

Sebastian Torres

☎ 720-326-9805 | @ seb.torres3600@gmail.com | 🌐 <https://seb-torres3600.github.io>
in <https://www.linkedin.com/in/sebastiantorresd> | 📍 Seattle, WA

EDUCATION

University of Colorado at Boulder

B.S. in Computer Science

Boulder, CO

Aug 2019 – May 2023

SKILLS

Languages: C/C++, Python, Go, Swift, Bash

Technologies: Git, AWS, Azure, Chef, Docker, Unix

Skills: Multi-threading, Computer Graphics, Object-Oriented Programming, Software Design Patterns, Problem Solving

WORK EXPERIENCE

Amazon Web Services - Device Farm

System Development Engineer

Seattle, WA

Jun 2023 – Current, Full-time

- Create and maintain infrastructure to facilitate the testing of mobile and web applications on real physical devices in the AWS cloud
- Analyze and solve customer problems to create a better testing platform so users can create the best app for their customers

University of Colorado at Boulder - OIT

Unix System Administrator

Boulder, Colorado

May 2021 – May 2023, Student Assistant

- Maintained and upgraded essential systems for the University of Colorado at Boulder used by staff and students everyday
- Analyzed and resolved customer problems associated with servers, applications, and software
- Quickly responded and solved issues related to critical pieces of infrastructure to maintain the operability of university functions

Amazon Web Services - Device Farm

System Development Engineer

Seattle, WA

Jun 2022 – Aug 2022, Internship

- Worked to create a CI/CD pipeline for safe deployments to get essential upgrades and code fixes to create a better experience for customers
- Created a deployment strategy that quickly disperses essential code to thousands of servers requiring minimal human interaction with automated rollback with status checks along the way to validate a safe deployment

RELEVANT COURSEWORK

Concurrent Programming

This class introduced the theory and practice of multicore programming. The first part of the course presented foundations of concurrent programming: mutual exclusion, wait-free and lock-free synchronization, spin locks, monitors, memory consistency models. The second part presented a sequence of concurrent data structures and techniques used in their implementations (coarse-grained, fine-grained, optimistic and lock-free synchronization)

Computer Graphics

This class introduced computer graphics techniques. Topics included interactive techniques, 3D viewing and models, clipping, transformations, projection, removal of hidden surfaces, lighting, textures and shadows using OpenGL

Design and Analysis of Operating Systems

This class introduced the structure and function of operating systems as an intermediary between applications and computer hardware. Topics included OS design goals, hardware management, multitasking, process and thread abstractions, file and memory management, security, and networking