

## Richard Cunard

---

445 SW Tunison Ave. Apartment 18  
Corvallis Oregon, 97333

[cunardr@oregonstate.edu](mailto:cunardr@oregonstate.edu)

(503) 686-9512

### **Objective**

I am seeking a position that will allow me to further develop my skills as a software developer.

### **Education**

#### **Oregon State University**

B.S. in Computer Science, Minor in Mathematics

Graduated: June, 2018

M.S. in Computer Science

Graduation year: 2020

### **Programming experience**

Proficiency in the following programming, scripting and assembly languages: **C, C++, C#, Java, Swift, Python, GLSL, PHP, HTML, SQL, Haskell, Matlab, Bash scripting, MASM32.**

### **Skills and project experience include:**

- Software Testing                      Technical Writing
- Artificial Intelligence                Machine Learning and data mining
- Numerical Analysis                  Statistical modeling and simulation
- 3D Modeling and Animation        Parallel programming (OpenMP, OpenCL)
- Graphics and Shader Programming (OpenGL, GLSL, Vulkan)
- CPU and GPU Architecture        GPGPU Programming
- iOS and Android Development    Database Programming
- Development in the Unity and Unreal game engines
- Agile Development Practices
- Familiarity with Unix and Linux Environments

### **Other skills/experience**

- Experience working in high pressure environment with strict deadlines.
- Strong written and spoken communication skills
- Conflict resolution/working with uncooperative parties to achieve a positive outcome

### **Work Experience**

#### **Teaching Assistant**

September 2018 - Present

OSU College of Electrical Engineering and Computer Science

- Oversee groups of students working on their senior projects.
- Provide academic and technical guidance to student groups.
- Review and grade student work.
- Help to find positive outcomes to conflicts between group members.

**Course Assistant**

January 2018 - March 2018

OSU College of Engineering, School of Mechanical, Industrial and Manufacturing Engineering

- Constructed 3D models used to simulate a robotic grasping arm.
- Began development on a Matlab program that performs geometric transformations on OBJ files for use in mechanical data visualization.

**Board of Directors, Owner Relations Committee Chair**

First Alternative Cooperative

June 2015-Present

- Elected to two consecutive three-year terms.
- Expanded the board's public presence to improve communication with co-op ownership
- Helped oversee board elections as part of the Board Recruitment and Elections Committee

**Project Experience**

**RipRap, Surfing App (ongoing):** Currently, I am working as the sole application developer on the project, building an iOS application that will work in conjunction with a wave forecast database to provide the user with available surfing locations.

**PolyVox, VR/Unity-based 3D Art Program:** Working in a small team, I was the primary developer on the application layer. My contributions include mapping the UI to the backend functions. Additionally, I developed painting tools and a save/load system for the program.

**Economic Growth Model:** Working with a PHD candidate, I developed an economic analysis program which modeled economic growth in a monte carlo simulation across ten million parameter configurations.

**Robotic Arm Visualization:**

Aided Dr. Cindy Grimm in developing a pipeline to allow robotic systems constructed in the OpenRAVE virtual environment to be rendered in the Unity game engine.

**Rental Listing Analysis:**

I worked in a small team to develop a machine learning system which uses previous apartment listings to predict interest level in newly posted listings. My main contributions were engineering additional features from the provided data (for example, determining the school district of a given apartment based on its address).

**Board Game AI:**

I developed two competing AI programs able to play the board game Tak for the purposes of comparing two designs. One of the AI operates using a Monte Carlo Search Tree algorithm across a wide breadth of potential game outcomes, while one operates using a limited ply Alpha-Beta search tree algorithm. Both AI were implemented using an existing open source Tak game engine to perform board state management and win-state evaluation.