캐슬 브레이커 — 세부기획서 (UIUX와 아트)

> 재구성 기획서 (역설계) / 작성 기준 시점: 2026-08-07 (프로젝트 시작 시점) / 문서 버전: Ver 0.1 프로젝트 ID: P016 / 근거 수준: B / 작성: Claude (재구성) 이 문서는 결과물과 제작 기록으로부터 기획 의도를 역추정해 당시 작성한 것처럼 재구성한 것이다. 확인된 사실과 추정을 [근거]/[추정] 태그로 구분한다.

1. 목적과 범위

- 목적: 화면별 명세, 조립 연출, 전투 HUD/결과, 튜토리얼, 아트(플랫 기하 SVG)와 리소스 파이프라인을 정한다.
- 범위: 메인, 인증, 격납고(설계/편성/퍽/랭킹), 매칭, 전투, 결과, 튜토리얼 6페이지, 팔레트, 스프라이트 합성. 사운드는 기록 없음.
- 기획서 §12는 격납고 탭을 설계, 편성, 랭킹 3개로 두었으나, 실제 README는 설계, 편성, 퍽, 랭킹 4개다. [근거: 오펜스게임_기획서.md §12.4, README.md]

2. 설계

화면 흐름

- 의도(Why): 조립 게임임을 첫 화면부터 알리고 격납고를 홈으로 삼아 전투 후 항상 복귀하게 한다.
- 설계(How): `MAIN`(로고 조립 연출 600ms, GAME START만 배치) → `IDENTIFY`(모달, "조종사 등록", 이름 2~12자 실시간 검증) → `HANGAR` → `MATCHING`(덱 6종 원형 회전, "상대를 찾는 중…", 경과 시간, 성립 시 상대 카드 1.5초 후 전환) → `BATTLE` → 결과 오버레이 → `HANGAR`. 메인에는 GAME START 외 버튼을 두지 않는다. [근거: 오펜스게임_기획서.md §12.1~§12.3, §12.7]
- 필요(What): `Main.tsx`, `Identify.tsx`, `Hangar/Hangar.tsx`, `Matching.tsx`, `Battle.tsx`, `Result.tsx`.
- 작업 방법: 화면 골격 → 순환 연결 → 연출.
- 완료 기준: 순환이 끊김 없이 돈다. 격납고 제목이 {입력한 이름}의 격납고.
- 리스크/대안: 신규 계정은 대표 빌드가 미리 채워져 있어 바로 출격 가능(진입 마찰 최소화). [근거: README.md]
- 근거: [근거: 오펜스게임_기획서.md §1.4, README.md]

격납고 헤더와 [출격]

- 의도(Why): 레이팅/전적과 출격 가능 여부를 항상 보이게 한다.
- 설계(How): 좌 {이름}의 격납고, 중 레이팅/전적, 우 [?] 튜토리얼 재열람과 [출격]. 출격 덱이 6개를 정확히 채워야 활성화되고 미충족 시 "덱에 프리셋 N개가 더 필요합니다" 툴팁. 한줄소개는 "+ 내 소개"로 편집한다(60자). [근거: 오펜스게임_기획서.md §12.4, README.md, constants.ts]
- 필요(What): `BioEditor.tsx`, 덱 편성 UI.
- 작업 방법: 덱 상태별 버튼 활성화 테스트.
- 완료 기준: 덱 5개일 때 비활성, 6개일 때 활성.

- 리스크/대안: 퍽 탭은 기획서에 없던 추가 탭이다(`PerkPicker.tsx`).
- 근거: [근거: 오펜스게임_기획서.md §12.4, README.md]

설계 탭(조립) 레이아웃과 실시간 반영

- 의도(Why): 스텟과 가격이 같은 조립에서 동시에 나온다는 핵심을 눈으로 확인하게 한다.
- 설계(How): 3열(좌 파츠 목록 카테고리 탭 / 중 슬롯 구조도 도면 + 실제 전투 스프라이트 / 우 스텟 패널). 장착/해제 즉시 총가격과 전 스텟이 갱신되고 "적용" 단계가 없다. 호버하면 반투명 고스트(투명도 45%)와 델타 뱃지(초록 +, 빨강 −)가 뜬다. 스텟 패널 하단에 이동속도, 회피율, DPS, 주기, 횡단시간, 비행, 대공, 공대지 가능 여부를 항상 표시하고, 대공 ✗→✓ 전환 시 해당 행이 0.4초 점멸한다. 사거리 50/0 문턱은 플레이어가 스스로 발견하기 어렵기 때문이다. [근거: 오펜스게임_기획서.md §12.6, §12.6.1]
- 필요(What): `Assembly.tsx`, `assembly-rules.ts`, `art/parts-svg.ts`.
- 작업 방법: 규칙(`assembly-rules.ts`) → UI → 델타/고스트.
- 완료 기준: 변경 시 가격과 전 스텟이 같은 프레임에 갱신된다.
- 리스크/대안: 유효성 위반 시 도면 하단 사유 배너와 저장 버튼 비활성.
- 근거: [근거: 오펜스게임_기획서.md §12.6]

