흥부전 (소설->게임) — 세부기획서: UIUX와아트사운드

> 재구성 기획서 (역설계) / 작성 기준 시점: 2026-08-04 (프로젝트 시작 시점) / 문서 버전: Ver 0.1 프로젝트 ID: P012 / 근거 수준: A / 작성: Codex (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

플레이어가 비용과 결과를 읽고 행동한 뒤, 그 결과가 흥부의 자세, 사물, 소리로 돌아오게 한다. [근거: ART_DIRECTION.md §1/§5] 기능 정보를 한지, 먹선, 오방색 안에 배치하되 장식이 인물/입력 안내/목표를 가리지 않도록 한다. [근거: ART_DIRECTION.md §2/§6]

범위

제목, 걷기, 작업, 문, 장, 제비 선택/밤 정산/계절 전환, 톱질, 개봉, 엔딩의 입력과 표현을 다룬다. [근거: BUILD_BRIEF.md; play.js; scenes.js] 외부 이미지, 웹폰트, 오디오 파일 없이 Phaser 도형/시스템 글꼴/WebAudio를 사용하며, 음성/상시 배경음악은 사용하지 않는다. [근거: BUILD_BRIEF.md; ART_DIRECTION.md §8/§9]

2. 설계

정보 배치 — 비용과 결과가 다른 자리를 차지한다

- 의도(Why): 행동 중 바뀌는 자원 문구가 흥부와 상호작용 대상을 가리지 않도록 한다. [근거: ART_DIRECTION.md §5; S1 16:45~16:57]
- 설계(How): 1280×800 기준 상단 HUD 높이 72, 플레이 영역 y 76~716, 반응 띠 y 624~688, 목표 영역 y 720~776으로 나누며 순간 결과 문구는 반응 띠 안에 배치한다. [근거: config.js LAYOUT]
- 필요(What): 자원, 목표, 프롬프트, 반응 문구/막간 카드와 인물 보호 공간이 필요하다. [근거: hud.js; ART_DIRECTION.md]
- 작업 방법: AI가 HUD와 장면을 분리하고 기획이 작업, 문, 제비의 복수 반응이 겹치는 상황을 캡처해 대조한다. [추정: 세션의 반복 중첩 수정 경험에 따른 절차]
- 완료 기준: 상주 자원/다음 비용/현재 목표와 최근 결과를 동시에 읽을 수 있다. [근거: GAME_DESIGN.md §7; ART_DIRECTION.md §5]
- 리스크/대안: 설계의 HUD 형상 아이콘은 텍스트로 대체된 상태이므로 미구현을 완료로 표시하지 않고 QA-03으로 남긴다. [근거: QA_REPORT.md QA-03]
- 근거: [근거: config.js; hud.js; QA_REPORT.md]

입력/모드 — 선택과 확정을 구분한다

- 의도(Why): 걷는 동안의 입력이 거래, 구조, 톱질 화면으로 중복 전달되지 않게 한다. [추정: play.js의 모드별 조기 반환 구조]
- 설계(How): 걷기는 방향키/WASD, 행동은 Z/Space, 장은 위아래 선택과 확인키 거래 및 Esc/X 종료로 구성하며, 제비 화면의 확인키는 구조/Esc와 X는 지나침 확정으로 처리한다. [근거: play.js update]
- 필요(What): 걷기, 장, 제비, 톱질, 막간 모드와 각 모드의 입력 안내, 선택 불가 이유가 필요하다. [근거: play.js]
- 작업 방법: AI가 모드별 입력을 연결하고 기획이 같은 키가 화면마다 어떤 결과를 만드는지 실제 상태와 대조한다. [추정: 입력 의미 일관성을 검수하기 위한 절차]
- 완료 기준: 장을 닫을 때 재산이 바뀌지 않고 막간에서는 확인키로 다음 장면을 진행한다. [근거: play.js] 제비 지나침의 확정 결과는 입력 전에 명확히 안내한다. [추정: 되돌릴 수 없는 입력의 안내 기준]
- 리스크/대안: 브리프의 일반적인 'Esc 취소'와 제비의 실제 지나침이 충돌하므로 해당 화면에 결과를 명시하고 단순 취소 기능처럼 설명하지 않는다. [근거: PRODUCT_BRIEF.md; play.js chick 분기]
- 근거: [근거: play.js; PRODUCT_BRIEF.md]

