Firestore 설계서 — 최종본 v1.2

프로젝트: **FANUZU** iOS/Android 1차 정식 출시 버전 개발 근거: **정책 원장 v1.1**(2026-09-16 발주사 회신본 — 확정 적용 기준 C-01 ~ C-43 · 일반 처리 G-01 ~ G-08 · 정정 R01 ~ R22 · 미확정 S01 ~ S06) · 별첨 B v1.4(2026-09-07) · 정책 설계서 최종본 v1.1 · 별첨 2 2주차 「Firestore 설계서」 · 견적서 v2.1 7p 「산출물 목록 · 설계」 작성: 스위치 주식회사(Sweech) · 개정 2026-09-18 (최초 2026-09-01) 대체 관계: `Firestore 설계서 최종본 v1.1` **(2026-09-11)을 대체**합니다 함께 제출하는 실물: `firestore.rules` · `firestore.indexes.json` · **Cloud Functions 소스 37종** 구현 상태 기준: **빌드 v0.3.0 · 2026-09-14**(원장 **G-06**)

✦ 개정 이력 — v1.1 → v1.2

2026-09-16 발주사 회신본을 반영한 개정입니다. 회신의 확정 적용 기준 43건 · 일반 처리 8항 · 정정 22건이 **정책 원장 v1.1** 에 먼저 등재되었고, 이 문서는 그 원장의 C · G · R 번호를 스키마 · 불변식 · 인덱스 · 함수로 옮깁니다. **정책값의 단일 기준은 원장이며, 이 문서는 데이터가 어떤 모양으로 저장되는지를 정합니다.**

#	v1.1 의 설계	v1.2 의 확정 설계	근거	절
1	대표 팬덤 필드 없음	`users.{primaryFandomId, primaryChangedAt}` **신설.** 대표 팬덤 1개 · 변경 무료 · 재변경 30일 대기 · Quest 참여 자격 판정의 입력	**C-15 · C-16**	§2.1
2	`userLevel` = 정수 · 등급명은 마스터 데이터(저희 해석)	`users.{userExp, userLevel}` **— User EXP 기반 Lv.1~100+ 연속 성장.** GLOW 1~9 / SPARK 10~29 / BEAM 30~59 / STARLIGHT 60~99 / NOVA 100 이상은 **레벨 구간 칭호. 누적 기여 POP 5등급 승급 폐기**	**C-18 · R04**	§2.1 · §0.1-4
3	Quest 분류 · 활동 마스터 미정의	`/config/questCatalog` **신설** — 카테고리 · 활동 · 순서 · 노출 · `ugcAllowed` · 활동별 입력 필드 스키마	**C-01 · C-02**	§2.8
4	`quests` 필드 미기재	`quests.{fandomId, rewardMode, link{…}, userDescription, publishedAt, endsAt, creatorTz, verifyMethods[], repeat{…}}` **확정** · `quests/{questId}/edits` **신설**(공개 후 수정 이력)	**C-03 ~ C-07 · C-12 · C-16**	§2.9
5	제출 · 검토 · 지급 상태 컬렉션 없음	`questSubmissions` **신설** — `status` 4단계(`submitted` · `approved` · `rejected` · `paid`) + `payoutStatus` · `fandomIdAtSubmit` · `evidence[]` · `reviewLog[]` **. 시간 경과 자동 승인 경로 없음**	**C-08 · C-17 · G-01 · G-03**	§2.10
6	이클립스 스키마 미기재 · 침식 선차감	`eclipseRounds.{startAt, windowSec, fpiAtStart, penalty, appliedAt, target, result, cancelReason}` **확정. 성공 시 감소 없음 · 실패 확정 시에만 1회 차감** — 선차감 후 복구 방식 폐기	**C-22 · C-23 · C-26 ~ C-28 · R09**	§2.11
7	이클립스 일정 = 하루 2회 고정	`config/eclipse.schedule[]` · `fandoms.eclipseSchedule[]` **신설.** 팬덤별 횟수 · 시각 설정 · 미설정 시 공통값. **「하루 정확히 2회」 제한 폐기**	**C-24**	§2.11
8	집계 분모 컬렉션 미정의	`fandomDailyStats.{dau, active30}` **신설** — Eclipse 목표는 7일 평균 DAU, Zap 가중치는 30일 Active. 두 집계는 별개	**C-21 · C-26 · C-27**	§2.12
9	Product Catalog 에 기간 필드 없음	`productCatalog.{recommendedDays, minDays, maxDays, executionLeadDays}` **신설.** 모금 기간과 집행 기간을 별개 필드로 분리	**C-14**	§2.13
10	Growth POP 원천 = Zap · 기여 이벤트	**보상 획득과 이후 기여 두 단계 모두 성장.** growth 이벤트에 `sourceEventId` **(원 이벤트 멱등 키)** · `fandomIdAtPayout` **(지급 시점 대표 팬덤) · 역분개** 신설. 예치 · 반환 · 수수료는 원천 아님	**C-29 ~ C-33 · G-02**	§2.14 · §3.6
11	지갑 잔액 = 타입별 1필드	**총 보유 · 예약 · 사용 가능 3필드로 분리.** `balance == reserved + available` 등식과 `Pool == 지급 + 예약 + 반환` 등식을 대조 항목에 추가	**G-05 · C-09**	§2.1 · §3.5
12	FPI 축 값의 원천을 명시하지 않음	**원천은 활동 이벤트이며 지급 POP 이 아닙니다.** 같은 활동 · 같은 조건이면 보상액이 달라도 점수가 같습니다	**C-34 · R07**	§4.1
13	축별 원천 이벤트 표 없음	**출석 · 성지 / 생일 팝업 / 응원 현장 방문 → COMMUNITY**, **걸음 · 운동 → HEALTH** 확정	**C-35**	§4.1
14	동률 = **최근 활동 시각이 빠른 쪽 상위**	**1 · 1 · 3 공동 순위.** 표시 반올림만 같으면 실제 값 순 · 동점 내부 배치는 결정적 정렬키 · **최근 활동 시각 차등 없음**	**C-19**	§4.3 · §4.4
15	보안 규칙 3원칙만 기재	**공개 후 잠금 필드 거부 · 이메일 로컬 파트 매칭 금지 · 사전예약 서비스키 서버 보관**을 규칙과 함수 경계로 명시	**C-07 · C-42 · G-07 · R01**	§5
16	「복합 인덱스 51건 · 필드 예외 6건」	**복합 인덱스 63건 · 필드 예외 6건**(빌드 v0.3.0 실물 대사) + v1.2 신설 인덱스 목록	**R17**	§6
17	「구현 32종 + 신설 3종 · 남은 6종」	**실물 37종 대사표.** v1.1 이 신설로 적은 POP 소멸 3종은 **미구현**이며, v1.1 목록에 없던 5종이 실물에 있습니다. 신설 함수 `fn-link-preview` · `fn-prereg-lookup` · 브리핑 수신 규격 · Eclipse 판정 변경	**R17 · C-05 · C-39 · C-42 · C-22**	§7
18	§9 = v1.0 의 🔴 6건 처리 결과	그 표는 그대로 두고 **「v1.1 → v1.2 변경 값」 표**를 이어 붙였습니다	원장 §16	§9

바뀌지 않은 것 — POP 원장의 네 가지 불변식(`popType` 분리 · 문서 ID 멱등 · append-only · 3자 대조)과 별첨 1 ② 판정 대상 설계입니다. **이 개정은 계약 범위 · 비용 · 일정을 바꾸지 않습니다.**

> 이 문서의 「구현 상태」는 빌드 v0.3.0(2026-09-14) 기준입니다(G-06). 화면이 있다는 것 · 시험 사례가 있다는 것을 검수 통과로 적지 않았습니다(G-08).

✦ 개정 이력 — v1.0 → v1.1

별첨 B 를 반영한 개정입니다. 별첨 B 87개 항목을 **정책 원장 v1.0** 에 전건 등재했고, 이 문서의 스키마·불변식·배치를 그 원장에 맞춰 고쳤습니다.

#	v1.0 의 설계	v1.1 의 확정 설계	원장
1	`authProviders` = **5종**(`email` 포함)	**4종 —** `google.com` · `apple.com` · `kakao` · `line`. **한 UID 에 복수 Provider 연결 허용**	**B-49**
2	`locale` = `ko` · `en` · `ja` · `zh` · `es`	`ko` · `en` · `ja` · `zh-CN` · `es` . `zh-TW` 는 구조만 준비	**B-76**
3	`tier` = `GLOW` | `SPARK` | `BEAM` | `STARLIGHT` | `NOVA` 단일 필드	`passportTier` (`SILVER` | `GOLD` | `DIAMOND`)**와** `userLevel` **을 별개 필드로 분리.** Badge 도 별도	**B-65** · **B-66**
4	`originNo` (ORIGIN 100 순번)	`firstLightBadge` **(사전예약 매칭)와** `fandomJoinNo` **(팬덤 최초 100명 순번)로 분리.** `ORIGIN` 명칭 폐기	**B-67** · **B-68**
5	`planetStages.stageNo` = **1~7**(태동/성운 포함)	`stageNo` **0~6 — Stage0 성운은 정식 6단계와 분리** 보관	**B-08** · **B-53**
6	**활동 POP 자동소멸·휴면 잠금 필드를 두지 않음**(W-03 정정선)	`lockAt` · `graceUntil` · `expiredAt` **신설.** 일반 180일 · 기여 365일 → 잠금 → 30일 Grace → 소멸	**B-05**
7	차감 순서 = **규범 없음**(`lotOrder: grantedAt-asc-nonnormative`)	**규범 확정** — 겸용 사용처는 **일반 POP 우선**, 부족분 기여 POP. 같은 타입 내부는 선입선출	**B-06**
8	불변식 「혼합 차감 금지」 — 한 거래가 두 재화를 건드릴 수 없음	**유지.** 겸용 사용처는 `popType` **별 거래 2건을 한 트랜잭션**으로 묶어 처리	**B-06**
9	3자 대조 축 = **활동/지원 POP**	**일반/기여 POP.** `tier` 검증을 `passportTier` · `userLevel` **2건**으로 분리	**B-01** · **B-66**
10	POP 소멸 배치 없음	`fn-pop-expire` **신설**(일 1회) · `fn-pop-expire-notify` 로 잠금 30일 전·7일 전·즉시·소멸 전 알림	**B-05** · **B-73**
11	§9 **남은 🔴 6건**	**전건 해소.** 처리 결과를 §9 에 적었고 🔴 는 **0건**	**원장 §14**
12	`rankings/{rankingId}` **— 문서 형태 미정의**	`rankings/country-{axis}` **신설**(§2.6). 축마다 문서 1건 · `rows` 사전 집계 · 5분 주기 · 미지정 국가는 `ZZ` 버킷	**B-17**
13	STATION 비노출 **구현 수단 미기재**	`config/features.stationEnabled` **(기본** `false` **) 신설**(§2.7). 앱 재배포 없이 노출 전환	**B-50**

추가로 반영한 항목 — Funding 후원은 **기여 POP 만(B-33)** · Campaign 에서 `seedPop` 제거(**B-31**) · Product Catalog 컬렉션(**B-59**) · Global Ticker 단일 큐(**B-80**) · 원장 5년 보존(**B-74**).

어느 것도 계약 범위를 줄이는 변경이 아닙니다. POP 원장의 네 가지 불변식과 3자 대조 구조는 그대로이며, 별첨 1 ② 판정 대상 설계는 유지됩니다.

✦ 0. v0.3에서 무엇이 바뀌었나

v0.3은 「마일스톤 1 승인 후 v1.0 확정」으로 판본을 열어 두었습니다. 그 사이에 두 가지가 생겼습니다.

1. **정책이 확정되었습니다.** 답변서 v1.1과 수치표 v1.4를 반영한 정책 설계서 최종본이 v0.3 당시 비어 있던 값 여럿을 채웠습니다.
2. **설계가 구현물이 되었습니다.** v0.3의 §8은 Cloud Functions 인터페이스 목록이었고 「구현은 이후 주차」라고 적혀 있었습니다. **지금은 32종이 실제로 동작하며 에뮬레이터 테스트를 지납니다.**

이 최종본은 그 둘을 반영합니다. **어느 것도 계약 범위를 줄이는 변경이 아닙니다.**

0.1 v0.3 → v1.0 에서 바뀐 값

아래 표는 v1.0 시점의 기록입니다. 이 중 **3 · 4 · 5 · 6 · 7 · 8 번은 v1.1 에서 별첨 B 로 다시 바뀌었습니다** — 지금 유효한 값은 위 「개정 이력」 표를 보십시오.

