Tirth Patel

t38patel@uwaterloo.ca 306-513-5508 tirth-patel.ca github.com/t38patel

Relevant Professional Experience

Blockchain Engineer, *Dandelion Networks (Startup)* | *Golang, gRPC, Solidity, Docker* 05/2022 – present

- Outperforming traditional Layer-1 PoW and PoS algorithms by developing an innovative client-leader **consensus architecture**
- Single-handedly decoupling their node validation mechanism using **Golang** to formulate their version of the Byzantine Agreement
- Implementing their patented lightning-fast client-leader paradigm to process more than **250,000** transactions per second
- Designed and deployed **Docker** containers with custom networks using Docker Compose to streamline **P2P** communication
- Leveraging protocols such as **gRPC** and **protobuf** to develop robust services and methods that effectively cater to diverse use cases
- Enforced various **OOP** design patterns; extensively applied the **Factory Pattern** and **Builder Pattern** for scalable, distributed systems
- Created a thread-safe in-memory data structure to maintain transactions associated with smart contract addresses
- Solving the **Buterin Trilemma** by employing a seamless transaction-linked directed acyclic graph versus a traditional blockchain
- Overcame uncertainties inherent in a startup environment, such as fundraising volatility and market fluctuations with resilience

Software Engineering Intern, *AlertDriving* | *PHP, PostgreSQL, DBeaver, Javascript* 01/2022 – 04/2022

- Led the integration of the license management audit trail using **PHP, Javacsript**, and **MySQL** to improve license inquiry and search
- Created several full-stack components for an internal dashboard using the **LAMP** (Linux, Apache, MySQL, PHP/Python/Perl) stack, aggregating thousands of key data points
- Designed and developed live HTML to PDF API endpoint providing up-to-date PDF reports for users
- Sped up tasks for the Operations Team by **50%** by automating the transfer of global client data across various spreadsheet suits

Software Engineering Intern, *Lumentum* | *C#, AWS, VB, Python, SQL* 01/2020 – 04/2020

- Boosted unit-search operation speed by ~**70%** by refactoring algorithms and data processes in **VB6** and **SQL**
- Saved **$12,000 / quarter** by designing an internal tool (Statistical Process Control system) from scratch with **C#, MS Access, SQL, JMP, JSL**, and **Python** which processed and allowed batch data to be visualized and analyzed for product engineers
- Optimized runtime complexities from **O(n^2) to O(n)** of internal **Visual Basic** and **C#** tools, speeding up common tasks by **3x**
- Used **AWS** to build an end-to-end log analytics solution that collects, ingests, processes, and loads both batch data and streaming data
- Successfully designed and integrated **Scrum/Agile** methodologies with **CI/CD** pipelines using Infrastructure as Code (IaC) principles

Projects

Blockchain Based Discord Clone, *dApp* 04/2023

- Leveraged **Solidity** and **Hardhat** to create smart contracts that utilized NFTs for memberships, allowing users to join specific channels on the Discord clone platform, with transactions being executed through the **Metamask** wallet
- Utilized **JavaScript**, **web3.js**, **React**, and **node.js** to add interactivity/functionality to the dApp, ensuring a seamless user experience
- Implemented **ERC721** tokens for the creation of NFTs, incorporating custom logic and functionality for ownership and access control
- Employed **Socket.io** to create a real-time, chat-based platform that could easily handle **1500+** users, ensuring efficient communication and interaction between users on the platform

AI NFT Generator, *dApp* 02/2023

- Leveraged **Solidity** for building smart contracts, **Hardhat** for testing and development, and **React** for UI design, resulting in a secure, user-friendly, and decentralized app that uses **AI**-generated art to mint NFTs
- Utilized **web3.js** for interacting with the **Ethereum** blockchain, **MetaMask** for secure user interaction, **Node.js** for building a backend server that communicated with blockchain and AI APIs, and **IPFS** for NFT storage, resulting in a fully functional and reliable platform

Triangular Arbitrage, *Cryptocurrency Arbitrage Bot* 06/2022

- Designed and implemented a custom triangular arbitrage bot with **Python** and **REST APIs**, leveraging real-time market data to execute profitable trades across multiple exchanges and achieving instant **2%** returns
- Tuned advanced trading patterns (**BUY-BUY-SELL/BUY-SELL-SELL**) to exploit inefficiencies across both **CeFi** & **DeFi** exchanges
- Leveraged the **Poloniex** and **Uniswap V3** exchanges which enabled profitable trading opportunities and provided in-depth knowledge of market trends and dynamics in the rapidly evolving world of cryptocurrency.

Skills

Core Languages: (Python, Golang, Solidity, C++, Javascript, Java, PHP, MATLAB, Bash)

Tools/Frameworks: (Smart Contracts, Ethereum, Hardhat, Metamask, gRPC, Docker, AWS, GCP, NodeJs, Flask, Flutter, EthersJs, Web3Js, REST, Ganache, Git, Kali Linux)

Education

Bachelor of Applied Science, Nanotechnology Engineering, *University of Waterloo* 09/2018 – 04/2023

- Relevant courses: Data Structures and Algorithms, Cryptography and System Security, Computer Networks, Computational Methods, Machine Learning A-Z, Engineering Programming, Simulation Methods, Advanced Statistics, Ethical Hacking