Team Makeup and its Contribution to Different Types of Deliverables within the Computer Science Field

Saul Gurgua Lopez

Follow this and additional works at: https://louis.uah.edu/honors-capstones

Recommended Citation
https://louis.uah.edu/honors-capstones/370

This Thesis is brought to you for free and open access by the Honors College at LOUIS. It has been accepted for inclusion in Honors Capstone Projects and Theses by an authorized administrator of LOUIS.
Team Makeup and its Contribution to Different Types of Deliverables within the Computer Science Field

by

Saul Gurgua Lopez

An Honors Capstone

submitted in partial fulfillment of the requirements

for the Honors Diploma

to

The Honors College

of

The University of Alabama in Huntsville

February 22, 2021

Honors Capstone Director: Dr. Harry Delugach

Associate Professor of Computer Science

Saul Gurgua Lopez 01/22/2021

Student (signature) Date

Harry Delugach 3/22/2021

Director (signature) Date

Department Chair (signature) Date

William Wilkerson

Honors College Dean (signature) Date
Honors Thesis Copyright Permission

This form must be signed by the student and submitted as a bound part of the thesis.

In presenting this thesis in partial fulfillment of the requirements for Honors Diploma or Certificate from The University of Alabama in Huntsville, I agree that the Library of this University shall make it freely available for inspection. I further agree that permission for extensive copying for scholarly purposes may be granted by my advisor or, in his/her absence, by the Chair of the Department, Director of the Program, or the Dean of the Honors College. It is also understood that due recognition shall be given to me and to The University of Alabama in Huntsville in any scholarly use which may be made of any material in this thesis.

Saul Gurgua Lopez

Student Name (printed)

Saul Gurgua Lopez

Student Signature

01/22/2021

Date
# Table of Contents

Abstract ................................................................................................................................. 1  
Introduction ............................................................................................................................. 2  
  Team Introductions ............................................................................................................. 2  
Team Methodology ............................................................................................................... 4  
  Team A’s Methodology ....................................................................................................... 4  
  Team B’s Methodology ....................................................................................................... 6  
  Team C’s Methodology ....................................................................................................... 8  
Team Meetings ...................................................................................................................... 10  
  Team A’s Meetings ............................................................................................................ 10  
  Team B’s Meetings ............................................................................................................ 12  
  Team C’s Meetings ............................................................................................................ 14  
  Average Team Meeting Statistics ....................................................................................... 15  
Team Communication .......................................................................................................... 17  
  Team A’s Communication ................................................................................................. 17  
  Team B’s Communication ................................................................................................. 19  
  Team C’s Communication ................................................................................................. 21  
  Total Team Messages ....................................................................................................... 22  
Team Member Perspectives ............................................................................................... 24  
  Team A’s Member Perspectives ......................................................................................... 24  
  Team B’s Member Perspectives ......................................................................................... 27  
  Team C’s Member Perspectives ......................................................................................... 29  
  Thoughts on Perspectives ................................................................................................. 31
Abstract

Within the computer science field, we are presented with projects that contain different topics and requirements that are not only handled by one person, but by multiple people. I have worked within three teams the first semester of my senior year that have given me valuable teamworking experiences that I feel can attribute to how different work experiences may vary.

In order to get a better understanding on what it takes to succeed within a group and provide a good insight on how each team treated their projects differently, I have collected data from each team based on initial methodologies, communication, and viewpoints on project handling. Since each project varies from team to team, data will vary as well and should depict different strategies taken among different projects under different situations. I will conclude with my thoughts on how each team experience made me consider approaching the real work environment, where I will be presented with different projects and different team members.
Introduction

The semester of Fall 2020 at the University of Alabama in Huntsville (UAH) provided me with three different team project work experiences within my computer science curriculum. The first team, further known as Team A, had to do a semester long project for the senior software design class, CS 499. The second team, further know as Team B, also had to do a semester long project for the Hacking for Defense Agency (H4D) under the network security class, CS 465. The third/last team, further known as Team C, had to do two smaller projects for the semester under the cloud computing class, CS 454.

Each team varies in number, therefore, in order to differentiate between members in each project and maintain their name anonymous, each member will be represented by “k#”, where ‘k’ is the team letter followed by the member number. For example, D5 would refer to the fifth member of team D, if one existed. Given each circumstance, I will discuss how each team handled each of their final deliverables with the following data gathered from each: methodology towards respective projects, team meeting lengths, team communication frequency, team member perspectives, and project results.

Team Introductions

Team A’s project involved creating a vacuum simulation graphical user interface (GUI) that allows a user to create a house plan with obstructions and different floor types, such as tables and carpet flooring, and run path-following algorithms through a vacuum object to simulate a vacuums performance on a user-designed house. The project was randomly
assigned by the professor. Team A was composed of 4 members and was required to follow a proper software development methodology to complete the project.