첫 이용/접근성 — 규칙을 화면에서 이해한다

- 의도(Why): 원작 지식을 가진 제작자가 아는 내용을 신규 이용자도 전달받았다고 착각하지 않도록 한다. [근거: QA_REPORT.md §10]
- 설계(How): 제목 조작 안내, 아내의 생계 대사, 상단 식구/굶은 날, 하단 오늘의 목표를 제공하고 T로 100%/125% 문안 크기, M으로 움직임 감소를 전환한다. [근거: BUILD_BRIEF.md; hud.js; play.js]
- 필요(What): 한국어 문안, OS 움직임 감소 선호 읽기, 최소 화면과 확대 상태, 안내 없는 시연 절차가 필요하다. [근거: main.js; PLAYTEST_PROTOCOL.md]
- 작업 방법: AI는 표시/넘침을 검사하고 기획은 첫 1분 신분, 목표, 종료, 장르 이해와 2분의 행동 이해를 관찰한다. [근거: PLAYTEST_PROTOCOL.md]
- 완료 기준: 원작 지식과 화면이 알려준 내용을 구분한 이용자 답변을 확보하며, 자동 문구 검사는 표시 여부의 근거로만 사용한다. [근거: QA_REPORT.md §10; PLAYTEST_PROTOCOL.md]
- 리스크/대안: 기록상 최초 이용자 검증은 미수행이므로 통과로 쓰지 않고 사람 시연의 미완료 항목으로 남긴다. [근거: _progress.md; QA_REPORT.md]
- 근거: [근거: main.js; hud.js; PLAYTEST_PROTOCOL.md]

인물/사물 — 상태를 먹선으로 표현한다

- 의도(Why): 염치, 기력, 선택의 결과를 숫자와 함께 생활 장면에서 느끼게 한다. [근거: ART_DIRECTION.md V1/V3]
- 설계(How): 흥부는 기력 5~6, 3~4, 0~2에 따라 자세를 바꾸고, 당사실 소모 시 아내의 실패를 없애며, 닫힌 문은 먹 가로줄과 '닫힘' 글자를 남기고, 구조한 제비는 흰 실로 표시한다. [근거: BUILD_BRIEF.md; play.js]
- 필요(What): 흥부, 아내, 제비, 문, 박, 톱의 상태별 도형과 한지 3색조, 먹선/오방색 팔레트가 필요하다. [근거: BUILD_BRIEF.md; ART_DIRECTION.md §4]
- 작업 방법: AI가 절차 도형을 생성하고 기획이 숫자만 바뀌는 상태가 없는지 상태 전후 이미지를 비교한다. [추정: V1 원칙을 검수로 옮긴 절차]

- 완료 기준: 구조됨, 닫힘, 기력 감소를 색에 의존하지 않고 문자와 형태로 구분한다. [근거: ART_DIRECTION.md §4]
- 리스크/대안: 제목의 작은 제비는 식별성이 낮다는 기록이 있으므로 실루엣 판독을 별도 확인하며, 실제 스크린샷 재검수 전 해결 완료로 쓰지 않는다. [근거: QA_REPORT.md QA-08]
- 근거: [근거: ART_DIRECTION.md; BUILD_BRIEF.md; play.js]

톱질/음향 — 보상의 순간도 노동으로 만든다

- 의도(Why): 박을 버튼 한 번으로 받는 보상 대신 부부가 톱을 마주 잡는 반복 동작으로 체험하게 한다. [근거: source-text.txt U6; S1 16:58~17:01]
- 설계(How): 흥부와 아내 사이 박을 두고 톱을 왕복시키며 일반 입력은 좌우 6회/움직임 감소는 확인키 3회로 완료하고, 개봉 내용을 막간 카드로 제시한다. [근거: config.js; play.js drawSawArt/sawInput]
- 필요(What): 박 크기 단계, 톱, 부부 도형, 진행점, 내용 문안과 톱질/개봉 합성음이 필요하다. [근거: BUILD_BRIEF.md; play.js; audio.js]
- 작업 방법: AI가 입력 진행과 도형/소리를 연결하고 기획이 톱 접촉, 오입력 무진행, 감소 모드의 동일 결과를 확인한다. [추정: 세션의 톱질 그림 수정과 상태 규칙에 따른 절차]
- 완료 기준: 연출 모드와 관계없이 같은 비용으로 같은 박을 열며 음성이 없어도 결과를 이해한다. [근거: ART_DIRECTION.md §7/§8; play.js]
- 리스크/대안: 국악 음색을 지향하는 합성음은 실제 악기 녹음과 구분하고 문을 여는 불편함/노동요 느낌은 청감 검증 전 미확인으로 둔다. [근거: audio.js; PLAYTEST_PROTOCOL.md]
- 근거: [근거: play.js; audio.js; ART_DIRECTION.md]