#	v0.3의 값	확정값	근거
1	FPI 축 = **action · content · contribution · community · growth**	**bpm · health · referral · community · contribution**	정책 설계서 §P-03 (답변서 v1.1 Q3)
2	FPI 산출식 = **가중 합**(가중치 각 0.2 · 합계 1)	**단순 합산** — 가중치 초기값 각 1.0 · Admin 조정	정책 설계서 §P-03 (PRD v9.0 PRD-03)
3	`authProviders` 병행 여부 CD-18 🔴	~~이메일 · Google · Apple · Kakao · LINE 5종~~ → **v1.1 에서 4종(Google · Apple · Kakao · LINE)으로 대체(B-49) — 현행값 아님**	정책 설계서 §P-01 (답변서 Q7)
4	등급 3단계 · 승급 조건 미확정	~~5단계 GLOW·SPARK·BEAM·STARLIGHT·NOVA · 누적 기여 POP 3,000/20,000/100,000/500,000~~ → **[v1.2 정정 · C-18 · R04] 폐기.** User EXP 기반 **Lv.1~100+ 연속 성장**이며 GLOW~NOVA 는 **레벨 구간 칭호**입니다(§2.1)	정책 설계서 §P-04 → **원장 C-18**
5	`planetStages` 7단계 구간값 🔴	~~Lv.1~100 · 7 Stage · 누적 POP 50,000,000 · sFPI 7,500,000~~ → **v1.1 에서 Stage0 + 6단계로 대체(B-53) — 현행값 아님**	정책 설계서 §P-08 (수치표 v1.4)
6	성운 → 행성 전환 = 사전예약 500명	**10명 이상 + 1,000 POP 이상 동시 충족**	정책 설계서 §P-08 (답변서 Q5)
7	`quest.stepTargets` = `[3000,6000,10000]` 🔴	~~1,000 · 5,000 · 10,000 · 15,000보 · 구간별 10 POP~~ → **v1.1 에서 구간별 5 · 6 · 7 · 8 POP 으로 대체(B-26) — 현행값 아님**	정책 설계서 §P-05 (답변서 Q3)
8	`i18n.supportedLocales` = `['ko','en','ja']` 🔴	**한국어 · English · 日本語 · 中文 · Español**	정책 설계서 §P-13 (앱 Figma `554:24126`)
9	이클립스 침식 수치 CD-91 🔴	**침식 0.25% FPI** · 발동 조건·참여 인원은 Admin 운영값	정책 설계서 §P-07 (수치표 v1.4)
10	캠페인 환불 기한 미기재	**마감 후 72시간 내 전건 완료**	정책 설계서 §P-09

[v1.2 정정 · R04] 위 표 **4번의 누적 기여 POP 5등급 승급은 폐기되었습니다.** 2026-09-16 회신이 User Level 을 **User EXP 기반 Lv.1~100+ 연속 성장**으로 확정했고, GLOW · SPARK · BEAM · STARLIGHT · NOVA 는 레벨 구간의 **칭호**입니다(**C-18**). 문턱값 3,000 / 20,000 / 100,000 / 500,000 을 승급 조건으로 쓰는 표기는 이 문서에 **남아 있지 않습니다.** 빌드 v0.3.0 의 등급 산출은 이 문턱으로 되어 있어 **정정 필요**이며, 정정 판본은 v0.4.0 입니다.

1번과 2번은 저희 잘못입니다. 답변서 v1.1이 FPI 지표 5종을 확정해 주셨는데, 저희가 v0.3에 적어 둔 다른 축 이름을 그대로 두었습니다. **집계 축과 화면 표기가 갈린 채로 코드까지 들어갔고, 이 최종본을 쓰면서 발견해 양쪽을 고쳤습니다.** 산출식도 PRD-03이 「초기 MVP는 단순 합산」이라고 정한 것을 저희가 가중 평균으로 적었습니다.

0.2 바뀌지 않은 것

POP 원장 설계는 v0.3 그대로입니다. 별첨 1 ②의 판정 대상이라 임의로 바꿀 수 없고, 바꿀 이유도 없었습니다. `popType` 필드 분리 · 문서 ID 멱등 · append-only · 3자 대조 — 네 가지가 그대로 구현되었습니다(§8.1).

W-03 정정선은 v1.1 에서 철회합니다. v1.0 은 활동 POP 자동소멸과 FIFO 를 철회하고 활성 스키마에 만료 시각·휴면 잠금 필드를 두지 않았습니다. **별첨 B 가 일반 180일 · 기여 365일 소멸과 「일반 POP 우선」 차감 순서를 확정**했으므로 두 가지를 모두 되살립니다 (원장 **B-05 · B-06** · 아래 §2.5 · §3.2).

바뀌지 않은 것은 네 가지 불변식입니다 — `popType` 필드 분리 · 문서 ID 멱등 · append-only · 3자 대조. 별첨 1 ② 판정 대상이라 임의로 바꾸지 않았습니다(§3.1).

✦ 1. 이 문서의 범위

다루는 것	다루지 않는 것
데이터가 어디에 어떤 모양으로 저장되는가	화면 배치·문구 (기능 명세서)
누가 무엇을 읽고 쓸 수 있는가	정책 수치의 확정 여부 (정책 설계서 최종본)
어떤 쿼리로 읽히고 어떤 인덱스가 받는가	검증 항목과 결과 (QA 체크리스트)
Cloud Functions 계약과 구현 상태	성능 측정값 (성능 테스트 리포트)

표기는 정책 원장·정책 설계서와 같습니다 — 🟢 **확정** · 🟡 **운영값**(Admin · Remote Config · 서버 설정으로 조정) · 🔵 **미확정(S)**(2026-09-16 회신 5절이 「확인 전 유지할 기준」과 함께 남긴 항목 · 원장 §19). 🔴 **미확정은 이 문서에 0건입니다**(§9). 🔵 항목은 유지 기준으로 진행하면서 명세를 제출하는 자리이며, 해당 절마다 **확인 전 유지 기준**을 함께 적었습니다.

✦ 2. 데이터 모델

v0.3 §2의 엔티티 26종과 컬렉션 구조를 그대로 씁니다. **바뀐 필드만 적습니다.**

구현 상태 표기(원장 **G-06** · 기준 빌드 **v0.3.0 · 2026-09-14**)

표기	뜻
구현 완료 · 미배포	빌드 v0.3.0 에 있고 에뮬레이터 시험을 지납니다. 개발 서버 배포는 Billing 전환(**C-36**) 뒤입니다
구현 중	일부 경로만 있습니다. 무엇이 비어 있는지 항목마다 적었습니다
정정 필요	빌드 v0.3.0 의 구현이 이번 확정 기준과 **다릅니다.** 정정 판본은 v0.4.0 입니다
미구현	이번 개정으로 설계가 확정되었고 구현은 v0.4.0 입니다

시험 사례 수 · 화면 존재는 검수 통과가 아닙니다(G-08). 이 열은 코드가 무엇을 하고 있는지만 적습니다.

2.1 `/users/{uid}`

필드	v1.0	v1.1	원장
`authProviders`	`email` · `google.com` · `apple.com` · `kakao` · `line` **5종**	`google.com` · `apple.com` · `kakao` · `line` **4종.** `email` 제거 · **배열이 복수 Provider 를 담아 한 UID 에 연결**	**B-49**
`locale`	`ko` · `en` · `ja` · `zh` · `es`	`ko` · `en` · `ja` · `zh-CN` · `es` . `zh-TW` 는 값만 추가하면 되도록 구조 유지	**B-76**
`passportTier`	(`tier` 로 있었음)	`SILVER` | `GOLD` | `DIAMOND` · 서버 산출 · **강등 없음**(단조 증가)	**B-65**
`userLevel`	(없음)	int · User EXP 기반 · `passportTier` **와 별개 체계** 🟢 **신설**	**B-66**
`firstLightBadge`	(`originNo` 로 혼재)	bool · **사전예약 완료·앱 매칭 사용자** · 일반 POP 50 지급 이력과 짝 🟢 **신설**	**B-67**
`fandomJoinNo`	(`originNo` 로 혼재)	int · **팬덤 최초 100명 순번 001~100** · 없으면 `null` . **경제적 혜택 없음** 🟢 **신설**	**B-68**
`nickname`	2~12자 · 유니크	**2~20자 · 중복 허용.** 유니크 인덱스 **제거** · 식별은 `uid`	**B-52**
`lastActiveAt`	(없음)	timestamp · **POP 휴면 판정 입력**(§2.5) 🟢 **신설**	**B-05**

`tier` **단일 필드를** `passportTier` **와** `userLevel` **로 나눈 것이 v1.1 의 가장 큰 스키마 변경입니다.** 별첨 B 가 「User Level · Passport Tier · Badge 는 서로 다른 체계로 분리한다」고 확정했으므로, 한 필드가 두 체계를 겸할 수 없습니다. **한 값의 변화가 다른 값을 자동으로 바꾸지 않도록 산출 함수도 분리**했습니다.

`passportTier` 를 사용자 문서에 두는 이유는 조회 때문입니다. 매번 계산하면 랭킹·프로필· 라운지에서 같은 계산이 반복됩니다. **파생값이므로 원본이 움직일 때 함께 갱신하고 대사 대상에 넣습니다**(§3.4). `passportTier` **는 강등하지 않으므로 갱신은 단조 증가만 허용**하고, 낮아지는 갱신 요청은 거부합니다.

[v1.2 정정 · C-18] v1.1 은 앱 Figma 「TIER 정보」의 GLOW~NOVA 5단계를 User Level 체계로 본 **저희 해석**을 적고 확인을 요청드렸습니다(원장 §15-①). **회신이 이 해석을 기각**하고, User Level 을 **User EXP 기반 Lv.1~100+ 연속 성장**으로, GLOW~NOVA 를 **레벨 구간 칭호**로 확정했습니다. `userLevel` 을 정수로 둔 설계는 그대로 쓸 수 있으며, 등급명은 열거형이 아니라 **레벨 구간 표시**입니다. 확정 내용은 아래 「v1.2 에서 더한 필드」에 있습니다.

v1.2 에서 더한 필드

필드	타입	값 · 규칙	근거	구현 상태
`primaryFandomId`	string | null	**대표 팬덤 1개.** 복수 팬덤 가입은 유지하되 대표는 하나입니다. Quest 신규 참여 자격 판정의 입력이며, 미가입이면 `null` 🟢 **신설**	**C-15 · C-16**	**정정 필요** — 빌드 v0.3.0 은 `fandomId` 단일 필드이며 대표 · 가입의 구분이 없습니다
`primaryChangedAt`	timestamp	**마지막 대표 팬덤 변경 시각.** `primaryChangedAt + 30일` 이 지나야 재변경이 허용됩니다. **변경은 무료**이며 과거 기여 · Passport 기록을 건드리지 않습니다. 30일은 🟡 운영값 🟢 **신설**	**C-15**	**미구현**
`userExp`	int	**User EXP 누적.** `userLevel` 의 유일한 입력입니다 🟢 **신설**	**C-18**	**미구현**
`userLevel`	int	**Lv.1 ~ 100+ 연속.** 100 이후에도 숫자가 계속 올라갑니다. 화면은 `Lv.N` 과 구간 칭호를 함께 표시합니다 — GLOW 1~9 / SPARK 10~29 / BEAM 30~59 / STARLIGHT 60~99 / NOVA 100 이상. **칭호는 표시 구간일 뿐 권한 · 보상이 붙지 않습니다**	**C-18 · R04**	**정정 필요** — 빌드 v0.3.0 은 누적 기여 POP 문턱(3,000 / 20,000 / 100,000 / 500,000)으로 5등급을 산출합니다

`userLevel` **은 더 이상 열거형 매핑이 아닙니다.** v1.1 은 등급명을 마스터 데이터로 빼 두고 확인을 요청드렸고, 회신이 **레벨 구간 칭호**로 확정했습니다(**C-18** · 원장 §15-①). 그래서 산출 입력은 `userExp` 하나이며, **누적 기여 POP 은** `passportTier` **산출에만 쓰입니다.** 두 체계가 서로의 입력을 공유하지 않는다는 것을 §3.4 대조가 확인합니다.

> **User EXP 요구량(레벨별 필요 EXP)은 🔵 미확정(S01)입니다.** 회신은 **「새 EXP 요구량을 신설하지 않는다」**고 하셨으므로 기존 자료에 정의된 값을 따르며, 저희가 임의로 채우지 않습니다. **확인 전 유지 기준** — `userLevel` 은 정수 필드로 두고 요구량 표를 마스터 데이터로 적재해, 확정 시 재배포 없이 반영합니다(원장 **§19 S01**).

대표 팬덤 변경이 다른 기능을 잠그지 않습니다(C-15). 변경으로 달라지는 것은 **Quest 신규 참여 자격**뿐이고(**C-16**), 이미 제출된 인증 건은 §2.10 의 `fandomIdAtSubmit` 이 붙들어 둡니다(**C-17**).

2.2 `/fandoms/{fandomId}`

필드	v0.3	v1.0
`fpiAxes`	`{action, content, contribution, community, growth}`	`{bpm, health, referral, community, contribution}` 🟢
`fpiScore`	가중 합	**단순 합산** (가중치 초기 1.0) 🟢

v1.2 에서 더한 필드

필드	타입	값 · 규칙	근거	구현 상태
`eclipseSchedule`	array<map>	**팬덤별 이클립스 일정.** `{hhmm, tz}` 의 배열이며 **횟수에 고정 제한이 없습니다.** 비어 있으면 `config/eclipse.schedule[]` 공통값을 씁니다 🟢 **신설**	**C-24**	**정정 필요** — 빌드 v0.3.0 은 1차 · 2차 시각 2개 필드로 하루 2회를 전제합니다
`rankingEligible`	bool	서버가 내리는 랭킹 모집단 판정(§4.4). 화면이 부등호로 거르지 않습니다	**B-17**	**구현 완료 · 미배포**

2.3 `/planetStages/{stageNo}` — 수치 확정

v0.3은 `requiredPop` 구간값을 🔴 로 비워 두고 `config` 참조로 미뤘습니다. 수치표 v1.4가 Level 1~100을 200행으로 확정해 주셔서 **마스터 데이터로 적재**합니다.