조립 연출 AN-1~AN-11

- 의도(Why): "목록에서 고른다"가 아니라 "기체에 붙인다"는 감각을 준다.
- 설계(How): AN-1 장착 비행 240ms(`cubic-bezier(0.2,0.9,0.3,1)`), AN-2 링 200ms, AN-3 셰이크 100ms(±2px 3회), AN-4 플래시 180ms, AN-5 스텟 막대 300ms(증가 초록/감소 빨강), AN-6 가격 롤업 300ms, AN-7 해제 180ms(−15° 회전), AN-8 장착 불가 흔들림 300ms+토스트 2초, AN-9 무기 드래그 시 유효 마운트 맥동 링, AN-10 프리셋 로드 시 40ms 간격 순차 조립, AN-11 저장 성공 400ms. 연출은 CSS `transform`/`opacity`와 `requestAnimationFrame`만 쓰고 리플로우 속성은 쓰지 않는다. 연출용 난수는 결정론 RNG와 분리한다. 빠른 연속 입력은 진행 중 연출을 즉시 최종 상태로 스냅한다. `prefers-reduced-motion`에서는 이동/셰이크를 생략한다. [근거: 오펜스게임_기획서.md §12.6.2, motion.ts]
- 필요(What): `art/motion.ts`.
- 작업 방법: 타이밍 상수 확정 → 연출 컴포넌트 → 접근성 분기.
- 완료 기준: 11개 연출이 모두 재생되고 reduced-motion에서 상태만 즉시 바뀐다.
- 리스크/대안: 구현에서 AN-9 무기 드래그가 실제 클릭 방식(README "파츠 카드 클릭 = 장착")으로 단순화되었을 가능성이 있다. [추정: README 조작 표에 드래그 서술이 없음]
- 근거: [근거: 오펜스게임_기획서.md §12.6.2, README.md "조작"]

전투 HUD와 소환 조작

- 의도(Why): 시간이 정지된 선택 구간에서 예산을 빠르게 읽고 배분하게 한다.
- 설계(How): 상단 중앙 라운드, 상태, 타이머, 상단 좌우 이름, 레이팅, 성 HP, 좌상단 내 잔액/예약 합계/잔액−예약, 우상단 상대 잔액과 준비 완료 램프. 하단 덱 6슬롯 소환 바(클릭 +1, 우클릭 −1, 키보드 1~6 +1, Shift −1, `Enter` 준비 완료)와 배속 1×/2×/4×(표시 배속의 2배로 실제 진행). 유닛 호버 시 상대 유닛을 포함한 전체 스텟 툴팁(덱이 공개 상태이므로). HP 바는 피격 후 3초만 표시. REVEAL 2초에 양측 소환 목록을

카드로 공개한다. [근거: 오펜스게임_기획서.md §12.8, README.md "조작", art/렌더 상수 HP_BAR_SHOW_SEC]

- 필요(What): `Battle.tsx`, `battle/renderer.ts`, `loop.ts`, `view.ts`, `effects.ts`.
- 작업 방법: Canvas 단독 전투 화면, 전투 상태는 스토어에 두지 않는다.
- 완료 기준: 예약 합계가 잔액을 넘으면 준비 완료가 비활성이다.
- 리스크/대안: 기획서에는 배속이 없었으나 구현에 추가되었다(원인 기록 없음). 온라인 전투는 `OnlineBattle`로 분리한다는 README 서술이 있으나 파일명은 확인하지 못했다.
- 근거: [근거: 오펜스게임_기획서.md §12.8, README.md]

결과 화면

- 의도(Why): 한 판의 손익을 되짚어 다음 설계를 개선하게 한다.
- 설계(How): 승리/패배/무승부 + 종료 사유(성 파괴/시간 종료, 항복, 연결 끊김), 레이팅 롤업(예: 1,024 → 1,041), 양측 성 HP 막대, R1~R7 지출 그래프, 프리셋별 소환, 처치, 입힌/받은 데미지. [격납고로] 버튼. 통계는 `MatchStats`가 틱 중 누적한다. [근거: 오펜스게임_기획서.md §12.11, types.ts]
- 필요(What): `Result.tsx`.
- 작업 방법: 통계 구조체 → 표시.
- 완료 기준: 7라운드 지출이 모두 표시된다.
- 리스크/대안: 자폭사는 처치 수에서 제외되므로 통계 해석 주의.
- 근거: [근거: types.ts, 오펜스게임_기획서.md §9.5]

튜토리얼 6페이지

- 의도(Why): 발견하기 어려운 규칙(특히 사거리 50 문턱)을 반드시 가르친다.
- 설계(How): 격납고 위 모달, 상단 2/3 일러스트/하단 1/3 설명, 인디케이터와 [이전][다음][시작하기]. 페이지: ① 목표 ② 전장(지상/공중 두 길, 공중은 공중만 막는다) ③ 라운드(30→120초) ④ 자금(시간이 흐를 때만) ⑤ 조립 ⑥ 스테이터스(사거리 50/0 지점을 붉게 표시). `tutorialSeen`이 false일 때 자동 표시, [?]로 재열람. 일러스트 6종은 전부 인라인 SVG(`tutorial-art.ts`). [근거: 오펜스게임_기획서.md §12.10, README.md]
- 필요(What): `Hangar/Tutorial.tsx`, `art/tutorial-art.ts`.
- 작업 방법: 6페이지 텍스트 → SVG → 표시 조건.
- 완료 기준: 신규 이름 입력 시 자동 표시, 기존 이름은 미표시.
- 리스크/대안: 기획서의 ② 전장 설명 "앞 유닛이 뒤 유닛을 막는다"는 겹침 허용(§15-9) 이후 사실과 달라졌다. 튜토리얼 문구 갱신 여부는 미확인. [근거: 오펜스게임_기획서.md §12.10 vs §15-9]
- 근거: [근거: 오펜스게임_기획서.md §12.10]

