

RYAN BURLESON

ryan.e.burleson@gmail.com
+1 480 274 2594
<https://www.linkedin.com/in/ryaneburleson>
<https://ryanburleson.dev>

EDUCATION

****B.S. Computer Science****
Milwaukee School of Engineering
September 2019 - May 2023

WORK EXPERIENCE

****Software Developer / System Administrator / Inventory Manager****
Digital Bridge
September 2023 - February 2024
Built and maintained internal tools using Python, REST APIs.
Collaborated with coworkers to determine software specifications and use cases.

PROJECT EXPERIENCE

****Email Classifier****
Python, Flask, Scala, Apache Spark, Postgres
Objective: Implement and benchmark different algorithms for computing convolutions using CUDA.
- Trained a machine learning model to automatically mark emails as spam.
- Created a REST API using Flask to allow communication between client applications and the database.
- Utilized Apache Spark to collect and process training data for the machine learning model.
- Deployed API and machine learning model on a cloud server.

****CUDA CNN Implementation****
C++, CUDA, Python, Matplotlib
Objective: Implement and benchmark different algorithms for computing convolutions using CUDA.
- Designed and implemented four different algorithms for computing convolution.
- Used CUDA's shared memory to improve throughput.
- Benchmarked each algorithm and visualized the results using PyPlot.

****Coin App****
Python, Shell Scripting, OpenCV
Objective: Build an app that would help users find rare coins and organize their coin collection.
- Wrote code to scrape Ebay and other online databases for training data
- Collected and cleaned 40,000 training images with labels
- Researched and tested methods of segmenting images for Optical Character Recognition
- Participated in weekly stand-up meetings and meetings with project advisor.

****Digital Painting Website****
Go, JavaScript, WebSocket, WebGL, GLSL, HTML, CSS
Objective: Create a website where users can draw a digital canvas.
- Developed server using Go to synchronize drawing actions between users.
- Implemented two-way communication using WebSockets to update canvas in real-time.
- Created a web page using HTML, CSS, and JavaScript to display the interactive canvas.
- Integrated GPU rendering using WebGL with GLSL shaders to improve performance of the client.
- Deployed project on a DigitalOcean Droplet.

TECHNICAL SKILLS

Go, Golang, Rust, Java, JavaScript, HTML, HTML5, CSS, REST, REST APIs, APIs, WebSocket, NodeJS, React, C, C++, CUDA, WebGL, GLSL, MySQL, MongoDB, Postgres, SQL, Databases, Shell, Shell Scripting, Scripting, Agile, Scrum, Python, NumPy, Pandas, Scikit-Learn, Sklearn, Matplotlib, PyPlot, Keras, PyTorch, OpenCV, Jupyter Notebook, Slurm, Git, Linux, Machine learning, Data science, Computer vision, Computer graphics