

Software Industry Experience for High School Students

Dr. Massood Towhidnejad, Embry-Riddle Aeronautical University, Daytona Beach

Massood Towhidnejad is Director of NextGeneration ERAU Applied Research (NEAR) laboratory, and Professor of Software Engineering in the department of Electrical, Computer, Software, and Systems Engineering at Embry-Riddle Aeronautical University. His research interest includes; Software Engineering, Software Quality Assurance and Testing, Autonomous Systems, and Air Traffic Management (NextGen).

In addition to his university position, he has served as Visiting Research Associate at the Federal Aviation Administration, Faculty Fellow at NASA Goddard Flight Research Center, and Software Quality Assurance Manager at Carrier Corporations.

He is senior member of IEEE.

Ms. Alexandria Spradlin, Embry-Riddle Aeronautical University

Alexandria Spradlin is attending ERAU as a graduate student studying software engineering. She is also employed as a research assistant at the Next-Generation ERAU Applied Research Lab. She hopes to continue her education after graduation from Embry-Riddle to earn a PhD in bioinformatics. She also has a keen interest in STEM education, hoping to assist in spreading her passion for mathematics and the sciences to the next generation.

Mr. Thomas Rogers Bassa, Embry-Riddle Aeronautical University

Thomas is a software engineering student at ERAU, pursuing both his Bachelor's and Master's degrees simultaneously. He enjoys helping others fix software bugs and teaching small tricks to make one's work easier.

Software Industry Experience for High School Students

Abstract— High school students typically do not have an understanding of what it takes to successfully transition between the high school and professional world. This paper describes how a group of ten high school students participated in a nine-week software development internship program. The interns worked on developing a semi-autonomous vehicle to be controlled remotely by a ground station. They went through the full development life-cycle using formal agile process. The participants gained experience in project management, and how to negotiate with the customer to reduce the complexity of the project to an achievable scope, given available resources (time, personnel, and cost).

I. INTRODUCTION

This paper describes one of our activities under the INSPRE-CT (Computational Thinking) project that has been funded by the National Science Foundation (NSF) CPATH-2 program NSF-DUE-0939028. The INSPIRE-CT project explored vertical integration of student teams to improve student learning and raise student interest in computing. Throughout the project, we looked for opportunities where we integrated professional, graduate, undergraduate, high school and middle school students to work on common project, where more experienced people leads, mentor and/or interact with less experienced people. During the last phase of the project we exposed high school students to real-world projects, as they integrated with professional software developers at the Nextgeneration Applied Research (NEAR) Laboratory (www.near.aero) at Embry-Riddle Aeronautical University (ERAU), Daytona Beach, Florida. The NEAR lab has eleven full time staffs with two holding PhD degree, eight holding master and one holding bachelor degree in computing and engineering fields. The staff have experience in software development ranging between three to over twenty five years. The NEAR lab staffs are almost fully funded by the external grants and contracts form government and industries. The projects are funded by organizations such as National Science Foundations, Federal Aviation Administrations (FAA), National Oceanic and Atmospheric Administration (NOAA), Boeing Corporations, New York Port Authority, etc.

During the last phase of the project, we announced an internship opportunity at the NEAR lab to multiple high schools across the Volusia and Seminole county (Florida). In order to apply, in addition to completing the application form, each applicant should submit a statement of the interest, specifically addressing why they should be accepted in the program, and also provide information on what STEM related courses they have completed. We received over seventy applications from students, and some of these students were living about 60 miles away from the lab. Total of thirty applicant were selected for a face-to-face interview. Some of the selection criteria used at this stage was;

- Did the candidate has any exposure to programming, either through course work, and/or out of the classroom activities
- Did the student had any work experience, either paid or unpaid (we considered volunteer work, that is highly encouraged by the state of Florida as part of the university scholarship qualification as work experience)
- Did the student has a way to get to and from their locations to the lab

Twenty five students were selected to participate during the fall 2014, and spring 2015 semester. During this time, students spend four hour in the lab every Wednesday (Wednesday is early dismissal day). While in the lab, students were given an opportunity to work one-on-one with the NEAR lab personnel on specific projects. They were also given an opportunity to shadow them while the staff were working on one of the funded projects that was underway at the lab. During the fall and spring semester, the NEAR lab staffs

monitor the technical and professional performance of these twenty five students, and at the completion of the spring semester, fifteen students were recommended by their mentor as possible candidates for the ten internship position during the summer 2015. These students were contacted for additional interview and their availability during the summer. Finally, ten students were selected to participate in a nine weeks paid internship at the lab. The following represent the demography of the ten interns;

- The interns' age ranged from fifteen to seventeen, and their grades ranged from tenth to twelfth. Nine of the interns were male.
- With the exception of one intern, they had all taken an advanced placement computer programming class prior to participating in the internship. The majority of the participants had also taken advanced mathematics courses and some have completed Calculus.
- Majority of the interns attended the same high school. Consequently, most of the participants had experience working with one another.
- Most students did not have an experience to work on team project, and the ones who have such experience, their teams were small (2-3 people)

During the internship, the interns worked under the direct supervision of three NEAR lab staffs and two graduate students. Each week, students met three days (Monday, Tuesday and Wednesday) on the campus, working 8 hours a day, and were encouraged to work at least another day from home.

