LEMOND WYATT

(661) 754-5093 | auraxiumdev@gmail.com | github.com/Auraxium | linkedin.com/in/lemondwyatt760568433

SUMMARY

I'm a **fullstack software developer** based in Los Angeles County, California. I specialize in **React** and its **NodeJS** eco-system to deliver applications across the **web**, **desktop**, and **mobile devices**. For an in-depth overview of me and my projects, check my website at **https://lemond.auraxium.dev**

EDUCATION

California State University of Northridge (CSUN) — **Northridge, CA**
*Bachelor of Science in **Computer Science*** — *Graduated May 2023*

TECHNICAL SKILLS

Languages: TypeScript, JavaScript, Python, C#, Kotlin, SQL
Frameworks & Tools: React, React Native, Next.js, Tailwind CSS, Tauri, Express, PostgreSQL, Chrome Extension APIs, Unity, Steam Networking API, Git
Concepts: Fullstack, Web Development, REST, APIs, Inter-Process Communication (IPC), Database design, Algorithms.

EXPERIENCE PROJECTS

Link Player – *Multi-platform music player – 2023-2026 (on going)*
Stack – React · TailwindCSS · Express · PostgreSQL · Tauri (Desktop) · React Native / Kotlin (Android)
Links – Linkplayer.auraxium.dev (Website)

- Built a polymorphic platform adapting architecture that allows dynamic playback of music regardless of source. Utilizes several **embeds**, **APIs**, and iframe manipulation tactics across services such as YouTube, SoundCloud, Spotify, and your local files. Platforms swap seamlessly so playback, seek, volume control, etc all work the same from song to song.
- Shipped desktop version via **Tauri** spawning a **Node.js** sidecar over **IPC** for local file streaming.
- Rebuilt the Android app from the ground up in **React Native** with native **Kotlin** to utilize WebView JS injection manipulation to enable background music playback.
- Designed an Express **REST API** for music platform queries, and **PostgreSQL** schema for multi-platform account auth and cross-device library sync.

Macros & Keybinds – *Key-macro and rebinding utility – 2025*
Stack – React · TailwindCSS · Tauri · Python · C++
Links – Github.com/Auraxium/Lo-Macro **(**GitHub release & source code)

- Built a macro utility with a **React+Tauri** and a **Python sidecar** handling macro logic, timing curves, conditional triggers, and recorded input sequences over **inter-process communication**.
- Solved a same device input loop problem by writing a native **C++** Windows service with kernel level hooks to distinguish human keystrokes from program-generated ones by virtual device.

Relics QOL – *Build-crafting tool for the videogame "Elden Ring Nightreign" – 2026*
Stack – React · TailwindCSS · Binary Parsing · Decryption
Links – Relics-qol.auraxium.dev (Website) | Github.com/Auraxium/Relic-Builder (Source code)

- Developed a **website** where players upload their save file and offers tools to plan and optimize equipment much easier
- **Reverse-engineered** and **decrypted** Elden Ring Nightreign's save-file format, parsing the binary into structured inventory data.
- Built a filtering and build-crafting interface, plus a scoring **algorithm** that searches thousands of items and displays the optimal loadout combinations in seconds.

Twitch Sidebar Favorites – *Chrome extension for Twitch.tv – 2024-2026 (on going)*
Stack – HTML/JS/CSS · Chrome Extension API · Twitch API · PostgreSQL
Links – "Twitch Sidebar Favorites" on Chrome Web Store | Github.com/Auraxium/TSF (Source code)

- Shipped a **Chrome extension** that injects **HTML** for a favorites list directly into Twitch's sidebar via content scripts.
- Built a popup UI for favoriting, watch-later, and VOD tracking channels. Synced across devices through an **Express REST API** and **PostgreSQL** backend.

VoidOut – *3D platformer game – 2021*
Stack – Unity 3D · C# · Multiplayer Networking
Links – Steampowered.com/app/1540790/VoidOut_Parkour **(**Steam) | Auraxium.itch.io/voidout-parkour (Play free on Itch)

- Designed and built a 3D platformer solo in **Unity engine** with procedural level generation. Published on **Steam** and **Itch.io**
- Implemented a multiplayer mode using the **Steam Networking API** for easy netcode logic.

All my projects are hosted on my domain auraxium.dev. served by my Hetzner VPS (shared cloud, $7/mo).