3. 데이터/리소스 명세

화면/표현 데이터 명세

필드명	타입	범위/허용값	예시/용도
`HB.LAYOUT.W / H`	정수	1280 / 800	가상 캔버스. [근거: config.js]
`FEEDBACK_BAND`	정수 2원소 배열	`[624,688]`	결과 문구의 전용 띠. [근거: config.js]
`GOAL_LINE`	정수 2원소 배열	`[720,776]`	오늘 목표 표시 영역. [근거: config.js]
`mode`	문자열	`walk`, `market`, `chick`, `saw`, `curtain`	`saw`: 보행 입력 차단. [근거: play.js]
`HB.reduced`	불리언	참/거짓	참: 톱질 확인키 방식. [근거: main.js; play.js]
문안 배율 `scaleT`	실수	1 / 1.25	`T`로 전환. [근거: hud.js; play.js]
`sawProgress` / `sawExpect`	정수 / 문자열	정상 0~6 / `left`, `right`	첫 기대 방향은 왼쪽. [근거: play.js]
문안 `HB.S`	객체	문자열/문안 생성 함수	제목, 목표, 비용, 엔딩. [근거: strings.ko.js; scenes.js]

리소스와 사건 연결

리소스/사건	제작 방식/역할
한지, 먹선, 오방색	Canvas/Phaser 도형, 기본 한지 #F2EDE1/먹 #2B2622, 청 #2E5C8A/적 #B33A3A/황 #C9A227/회록 #7A8471. [근거: ART_DIRECTION.md § 4]
흥부, 아내, 성체, 새끼	자세, 소지품, 흰 실을 상태에 연결하는 절차 도형. [근거: BUILD_BRIEF.md]
작업, 문, 박, 톱	작업 7종/문 3종/박 4단계와 톱질 그림을 구성한다. [근거: config.js; BUILD_BRIEF.md]
작업/자원 회수	소고 계열과 놋그릇 계열 합성음으로 누름/보상을 구분한다. [근거: audio.js sogo/brass]
문/구조	북 테 긁기 계열과 대금 계열 합성음으로 비용/구조를 구분한다. [근거: audio.js scrape/daegeum]
정산, 톱질, 개봉, 엔딩	징, 박자, 짧은 합주를 쓰며 굶주림 엔딩에는 새 음악을 추가하지 않는다. [근거: ART_DIRECTION.md § 8; scenes.js]

문안은 한 파일 집중을 목표로 하지만 현재 `play.js`의 상태 라벨과 `scenes.js`의 조언 일부가 직접 문자열이므로, 완전한 문안 분리 완료라고 쓰지 않는다. [근거: play.js spotLabel; scenes.js advice] 기능 라벨 13px 이상/본문 15px 이상은 미술 문서의 목표다. [근거: ART_DIRECTION.md § 9] 최소 화면에서의 실제 축소 크기 판독은 별도 검수한다. [추정: 가상 캔버스 배율을 고려한 검수 해석]

4. 작업 순서

화면 구획과 상태별 문구를 먼저 연결하고 인물, 지점, 반응 도형을 배치한 뒤 톱질, 개봉, 음향을 통합하며, 최소 화면, 확대, 움직임 감소에서 중첩을 확인한다. [추정: 기존 세션의 가시성 수정 순서를 반영한 재구성 계획] 마지막에 스크립트를 HTML에 인라인하고 별도 위치에서 제목, 걷기, 작업을 확인하는 교부 검수를 수행하도록 한다. [근거: bundle-single-file.mjs; S1 17:24~17:25] 명칭은 최신본의 '염치의 한 해'와 standalone/Downloads의 '구걸로 먹고살기 시뮬레이터'를 구별해 표시한다. [근거: 최신본/standalone의 strings.ko.js; Downloads HTML 머리말] 포장 시 어느 소스를 썼는지 기록한다. [추정: 사본 혼동 방지 기록 방식]

5. 검수 기준

제목, 마당, 들, 형님 집, 제비, 밤 정산/계절 전환, 톱질, 개봉, 세 엔딩의 실제 화면과 입력 결과를 확인한다. [근거: BUILD_BRIEF.md; QA_REPORT.md] 1280×800과 1024×640, 문안 125%, 움직임 감소에서도 인물, 문구, 목표가 중첩되지 않아야 한다. [근거: BUILD_BRIEF.md; QA_REPORT.md 화면/접근성 검사] HUD 형상 아이콘, 밤 정산/계절 전환 단독 프레임, 제목 제비 식별성의 미해결 기록을 유지하며 첫 이용자 이해/음향 청감/실기 프레임 분포를 미확인으로 남긴다. [근거: QA_REPORT.md QA-03, QA-04, QA-08 및 미검증 범위]

흥부전 (소설->게임) — 세부기획서: 성장경제와데이터

> 재구성 기획서 (역설계) / 작성 기준 시점: 2026-08-04 (프로젝트 시작 시점) / 문서 버전: Ver 0.1 프로젝트 ID: P012 / 근거 수준: A / 작성: Codex (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

