# Applicant Questionnaire: Software Developer

## Instructions

For some of our positions, we refer to portfolios or code samples when determining if interested people will fit with our team. We have found that some people's previous experience doesn't lend itself towards the creation of an online presence, so we have prepared a brief questionnaire.

Please answer the following questions to the best of your abilities. Good luck; may the force be with you!

|  |  |
| --- | --- |
| **Applicant Name:** | Sourabh Arora |

|  |
| --- |
| **Our main programming languages at Clearpath are Python and C++. What factors would you consider when choosing between these for a given project?** |
| Response:  We can create ROS node in either C++ or Python. Preference depends on familiarity and libraries required for a package and purpose.  For a loop thread requiring heavy computation, C++ is preferably a better option. C++ also gives developer to do memory management and is type safe.  Python whereas is having simpler syntax with higher memory usage and automatic garbage collection. I have recently started using Python in ROS for updating the map based on the type of localization environment to choose. |

|  |
| --- |
| **What is the most interesting problem you had to resolve through specialized tools, whether in-house or commercial? How were they used to explore and resolve the issues? Tell your debugging war story.** |
| Response:  For the cleaning robot, had used **rqt\_reconfigure** GUI Interface to dynamically configure ROS parameters while the nodes are running on sim for safety, diagnostics and other states, like Estop or squeegee detached, dirty tank full etc. and via **bag** files.  Also used tools like **Flatland** plugin and **RViz** for robot running in simulation.  I have used **Deleaker** to see the hit counts on seeing if there is memory leak like **valgrind** in Linux. While in the stress test for FLISR application, the base Distribution management system’s data was imported to Fault Location Data table that was trying to access GIS Spatial data, and memory deallocation for switch status for all the switches under a given transformer wasn’t done properly causing the application to crash. So, used deleaker to solve memory leak issue.  Another normal issue is resolved usually using breakpoints, watchpoints and **logs**. |

## Team Project:

Please focus on a recent project you completed as part of a larger team.

|  |
| --- |
| **Please describe your contributions to the project. What components were you personally responsible for, and what were their interfaces to the other components? How did those interfaces get defined?** |
| Response:  Avidbots  One of the first task I got was to merge the existing parallel development branch for GUI redevelopment interface to new pre-release branch which eventually became a master branch. It involved me navigating through different ros nodes and interacting with different teams for resolving merge conflicts.  Also, created new sub module interfaces for cleaning robot sub system Light Subsystem, Cleaning Subsystem.  Thales Group (R&D Division)  Train Autonomous Platform, I have developed components for Cross Comparing the Data from Instances, Pre Processing the data for latitudinal and longitudinal velocity, hit counts for validating the data, data starvation etc. Those were independent classes interacting with each other via Interface class to interact with other components. Based on the features required, I had in process to load all the components in the Factory class to collect stationary data.  Survalent Corporation  Fault location class in the FLISR application was bringing in the data form spatially distributed geographical data of electrical devices from distribution transformer level downstream to feeder level, where electrical calculation was done to evaluate the voltage, current etc to see the fault and isolate the area of fault |

|  |
| --- |
| **What development environment and programming language did you use? What costs and benefits did this choice have to the overall project outcome?** |
| Response:  Visual Studio as Survalent’s SCADA on Windows platform.  VSCode and CLion for ROS packages, Train Autonomous Platform’s component development & GTest code coverage.  Programming Language- C++, Python (Basic), Qt/QML, SQL |

|  |
| --- |
| **What steps did you take to future-proof your code? Were there areas where you (and the team) chose specifically not to (or not to attempt to) future-proof? What was the reasoning in each case? Is there an example where this strategy paid off, or you got bitten?** |
| Response:  I have done Unit testing for the code coverage and then the component is going through performance test that includes stress test to future proof the code.  Testing the application on simulation locally to future proof the code. |

## Individual Project

Please focus on a project completed individually (or mostly individually). Side projects, open source contributions, stuff for school, and stuff for previous employers are all valid.

|  |
| --- |
| **What was the project? What was the motivation and what did you accomplish? How did it turn out differently from your plan going in?** |
| Response:  At Avidbots -  The new development effort I worked on individually is to re-develop the sub module of Light Sub System for the cleaning Neo2 robot. It included first working on a spike to purpose the design document of the Light Subs system, which I purposed to follow Factory Method design pattern.  I started with first supporting the change with using the existing Neo 2 robots to support backward compatibility. The same approach is been used to replicate it for other sub systems like cleaning , drive sub systems etc.  The motivation for this redevelopment effort for new sub module on the Autonomy stack is for quicker launch of new products, for example a small cleaning robot with each light on different PCB board provided by the Firmware team. We would have a modular approach to support each light interface using a single light manager object to update left light, right lights, brake light etc. Similarly worked on other sub systems of cleaning robot as well.  At Thales Group –  In my current role for Train Autonomous Platform, being an R&D project. Since it is being built from scratch and had to develop and interface along with feature component development. On how we create an interface to basically inter exchange the data between the component was important. Initially everything was via getting communicated via message structure making it heavy computation and then on consultation with architect, I had made the interface to directly collect the state of component for small messages instead of using same large structure for component inter communication. |

|  |
| --- |
| **What is something that you know now that could have helped you with this project if you had known it at the time? Please focus on technical knowledge rather than personal development (eg, time management).** |
| Response:  At Thales- I was more focused on functionality implementation and later to deal with AUTOSAR and MISRA compliance for the code structure. So, once the first component was built I had to restructure it and some code structure do need code compliance exemption based on scenario and component functionality to make it more logical to implement that we realized going forward. |

|  |
| --- |
| **How did you perform system testing and verification?** |
| Response:  Unit testing - functional testing, performance testing,  Testing robotics application on sim using plugins like rqt\_reconfigure, flatland plugin/ RViz on the robots.  Also applying my test build on the physical robot as well. |

## Lightning Round

|  |
| --- |
| **What is your favorite network protocol debugging tool and why? If you don’t have one, what might you try out next? Where would using one be effective in troubleshooting a problem with an application or service?** |
| Response:  I have used Wireshark to debug the networking to trace IP of the field devices on field communicating with SCADA with different networking protocols like DNP3.0, IEC61830, TCP/IP, EthernetIP  The next I would want to try out is Socat, which is the virtual port that is used to pass serial data for emulating the robot sub system states like for robot’s light sub system, cleaning sub system etc. of robot. This can be done by creating a ROS serial node using the serial package. |

|  |
| --- |
| **Have you ever reverse engineered a protocol or some other aspect of a piece of unknown hardware or software? What was your process and what did you discover?** |
| Response:  I have introduced the staleness faults to test robot’s behavior for velocity or data integrity feedback.  Also used mocked data and executed a section of code in software testing to test the interface of each component before adding new component. This way, it helped me optimize the code stack and gave me better understanding of architecture. |

|  |
| --- |
| **Have you ever successfully evangelized a tool or practice to a team you were working on?** |
| Response:  Automated the SQL query to get the status of switches downstream of the transformer from GIS data table that had one to one mapping with DMS table. Some of the switch and other transformer data couldn’t get updated. So created the Job to get those switches and other columns status to get updated automatically. |

|  |
| --- |
| **What is your favorite robot, real or fictional, and in one sentence tell us why?** |
| Response:  Best fictional robot is Ironman that does everything and the reason why I started getting interested in robotics. In reality, I have worked on cleaning robot running on ROS1 and ABB robots integrating with PLC for material handling applications. |