Team B’s project was to test and write about the performance of quantum algorithms in cloud-based quantum computing platforms to help improve future problem-solving strategies based on the platforms’ capabilities. The team decided to construct quantum key distribution protocols to run on IBM’s quantum computers due to team interest in cybersecurity and IBM’s Quantum Experience’s ease of access. Each student in the networking class was given a list of projects to rank based on interest. Teams and their corresponding projects were chosen according to the students’ rankings. Team B was composed of 3 members and was required to interview professionals to gain more knowledge about quantum mechanics, an otherwise unknown topic to all members of the team.

Team C’s projects were spread out between the beginning and the end of the semester. The first project required the team to create virtual machines within other virtual machines that would then be connected via an internal network, while the second project required the use of OpenStack’s RESTful interface and the AWS (Amazon Web Services) EC2 interface to create instances of a virtual machine and list them based on ID keys. Both projects were the same for all teams formed in the cloud computing class. In addition, all students were required to learn how to use the Linux command prompt. Team C was composed of 2 members and was required to present and explain their projects.
Team Methodology

A methodology refers to a logical procedure that guides one towards the completion of a project based upon certain views and values. Based upon the project, different teams may decide different ways to approach it. A methodology does not assure that a project will be concluded with all requirements met, nor does it ensure best possible results, rather it simply provides the framework a team can follow based upon what they prioritize when working on a project.

Team A’s Methodology

The focus of the senior software design class is to provide a software engineering work experience that would require one to work with colleagues to complete a customer-given project. In a professional software development work environment, a team is usually expected to follow a proper software development methodology to complete a project and ensure that it will follow customer requirements and will fall within time and budget. A budget, however, was not necessary for the project and will, therefore, remain unmentioned from now on. Such an approach consists of proper documentation of all team discussions, decisions, and work done. In order to properly complete the project, the team decided to follow the Scrum methodology, an agile software development methodology.

The Scrum methodology is designed to aid teams with software development in a way that the team can deliver software quickly and be able to adjust to changing customer requirements. The team believed this was an appropriate methodology to follow due to our busy schedules and the possibility that functional requirements for our GUI may change due to the time it may take to implement certain functionality. Work was divided into epics, which
were then divide into user stories, that where to be completed within an allotted time of three weeks, otherwise known as sprints. Epics are large bodies of work that can be divided into smaller bodies of work, known as user stories. User stories are then assigned to each team member. In addition, Scrum meetings were held after the end of each sprint, of which there where 5 in total, to decide new user stories for the upcoming sprint. The amount of user stories assigned to a member during a meeting was decided upon the difficulty assigned by all members to each user story, rated either 3, 5, or 8, with 3 being of least difficulty and 8 being most difficult.

Each team member was appointed roles at the beginning of the semester, which often meant that they were going to complete a certain type of work, depending on what best fits their role or availability.

Figure 1: Team A Roles

<table>
<thead>
<tr>
<th>Member</th>
<th>Primary Role</th>
<th>Secondary Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>Team Lead</td>
<td>Technical Writer</td>
</tr>
<tr>
<td>A2</td>
<td>Scrum Master</td>
<td>Technical Lead</td>
</tr>
<tr>
<td>A3</td>
<td>Test Lead</td>
<td>Technical Writer</td>
</tr>
<tr>
<td>A4</td>
<td>Product Owner</td>
<td>Quality Lead</td>
</tr>
</tbody>
</table>

Member roles were based on the individual strengths and experience of each member. Member A1 was appointed team lead due to his/her experience with Unity, our chosen game engine that helped us develop our GUI. Member A2 was assigned his/her roles due to his/her experience within Scrum driven software development projects. Member A3’s roles were given to him/her due his/her experience with the same role on previous projects. Member A4, like A3, had more experience with his/her roles from previous projects and was,
therefore, assigned his/her roles. Team A wanted to maximize each other’s strengths in order to maximize potential success.

Team B’s Methodology

Team B was given the challenge of going beyond the general knowledge of computer science and into a realm mostly known by physicists—Quantum Mechanics. The team was not required to turn in a certain type of deliverable by the end of the semester, but since our project was about the performance of quantum algorithms on quantum hardware, we figured that comparing the results of certain quantum algorithms from simulations and actual quantum hardware runs would be a great way to present the capabilities of current quantum hardware. The environment causes errors unpredictably on real quantum hardware that affect the results of the protocols, however, they can be controlled in a simulation.