필드	타입	설명
`stageNo`	int	**0~6. Stage0 성운(Pre-Planet) · Stage1 시작 · Stage2 성장 · Stage3 함께 · Stage4 동행 · Stage5 빛 · Stage6 영원.** Stage0 은 **정식 6단계와 분리** 보관하며 FPI 랭킹 모집단에서 제외(원장 **B-08 · B-53 · B-17**)
`levelFrom` / `levelTo`	int	Stage 가 덮는 Level 구간
`requiredPop`	int	해당 Level 진입 누적 기여 POP 🟢 **확정**
`requiredSfpi`	int	해당 Level 진입 누적 sFPI 🟢 **확정**

Level-up 은 누적 Growth POP 과 sFPI 를 동시에 충족해야 일어납니다(원장 **B-11 · B-54**). 하나만 채우면 현재 Level 에서 대기하며, 초과분은 다음 Level 진행도로 이월합니다. `planetExp` **필드는 두지 않습니다** — 별첨 B 가 Planet EXP 를 삭제하고 성장 기준을 이 둘로 일원화했습니다(**B-11**).

`sfpi` **는 Daily FPI Snapshot 의 단순 누적값입니다(B-54).** 별도 환산비율을 두지 않으므로 `sfpi == Σ dailyFpiSnapshots.value` 가 성립해야 하며, 이 등식을 §3.4 대조 항목에 넣었습니다.

`config` **로 빼지 않고 마스터 데이터로 둔 이유** — 200행짜리 구간표를 설정 문서 하나에 넣으면 문서 크기 한도(1MiB)에 가깝고, Admin 화면에서 한 줄만 고치기도 어렵습니다. `planetStages` 는 읽기 공개·쓰기 마스터 전용이라 규칙도 이미 서 있습니다.

2.4 `/fandoms/{fandomId}/planets/{planetId}` — 성운 전환

필드	v0.3	v1.0
전환 조건	사전예약 500명 (CD-16)	**멤버 10명 이상 + 누적 1,000 POP 이상** ●
`nebulaDeadline`	(없음)	timestamp · **기본 7일** · 미달성 시 **소멸시키지 않고 유지·연장** ●
`zapWeight`	(없음)	double · **상한 1.5** · 레벨업 진행도에만 적용 ● **신설**

`zapWeight` **는 POP 잔액·원장·누적 기여 POP 을 건드리지 않습니다**(정책 §P-08). 성장 진행도 계산에만 곱합니다. 원장에 곱하면 재화가 늘어나 3자 대조가 깨집니다.

팬덤당 행성은 1건입니다 — R17 확인 결과

축	내용
확정 제약	**팬덤 1개에 행성 1건.** 성운(Stage0)에서 정식 행성으로 전환되는 것도 같은 1건이며, 팬덤 안에 행성이 둘 생기면 성장 진행도 · FPI 귀속이 갈립니다
제약을 세우는 자리	**문서 ID.** 조건문으로 「이미 있으면 만들지 않는다」를 쓰면 동시 요청에서 뚫립니다. 문서 ID 충돌은 Firestore 가 원자적으로 판정합니다
빌드 v0.3.0	행성 문서 ID 가 **생성자** `uid` 로 고정되어 있어 **사용자당 1건**은 보장되지만 **팬덤당 1건은 보장되지 않습니다.** 한 팬덤에 두 사용자가 온보딩하면 행성 문서가 2건 생깁니다
정정	문서 ID 를 **팬덤 안에서 고정된 단일 ID** 로 바꾸고 생성자는 `ownerUid` 필드로 남깁니다. **정정 필요** · v0.4.0. 기존 문서의 통합은 §3.7 마이그레이션 대사 항목입니다

2.5 `/users/{uid}/popLots/{lotId}` — 휴면 잠금 · 소멸 ● 신설

v1.0 은 만료 시각과 휴면 잠금 필드를 두지 않았습니다. 별첨 B 가 소멸 정책을 확정해 되살립니다(원장 **B-05**).

필드	타입	설명
`popType`	string	`general` | `contribution` . **표기는 일반 POP / 기여 POP**(원장 **B-01**)
`status`	string	`active` → `locked` → `expired` . **되돌아가는 전이는** `locked` → `active` **재접속 복구 1건뿐**
`lockAt`	timestamp	휴면 잠금 예정 시각. **일반 =** `lastActiveAt` **+ 180일 / 기여 = + 365일**
`graceUntil`	timestamp	잠금 후 **30일**. 이 안에 재접속하면 **전액 복구**
`expiredAt`	timestamp	Grace 종료 후 소멸 시각. 소멸은 원장에 **반대 부호 거래로 append**

소멸 기간 3종(180 · 365 · 30일)은 ● **운영값**이며 `config` 에서 조정합니다.

소멸이 잔액을 직접 깎지 않습니다. `fn-pop-expire` 가 lot 을 `expired` 로 바꾸고 **같은 트랜잭션에서 소멸 거래를 원장에 append** 한 뒤 잔액을 재계산합니다. 잔액만 깎으면 `잔액 == Σ 활성 lot == Σ 원장` 3자 등식이 깨집니다(§3.1).

재접속 복구는 소멸 전까지만 가능합니다. `expired` lot 은 되살리지 않습니다 — 이미 원장에 소멸 거래가 쌓여 있어 되살리면 재화가 늘어납니다. 대신 **잠금 30일 전 · 7일 전 · 잠금 발생 즉시 · 소멸 전** 네 번 알립니다(원장 **B-73** · §7.1 `fn-pop-expire-notify`).

2.6 `/rankings/country-{axis}` — 국가별 지표 랭킹 사전 집계 ● 신설

v0.3 §10 은 `rankings/{rankingId}` 를 「랭킹 화면은 이 문서 1건만 읽는다」로 정의해 두고 문서 형태를 비워 두었습니다. **지표 랭킹의 「국가별」 필터가 읽는 문서를 여기서 확정**합니다.

축마다 문서를 따로 둡니다. 한 문서에 5축을 담으면 화면이 어느 축을 그릴지 다시 분기해야 하고, 그 분기가 실제로 수치가 갈렸던 자리입니다.

필드	타입	값
문서 ID	string	`country-bpm` · `country-health` · `country-referral` · `country-community` · `country-contribution` — **FPI 5축**(§4.1)과 1:1
`axis`	string	축 이름
`rows`	array<map>	`{rank, code, name, value, change}` — 순위 오름차순. `code` 는 ISO 3166-1 alpha-2
`updatedAt`	timestamp	마지막 집계 시각. 화면이 갱신 시각으로 띄웁니다

- **집계 주기** — FPI 본 집계는 **1분**(§4.3)이고, **국가 집계는 전 멤버 스캔이라 5분으로 분리**합니다. 같은 주기로 묶으면 1분 배치가 멤버 전수를 매번 훑습니다. 두 주기 모두 ● 운영값입니다.

- **모집단** — §4.3 과 같습니다. **정식 행성(`stageNo >= 1`) 중 최근 7일 유효 활동이 있는 행성**의 멤버만 합산합니다(원장 **B-17**).
- **국가 기준** — 사용자 프로필의 국가를 멤버 문서에 `countryCode` 로 비정규화해 둔 값을 씁니다. **값이 없으면 `ZZ` (미지정) 버킷**으로 모아 총합이 새지 않게 합니다.
- **보안 규칙** — 기존 `match /rankings/{rankingId}` 가 `read: signedIn()` · `write: false` 로 이미 서 있어 규칙 변경이 필요 없습니다.
- **문서가 없을 때** — 화면은 빈 목록과 함께 **집계 미생성 상태를 표시**합니다. 조용히 다른 축의 목록을 대신 그리지 않습니다.

필드명 확정은 구현 브랜치 병합 시점입니다. 위 형태는 국가별 랭킹 구현이 실제로 쓰고 있는 형태이며, 병합과 함께 이 표를 확정본으로 고정합니다.

2.7 `/config/features` — 기능 노출 플래그 ● 신설

필드	타입	기본값	근거
`stationEnabled`	bool	`false`	**B-50** — STATION 은 1차에 사용자 탭과 Coming Soon 을 모두 비노출하되 **Route·데이터 구조를 유지해 앱 재배포 없이 활성화**한다

이 플래그가 B-50 의 구현체입니다. 탭 구성은 계약대로 5탭(견적서 v2.1)으로 만들어 두고, 노출 여부만 이 값 하나로 가릅니다. `true` **로 바꾸면 스토어 심사 없이 다음 앱 실행부터 STATION 이 뜹니다.**

`config` 컬렉션에 두는 이유는 읽기 시점 때문입니다. `firestore.rules` 의 `match /config/{configId}` 가 `allow read: if true` 이므로 **로그인 전 스플래시도 읽을 수 있고**, 쓰기는 `isMaster()` 뿐입니다. 탭 구성을 로그인 이후에 결정하면 첫 화면이 한 번 다시 그려집니다.

2.8 `/config/questCatalog` — Quest 분류 · 활동 마스터 ● 신설

한 화면에서 **카테고리 → 활동**을 고르고 **그 활동에 필요한 입력만** 뜨게 하는 구조가 확정되었습니다 (**C-01**). 분류 · 순서 · 노출이 운영 관리 대상이므로 앱에 열거형으로 박지 않고 마스터 문서로 둡니다.

필드	타입	값 · 규칙
`categories`	array<map>	`{key, label, order, visible}`. **건강 · 행성 활동 · 커뮤니티 · 특정 위치** 4종
`activities`	array<map>	`{key, categoryKey, order, visible, ugcAllowed, inputSchema}`
`activities[].key`	string	**Walk · Shake · Streaming · YouTube · Vote · SNS · Community · Location · QR · NFC** 10종. Quest Type 은 10종을 유지합니다(**C-03**)
`activities[].ugcAllowed`	bool	**사용자 생성에 열린 활동인가.** ● 운영값
`activities[].inputSchema`	array<map>	활동별 입력 필드 정의. 생성 화면은 이 스키마만 렌더링합니다

카테고리 ↔ 활동 매핑(C-01 확정값) — 건강 = Walk · Shake / 행성 활동 = Streaming · YouTube · Vote · SNS / 커뮤니티 = Community / 특정 위치 = Location · QR · NFC.

`ugcAllowed` **의 초기값은 저희가 정한 값입니다**(원장 **§19 S03**). 초기값은 Walk · Shake · YouTube · Streaming 을 운영 등록 전용(`false`)으로, Location · QR · NFC · Vote · SNS · Community 를 사용자 생성 허용(`true`)으로 둡니다. **위치 무관 QR / NFC 유형은 유효성 · 중복 · 복제 검증 명세가 확정될 때까지** `ugcAllowed = false` 입니다(**C-03**).

홈 카테고리와 FPI 축은 별개입니다. 「건강」 카테고리의 활동이 전부 HEALTH 축으로 가지 않습니다 — 축 매핑은 §4.1 이 정합니다(**C-35**).

구현 상태 — 미구현. 빌드 v0.3.0 은 Quest 유형이 함수별로 나뉘어 있고 분류 마스터가 없습니다.

2.9 `/quests/{questId}` — 생성 · 기간 · 인증 · 수정 ● 확정

필드	타입	값 · 규칙	근거
`fandomId`	string | null	팬덤 Quest 는 팬덤 ID, **일반 Quest 는** `null`. 참여 자격은 `fandomId == users.primaryFandomId` 또는 `fandomId == null` 일 때만 허용합니다	C-16
`rewardMode`	string	`none` | `deposit`. **Streaming · YouTube 시청은** `none` — 무보상 기본 참여 Quest 이며 POP 지급 원장이 생기지 않습니다. Music Shake 의 BPM 판정 보상은 별개 경로입니다	C-04
`link`	map	`{url, title, thumbnail, provider, fetchedAt, status}`. **원본 제목 · 썸네일 · 서비스명은 서버가 채우는 읽기 전용 값**입니다. `status` 는 `ok` | `preview_failed` | `invalid`	C-05
`userDescription`	string	**사용자가 쓴 설명.** `link.title` 과 **다른 필드**입니다 — 한 필드에 담으면 화면이 원본 제목과 사용자 문구를 구분할 수 없습니다	C-05
`publishedAt`	timestamp	공개 시각. 기간 계산의 기점이며 **최대 30일은 공개일을 1일째로** 셉니다	C-06
`endsAt`	timestamp	**UTC instant 로 확정 저장.** 기본은 `publishedAt + 24시간`, 종료 날짜를 고르면 **생성자 현지시간의 그 날짜 23:59:59** 를 UTC 로 굳힙니다	C-06 · G-04
`creatorTz`	string	생성자 시간대(IANA). `endsAt` 을 만든 근거로 남깁니다. 참여자 화면은 자기 기기 시간대로 **표시만** 바꿉니다	C-06
`verifyMethods`	array<string>	`gps` | `photo` | `qr` | `nfc` … **현장 방문 조건이면** `gps` **가 반드시 들어갑니다.** QR · NFC 는 독립 유형이면서 Location 의 부가 수단이기도 합니다	C-03
`repeat`	map	`{allowed, intervalHours, maxCount}`. 기본은 동일 Quest · 동일 사용자 **1회**. 반복 허용 시 **직전 유효 인증부터 초기 24시간** 간격이며, `maxCount = ⌊운영 기간 ÷ 간격⌋` 을 자동 산정하고 운영자는 **하향만** 합니다	C-12

기간 판정은 서버 시각으로 합니다(G-04). 기기 시각이나 시간대를 바꿔도 `endsAt` 은 움직이지 않습니다. **최대 30일 · 최소 1시간은** ● **운영값**입니다(C-06).

