개인 프로젝트

BuildTest
세부 기획서

분석대상-규칙/데이터명세 / 값의 출처와 검수

BuildTest

문서 최초 작성	문서 최종 수정
20260718 20:11	20260718 20:25
박도원	박도원

목차

개요 – 2P

1. 목차
2. 로그
3. 문서 계층

현황 – 5P

1. 세부 착수 조건

목표 – 6P

1. 데이터 명세 인수 기준

핵심 요소 – 7P

1. 명세 핵심 원칙

본문 – 8P

1. 주행/추격 수치 등록안
2. 피격/사운드 수치 등록안
3. 씬 미확인 항목
4. 대상 클래스 역할 대응
5. 이벤트 계약과 검수 항목

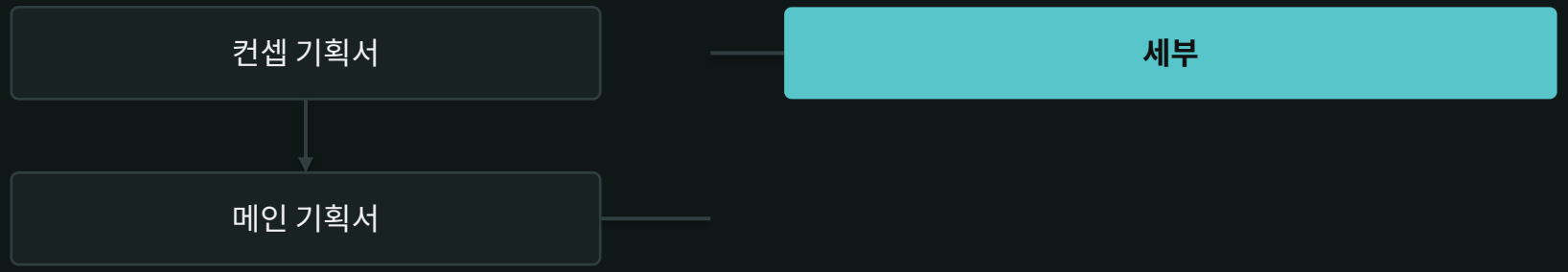
작업 계획 – 13P

1. 명세 제작 작업
2. 검증/인계 방안

개요 – 로그

순번	작성일	작성자	작성 내용	비고
1	2026-07-18 20:11	박도원	데이터 명세 착수	수치 후보와 보류 목록을 인계함
2	2026-07-18 20:18	박도원	출처별 값 분류	상수/코드 기본값/씬 미확인 항목을 나눔
3	2026-07-18 20:25	박도원	규칙 명세 인계	명세와 보류 목록을 정리하고 실행 항목은 미실행으로 남김
4				
5				
6				

개요 – 문서 계층



컨셉 → 메인 → 세부 순으로 상위 문서의 결정을 목표로 받아 구체화함.
강조 표시 = 이 문서

현황 – 세부 착수 조건

입력 소스

분석 대상 C# 17개와 핵심 관계를 입력으로 삼음

의도 전수 분류의 기준을 고정함

근거 생성 코드/어셈블리 메타는 별도임

미확인 설정

토크/적 속도/아이템 총수는 보류 상태임

의도 실제 밸런스를 단정하지 않음

근거 씬 직렬화값을 읽지 않음

검증 공백

실행 체크리스트와 카메라 설정은 추가 확인 대상임

의도 정적 분석과 관찰을 나눔

근거 이전 단계는 소스 관계 분석임

목표 – 데이터 명세 인수 기준



모든 수치에 출처, 단위, 확정 범위를 붙일 예정임

3분류	8항목	17개
상수 / 기본값 / 씬 미확인	실행 확인 체크리스트	분석 대상 역할 대응

의도 기본값을 실제 플레이 값으로 오인하지 않게 함

근거 직렬화 초기값은 씬에서 덮어쓸 수 있음

핵심 요소 – 명세 핵심 원칙

출처 보존

클래스, 필드, 함수와 값을 함께 기록함

의도 후속 수정 시 근거를 재확인함

근거 추출 소스가 기준 입력임

주체별 시간

플레이어/접촉 적/유령 시간을 나눔

의도 보호 범위를 정확히 설명함

근거 EnemyChaser와 PlayerHP는 별도 상태임

확장 분리

속도 연출/차량 해금은 제안으로 관리함

의도 현재 기능과 구별함

근거 세션의 가정형 확장 요청임

본문 – 주행/추격 수치 등록안

값과 단위를 함께 등록할 예정임 이전 단계의 차량/추격 소스가 근거임

출처	값 / 단위	분류
CarController.ghostTime	3초	코드 기본값
CarController 감쇠 / 환산	0.993배 / 3.6배	코드 상수 / 물리 프레임 / km/h
EnemyChaser.predictionTime	0.5초	코드 기본값
EnemyChaser.updateInterval	0.1초	코드 기본값
rotationSpeed / cornerSlowDown	360도/초 / 0.5	코드 기본값
NavMesh 이탈 탐색	반경 5 Unity 단위	코드 상수

