천호: 살과 강철 — 메인기획서

> 재구성 기획서 (역설계) / 작성 기준 시점: 2026-08-04 (프로젝트 시작 시점) / 문서 버전: Ver 0.1 프로젝트 ID: P019 / 근거 수준: C / 작성: Claude (재구성) 이 문서는 결과물과 제작 기록으로부터 기획 의도를 역추정해 당시 작성한 것처럼 재구성한 것이다. 확인된 사실과 추정을 [근거]/[추정] 태그로 구분한다.

1. 핵심 메카닉스

이동, 자세, 소음

- 의도(Why): 속도와 은밀함을 맞바꾸는 가장 기본 선택을 만든다.
- 설계(How): 자세 3종. 은신 1.15 / 걷기 2.45 / 달리기 5.0 m/s. 발소리가 누적되어 1.0이 되면 소음 방출: 달리기 반경 15, 걷기 8, 은신 3. 운반 중 발소리 누적 1.7배. 근육 파츠는 속도 1.16배, 운반 페널티 0.55→0.62. 운반 중 이동 배율은 개조 `joints`로 페널티 절반. [근거: cheonho-flesh.html]
- 필요(What): 키 입력, 충돌 그리드(1m 셀), 플레이어 반경 0.42m.
- 작업 방법: 입력→`playerSpeed`→`moveEntity`→소음 누적→`makeNoise`. 검증은 튜토리얼 거리 측정 (걷기 20m, 달리기 10m, 은신 7m).
- 완료 기준: 세 자세의 속도/소음 차이를 튜토리얼에서 체감한다.
- 리스크/대안: 달리기 일변도 → 소음 파동을 노란 원으로 시각화(튜토리얼 안내).
- 근거: [근거: cheonho-flesh.html]

감지(의심 게이지)

- 의도(Why): 잠입의 판정을 눈에 보이는 규칙으로 만든다.
- 설계(How): 보행자마다 시야 거리 vr/각도 va/예민도. 시민은 범죄 상태가 아니면 의심을 줄이며 지나친다. 범죄 상태는 운반 중, 수배 중, 범죄 타이머 중, 트렁크 시체 적재 중 하나. 의심 증가율 = (0.35 + 근접도 ×1.35) × 조명계수 × 엄폐계수 × 노출 × 예민도 × 파츠배율. 조명계수 = 0.28 + 밝기×0.72, 엄폐계수 = 1 − 엄폐×0.40. 노출: 은신 0.38 / 걷기 0.60 / 달리기 1.25, 운반 중 ×2.4. 의심 상한: 시민 평상 0.76, 범죄 중 1.35, 경찰/경비 1.2. 1.0 도달 시 목격. 휴대폰 보는 시민은 50% 확률로 시야 무효. 길에 놓인 시체는 13m 안의 시선이 닿는 사람의 의심을 올린다. [근거: cheonho-flesh.html]
- 필요(What): 조명/엄폐 그리드(4m 셀), `losBlocked` 시야 검사.
- 작업 방법: 수식 확정 → 구현 → HUD 노출 수치로 확인.
- 완료 기준: 가로등 밑 달리기는 빨리 들키고, 그늘 은신 보행은 근접해도 한참 버틴다.
- 리스크/대안: 곱셈 구조라 한 변수만 바꿔도 체감이 크게 변함 → 밸런스 5절에서 변수별 측정.
- 근거: [근거: cheonho-flesh.html]

목격, 수배, 추격

- 의도(Why): 걸렸을 때의 연쇄 반응을 도시 규모로 보여 준다.
- 설계(How): 목격 시 시민은 비명 후 도주(6~11초), 경찰/경비는 추격. 반경 42m 내 주변인 의심 0.65로 상승, 경찰이면 추격 전환, 150m 내 경찰은 수색 상태. 수배(최대 6)는 상승 간 3초 쿨다운. 추적 시야 밖에서

시간 `13 + 수배×4`초 지나면 1 감소. 경찰 최대 10명은 `1 + 수배×1.6 + 악명×0.4`로 증가. 수배 2 이상에서 순찰차 수배–1대(최대 6). 경찰 근접 1.8m이면 체포. 차량 중에는 1.0초 매달리면 체포. [근거: cheonho-flesh.html]