플랫 기하 아트와 판독성

- 의도(Why): 에셋 제작 없이 수백 조합을 처리하고 밀집 유닛을 구분한다.
- 설계(How): 아웃라인, 그라디언트, 텍스처, 그림자 없음, 단색 면, 사각, 사다리꼴, 삼각, 원, 반원, 직각, 45°, 30°. 판독성 장치 4가지: ① 계단 배치(같은 진영/레인에서 12u 미만 겹침 유닛에 행 0~3을 돌려 배정, 행마다 위로 7px/뒤로 3px, 표시 전용으로 시뮬 좌표는 불변) ② 네거티브 스페이스(할당 24px, 그리기

20px, 좌우 2px 틈) ③ 카테고리 명도 위계(다리 → 몸통 → 팔 → 무기 → 특수 순으로 밝게) ④ 진영 보색(P1 청록 / P2 주황). 다리 카테고리별 보조 모션(바퀴/궤도 회전, 이족/사족 ±1px 바운스, 제트 후방 삼각 트레일). 전투 연출은 단색 도형 1개 128ms 발사, 피격 흰 플래시 80ms, 회피 12px 점프+MISS 300ms, 사망 3~5도형 분해 250ms, 스플래시 원 200ms. [근거: 오펜스게임_기획서.md § 12.9, palette.ts 규약]

- 필요(What): `art/palette.ts`, `parts-svg.ts`, `prerender.ts`, `battle/effects.ts`.
- 작업 방법: 파츠 도형 → 앵커 합성 → 30기 밀착 스크린샷.
- 완료 기준: 30기 밀착 상태에서 개별 유닛이 육안 구분된다.
- 리스크/대안: 계단 배치가 판독성의 1차 방어선이며 네거티브 스페이스만으로는 같은 자리 유닛을 구분하지 못한다.
- 근거: [근거: 오펜스게임_기획서.md § 12.9.2, § 15-8]

SVG 에셋 파이프라인

- 의도(Why): 외부 파일 0개, 매치 시작 시 필요한 것만 생성해 메모리와 제작 비용을 줄인다.
- 설계(How): 파츠 SVG(path+앵커+viewBox) → 앱 기동 시 파츠 16종×진영 2색 = 32장 프리렌더(파츠 18종 중 구현 수는 기획서 § 12.9.4 표기 기준, 실제 장수는 미확인) → 매치 시작 시 덱 6종×진영 2 = 12장만 앵커 합성 캐시 → 렌더 루프는 `drawImage`만 호출. 레이어 순서는 다리→몸통→팔→무기→특수. P2는 x축 미러. 가로 몸통의 `ARM0`은 차체 위(0, −6.5)라 120mm를 얹으면 전차 실루엣, 세로 몸통은 몸통 옆(4,−7 / 4,−1)이라 인간형. [근거: 오펜스게임_기획서.md § 12.9.4, § 12.9.5]
- 필요(What): `prerender.ts`, `parts-svg.ts`.
- 작업 방법: 앵커 규격 확정 → 합성 → 캐시.
- 완료 기준: 빌드 산출물에 `.png`/`.svg`/`.woff` 없음.
- 리스크/대안: 파츠가 18종인데 문서가 16종이라 적어 불일치가 있다(몸통 2+팔 3+무기 4+다리 5+특수 4=18). 기획서 오기로 추정. [추정: 파츠 표 합계 18]
- 근거: [근거: 오펜스게임_기획서.md § 12.9.4, § 7]

사운드

- 의도(Why): 해당 없음(사유: 기획서/코드에 사운드 항목이 없다).
- 설계(How): 기록 없음.
- 필요(What): 기록 없음.
- 작업 방법: 기록 없음.
- 완료 기준: 기록 없음.
- 리스크/대안: 전투 타격/조립 결합음이 없으면 조립 연출 효과가 약해질 수 있다. [추정: 연출 중심 설계]
- 근거: [추정: 사운드 관련 파일/절 부재]

3. 데이터/리소스 명세

3.1 팔레트 (`palette.ts`)

용도	값
앱 배경 / 패널 / 패널 호버	#141C26 / #1E2936 / #27364A

용도	값
전장 하늘 / 지면 / 지면선	`#1B2430` / `#2A3644` / `#3C4C5E`
텍스트 1차 / 2차 / 비활성	`#E6EDF3` / `#8B9AAB` / `#54636F`
P1(다리→특수)	`#1C6E77` `#2A94A0` `#38BEC9` `#6FD3DB` `#A7E8ED`
P2(다리→특수)	`#8C4418` `#C4632A` `#F0803C` `#F5A268` `#FAC79E`
HP 높음 / 중간 / 낮음	`#4ADE80` / `#FACC15` / `#EF4444`
자금 / 위험 / 성공 / 플래시	`#FACC15` / `#EF4444` / `#4ADE80` / `#FFFFFF`
[근거: 오펜스게임_기획서.md §17 palette.ts]	

3.2 렌더 상수

항목	값
캔버스	1280×720, 1u = 2px, 좌우 여백 40px
지상 기준선 / 공중 기준선	y = 560 / y = 300
성 폭	80px
스프라이트 할당 / 그림 / 틈	24 / 20 / 2px
HP 바 표시	피격 후 3초
[근거: 오펜스게임_기획서.md §17]	

3.3 리소스 목록

항목	형태	수량
파츠 SVG	코드 문자열	파츠 수만큼(기획서 16종 표기, 파츠 표 18종)
로고	`art/logo.ts` SVG	1
튜토리얼 일러스트	`art/tutorial-art.ts` SVG	6
이미지/폰트 파일	없음	0
사운드	기록 없음	—
[근거: 오펜스게임_기획서.md §12.9.4, §12.10]		

4. 작업 순서

1. 팔레트/렌더 상수 → 2. 파츠 SVG/앵커 → 3. 프리렌더/합성 캐시 → 4. 전투 렌더/이펙트 → 5. 조립 화면과 연출 → 6. 격납고 셸, 인증, 매칭 → 7. 튜토리얼 → 8. 결과 화면.