The remainder of this paper describes the project, and the development process used during the internship.

II. PROJECT OVERVIEW

The project customer was a NEAR lab employee, which provided a half page description of what is his expectation for the product at the completion of the internship. According to the need statement, the final product would be a semi-autonomous vehicle to be controllable remotely from a ground station. The vehicle would also have a fully-autonomous mode in which it would navigate between waypoints by traveling over sidewalks throughout the campus. These requirements were changed intentionally over the course of development to expose the students to one of the challenges of software development in the real world.

The system requirements were drafted during the first week of the internship. The interns met with the customer with whom they discussed the desired product. The customer's description was vague, and he sometimes contradicted himself to simulate difficulties experienced when communicating with customers. When the interns mentioned an interesting feature not originally envisioned, the customer would sometimes latch on to the idea and insist that it be included even when it was unrealistic. Several of the desired functionalities were purposefully beyond the scope of the project given the allotted time frame and the interns' knowledge at that time. Again, this was designed as part of the program, in order to demonstrate what would happen in the real world, and one should be careful about the project scope and team capabilities and available resources. It was made clear that the interns will have access to the customer throughout the internship, therefore they have the ability to fine tune the project requirement beyond the first week.

A. *Project Requirements*

At the beginning of the internship, the envisioned product was a semi-autonomous vehicle that could be driven remotely. To be controllable remotely, it was required that the vehicle stream live high definition (HD) video at all times. The vehicle would also have a fully-autonomous mode in which it would travel along sidewalks to navigate between waypoints. While in the autonomous mode, the vehicle would avoid dynamic obstacles. To protect the vehicle from rain and water accumulation on the ground, it was required that the vehicle be IP67 compliant ^[1], meaning that the vehicle be dust-tight and immersible in 1~m of water.

The vehicle also need be able of carrying a payload of at least 3lbs. The vehicle also shall have a light and sound source to warn pedestrians of its presence.

In addition, the user shall be able to remotely drive the vehicle via ground station, and be able to set waypoints on a graphical user interface and switch the vehicle between fully-autonomous and manual mode. The video feed shall be viewable from the ground station at 1080p with a minimum of 30 fps. The location of the vehicle, the charge on the battery, and the vehicle's speed shall be observable from the ground station.

After the initial requirement elicitation, the NEAR lab staffs and the graduate assistants, questioned the intern about their level of confidence on possibility of delivering the complete product. Through several hints and questions, and good period of discussion/arguments between themselves, the interns started to realize that they would be unable to deliver on all of the promised functionality within the allotted time frame. They started identifying the functionalities that they thought are hard to deliver, and arranged a meeting with the customer to negotiate those requirements to be considered out of scope. Since this was expected, after some discussion as to why they want to remove those requirements (they have to have a good justification), a large number of initial requirements were removed.

B. Development Process

During the last day of the week one, students were introduced to an agile process that is used at the lab, and were asked to use this process for their project. A light weight agile methodology suited for small development team called Crystal Clear^[2,3,4] was chosen for the development process. With this process, the internship period was divided into four two-week iteration called sprints. One of the major component of the Crystal Clear is the development of the “walking skeleton” where at the end of each sprint the development team would have an end-to end product to be demonstrated to customer, where initial walking skeleton has a very limited functionality, and each proceeding iteration, additional functionalities will be presented. This will allow the customer to see the team’s progress, and also give the team and customer an opportunity to make any course correction that is deemed necessary. Also, as it was mentioned previously, some of the requirements that was identified during the first week was intentionally beyond the scope of the project, and some was beyond the capabilities of the development team. Furthermore, due to the lack of students working on such project, and also high enthusiasm of them (perhaps as a result of watching too many movies), there were some requirements included (added by students through Q&A with the customer), that was not technically feasible even by very experience hardware and/or software professional. The bi-weekly meeting with the customer provided the interns an opportunity to clarify their understanding of certain requirements and also negotiate with the customer over requirements that they had discovered were unattainable with the available resources. Also, during these bi-weekly meeting, the customer introduced some additional requirements either as a replacement for the ones that the interns wanted to eliminate, or just the fact that “he has forgotten to mentioned them during the initial requirement phase”. Again, this was intentionally incorporated to the project, to reflect what can happen in the real world, and also introduce the concept of change control.