성장을 재화의 무한 축적이 아니라 끼니를 유지하면서 남은 염치와 기력으로 더 많은 박을 여는 능력으로 설계한다. [근거: CONCEPT.md; GAME_DESIGN.md §9] 자원은 생산, 소비, 판정, 결산에서 쓰이는 값을 중심으로 두고, 구현 수치는 `config.js`와 `state.js`에서 읽은 값만 확정한다. [근거: GAME_DESIGN.md §8; config.js; state.js]

범위

기력, 쌀, 냥, 짚, 염치, 당사실, 굶은 날/켠 박, 작업, 문, 상점, 급식, 톱질의 연결을 다룬다. [근거: GAME_DESIGN.md 자원 표] 경험치, 레벨, 장비, 영구 강화/확률형 보상은 해당 없음이며, 게임 규칙은 결정론적으로 설계한다. [근거: GAME_DESIGN.md §7/§15; config.js; state.js]

2. 설계

급식 경제 — 먹은 양이 곧 다음 날 예산이다

- 의도(Why): 쌀을 결산 점수가 아니라 노동을 재생산하는 자원으로 만든다. [근거: GAME_DESIGN.md W1]
- 설계(How): 급식량은 `min(쌀,3)`이며 0→기력 1, 1→3, 2→5, 3→6으로 변환하고, 급식 0이면 굶은 날을 1 늘리며 누적 3에서 종료한다. [근거: config.js WAKE/STARVE_LIMIT; state.js sleep]
- 필요(What): 기력 최대 6, 쌀 잔량, 마지막 급식량, 굶은 날, 정산 표시가 필요하다. [근거: config.js; state.js]
- 작업 방법: 기획이 급식별 다음 행동 가능량을 비교하고 AI가 쌀 소비→실패 검사→다음 날 회복 순서를 구현한다. [추정: 실제 sleep 순서에 따른 작업 분담]
- 완료 기준: 보유 쌀이 3을 넘으면 남는 쌀은 보존하고 다음 날 기력은 이전 잔량에 더하지 않고 회복값으로 교체한다. [근거: state.js sleep]
- 리스크/대안: 하루 종료 시 쌀 3을 자동 소비하므로 플레이어가 급식을 1로 제한해 비축하는 전략은 현재 입력으로 제공하지 않는다. [근거: state.js feedAmount]
- 근거: [근거: config.js; state.js]

노동/상점 — 같은 기력을 다른 자원으로 바꾼다

- 의도(Why): 현금과 직접 식량, 계절 한정 식량의 효율 차이로 그날의 조합을 고르게 한다. [근거: GAME_DESIGN.md §9/§10]
- 설계(How): 냥은 쌀 1당 3냥에 변환하며 각 작업은 하루 한 번만 가능하고, 짚신은 짚 1을 추가 소비하며 나물은 봄에만 허용한다. [근거: config.js; state.js]
- 필요(What): `WORKS`의 ID, 비용, 산출, 해금, 위치, 조건 필드와 `workDone` 일일 집합이 필요하다. [근거: config.js; state.js create]
- 작업 방법: AI가 작업 데이터를 공통 함수로 처리하고 기획이 표의 모든 작업을 실행 가능/불가 상태와 대조한다. [추정: 데이터별 누락 방지 절차]

- 완료 기준: 실패한 작업은 기력, 짚, 쌀, 냥을 변경하지 않으며 구매는 잔액 부족에서 실패한다. [근거: state.js doWork/buyRice]
- 리스크/대안: 장 거래를 모두 '매매'라고 쓰면 쌀 판매 기능을 오인하므로 현재는 쌀 구입과 당사실 판매만 명시한다. [근거: play.js showMarket; state.js]
- 근거: [근거: config.js WORKS; state.js]

문/당사실 — 회복 예외를 가진 소진 자원

- 의도(Why): 도움을 받을 능력과 미래 보은의 선택을 하나의 희소 재산에 연결한다. [근거: CONCEPT.md §2]
- 설계(How): 시작 염치 12에서 문 비용을 빼고 당사실 판매만 염치 4를 더하며, 당사실을 구조에 쓰면 이 회복과 15냥을 포기한다. [근거: config.js; state.js]
- 필요(What): 문별 기본값, 증가값, 일회성 플래그와 소지품 불리언, 사용 기록 두 종류가 필요하다. [근거: config.js DOORS; state.js]
- 작업 방법: 기획이 첫 사용, 반복, 잔량 부족/판매 후 상태를 비교하고 AI가 누적 사용과 일일 제한을 별도로 갱신한다. [추정: 회복 예외와 영구 소진을 혼동하지 않기 위한 절차]
- 완료 기준: 아전은 판매 후에도 재사용할 수 없고, 다른 문은 그 시점의 염치와 오늘 사용 여부로 판정한다. [근거: state.js doorStatus]
- 리스크/대안: 염치에는 최대값 제한이 없어 12에서 판매하면 16이 되므로 HUD의 최대값을 12로 고정한 설명을 쓰지 않는다. [근거: state.js sellThread; config.js THREAD_SELL_CHAE]
- 근거: [근거: config.js; state.js]

