Mustafa Arif

Website: mustafaarifio.web.app | Phone: (248) 525 1754 | Email: mustaf@umich.edu

Education

University of Michigan Ann Arbor | December 2023

- Bachelor of Science in Computer Science
- GPA: 3.89 / 4.0

Experience

FREELANCE FULL STACK WEB DEVELOPER | APRIL 2023 – PRESENT

- Conceptualized and designed single-page web applications using React to craft modern user interfaces with enhanced responsiveness and presentation for a bakery
- Engineered a robust RESTful API backend with Django, facilitating seamless data processing, fortified user authentication, and enhanced security measures
- Utilized Amazon's Relational Database to establish a scalable and multi-user-supported infrastructure, ensuring data integrity and robust secure management of user data
- Implemented Amazon's Route 53 service for domain name assignment, rigorously enforcing end-to-end data encryption via HTTPS to guarantee optimal data security
- Deployed application backends on AWS Elastic Beanstalk and frontends to AWS Amplify, utilizing EC2 instances and load balancers to efficiently handle user requests and prepare for future scalability requirements
- Leveraged Amazon's S3 Buckets for scalable data storage of static files, ensuring efficient and reliable data management

COMPREHENSIVE STUDIES PROGRAM PEER TUTOR | SEPTEMBER 2022 – PRESENT

- Guided students in Computer Science courses to master lecture concepts through weekly sessions
- Catered each lesson to individual student's needs and learning style

Projects

ELECTROMYOGRAPHY CLASSIFER PROGRAM | JUNE 2022 - PRESENT

- Collaborated with the University of Michigan's Neuroprosthetics team to design an interface translating electrical signals from a patient's residual limb into precise mechanical movements in a prosthetic arm using Arduino
- Currently developing a Python-based machine learning program, employing scikit-learn, to interface with electromyography sensors to pave the way for seamless integration with future Python-native hardware solutions
- Developed a Python script to visualize live EMG readings from the sensor for testing purposes

THREAD MANAGER LIBRARY | AUGUST 2023 – OCTOBER 2023

- Designed a thread manager library in C++ that would allow user programs to leverage multi-threaded processing for their applications

- Engineered a sophisticated thread-switching mechanism within library functions, optimizing the process to minimize CPU context switches
- Implemented advanced features for mutexes and condition variables, enabling user programs to effortlessly incorporate synchronization into their multi-threaded endeavors
- Efficiently managed memory by cleaning up the stack of each thread once execution was complete

VIRTUAL MEMORY MANAGER | OCTOBER 2023 – NOVEMBER 2023

- Designed and implemented a virtual memory manager in C++ that abstracted the constraints of physical memory from user programs
- Innovatively deferred disk writes and accesses, coupled with efficient physical memory allocation strategies, ensuring optimal system behavior without compromising correctness
- Successfully conserved RAM space by enabling the sharing of certain virtual pages on the same physical pages, fostering efficient memory utilization potentially across different programs

NETWORK FILE SYSTEM| NOVEMBER 2023 – DECEMBER 2023

- Designed and implemented a persistent network file system in C++ that could be used by multiple users simultaneously
- Leveraged advanced socket communication for seamless network interactions, enhancing the file system's responsiveness and reliability
- Implemented a concurrent design utilizing threads, enabling the file system to efficiently serve multiple user requests simultaneously
- Engineered a fault-tolerant file system that ensured persistent and valid disk states even in the face of potential crashes during user request processing

LANDMARK IMAGES CLASSIFICATION | OCTOBER 2023 – DECEMBER 2023

- Engineered a multi-layered convolutional neural network in PyTorch to accurately classify diverse landmark images and assign precise landmark labels
- Implemented the Grad-CAM algorithm for visualizing image regions contributing to classifications, enhancing analytical insights
- Successfully feature engineered data by converting images to their gray scale counterparts for better performance

MOVIE REVIEW RECOMMENDATION CLASSIFIER | AUGUST 2023 – OCTOBER 2023

- Engineered a Python machine learning algorithm, leveraging the scikit-learn library, to create a powerful linear classifier for classifying movie reviews as positive or negative
- Designed an initial feature vector based on word count for each word in movie reviews
- Employed advanced feature engineering techniques, including the exclusion of stop words, to enhance classifier performance and accuracy
- Implemented numpy and pandas libraries for effective visualization of the classifier's performance
- Conducted in-depth analysis to interpret and address potential biases stemming from the training data

WIKIPEDIA SEARCH ENGINE CLONE | MARCH 2023 – APRIL 2023

- Created a robust pipeline for computing the TF-IDF scores for each word within 3000+ Wikipedia articles
- Crafted a RESTful API using the Flask library in Python to proficiently manage search query requests,

facilitating the seamless communication of document matches ranked by relevance to users

- Engineered a dynamic server-side website enabling clients to search through Wikipedia pages, which seamlessly interacts with a REST API to deliver relevant user document results

MAP REDUCE LIBRARY | FEBUARY 2023 – MARCH 2023

- Engineered a versatile framework capable of leveraging multiple computers to execute map reduce tasks in Python
- Implemented TCP sockets to establish efficient communication channels between computers within the distributed system, optimizing data exchange and task coordination
- Established a fault-tolerant architecture by incorporating heartbeats via UDP sockets, ensuring system stability and resilience in the face of potential computer crashes

Technologies and Languages

- AWS
    - Elastic Beanstalk: Can deploy a backend application that supports auto-scaling
    - Amplify: Can deploy a frontend application that supports auto scaling and auto deployment with github pushes
    - S3 Buckets: Can use S3 Buckets to serve static files globally with low latency
    - Amazon Simple Email Service: Can send custom emails to users on behalf of an application
    - Amazon Relational Database: Design a database for an application to perform read/writes on
- CUDA
    - Can create kernels that execute 1000s of threads in parallel on NVIDIA GPUs
    - Nsight Compute: Can perform application performance analysis to determine various performance metrics of GPU during execution such as occupancy and branch divergence
- C++
    - Can work with language on high level with hash tables, command line arguments, c string manipulation, heaps, dynamic memory, linked lists, iterators, dynamic programming, and deques
- Python
    - Pandas: Can leverage library to visualize and analyze large sets of data
    - Scikit Learn: Can use library to learn linear classifiers for binary and multiclass datasets
    - Pytorch: Can design and test neural networks using library
    - Numpy: Can leverage library to perform efficient mathematical operations on arrays and matrices
- Linux
    - CLI: Can comfortably navigate and work on a linux machine from the CLI
    - Nano: Can edit files directly from command line using nano within CLI
    - Shell/Bash script: Can design scripts to automate executing, creating, deleting, and testing files
- React
    - Can create frontend applications that interact with REST API's
    - Can create a Single Page Application using React
    - Can efficiently manage data obtained from backend locally for seemingly instantaneous response times while reflecting user changes to backend and/or database
- Google Cloud
    - Firebase: Can deploy frontend applications with HTTPS and assign custom domain name
- Other: C, REST API, SQL, Flask