Mustafa Arif

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

EDUCATION

University of Michigan - Ann Arbor Aug 2020 - Dec 2023
Bachelor of Science in Computer Science
GPA: 3.90

Related Coursework:

- EECS 281: Data Structures and Algorithms (C++)
- EECS 370: Computer Organization (C, ARM)
- EECS 376: Foundations of Computer Science (P-NP Problems, Encryption, Deciders)
- EECS 485: Web Systems (Python, React, HTML/CSS, SQL)
- EECS 445: Introduction to Machine Learning (Python, Neural Networks, Classifiers)
- EECS 482: Introduction to Operating Systems (OS, Concurrency, Threads)
- EECS 493: User Interface Development (Javascript, HTML/CSS, Iterative Design)
- EECS 497: Human Centered Software Engineering (React, Python, SQL)
- MATH 214: Applied Linear Algebra (Matrix Multiplication, Least Squares Algorithm)

EXPERIENCE

Freelance Full-Stack Web Developer Apr 2023 - Present

- Conceptualized and designed single-page web applications using React to craft modern user interfaces with enhanced responsiveness and presentation to meet client specifications
- Successfully delivered a business website for a bakery and a portfolio website for a chemical factory
- 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

PROJECTS

Electromyography Classifier Program Jun 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

Thread Manager Library Sep 2023 - Oct 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 Oct 2023 - Nov 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 Nov 2023 - Dec 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

Movie Review Recommendation Classifier Sep 2023 - Oct 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

Convolution Neural Network for Landmark Images Classification Oct 2023 - Nov 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

Open Source Forum Backend May 2023 - July 2023

- Developed an open-source forum backend RESTful API using Django, facilitating efficient data management and user interaction
- Implemented seamless user registration and authentication through Google Cloud services, enhancing user onboarding and security while adhering to industry best practices
- Integrated Amazon SES (Simple Email Service) to automate and streamline account update notifications, ensuring timely and reliable communication with forum members
- Employed industry-standard coding practices, version control, and documentation to promote the project's accessibility and future contributions from the developer community

Map Reduce Library Mar 2023 - Apr 2023

- Engineered a versatile framework capable of leveraging multiple computers to execute a submitted map reduce task 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

Wikipedia Search Engine Clone Mar 2023 - Apr 2023

- Created a robust pipeline for computing the TF-IDF scores for each word within a series of Wikipedia articles by using natural language processing and data analysis
- 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

Instagram Clone Jan 2023 - Mar 2023

- Created an Instagram clone web application using React and Python
- Engineered REST API endpoints utilizing the Flask libray, ensuring the secure and efficient delivery of requested data in JSON format
- Designed and implemented React components with a focus on efficiency, minimizing the number of renders to optimize overall user interface performance
- Implemented robust security measures for sensitive user data, employing the AES 256 encryption and salts to safeguard sensitive user data in cases of cyberattacks
- Skillfully managed client sessions through the use of cookies, ensuring a secure and seamless user authentication process

Memory Cache Simulator Nov 2022 - Dec 2022

- Designed and implemented a memory cache simulator in C to simulate how memory is managed for a given hardware specification
- Evicted cache lines based on the Last-Recently-Used policy as a heuristic for optimal eviction
- Simulated efficient use of hardware by deferring memory accesses and writes unless necessary

LC2K Pipeline Simulator Nov 2022 - Dec 2022

- Orchestrated the simulation of pipeline behavior in C by executing instructions in cycles and providing live updates of memory and register changes directly to the console.
- Implemented advanced strategies to avoid data hazards that occur in pipelines through detection and forwarding while managing control hazards by predicting branch outcomes.

Technologies and Languages

- AWS
  - 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
  - Elastic Beanstalk: Can deploy a backend application that supports auto-scaling
  - Amplify: Can deploy a frontend application that supports auto scaling and auto updates with pushes to github
  - Route 53: Can obtain an SSL certificate for upgrading website protocol to HTTPS and acquire domain name
  - S3 Buckets: Can use S3 Buckets to serve static files globally with low latency
- C++
  - Can work at a high level with hash tables, command line arguments, C-string manipulation, implementation of heaps, dynamic memory, linked lists, iterators, dynamic programming, and stack/queues
- Python
  - Scikit Learn: Can use library to learn linear classifiers for binary and multiclass datasets
  - Django: Can develop a RESTful API using Django for applications
  - Pytorch: Can design and test neural networks using pytorch library
  - Pandas: Can leverage library to visualize and analyze large sets of data
  - Numpy: Can use numpy for performing efficient mathematical operations on arrays and matrices
- 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
- Other Skills
    - Linux: Can use linux shell and commands for project development eg. github commands
    - Shell/Bash script: Can create scripts that automate testing and database creation/deletion
    - Bootstrap: Can use CSS library to create modern looking web applications
    - Firebase: Can deploy a frontend application to Firebase with HTTPS and custom domain name