C. Team Organization

Given the the nature of the project, the development team required expertise in both hardware and software. There were number of students who had higher level of interest in one of these areas, and the rest did not have a strong preference, this allowed us to divide the interns to two groups of five. The two groups remained fluid throughout development, but the participants tended to remain with those they had grouped with on the first day. During the first three weeks (completion of iteration one) of the project, the two groups worked in the same room, making communication amongst them efficient. At the start of the second iteration, the groups were separated. There was two reasons behind this decision,

- We wanted to simulate the working environment, where projects is distributed across the nation and/or the world.
- Due to the fact that the majority of the hardware related equipment was located in the lab, and the software team really did not need any access to the lab equipment other than during the integration, we moved the software group to a classroom (located at the second floor of the building).

When the two groups were separated, their means of communication changed. They decided to use an instant messenger to communicate throughout the workday but often found it difficult to understand one another. To remedy the problem, the two groups set designated times for video chat meetings before lunch and before each workday ended. After encountering some difficulties with this method, a leader was elected in each group and the leaders were allowed to meet in person at the end of each workday to discuss the progress made that-day and plans for the following day. The forced separation also introduced technical difficulties. Although the requirement for the project was identified and documented, some of the requirements where interpreted differently by the two groups, and when the requirement was critical for integration, the teams had a hard time developing integrating all the components of the system at the end of the second iterations. All these pointed to the importance of the documentation, and identification of clear communication path. At the beginning of the third iteration, we collocated the two teams again.

D. Project Resources

This project was funded by the National Science Foundation (NSF), and NEAR lab. The NSF funding covered the effort for project PI, two graduate students, intern's salary, and some of the parts that were used for the project. The NEAR lab funding supported two additional staff, in addition to providing access to the facility and major equipment's (i.e., high quality 3-D printer, electronic bench marks, etc.). The additional staffs that was funded by the lab served as the subject matter expert (SME), and have significant experience in the development of the hardware and software for unmanned systems, this experience was critical for interns. Prior experience on similar projects gave these staffs an extensive knowledge on possible solutions to almost all challenges faced by the interns. However, in order to control the intern access to these SMEs; when a challenged was faced by the interns that cannot be overcome by their peer, they involved their mentors (graduate students and then the PI) in a finding solutions to the challenge, if this was not possible, then the mentors will discuss the issue with the SMEs, and they either communicate the SME's solutions and/or recommendations back to the interns, or arranged for a meeting between the interns and the SME to discuss the problem and the potential solution to the problem. There were two reasons for this communication process, these are;

- We did not want to allow the interns to have easy access to the SME, therefore for every obstacle/challenge, they have to first try to sort it out themselves, and then get help from their mentors, where they would be given hints and even challenges to see if the intern can solve the problem themselves.

- The SMEs were already involved in other funded projects, with tight deliverable schedule, therefore we did not want to interrupt their activities on other projects with continuous interruption of the interns for every little obstacle.

In addition to project personnel discussed above, there were number of products and other resources that was available to the students to complete their project. The following are some of resources that was used for the project.

- LulzBot TAZ 5 3D printer, and SOLIDWORKS D CAD software
- Two RC vehicle chassis, including wheels and Lithium polymer (LiPo) batteries and charger
- Three Raspberry Pi B+'s
- A Ubiquiti Rocket M2
- LS20031 GPS Receiver
- Electronics (PCB boards, wire of various gauges, a soldering iron, and miscellaneous electrical prototyping materials
- Etc.

III. PROJECT IMPLEMENTATION

It was evident from the requirements that the two most significant components of the system would be a vehicle to drive and a ground station to control the vehicle. The software developed for this project was split into two major functional subgroups: the software present on the vehicle's Raspberry Pi B+ and the software used on the ground station. The vehicle code was written in Python and consisted of a module to handle the communications between the ground station and vehicle, and a module to control the vehicle. There was also a need for a module to handle network changes that would determine the IP address of the Ubiquiti Rocket M2. The ground station code consisted of a website written using HTML and JavaScript and a Python script to take input from a USB joystick and communicate the commands to the vehicle.

One of the most crucial aspects of this project was communication between the vehicle and ground station. A great deal of information would need to be exchanged between the two, and finding a way to do so in real time was a significant challenge in the early phases of the project. Ultimately, with help from the SMEs, the interns decided to use the Web Application Messaging Protocol (WAMP). WAMP provides facilities for remote procedure calls (RPCs) and the Publish and Subscribe communication model (PubSub) ^[5].

