

Manav Darji

manav.d.2401@gmail.com | (+91) 9825864164 | [Website](#) | github.com/manav2401 | [LinkedIn](#) | [X/Twitter](#) | Ahmedabad, India

Summary: Blockchain protocol engineer with 5+ years of experience working on consensus and execution layers. Comfortable turning ambiguous, research-stage problems into structured designs, and then shipping them as performant and reliable systems.

EXPERIENCE

Senior Blockchain Protocol Engineer, [Polygon Labs](#)

Sept 2021 - Jul 2026

- Core protocol engineer for **Polygon Proof of Stake (PoS)** chain across three clients: [Bor](#) (execution layer, a fork of go-ethereum), [Erigon](#) (a performance-focused Ethereum client), and [Heimdall-v2](#) (CometBFT consensus layer).
- Sole architect of **reserved blockspace**, letting users reserve dedicated space in a block for guaranteed inclusion at predictable cost. Researched the approach and wrote an extensive [implementation specification](#) covering architecture, economic design, consensus implications, and end-user UX.
- Optimized the transaction pool design from **strict** to **eventual consistency**, improved the block building latency by cutting the **p95** pool snapshot time from **250ms** to **50ms**.
- Profiled and tuned hot paths across miner, storage/db, consensus, and RPC layers using garbage collection and performance analysis to find bottlenecks; contributed to a team effort to reduce block time from **2s** to **1.5s** and raise gas limit from **30M** to **160M** (approx. 7.5k TPS on transfers) on the same distributed validator set.
- Designed a soft **pre-confirmation** service for inclusion guarantees and a **private transaction relay** that bypasses the public mempool, delivered as a private RPC offering. Owned it end to end through research, design, implementation, layered testing (unit, e2e, stress, and chaos), rollout, and SLA monitoring. Live on mainnet with **44M transactions** in 2 months.
- Led research, design, and implementation of a **zero-knowledge proof system for PoS consensus proofs** using [SP1](#), establishing soundness of the construction and validating proof generation and verification end to end on mainnet.
- Co-authored and implemented [PIP-11](#) (Milestones), replacing probabilistic finality with **deterministic finality** using **Tendermint** consensus. Chain finality reduced to avg. **4-6 blocks** from a 256-block safety heuristic for users.
- Delivered a range of protocol work: p2p observability, snapshot (snap) sync diagnosis of mainnet sync issues, ancient-data pruning, tracing for L1 to L2 bridge system transactions, account abstraction, and extensive consensus tests.
- Shipped **10+ hard forks** and coordinated **3 private security releases** for an open-source client. Reproduced and fixed **15-20 bug bounty reports** across consensus and execution layers. **Tech Stack:** Go, Rust, Solidity, JavaScript, Python

Guest Lecturer, Ahmedabad University

Jan 2023 - Present

- Invited by [Dr. Sanjay Chaudhary](#) to co-teach **CSE 542: Blockchain Technologies** for CS undergraduates. Cover distributed-systems and blockchain foundations, Ethereum internals (EVM, State/Storage), consensus designs, and scaling solutions.

Ethereum Protocol Fellow, Ethereum Foundation

Jul 2023 - Nov 2023

- Selected for the EF fellowship to work on **Enshrined Proposer-Builder Separation (ePBS)** with **forward inclusion lists**, mentored by researchers and engineers from Ethereum Foundation and Prysm.
- Surveyed existing PBS designs, and worked with mentors to shape the **consensus spec** for a [commit-reveal](#) based design.
- Inclusion list work got converted into [EIP-7547](#) and implemented a PoC for the same in **go-ethereum** execution client. Code: [Github](#). Report: [Github](#). Presented work at [Devconnect Istanbul](#): [Presentation](#).

PROJECTS

Ethereum Virtual Machine (EVM) Simulator

Jun 2024

- Built a minimalistic EVM implementation in Go which includes **stack, memory, storage, and opcode** execution paths with an **opcode-level tracer**. Designed a **pluggable storage interface** with two backends: simple in-memory state, and a read-only adapter over a go-ethereum node that lets an opcode fetch data from a real chain's **LevelDB**. Code: [Github](#).

Indexoor - for indexing and querying state data for EVM based chains

Scaling Ethereum 2023

- Developed core modules for indexing **slot-level smart contract data** without the need to emit events pushing data into an off-chain SQL database allowing users to perform complex queries and joins on top. Integrated into OP Stack for Layer 2 chains. Code: [Github](#). **Tech Stack:** Golang, Python, Optimism Stack, Solidity, Slither (Static Solidity Analyzer).

EDUCATION

B.Tech in Information and Communication Technology (ICT), Ahmedabad University, India

Aug 2017 - May 2021

Relevant Coursework: Advanced Data Structures and Algorithms, Database Systems, Operating Systems, Computer Networks, Models of Computation, Cloud Computing, Probabilistic Graphical Models, Machine Learning.

TECHNICAL SKILLS

Languages: Go, Rust, C, C++, Python, JavaScript, Java, Solidity, MATLAB, Shell

Databases: LevelDB, PebbleDB, MDBX, MySQL, PostgreSQL, Redis

Tools: Git, Docker, NGINX, GraphQL, AWS/GCP Suite, AI agentic workflows, MCP, AI tools (Claude Code, Codex, etc.)

WRITING

[Why two-thirds](#): Deriving why consensus systems have **two-thirds quorum** and why crash-fault systems only need a simple majority.