본문 – 피격/사운드 수치 등록안

보호 시간과 음향을 분리 등록할 예정임 필드 초기값과 함수 리터럴은 다른 출처임

출처	값	적용 범위
PlayerHP.maxHP / invincibleTime	100 HP / 1초	코드 기본값 / 플레이어
TakeDamage / PauseChase	10 HP / 3.5초	코드 상수 / 충돌한 적의 정지
minPitch / maxPitch	0.8 / 2	코드 기본값 / 엔진 피치
엔진음 / 브레이크음 조건	속도 > 0.5 / > 1 m/s	상수 / 브레이크는 Space 병행
sirenDistance / maxSirenVolume	50 / 1	코드 기본값 / 거리 / 볼륨
crashVolume / 피치 포화 기준	1 / 30 m/s	기본값 / 상수

본문 – 씬 미확인 항목

미확인 값을 임의로 채우지 않을 예정임 씬 전용 설정이 존재함

항목	미확인 범위	확인 방법
차량	motorForce / breakForce / maxSteerAngle	직렬화 필드를 읽어야 함
배치	spawnPoints / enemyCars / totalItemCount	배열 참조와 배치를 대조함
추격	NavMeshAgent.speed / acceleration	컴포넌트 필드를 읽어야 함
UI/카메라	pages 수 / Cinemachine 설정	씬 참조와 화면을 대조함
도보	rotationLimit / 모듈 사용 씬	배치 확인 전까지 보류함

본문 – 대상 클래스 역할 대응

17개 대상의 대응을 만들 예정임 생성 코드와 Properties는 별도 집계임

분류	대상 클래스	등록 규칙
주행/위협	CarController / EnemyChaser / PlayerHP	입력, 상태, 예외를 연결함
진행	MainMenu / TutorialManager / GameManager	씬 전환과 종료를 기록함
수집	Item / ItemCollector / Goal	카운터, 골, 승리 조건을 기록함
표시	UIManager / Speedometer	HP/속도 출력을 기록함
주변 모듈	PlayerMovement / DoorController / KeyController	실사용 미확인으로 등록함
보조/템플릿	ObjectRotate / ObjectRotate2 / Readme	회전 20도/초 기본값과 템플릿을 구분함

본문 – 이벤트 계약과 검수 항목

통과, 실패, 미실행으로 기록할 예정임 현재 소스 해석은 실행 검증 결과가 아님

검수	조건 → 기대 결과	확인 범위
① 씬 진행	메뉴 → TutorialScene → RaceScene	파일 인덱스 대응 미확인
② 조작	축 입력/Space → 구동, 조향, 제동	키 매핑 실행 확인 대상
③ 리스폰	R → 최근접 스폰 / 속도 0 / 유령	씬 기본값 변경 확인 대상
④ 피격	접촉 → 피해 10 / 해당 적 정지	무적 중 피해 제외 확인 대상
⑤ 수집	수집 수 ≥ 총수 → 골 활성	총수 씬 확인 대상
⑥ 클리어	수집 완료 후 골 → 패널/시간 정지	실행 결과 기록 대상
⑦ 실패	HP ≤ 0 → 패널/시간 정지	실행 결과 기록 대상
⑧ 재시작	timeScale 1 / 씬 재로드	상태 초기화 확인 대상

작업 계획 – 명세 제작 작업

1

값 등록

상수, 초기값, 미확인 필드를 전수 분류함

의도 출처 없는 수치를 방지함

근거 17개 대상 소스를 기준으로 함

2

참조 대응

level 파일과 씬 이름/컴포넌트를 대응함

의도 값의 사용처를 확인함

근거 씬 이름과 인덱스는 별개임

3

검수 기록

8항목 결과와 주변 모듈 사용 여부를 남김

의도 검증 공백을 인계함

근거 도보 모듈 사용은 미확인임

작업 계획 – 검증/인계 방안

1

소스 재대조

함수 리터럴과 초기값을 원문에 대조함

C# 소스 / 검색

의도 분류 오류를 줄임

근거 기본값은 실효값이 아님

2

씬 확인

필드 추출 가능성과 불가 사유를 기록함

씬 추출 도구 후보

의도 실효값을 확보함

근거 빌드 설정과 씬 연결이 필요함

3

실행/인계

관찰 결과 또는 미실행 상태를 명세에 기록함

Windows exe / 체크리스트

의도 추정을 검증으로 오인하지 않게 함

근거 실행은 별도 증거가 필요함