- 필요(What): 도로 그래프 위 보행자 AI, 순찰차 AI, 상태 머신(walk, stand, alert, flee, chase, search).
- 작업 방법: 상태 전이표 작성 → 구현 → 추적 이탈 시나리오 테스트.
- 완료 기준: 수배 3에서 추격을 지하철로 끊을 수 있다.
- 리스크/대안: 추격이 지나치게 끈질길 수 있음 → 추적 포기 22초 후 수색 전환 값으로 조정.
- 근거: [근거: cheonho-flesh.html]

무력화, 운반, 반입

- 의도(Why): 위험을 감수해 표적을 데려오는 핵심 행동.
- 설계(How): E 길게 눌러 근접 대상 무력화. 시간 정면 2.4초, 뒤 1.15초(배후 판정 각도 1.35rad 초과). 개조 `tools` ×0.6, 근육 ×0.85. 완료 시 소음 반경 정면 16/뒤 7, 범죄 타이머 40초. 시체는 F로 들고 내려놓는다. 차량에 타면 트렁크에 적재, 트렁크 개조가 없으면 발각 위험 노출 0.25(있으면 0.10). 은신처 반경 안에서 자동 반입. [근거: cheonho-flesh.html]
- 필요(What): 근접 판정, 보관고 슬롯(기본 2).
- 작업 방법: 접근 → 배후 → 무력화 → 반입 순서를 튜토리얼 후 브리핑으로 안내.
- 완료 기준: 표적 반입 시 표적 조직이 보관 목록에 표시된다.
- 리스크/대안: 표적이 사고사하면 `respawnTarget`이 110m 이상 떨어진 곳에 새 표적을 세움. 임무 정체 방지.
- 근거: [근거: cheonho-flesh.html]

지하철 잠적

- 의도(Why): 도시 규모 추격에서 빠져나가는 유일한 합법 탈출구이자 지도 지식 보상.
- 설계(How): 출구 3.2m 안에서 E. 누가 보고 있으면 불가, 차 탑승 중 불가. 같은 역의 25m 이상 떨어진 다른 출구 중 경찰에서 가장 먼 곳으로 이동(표적 운반 중이면 은신처와의 거리 가중). 추격자는 수색 전환, 수배 1 감소. [근거: cheonho-flesh.html]
- 필요(What): OSM 지하철 출구(천호역 5/8호선, 강동역 5호선 명명).
- 작업 방법: 출구 클러스터링 250m → 역 이름 부여.
- 완료 기준: 추격 중 지하철로 한 구간 이탈이 가능하다.
- 리스크/대안: 역 이름은 출구 수가 많은 순서로 임의 부여 → 오배정 가능. [추정: 코드가 정렬 순서로 이름을 매김]
- 근거: [근거: cheonho-flesh.html, bake.mjs]

해체/이식(성장)

- 의도(Why): 위험의 보상과 대가를 한 곳에서 정산한다.
- 설계(How): 해체대에서 재료 환산, 표적이면 조직 확보, 악명 +1(소각로 있으면 0, 뼈 장식 있으면 ×0.75). 이식대에서 조직을 파츠로 전환하고 챕터 +1. 파츠 7종마다 이득과 대가가 쌍으로 붙는다(03 문서 표). 7장 조직은 이식 대신 최종 선택으로 연결. [근거: cheonho-flesh.html]

- 필요(What): 재료 5종, 파츠 정의표.
- 작업 방법: 패널 UI에 이득/대가를 같은 화면에 병기.
- 완료 기준: 이식 후 HUD, 감지, 이동에 해당 효과가 반영된다.
- 리스크/대안: 일부 효과 구현 여부 미확인(00 문서 6절).
- 근거: [근거: cheonho-flesh.html]

