

플레이북 (Playbook)

비개발자가 AI를 지휘해 기획부터 출시까지 가는 법 — 제가 4개 제품을 혼자 출시하며 실제로 쓴 방법

이건 강의가 아니라 **실전 기록**입니다. 제가 THE HOUSE·MECHOOSE·Momento를 출시하고 해커톤에서 Money Navigation을 완주하며 실제로 지킨 절차를 정리했습니다. 공개용으로 뼈대와 체크리스트만 담았고, 세부 조건은 면접·협업 시 공유합니다.

1. 시작 전 체크리스트

- 문제를 한 문장으로 정의했는가 (누가, 어떤 불편을)
- 폴더·파일 구조와 "이렇게 동작해야 한다"는 기준을 문서로 적었는가
- 스택·플랫폼 정책은 추측이 아니라 공식 문서로 확인했는가

2. 규칙 문서 세우기

- 일하는 방식 / 금지사항 / 검증 기준을 한 파일에
- 매 세션 시작: "먼저 읽고 현재 단계를 말해" 로 동기화
- 핵심 규칙 예: 가짜(하드코딩) 금지 · 세 지점에서 테스트 · 혼자 아는 걸로 단정 금지

3. 선(先)설계: 화면·데이터 먼저

- 디자인/UI와 데이터 구조를 먼저 다 뽑은 뒤 코드에 착수 (병행하면 꼬임)

4. 지시 → 확인 → 교정 루프

- 화면 하나·기능 하나 단위로 작게 지시
- "다 됐다"를 믿지 말고 폰·시뮬레이터로 직접 확인
- 어긋나면 원인을 짚어 다시 지시하거나 재작성 요구

5. 검증 체크리스트

- 빌드 전 드라이런 / 빌드 후 정적분석·시뮬레이터·스크린샷 비교 / 큰 파트마다 전체 데모 재검
- 그럴듯한 성공은 **입력을 바꿔 재현**시켜 확인
- 비밀값은 환경변수·시크릿에만, 화면 캡처엔 더미 데이터

6. AI가 자주 틀리는 패턴과 대처

- 데모용 하드코딩 → 다른 입력으로 재현
- 생성 코드 반복 오류 → 설계·데이터 모델만 남기고 코드 재작성

- 버전 비호환(DI 등) → 억지로 끝지 말고 대안 먼저
 - 자기보고와 실제 불일치 → 실제 코드·결과물로 재검증
-

7. 막혔을 때

- 권한 밖(조직 정책)·불안정한 경로는 과감히 보류
 - 되는 방식(무료·온디바이스·다른 채널)으로 전환해 하나라도 동작하게
-

8. 멀티세션(1인 AI 팀) 운영

- 세션이 여러 개면 한 세션을 PM으로 세워 순서·보고를 집중
 - 파일 소유권·커밋 경계·빌드 락 규칙으로 충돌 방지
 - 컨텍스트가 바뀌어도 맥락이 끊기지 않게 상태를 문서로
-

9. 출시 체크리스트 (플랫폼별)

- 공통: 비밀값 분리, 개인정보 노출 점검, 링크 전수 확인
 - iOS: TestFlight / App Store Connect 세금·계좌, 시뮬레이터 캡처
 - Android: 서명 keystore 백업, 내부 테스트 트랙
 - 웹/엣지: 보안 헤더, 배포 자동화 스크립트
 - 게임(Unity): 에셋 분할 배포, 빌드 자동화, 언어별 스토어 메타
-

10. 사례 색인

- THE HOUSE(게임) · MECHOOSE(커머스) · Momento(네이티브 2종) · AttenNFC(설계·보안) · Money Navigation(해커톤·멀티세션)
 - 각 사례의 실제 지시 발췌와 트러블슈팅은 프로젝트 케이스스터디 참조.
-