예치 한도 = 회당 기본 보상 × `repeat.maxCount` 입니다(C-12). **회사 부담 광고 보상은 이 한도 밖**이며 생성자 Reward Pool 에서 차감하지 않습니다(C-02 · §3.6).

`/quests/{questId}/edits/{editId}` — 공개 후 수정 이력 ● **신설**

필드	타입	설명
`editorUid` · `editedAt`	string · timestamp	누가 언제 고쳤는가
`field`	string	`title` **또는** `description` **만** 들어올 수 있습니다
`before` · `after`	string	고치기 전후 원문. append-only

공개 후 잠금 필드 — 기본 보상액 · 목표 · 인증 조건 · 대상 링크 · `endsAt` 입니다(C-07). 잠금 필드 변경 요청은 **서버가 거부**하며, 규칙에도 같은 경계를 세웁니다(§5). 핵심을 바꾸려면 새 Quest 를 만듭니다. **설명을 통한 조건 변경도 불가**이며, 이것은 스키마가 아니라 신고 · 검토 기준으로 잡습니다.

구현 상태 — 미구현. 빌드 v0.3.0 에는 사용자 생성 Quest 문서 스키마와 수정 이력이 없습니다.

2.10 `/questSubmissions/{submissionId}` — 제출 · 검토 · 지급 🟢 신설

제출 · 검토 · 승인 · 실제 지급을 한 필드로 겸하지 않습니다(G-03). 지급이 실패한 건이 화면에 「완료」로 뜨는 것을 막는 것이 이 분리의 목적입니다.

필드	타입	값 · 규칙	근거
`status`	string	**4단계** — `submitted`(검토 중) → `approved` → `paid`, 또는 `rejected`. **시간 경과 자동 승인 경로는 두지 않습니다**	C-08
`payoutStatus`	string	`pending` | `succeeded` | `failed`. `status` **와 별개 필드**입니다. 지급 실패는 `approved` + `failed` 이며 `paid` 로 올리지 않습니다	G-03
`fandomIdAtSubmit`	string | null	**제출 당시 대표 팬덤을 고정 저장.** 대표 팬덤이 바뀌어도 이 건의 검토는 계속되고, 승인되면 지급합니다	C-17 · G-01
`submittedAt`	timestamp	제출 시각. **주말 포함 24시간** 검토 목표의 기점이며, 경과 시 사용자 지연 안내와 운영 미처리 표시의 입력입니다	C-08 · G-01
`evidence`	array<map>	인증 자료(사진 · 좌표 · 코드 · 시각). **보존 대상**입니다	G-01
`reviewLog`	array<map>	`{actorUid, action, reason, at}`. 반려는 **사유와 재제출 가능 여부**를 함께 남깁니다	C-08 · G-01
`rejectReason` · `resubmitAllowed`	string · bool	반려 화면이 읽는 두 값	C-08

보완은 원 제출 건에 붙습니다(G-01). 새 문서를 만들지 않으므로 **변경 후 다른 팬덤의 새 활동이 보완으로 인정되는 경로가 없습니다.**

예약 잔액과 짝입니다 — 미결 제출 건의 보상 합이 §3.5 의 `reserved` 와 일치해야 합니다(**G-03**).

구현 상태 — 미구현. 빌드 v0.3.0 의 Quest 인증은 판정 즉시 지급하는 자동 검증형만 있고, **사람 검토 대기열이 없습니다.** 자동 검증 대상은 기존 방식을 그대로 둡니다(**C-08**).

Community Quest 반응 집계 — C-11
기본 조건은 응원글 1개 + 다른 회원의 유효 반응 1개입니다. 유효 반응을 세는 규칙이 스키마에 붙습니다.

규칙	세는 방법
본인 반응 제외	반응 문서의 작성자가 글 작성자와 같으면 집계에서 뺍니다
동일 회원 중복 제외	반응 문서 ID 를 반응한 회원의 `uid` 로 고정해, 같은 사람이 두 번 반응해도 1건입니다
삭제 · 어뷰징 확정 제외	글이나 반응이 삭제되거나 어뷰징으로 확정되면 집계에서 빠집니다
목표 · 인정 반응 종류	🟡 운영 설정입니다. Admin 에서 바꾸면 앱 재배포 없이 반영됩니다

지급 후 무효가 확정되면 정정합니다. 지급을 되돌리는 것이 아니라 **C-32 의 역분개 원칙**을 그대로 따릅니다(§2.14). **지급 후 무효 처리의 세부 명세는 🔵 미확정(S03)** 이며, **확인 전 유지 기준은 위 네 규칙**입니다.

구현 상태 — 구현 중. 빌드 v0.3.0 에 라운지 글 · 반응은 있으나 **Quest 목표 충족 판정과 연결되어 있지 않습니다.**

2.11 `eclipseRounds` — 회차 · 목표 · 결과 🟢 확정

경로는 팬덤 하위입니다(`fandoms/{fandomId}/eclipseRounds/{roundId}`**).** 회차 ID 는 `{KST 날짜}-{회차번호}` 로 고정해 **중복 발동을 문서 ID 충돌로 막습니다.**

필드	타입	값 · 규칙	근거
`startAt`	timestamp	회차 시작. **푸시는** `startAt − 3분` 입니다	C-22 · C-24
`windowSec`	int	방어 창(초). **기본 60초** · 🟡 운영값. 공통 창이며 **늦게 들어온 참여자에게 60초를 새로 주지 않습니다**	C-22 · C-25
`fpiAtStart`	number	**회차 시작 시점 FPI 스냅샷.** 감소량 산정의 유일한 입력입니다	C-23
`penalty`	number	`fpiAtStart × rate`. **rate 초기 0.25%** · 🟡 운영값	C-23
`appliedAt`	timestamp | null	**실패가 확정된 시각에만** 채워집니다. 값이 있으면 그 회차의 차감은 끝난 것이며, **회차당 중복 차감을 이 필드가 막습니다**	C-23 · G-02
`target`	int	`max(1, ceil(avgDau × rate))`. **초기 목표율은 7일 평균 DAU 의 5%** · 🟡 운영값	C-25 · C-26
`result`	string	`success` | `failed` | `cancelled_error`	C-22 · C-28
`cancelReason`	string | null	목표나 결과를 정확히 판정할 수 없을 때의 사유. 운영자용 기록이며 사용자에게는 취소 안내가 나갑니다	C-28

`/…/eclipseRounds/{roundId}/participants/{uid}` 🟢 **신설**

필드	타입	값 · 규칙
`uid`	string	**문서 ID 가** `uid` 입니다. 같은 사용자가 여러 단말에서 참여해도 **1명**입니다(**G-04**)
`validShakes`	int	회차 안에서 누적한 유효 Shake. **60회 이상이면 고유 참여자 1명으로 인정**하며, 120회를 흔들어도 1명입니다

성공 · 실패 판정 — 고유 참여자 수가 `target` 이상이면 `success` 이고 **FPI 는 움직이지 않습니다.** 미달로 창이 닫히면 `failed` 이며 **그 시점의 FPI 에서** `penalty` **를 1회 차감**합니다 (**C-22 · C-23**). **방어 중에 늘어난 활동분은 그대로 유지**되고, **과거 sFPI 는 건드리지 않습니다.**

`config/eclipse` · `fandoms.eclipseSchedule[]` **— 발생 일정** 🟢 **신설**

필드	타입	값 · 규칙
`config/eclipse.schedule`	array<map>	공통 일정. `{hhmm, tz}` 배열이며 **기본값은 KST 07:50 · 17:50**(푸시 07:47 · 17:47)
`fandoms.eclipseSchedule`	array<map>	팬덤별 일정. **있으면 우선**, 없으면 공통값
`config/eclipse.rate` · `windowSec` · `targetRate`	number · int · number	🟡 운영값 — 감소율 0.25% · 창 60초 · 목표율 5%

하루 2회는 고정 제한이 아닙니다(**C-24**). 배열 길이가 곧 하루 회차 수이며, 3회 설정도 가능합니다. 일정을 바꾸면 **푸시 시각과 종료 시각이 함께 다시 계산**됩니다.

구현 상태 — 정정 필요(빌드 v0.3.0). 다섯 가지가 이번 확정과 다릅니다.

#	빌드 v0.3.0	v1.2 확정	근거
1	회차 활성화 시점에 **FPI 를 먼저 깎고** 방어 성공 시 되돌립니다	**성공 시 감소 없음 · 실패 확정 시에만 1회 차감**	**C-22 · R09**
2	사전 알림 **300초** 전	**3분(180초) 전**	**C-22 · R09**
3	기본 발생 시각 **13:00 · 21:00**, 하루 2회 고정	**07:50 · 17:50** 기본 2회 · 고정 제한 아님 · 팬덤별 횟수·시각 설정	**C-24 · R09**
4	방어 목표 = **팬덤 공동 누적 100,000 Shake**	**고유 참여자 수 ≥** `max(1, ceil(7일 평균 DAU × 5%))` , 개인 인정은 60회	**C-25 · C-26 · R09**
5	방어 창 **30분**	**기본 60초**	**C-22**

컬렉션 명칭도 맞춥니다. 빌드 v0.3.0 은 회차를 `eclipses` , 참여자를 `defenders` 로 두고 있습니다. v1.2 확정 명칭은 `eclipseRounds` · `participants` 이며, 전환은 §3.7 마이그레이션 대사 항목입니다. **방어 참여가 POP 원장을 건드리지 않는 것**(**C-20**)과 **같은 사용자 복수 단말을 1명으로 세는 것**(**G-04**)은 빌드 v0.3.0 에서 이미 성립합니다.

2.12 `/fandoms/{fandomId}/fandomDailyStats/{dayKey}` — 집계 분모 🟢 신설

필드	타입	값 · 규칙	근거
`dau`	int	그 날짜의 팬덤 유효 활동 사용자 수. **Eclipse 목표는 최근 7일 평균**	**C-21**
`active30`	int	최근 30일 Active. **Zap 가중치 전용**이며 Eclipse 와 **다른 집계**입니다	**C-21**

두 분모를 한 필드로 겸하지 않습니다. 기간이 다르고 쓰이는 자리가 달라, 하나로 묶으면 한쪽 기간을 바꿀 때 다른 쪽이 따라 움직입니다. 두 기간 모두 🟡 운영값입니다.

신생 팬덤(**C-27**) — 집계 시작일은 **팬덤 생성일**입니다. 완료 집계일이 1~6일이면 **그 날짜만의 평균**, 7일 이상이면 최근 7일 평균입니다. **생성 후 활동이 0명인 날은 분모에 포함**하고, 완료 집계일이 없는 생성 당일은 **임시 목표 1명**으로 진행합니다.

집계 누락 · 오류(**C-28**) — 누락을 **활동 0명으로 대체하지 않습니다.** 판정할 수 없는 회차는 `result = cancelled_error` 로 취소하고 FPI 를 건드리지 않습니다.

구현 상태 — 미구현. 빌드 v0.3.0 에 일별 팬덤 활동 집계 컬렉션이 없습니다.

2.13 `/productCatalog/{productId}` — 모금 기간 · 집행 기간 🟢 확정

필드	타입	값 · 규칙	근거
`recommendedDays`	int	**권장 모금 기간.** 캠페인 생성 화면에 자동 입력됩니다	**C-14**
`minDays` · `maxDays`	int	운영자가 정한 범위. **날짜 선택기가 이 범위 밖을 거부**합니다	**C-14**
`executionLeadDays`	int	**집행 준비 기간.** 모금 기간과 **별개 필드**입니다	**C-14**

모금 기간과 집행 기간을 한 값으로 겸하지 않습니다. Governance 는 기본 72시간을 유지합니다 (**B-37**). **상품별 초기 일수 제안표는** 🔵 **미확정(S03)** 이며, **확인 전 유지 기준은 Catalog 권장 기간 자동 입력**입니다.

구현 상태 — 미구현. 빌드 v0.3.0 의 캠페인 상품에 기간 필드가 없습니다.

2.14 Growth 이벤트 — 멱등 · 귀속 · 역분개 ● 확정

행성 성장은 POP 원장과 다른 계층입니다. 원장 행이 생길 때마다 성장이 붙는 것이 아니라, **성장 원천으로 지정된 이벤트**에만 growth 이벤트 1건이 붙습니다(**C-29**).