5. 검수 기준

- 외부 이미지/폰트 파일 0개.
- 30기 밀착 스크린샷에서 유닛이 구분된다.
- 장착/해제 시 가격/스텟이 같은 프레임에 갱신, 대공 ✗→✓ 점멸.

- `prefers-reduced-motion`에서 이동 연출 생략.
- 튜토리얼 6페이지 SVG 표시, 신규 계정 자동 표시.
- 튜토리얼 문구가 겹침 허용 규칙과 일치한다(현재 미확인).

캐슬 브레이커 — 세부기획서 (시뮬레이션과 데이터)

> 재구성 기획서 (역설계) / 작성 기준 시점: 2026-08-07 (프로젝트 시작 시점) / 문서 버전: Ver 0.1 프로젝트 ID: P016 / 근거 수준: B / 작성: Claude (재구성) 이 문서는 결과물과 제작 기록으로부터 기획 의도를 역추정해 당시 작성한 것처럼 재구성한 것이다. 확인된 사실과 추정을 [근거]/[추정] 태그로 구분한다.

1. 목적과 범위

- 목적: 파츠, 상수, 대표 빌드, AI, 퍽의 수치를 확정하고, 그 수치가 상성 순환을 만들도록 검증하는 방법을 정한다.
- 범위: 조립 데이터, 상수, 대표 빌드, 상성 설계, 밸런스 목표와 실측, AI 5인, 퍽 10종. 네트워크/화면은 다른 03 문서에서 다룬다.
- 수치는 결과물(`constants.ts`, `parts.ts` 표, 기획서 §17)에서 읽은 값만 확정 표기한다.

2. 설계

파츠 데이터 설계

- 의도(Why): 열등 파츠 없이 18종 모두가 최소 하나의 1순위 빌드를 갖게 해 설계 선택을 의미 있게 한다.
- 설계(How): 몸통 2/팔 3/무기 4/다리 5/특수 4. 보정은 유닛 전체 스텟에 합산한다. 가격은 "현실적 제작 난이도" 순으로 매겼다(다관절 로봇 팔 35 > 고정 무장창 15, 이족 55 > 사족 45 > 캐터필러 30 > 바퀴 20, 제트 90 최고가). 열등 파츠였던 로봇 팔, 이족, 사족은 보정 추가로 포지션을 만들었다. 날개는 고도만 부여하고 기동성은 주지 않아 정석 항공기는 날개+제트(160원)가 된다. [근거: 오펜스게임_기획서.md §7, §15-2, §15-4, §15-5]
- 필요(What): `parts.ts`(18종), 죽은 파츠 검사표.
- 작업 방법: 표 확정 → `assemble.ts` → 대표 빌드로 모든 파츠 커버 확인.
- 완료 기준: 파츠 18종 전부가 대표 빌드에 최소 1회 등장한다.
- 리스크/대안: 신규 파츠 추가 시 검사 필수.
- 근거: [근거: 오펜스게임_기획서.md §16.3, parts.ts]

스텟 파생 설계

- 의도(Why): 스텟을 6종으로 유지하면서 속도와 회피를 하나의 기동성으로 묶어 제트의 높은 가격에 설득력을 준다.
- 설계(How): 기본값에 파츠 보정 합산 후 하한 적용. 이동속도 = 기동×20 밀리유닛/틱, 회피 = min(기동×25, 4000)bp, 공격 틱 = 주기×30(주기 보정은 모두 0.1초 배수라 항상 정수). DPS는 표시용. 제트(기동

80)는 12.5초에 전장 횡단, 회피 20%. [근거: 오펜스게임_기획서.md §8.2, constants.ts]
- 필요(What): 상수 `SPEED_PER_MOBILITY_MILLI`, `EVASION_BP_PER_MOBILITY`, `EVASION_BP_CAP`.
- 작업 방법: 정수만 사용하고 부동소수는 렌더 직전 표시에서만 쓴다.
- 완료 기준: `cycleTicks`, `speedMilli`, `evasionBp`가 모두 정수다.
- 리스크/대안: 퍽의 비율 보정(`PERK_ARMOR_PCT=0.12` 등)은 소수 상수라 반올림 규칙이 결정론에 영향을 줄 수 있다. [근거: constants.ts, perks.ts `Math.round`] [추정: 반올림이 양측 동일 코드이므로 결정론은 유지됨]
- 근거: [근거: 오펜스게임_기획서.md §8, constants.ts]

상성 순환 설계

- 의도(Why): 가위바위보 구조로 "정답 편성"이 없게 하고 블라인드 배팅에 의미를 만든다.
- 설계(How): 겹침 허용과 무장 0 정지 예외로 근접 물량이 화력으로 환산되게 했다. 결과 순환은 "장거리 포대 → 근접 물량 → 고기동 러셔 → 장거리 포대"와 "항공 → 지상 → 대공 → 항공". 자폭은 뭉침 카운터로 전 구도에서 강하다. 대안인 기본 사거리 30(광선검 항공기 봉쇄 붕괴)과 유닛 간격 8u(판독성 재설계)는 기각. [근거: 오펜스게임_기획서.md §15-9, §15-11, §16.5]
- 필요(What): 헤드리스 교환비 측정 `npm run sim -- duel`.
- 작업 방법: 동일 예산 560원 교환비를 15쌍 측정해 순환을 확인한다.
- 완료 기준: 순환의 각 고리에서 해당 카운터가 완승한다.
- 리스크/대안: 고정 봇은 카운터를 제때 편성하지 못해 장거리가 여전히 강하다. 120mm 활강포/이족 사거리 하향이 후보다. [근거: 오펜스게임_기획서.md §16.5]
- 근거: [근거: 오펜스게임_기획서.md §16.5]

