Crypto Portfolio Tracker with Risk & Scam Analysis

Project Statement:

The project aggregates crypto holdings across exchanges, tracks P&L, and flags risky/scam tokens.

Key Features:

● Multi-exchange aggregation (API keys)

● Real-time pricing & historical charts

● Scam risk alerts & watchlists

● P&L and tax-ready summaries

● Notifications for price/risk thresholds

Tech Stack:

- Frontend: React.js + Tailwind CSS
- Backend: Spring Boot (Java)
- Database: MySQL
- Authentication: JWT (access + refresh tokens)
- API Integrations: CoinGecko, Binance, Etherscan, CryptoScamDB

Modules:

● Module A: Auth & Exchange Connections

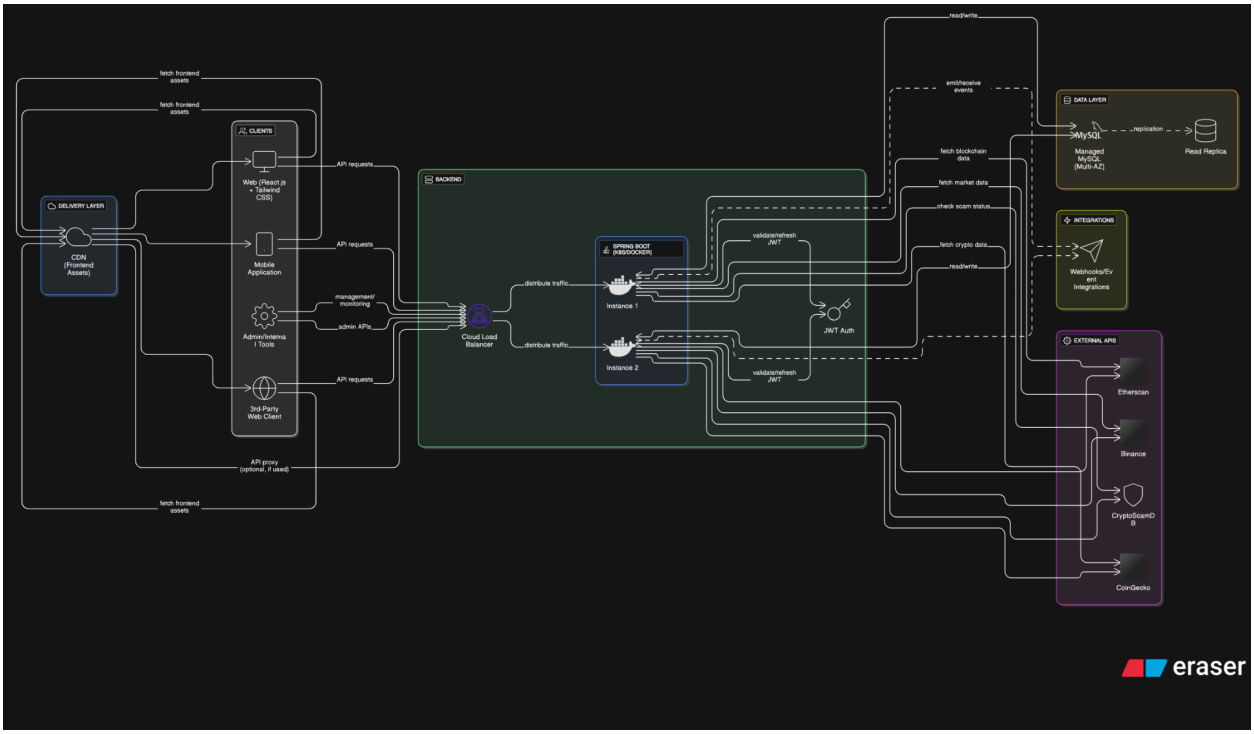
● Module B: Portfolio & Trades

● Module C: Pricing, Charts & Snapshots

● Module D: Risk/Scam Detection & Alerts

● Module E: P&L & Tax Reports

Architecture Diagram:



8-Week Milestone Plan

Milestone 1: Weeks 1–2 – Auth & Connections

Week 1: Auth scaffold, Users/Exchanges/ApiKeys schema

Week 2: Add exchange connectors (Binance first), key encryption at rest

Expected Output:

User can connect an exchange with stored credentials securely.

Milestone 2: Weeks 3–4 – Portfolio & Trades

Week 3: Fetch balances → Holdings; manual add/edit

Week 4: Sync recent trades → Trades; compute cost basis

Expected Output:

Accurate holdings + trade history in dashboards.

Milestone 3: Weeks 5–6 – Pricing & Risk

Week 5: CoinGecko pricing; PriceSnapshots cron; charts

Week 6: Risk alerts (contract reputation via Etherscan/CryptoScamDB), notifications

Expected Output:

Live pricing with historical charts & scam warnings.

Milestone 4: Weeks 7–8 – P&L & Reports

Week 7: P&L summaries, realized/unrealized gains, export CSV

Week 8: Tax hints, QA, security checks, deployment

Expected Output:

P&L dashboards, exportable tax-ready data, production build.

Expected Project Outcome:

By Week 8, users have consolidated portfolios, automated pricing, risk alerts, and exportable P&L reports.

Database Schema:

- **Users:** id (INT, PK), name (VARCHAR), email (VARCHAR, UNIQUE), password (VARCHAR), created_at (TIMESTAMP)
- **Exchanges:** id (INT, PK), name (VARCHAR), base_url (VARCHAR), created_at (TIMESTAMP)
- **ApiKeys:** id (INT, PK), user_id (FK to Users.id), exchange_id (FK to Exchanges.id), api_key (VARCHAR), api_secret (VARCHAR), label (VARCHAR), created_at (TIMESTAMP)
- **Holdings:** id (INT, PK), user_id (FK to Users.id), asset_symbol (VARCHAR), quantity (NUMERIC), avg_cost (NUMERIC), wallet_type (ENUM: 'exchange', 'wallet'), exchange_id (INT, NULL, FK to Exchanges.id), address (VARCHAR, NULL), updated_at (TIMESTAMP)
- **Trades:** id (INT, PK), user_id (FK to Users.id), asset_symbol (VARCHAR), side (ENUM: 'buy', 'sell'), quantity (NUMERIC), price (NUMERIC), fee (NUMERIC), exchange_id (INT, FK to Exchanges.id), executed_at (TIMESTAMP)
- **PriceSnapshots:** id (INT, PK), asset_symbol (VARCHAR), price_usd (NUMERIC), market_cap (NUMERIC), source (VARCHAR), captured_at (TIMESTAMP)
- **RiskAlerts:** id (INT, PK), user_id (FK to Users.id), asset_symbol (VARCHAR), alert_type (ENUM: 'rugpull_warning', 'contract_risk', 'news'), details (TEXT), created_at (TIMESTAMP)
- **ScamTokens:** id (INT, PK), contract_address (VARCHAR), chain (VARCHAR), risk_level (ENUM: 'low', 'medium', 'high'), source (VARCHAR), last_seen (TIMESTAMP)