Shreepa Parthaje

shreepa.parthaje@gmail.com | (571) 599-6003 | Chantilly, VA | linkedin.com/in/shreepa-parthaje-352241192/

EDUCATION

University of Virginia Charlottesville, VA | Expected Graduation 2024

B.S. in Computer Science with Data Science Minor

Thomas Jefferson HS for Science and Technology Alexandria, VA | August 2017 - May 2021

Relevant Coursework:

Computer Science: Artificial Intelligence, Data Structures and Algorithms, Computer Systems and Organization, Software Development Essentials ♦ Math: Probability, Multivariable Calculus, Ordinary Differential Equations, Linear Algebra, Advance Math Theory, Discrete Math ♦ Other: Physics I and II, Analog and Digital Electronics, Automation and Robotics, Chemistry

SKILLS and CERTIFICATIONS

- Languages: Python, C, C++, Java Script, TypeScript, MATLAB, Java, C#, Swift, Bash
- Software: Git, AWS, Unix, Docker, React, NodeJS, Tensorflow, NumPy, Keras, OpenCV, ROS, Unity, Django, Flask
- Certifications: AWS Developer (Associate), AWS Solution Architect (Associate), AWS Cloud Practitioner

EXPERIENCE

Amazon Web Services, *Cloud Computing Intern* Herndon, VA | June 2022 - September 2022

- Created web service for auditing architecture against AWS Well-Architected Framework for Fortune 500 workflows
- Built DevOps CI/CD process with CodePipeline to rapidly build, test, and deploy cloud resources for new features
- Secured API Gateway and Lambda architecture with CloudFront behaviors, lowering attack surface for DDoS threat
- Implemented DynamoDB storage solution with projection expressions on backend, reducing page latency by 75+%
- Worked backwards from customer needs & built tools in React to visualize + export data across all customer audits

Phone2Action, *Software Development Intern* Alexandria, VA | July 2019 - August 2019

- Built 3D C# VR application with Unity integrating voice activation with IBM Watson in a team of 4
- Developed internal API with Express.js server and Mongo DB database to handle backend calls for VR application
- Ran code review sessions and tracked development cycle through GitHub Issues and Kanban boards

RESEARCH AND TEAMS

Cavalier Autonomous Racing Team, *Perception Engineer* August 2022 - Present

- Implement perception algorithms to maintain position estimates of other vehicles on track and estimate trajectories
- Develop algorithm in Python to create augmented ROS bag files with simulated LiDAR noise and dropout

UVA Solar Car Team, *Software Operations Lead* April 2022 - Present

- Lead team of 13 software engineers to develop new distributed embedded software to manage communication via CAN between system, control regenerative breaking, handle data collection/storage, and more
- Assemble team of 8 software engineers to build Telemetry visualization tools using React, Python, and socket.io
- Overhaul embedded development environment creating standardized CMake build environments with Docker

UVA Solar Car Team, *Embedded Software Engineer* October 2021 - April 2022

- Developed C++ embedded systems on Mbed platform, leveraging CAN bus for communications between 4 boards
- Debugged several discrepancies in CAN bus messages between in-house boards and Raspberry Pi

TJREVERB (CubeSat), *Flight Software Lead* October 2019 - June 2021

- Spearheaded a team of 7 developers to build flight software to meet NASA's deadline for the 2U CubeSat
- Mentored underclassmen and taught development principles (e.g. continuous testing and version control)
- Integrated Python flight software with Iridium and SATT-4 radios and drafted CONOPS presented to NASA

PROJECTS

Ascent, A vertical take-off and landing aircraft | C++, MATLAB, Python, Fusion360, GitHub

- Simulated and tuned aircraft's PID control loops by modeling environmental torques and forces in Simulink
- Simulated and verified Kalman filter in Python Jupyter notebooks and extensive physical exterminations
- Wrote C++ navigation and control software with PlatformIO for a Teensy flight computer

MigosNET, Machine learning for lyric generation | Python, Keras, Tensorflow, BeautifulSoup

- Designed a LSTM neural network with Tensorflow/Keras that generates Migos (a music group) lyrics

Othello Artificial Intelligence | Python

- Created non-deterministic heuristic based artificial intelligence algorithm to play Othello moves within 5s
- Wrote game state calculations using binary operations on bit boards (bit arrays) for high speed calculations

Opener, A macOS keybinding utility | Swift, GitHub, Cocoa and Carbon API

- Released an open-source application implementing ideas from i3wm for macOS with 50+ downloads