Jacob Rogers

801-870-8053 | staringatnebulas@gmail.com | linkedin.com/in/Jacob | github.com/Jacob | My Website

OBJECTIVE

To advance my career with a leading organization that will utilize my extensive software engineering knowledge while also allowing me to develop further as a professional. Proven talent for aligning project objectives with established and emerging technology paradigms to achieve optimal results with minimal challenges. Growth-focused leader with expertise in utilizing various languages (Python, Java, C++, C#, JavaScript) for scripting, programming, and backend/frontend development. Strong proficiencies in data structures, primarily object-oriented best practices, implementation of core algorithms, and testing through JUnit. Exceptional professional with keen interpersonal, communications, and task prioritization expertise.

EDUCATION

University of Utah   Salt Lake City, UT
Master of Science (M.Sc) in Computer Science   *2023*

University of Utah   Salt Lake City, UT
Bachelor of Science (B.Sc) in Physics, Minor in Computer Science   *2021*

Salt Lake Community College   Salt Lake City, UT
Associate of Science (A.Sc) in Physics graduated with High Honors   *2018*

RELATED EXPERIENCE

Undergraduate Researcher   Jan. 2018 – May. 2018
Salt Lake Community College   *Salt Lake City, UT*

- Studied, analyzed, and interpreted complex astronomical spectroscopic measurements for the contact binary HV Uma.
- Detected, isolated, and filtered noisy data, originated extrapolation techniques, organized temporal patterns, and visualized the results utilizing multiple components of Python, including NumPy, Matplotlib, Seaborn, and SciKit-Learn.
- Applied gradient descent methods to formulate estimates of the epoch of binary star systems and employed clustering techniques to categorize stars based on their attributes, such as: observed luminosity, mass and period of rotation.
- Delivered a dynamic presentation titled "HV Uma - A Contact Binary" to a large audience at the Salt Lake City Community College Symposium.

Volunteer STEM Tutor   Feb. 2017 – May 2018
Salt Lake Community College   *Salt Lake City, UT*

- Tutored: Algebra, Trigonometry, Calculus I, II, III, Physics for Scientists and Engineers I and II, and CS 1410 (Object-Oriented Programming) at the Dumke Center for STEM Learning.
- Collaborated with a team of tutors to facilitate problem-solving in complex STEM-related topics through effective communication and support for students in need.

PROJECTS

SpamScanner | *Python, TensorFlow, Kivy, Buildozer; GitHub.com/SpamScanner*   April 2023 – Present

- Designed and developed a unique Android application that utilizes machine learning techniques to detect potential fraudulent (ham or spam) SMS while leading a highly knowledgeable team.
- Compiled, systematized, and cleaned a large-scale dataset for both training and testing purposes. Integrated natural language processing methods, such as word stemming and lemmatization, as well as advanced techniques including Term Frequency-Inverse Document Frequency (TF-IDF) transformation and component scaling, to preprocess and analyze the data effectively.
- Generated and implemented a deep neural network for training a spam classifier through the TensorFlow framework, achieving an impressive 96.9% detection accuracy.
- Utilized the Kivy framework to create an internal connection between the spam classifier and a user-friendly UI.
- Packaged and deployed the Android application using the Buildozer framework.

Tank Wars | *C#, .NET, GitHub; gitfront.io/Tank-Wars*   April 2021 – May 2021
- Played a key role in the development of an online multiplayer tank battle game using C# and .NET, with accountability for full-stack deployment, server-side development, and client-side development.
- Tested and validated thread safety in a multi-threaded application by implementing client-server communication and rendering for game data and drawing on object-oriented programming principles in addition to separating the model, view, and controller for each system component.
- Participated actively in the overall design and gameplay mechanics for a clean codebase and smooth user experience.

Tic Tac Toe | *Java, Swing, AWT, Eclipse; GitHub.com/TicTacToe*   April 2017 – May 2017
- Leveraged Java programming language to architect and develop a scalable tic-tac-toe game application.
- Implemented a modular and reusable codebase using Object-Oriented Programming (OOP) principles, promoting code organization and maintainability.
- Employed data structures and algorithms techniques to facilitate game functionality, including win condition checks and player turn management.
- Applied error handling mechanisms to ensure robustness and reliability in handling user input and edge cases.

Other Project Accomplishments
- Enhanced a concurrent high-performance database system in C++ by integrating robust write-ahead logging functionality, ensuring seamless data integrity and fault tolerance across concurrent operations.
- Designed and implemented High-Performance Computing tasks in C and C++, utilizing advanced parallelization techniques, CUDA for matrix operations, and multithreading to optimize memory usage and enhance execution efficiency.
- Developed a comprehensive machine learning library in Python, encompassing regression, decision trees, ensemble learning methods, Support Vector Machines (SVMs), and neural networks. Conducted rigorous testing on each algorithm using real-world datasets to evaluate performance and accuracy.
- Engineered advanced AI algorithms for PacMan in Python, implementing a range of search algorithms, reinforcement learning techniques, behavioral algorithms, and Bayesian networks. Achieved enhanced maze navigation, effective adversary defeat strategies, and seamless game continuation, contributing to an enriched gaming experience.
- Designed and implemented secure Java network protocols utilizing the Security API for encryption, authentication, and seamless data transmission over TCP/IP connections. Highlights include a Substitution-Permutation Network, Diffie-Hellman key exchange, Needham-Schroeder Authentication, and SSL3 protocol recreation, ensuring successful handshakes, authentication, encryption, and uninterrupted data flow between sockets.
- Developed a Linux OS clone using C and ARM assembly language, leveraging QEMU for emulation. Key contributions encompassed designing command line interfaces, implementing memory management via paging, and creating a bootstrap loader for successful OS booting on real devices.

TECHNICAL SKILLS

Languages: Java, Python, C, C#, C++, SQL, JavaScript, HTML, CSS
Frameworks: React, .NET, JUnit, Kivy, Buildozer, TensorFlow
Developer Tools: GitHub, Linux, Windows, REST API, Docker, VS Code, Visual Studio, GDB, Eclipse, AWS EC2
Libraries: pandas, NumPy, Matplotlib, SciKit-Learn, Java SE Security