박/성과 곡선 — 생존을 남겨 둔 채 수확한다

- 의도(Why): 막판에 식구를 굶기고 박만 늘리는 결과를 최고 성과와 구분한다. [근거: S1 16:06~16:11; GAME_DESIGN.md §9]
- 설계(How): 심기 1, 톱질 순서별 1, 2, 3, 4 기력과 일일 한 통 제한을 사용하고 설계 목표로 4통/굶은 날 0/염치 잔존을 함께 본다. [근거: config.js; GAME_DESIGN.md §9]
- 필요(What): 박 개봉 횟수, 오늘 톱질 여부, 생존/염치 결산, 비용 배열이 필요하다. [근거: state.js create/tally]
- 작업 방법: 기획이 도달 가능 경로와 조기 소진 경로를 비교하고 AI가 착수/완료 사이의 중복 지급을 막는다. [추정: 기존 박 상태 구조에 대응한 검증 계획]
- 완료 기준: 박은 최대 4통이며 비용 부족/오늘 완료/전량 완료를 각각 구별한다. [근거: state.js sawStatus]
- 리스크/대안: 복합 최고 성과는 현재 별도 엔딩 등급이 아니라 설계 평가 기준이며, 박 속 재물은 실제 쌀/냥 경제로 유입되지 않는다. [근거: state.js finishSaw/tally; scenes.js Ending]
- 근거: [근거: config.js; state.js; GAME_DESIGN.md §9]

3. 데이터/리소스 명세

작업/문 수치표

항목	비용	산출/제한
멍석 맺기	기력 1	1냥, 1막부터. [근거: config.js WORKS]
가래질	기력 2	2냥, 1막부터. [근거: config.js WORKS]
방아 찧기	기력 3	쌀 1, 1막부터. [근거: config.js WORKS]
나물 캐기	기력 1	쌀 1, 1~4/12~13일만. [근거: config.js WORKS/isSpring]
이엉 엮기	기력 2	2냥, 2막부터. [근거: config.js WORKS]
무논 갈기	기력 3	4냥, 2막부터. [근거: config.js WORKS]
짚신 삼기	기력 1/짚 1	2냥, 2막부터. [근거: config.js; state.js]
김동지 문	염치 1부터 사용마다 +1	짚 3, 하루 한 번. [근거: config.js DOORS; state.js]
형님 문	염치 2부터 사용마다 +2	쌀 1, 하루 한 번. [근거: config.js DOORS; state.js]
아전 문	염치 3	5냥, 전체 한 번. [근거: config.js DOORS; state.js]
당사실 판매	당사실 보유 소모	15냥/염치 4, 전체 한 번. [근거: config.js; state.js]

데이터 필드 명세

필드명	타입	범위	예시/읽는 곳
`energy`	정수	정상 상태 0~6	3: 방아 가능 여부. [근거: config.js; state.js]
`money` / `rice` / `straw`	정수	0 이상, 별도 저장 상한 없음	쌀 2: 다음 기력 5. [근거: state.js]
`chae`	정수	정상 상태 0~16, 초기 12	1: 형님 첫 비용 2 미달. [근거: config.js; state.js]
`thread`	불리언	참/거짓	참: 판매/구조의 선행 조건. [근거: state.js]
`starved`	정수	정상 진행 0~3	3: 즉시 종료. [근거: state.js sleep]
`workDone` / `doorToday`	ID→불리언 객체	해당 ID 또는 빈 객체	`{kim:true}`: 오늘 김동지 사용 완료. [근거: state.js]
`doorUse`	ID→정수 객체	0 이상, 미등록은 0으로 읽음	`{kim:2}`: 다음 비용 3. [근거: state.js doorCost]
`sawed` / `sawedToday`	정수 / 불리언	0~4 / 참/거짓	2/참: 그날 세 번째 박 불가. [근거: state.js]
작업 `cost` / `money` / `rice`	정수	현 작업표 각각 1~3 / 0~4 / 0~1	방아: 3 / 0 / 1. [근거: config.js WORKS]
문 `base` / `step` / `once`	정수 / 정수 / 선택 불리언	1~3 / 0~2 / 참 또는 생략	아전: 3 / 0 / 참. [근거: config.js DOORS]