필드	타입	값 · 규칙	근거
`sourceEventId`	string	**원 이벤트 ID 를 그대로 쓴 멱등 키.** 문서 ID 로 삼아 `create()` 하므로 재시도 · 응답 유실 · 중복 요청에도 성장이 두 번 붙지 않습니다	**G-02**
`sourceType`	string	`quest_reward` · `company_ad_bonus` · `zap` · `funding` · `deposit` · `refund` · `fee` … 성장 부여 여부를 이 값으로 가릅니다	**C-29 ~ C-31**
`fandomIdAtPayout`	string	**지급 시점의 대표 팬덤을 고정 저장.** 재시도는 현재 대표 팬덤을 다시 읽지 않고 이 값을 씁니다	**C-33 · G-02**
`amount`	int	성장 반영량. 보상 획득 단계는 지급량, 기여 단계는 기여량입니다	**C-29**
`reversalOf`	string | null	**역분개 대상 이벤트 ID.** 값이 있으면 이 행은 정정 행입니다	**C-32**
`reason`	string | null	역분개 사유. **수치와 사유를 함께 남깁니다**	**C-32**

두 단계로 성장합니다(C-29). 보상을 **받을 때** 지급량만큼, 그 POP 을 **기여할 때** 기여량만큼 붙습니다. 10 POP 을 받고 그대로 Zap 하면 합계 20 입니다. **의도된 두 단계이므로 중복으로 보고 지우지 않되, 같은 지급 이벤트의 기술적 재처리는 `sourceEventId` 가 막습니다.**

정정은 역분개로만 합니다(C-32). 원 이벤트를 고치지 않고 반대 부호 행을 씁니다. Quest 가 정상 종료 · 취소된 것만으로는 회수하지 않고, **허위 참여 · 중복 지급 · 오지급이 확정된 건의 잘못 반영분만** 되돌립니다. **의심 단계에서는 아무것도 차감하지 않습니다.**

구현 상태 — 미구현. 빌드 v0.3.0 은 팬덤 · 멤버 문서의 누적 기여 POP 을 직접 올릴 뿐, 성장 이벤트 계층과 역분개 경로가 없습니다.

✦ 3. POP 원장 — 구현 반영

설계는 v0.3 §3 그대로입니다. **구현이 그 설계를 어떻게 지키는지**를 적습니다.

3.1 네 가지 불변식

불변식	어떻게 지키는가	검증
클라이언트 단독 지급 0건	보안 규칙이 `wallet` · `popLots` · `popTransactions` 쓰기를 전면 차단. Admin SDK 만 우회	규칙 테스트 118건 + 음성 통제 1건
멱등	`txId = SHA-256(멱등키)[0..20]` · 항상 `create()`	원장 테스트 — 지급·차감 재시도
append-only	환불이 원 거래를 고치지 않고 반대 부호 새 행을 씀	원장 테스트 — 원 금액 보존 확인
혼합 차감 금지	모든 호출이 `popType` 을 지정. **한 거래가 두 재화를 건드릴 수 없음.** 겸용 사용처는 `popType` 별 **거래 2건을 한 트랜잭션으로 묶어** 처리(§3.2)	원장 테스트 — 잔액 분리 · 겸용 차감 시 거래 2건 생성 확인
소멸도 원장을 지남	`fn-pop-expire` 가 lot 을 `expired` 로 바꾸면서 **같은 트랜잭션에서 소멸 거래를 append.** 잔액만 깎는 경로 없음(§2.5)	원장 테스트 — 소멸 후 3자 등식 유지 확인
멱등 키 = 원 이벤트 ID ● **v1.2**	지급 · 반환 · 성장의 재처리는 **원래 이벤트와 연결해 처리 결과를 확인한 뒤 미처리분만** 진행. 응답 유실 · 재접속 · 중복 요청이 보상 · 반환 · 성장을 두 번 반영하지 않음(원장 **G-02**)	같은 이벤트 ID 재요청 시 원장 행 · growth 행 증가 **0**
지급 실패 ≠ 지급 완료 ● **v1.2**	제출 · 검토 · 승인 · 실제 지급을 각각 다른 필드로 둠(§2.10 `status` · `payoutStatus`). 지급 실패 건을 「완료」로 표시하는 경로 없음(원장 **G-03**)	지급 실패 건이 사용자 화면에 「완료」 **0건**
잔액 등식 ● **v1.2**	`총 보유 == 예약 + 사용 가능`. 예약분은 미결 제출 건의 보상 합과 일치(§3.5 · 원장 **G-05**)	대조 항목 — 등식 위반 시 `mismatch` 표시
Pool 등식 ● **v1.2**	개인 Quest Reward Pool 은 `예치 == 지급 + 예약 + 반환`. 정상 종료 시 미사용분을 **즉시** 반환하며, **검토 지연만으로 예약을 풀지 않음**(원장 **C-09**)	종료 직후 반환 원장 존재 · 검토 대기 건의 예약 유지

3.2 차감 순서 — 규범이 확정되었습니다

v1.0 은 「차감 순서에 규범이 없다」고 적고 원장에 `lotOrder: 'grantedAt-asc-nonnormative'` 표시를 남겼습니다. 별첨 B 가 규범을 확정했습니다(원장 **B-06**).

두 단계로 차감합니다.

단계	규칙	근거
① **재화 타입**	**겸용 사용처는 일반 POP 을 먼저 소진하고 부족분만 기여 POP** 에서 차감	**B-06**
② **같은 타입 내부 lot**	**선입선출** — `grantedAt` 오름차순	**B-06** · 원장 추적성

사용처별 허용 타입이 다릅니다.

사용처	허용 타입	원장
Funding 후원	**기여 POP 만.** 일반 POP 으로는 후원 불가	**B-33**
Campaign 등록비 · Quest 수수료 · Q&A 현상금 예치	**겸용**(일반 우선)	**B-57** · **B-30** · **B-39**
Governance 발의 등록비 100 POP	**일반 POP 전용** — 기여 POP 으로 대체하지 않음. `popType = general` 만 허용	**B-38** · **G-05** · **R03**

겸용 사용처에서 두 타입이 함께 쓰이면 거래를 2건 만듭니다. 「혼합 차감 금지」 불변식(§3.1)은 그대로이며, **한 트랜잭션 안에서 `popType` 별로 각각 append** 해 원자성만 함께 보장합니다. 잔액 대조는 타입별로 따로 성립합니다.

`lotOrder` **표시는 유지합니다.** 값을 `'general-first-fifo'` 로 바꾸고, **v1.0 시절의 `grantedAt-asc-nonnormative` 표시가 붙은 과거 거래와 구분**할 수 있게 둡니다. 규범 확정 전후를 원장에서 가려낼 수 있어야 사후 정산 근거가 남습니다.

FEFO 는 쓰지 않습니다. 상세정의서 v1.6.1 6장이 Earliest Expiry First 를 정했으나 **별첨 B 가 「일반 POP 우선」으로 대체**했습니다. 소멸 기한이 아니라 **재화 타입이 우선순위를 정합니다**(원장 **B-06**).

3.3 잔액과 lot 이 어긋난 경우

지갑 잔액은 충분한데 lot 합이 모자란 상태가 나올 수 있습니다. 이때 구현은 **차감을 진행하지 않고** 지갑을 `reconcileStatus='mismatch'` 로 표시한 뒤 `E-POP-02` 로 끊습니다.

차감을 강행하지 않는 이유 — 이미 원장이 어긋나 있다는 뜻이므로, 그 위에 거래를 얹으면 원인을 찾을 수 없게 됩니다. **감지 → 정지 → 확인 → 수동 정정** 순서이며, 자동 정정은 어느 단계에도 없습니다.

3.4 3자 대조 — 구현

`fn-pop-reconcile` 이 일 1회 전 사용자를 훑고 **아홉 가지**를 봅니다. v1.0 의 여섯 가지에 **Passport Tier · User Level 분리 검증**과 **sFPI 누적 · 소멸 정합**이 더해졌습니다.

검증	식
일반 POP 잔액 ↔ lot	`generalBalance == Σ popLots[general, active].remaining`
일반 POP 잔액 ↔ 원장	`generalBalance == Σ popTransactions[general].amount`
기여 POP 잔액 ↔ lot	〃
기여 POP 잔액 ↔ 원장	〃
통계 ↔ 원장	`totalContributedPop == Σ 기여 차감 − Σ 환불`
Passport Tier ↔ 통계	`passportTier == tierOf(userLevel, 활동기간, badges, totalContributedPop)` — **강등이 없으므로 낮아진 값이 나오면 불일치로 표시**(원장 **B-65**)
User Level ↔ EXP	`userLevel == levelOf(userExp)` — **Passport Tier 와 별개 체계이므로 따로 검증**(원장 **B-66**)
sFPI ↔ Snapshot	`sfpi == Σ dailyFpiSnapshots.value` — **환산 계수 없이 단순 누적인지**(원장 **B-54**)
소멸 정합	`Σ popLots[expired].amount == Σ popTransactions[expire].amount` — 소멸이 원장을 지났는지(§2.5 · 원장 **B-05**)

불일치를 자동으로 고치지 않습니다. 표시만 하고 값은 그대로 둡니다. 테스트가 이것을 직접 확인합니다 — 지갑을 원장 밖에서 999로 바꾼 뒤, 대사가 잡아내면서도 999를 999로 남겨 두는지를 봅니다. `passportTier` 와 `userLevel` **도 각각 위조해 서로를 오염시키지 않고 따로 잡히는지 확인합니다** — 두 체계가 실제로 분리되어 있는지를 대사가 증명하는 자리입니다.

전수 스캔의 한계 — 사용자가 늘면 일 1회 전수 훑기는 실행 시간을 넘깁니다. 샤딩은 9주차 부하 설계에서 다룹니다. **대조 자체가 계약 검수 항목이라 먼저 동작하게 두었습니다.**

3.5 잔액 3필드 — 총 보유 · 예약 · 사용 가능 ● v1.2 신설

잔액 하나로는 「예치했지만 아직 나가지 않은 POP」을 표현할 수 없습니다. 개인 Quest Reward Pool 예치, 검토 대기 중인 보상, Q&A 현상금이 모두 이 자리에 있습니다(**G-05 · C-09**).

필드	타입	뜻
`balance`	int	**총 보유.** 원장 합계와 같아야 합니다(§3.4)
`reserved`	int	**예약.** 예치 · 검토 대기 보상 · 현상금처럼 **묶여 있는 몫**
`available`	int	**사용 가능.** 지금 쓸 수 있는 몫

`balance == reserved + available` **이 불변식입니다**(**G-05**). 세 필드는 **일반 POP · 기여 POP 각각** 따로 성립하며, 혼합 차감 금지 불변식(§3.1)은 그대로입니다.

Reward Pool 등식(**C-09**) — `예치 == 지급 + 예약 + 반환` . 정상 종료 시 **지급 · 예약되지 않은 미사용분을 즉시 자동 반환**합니다. 검토 · 보완 · 재검토 예약분은 유지하고, 승인되면 지급 · 최종 반려면 반환입니다. **검토가 늦어졌다는 이유만으로 예약을 풀지 않습니다.**

구현 상태 — 정정 필요(빌드 v0.3.0). 지갑에 타입별 잔액 1필드씩만 있고 예약 · 사용 가능이 없습니다. 필드 이름도 활동 / 지원 계열이라 **일반 / 기여 표기**(**B-01**)와 어긋납니다.

3.6 Growth POP 원천과 제외 ● v1.2 신설

구분	`sourceType`	성장	근거
Quest 보상 지급	`quest_reward`	**부여** — 지급량만큼	**C-29**
사용자 생성 Quest 광고 추가 보상	`company_ad_bonus`	**부여** — 지급 시 10, 이후 기여 시 기여량. **회사 부담 고정형**이라 생성자 Reward Pool 에서 차감하지 않습니다	**C-02 · C-31**
Zap · Funding 후원 등 기여	`zap` · `funding`	**부여** — 기여량만큼	**C-29**
Reward Pool 예치	`deposit`	**없음**	**C-30**
미사용분 반환	`refund`	**없음**	**C-30**
등록비 · 취소 · 환불 수수료	`fee`	**없음**	**C-30**

모든 POP 이동에 성장을 붙이는 규칙이 아닙니다(**C-29**). 위 표에 없는 이동은 성장 원천이 아닙니다. **광고 보상의 성장 규칙이 다른 광고 · 구독 보너스로 자동 확대되지도 않습니다**(**C-31**) — Focus 의 일 40 기여 POP 등에 같은 규칙을 적용할지는 🔵 미확정(**S01**)입니다.

귀속은 지급 시점의 대표 팬덤입니다(C-33). 일반 Quest 보상은 지급 순간의 `primaryFandomId` 를 growth 이벤트에 고정 저장하며, 재시도는 저장된 값을 씁니다. **팬덤 Quest 지연 지급의 귀속은 🔵 미확정(S01)** 이며, 확인 전에는 제출 건의 `fandomIdAtSubmit` 을 그대로 두고 지급합니다.

발의비 · 사유 코드와 재화 타입(G-05 · R03) — Governance 발의 등록비는 **일반 POP 100** 기준이며 기여 POP 으로 대체하지 않습니다. 차감 함수가 **사유 코드별 허용 재화 타입을 검증**해야 하며, 빌드 v0.3.0 은 호출자가 넘긴 재화 타입을 그대로 쓰므로 **정정 필요**입니다.

3.7 lot 소진 규칙 · 마이그레이션 대사 — R17 🟢 v1.2 신설

lot 소진 규칙(B-06) — ① 겸용 사용처는 **일반 POP 을 먼저** 소진하고 부족분만 기여 POP, ② 같은 타입 안에서는 **선입선출**(`grantedAt` 오름차순), ③ 소진된 lot 은 `remaining = 0` 으로 남기고 지우지 않습니다. 원장에는 `lotOrder = 'general-first-fifo'` 를 남겨 규범 확정 전후를 구분합니다(§3.2).

빌드 v0.3.0 은 `lotOrder` 가 아직 규범 없음 표시(`grantedAt-asc-nonnormative`)입니다 — 정정 필요. 차감 순서 자체는 `grantedAt` 오름차순이라 같은 타입 안에서는 이미 선입선출이지만, **재화 타입 우선순위(일반 → 기여)가 구현되어 있지 않습니다.**

마이그레이션 대사 — v1.2 로 바뀌는 데이터는 다음 다섯 건이며, **원장 데이터는 건드리지 않고 파생 · 인덱스 · 명칭 계층만** 옮기므로 3자 대조 정합성이 유지됩니다.