밸런스 검증 설계

- 의도(Why): 사람의 플레이 없이 상수만 조정하는 루프를 만든다.
- 설계(How): 고정 전략 봇을 100~1000회 대전시키고 좌우 미러를 포함해 편향을 점검한다. 합격 기준은 BAL-1~3. 미달 시 상수만 조정하고 로직은 수정하지 않는다. [근거: 오펜스게임_기획서.md §16, §18 M1.5]
- 필요(What): `scripts/headless.ts`(`sim`, `sim -- duel`, `sim -- bal`, `sim -- perk`).
- 작업 방법: 변경 → 재측정 → 표에 기록.
- 완료 기준: BAL-1~3 모두 합격(현재 BAL-1/2 불합격).
- 리스크/대안: 봇의 편성 능력이 한계라 실제 사람 플레이 데이터가 필요하다.
- 근거: [근거: 오펜스게임_기획서.md §16.5, README.md]

AI 가상 플레이어 설계

- 의도(Why): 온라인 상대가 없어도 연습과 시연이 가능하게 하고, 상성 구조를 체험시키는 교재 역할을 겸한다.
- 설계(How): 5인 각자 고정 덱 6종, 행동 패턴, 고정 퍽을 가진다. 매칭 시 플레이어 레이팅에 가까운 2명 중 1명을 무작위로 고른다. AI는 이미 전장에 나온 유닛만 읽고 플레이어의 이번 라운드 예약은 참조하지 않아 블라인드 규칙을 공유한다. [근거: opponents.ts, README.md]

- 필요(What): `ai/brain.ts`, `ai/opponents.ts`, `sim/builds.ts`.
- 작업 방법: 라운드로빈으로 순위/상성 확인.
- 완료 기준: AI 5인 사이에서 질풍 > 포화 > 창공 > 질풍의 순환이 나타난다. [근거: 오펜스게임_기획서.md §15-9]
- 리스크/대안: 총사령관(ADAPTIVE)의 적응 로직 세부는 `brain.ts`를 읽지 못해 미확인.
- 근거: [근거: opponents.ts]

퍽 설계

- 의도(Why): 계정별 정체성을 주되 대칭 구조를 깨지 않게 효과를 작게 한다.
- 설계(How): 계정당 1개. 효과는 경제 2, 전투 4, 방어 2, 지원 1, 방해 1. 상대가 NULLIFY이면 내 정적 보정, 경제, 방어 효과가 모두 꺼진다. [근거: perks.ts]
- 필요(What): `perks.ts`, `PERK_*` 상수.
- 작업 방법: `sim -- perk`로 미러 승률과 무효화 검증.
- 완료 기준: 10종 모두 단독 보유 시 유리하고 NULLIFY는 효과를 지운다.
- 리스크/대안: 기획서 본문에 퍽 절이 없어 의도 서술은 추정이다. 이런 추가 규칙은 기획서에 환류해야 한다.
- 근거: [근거: perks.ts, constants.ts] [추정: 기획서에 퍽 서술이 없음]

3. 데이터/리소스 명세

3.1 파츠 (`parts.ts`)

ID	분류	가격	보정
B_H 가로 몸통	BODY	30	안정 +10, 팔 슬롯 1
B_V 세로 몸통	BODY	40	안정 −10, 사거리 +20, 팔 슬롯 2
A_ROBOT 로봇 팔	ARM	35	안정 −15, 사거리 +10, 주기 −0.3s, 마운트 1
A_CANNON 120mm 활강포	ARM	60	공격 +20, 사거리 +30, 주기 +1.0s, 마운트 0
A_PYLON 무장창	ARM	15	사거리 −10, 주기 +0.3s, 마운트 1
W_SABER 광선검	WEAPON	50	공격 +30, 사거리 −20
W_BOMB 항공폭탄	WEAPON	20	공격 +20, 스플래시 20u
W_MSL 미사일	WEAPON	45	공격 +10, 사거리 +30
W_CHARGE 자폭 탄두	WEAPON	45	공격 +90, 사거리 −20, 스플래시 30u, 자폭
L_WHEEL 바퀴	LEG	20	기동 +20
L_TRACK 캐터필러	LEG	30	기동 +10, 안정 +10
L_QUAD 사족보행	LEG	45	기동 +15, 안정 +15
L_BIPED 이족보행	LEG	55	기동 +10, 안정 −20, 사거리 +20
L_JET 제트엔진	LEG	90	기동 +80, 안정 −20
S_HEAD 머리	SPECIAL	25	사거리 +10, 안정 −10
S_HMG 중기관총	SPECIAL	30	공격 +10, 주기 −0.4s
S_WING 날개	SPECIAL	70	안정 −20, 비행

ID	분류	가격	보정
S_CHARGE 자폭 장치	SPECIAL	40	공격 +60, 스플래시 45u, 자폭(날개와 배타)
[근거: 오펜스게임_기획서.md §17 parts.ts]			

3.2 핵심 상수 (constants.ts)

