**TECHNICAL SKILLS**

|  |
| --- |
| **Programming Languages:** JavaScript, TypeScript, Java, HTML/CSS, C++, C#, SQL, Python  **Frameworks & Technologies:** Git, React, Next.js, Node.js, Express.js, MySQL, MongoDB, Firebase, Web3, Socket.io, Vercel, Netlify, Docker, Figma, Photoshop, Prisma, GraphQL, Prettier, Husky, Three.js, EC2, S3 |

**EXPERIENCE**

|  |  |
| --- | --- |
|  | |
| Full Stack Developer at [Ambition.so](https://ambition.so/)   * Built a no-code user interface that allows users to generate NFT collections using HTML Canvas which was 7x faster than the previous generator. * Created an entire admin dashboard that required using Apollo GraphQL to communicate with the backend server. * Worked on improving user flow and UI design that follows usability and accessibility best practices while utilizing auditing tools like for better quality. * Communicated with team members through daily standup meetings using Discord and used Notion for agile project management. * Debugged, fixed, and maintained a polyrepo of 7 GitHub repositories. * Used Git as a version control system. Submitted, reviewed, and merged pull requests on GitHub. | January 2022 - Present |
| CEO/Founder at [NFTHost.app](https://www.nfthost.app/)   * Developed an entire full-stack web application alone that involves planning, designing, implementing, and maintaining a web application (offering Website Hosting and NFT Utilities). * Implemented a crypto wallet authentication with JWT and secured REST API routes by using access tokens and middleware validators and sanitizers. * Built an entire website hosting platform using Next.js that allows users to have a custom subdomain, layout templates, and user analytics. * Added a CI system by creating GitHub Actions for running CodeQL Analysis and DockerHub Deployment. * Partnered with other Web3 companies such as [Thirdweb](https://thirdweb.com/), [Flair](https://flair.finance/), and [Web3 Philippines](https://web3philippines.org/) | December 2021 - Present |

**PERSONAL PROJECTS**

|  |
| --- |
| [stephenasuncion](https://github.com/stephenasuncionDEV/stephenasuncion) — (2022) Portfolio Website   * Built a full-stack web application using Next.js along with its serverless functions. * Created a 3D model of my room using Blender and implemented it on the website using Three.js. * Set up code formatter and git hooks using Prettier and Husky which enhanced code quality by 80% resulting in better performance. * Fetched data from GitHub’s API by using Octokit and Spotify’s API by making my own JS class. * Created a clone of GitHub’s Git Activity, MinGW’s console, and VSCode Editor by using React Components and ChakraUI |

[create-typedef-app](https://github.com/stephenasuncionDEV/create-typedef-app) — (2022) A Full-Stack Web Application Template

* Built an entire full-stack web application starter pack using Next.js, TypeScript, ChakraUI, NextAuth.js, Prisma and MongoDB.
* Integrated Prisma with MongoDB for strong type-safety which resulted in better code quality and scalability.
* Implemented an authentication system with NextAuth.js that includes GitHub, Google, and Credentials login.
* Secured routes by using Next.js’s middleware API, checking if session token inside cookie is present in the database.

[emoji.io](https://github.com/stephenasuncionDEV/emoji.io) — (2022) Multiplayer Online Game created with Web Sockets.

* Developed an entire online multiplayer keyboard game using Socket.io and Canvas with features including chat messaging and player movements.
* Implemented a payment system using Stripe.
* Deployed backend on Heroku (supports web sockets) and frontend on Netlify.
* Collaborated with a fellow college student.

[gencomp](https://github.com/stephenasuncionDEV/gencomp) — (2022) VSCode Extension that generates new React Component from selected code.

* Composed an entire VSCode Extension that could potentially help millions of new React users.
* Implemented a CI system that automatically publishes GitHub Repository to Visual Studio’s Marketplace.
* Used VSCode API to create custom commands, and menus on VSCode.

kaldereta — (2021) Unsigned Kernel Mode Driver that does memory modifications.

* Created an Unsigned Windows Driver that can read/write, allocate, and free a process’ memory.
* Connected driver with user mode window applications by hooking a window function.
* Built a sample program that can scan for memory patterns and inject dll files into a process.
* Implemented mouse and keyboard events simulation through kernel.

**EDUCATION**

|  |  |
| --- | --- |
| **Diploma in Computer Studies**  Langara College – Vancouver, BC  Cumulative GPA: 3.79/4.33  Awards: Dean’s Honor Roll, 3 Terms | September 2020 – August 2022 |