#	대상	옮기는 방법	대조
1	지갑 잔액 1필드 → **3필드**	기존 잔액을 `balance` 로 옮기고 `reserved = 0` · `available = balance` 로 초기화한 뒤, 미결 제출 건을 훑어 `reserved` 를 채웁니다	`balance == reserved + available` · `balance == Σ 원장`
2	등급 산출 입력 **누적 기여 POP → User EXP**	`userExp` 를 신설하고 기존 5등급 값은 **버립니다.** `passportTier` 는 그대로 두고 `userLevel` 만 다시 산출합니다	`userLevel == levelOf(userExp)` · `passportTier` 불변
3	`users.fandomId` → `primaryFandomId`	값을 그대로 옮기고 `primaryChangedAt` 은 비워 둡니다(첫 변경까지 대기 없음)	이동 건수 == 기존 필드 보유 사용자 수
4	행성 문서 ID `uid` **→ 팬덤 고정 ID**	팬덤에 행성이 2건 이상이면 **가장 이른 것을 남기고** 나머지의 누적 기여 POP 을 합산해 옮긴 뒤 `ownerUid` 를 보존합니다	팬덤당 행성 **1건** · 합산 전후 누적 기여 POP 합 동일
5	`eclipses` · `defenders` → `eclipseRounds` · `participants`	진행 중 회차가 없는 시각에 이름만 옮깁니다. **선차감된 FPI 가 남아 있으면 먼저 되돌립니다**	이관 전후 FPI 합 동일 · 미종료 회차 **0건**

마이그레이션은 되돌릴 수 있어야 합니다. 각 단계는 원본 필드를 지우지 않고 남겨 두었다가, 대조가 통과한 뒤 다음 판본에서 정리합니다.

✦ 4. FPI 집계

4.1 지표 5종 — 확정

지표	`fpiAxes` 키	집계 입력
BPM	`bpm`	YouTube 완청 · Music Shake 등 검증된 음악 활동
HEALTH	`health`	만보기 등 검증된 신체 활동
REFERRAL	`referral`	초대 · 유입 전환
COMMUNITY	`community`	라운지 · INSIDE · 공유 등 유효 커뮤니티 활동
CONTRIBUTION	`contribution`	POP 후원 · 캠페인 · Q&A 등 서버검증 기여 원장

총점 `fpiScore = Σ (axisValue[i] × weight[i])` **, 초기** `weight` **는 전부 1.0** — 정책 설계서 §P-03 「초기 MVP는 단순 합산」.

가중치 합이 1일 필요가 없습니다. 단순 합산에서는 합이 5입니다. 구현은 합이 1인지를 검사하지 않고 **축이 빠졌거나 음수인지**를 봅니다. 음수 가중치는 활동할수록 점수가 깎이는 뒤집힌 랭킹을 만들기 때문입니다. 검사에 걸리면 집계를 멈추고 **직전 스냅샷을 유지**합니다.

4.1.1 축 값의 원천은 활동 이벤트입니다 🟢 **v1.2 확정**
FPI 축 값은 지급 POP 이 아닙니다(C-34 · R07). **실제로 인정된 활동**이 원천이며, 같은 활동 · 같은 조건이면 **생성자가 정한 보상액이나 광고 지급량이 달라도 점수가 같습니다.** 보상 750 POP 인 Quest 와 1 POP 인 Quest 의 같은 활동은 같은 점수입니다.

무보상 활동도 산정됩니다. 무보상 기본 참여 Quest(**C-04**)의 참여 기록은 POP 원장에 아무것도 남기지 않지만 FPI 원천에는 남습니다. **지급 POP 을 축 값으로 합산하는 경로를 두지 않습니다.**

축별 원천 이벤트(C-35 확정)

축	원천 이벤트
BPM	Music Shake · YouTube 완청 등 검증된 음악 활동
HEALTH	**걸음 · 운동** 등 검증된 신체 활동
REFERRAL	초대 · 유입 전환
COMMUNITY	라운지 · INSIDE · 공유 · **출석 · 성지 / 생일 팝업 / 응원 현장 방문**
CONTRIBUTION	POP 후원 · 캠페인 · Q&A 등 서버검증 기여

출석과 현장 방문은 COMMUNITY 입니다. 저희는 요청서에서 HEALTH 를 권고드렸고, 회신이 COMMUNITY 로 확정했습니다(**C-35**). **홈 화면의 카테고리(건강 · 행성 활동 …)와 FPI 축은 별개 체계**입니다 — 「특정 위치」 카테고리의 방문 활동이 COMMUNITY 축으로 갑니다.

환산계수는 🔵 미확정(S01) 입니다. 확인 전 유지 기준은 **축별 계수 초기값 1.0** 이며, 새 계수 · 일일 성장 상한 · 조합 제한을 저희가 만들지 않습니다.

구현 상태 — 정정 필요(빌드 v0.3.0). 두 가지가 확정과 다릅니다.

#	빌드 v0.3.0	v1.2 확정	근거
1	걸음 Quest 의 축 값으로 **지급된 POP 수량**을 기록합니다	**활동량 기준** — 지급 POP 을 축 값으로 쓰지 않습니다	**C-34 · R07**
2	**출석 · 방문 활동이 FPI 축에 연결되어 있지 않습니다**(축 근거가 없어 비워 두었던 자리)	**COMMUNITY 축 이벤트로 기록**	**C-35**

4.2 최근 7일 Rolling Window

`occurredAt >= now − config.fpi.windowDays` 인 검증 완료 이벤트만 합산합니다. 기본 7일이고 **별도 고정 감쇠를 두지 않습니다** — 창 밖 이벤트가 자연히 빠지므로 감쇠를 겹치면 두 번 깎입니다.

세 가지 감소를 구분합니다(**R08**).

감소	무엇인가	원장에 남는 것
7일 창 만료	이벤트가 창 밖으로 나가 합산에서 빠지는 것. **정상 동작**이며 정정이 아닙니다	원천 이벤트는 그대로 보존
무효 정정	허위 · 중복 · 오지급이 확정되어 해당 활동을 **무효로 되돌리는 것**	원 이벤트 참조 · 사유 · 수치가 붙은 정정 행(**C-32**)
Eclipse 실패 차감	회차 실패 확정 시 **FPI 에서 1회 차감**하는 것(**C-22 · C-23**)	회차 ID 로 멱등 · `appliedAt` 기록 · **sFPI 불변**

셋을 한 경로로 묶지 않습니다. 창 만료를 정정으로 적으면 정상 동작이 오류처럼 보이고, Eclipse 차감을 무효 정정에 섞으면 **과거 sFPI 까지 재산정하는 경로가 열립니다.**

4.3 갱신 주기

별첨 B 가 「원천 이벤트는 즉시 적재하고 FPI/랭킹은 기본 1분 주기로 집계」로 확정했습니다(원장 **B-15**). Cloud Scheduler 의 최소 주기도 1분이라 **1분 스케줄**로 둡니다. 화면에는 마지막 갱신 시각을 띄웁니다. 주기는 🟡 운영값입니다.

원천 이벤트 적재는 집계 주기와 분리되어 있습니다. 사용자 행동은 Raw Event 로 즉시 append 되고, 1분 배치는 그것을 읽어 축별로 합산합니다. **집계가 늦어져도 원천이 유실되지 않으므로 과거 재계산이 가능합니다.**

랭킹 모집단은 정식 행성 중 최근 7일 유효 활동이 있는 행성입니다(원장 **B-17**). `stageNo == 0` 인 성운은 쿼리에서 제외합니다.

> **[v1.2 정정 · C-19]** v1.1 은 **동률을 최근 활동 시각이 빠른 쪽 상위**로 두고 확인을 요청드렸습니다(원장 §15-②). 회신이 이 안을 **기각**하고 **1 · 1 · 3 공동 순위**로 확정했습니다. **최근 활동 시각으로 순위를 가르는 경로는 두지 않습니다**(§4.4).

Daily FPI Snapshot 은 하루 1회 그대로 sFPI 에 누적합니다(원장 **B-54**). 환산 계수를 두지 않으므로 `sfpi == Σ dailyFpiSnapshots.value` 가 성립하고, §3.4 대조가 이것을 확인합니다.

4.4 랭킹 쿼리 — 축마다 다른 쿼리

지표 랭킹은 축마다 다른 쿼리를 씁니다. 행성 랭킹은 `fandoms` 를 축 값 (`fpiAxes.<축>`) 내림차순으로, 사용자 랭킹은 `fandoms/{id}/members` 를 같은 방식으로 정렬합니다. 축을 인자로 받고도 쓰지 않으면 **어느 지표로 들어가도 같은 목록**이 뜹니다.

국가별 필터만은 실시간 쿼리가 아니라 사전 집계 문서 1건을 읽습니다 — `rankings/country-{축}` (§2.6). 국가 합산은 멤버 전수를 훑어야 해서 화면에서 매번 계산할 수 없습니다.

모집단 판정은 서버에서 끝내고 `rankingEligible` 불리언 하나로 내립니다. 「최근 7일 유효 활동」을 화면이 `lastActiveAt` 부등호로 거르면 정렬 키(`fpiAxes.<축>`)와 부등호 필드가 달라 **Firestore 가 질의 자체를 거부**합니다.

동률은 공동 순위입니다(C-19). 실제 산정값이 같으면 **같은 순위를 주고 다음 순위를 건너뜁니다** — 1 · 1 · 3 입니다. **표시 반올림만 같고 실제 값이 다르면 실제 값 순**이므로, 정렬은 언제나 반올림 전 값으로 합니다. **공동 순위 안의 화면 배치는 결정적 정렬키**(예: `fandomId`)로 고정해 다시 조회해도 순서가 흔들리지 않게 합니다.

순위는 서버가 매겨 `rank` 로 내립니다. 화면이 목록을 받아 자기 자리에서 번호를 붙이면 공동 순위가 1 · 2 · 3 으로 다시 매겨집니다.

> **정밀도와 정렬키 명세는 🔵 미확정(S01)** 입니다. **확인 전 유지 기준** — 산정값은 저장 정밀도 그대로 비교하고, 2차 정렬키는 `fandomId` 오름차순입니다.

4.5 Eclipse 실패 차감과 FPI

성공한 회차는 FPI 를 건드리지 않습니다(C-22). 실패가 확정된 시점에 **그때의 FPI 에서** `penalty` 를 1회 차감하며, `penalty` 는 **회차 시작 시점 FPI × 초기 0.25%** 입니다(**C-23**).

선차감 후 복구 방식은 쓰지 않습니다. 먼저 깎고 방어에 성공하면 되돌리는 구조는 사용자 눈에 이미 깎인 것으로 보이고, 회복 실패 시 복구되지 않은 차감이 남습니다.

방어 중에 늘어난 활동분은 유지되고, **과거 sFPI 는 재산정하지 않습니다(C-23** · §4.2). 같은 회차에서 두 번 차감되지 않도록 **회차 ID 로 멱등**을 겁니다(**G-02**).

✦ 5. 보안 규칙

v0.3 §7 그대로입니다. 실물은 `firestore.rules` 이며 매치 블록마다 근거 주석이 붙어 있습니다.

세 원칙

1. POP·FPI 집계·원장 필드는 클라이언트가 쓸 수 없다. 전부 Functions 가 쓴다.
2. 개인 리소스는 본인만 읽고, 쓰기는 화이트리스트 필드로만 허용한다.
3. 운영자 권한은 커스텀 클레임 `role` 로만 판정한다. **문서 조회로 판정하지 않는다** — 판정 1회당 읽기 비용이 붙고 우회 표면이 는다.

규칙이 할 수 없는 것 — 규칙은 거부만 할 수 있고 로그를 남기지 못합니다. 그래서 Admin 쓰기는 규칙이 아니라 `fn-admin-write` 라는 단일 관문을 지나며, 거기서 `adminActionLogs` 에 적재합니다.

5.1 v1.2 에서 더한 경계

경계	규칙 · 함수가 하는 일	근거	구현 상태
공개 후 잠금 필드	공개된 Quest 의 **기본 보상액 · 목표 · 인증 조건 · 대상 링크** · `endsAt` 변경 요청을 **전건 거부**합니다. 규칙은 생성자 쓰기를 `title` · `description` 으로만 좁히고, 함수는 편집 이력(§2.9 `edits`)을 append 합니다	**C-07**	**미구현**
이메일 로컬 파트 매칭 금지	다른 도메인 이메일의 **앞부분만으로 같은 소유자로 판단하지 않습니다.** 제공자가 인증한 **전체 이메일을 정규화해 일치할 때만** 연결합니다	**G-07 · R01**	**정정 필요** — 빌드 v0.3.0 의 계정 연결 판정이 **로컬 파트만** 비교합니다
사전예약 서비스키 서버 보관	사전예약 기록 조회는 **서버 함수만** 수행합니다. 서비스키는 **클라이언트 번들에 들어가지 않습니다**	**C-42**	**미구현** — §7 `fn-prereg-lookup` 신설

규칙만으로는 막을 수 없는 것이 있습니다. 잠금 필드 거부는 규칙으로도 거르지만, **편집 이력을 남기는 일은 규칙이 하지 못합니다**(규칙은 거부만 하고 쓰기를 못 합니다). 그래서 공개 후 수정은 함수를 지나게 하고, 규칙은 **직접 쓰기를 닫아** 우회로를 없앱니다.

