

기획 설계 문서가 준비되었습니다

아이디어가 구체적인 실행 계획으로 변환되었습니다.

[PRD \(요구사항\)](#)[TRD \(기술\)](#)[User Flow](#)[DB 설계](#)[디자인 시스템](#)[개발 태스크](#)[가이드](#)

7. Coding Convention & AI 협업 가이드

7.1 코드 작성 원칙

- **가독성 우선**: 변수명은 `depositAmount`, `taxRate`와 같이 의미를 명확히 알 수 있는 영문 풀네임을 사용합니다. (축약어 지양)
- **함수형 프로그래밍**: 상태 변경을 최소화하고, 입력값에 대해 항상 동일한 출력을 보장하는 순수 함수로 계산 로직을 분리합니다.
- **상수 관리**: 세율(15.4%), 기본 기간 등 매직 넘버는 별도의 상수 파일(`constants.js`)로 관리 합니다.

7.2 AI 프롬프트 가이드

- AI에게 코드 생성을 요청할 때는 "비즈니스 로직"과 "UI 컴포넌트"를 분리해서 요청하세요.
- **예시**: "예금과 적금의 세후 이자를 계산하는 순수 함수를 작성해줘. 입력값은 원금, 기간, 금리, 타입이고 결과값은 객체로 반환해."
- 코드 내에 주석은 복잡한 금융 계산 공식에 대한 설명 위주로 작성하도록 요청합니다.

7.3 디렉토리 구조 권장안

- `/components`: UI 컴포넌트 (Grid, Card, Input)
- `/utils`: 계산 로직 (financeCalculator.js, formatter.js)
- `/hooks`: 상태 관리 및 데이터 로직 (useCalculator.js)
- `/assets`: 스타일 및 아이콘

[PDF로 저장하기](#)[새로운 기획 시작](#)