상수	값	비고
TICKS_PER_SEC	30	
LIMIT_SEC	600	실시간 누적 제한
SELECTION_MAX_SEC / REVEAL_SEC	60 / 2	
TERM_BASE_SEC / TERM_MAX_SEC	30 / 120	텀 = min(30n, 120)
FIELD_WIDTH	600u	SPAWN_POS_MILLI 15,000
UNIT_GAP_MILLI	12,000	반경 6u×2
MAX_UNITS_PER_SIDE	30	
SPAWN_INTERVAL_TICKS	12	0.4초
CASTLE_HP	3,500	
START_FUNDS / INCOME_PER_SEC	200 / 7	
BASE_STABILITY/BASE_RANGE/BASE_CYCLE_DECI	100 / 20 / 20	
MIN_STABILITY/MIN_CYCLE_DECI	10 / 8	
SPEED_PER_MOBILITY_MILLI	20	
EVASION_BP_PER_MOBILITY / EVASION_BP_CAP	25 / 4,000	
AA_MIN_RANGE / AG_MIN_RANGE	50 / 1	
PRESET_SLOTS/DECK_SIZE/DECK_SAVES	30 / 6 / 5	
ELO_K/RATING_START/RATING_FLOOR	32 / 1000 / 100	
RECONNECT_GRACE_SEC / PRESENCE_PING_SEC	60 / 5	
[근거: constants.ts]		

3.3 대표 빌드 (기획서 §8.3, 14종)

빌드	구성	가격	핵심 역할
방벽	가로+캐터필러	60	최저가 방패
자폭병	가로+무장창+자폭 탄두+바퀴	110	뭉침 처벌
전차	가로+120mm+캐터필러	120	최저가 대공(사거리 50)
대공 스크린	가로+로봇팔+미사일+바퀴	130	대공(사거리 60)
경보병	가로+로봇팔+광선검+바퀴	135	근접 물량
가미카제	가로+제트+자폭 장치	160	고기동 자폭
중장병	가로+로봇팔+광선검+사족	160	오래 버티는 근접
속사병	가로+로봇팔+광선검+중기관총+바퀴	165	고DPS 근접
러셔	가로+무장창+광선검+제트	185	장거리 카운터

빌드	구성	가격	핵심 역할
고정 포탑	세로+120mm×2+머리	185	사거리 110 방어
포병	세로+120mm×2+캐터필러	190	사거리 100
장포대	세로+120mm×2+이족	215	사거리 120
폭격기	가로+무장창+항공폭탄+제트+날개	225	공중 스플래시
쌍검병	세로+로봇팔×2+광선검×2+바퀴	230	최고 DPS(42.9)
[근거: 오펜스게임_기획서.md §8.3]			

3.4 밸런스 목표와 실측

구도	합격 기준	실측	판정
BAL-1 러시 vs 방벽	어느 쪽도 60% 초과 금지	러시 100%	불합격
BAL-2 포병 vs 근접	포병 55% 미만	포병 95%	불합격
BAL-3a 항공 vs 대공 0기	85% 초과	100%	합격
BAL-3b 항공 vs 대공 편성	40% 미만	0%	합격
[근거: 오펜스게임_기획서.md §16.2, §16.5]			

3.5 AI와 퍽

AI	레이팅	패턴	퍽
고철	850	TURTLE	철벽 성채
질풍	960	RUSH	기동 강화
포화	1060	ARTILLERY	조준 보정
창공	1150	AIR	공습 지원
총사령관	1280	ADAPTIVE	퍽 무효화
[근거: opponents.ts]			

퍽 ID	효과 수치
FUNDS_START	자금 +100
INCOME_BOOST	수입 +3/초
RANGE_BOOST	사거리 +8u
ARMOR_BOOST	안정성 +12%
SPEED_BOOST	이동속도 +15%(회피 불변)
RAPID_FIRE	주기 −8%(하한 6틱)
FORTRESS	성 HP +500, 반사 15%
SLOW_FIELD	성 100u 내 적 속도 ×0.7
AIRSTRIKE	45초마다 반경 30u, 80 데미지
NULLIFY	상대 퍽 무효
[근거: constants.ts, perks.ts]	

4. 작업 순서

1. 상수/파츠 표 확정 → 2. `assemble` 및 `npm run check` → 3. 전투 틱 구현 → 4. 결정론 검증 → 5. `sim`, `sim -- duel`로 순환 확인 → 6. AI 5인 라운드로빈 → 7. 퍽 검증 → 8. BAL-1/2 상수 조정(현재 미완). [근거: 오펜스게임_기획서.md § 18]

5. 검수 기준

- 대표 빌드 14종 수치가 표와 정확히 일치한다.
- 사거리 49/50 경계 동작, 같은 시드 비트 동일 재현.
- 죽은 파츠 0종.
- BAL-1~3 합격(현재 불합격 2건, 후속 조정 필요).
- 좌우 미러 승률 일치(실측 일치).

캐슬 브레이커 — 세부기획서 (온라인과 메타)

> 재구성 기획서 (역설계) / 작성 기준 시점: 2026-08-07 (프로젝트 시작 시점) / 문서 버전: Ver 0.1 프로젝트 ID: P016 / 근거 수준: B / 작성: Claude (재구성) 이 문서는 결과물과 제작 기록으로부터 기획 의도를 역추정해 당시 작성한 것처럼 재구성한 것이다. 확인된 사실과 추정을 [근거]/[추정] 태그로 구분한다.

1. 목적과 범위

- 목적: 두 사람이 같은 전투를 보도록 만드는 온라인 구조(매칭/라운드 동기, presence, 종료)와 계정, 레이팅, 랭킹 같은 메타 진행을 정한다.
- 범위: 인증, 프로필 저장, 대기열, 라운드 예약 교환, 해시, presence, 몰수패, Elo, 리더보드, 배포. 전투 규칙과 화면은 제외한다.
- 기획서(Node+ws 서버 권위)와 실제(Cloudflare Workers+D1 폴링)가 다르므로 두 가지를 구분해 쓴다.

