

Samuel Atkins

Software Engineer

sjatkins12@gmail.com | +33 0615863193
github.com/sjatkins12 | linkedin.com/in/sam-atkins

PROFILE

Passionate programmer with professional experience as an engineer solving complex real world problems by bringing products from ideas into designs into shipped solutions. My excitement for programming bleeds into my work and can be seen in the practices I adhere to and in the formation of my contributions. I take pride in my everyday work and I enjoy helping others do the same.

WORK EXPERIENCE

Ubble.ai

Full Stack Software Engineer

Paris, FR

11/2019 - 6/2020

- Startup Development Team of 5 Engineers on AWS-Postgres-Django-React Stack
- Contributed to building a Kubernetes ISTIO operator to manage releases in multiple deployed environments
- Built standalone micro services

Tesla Motors

Software Engineer Intern (Autopilot)

Palo Alto, CA

1/2019 - 6/2019

- Computer Vision Infrastructure and Frameworks team
- Worked across autopilot teams to support computer vision pipelines
- Automated local ECU testing to improve developers experience with ARM environment
- Improved CI build queues by implementing a workerpool with a callback interface

Optum / UnitedHealth Group

Technology Development Program

Boston, MA

5/2018 - 8/2018

- Developed machine learning model to classify tasks given to software teams
- Predicted the Time to Build of software features from historical development data

PROJECTS

N-Puzzle - C++

Designed and built a pathfinding algorithm to solve a classical N-Puzzle in the minimum number of moves. Algorithm also focused on minimizing auxiliary data, without redundancy.

IRC & FTP - C

Created ftp and irc server and client programs, with an asynchronous I/O server. I implemented my own class structure to simulate an object oriented language with C.

Corewar - C

Designed a Virtual Machine to act as a battlefield where multiple processes compete over a shared memory space.

Worker Server - Go

Designed a general http server that can distribute tasks. Easily extensible for any microservice architecture.

42sh - C

A complete shell that handles UI, syntax, pipelined execution and job control.

Malloc - C

Recoding of the classic functions malloc(), realloc(), calloc() and free(). Built with the focus on minimizing system calls.

EDUCATION

42 Silicon Valley

MS Software Engineering equivalent

Fremont, CA

9/2017 - 9/2018

- Starfleet curriculum; One year intensive (80+ hours per week) Software Engineering Program

- Programming languages: C; C++; Python

Pennsylvania State University

Bachelors of Science in Computer Science Curriculum

University Park, PA

9/2015 - 6/2017

Coursework:

- Data Structures and Algorithms
- Object Oriented Design
- Database Design
- Computer Architecture
- Discrete Mathematics

SKILLS

- C
- C++
- Go
- Python