In order to implement WAMP in the project's code, a library named Autobahn was utilized ^[6]. This library is freely available under an MIT license, and it supports a number of programming languages, including Python and JavaScript-- the two main languages used in the project.

The only significant downside to using WAMP was that it requires the use of a centralized server to handle routing communications ^[7]. While using a centralized server helped to avoid coupling components, it introduced an annoyance in both finding software to fit this purpose and in deciding what machine to run it on. The router software was easy to find; Crossbar.io was found to suit this purpose well and is developed by the same entity as Autobahn ^[8].

The vehicle was built primarily with parts for RC cars and had 3D printed struts. The final vehicle was equipped with a Ubiquiti AIRCAM mounted to the front strut for live video communication with the ground station, a Ubiquiti Rocket M2 to provide Internet access, and a Raspberry Pi B+ which hosted the vehicle software for command and control, and communication with the ground station.

As initially mentioned, we intentionally defined a project scope that was almost impossible to be delivered by the interns due to the time, available resources, and intern's prior knowledge and expertise. However, we were hoping that no matter what, we try to get the interns to deliver a product that is functional by the end of

the internship, so they do not look at the overall experience as a failure. At the end of the internship, the delivered product had the following capabilities:

- The ground station has the capability to communicate (bi-directional) with the vehicle and able to transmit the following information
 - Be driven through a joystick via ground station
 - Command the vehicle to drive between waypoints using the interactive map
 - Receive vehicle location and display it on a map
 - Receive live video feed from the vehicle and display it on the ground station
- The vehicle was able to
 - Transmit its location to the ground station
 - Transmit live video to the ground station
 - Follow way points
 - Although the vehicle was able to detect obstacle, it was NOT able to perform obstacle avoidance or even react fast enough to stop the vehicle

All in all, we believed what was delivered at the end of the nine week period actually meet and exceed our initial expectations. The project artifacts are available on Github ^[9].

IV. CONCLUSION

This paper described an experience where a group of ten high school students spent nine weeks in a laboratory, which is involved in development of the products for industry. The main goal of the project, was to integrate the students in an industry setting, and allow them to work on a project from inception to final delivery. The student survey that was conducted at the completion of the internship, points to the following facts.

- Almost all students believed that the internship experience was very good, and they learned technical and professional knowledge that they would not believe would be available to them through their high school education.
- They thought that the interaction with the customer was very good, but they complained about the changes to the requirement by the customer (this was actually designed into the experience by the PI).
- They now have a better understanding of what it takes to work on a complex project and be part of a big team.
- They now have a better understanding of what potential career opportunities exist, if they pursue a computing degree.
- Most of the students thought that they underestimate the complexity of the project, but as they proceed through the life cycle they better realized what they can and what they cannot deliver.
- They thought that they are better prepared to join the work force.
- Number of students point to some of the issues associated with the personality conflicts between the team members.

The mentors (PI, assigned staff, the two SMEs, and the two graduate students), were pleasantly surprised by what the interns were able to accomplish during this nine weeks. The level and complexity of the project given to the team was almost close to what is assigned at the capstone project for students pursuing bachelor degree in computing field.

REFERENCES

- [1] *Degrees of Protection Provided by Enclosures (IP Code)*, IEC Standard 60529, 2004.
- [2] A. Cockburn, "*Crystal Clear a Human Powered Methodology for Small Teams*," Reading, MA: Addison-Wesley, 2004.
- [3] R. Stansbury, M. Towhidnejad, J. Clifford, M. Dop, "*Agile Methodologies for Hardware/Software Teams for a Capstone Design Course: Lessons Learned*", Proceedings of the 118th Annual Conference for the American Society for Engineering Education (ASEE), 2011
- [4] J. Highsmith and A. Cockburn, "*Agile Software Development: The Business of Innovation*", Computer, vol. 34, pp. 120-122, 2001.
- [5] "Why WAMP," Internet: <http://wamp.ws/why/>, [Aug. 17, 2015].
- [6] "Autobahn," Internet: <http://autobahn.ws/>, [Aug. 17, 2015].
- [7] "WAMP Implementations," Internet: <http://wamp.ws/implementations/>, [Aug. 17, 2015].
- [8] "Crossbar.io," Internet: <http://crossbar.io/>, [Aug. 17, 2015].
- [9] T. Bassa, A. Spradling "*NEAR-netcar*," Internet: <https://github.com/ThomasBassa/NEAR-netcar>, 2015.