■ Rust for Blockchain Development Roadmap

Goal: Learn Rust from scratch and master blockchain development to create your own chain, smart contracts, and blockchain tools.

■ PHASE 1: Rust Fundamentals (0–2 Months)

Goal: Become comfortable writing safe, efficient Rust programs.

1 Basic Syntax – Variables, functions, control flow

2 Ownership & Borrowing – Move semantics, lifetimes

3 Structs, Enums, Traits

4 Error Handling – Result, Option, match

5 Concurrency – Threads, async/await

Practice Projects: CLI calculator, To-do app, Guessing game

■ PHASE 2: Intermediate Rust + System Programming (2–4 Months)

Goal: Build efficient systems and prepare for blockchain-level coding.

1 Memory management, Cargo & Crates

2 Serialization with Serde

3 File handling, Networking, Multithreading

4 Cryptography basics (sha2, ed25519-dalek)

Practice Projects: Key-value store, CLI wallet, Proof of Work demo

■ PHASE 3: Blockchain Fundamentals (4–6 Months)

Goal: Learn how blockchains work and implement core components.

1 Blocks, Transactions, Hashes, Merkle Trees

2 Consensus algorithms – PoW, PoS, BFT

3 P2P networking and data propagation

Projects: Mini blockchain, Wallet system, Blockchain explorer

■ PHASE 4: Framework-Level Learning (6–9 Months)

Goal: Learn professional blockchain frameworks built with Rust.

1 Substrate (Polkadot) – Runtime, pallets, consensus

2 Solana – Smart contracts, Anchor framework

3 NEAR Protocol – Rust SDK and deployment

Projects: Substrate chain, Ink! token, Solana NFT app

■ PHASE 5: Advanced Blockchain Development (9–12+ Months)

Goal: Create production-grade blockchain systems and contribute to open-source.

1 Smart contract security and optimization

2 Advanced consensus and Layer 2 solutions

3 zk-SNARKs, elliptic curves, bridges

Capstone Projects: Build your own L1 chain, DeFi contract, Blockchain explorer

■ Estimated Timeline

1 Beginner Rust – 0–2 months

2 Intermediate Rust – 2–4 months

3 Blockchain Basics – 4–6 months

4 Frameworks – 6–9 months

5 Advanced – 9–12+ months