

DYLAN JOHNSON

3101 S. Taylor St. Apt 745 ◇ Little Rock, Arkansas 72204
(501) · 547 · 1621 ◇ dylanatasmsa@gmail.com

EXPERIENCE

Emerging Analytics Center, UALR

Software Engineering Intern

October 2014 - Present

Little Rock, AR

- Worked under Dr. Carolina Cruz-Neira, the inventor of the CAVE system.
- Developed Data Visualization solutions for Oculus Rift and CAVE system.
- Used Unity 3D (scripting with C#) for 3D programming and model manipulation.
- Used OpenCV (used OpenCvSharp to integrate with C#) for computer vision applications.

Search User Interface Team, Ancestry.com

Software Engineering Intern

June 2014 - August 2014

Provo, UT

- Worked with a team of engineers to remove and update an old and restrictive codebase.
- Used C#, ASP.NET MVC3, Microsoft SQL Server, Visual Studio 2012, Team Foundation Server, Windows 7 to develop software.
- Used the SCRUM agile software development framework.

EDUCATION

University of Arkansas, Little Rock

B.S. in Computer Science

August 2014 - Expected Graduation Year: 2017

- Courses: Language Structure, Computer Systems and Assembly Language, Linear Algebra, Operating Systems, Databases, Theory of Computation.

Arkansas School for Mathematics, Sciences, and the Arts

High School Graduation

August 2012 - May 2014

- Third place Intel International Science and Engineering Fair project in Materials Engineering at the local level for my research on the optimization of aluminum can camping stoves.
- Courses: AP Calculus AB, Calculus 2, Calculus 3 (Vector Calculus), Advanced placement Physics C Mechanics, Advanced Placement Physics C Electricity and Magnetism, Computer Programming 1, Computer Programming 2, Data Structures and Algorithms, Introduction to Web Application Development, Graphics Programming, Discrete Mathematics.

ONLINE

<https://github.com/DylanTheVillain>

<https://www.linkedin.com/in/DylanJohnson1>

LANGUAGES

English: Native Speaker

PERSONAL PROJECTS

[CodeBin](#)

December 2014 - Present

- An online Python development environment.
- The backend was build using Python 3.4 and Django 1.7.
- The frontend uses Ace for the code editor and Skulpt for the Python interpreter.
- Currently the user can create new projects and access them with permalinks, fork projects, browse projects, edit and save projects, and make projects public or private.

Rust

November 2014 - Present

- My current projects through which I am learning the Rust Programming Language that is being developed by Mozilla.
- [Rust-Game](#) My current attempt to build a small multiplayer game in Rust.
- [Little-Rust-Tcp](#) The small TCP Socket library that I am writing for the game.
- Both of the mentioned projects are being managed using Cargo.

TECHNICAL STRENGTHS

General Programming

Languages: C, Java, Python, C#, Rust, JavaScript.
Version Control Systems: git, Team Foundation Server.
Debuggers: gdb, Visual Studio 2012 Debugging Software.
Package Managers: Cargo(Rust), PIP(Python).

Web Programming

Server side: Python, C#, PHP, and SQL (MySQL, Microsoft SQL Server).
Client side: JavaScript, CSS, (X)HTML.
Servers: Apache, Internet Information Services.
Web Frameworks: Django, ASP.NET MVC3 C#.
Common Gateway Interface: Python.

Development Enviromnents

IDEs: Visual Studio 2012, Visual Studio 2013, Monodevelop, Xamarin.
Text Editors: Vim, Sublime Text, Atom.

Operating Systems

Linux: Debian, Ubuntu, Manjaro, Linux Mint, Slackware.
Windows: XP, 7, 8.

Unix System Administration

Proficient with command line utils: grep, ssh, vim, nano, tar etc.
Experience with: configuration files, package managers, compilation, makefiles.