

Ryan Dern

80 Candlewood Lane, Hanover, Massachusetts, 02339 * 7814240773 * ryan.m.dern@gmail.com

WORK EXPERIENCE

OPEN LEARNING EXCHANGE, Cambridge (Remote), MA

Software Engineer Intern, Jul 2017 - Present

- Used GitHub and Gitter to manage workflow and collaborate with members of the development team.
- Used Vagrant and VirtualBox to build and manage virtual software development environments.
- Wrote and maintained clear documentation using Markdown.

TARGET, Hanover, MA

Electronics Sales Floor Team Member, Jun 2015 - Dec 2016

- Responsible for Electronics, Toys, Sport, and Seasonal departments.
- Assisted guests in finding the best solutions for their technology needs.

CHAMPLAIN COLLEGE, Burlington, VT

Research & Development Project Assistant, Aug 2013 - Jan 2014

- Collaborated with a small team to complete miscellaneous technical projects as directed by supervisors.
- Prepared and distributed career-related documents to the student body.
- Served as an ambassador at the Career and Internship Fair.

EDUCATION

CHAMPLAIN COLLEGE, Burlington, VT

Game Programming

- Completed 75 credits towards a Bachelor's of Science degree in Game Programming.
- Hackathon participant and IGDA chapter member.

SOUTH SHORE VOCATIONAL TECHNICAL HIGH SCHOOL, Hanover, MA

High School Diploma, Jun 2012

- Obtained trade certificate in Information Technology.
- SkillsUSA competition gold medalist for Technical Computer Applications.

VOLUNTEER EXPERIENCE

TURBINE, Needham, MA

User Research - Game Tester, Mar 2017 - Mar 2017

- Acted as a user research subject during focus test sessions and provided qualitative feedback on development builds of "Game of Thrones: Conquest".

ADDITIONAL SKILLS

- Primary languages: C/C++, C#, Python, MySQL.
- Experienced with command line interfaces (Git Bash, PuTTY, UNIX) and virtual environments (VirtualBox, VMWare).
- Version control with Git and SVN. Issue tracking and project management with JIRA, GitHub, etc.
- Experience working in multidisciplinary teams in an Agile/Scrum environment.
- Familiar with cross-platform development, including deployment to mobile devices, game consoles, and embedded systems.
- Familiar with cloud technologies such as Amazon Web Services (AWS), Heroku, and GE Predix. AWS Certified Developer Associate in progress, expected October 2017.

PROJECT EXAMPLES

Omega Warp: A game for Windows and Xbox 360 written in C# using the Microsoft XNA 4.0 API.

Demo video available at: <http://bit.ly/OmegaWarpVideo>

Features:

- Particle system that made use of object pooling design patterns to avoid memory fragmentation and drastically increase performance.
- Input system that allowed the use of keyboard/mouse controls on PC and game controllers on Xbox.
- Behaviors for enemy units are modular so they can be easily extended, mixed, and matched to create new content quickly.
- Pixel shaders implemented in HLSL for visual effects such as bloom.

Custom C++ Game Engine with Allegro 5: Created classes to encapsulate Allegro functionality for use as a generic game engine.

These included:

- A Graphics System responsible for drawing images and text to the display.
- An Animation System that held lists of sprites and timing information.
- A Unit Manager that contained a map of all Units in a game and was responsible for calling the Update and Draw functions for those Units.
- An Input Manager to detect input from various input devices and fire Events in an Event System to carry out their game functionality.
- A Game object to hold all game systems and managers as well as the main game loop.

Predix Digital Volcano: Forked GE's Predix Volcano App repository on GitHub and set up my own instance of the app to run on the Predix cloud platform.

Try the app at: <http://bit.ly/RMDernPredixVolcano>

Tools used:

- Cloud Foundry CLI
- Python 2.7 and PredixPy
- npm and gulp

Tasks and Scheduling in Real-Time Systems: Ongoing project for a "Development of Real-Time Systems" online Master's course offered by EIT Digital.

Overview:

- Using the FreeRTOS kernel to learn about and implement task scheduling for real-time embedded systems.