위 범위는 정상 게임 경로의 상태이며 외부 훅으로 임의 값을 주입하는 경우의 유효성 보장 범위가 아니다. [추정: 상태 함수의 정상 호출을 기준으로 정리한 명세] 필요 리소스는 수치 원본 `config.js`, 상태 처리 `state.js`, 독립 기대값과 검증 결과이며, 별도 스프레드시트 원본은 기록 없음으로 둔다. [근거: 최신본 파일 목록; QA_REPORT.md §5]

4. 작업 순서

급식과 노동의 하루 계산을 먼저 확정하고 문/판매의 예외를 연결한 뒤 제비/박의 장기 상태를 얹어 극단 전략을 비교한다. [추정: 단기 자원과 장기 분기의 의존 관계에 따른 순서] 수치 변경 시 문서의 기대값을 독립적으로 정리하고 구현 상수와 비교하며, 같은 소스에서 기대값을 읽어 자동으로 일치하게 만드는 검사는 피한다. [근거: GAME_DESIGN.md §9 독립 불변량 요구]

5. 검수 기준

모든 작업의 하루 한 번 제한, 문 비용 상승, 판매와 구조의 상호 배제, 급식 0~3의 회복, 박 하루 한 통을 개별 확인한다. [근거: config.js; state.js] 문 0, 1, 2회와 박 2, 3, 4통의 관계는 기존 연산의 가정 아래 검증한다. [근거: GAME_DESIGN.md §9] 시작 자원/당사실 판매 여부/굶은 날 제약을 결과와 함께 남긴다. [추정: 비교 조건 누락 방지를 위한 기록 방식] 기존 QA의 상수 30/30 일치와 64/64 검사는 과거 기록으로 인용하며 이번 문서 작성 중 재실행한 결과로 서술하지 않는다. [근거: QA_REPORT.md; 이번 조사 범위]

흥부전 (소설->게임) — 세부기획서: 콘텐츠와서사

> 재구성 기획서 (역설계) / 작성 기준 시점: 2026-08-04 (프로젝트 시작 시점) / 문서 버전: Ver 0.1 프로젝트 ID: P012 / 근거 수준: A / 작성: Codex (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

원작의 품팔이, 염치, 제비 구조/보은을 각각 반복 행동, 소진 압력, 되돌릴 수 없는 결정, 지연된 결과로 이어지게 한다. [근거: SOURCE_BIBLE.md; CONCEPT.md] 원작 줄거리를 모두 재연하기보다 플레이어가 한 행동의 결과를 다음 계절과 엔딩에서 확인하도록 한다. [근거: PRODUCT_BRIEF.md 비목표; GAME_DESIGN.md §17]

범위

7개 구역의 이동과 3막/16일 진행, 제비 사건, 박 4통, 엔딩 3종을 다루며, 전투, 던전, 보스, 놀부의 보수박 전체 플레이는 해당 없음으로 둔다. [근거: config.js; state.js; PRODUCT_BRIEF.md] 원문 인용은 추출 텍스트의 페이지 단위 U1~U12를 기준으로 하고, 원작 사실과 게임의 날짜/비용 각색을 구분한다. [근거: analysis/SOURCE_BIBLE.md; GAME_DESIGN.md]

2. 설계

1막 — 오늘의 끼니를 배우는 봄

- 의도(Why): 염치 선택을 제시하기 전에 노동, 쌀, 내일 기력의 연결부터 익히게 한다. [근거: GAME_DESIGN.md §12/§13]
- 설계(How): 1~4일을 봄으로 두고 기력 6/염치 12/쌀과 냥 0으로 시작하며, 멍석, 가래질, 방아, 나물을 일일 작업으로 제공하고 염치 HUD는 잠긴 상태로 둔다. [근거: config.js; state.js create; hud.js]
- 필요(What): 마당, 들, 동산 배경, 첫 아내 대사, 작업 비용 프롬프트, 잠자리와 첫 밤 정산이 필요하다. [근거: config.js ZONES/WORKS; BUILD_BRIEF.md]
- 작업 방법: 기획이 첫 행동 경로를 정리하고 AI가 시작 장면을 연결한 뒤 작업→급식→기력 회복을 한 번 수행해 대조한다. [추정: 첫 막의 학습 목표에 대응한 제작 절차]
- 완료 기준: 이용자가 식구를 먹여야 함과 굶은 날 누적 한도를 화면에서 설명할 수 있어야 한다. [근거: PRODUCT_BRIEF.md 교학; PLAYTEST_PROTOCOL.md]
- 리스크/대안: 설계의 첫 막 공간 표는 마당/들만 적지만 실제 나물은 동산이며 출구도 열려 있으므로 콘텐츠 표는 실제 위치를 따른다. [근거: GAME_DESIGN.md §4; config.js; play.js tryExit]
- 근거: [근거: GAME_DESIGN.md §12; config.js; hud.js]