2. 설계

동기화 모델

- 의도(Why): 실시간 입력이 없다는 구조적 이점으로 네트워크 부담과 지연 영향을 없앤다.
- 설계(How): 양측 클라이언트가 같은 매치 시드와 같은 라운드 예약을 받아 독립적으로 동일한 시뮬레이션을 돌린다. 교환은 라운드 경계에서 한 번이다. 기획서는 서버가 같은 시뮬레이션을 권위 실행하고 해시 불일치 시 스냅샷으로 교정하며 3라운드 연속 불일치 시 재접속시키도록 정했다. 실제 구현은 해시를 D1에 저장하는 수준까지만 확인된다. [근거: 오펜스게임_기획서.md § 11.2, worker/src/index.ts]
- 필요(What): 결정론 코어, `match_rounds` 테이블(`p1/p2_orders_json`, `p1/p2_hash`).
- 작업 방법: 두 브라우저로 완주해 트래픽이 라운드 경계에만 발생하는지 확인.
- 완료 기준: 두 클라이언트 해시가 라운드마다 일치한다.

- 리스크/대안: 서버 쪽 시뮬레이션/교정이 없으면 한쪽 조작이 검출되지 않는다. 내부 소수 인원 전제로 수용했다. [추정: 서버 시뮬레이션 코드를 읽지 못함]
- 근거: [근거: 오펜스게임_기획서.md §11.2, README.md "핵심 설계"]

매칭

- 의도(Why): 무료 플랜에서 이중 매칭 없이 두 사람을 짝짓는다.
- 설계(How): `/api/queue/join`으로 대기열 등록, `/api/queue/poll`(1.5초)에서 Worker가 직접 후보를 골라 `claimQueuePair`(조건부 UPDATE)로 확정한다. 후보 선택은 레이팅 차 최소. 확정 시 시드 생성 후 `matches` 행을 만든다. 먼저 큐에 들어온 쪽이 P1이라는 기획서 규칙의 구현 여부는 미확인. 대안(Firebase 클라이언트 트랜잭션)은 서버 코드가 없어 임시방편이었으므로 폐기했다. [근거: worker/src/matchmaking.ts, README.md]
- 필요(What): `queue`, `matches` 테이블.
- 작업 방법: 두 탭으로 동시 대기 → 하나의 매치만 생기는지 확인.
- 완료 기준: 이중 매칭 0건, 취소(`/queue/leave`) 시 대기열 제거.
- 리스크/대안: 오래된 큐 행 정리 로직 미확인.
- 근거: [근거: matchmaking.ts, index.ts]

라운드 예약 교환

- 의도(Why): 블라인드 규칙을 서버 보관으로 지킨다.
- 설계(How): 클라이언트가 `/orders`로 `{side, round, orders}`를 제출하고 `/round/:n`을 0.8초 폴링해 양측 예약이 모이면 REVEAL로 간다. 상대 준비 여부만 보이고 내용은 REVEAL 전까지 비공개라는 기획서 규칙은 서버가 양측 모두 제출한 뒤에만 내용을 돌려주는 방식이라야 지켜지는데, 실제 `getRound`의 응답 정책은 미확인이다. 기획서 M4 수용 기준 ④(조작된 예약 거부)도 서버 코드에서 확인하지 못했다. [근거: worker/src/index.ts, 오펜스게임_기획서.md §11.4, §18]
- 필요(What): `submitOrders`, `getRound` 헬퍼(`db.ts`).
- 작업 방법: 한쪽만 제출했을 때 상대 내용이 노출되지 않는지 확인.
- 완료 기준: 양측 제출 전에는 내용이 비공개다.
- 리스크/대안: 미검증 시 클라이언트가 서버 응답을 임의 해석할 수 있다. 검증 로직을 Worker에 추가한다 [제안값: 잔액/슬롯 범위 서버 재검증].
- 근거: [근거: worker/src/index.ts] [추정: 헬퍼 본문 미확인]

연결 끊김과 종료

- 의도(Why): 한쪽이 사라져도 매치가 영원히 멈추지 않게 한다.
- 설계(How): 클라이언트가 5초마다 `/presence`로 핑을 보내고 상대는 `staleMs`가 60초를 넘으면 stale로 본다. 기획서 §11.5는 선택 중 끊김 시 60초 유예 후 몰수패, 실시간 중 끊김은 서버 계속 진행/스냅샷 복귀, 양측 동시 끊김은 무승부로 정의하나 구현은 stale 표시까지이며 자동 몰수패는 없다. 종료는 클라이언트가 `/end`로 승자/사유를 보고한다. 향후 Durable Objects(유료)로 올리면 소켓 close로 즉시 판정하며, 이때 `backend/cloudflare/matchmaking.ts/match.ts`만 교체하도록 공개 인터페이스를 유지했다. [근거: README.md, worker/src/index.ts, 오펜스게임_기획서.md §11.5]

- 필요(What): `players/matches`의 `p1_last_seen`, `p2_last_seen`.
- 작업 방법: 클라이언트 종료 후 60초 뒤 상대 화면에 몰수승 안내가 뜨는지 확인(미구현).
- 완료 기준: 유예 초과 시 서버나 상대 클라이언트가 `/end`를 호출해 몰수패로 기록된다.
- 리스크/대안: 클라이언트 보고 방식은 양쪽이 모두 종료하면 기록이 누락된다. 임시안은 Worker에서 stale 확인 시 `/end` 자동 호출 [제안값].
- 근거: [근거: README.md, constants.ts]

