

Leo Hoerdemann

leo.h.trip@gmail.com | leohoerdemann.xyz | github.com/leohoerdemann | linkedin.com/in/l30-h/

Skills

Programming Languages: C#, Python, C++, Java, JavaScript, Arm Assembly, and C

Framework and Libraries: .NET, Unity, Wwise, Android SDK, React, Astro.js, Node.js

Tools and Technologies: Git, Docker, Kubernetes, Jenkins, Jira, Google Cloud Platform, Linux, Ableton

Databases: MSSQL/ MySQL, MongoDB, Firestore

Education

Michigan State University – BE in Computer Science, Minor in Computational Math, Minor in Business 2021-2024

Experience

IT Intern, Truly Engaging – Carol Stream, IL | *Asp.NET, C#, MSSQL, Windows Server* May 2023 – August 2023

- Developed a minimal API service using **ASP.NET** to streamline interactions with an **MSSQL** server for internal use
- Maintained various printing, packaging, and inventory machines and systems daily, reducing down-time

IT Intern, Perma-Pipe International Holdings – Niles, IL | *Freshdesk* May 2022 – July 2022

- Managed IT services for a branch office using **Freshdesk**, **MySQL**, and **VirtualBox** to ensure seamless operations
- Decommissioned, logged, and properly disposed of a backlog of out-of-service devices, improving equipment tracking

Easy IT Support, System Engineering Intern – St. Charles, IL | *Windows Server* June 2020 – December 2020

- Serviced various clients in setting up and maintaining their IT operations, aiding in operation enhancements
- Repaired individual client devices for data recovery or restored continued use

Gibson Electrical, Warehouse Team Member– Downers Grove, IL | *Excel* June 2021 – August 2021

- Built out a robust digital system for inventory locations within the warehouse to prepare orders faster
- Checked in and out various tools and equipment, ensuring all items were logged and functional
- Managed warehouse inventory availability for various sites and jobs ensuring timely availability of proper equipment

Projects

UWM IT Datamart Microservice for Bitbucket | *Asp.NET, C#, MSSQL, Orkes, Jenkins, Docker, Bitbucket* [Capstone Page](#)

- Collaborated with United Wholesale Mortgage to create a full stack application for data aggregation surrounding the Bitbucket repositories within the company
- Created a microservice using **ASP.NET**, **MSSQL**, **Docker**, and **Orkes** to collect data on commits, dependencies, frameworks, team activities and more, powering data visualization and future data modeling
- Implemented a **React** and **Node.js** frontend to display insights, aiding in informed decision-making across the company

ChatGPT Powered Chat Bot | *Asp.NET, C#, Google Cloud Platform, React, Docker* [GitHub Repository](#)

- Developed a Discord chat bot integrated with the **OpenAI API** to provide the latest model to many users at low cost
- Built using **ASP.NET** and hosted on **Google Compute Engine** via a **Docker** container, utilizing **Firestore** and **Secret Manager** for data storage and security

Unity Based First Person Movement Shooter (ChromaKill) | *Unity, C#, Ableton* [Itch.io](#)

- Contributed as a sound designer on a team of ten over four months to create a movement FPS game in **Unity**
- Created and implemented sound effects using the default unity audio system, enhancing the gaming experience
- Participated in weekly feedback sessions with Bungie to improve game mechanics and development workflows

Audio Design and Implementation (Spartasoft Studio) | *Unity, C#, Ableton, Wwise* [Itch.io](#)

- Created in **Ableton** and implemented using **Wwise**, audio assets for various game projects across multiple semesters
- Worked on a team of approximately eighty, on games that include 3D, 2D, platformers, party, and networked games

Reddit Sentiment Analysis Stock Trading | *Python, Keras, Finance APIs, MongoDB* [GitHub Repository](#)

- Collected and stored Reddit posts relating to investing over the course of a month into a **MongoDB** database
- Utilized **Keras** to assign sentiment values to each post
- Implemented a stock trading simulation based on sentiment analysis, achieving a 7% return using real-world stock data