Jaineel Upadhyay

+1-240-481-2669| ujaineel@gmail.com | linkedIn/jaineel-upadhyay | github/ujaineel

EDUCATION

BSc. Computer Science Baltimore, MD | May 2021

University of Maryland, Baltimore County

Coursework: Design & Analysis of Algorithms; Data Structures; Software Engineering; Artificial Intelligence; Intro. To Data Science; Intro. to Machine Learning; Principles of Operating Systems; Parallel and Distributed Processing

SKILLS

Proficient

Programming Languages: Java, Python, C++, C, HTML, JavaScript
Web Development: JavaScript, HTML, CSS
Languages: English, Hindi, Gujarati
Data Structures & Algorithm Analysis

Familiar

Programming Languages: CSS, PHP, Shell Scripting (Bash), MIPS Assembly
Framework/Libraries: TensorFlow, Keras, Sci-kit Learn, Pandas, Numpy, OpenMP Interface
Software Development: Agile, Waterfall, SDLC, UML Documentation
Databases: MySQL, MongoDB
Other Tech: Git, Heroku, Visual Studio Code

PROJECTS

DISTRIBUTED PROCESSING OF ROUND-ROBIN SIMULATION

Java, Eclipse, OOP, Parallelization

- Analyzed and compared performance and runtime for a parallel round-robin AI match simulations of different AI players for the game of Dominion, in a collaborative team of 3.
- Utilized Object Oriented Design, Monte-Carlo simulations and game state reductions to devise a fairly competitive and parallel processing AI player, whilst improving speedup of Monte-Carlo AI by at least 4X, whilst reducing running times by 80%.
- Tested game logic and AI logic and dealt with them, along with multithreading concerns, which led to a logically flawless game.

KERNEL INTER-PROCESS COMMUNICATION SYSTEM MECHANISM

C, Kernel, Linux, VirtualBox

- Implemented a communication mechanism for Debian GNU/Linux that allows inter-process communication freely with a mailbox-type communication.
- Wrote CRUD operations for the mailbox in Linux kernel-space with kernel programming, demonstrating Linux's openness.
- Evaluated mechanism logic with a user-space prototype of the communication system calls, to ensure kernel-programmed calls would execute as intended.

AMERICAN SIGN LANGUAGE TRANSLATOR

Python, TensorFlow, OpenCV, Neural Networks

- Modeled an American Sign Language (ASL) translator that translates the basic 24 ASL letter gestures and a few other poses using a Convolutional Neural Network (CNN) on top of a pre-trained VGG-16 model.
- Attempted live-video prediction using OpenCV to capture frames which are passed to the model after required preprocessing, opening the potential to communicate/translate on-the-spot.

DISCORD ASSIGNMENT SCHEDULER

HTML, CSS, JavaScript, Node.JS, Heroku

- Designed the front-end of the assignment scheduling web application using HTML, CSS, JavaScript to provide a simple yet efficient user interface, whilst also making it interactive for task/user object manipulation through GET, POST, DELETE, and PUT operations.
- Deployed the web application and the Discord bot on Heroku using Node.JS whilst also using JavaScript to authorize Discord user logins using OAuth2 authentication, to keep login secure and easier for the user.