온리 업: 자오 — 세부기획서 (레벨디자인과 수치)

> 재구성 기획서 (역설계) / 작성 기준 시점: 2026-08-04 (프로젝트 시작 시점) / 문서 버전: Ver 0.1 프로젝트 ID: P020 / 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. 목적과 범위

500m 탑의 생성 규칙, 발판 패턴, 난이도 곡선과 물리 상수를 구현 가능한 수준으로 확정한다. UI, 아트, 사운드 리소스도 이 문서에서 목록으로 다룬다. 서사, 성장, 경제는 해당 없음(기록 없음).

2. 설계

진행 커서와 시드

- 의도(Why): 같은 탑을 반복해 코스를 외울 수 있게 한다.
- 설계(How): 시드 20260804의 mulberry32로 난수를 만든다. 커서(x, y, z, 각도)가 패턴마다 전진하며, 중심축에서 반경 26m를 넘으면 안쪽으로 최대 0.55 비율로 방향을 튼다. [근거: level.js `steer`, game.js `SEED`]
- 필요(What): 난수 유틸, 충돌체 배열.
- 작업 방법: 시작 광장(14×14) 생성 → 패턴 루프(고도 484m까지, 최대 900회) → 접근 계단 → 골인.
- 완료 기준: 같은 시드에서 항상 같은 배치.
- 리스크/대안: 시드 변경 시 도달 불가 발판 가능 → 시드 후보 검증.
- 근거: [근거: level.js]

난이도 구간

- 의도(Why): 초반은 학습, 후반은 정밀 점프를 요구한다.
- 설계(How): 고도 비율 t 기준 EASY(t<0.16, 약 80m) / MID(<0.45, 약 225m) / HARD(<0.78, 약 390m) / BRUTAL(그 이상). 각 구간의 패턴 집합은 아래 표. 정착지는 약 34m에서 시작해 이후 30~42m 간격. [근거: level.js]
- 필요(What): 구간별 패턴 풀.
- 작업 방법: 구간별 풀 → 무작위 선택.
- 완료 기준: 구간 경계에서 난이도가 단계적으로 오른다.
- 리스크/대안: 구간 길이 균형은 실제 플레이 검증 없음.
- 근거: [근거: level.js]

패턴 설계 원칙: 점프 성능에서 역산

- 의도(Why): 불가능한 점프가 나오면 게임이 파탄난다.
- 설계(How): 연쇄 발판의 상승량은 약 0.5~1.55m, 수평 간격은 2.9~4.6m(t 보간)로 한정한다. 트램폴린 도약은 스페이스 없이도 닿는 높이(최고점 5.6m)를 상한으로 해 상승 4.0~5.0m. 겹침이 생기는 구성

(트램폴린 착지 발판, 왕복, 엘리베이터, 궤도 발판, 골인)은 지배 축 한 방향으로 밀어내 AABB가 겹치지 않게 한다. 궤도 발판은 원 최근접 거리를 판정 대각 반경(약 3.96)보다 크게(여유 4.25) 잡는다. [근거: level.js 주석]

- 필요(What): 점프 상수(`game.js`)와 패턴 수치의 일치.
- 작업 방법: 패턴 구현 → 겹침/도달 확인 → `OU.debug.tp`로 구간 점검.
- 완료 기준: 모든 패턴에서 AABB 겹침 없이 도달 가능.
- 리스크/대안: 자동 검증기 없음.
- 근거: [근거: level.js]

골인 구간

- 의도(Why): 마지막은 반드시 도달 가능해야 한다.
- 설계(How): 정점(GOAL_Y=500)까지 상승량을 0.5~1.3m로 직접 통제하는 접근 계단, 골인 직전 발판 윗면을 499m에 맞춘다. 골인 발판은 6×6, 직전 발판과 수평 5.8m(실제 점프 거리 1.1m)로 겹침 없이 배치한다. 정점에 12개의 왕관 기둥(올라오는 쪽은 비움)과 빛기둥을 둔다. [근거: level.js]
- 필요(What): 기록 없음.
- 작업 방법: 골인 접촉 판정 → 3.2초 연출 → 클리어 카드.
- 완료 기준: 클리어 화면에 시간, 추락, 고도가 표시된다.
- 리스크/대안: 이전에는 골인 발판이 직전 발판을 덮어 막혔던 버그가 있었다. [근거: level.js 주석]
- 근거: [근거: level.js, game.js]

UI 화면 구성

- 의도(Why): 고도 중심 정보 설계.
- 설계(How): 타이틀(규칙 4칸 안내, 시작/어시스트 버튼), HUD(고도 게이지, 패널, 키 안내, 토스트, 비네트), 일시정지(현재, 최고, 추락, 계속, 어시스트, 처음부터), 클리어(시간, 추락, 고도). 키: WASD, Shift, Space, 마우스, 휠, ESC, R, M, T. [근거: index.html, game.js]
- 필요(What): HTML 오버레이.
- 작업 방법: 상태에 따른 표시 전환.
- 완료 기준: 포커스 상실 시 일시정지.
- 리스크/대안: 모바일 미지원.
- 근거: [근거: index.html, game.js]

3. 데이터/리소스 명세

패턴 풀 | 구간 | 패턴 | |---|---| | EASY | 발판 연쇄, 나선 계단, 외줄 다리 | | MID | + 기둥, 트램폴린, 왕복 발판, 지그재그, 엘리베이터 | | HARD | 기둥, 지그재그, 궤도, 상승 기류, 왕복, 트램폴린, 미끄럼, 엘리베이터, 발판 연쇄 | | BRUTAL | 기둥, 지그재그, 궤도, 미끄럼, 상승 기류 | [근거: level.js]

충돌체 필드: kind(solid/bounce/goal/slip), cx, cy, cz, hx, hy, hz, min/max 6개, mover(type lin/orb, 진폭, 속도, 위상), dx, dy, dz, vx, vy, vz. [근거: level.js pad] **정착지 필드**: x, y, z, active, ring, idx. [근거: level.js] **저장**

키: `ouz_best`(실수), `ouz_assist`('1'/'0'). [근거: game.js] **리소스**: 외부 파일 없음. 합성 효과음 7종 이상, 프리미티브 캐릭터, 파티클(먼지 1400, 상승 기류 60, 폭죽 240). BGM 기록 없음. [근거: util.js, game.js]

4. 작업 순서

1. 상수/물리 → 2. 패턴별 구현과 겹침 방지 → 3. 구간 튜닝 → 4. 정점 접근/골인 → 5. HUD/연출 → 6. 단일 빌드. [추정: 코드 구조]

5. 검수 기준

- 시드 고정 탑이 새로고침마다 동일하다.
- `OU.debug.tp`로 각 구간에서 시작해 패턴 전부 통과 가능하다(자동 검증은 미구현).
- 어시스트 ON/OFF 복귀 위치가 규칙대로 동작한다.
- 분리형 7파일과 단일 HTML의 동작이 같다.
- 미해결: 타이틀 문구와 어시스트 OFF 규칙의 불일치, 완주 가능성 미확인.