|  |  |  |  |
| --- | --- | --- | --- |
| **SINA MALEK** | | | |
| U.S Citizen – Secret Clearance – sinamalek.com – [maleksina1998@gmail.com](mailto:maleksina1998@gmail.com) – (602) 471 - 7084 | | | |
| **SUMMARY** | | | |
| Experienced software engineer in need of a full-time direct hire position with a dynamic company. Resolute collaborator with strong ethics who can work independently or with a team. Has experience in test software and architecture, clean up, ESMs, and producing standardizations for future software developments. | | | |
| **EDUCATION** | | | |
| Bachelor of Science in Computer Systems Engineering | | | **May 2020** |
|  | Arizona State University, Ira A. Fulton School of Engineering – Tempe, AZ | | **GPA: 3.50** |
| **SKILLS** | | | |
|  | **Languages:** | Python, C/C++, CMake, MATLAB, SCPI, YAML, Markdown, Doxygen | |
|  | **IDEs:** | Visual Studios, VSCode, PyCharm, Notepad++ | |
|  | **Tools:** | Workbench, BeyondCompare, NiMax | |
|  | **Microsoft Office:** | Excel, PowerPoint, OneNote, Outlook, Word | |
| **WORK EXPERIENCE** | | | |
| **Software Engineer I – Raytheon Missile & Defense** | | | **May 2021 – Present** |
|  | **TECHNICAL PROJECTS** | | |
|  | **Factory BIT – Update Common Software & Factory Tests** | | **November 2022 – Present** |
|  | Set forth documentation standardization process for each test created. | | |
|  | Using Doxygen in build process to document code, ensuring readability and organization for onboarding engineers. Saved the company about three-man hours a day for senior engineers to assist with onboarding process. | | |
|  |  | | |
|  | **Pilot Test Line – Factory Tests** | | **August 2022 – November 2022** |
|  | Modified and refactored existing test software to far better standard. Saved the company time by a factor of two and about $1 million when delivering hardware stacks to programs. | | |
|  | Stood up all test equipment. Configured oscilloscopes, signal generators, signal analyzers, power supplies, FSWP, RF switch boxes to be compatible with internal software and able to communicate using SCPI and Python scripts to programs specifications. | | |
|  | Established backlog in Azure DevOps, functioned as Scrum Master for product | | |
|  | Improved current UI launcher for Pilot Test Line to be customizable for each to test for the end user; also, to ensure quick turn around when adding new capabilities. | | |
|  | Refactored and cleaned up software to improve readability and functionality. Removed scripts not being utilized and reduced hard paths in scripts. | | |
|  | Multiple scripts created to automate communication and startup of test equipment using SCPI commands using PyVisa. | | |
|  | Collaborated with Pilot Line engineers to ensure test outputs are expected based off certain equipment. | | |
|  |  | | |
|  | **Ultra – Software Maintenance & Refactoring** | | **January 2022 – August 2022** |
|  | Using Azure DevOps, YAML and Python scripts, created automation process of BIT testing to ensure health of hardware. Hardware is powered on, DKMs are loaded on for boot, runs a list of BIT tests and outputs results into a database. | | |
|  | Implemented different kinds of modules specific for program space, specified for RF and BIT testing. | | |
|  | Updated the health of the embedded code by adding safety conditions to multiple functions and utilizations. | | |
|  | From legacy COE made refinements to seeker software to improve message communication speed. | | |
|  | Created stress tests for HSSB message sending between modules on hardware and emulation. | | |
|  | Set standard when creating demo tests for testing certain aspects of the hardware. Structure of script and architecture of sending and receiving messages to determine if test pass or fail. | | |
|  | Changed the domain of messages being sent and received to match current standard. | | |
|  | Improved the current EQT test software to include specified tests and customizable. | | |
|  |  | | |
|  | **IRAD – Update & Clean-Up Common Software** | | **August 2021 – January 2022** |
|  | Updated software version through merge techniques and ensuring the entirety of the software remained functional. | | |
|  | Created CMake functions to be improve the build process of the common software. | | |
|  | Brought about expected standardization of code implementation; refactored existing code to match standard. | | |
|  | Added Windows DLL’s to improve linking of scripts and dependencies upon each other. | | |
|  |  | | |
|  | **IRAD - AI/ML Classifiers and Signal Processing** | | **May 2021 – August 2021** |
|  | Trained basic regression neural network with Henze-Penrose Divergence loss. Converted existing MATLAB scripts to Python and used existing NumPy data sets to determine loss. | | |
|  | Manipulated large NumPy datasets to smaller sizes so computation may be possible. | | |
|  | Utilized MST using either the Prim or Dijkstra’s algorithm. | | |
|  | Tests results were that it was too complicated of a task for the neural network to learn, was not used. | | |
|  |  | | |
| **IT Service Desk Intern – Amkor Technology Inc.** | | | **May 2019 – October 2019** |
|  | Project leader of Windows 7 to Windows 10 conversion for Tempe location. | | |
|  | Collaborated with various departments to utilize Clonezilla to create standard computer image. | | |
|  | Assisted in IT services to increase company output by providing technology support fort all users; ensure users workstations function to company standard, and have thoroughly documented results for future reference | | |
|  | Implemented configuration and deployment of new equipment such as laptops, desktops, and phones. | | |
| **ACADEMIC PROJECTS** | | | |
| **A.I Stanford Pac-Man Project** | | | **August 2019 – December 2019** |
|  | Determine the optimal criteria and algorithms for different simulations in Python, using TensorFlow. | | |
|  | Uses of state space search, heuristic search, games, knowledge representation techniques, expert systems, and automated reasoning, Manhattan and Euclidean distances. | | |
|  | Implemented Bayesian networks and learned fundamentals of neural networks. | | |
| **Project Member – Cyber-Physical System: Autonomous RC Car** | | | **January 2019 – May 2019** |
|  | Received ***third*** place in autonomous car race competition against fifteen other teams. | | |
|  | Conceptualized cyber-physical system design using real-time software, control theory, and system components. | | |
|  | Arduino based design, using an IMU, LIDAR, GPU and motors on a radio-controlled car. | | |
|  | Utilized fundamental concepts of digital system design for embedded system applications. | | |