이메일 매칭은 규칙이 아니라 함수와 앱의 판정 코드에 있습니다. 정정 대상은 계정 연결 판정 경로이며, 정정 판본은 v0.4.0 입니다(**R01**).

✦ 6. 인덱스

실물 대사 결과 — 복합 인덱스 63건 · 필드 예외 6건(빌드 v0.3.0 · **R17**).

> **v1.1 본문이 이 자리를 「51건」으로 적었습니다.** 실물은 63건이며 그동안 랭킹 · 브리핑 · 이클립스 인덱스가 더해졌습니다. 옮겨 적기 오류라 v1.2 에서 실물 수로 바로잡습니다.

FPI 축 이름 변경이 인덱스에 영향을 주지 않습니다. 축은 `fpiAxes` 맵 안의 키이고, 인덱스 대상은 `fpiScore` 스칼라입니다.

랭킹 쿼리를 받는 인덱스(구현 완료 · 미배포) — `fandoms` 의 `rankingEligible` + `fpiAxes.<축>` 복합 인덱스가 **5축 각각** 서 있습니다. 모집단 판정을 불리언 하나로 내린 이유가 여기 있습니다 (§4.4). 국가별은 `fandoms` 의 `countryCode` + `fpiScore` 이며, 사전 집계 문서(§2.6)를 읽는 화면은 인덱스를 쓰지 않습니다.

v1.2 가 더해야 하는 인덱스(미구현 · v0.4.0)

컬렉션	필드	받는 쿼리	근거
`questSubmissions`	`status` + `submittedAt`	Admin 미처리 목록 · 24시간 경과 건	**C-08**
`questSubmissions`	`fandomIdAtSubmit` + `status` + `submittedAt`	팬덤별 검토 대기열	**C-17**
`quests`	`fandomId` + `endsAt`	대표 팬덤 Quest 목록 · 마감 임박	**C-16 · C-06**
`quests`	`rewardMode` + `publishedAt`	무보상 참여 Quest 목록	**C-04**
`eclipseRounds` (collection group)	`result` + `startAt`	회차 이력 · 취소 회차 조회	**C-22 · C-28**
`fandomDailyStats`	`dayKey` 내림차순	최근 7일 · 30일 분모 집계	**C-21 · C-27**
growth 이벤트	`fandomIdAtPayout` + `createdAt`	팬덤별 성장 원장 조회	**C-33**

정렬 키와 부등호 필드가 다르면 Firestore 가 질의를 거부합니다. 위 조합을 미리 세워 두는 이유이며, 화면에서 거르는 방식으로 우회하지 않습니다(§4.4).

✦ 7. Cloud Functions — 구현 상태

v0.3 §8은 **인터페이스 목록 36종**이었고 「구현은 이후 주차」였습니다. **빌드 v0.3.0 의 실물은 37종**이며(§7.4 대사표), 그중 32종을 v1.1 이 §7.1 에 적었습니다. 남은 외부 연동 대기 6종은 §7.2 입니다.

[2026-09-21 갱신] 이 문서의 「구현 완료 · 미배포」 표기는 2026-09-18 작성 시점 기준입니다. **2026-09-21 에 Blaze 전환과 37종 배포를 완료**했습니다(Callable 27 + Scheduler 9 · asia-northeast3, Auth 트리거 1 · us-central1). 배포 과정에서 환불 · 구독 웹훅의 권한 검사 누락(§18 R03 관련 보안 항목)을 서버 관리자 전용으로 정정해 재배포했고, FPI · Eclipse · 질문 컬렉션의 누락 인덱스 3종을 보완했습니다. 이 두 변경은 **v1.3** 에서 §5 · §6 · §7.4 표에 반영합니다.

v1.0 본문이 이 자리를 「17종」으로 적었습니다. §7.1 표의 실제 행 수는 32종이며 표가 맞습니다. 옮겨 적기 오류라 v1.1 에서 바로잡습니다.

[v1.2 대사 · R17] v1.1 이 「신설」로 적은 POP 소멸 3종은 **빌드 v0.3.0 에 없습니다** — 설계만 확정된 상태입니다. 반대로 v1.1 목록에 없던 **5종이 실물에 있습니다.** 두 방향을 §7.4 에 그대로 적었습니다.

v1.1 에서 신설한 함수 3종은 별첨 B 로 확정된 POP 소멸·알림 정책을 구현합니다. **셋 모두 빌드 v0.3.0 에 없으며(미구현) 구현 판본은 v0.4.0 입니다.**

함수	하는 일	원장
`fn-pop-expire`	일 1회. `lastActiveAt` 기준으로 lot 을 `active` → `locked` → `expired` 로 전이시키고 **같은 트랜잭션에서 소멸 거래를 원장에 append** 후 잔액 재계산(§2.5)	**B-05**
`fn-pop-expire-notify`	잠금 **30일 전 · 7일 전 · 잠금 발생 즉시 · 최종 소멸 전** 알림 발송. 재접속 시 잔여 알림 취소	**B-73**
`fn-pop-restore`	Grace 기간 내 재접속 시 `locked` lot 을 **전액** `active` **로 복구**. `expired` lot 은 되살리지 않음	**B-05**

세 함수 모두 별첨 1 ② 3자 대조 대상입니다. 소멸·복구가 잔액만 건드리고 원장을 비껴가면 대사가 즉시 잡습니다(§3.4).

7.1 구현 완료 — 32종(v1.1 기재분 · 실물 대사는 §7.4)

함수	하는 일	검증
`fn-pop-grant`	관리자 지급. lot·원장·지갑을 한 트랜잭션으로	●
`fn-pop-spend`	차감. 잔액 부족 E-POP-01 · lot 소진 · 기여 통계 반영	●
`fn-pop-refund`	환불. 반대 부호 새 원장 · 중복 환불 차단	●
`fn-pop-reconcile`	일 1회 3자 대조. 자동 정정하지 않음	●
`fn-admin-pop-grant-batch`	일괄 지급. 미리보기 + 대상 수 확인 2단계 · 500건 청크	●
`fn-quest-attend`	출석. 하루 1회	●
`fn-quest-verify-steps`	만보기. 목표 미달 E-QST-01 · 비정상 E-QST-02	●
`fn-ad-reward`	광고 보상. 일일 한도 E-AD-02 · 노출 ID 멱등	●
`fn-eclipse-defend`	방어. 샤드 10분할 · 종료 이클립스 E-ECL-01	●
`fn-eclipse-aggregate`	샤드 합산 · 침식률 갱신	●
`fn-fpi-aggregate`	7일 창 집계 · 스냅샷 · 가중치 검사	●
`fn-campaign-fund`	후원. 승인 전 E-CMP-03 · 목표 초과 E-CMP-01	●
`fn-auth-oncreate`	계정·지갑 문서의 유일한 생성 주체	●
`fn-planet-create`	온보딩 1회. 문서 ID 를 uid 로 고정해 중복 차단	●
`fn-consent-record`	약관 동의 이력. append-only · 철회도 새 행	●
`fn-admin-write`	모든 Admin 쓰기의 단일 관문 + 로그 적재	●
`fn-admin-set-role`	마스터 전용. 커스텀 클레임 + `admins` 갱신	●
`fn-iap-verify`	상품 표로 지급 수량 결정 · 스토어 거래 ID 멱등	●
`fn-iap-refund-webhook`	환불 회수. 잔액 부족 시 음수 대신 회수 보류	●
`fn-subscription-webhook`	구독 전이. 해지와 만료를 가른다	●
`fn-quest-verify-music-shake`	판정표 모순 검사 후 등급 재산출 · 등급별 배수	●
`fn-quest-verify-place`	GPS + 사진/QR · 반경 밖 E-QST-03 · 인증 재사용 차단	●
`fn-herodrop-trigger`	발동. 구매자에게 **기여 POP**, 팬덤 Drop Pool 에 **동일 수량 일반 POP** 발생(원장 **B-84**)	●
`fn-herodrop-claim`	수령. 문서 ID 로 중복 차단 · 총량은 트랜잭션에서 감소	●
`fn-campaign-settle`	마감 판정 · 실패 시 환불 기한 72시간 기입	●
`fn-campaign-refund-batch`	전액 환불. 부분 실패를 `partial` 로 남기고 Admin 큐로	●
`fn-campaign-admin-review`	승인·반려. 승인해야 모금이 열린다	●
`fn-question-create`	현상금 예치가 먼저. 잔액 부족 시 질문이 남지 않는다	●
`fn-answer-accept`	자기 답변 채택 금지 · 번복 불가	●
`fn-question-bounty-expire`	기한 만료 환불. 채택된 질문은 건드리지 않는다	●
`fn-referral-attribute`	자기 추천·중복 전환 차단	●
`fn-translation-settle`	24시간 후 확정. **자기 마킹은 세지 않는다**	●

결제 3종의 스토어 호출만 비어 있습니다. 자격증명이 스토어 계정에 묶여 있어 검증만 인터페이스 뒤로 뺐고, 지급·회수·구독 로직은 완성되어 테스트를 지납니다. **비워 둔 자리가 조용히 통과하지 않도록** 자격증명 없이 부르면 `E-PAY-02` 로 끊습니다.

7.2 남은 6종 — 외부 연동 또는 후속 구현이 남았습니다

함수	무엇이 더 필요한가	주차
`fn-campaign-ai-precheck`	AI 검수 모델·API 키	7주
`fn-briefing-generate`	AI 생성 모델·API 키	7주
`fn-share-poster-generate`	이미지 렌더 파이프라인 · Storage	9주
`fn-prereg-submit`	reCAPTCHA 키 · 사전예약 웹앱	9주
`fn-pop-grant` (사용자 생성 퀘스트 재원)	**보상 재원 확정(B-28 · B-30)** · 예치 원장 구현 — §9-3	7주
`fn-admin-*` 세부 화면 연동	Admin 화면	7주

나머지 32종은 지금 동작합니다. 운영값이 비어 있는 것을 임의값으로 채워 구현하지 않았습니다 — 그 값이 운영값처럼 굳습니다. 영웅의 선물이 그 예로, 총량·1인 한도·지속 시간이 `config` 에 없으면 **발동 자체를 거부**합니다.

결제 3종(8주)은 스토어 계정이 열려야 샌드박스 검증에 들어갑니다. 함수 자체는 계정 없이도 작성할 수 있으나, 영수증 없이는 검증이 불가능해 그 시점에 맞췄습니다.

7.2.1 v1.2 에서 신설 · 변경하는 함수 🟢

함수	하는 일	근거	구현 상태
`fn-link-preview` 🟢 **신설**	Quest 생성 시 URL 의 **원본 제목 · 썸네일 · 서비스명**을 서버가 조회해 읽기 전용으로 저장합니다. 유효 링크인데 조회만 실패하면 **기본 카드 + 제목 최소 입력**을 허용하고, 무효 링크는 저장을 거부하며 수정을 요청합니다	**C-05**	**미구현**
`fn-prereg-lookup` 🟢 **신설**	사전예약 기록을 **서버에서** 조회합니다. 제공자가 인증한 이메일을 **전체 정규화 일치**로 매칭하고, **서비스키는 서버에만** 둡니다. 동일 예약 건의 중복 보상을 문서 ID 로 막습니다	**C-42 · G-07**	**미구현**
브리핑 수신 함수	**인증된 수신 → `briefingCandidates` 적재 → 관리자 승인 → 앱 노출**의 3단계로 바꿉니다. **표본 수신 경로와 실제 Agent 연동 완료를 상태로 구분**합니다	**C-39**	**구현 중** — 빌드 v0.3.0 은 호출형 수신이며 초안을 `briefings` 에 `draft` 로 적재합니다. 수신 규격 · 인증 · 필수 필드 · 오류응답 · 샘플은 🔵 **S05** 로 제출합니다
Eclipse 판정 함수	**선차감을 없애고** 실패 확정 시에만 1회 차감하도록 바꿉니다. 목표는 **고유 참여자 수**이며 개인 인정은 유효 Shake 60회입니다. 판정 불가 회차는 `cancelled_error` 로 취소합니다	**C-22 · C-23 · C-25 ~ C-28**	**정정 필요** — §2.11 의 5건
차감 함수	**사유 코드별 허용 재화 타입을 검증**합니다. Governance 발의 등록비는 일반 POP 100 이며 기여 POP 으로 대체되지 않습니다	**G-05 · R03**	**정정 필요**
계정 연결 판정	이메일 **로컬 파트 비교를 제거**하고 전체 정규화 일치로 바꿉니다	**G-07 · R01**	**정정 필요**

신설 함수도 별첨 1 ② 서버검증 원칙을 따릅니다. 단말이 보낸 값을 그대로 믿지 않고, 멱등 키를 문서 ID 로 삼아 재시도가 두 번 반영되지 않게 합니다(**G-02**).

7.3 배포 리전

`asia-northeast3` **(서울)로 고정**합니다. 기본값 `us-central1` 이면 호출마다 태평양을 왕복합니다. 앱 쪽 클라이언트도 같은 리전을 지정합니다 — 어긋나면 호출이 `not-found` 로 떨어지는데, 원인이 함수 부재처럼 보여 찾기 어렵습니다.

7.4 함수 목록 대사 — R17 ● v1.2

빌드 v0.3.0 에 실제로 등록된 함수는 37종이며, 개발 프로젝트 배포는 Billing 전환(**C-36**) 뒤입니다.

