들의 공감을 통해 위로받고 싶다.
* **문제/불편**: 기존 SNS는 평가와 비교가 두렵고, 실패를 공유하기 어렵다.
* **앱에서 기대하는 가치**: 익명 기반의 안전한 공간, 간단한 기록과 빠른 공감 피드백.
* **핵심 행동 시나리오**: 시험 망친 경험 작성 → 교훈 한 줄 기록 → 다른 사용자들의 공감 확인.

### 🎯 페르소나 2: 직장인 (30대)

* **배경**: 회사 생활 중 실수와 시행착오가 잦다. 이를 정리해 두어 향후 참고하고 싶다.
* **목표**: 실패를 기록하고 교훈을 체계적으로 남겨 성장의 자산으로 삼는다.
* **문제/불편**: 개인 노트는 혼자만 보게 되고, 다른 사람과 교류할 기회가 없다.
* **앱에서 기대하는 가치**: 기록의 구조화(제목/본문/교훈/태그), 커뮤니티와의 상호작용.
* **핵심 행동 시나리오**: 프로젝트 실패 경험 정리 → 태그 추가 → 비슷한 경험을 한 사용자들의 댓글/피드백 수집.

📌 **핵심 요약**  
현재 상태: **텍스트 기반 실패 공유 갤러리 v1 완성**  
제약: Storage(이미지 업로드)는 Blaze 필요, 현재 Spark라 미구현  
다음 스텝: **검색/태그 필터**, **공감/댓글 UX 보강**, **실패 붙이기 기능 개발**