Joni Laitinen

joni.laitinen.m@gmail.com

SUMMARY

Blockchain Engineer skilled in Rust, Go, and Solidity, with expertise in building blockchain infrastructure, protocols, smart contracts, and dApps.

Rust Engineer adept in system programming, Wasm, backend systems, web applications, and blockchain technology.

SKILLS

Programming Languages:

Rust, Go, Solidity, Python, TypeScript, JavaScript

Blockchain:
Solana, Polkadot, Ethereum, Arbitrum, Polygon

Infrastructure, Protocol, Smart Contract, Layer1, Layer2, DeFi, DEX, Consensus Algorithms, Cryptography, Cybersecurity, NFT, dApps, Crypto Game, Bot

Libraries & Frameworks:
Anchor, Substrate, Serum, Hardhat, Truffle, Rollup, Wormhole, Tokio, Serde, Rocket, Diesel, Handlebars, ink!, Libp2p, Rust-Bitcoin, Actix, Actix Web, Seed, Wasmer, Stdweb, Clap, Wrap, Gin, Fiber, gRPC, Protobuf, RabbitMQ, Kafka, Web3.js, Ethers.js, Polkadot.js

Web:
Node.js, Express, GraphQL, HTML5, CSS3, React, Next.js, NestJS, WebSocket, Material UI, Bootstrap, Tailwind CSS

DevOps:
AWS, Google Cloud, Github, Docker, Jira, Jenkins, Kubernetes, Microsoft Azure

Database:
MongoDB, MySQL, SQLite, PostgreSQL, Sequelize, Redis

PROFESSIONAL EXPERIENCE

SlickDevs 07/2022 - 08/2024

Senior Blockchain Engineer

- Engineered high-performance Solana programs(Smart Contracts) using Rust and the Anchor framework, optimizing transaction latency by 20% and increasing throughput by 30% for DeFi applications and NFT games.
- Architected and developed custom DEX platforms, integrating features such as AMMs, liquidity pools, and DAO governance, achieving 99.9% uptime and processing over $1 million in transactions daily.
- Built customized blockchain infrastructures and protocols using Rust with Substrate and ink!, improving scalability by 25% and reducing network latency by 20%.
- Integrated blockchain solutions for secure interactions and enhanced decentralized functionalities using Wormhole for cross-chain messaging, achieving a 40% improvement in cross-chain transaction efficiency.
- Led the development of smart contracts using Solidity on Ethereum, focusing on performance optimization and security, including compliance with EIP-4337 standards, resulting in a 30% reduction in gas fees and zero critical vulnerabilities.
- Automated testing scripts in Python for backend systems, enhancing the CI/CD pipeline, reducing deployment times by 35%, and increasing code reliability across blockchain protocols and dApps.

Honey Finance 07/2021 - 06/2022

Blockchain Protocol Engineer

- Developed and implemented NFT lending and borrowing protocols using Rust, integrating with Solana, Polygon, and Arbitrum, achieving a 50% increase in transaction throughput and an 80% reduction in gas fees.
- Engineered high-performance smart contracts for variable interest rate loans and liquidity pools using Rust on Solana and Solidity on Ethereum, focusing on cross-chain interactions to leverage Solana's high throughput and reduce transaction costs by up to 60%, improving processing efficiency by 40%.
- Enhanced the Honey DAO governance framework by creating tools with TypeScript and Node.js to streamline voting and proposal management, improving governance participation by 35% and facilitating more effective decision-making.

Jungle Cats 04/2021 - 06/2021

Smart Contract Engineer

- Designed and implemented smart contracts for the Jungle Cats NFT project on Solana, using Rust and Anchor to handle over 10,000 NFT transactions efficiently.
- Integrated Metaplex for NFT minting and metadata management, enhancing contract functionality and optimizing interactions with the Solana blockchain.
- Utilized Solana CLI and localnet for deployment and testing, achieving a 30% improvement in contract performance and ensuring secure, scalable NFT operations.

Bifrost 10/2020 - 03/2021

Rust/Typescript Engineer

- Developed staking smart contracts in Rust for Polkadot, leveraging Substrate's modular framework and achieving a 30% increase in cross-chain liquidity through XCMP (Cross-Consensus Messaging).
- Engineered backend infrastructure with Rust and WebAssembly (Wasm) to enable secure cross-chain staking operations, reducing transaction latency by 40%.
- Built RESTful APIs in Rust and Node.js to facilitate real-time communication between the frontend and blockchain nodes, ensuring 99.9% uptime.
- Built the frontend with React.js and TypeScript, improving transaction processing times by 20% through optimized WebSocket handling.
- Integrated Polkadot.js API and Substrate RPC nodes to facilitate staking management and governance voting, overseeing more than $5M in staked assets across multiple blockchain ecosystems.

RumbleFish 07/2017 - 09/2020

Backend Engineer

- Contributed to building Radicle's decentralized protocol by designing and optimizing peer-to-peer communication using Rust's libp2p and mio libraries, improving message propagation speed by 30% and ensuring reliable network connectivity.
- Implemented cryptographic identity and authentication mechanisms using libsodium and ring libraries in Rust, enhancing security for code collaboration by 40%.
- Engineered microservice architecture and backend systems in Rust using Actix and Tokio, enhancing system scalability and reducing latency by 40% for high-traffic client applications.
- Developed and optimized RESTful APIs with Go and the Gin framework, implementing advanced authentication and rate limiting, leading to a 30% increase in API performance and security.
- Developed cloud-native backend solutions in Go using Docker, Kubernetes, streamlining CI/CD pipelines and reducing deployment times by 50%.
- Implemented real-time data streaming platforms in TypeScript using Node.js, NestJS, and RxJS, providing low-latency solutions for high-reliability data delivery.

EDUCATION

Oulu University 05/2013 - 04/2017

Bachelor's degree in Computer Science

- Operating System, Data Structures and Algorithms, Programming Language
- Networking, Database Management, Web Development
- Cryptography, Cybersecurity, Blockchain Technology
- Software Design Principles, Software Engineering, Project Management