**Selltrat 모노레포 구조**

selltrat/

├── apps/

│ ├── web/ # Next.js 14 프론트엔드

│ │ ├── app/ # App Router

│ │ │ ├── (auth)/ # 인증 관련 페이지

│ │ │ │ ├── login/

│ │ │ │ ├── signup/

│ │ │ │ └── forgot-password/

│ │ │ ├── (dashboard)/ # 대시보드 페이지

│ │ │ │ ├── dashboard/

│ │ │ │ ├── chat/

│ │ │ │ ├── reports/

│ │ │ │ ├── tools/

│ │ │ │ └── settings/

│ │ │ ├── (landing)/ # 랜딩 페이지

│ │ │ │ ├── page.tsx

│ │ │ │ ├── pricing/

│ │ │ │ ├── features/

│ │ │ │ └── success-stories/

│ │ │ ├── api/ # API Routes

│ │ │ │ ├── auth/

│ │ │ │ ├── webhook/

│ │ │ │ └── upload/

│ │ │ ├── layout.tsx

│ │ │ └── globals.css

│ │ ├── components/

│ │ │ ├── ui/ # shadcn/ui 컴포넌트

│ │ │ ├── chat/ # 채팅 관련 컴포넌트

│ │ │ ├── dashboard/ # 대시보드 컴포넌트

│ │ │ ├── landing/ # 랜딩 페이지 컴포넌트

│ │ │ └── shared/ # 공통 컴포넌트

│ │ ├── hooks/ # Custom Hooks

│ │ ├── lib/ # 유틸리티 함수

│ │ ├── stores/ # Zustand 스토어

│ │ ├── types/ # TypeScript 타입

│ │ └── public/ # 정적 파일

│ │

│ ├── api/ # NestJS 백엔드

│ │ ├── src/

│ │ │ ├── modules/

│ │ │ │ ├── auth/ # 인증 모듈

│ │ │ │ ├── users/ # 사용자 모듈

│ │ │ │ ├── agents/ # 에이전트 모듈

│ │ │ │ ├── chat/ # 채팅 모듈

│ │ │ │ ├── reports/ # 보고서 모듈

│ │ │ │ ├── payments/ # 결제 모듈

│ │ │ │ └── analytics/ # 분석 모듈

│ │ │ ├── common/ # 공통 모듈

│ │ │ │ ├── decorators/

│ │ │ │ ├── filters/

│ │ │ │ ├── guards/

│ │ │ │ ├── interceptors/

│ │ │ │ └── pipes/

│ │ │ ├── config/ # 설정 파일

│ │ │ ├── database/ # DB 관련

│ │ │ │ ├── entities/

│ │ │ │ ├── migrations/

│ │ │ │ └── seeds/

│ │ │ ├── queue/ # BullMQ 큐

│ │ │ └── main.ts

│ │ └── test/ # 테스트 파일

│ │

│ └── agent-runtime/ # 에이전트 실행 엔진

│ ├── src/

│ │ ├── agents/ # 개별 에이전트

│ │ │ ├── market/

│ │ │ ├── legal/

│ │ │ ├── finance/

│ │ │ ├── marketing/

│ │ │ └── operation/

│ │ ├── orchestrator/ # LangGraph 오케스트레이터

│ │ ├── tools/ # 에이전트 도구

│ │ ├── prompts/ # 프롬프트 템플릿

│ │ └── workflows/ # 워크플로우 정의

│ └── test/

│

├── packages/ # 공유 패키지

│ ├── shared/

│ │ ├── types/ # 공통 타입 정의

│ │ │ ├── user.types.ts

│ │ │ ├── agent.types.ts

│ │ │ ├── chat.types.ts

│ │ │ └── index.ts

│ │ ├── utils/ # 유틸리티 함수

│ │ │ ├── date.utils.ts

│ │ │ ├── format.utils.ts

│ │ │ └── validation.utils.ts

│ │ └── config/ # 공통 설정

│ │ ├── constants.ts

│ │ └── env.ts

│ │

│ ├── ui/ # 공통 UI 컴포넌트

│ │ ├── components/

│ │ ├── styles/

│ │ └── hooks/

│ │

│ └── agents/ # 에이전트 코어

│ ├── core/ # 코어 엔진

│ ├── llm/ # LLM 통합

│ └── tools/ # 에이전트 도구

│

├── infrastructure/ # 인프라 설정

│ ├── docker/

│ │ ├── Dockerfile.web

│ │ ├── Dockerfile.api

│ │ └── docker-compose.yml

│ ├── k8s/ # Kubernetes 설정

│ └── terraform/ # Terraform 설정

│

├── scripts/ # 스크립트

│ ├── setup.sh

│ ├── deploy.sh

│ └── seed-db.ts

│

├── docs/ # 문서

│ ├── API.md

│ ├── ARCHITECTURE.md

│ └── CONTRIBUTING.md

│

├── .github/ # GitHub 설정

│ ├── workflows/

│ │ ├── ci.yml

│ │ ├── cd.yml

│ │ └── test.yml

│ └── PULL\_REQUEST\_TEMPLATE.md

│

├── turbo.json # Turborepo 설정

├── package.json # 루트 package.json

├── pnpm-workspace.yaml # PNPM 워크스페이스

├── .env.example # 환경변수 예제

├── .gitignore

├── README.md

└── LICENSE

**주요 파일 설명**

**1. apps/web (프론트엔드)**

* **app/**: Next.js 14 App Router 구조
* **components/**: React 컴포넌트 (Atomic Design 패턴)
* **stores/**: Zustand 상태 관리
* **hooks/**: 커스텀 React 훅

**2. apps/api (백엔드)**

* **modules/**: NestJS 모듈 (DDD 구조)
* **common/**: 공통 기능 (Guards, Filters, Interceptors)
* **database/**: TypeORM 엔티티 및 마이그레이션

**3. apps/agent-runtime (AI 엔진)**

* **agents/**: 전문 에이전트 구현
* **orchestrator/**: LangGraph 워크플로우
* **tools/**: 에이전트가 사용하는 도구

**4. packages (공유 코드)**

* **shared/**: 타입, 유틸리티, 설정
* **ui/**: 공통 UI 컴포넌트
* **agents/**: 에이전트 코어 로직