계정/프로필

- 의도(Why): 내부 소수 인원이므로 가입 마찰을 없애고 기록을 유지한다.
- 설계(How): 기획서는 이름 정규화(trim+소문자) 키, 2~12자 한글/영문/숫자, 비밀번호 없음, 중복 접속 시 기존 세션 종료(`SessionTakenOver`)를 정했다. 실제는 클라이언트 uid(`crypto.randomUUID`, localStorage 고정)로 계정을 식별하고 `/api/identify`가 이름으로 조회/생성한다. 프리셋/덱은 JSON 문자열로 저장하고 `PUT /api/profile/:uid`로 갱신한다. 신규는 대표 빌드를 설계 슬롯에 채우고 6종을 기본 덱으로 편성해 즉시 출격 가능하다. [근거: 오펜스게임_기획서.md §12.3, README.md, 0001_init.sql]
- 필요(What): `players` 테이블, `storage/local.ts`(오프라인).
- 작업 방법: 같은 이름 재접속 시 프로필 복원 확인.
- 완료 기준: 프리셋 30/덱 5/레이팅 복원.
- 리스크/대안: 이름만 알면 접속되는 구조는 의도된 제약. PIN은 확장 항목 최우선. `SessionTakenOver` 구현 여부 미확인.
- 근거: [근거: 오펜스게임_기획서.md §12.3, §19]

Elo와 리더보드

- 의도(Why): 판수가 아니라 실력이 순위를 정하게 한다. 고정 증감(+30/−20)은 판수에 수렴해 기각.
- 설계(How): 시작 1000, 기대승률 `1/(1+10^((Rb-Ra)/400))`, `ΔR=round(32×(S-Ea))`, 하한 100, 몰수패도 동일. 리더보드는 `idx_players_rating` 인덱스로 상위 50위와 본인 고정 행. [근거: 오펜스게임_기획서.md §12.5, §15-7, constants.ts, 0001_init.sql]
- 필요(What): `sim/elo.ts`, `/api/leaderboard`.
- 작업 방법: 승, 패, 무 케이스 단위 테스트.
- 완료 기준: 기획서 §12.5 산식과 일치.
- 리스크/대안: 레이팅 갱신을 누가 호출하는지(클라이언트 보고 vs 서버)는 미확인.
- 근거: [근거: constants.ts]

배포/운영

- 의도(Why): 무료 호스팅과 push 한 번으로 갱신한다.
- 설계(How): `main` push 시 `deploy-cloudflare.yml`이 Worker → Pages 순으로 재배포. 필요한 비밀키는 `CLOUDFLARE_API_TOKEN`, `CLOUDFLARE_ACCOUNT_ID`. D1 마이그레이션은 스키마 변경 시에만 수동. GitHub Pages 미러(`deploy-pages.yml`)는 `VITE_WORKER_URL`이 없어 AI 연습만 가능하다. 레거시 Firebase 코드는 저장소에 남아 있으나 참조하지 않는다. [근거: README.md]

- 필요(What): Cloudflare 계정, `wrangler.toml`의 D1 `database_id`.
- 작업 방법: 수동 배포 순서는 README에 명시.
- 완료 기준: Pages URL에서 온라인 대전 버튼이 활성화된다.
- 리스크/대안: 현재 서비스 작동 여부 미확인.
- 근거: [근거: README.md]

3. 데이터/리소스 명세

3.1 D1 테이블 (`0001_init.sql`)

테이블	주요 필드	비고
`players`	`uid` PK, `display_name`, `rating`(기본 1000), `wins/draws/losses`, `tutorial_seen`, `perk`, `bio`, `active_deck_id`, `presets_json`, `decks_json`, `created_at`, `last_seen_at`	`idx_players_rating`
`queue`	`uid` PK, `display_name`, `rating`, `bio`, `perk`, `deck_json`, `joined_at`, `matched_match_id`	값이 채워지면 매칭 성립
`matches`	`id` PK, `seed`, `created_at`, `status`(기본 active), `p1_uid/p2_uid`, `p1_json/p2_json`, `winner`, `end_reason`, `p1_last_seen/p2_last_seen`	`idx_matches_p1`, `p2`
`match_rounds`	(`match_id`, `round`) PK, `p1_orders_json`, `p2_orders_json`, `p1_hash`, `p2_hash`	
[근거: worker/migrations/0001_init.sql]		

3.2 API (`worker/src/index.ts`)

메서드/경로	용도
POST `/api/identify`	이름으로 조회/생성
PUT `/api/profile/:uid`	프로필 저장
GET `/api/leaderboard`	리더보드
POST `/api/queue/join`, `/api/queue/leave`	대기열
GET `/api/queue/poll?uid=`	매칭 확인/시도
POST `/api/match/:id/orders` / GET `/round/:n`	라운드 예약
POST `/api/match/:id/hash`	상태 해시
POST/GET `/api/match/:id/presence`	접속 핑/stale
POST `/api/match/:id/end`	종료 기록
[근거: worker/src/index.ts]	

3.3 폴링 주기

항목	값
대기열 폴링	1.5초
라운드 예약 폴링	0.8초
presence 핑	5초(`PRESENCE_PING_SEC`)
재접속 유예	60초(`RECONNECT_GRACE_SEC`)
[근거: README.md, constants.ts]	

4. 작업 순서

1. D1 스키마 → 2. identify/profile → 3. 리더보드/Elo → 4. 대기열/매칭 → 5. 라운드 예약/해시 → 6. presence/종료 → 7. GitHub Actions 배포 → 8. 몰수패 자동화/서버 검증(미완).

5. 검수 기준

- 두 브라우저로 한 판 완주, 실시간 구간 트래픽 0.
- 동시 큐 진입 시 매치 1개.
- 같은 이름 재접속 시 데이터 복원.
- 한쪽 종료 후 60초가 지나면 몰수패가 기록된다(현재 불합격).
- 조작된 예약/스텟을 서버가 거부한다(현재 미확인).