January 19 2023

Morgan Davis

Santa Cruz, CA
mkd@mkd.net
(831) 419-8651

SUMMARY

I'm a software generalist and an experienced technical lead. I'm self-motivated, communicate well, and enjoy solving difficult problems.

EXPERIENCE

Study Sabbatical — Sept 2022–Present
Studying computer graphics, realtime rendering, C++, and other areas of personal interest.

Principal Software Engineer, Turtle Beach Inc. — Remote — Feb 2020–Sept 2022
I lead a team of six, developing several cross-platform native apps for iOS, Android, Windows and macOS. I also worked closely with engineering and product management stakeholders to define requirements and schedules, did code reviews and mentored junior engineers. I wrote code daily, primarily C#/Xamarin, with some Obj-C, C++, Swift and Java mixed in.

Consultant, Turtle Beach Inc. — Remote — Jun 2018-Feb 2020
I provided technical expertise and code contribution for multiple successful product launches. I worked closely with in-house and offshore development teams to develop and debug native applications for iOS, Android, macOS and Windows.

Senior Software Engineer, Innovation, Plantronics Inc. — Santa Cruz, CA — Oct 2016-Jun 2018

Software Engineer, Innovation, Plantronics Inc. — Santa Cruz, CA — Oct 2012-Oct 2016
I owned software development for a team of multidisciplinary engineers exploring new ideas and technologies. The Innovation Team worked hand-in-hand with the Corporate Strategy department, doing things that were outside the comfort zone of the company as a whole -- we created conceptual prototypes, early product iterations, and strategic partner integrations. I used a breadth of software languages and technologies on a regular basis, created many demo applications and partner integrations, device SDKs, and pilot program software components. I also acted as a brand hero, performing internal and external technology demos, working directly with partners on technology integrations, and sponsoring numerous hackathons.

Software Engineer, AOptix Technologies Inc. — Campbell, CA — Jul 2011-Oct 2012
I ported and optimized core image processing algorithms, and developed middleware and front-end UI for an iPhone-based iris and fingerprint recognition platform. I ported many image processing algorithms from Linux/C++ to iOS/CoreImage/CoreGraphics, used CoreImage and OpenCV to provide real-time iris & face finding functionality, and specified & implemented the 30-pin MFi communication protocol. I also created and skinned the native iOS app interface, refactored the core application into a modular SDK, and designed and implemented several manufacturing calibration and configuration tools. Mathematics and optical physics knowledge were frequently utilized.

Founder, MKD Software LLC — Placerville/Santa Cruz, CA — Jun 2001-Jul 2011
I developed free and shareware applications for Mac OS X. All operations were performed by myself including: application design & development, sales & marketing, web & sales backend, and customer relations & support. Most native application code was written in Objective-C with Cocoa, with other components written in C with Carbon or other libraries. Several of my titles enjoyed widespread public use and commercial success.

EDUCATION

B.S. Computer Science, 2011 – University of California, Santa Cruz

ENUMERATED SKILLS

- Mathematics and Computer Science theory
- OpenGL, GLSL and associated technologies
- Cross-platform C++ development
- macOS & iOS development with Objective-C and Swift
- C#, .NET, and Xamarin development
- Ability to learn new technologies quickly
- Good at building productive & respectful interpersonal relationships
- Excellent written and verbal communication skills

EXTRACURRICULAR

- 👇 Game engine
- Astronomy photography
- Downloading NOAA weather satellite images in my attic
- Wrenchin’ on, tunin’, and trackin’ a rotary Mazda RX-7

GAME ENGINE

My passion project is developing a cross-platform 3D game engine. Source/binaries/demo are available upon request.

Key features:
- Written primarily in C++
- Runs on Linux, macOS, Windows, and Raspberry Pi OS
- OpenGL rendering API
- Ridigidbody physics via Bullet
- Robust scene graph with common manipulation & utility functions
- Model importing via Assimp
- Robust materials system
- Basic forward-rendering -- Phong-Blinn lighting, Skyboxes, transparency
- Keyboard and direct mouse input
- Planned & in-progress: kinematics integration, normal maps, shadows, deferred shading effects, skeletal animation, multithreading, client-server networking, audio, and scripting support.