영역	함수	건수
POP 원장	`fn-pop-grant` · `fn-pop-spend` · `fn-pop-refund` · `fn-pop-reconcile`	4
Quest	`fn-quest-attend` · `fn-quest-verify-steps` · `fn-quest-verify-music-shake` · `fn-quest-verify-place` · `fn-ad-reward`	5
계정	`fn-auth-oncreate` · `fn-planet-create` · `fn-consent-record` · `fn-admin-set-role`	4
팬덤 · FPI · Eclipse	`fn-eclipse-defend` · `fn-eclipse-aggregate` · **`fn-eclipse-trigger`** · `fn-fpi-aggregate` · **`fn-fpi-daily-snapshot`** · **`fn-ranking-country-aggregate`** · `fn-campaign-fund`	7
Admin	`fn-admin-write` · `fn-admin-pop-grant-batch`	2
결제	`fn-iap-verify` · `fn-iap-refund-webhook` · `fn-subscription-webhook`	3
Campaign	`fn-campaign-settle` · `fn-campaign-refund-batch` · `fn-campaign-admin-review`	3
Q&A	`fn-question-create` · `fn-answer-accept` · `fn-question-bounty-expire`	3
Hero Gift	`fn-herodrop-trigger` · `fn-herodrop-claim`	2
추천 · 번역	`fn-referral-attribute` · `fn-translation-settle`	2
브리핑	**`fn-briefing-ingest`** · **`fn-briefing-publish`**	2
합계		**37**

굵게 표시한 5종은 v1.1 §7.1 목록에 없던 함수입니다. 이클립스 발동 스케줄러 · Daily FPI Snapshot · 국가별 랭킹 사전 집계 · 브리핑 수신 · 브리핑 게시이며, v1.1 본문이 §2.6 과 §4.3 에서 설명한 동작의 실행 주체입니다. 목록에서 빠졌던 것이라 v1.2 에서 넣습니다.

v1.1 이 「신설」로 적었으나 빌드 v0.3.0 에 없는 함수 3종 — `fn-pop-expire` · `fn-pop-expire-notify` · `fn-pop-restore` **. 설계는 §2.5 에 확정되어 있고 구현은 v0.4.0** 입니다. **「신설」이 곧 구현 완료를 뜻하지 않는다는 것을 이 자리에 적습니다**(**G-08**).

✦ 8. 검증

8.1 실행 결과 — 2026-09-01

대상	건수	결과
보안 규칙 (에뮬레이터)	**118**	전부 통과
규칙 음성 통제	**2**	통과
Cloud Functions (에뮬레이터)	**64**	전부 통과
타입 검사 `tsc --noEmit`	—	**0건**

음성 통제 2건을 따로 두는 이유 — 「인증된 사용자의 지갑 쓰기가 거부되는가」와 「소유자의 지갑 읽기가 허용되는가」입니다. 규칙이 통째로 열리거나 통째로 닫히면 나머지 118건이 의미를 잃으므로, **테스트가 정상 동작하는지를 먼저 확인하는 장치**입니다.

빌드 v0.3.0 의 시험 사례 수(2026-09-14 기준) — 보안 규칙 **72건** · Cloud Functions **138건**.

> 위 8.1 표(2026-09-01)와 **세는 단위가 달라 두 수를 직접 비교하지 마십시오.** 그리고 **시험 사례 수는 검수 통과 건수가 아닙니다(G-08)**. 이 문서는 에뮬레이터에서 무엇을 확인하고 있는지만 적으며, 실서버 시나리오 통과 여부는 QA 체크리스트와 빌드노트가 담당합니다.

8.2 재현 방법

```
cd app/firebase
npm install
npm test          # 규칙 118건 + Functions 64건
```

에뮬레이터 실행에 **JDK 21 이상**이 필요합니다(firebase-tools 15 요구사항).

✦ 9. v1.0 이 남긴 ● 6건 — 처리 결과

전건 해소되었습니다. 이 문서에 ● 는 0건입니다.

v1.0 §9는 데이터 설계에 영향을 주는 미확정 6건을 적어 두었습니다. 별첨 B 가 그중 5건을 직접 확정했고, 1건은 초기값으로 메웠습니다.

#	v1.0 의 ● 항목	확정된 설계	원장
1	**sFPI 전환 기준**	**별도 환산비율 없음.** `sfpi == Σ dailyFpiSnapshots.value` — Daily FPI Snapshot 을 하루 1회 그대로 누적. `planets.sfpi` 의 입력이 확정되었고 §3.4 대조 항목에 넣었습니다	**B-54**
2	**FPI 동률 처리**	~~최근 활동 시각 우선~~ → **[v1.2 정정 · C-19] 1 · 1 · 3 공동 순위.** 표시 반올림만 같으면 실제 값 순이며, **최근 활동 시각으로 순위를 가르지 않습니다**(§4.4)	**C-19** · 원장 §15-②
3	**사용자 생성 퀘스트 보상 재원**	**예치 원장을 둡니다.** 기본 보상은 생성자 Reward Pool 사전 Deposit, **광고 증분은 FANUZU 재원**이라 Pool 에서 차감하지 않습니다 — 두 지급의 `sourceType` 이 달라 원장에서 구분됩니다	**B-28 · B-30**
4	**캠페인별 POP 사용 범위**	**Funding 후원은 기여 POP 만.** 등록비·수수료는 겸용(일반 우선). 정산 구조가 확정되어 후원 함수의 허용 타입을 좁혔습니다(§3.2)	**B-33 · B-57**
5	**이클립스 방어 창 · 1회 소모 POP**	**공동 Shake 방어 이벤트. 1회당 POP 소모를 두지 않습니다** — 방어는 Shake 참여로 이루어집니다. **[v1.2 정정 · C-24]** 「하루 2회」는 **고정 제한이 아니라 기본값**이며 팬덤별로 횟수 · 시각을 설정합니다(§2.11)	**B-55 · C-24**
6	**단결형 ↔ 영웅의 선물 관계**	**같은 기능이므로 컬렉션 하나입니다.** Hero Gift 상품 Catalog 로 일원화하고 단결형 아이템 전용 컬렉션을 두지 않습니다	**B-40 · B-84 · B-85**

4번이 가장 이르다고 말씀드렸던 항목입니다. v1.0 시점에는 후원 함수가 두 재화를 모두 받도록 열려 있었고 「원장이 쌓이기 전에 정해 주시는 편이 좋다」고 적었습니다. **별첨 B 가 기여 POP 만으로 확정**해 주셔서 원장이 본격적으로 쌓이기 전에 좁혔습니다.

저희가 초기값·해석으로 메운 항목은 정책 원장 §15 「발주사 확인 요청 목록」에 근거와 확인 문항을 붙여 모았습니다. 이 문서와 관련된 것은 **§15-②(FPI 동률)** 와 **§15-①(Passport Tier ↔ User Level 매핑 · §2.1)** 두 건입니다.

9.1 v1.1 → v1.2 변경 값 ● v1.2 신설

값이 실제로 달라진 것만 모았습니다. 신설 필드 · 신설 컬렉션은 §2 에, 정정 대상 구현은 각 절의 「구현 상태」에 있습니다.

#	대상	v1.1 의 값	v1.2 의 확정값	근거
1	User Level 산출 입력	누적 기여 POP 5등급(3,000 / 20,000 / 100,000 / 500,000)	**User EXP 기반 Lv.1~100+ 연속** · GLOW~NOVA 는 레벨 구간 칭호	**C-18 · R04**
2	FPI 동률	최근 활동 시각이 빠른 쪽 상위	**1 · 1 · 3 공동 순위** · 동점 내부는 결정적 정렬키	**C-19**
3	FPI 축 값의 원천	(명시 없음 — 구현은 지급 POP)	**실제 인정 활동.** 보상액 · 광고 지급량과 무관	**C-34 · R07**
4	출석 · 현장 방문의 축	(근거 없음 — 미연결)	**COMMUNITY**	**C-35**
5	Eclipse 차감 시점	(선차감 후 성공 시 복구)	**성공 시 감소 없음 · 실패 확정 시에만 1회**	**C-22 · R09**
6	Eclipse 사전 알림	300초 전	**3분 전**	**C-22 · R09**
7	Eclipse 기본 발생 시각	13:00 · 21:00 KST	**07:50 · 17:50 KST**(푸시 07:47 · 17:47)	**C-24**
8	Eclipse 방어 목표	팬덤 공동 누적 100,000 Shake	**고유 참여자 수 ≥** `max(1, ceil(7일 평균 DAU × 5%))` · 개인 인정 60회	**C-25 · C-26**
9	Eclipse 회차 수	하루 정확히 2회	**기본 2회이며 고정 제한 아님** · 팬덤별 설정	**C-24**
10	Eclipse 방어 창	30분	**기본 60초** · ● 운영값	**C-22**
11	개인 Quest 기본 기간	(정의 없음)	**공개 후 24시간** · 종료일 선택 시 생성자 현지시간 23:59:59 · 최대 30일 · 최소 1시간	**C-06**
12	Location Quest 반복	(정의 없음)	**기본 1회** · 반복 허용 시 직전 유효 인증부터 **24시간** 간격	**C-12**
13	사용자 생성 Quest 광고 보상	「광고 증분은 FANUZU 재원」	**초기 10 기여 POP / 회** · 회사 부담 고정형 · Pool 무차감	**C-02 · C-31**
14	Reward Pool 미사용분 반환	(익일 반환 · Figma 기준)	**정상 종료 시 즉시 반환** · 예약분은 검토 결과에 따름	**C-09**
15	검토 · 지급	(단일 상태)	**제출 · 검토 · 승인 · 지급 4단계** + 지급 상태 별개 · **자동 승인 없음** · 24시간 검토 목표	**C-08 · G-03**
16	Growth POP 원천	Zap · 기여 이벤트	**보상 획득 + 이후 기여 두 단계.** 예치 · 반환 · 수수료는 제외	**C-29 · C-30**
17	잔액 필드	타입별 1필드	**총 보유 · 예약 · 사용 가능 3필드**	**G-05**
18	복합 인덱스	51건	**63건**(실물 대사) · 필드 예외 6건	**R17**
19	Cloud Functions	「32종 + 신설 3종」	**실물 37종** · POP 소멸 3종은 미구현	**R17**

🔵 **미확정으로 남는 것**(원장 §19) — User EXP 요구량 · FPI 정밀도와 정렬키 · 환산계수 · 유효 활동 정의와 집계 시간대(**S01**), 사용자 생성 허용 활동 초기값과 위치 무관 QR / NFC 검증 명세(**S03**), 상품별 모금 기간 제안표(**S03**), 브리핑 수신 규격과 권한 매트릭스(**S05**). **확인 전 유지 기준은 각 절에 적었으며, 임의로 단순화하지 않았습니다.**

✦ 10. 이 문서의 효력

항목	내용
판본	**최종본 v1.2** — 2026-09-18
대체	**Firestore 설계서 최종본 v1.1**(2026-09-11)을 **대체**합니다. 두 문서가 다르게 적은 곳은 이 문서를 따릅니다
기준	**정책값은 정책 원장 v1.1 이 단일 기준**입니다. 이 문서는 **데이터가 어떻게 저장되는지**를 정하고 **값이 얼마인지는 원장이 정합니다.** 어긋나면 원장이 맞습니다
근거	**2026-09-16 발주사 회신본**(C-01 ~ C-43 · G-01 ~ G-08 · R01 ~ R22 · S01 ~ S06) · 별첨 B v1.4(2026-09-07) · 정책 설계서 최종본 v1.1 · 상세정의서 v1.6.1 · 별첨 2 2주차 · 견적서 v2.1 7p
구현 상태	**빌드 v0.3.0 · 2026-09-14 기준**입니다(**G-06**). 「구현 완료」는 에뮬레이터 시험을 지난다는 뜻이며 **실서버 검수 통과와 구분**합니다(**G-08**)
범위	**계약 범위를 새로 정하지 않습니다.** POP 원장의 네 가지 불변식과 3자 대조 구조는 그대로이며 별첨 1 ② 판정 대상 설계가 유지됩니다
일정	**계약 일정은 재기산되지 않습니다.** 12주 안에서 정상화합니다

v1.1 의 스키마 변경은 마이그레이션이 필요합니다. `tier` → `passportTier` · `userLevel` 분리, `originNo` → `firstLightBadge` · `fandomJoinNo` 분리, `popLots` 의 잠금·소멸 필드 신설, `nickname` 유니크 인덱스 제거 네 건입니다. **원장 데이터는 건드리지 않고 파생·인덱스 계층만 바꾸므로 3자 대조 정합성이 유지됩니다.**

v1.2 가 더하는 마이그레이션은 다섯 건이며 §3.7 에 대조 기준과 함께 적었습니다 — 잔액 3필드 분리 · 등급 산출 입력 교체 · 대표 팬덤 필드 개명 · 행성 문서 ID 교체 · 이클립스 컬렉션 개명입니다. **여기에서도 원장 행은 손대지 않습니다.**

이 판본은 계약 범위 · 비용 · 일정을 바꾸지 않습니다. 2026-09-16 회신의 확정 기준을 데이터 설계에 반영한 것이며, 12주 일정 안에서 정상화합니다.

서면 승인을 요청드립니다. 승인해 주시면 이 판본이 5주차 이후 구현의 데이터 기준이 됩니다. 다르게 보시는 항목이 있으면 그 항목만 알려 주십시오.