2막 전반 — 나물이 사라지고 문을 연다

- 의도(Why): 첫 막에 배운 안전한 노동 조합을 바꾸어 염치를 사용할 이유를 만든다. [근거: GAME_DESIGN.md §12]
- 설계(How): 5일차에 제비집/염치 안내를 보여주고 봄 전용 나물을 막으며, 이엉, 무논, 짚신과 세 문의 행동을 해금한다. [근거: config.js; state.js sleep, workStatus, doorStatus]
- 필요(What): 여름 전환 문안, 문별 다음 비용, 짚 사용 작업, 제비집과 당사실 정보가 필요하다. [근거: GAME_DESIGN.md §13; BUILD_BRIEF.md]
- 작업 방법: AI가 날짜 변경 뒤 상태와 문구를 함께 갱신하고 기획이 4일차/5일차의 동일 대상 반응을 비교한다. [추정: 날짜 기반 해금의 경계 검수]
- 완료 기준: 4일차 문 행동은 잠기고 5일차부터 염치 비용을 지불하며, 나물은 사용할 수 없음을 표시한다. [근거: state.js; QA_REPORT.md §6]
- 리스크/대안: 구역 자체는 먼저 방문할 수 있으므로 '새 지역이 열린다'보다 '그곳의 새 행동이 가능해진다'는 구현 기준으로 설명한다. [근거: play.js tryExit; state.js doorStatus]
- 근거: [근거: config.js; state.js]

2막 후반 — 당사실의 사용처를 고른다

- 의도(Why): 하나뿐인 재산을 팔지 않고 구조에 쓰는 행위가 손해를 감수하는 결정으로 읽히게 한다. [근거: CONCEPT.md §2]
- 설계(How): 8일차에 새끼 제비를 떨어뜨리고 미결 상태를 유지하며, 구조하면 당사실을 소모해 saved, 지나치면 passed로 확정하고, 11일차 밤까지 미결이면 지나침으로 정리한다. [근거: config.js CHICK_DAY/SWALLOW_LEAVE_DAY; state.js]
- 필요(What): 새끼와 흰 실 상태, 아내의 실패, 구조/지나침 문안, 판매 대가 안내가 필요하다. [근거: ART_DIRECTION.md; play.js showChick]
- 작업 방법: 기획이 판매/구조의 기회비용을 대조하고 AI가 세 선택 경로를 구현한 뒤 제비 사건과 소지품/엔딩 상태를 검사한다. [추정: 명시 선택과 시간 경과 방치를 함께 검수하기 위한 절차]

- 완료 기준: 실을 판 경우 구조할 수 없고, 구조한 경우 판매할 수 없으며, 선택 이후 다시 결정할 수 없다. [근거: state.js sellThread, helpChick, leaveChick]
- 리스크/대안: 사람의 망설임은 기록상 미검증이므로 선택창 체류 시간, 발화, 소지품 확인을 시연에서 관찰한다. [근거: PLAYTEST_PROTOCOL.md]
- 근거: [근거: state.js; CONCEPT.md; PLAYTEST_PROTOCOL.md]

3막 — 돌아온 제비와 남겨 둔 문

- 의도(Why): 이전 계절의 선행과 염치 관리가 지금 가능한 행동으로 돌아오게 한다. [근거: CONCEPT.md P3; GAME_DESIGN.md §9]
- 설계(How): 12일차 구조 여부로 박씨 도착 또는 조기 종료를 나누고, 12~13일 나물을 다시 허용한 뒤 14~16일 추석 가을에는 막으며, 박 심기와 하루 한 통 톱질을 연결한다. [근거: config.js isSpring; state.js sleep, plant, sawStatus]
- 필요(What): 빈 처마/제비 귀환/동편 울타리의 박, 톱질 장면, 개봉 순서별 문안이 필요하다. [근거: BUILD_BRIEF.md; strings.ko.js]
- 작업 방법: AI가 구조 결과를 다음 계절에서 읽고 기획이 미구조 종료/구조 후 수확/가을 노동 선택을 각각 확인한다. [추정: 지연 인과의 상태 연결 검수]
- 완료 기준: 제비 미구조 경로는 박씨 없이 끝나며 구조 경로는 남은 기력을 사용해 박 개봉 수를 늘릴 수 있다. [근거: state.js]
- 리스크/대안: 실제 심기 직후 톱질이 가능하므로 추석 이전 개봉 금지는 현재 규칙에 포함하지 않고 원작/연출 의도와의 차이로 남긴다. [근거: source-text.txt U6; state.js sawStatus]
- 근거: [근거: config.js; state.js; GAME_DESIGN.md §12]

