

# Eric Johnson

[eric.johnson.dev123@gmail.com](mailto:eric.johnson.dev123@gmail.com)

<https://Eric-Johnson-1.github.io/Portfolio>

<https://github.com/Eric-Johnson-1>

---

## SUMMARY

Web3 Full-Stack Software Engineer who has **architected 30+ Solidity smart contracts** and **shipped 4+ production-grade dApps** across DeFi, NFT, DAO governance, and healthcare. Backed by strictly enforced security standards, establishing **100% on-chain core mechanics verification** via comprehensive **Foundry** test suites (**650+ invariant and fuzz tests**). 3+ years of remote Web3 experience spanning **EIP-7201 UUPS-upgradeable contract systems**, **ERC-4337 Account Abstraction** with gasless onboarding, and **ERC-5192 Soulbound Token** architectures, plus 2 years of Web2 full-stack development. Deep expertise in building **censorship-resistant systems** (Merkle proof + IPFS anti-censorship), **on-chain governance** (Governor + Timelock + quadratic funding), and **gas-optimized** storage layouts. Seeking to bring end-to-end blockchain engineering depth to a team building impactful, **user-centric** decentralized infrastructure.

## TECHNICAL SKILLS

**Blockchain & Smart Contracts:** Solidity, Foundry, OpenZeppelin, UUPS Proxy (EIP-7201 Namespaced Storage), ERC-4337 (Account Abstraction — Smart Account + Paymaster + Factory), ERC-5192 (Soulbound Tokens), ERC-7821 (Batched Execution), ERC-1271 (On-chain Signature Validation), ERC-721A, EIP-712 (Typed Data Signatures), ERC-1155, ERC-20, Create2 Deterministic Deployment, Merkle Proof Verification, Quadratic Funding (CLR), DAO Governance, Gas Optimization, NFT Marketplace Architecture

**Frontend & Wallet Integration:** TypeScript, Next.js (App Router), React, Wagmi, Viem, RainbowKit, ConnectKit, Tailwind CSS, Radix UI, Framer Motion, Zustand, TanStack Query

**Backend & Infrastructure:** Bun, Hono, Node.js, Express.js, Drizzle ORM, PostgreSQL, MongoDB, The Graph (Subgraph), IPFS, Upstash Redis, Docker, Zod (Schema Validation)

**Tools & DevOps:** Turborepo (Monorepo), Git, GitHub Actions (CI/CD), Satori + Sharp (Dynamic SVG → PNG/WebP Rendering)

## EXPERIENCE

### Web3 Full-Stack Software Engineer | Remote - Bianjie.AI

2023.02 - present

*Core developer on a production enterprise-grade IDO (Initial DEX Offering) platform with staking-based allocation, LP yield farming, and on-chain token vesting. Turborepo monorepo across 4 workspaces (web, api, contracts, subgraph) running on Bun.*

- Architected 5 UUPS-upgradeable Solidity smart contracts (SalesFactory, Sale, AllocationStaking, FarmingNBL, Vesting) with **EIP-7201** namespaced storage layouts, ReentrancyGuard, and Pausable emergency circuit-breakers — enabling upgrade-safe infrastructure with **zero storage collision risk**.
- Engineered a Create2 deterministic clone deployment system via `Clones.cloneDeterministic`, enabling **predictable Sale contract addresses before deployment** and **gas-efficient factory-pattern** IDO launches.
- Designed a Merkle proof + IPFS anti-censorship whitelist architecture: backend generates Merkle trees, uploads complete whitelist JSON to IPFS, and stores **roots on-chain**. Frontend implements TanStack Query **dual-source fallback** (API → IPFS + on-chain root), ensuring **100% trustless participation** even if the centralized backend goes offline.
- Built a MasterChef V2-style LP farming contract (`FarmingNBL`) with multi-pool support, time-weighted reward accumulation, external `IRewards` plugin interface, duplicate LP token detection, and **balance-capped safe distribution**.
- Implemented per-user Vesting contract deployment on token claim: Sale contract dynamically deploys individual `Vesting` contracts with configurable linear release schedules, SafeERC20 transfers, and **owner-recoverable `recoverERC20`** for accidentally sent tokens.
- Developed the **Hono + Drizzle ORM backend** (TypeScript/Bun) with RESTful project CRUD, IPFS upload pipeline, Zod schema validation, and PostgreSQL persistence with `onConflictDoUpdate` upsert resolution for atomic snapshot updates.
- Implemented on-chain safety invariant checks: Sale contract verifies `IERC20.balanceOf(address(this)) >= tokensSold + tokensToBuy` (C5), and `withdrawLeftoverTokens` uses \*\*`min(unsold, balance)`\*\* safe withdrawal pattern\*\* (H2).
- Deployed a **Graph Protocol subgraph** indexing 4 entity types with `@derivedFrom` reverse-lookup relationships — replacing expensive on-chain enumeration with real-time event-indexed queries.
- Built the **Next.js frontend** with Wagmi / Viem for **type-safe** contract interactions, reactive on-chain state updates, and managed Docker Compose for PostgreSQL and environment-driven chain configuration.

#### Web3 Full-Stack Software Engineer | Remote - TheOne.art

2022.02 - 2023.02

- Engineered a high-performance NFT marketplace supporting primary issuance and secondary trading, leveraging **ERC721A** to achieve **gas-optimized** batch minting operations.
- Architected a decentralized order book using **EIP-712** typed data signatures, enabling gasless off-chain maker order placements with **atomic on-chain taker matching**.
- Developed comprehensive **Subgraph (The Graph)** indexing for complex marketplace events (listings, bids, sales), **reducing** frontend query **latency** by removing on-chain RPC bottlenecks.
- Designed an automated **hybrid storage pipeline** utilizing **IPFS** for decentralized metadata pinning combined with a robust **Node.js/Express.js** backend.

- Implemented diverse wallet standard support with Ethers.js, driving **strict security patterns** including `ReentrancyGuard` and **pull-over-push** fund distribution mechanisms.