

Nicholas Jean Lochner

Email: njlochner@gmail.com

Personal website with code samples <http://njlochner.com>

LinkedIn: <https://www.linkedin.com/in/nicholaslochner>

EDUCATION

University of Illinois at Urbana-Champaign

- Bachelor of Science in Computer Science
- College of Engineering Dean's List
- Graduated with Honors

Urbana, IL
May 2016
GPA: 3.71/4.00

SKILLS / STRENGTHS

Technical Skills:

- **Languages:** C#, C, C++, Java, Python, Ruby, ASP.NET, JavaScript, TypeScript, HTML, CSS, SASS, PHP, SQL, PowerShell, Bash, InterSystems Caché, Verilog, LaTeX, R, Visual Basic
- **Software and technologies:** Subversion, Git, jQuery, Bootstrap, Python Flask framework, Microsoft Visual Studio, JetBrains IDEs, Xcode, other IDEs, IIS, Amazon Web Services, Mathematica, Unity Game Engine, Microsoft Office/LibreOffice suite, MIPS, Oculus Rift development, Familiar with Android Development
- **UNIX/Linux & Windows Systems / Web Server Administration**

Professional strengths:

- **Self-motivated, fast learner, creative thinker, eagerness to learn new technologies, effective time manager**

EXPERIENCE

Epic Systems Corporation

Software Developer

Verona, WI
July 2016 – April 2019

- Primarily worked on Epic's internal version control, build utilities, bug tracking, and project management software.
- Developed a C# ASP.NET web application to track the migration of Epic's entire codebase to a modern architecture. The application allowed developers to indicate which files in version control are migrated and obsolete during the code-merge process. The tool also allows reporting for management purposes and saves hundreds of hours of manual work each year tracking the progress.
- Developed build utilities for Epic's proprietary C# ASP.NET web application framework. The suite includes a set of servers which automatically build the codebase when there are new revisions. The ability to retrieve the latest build was added to the process of source code checkout. The servers also make commits to source control to update transpiled source code (TypeScript compiled into JavaScript, and SASS compiled into CSS). A tool was also included to allow developers to manually build a minimal set of projects, automatically calculating and building the dependency tree. The utilities save thousands of hours per year.
- Maintained and added new features to Epic's continuous integration system which builds and tests new changes before they are committed to version control. Added an automated C# unit test runner which runs and reports results of unit tests before source code is committed. Assisted in the transition of the continuous integration system to a distributed server network.
- Developed a framework to track user activity and system performance across the majority of our internal version control and applications. The data is used to help troubleshoot issues and to identify areas for improvement.
- Maintained and assisted in project design/code review for other internal C# ASP.NET web applications for version control, source code review, bug tracking, project management, internal/customer exams, time logging, and tracking for customer billing/trips.
- Developed support to use a customer-facing data analytics tool for internal use to generate reports on internal/customer exams and customer trips.

Wolfram Research

Software Engineer Intern

Champaign, IL
May 2015 – May 2016

- Worked on various projects involving connected devices and audio processing in the Wolfram Language and Mathematica in an agile development environment.

- Developed a device driver for the Philips Hue connected home lighting system using the Wolfram Language DeviceFramework.
- Developed a real-time audio visualizer for the Philips Hue system. The visualization synchronizes the lights with audio in real time, and can analyze a file during playback or use microphone capture. This was publicly demoed at the Pygmalion Festival 2015.
- Projects involved using the Wolfram WSTP and JLink frameworks to create C, C++, and Java libraries for use with the Wolfram Language and Mathematica.

University of Illinois at Urbana-Champaign
Course Assistant for Systems Programming

Urbana, IL
 Summer 2014 – Spring 2015

- Developed two digital programming exams testing students on C string manipulation, file I/O, system calls, POSIX threads, inter-process communication (signals, pipes), and networking.
- Assisted the students during weekly office hours and on an online discussion board.

PERSONAL PROJECTS

Senior thesis – The Design and Implementation of an Autonomous Surveillance and Security Drone

- For my senior thesis, I designed and implemented an autonomous unmanned aerial vehicle which can patrol an area and identify humans. The thesis is available here in PDF form: <http://njlochner.com/thesis.pdf>
- The drone includes a Raspberry Pi computer which controls it. The computer runs an on-board web server and a secure WiFi access-point running a controller web-application built with Python.
- The web application uses the Google Maps API and allows for the drone to be sent to a GPS way-point, specify the speed and altitude, and specify a circuit for the drone to patrol.
- The drone has a camera, and sends images to a computer connected to the drone's WiFi network. This computer continuously processes images with OpenCV in Python to detect humans. Humans are identified using a Histogram of Oriented Gradients method conjunction with a Linear Support Vector Machine. When a person is detected, the machine sends an email to notify the user.

Automated web application vulnerability tester

- Checks a list of IP addresses for vulnerabilities, using Nmap to scan for running services.
- Checks for Heartblead, Shellshock, XSS, and a return-to-libc buffer overflow via non-malicious exploits.
- Aggregates URLs on host systems and writes vulnerability data to a JSON file.
- Includes Metasploit auxiliary module support.

Real-time strategy game with procedurally generated planet terrain

- Features a procedural terrain generator written in C++ using the Diamond-Square algorithm.
- Program generates terrain for an Earth-sized planet, including ice caps.
- Displays a two-dimensional topographic map with the player's cities and units. Player and AI can manage their cities and units, and attack and conquer enemy cities, using path finding via the A* algorithm.

Teamspeak 3 chat/music bot

- Developed a chat bot in C for Teamspeak 3 voice communication software which communicates with the server via a telnet interface, reading in text commands from users to perform functions.
- Parses URLs containing audio to play back through the Teamspeak 3 server.

Subversion Portfolio website

- Developed a personal portfolio website written in Ruby with Bootstrap for CSS styling which includes a file browser for all files in the projects, capable of showing past revisions.
- Parses SVN log and list XML data to render the portfolio.
- Allows viewers to comment on each project, storing the comments in a MySQL database.

REFERENCES

- Available upon request