2. 구현 방법

- 구조: 단일 HTML 안에 `<script>` 하나, 전역 상태 `S`, 맵 데이터 `window.MAPDATA`를 마지막 스크립트로 임베드하고 `__boot()` 호출. [근거: cheonho-flesh.html, inject.mjs]
- 데이터 파이프라인: OSM XML → `bake.mjs`(좌표 변환, 도로 폴리라인 단순화 RDP, 면, 건물, POI, 담장 추출, 1m 래스터 마스크, 블록 라벨링, 그리디 건물 생성, 도로 그래프) → `map.json` → `inject.mjs` → HTML. `viz.mjs`로 PNG 미리보기. [근거: bake.mjs, inject.mjs, viz.mjs]
- 좌표 변환: 위도 중점 기준 경도 1도 약 88,266m, 위도 1도 110,574m. bbox 127.1145,37.5315,127.1325,37.5525. [근거: bake.mjs, map.json]
- 도로 등급 6단계(0 고속화~5 골목), 등급별 기본 폭(예: primary 30m, residential 8m), `lanes`가 있으면 차선×3.35m 기반 폭 재계산. [근거: bake.mjs]
- 건물: OSM 실측 + 빈 블록은 시드 난수(20260804)로 사각 건물 채움. 90셀 미만 자투리 제외, 18% 확률로 큰 블록(4000셀 초과)을 주차장/공터로. [근거: bake.mjs]
- 런타임: 충돌 1m 그리드, 조명/엄폐 4m 그리드, 공간 인덱스 72m, 건물은 5% 축소 렌더. 조명: 도로 등급별 가로등 간격 28~46m, 상점 간판 POI 조명. 엄폐도: 주변 3셀(12m) 내 고체 비율×420. [근거: cheonho-flesh.html]
- 상태 머신: 장면 `city|hideout`, 보행자 `walk|stand|alert|flee|chase|search`, 플레이어 `sneak|walk|run`. [근거: cheonho-flesh.html]
- 인수인계: HANDOFF 기록 없음. 재생성 명령은 코드 머리 주석(OSM API → bake → MAPDATA)이 유일. [근거: cheonho-flesh.html]

3. 시스템 간 상호작용과 밸런스 기준

- 핵심 변수: 노출(자세) × 조명 × 엄폐 × 파츠 배율 × 예민도 → 의심 속도. [근거: cheonho-flesh.html]
- 파츠 배율: `1 - 0.085×(이식 파츠 수)`, 위장 옷장 ×0.8, 하한 0.25. 눈(강한 조명 0.62 초과 시 ×1.25)/심장(과열 0.5 초과 시 ×1.18)/피부(손상 시 ×1.3)는 페널티, 파츠 정비대가 페널티를 40% 줄임. [근거: cheonho-flesh.html]
- 악명: 체포 시 +1. 순찰 밀도 +10%/점이라 안내하나 코드는 경찰 수 계산에 ×0.4/점 반영. 해체마다 상승. 소각로는 악명 감소 수단. [근거: cheonho-flesh.html]
- 체포 손실: 고기/뼈 50%, 부품 60% 잔존, 보관 시체 전부 상실. 위조 신분증 1회 방어. [근거: cheonho-flesh.html]
- 곡선 목표 [제안값]: 1장 취객은 2~3분 내 성공, 6~7장은 수배 3 이상 상황 1회 이상. 실측 기록 없음.

4. 게임의 나머지 부분의 방향

- 콘텐츠 구조: 7챕터 직선 진행, 챕터마다 지정 구역/표적 유형(취객, 러너, 가수, 간호, 배달, 경비, 연구원). 구역 밖 시민도 재료로 활용 가능. [근거: cheonho-flesh.html]
- 메타 진행: 개조 18종(기능 13/확장 2/소모품 1/장식 2), 악명, 재료. 별도 레벨업 없음. [근거: cheonho-flesh.html]
- UI/UX 방향: 상황 프롬프트 중심, 말투는 1인칭 독백. 03 세부기획서 참고.
- 아트/사운드: 코드 렌더와 합성음 유지. 추가 에셋 계획 기록 없음.
- 저장/설정: localStorage 키 `cheonho_flesh_v1`, 버전 2. 챕터, 파츠, 개조, 재료, 악명, 조직, 누적시간, 보관, 소모품을 저장하며 은신처 입장과 체포, 반입 때 저장. 설정 메뉴 기록 없음. [근거: cheonho-flesh.html]

5. 제작 과정 설계

1. 지도 파이프라인 확립(선행). 2. 이동, 충돌, 카메라. 3. 감지, 수배, AI. 4. 납치, 운반, 차량. 5. 은신처 씬. 6. 챕터/엔딩. 7. 튜토리얼/저장. 8. 플레이 검증(미착수 추정). [추정: 코드 섹션 번호 순서가 제작 순서와 같을 가능성] - 병렬 가능: 은신처 렌더와 오디오, 인트로 슬라이드는 도시 AI와 독립. - AI/도구 분담: 기획이 규칙 정의, AI가 구현. 실제 분담 기록 없음. - 검증: 브라우저에서 튜토리얼 통과 → 챕터 1~7 순주행 → 체포 복귀 → 로드. 인계는 HANDOFF 작성 필요(현재 없음).

6. 세부기획서 목록

- `03_세부기획서_챕터-은신처-수치.md`: 챕터 구역/표적 표, 파츠/개조 데이터, 감지 수치표, UI 구성, 검수 기준.