결산 — 무엇을 남겼는가

- 의도(Why): 재물 총액만이 아니라 생계와 관계의 결과를 함께 보여준다. [근거: GAME_DESIGN.md §14/§17]
- 설계(How): 켠 박/굶은 날/닫힌 문 이름/형님 집 방문/남은 염치를 표시하고 보은 엔딩의 후일담은 형님 방문 횟수를 읽어 문안을 바꾼다. [근거: state.js tally; scenes.js Ending]
- 필요(What): 세 엔딩의 제목, 본문, 다음 시도 조언, 후일담, 재시작 입력이 필요하다. [근거: scenes.js; strings.ko.js]
- 작업 방법: 기획이 결산 숫자와 문안의 원인을 대조하고 AI가 종료 시점의 상태를 전달한 뒤 새 판의 초기화를 검사한다. [추정: 누적 상태가 결산에 정확히 반영되도록 하는 절차]
- 완료 기준: 굶주림, 미구조, 보은 결과를 구분하고 다시 시작할 수 있다. [근거: GAME_DESIGN.md §16; QA_REPORT.md]
- 리스크/대안: 굶은 날짜 목록과 당시 열린 문은 상태에 저장하지 않으므로 구현에 없는 상세 회고를 제공한다고 쓰지 않는다. [근거: GAME_DESIGN.md §14; state.js create/tally]
- 근거: [근거: scenes.js; state.js]

3. 데이터/리소스 명세

실제 구역 연결

구역 ID	장소/연결	콘텐츠
`yard`	마당: 위 동산/왼쪽 김동지/오른쪽 들/아래 형님	멍석, 방아, 짚신, 잠자리, 제비, 박. [근거: config.js; play.js]
`hill`	동산: 아래 마당	봄 나물, 3막 박 덩굴 표현. [근거: config.js; play.js]
`field`	들: 왼쪽 마당/아래 장	가래질, 이엉, 무논. [근거: config.js]
`market`	장: 위 들	쌀 구입/당사실 판매. [근거: config.js; play.js]
`kim`	김동지 집: 오른쪽 마당	짚을 얻는 문. [근거: config.js]
`nolbu`	형님 집: 위 마당/아래 읍내	쌀을 얻는 문. [근거: config.js]
`town`	읍내 관청: 위 형님 집	매품 선금의 문. [근거: config.js]

사건 상태 명세

필드명	타입	범위/허용값	예시/용도
`day`	정수	1~16	8: 제비 추락일. [근거: config.js; state.js]
`nest` / `chickFallen`	불리언	참/거짓	5일차 둥지, 8일차 낙하. [근거: state.js sleep]
`chickResolved`	열거/널	`null`, `saved`, `passed`	`saved`: 다음 봄 박씨 도착. [근거: state.js]
`seedArrived` / `planted`	불리언	참/거짓	도착 참/심기 거짓이면 심기 허용 검사. [근거: state.js]
`gourdOpened`	정수 배열	정상 완료 시 0~3 인덱스	`[0,1]`: 앞의 두 박 개봉. [근거: state.js finishSaw]
`ending`	열거/널	`null`, `starved`, `passed`, `boeun`	`passed`: 12일차 미구조 종료. [근거: state.js]
`begCount`	정수	0 이상, 별도 상한 상수 없음	형님 문 사용 때 증가. [근거: state.js useDoor]

사건별 리소스는 둥지, 성체, 새끼, 실, 박, 톱, 막간 카드/결산 화면으로 제한하고, 원작의 모든 등장인물에게 독립 행동 시스템을 만들지 않는다. [근거: BUILD_BRIEF.md; GAME_DESIGN.md §17]

4. 작업 순서

원문 근거와 각색 경계를 정리한 뒤 날짜 전이, 구역, 상호작용을 연결하고, 제비 선택에서 엔딩까지의 지속 상태를 구현한 후 문안과 시각을 입힌다. [추정: 원작 인과와 상태 의존성을 우선하는 재구성 순서] 기획은 원작 사실/각색 이유/종료 문안을 검수하고 AI 구현은 날짜 경계와 장면 전환을 검사하며, 별도 제작 인원/담당자 배정표는 기록 없음으로 둔다. [추정: 기존 파이프라인 역할에 근거한 분담]

5. 검수 기준

4→5일, 7→8일, 11→12일, 13→14일, 16일 종료에서 사건이 한 번씩 발생하고 오늘 행동 제한이 정상 초기화되어야 한다. [근거: config.js; state.js sleep] 판매, 구조, 명시적 지나침/11일차까지 방치/누적 굶주림을

각각 검사하며, 같은 상태가 같은 결과를 내야 한다. [근거: state.js; GAME_DESIGN.md §7] 밤 정산과 계절 전환은 독립 프레임을 확보해야 하며 기존 기록의 누락을 완료로 바꾸지 않는다. [근거: QA_REPORT.md QA-04]