sweech Firestore 설계서 — 최종본 v1.0

프로젝트: **FANUZU** iOS/Android 1차 정식 출시 버전 개발 근거: 별첨 2 2주차「Firestore 설계서」· 견적서 v2.1 7p「산출물 목록 · 설계」작성: 스위치 주식회사(Sweech) · 2026-09-01 대체 관계: `2주차_Firestore_설계서.pdf` **(v0.3 · 2026-08-13)를 대체**합니다 함께 제출하는 실물: `firestore.rules` · `firestore.indexes.json` · **Cloud Functions 소스 32종**

✦ 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 바뀐 값 — 먼저 밝힙니다

#	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**	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	정책 설계서 §P-04 (앱 Figma「TIER 정보」)
5	`planetStages` 7단계 구간값 🔴	**Lv.1~100 · 7 Stage · 누적 POP 50,000,000 · sFPI 7,500,000**	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**	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

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 정정선도 그대로입니다. 활동 POP 자동소멸과 FIFO 철회는 유효하며, 활성 스키마에 만료 시각·휴면 잠금 필드를 두지 않았습니다. N-10/Q32 제안은 여전히 **비활성·미배포**입니다.

✦ 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 설계서와 같습니다 — 🟢 확정 · 🟡 운영값 · 🔴 미확정.

✦ 2. 데이터 모델

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

2.1 `/users/{uid}`

필드	v0.3	v1.0
`authProviders`	`array<string>` · 이메일 병행 CD-18 🔴	`array<string>` — `email` · `google.com` · `apple.com` · `kakao` · `line` 5종 🟢
`locale`	지원 목록 D-14 🔴	`ko` · `en` · `ja` · `zh` · `es` 🟢
`tier`	(없음)	`GLOW` | `SPARK` | `BEAM` | `STARLIGHT` | `NOVA` · 서버 산출 🟢 **신설**
`originNo`	(없음)	int · **ORIGIN 100 창립 멤버 순번 001~100** · 없으면 `null` 🟢 **신설**

`tier` 를 사용자 문서에 두는 이유는 조회 때문입니다. 누적 기여 POP 에서 매번 계산하면 랭킹·프로필·라운지에서 같은 계산이 반복됩니다. **원본은** `totalContributedPop` **이고** `tier` **는 파생값**이므로, 원장이 움직일 때 함께 갱신하고 대사 대상에 넣습니다.

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) 🟢

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	1~7. 태동(성운) · 시작 · 성장 · 함께 · 동행 · 빛 · 영원
`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 POP 과 sFPI 를 동시에 충족해야 일어납니다(정책 §P-08). 하나만 채우면 현재 Level 에서 대기하며, 초과분은 다음 Level 진행도로 이월합니다.

`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.

✦ 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 둘을 건드릴 수 없음	원장 테스트 — 잔액 분리 확인

3.2 차감 순서 — 규범이 없다는 것을 원장에 남깁니다

W-03으로 FIFO 가 철회되어 **차감 순서에 규범이 없습니다.** 구현은 `grantedAt` 오름차순으로 훑는데, 이것은 정책이 아니라 **안정적인 순회를 위한 구현 선택**입니다.

그래서 차감 원장에 `lotOrder: 'grantedAt-asc-nonnormative'` 를 남깁니다. 나중에 순서가 서면으로 확정되면, **이 표시가 있는 과거 거래와 없는 거래를 구분**할 수 있습니다.

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 정정이 §3.6의 순서입니다.

3.4 3자 대조 — 구현

`fn-pop-reconcile` 이 일 1회 전 사용자를 훑고 여섯 가지를 봅니다.

검증	식
활동 POP 잔액 ↔ lot	`activityBalance == Σ popLots[activity, active].remaining`
활동 POP 잔액 ↔ 원장	`activityBalance == Σ popTransactions[activity].amount`
지원 POP 잔액 ↔ lot	〃
지원 POP 잔액 ↔ 원장	〃
통계 ↔ 원장	`totalContributedPop == Σ 기여 차감 − Σ 환불`
등급 ↔ 통계	`tier == tierOf(totalContributedPop)` — 파생값이 원본과 어긋나지 않는지

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. 등급도 같은 방식으로 위조해 잡히는지 확인합니다.

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.**

✦ 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.

4.2 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.

4.3 갱신 주기

정책 §P-03이「30초~1분」으로 정했고, Cloud Scheduler 의 최소 주기가 1분이라 **1분 스케줄**로 둡니다. 화면에는 마지막 갱신 시각을 띄웁니다.

✦ 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.

✦ 6. 인덱스

`firestore.indexes.json` — 복합 인덱스 51건 · 필드 예외 6건. v0.3에서 바뀌지 않았습니다.

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

✦ 7. Cloud Functions — 구현 상태

v0.3 §8은 **인터페이스 목록 36종**이었고 「구현은 이후 주차」였습니다. 현재 **17종이 동작하며 에뮬레이터 테스트를 지납니다.**

7.1 구현 완료 — 32종

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. 지원 POP 차감	●
`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 — 전부 외부 연동이 필요합니다

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 미확정** — §9-3	발주사 회신 후
`fn-admin-*` 세부 화면 연동	Admin 화면	7주

나머지 32종은 지금 동작합니다. 미확정 정책에 걸린 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.

7.3 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.

✦ 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.

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. 남은 ●

정책 설계서 최종본이 정리한 12건 중 **데이터 설계에 영향을 주는 것**만 옮깁니다.

#	항목	왜 데이터 설계에 걸리는가	결정 주체
1	**sFPI 전환 기준**	FPI → sFPI 전환 규칙이 없으면 `planets.sfpi` 를 채울 입력이 정해지지 않습니다	발주사 · 09-10 대조 후
2	**FPI 동률 처리**	랭킹 쿼리의 2차 정렬 키가 정해져야 인덱스가 확정됩니다. 저희 제안은 최근 활동 시각	발주사
3	**사용자 생성 퀘스트 보상 재원**	예치 원장을 둘지가 갈립니다. 예치가 필요하면 컬렉션이 하나 늘어납니다	발주사
4	**캠페인별 POP 사용 범위**	활동 POP 만인지 지원 POP 도 되는지에 따라 **정산 구조가 갈립니다.** 원장이 분리돼 있어 사후 변경 비용이 큽니다	발주사
5	**이클립스 방어 창 · 1회 소모 POP**	`config` 초기값입니다. 스키마는 서 있고 값만 비어 있습니다	발주사 · Admin 운영값
6	**단결형 ↔ 영웅의 선물 관계**	같은 기능이면 컬렉션 하나, 다르면 둘입니다	09-10 저희 안 제시

4번이 가장 이릅니다. 캠페인 후원 함수가 이미 동작하는데 현재는 두 재화를 모두 받도록 열어 두었습니다. 지원 POP 을 막아야 한다면 **원장이 쌓이기 전에** 정해 주시는 편이 좋습니다.

✦ 10. 이 문서의 효력

v0.3을 대체합니다. v0.3과 이 문서가 다르게 적은 곳은 이 문서를 따릅니다.

정책 수치가 이 문서와 어긋나면 정책 설계서 최종본을 따릅니다. 이 문서는 데이터가 어떻게 저장되는지를 정하고, 값이 얼마인지는 정책 설계서가 정합니다.

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