During normal class hours, we went over key distribution protocols and their contribution on secure communication between parties over several networks. As a result, we developed a good understanding on how key distribution protocols worked. Due to our understanding of key distribution protocols and a mutual interest in cybersecurity, we decided to base our deliverables on quantum key distribution (QKD) protocols. Since we did not know anything about quantum mechanics nor how QKD protocols worked compared to normal key distribution protocols, we decided to adopt a qualitative research methodology. This means we focused on analyzing information given by professional physicists through videos, reports, and interviews in order to explore ideas regarding QKD protocols. Like Team A, we knew we had to meet and communicate frequently to understand what work we had to
tackle based on the semester length. As a result, we separated the semester in halves, in which we spent understanding how QKD protocols work during the first and working on our two QKD programs and whitepaper during the last. Our intention was not to stop researching after the first half, but to lessen it.

Like Team A, roles were appointed to each member, which meant each member would complete a certain type of work based on role or availability. Unlike Team A, Team B only assigned one role per member.

**Figure 2: Team B Roles**

<table>
<thead>
<tr>
<th>Member</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>B1</td>
<td>Team Lead</td>
</tr>
<tr>
<td>B2</td>
<td>Programming Lead</td>
</tr>
<tr>
<td>B3</td>
<td>Research Lead</td>
</tr>
</tbody>
</table>

Unlike Team A, the roles for each member were decided throughout the first 5 weeks, with Team Lead being an exception. Member B1 was given the role of Team Lead because he/she wanted to work on the project the most. All students within the networking class were asked to choose which projects they would like to work on from a certain number of projects available and rank them based on their interest to work on them. Members B2 and B3 had the project ranked last from their selected projects, while member B1 had it ranked first. Due to other projects overflowing with members, members B2 and B3 were put on the project alongside member B1. In addition, upon reaching the tenth week of the semester, we had decided which QKD protocols to simulate. Since members B1 and B2 understood the material best and member B1 was already Team Lead, member B2 became the Programming Lead. Member B3 was given his/her role in order to help find noteworthy QKD
articles and information that may be helpful for the simulations. As a result, roles for Team B were appointed more on developed strengths and necessity.

**Team C’s Methodology**

Team C did not have to do a semester-long project, rather two smaller projects. As a result, we mostly focused on the projects’ requirements and on thoroughly understanding each Linux command’s effect on the function of each project. Unlike the Scrum software development methodology, we did not prioritize time nor change due to the small length of each project and the small number of non-changing requirements. Instead, like the waterfall software development methodology, we made sure each step of a project was complete and understood before proceeding to the next. We essentially took the idea behind the waterfall software development methodology and applied it to how we approached our projects. Since our projects were not software deliverables, we simplified the waterfall methodology to the requirements, implementation, and verification phases where we made sure we knew the requirements of the project, implemented the commands and files needed, and verified the project worked as it should.

Team C did not assign specific roles for each member due to only having to deal with two small projects and the fact that there were only two members. As a result, work between members had to be distributed as evenly as possible.
Figure 3: Team C Roles

<table>
<thead>
<tr>
<th>Member</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>N/A</td>
</tr>
<tr>
<td>C2</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Despite not having specific roles appointed to each member, member C1 often took on the tasks that required more time to complete or, better said, install due to his/her high internet reliability. For example, the first project required two Ubuntu installs under UAH’s pre-installed virtual machine, which took a substantial amount of time just to wait for their installations. As a result, member C1 did those tasks while member C2 took other tasks, such as establishing a network between both of the Ubuntu installs, which didn’t require a particularly great amount of internet reliability.
Team Meetings

Each team scheduled team meetings that were spread out throughout the semester according to the advancement of each member. Each meeting was a group voice-call through Discord held at a convenient time for everyone. Due to the COVID-19 virus that unfortunately closed campus for the entirety of the 2020 Fall semester, all meetings were held online.

Team A’s Meetings

Before recording data about Team A’s meetings, I hypothesized that the frequency of our meetings would increase in a somewhat linear fashion due to the deadline approaching more and more every week and the fact we would be more focused on completing GUI functionality, however, I was mostly wrong.

Figure 4:
According to figure 4, the length of each meeting seemed to decrease and become less varied as the semester approached the end, with the exception of the last two meetings. The first six weeks were about creating a game plan, essentially discussing how we want our GUI to look and do and how we were going to work towards that goal. The next nine weeks, weeks 8 through 15, were about execution, working our assigned tasks for the GUI and final deliverable, which involved less meetings, but more work. The last two weeks were about resolving bugs found on our GUI based on test cases ran and making sure our GUI followed all functional requirements. Other deliverables were also mixed in during our meetings, such as our GUI designs during week 3, our Software Development Plan during Week 4, and our Requirements Definition Document during week 8 for example, however, a brief description of our main focus during our semester weeks seemed better than describing the importance of each deliverable.

The main reason my hypothesis was not correct is because I put more value into conducting weekly meetings than understanding what happened during them. As a result, I assumed we had to meet more frequently to discuss each of our progress in detail so we can all produce quality work without taking into consideration time constraints and the inability to know each detail of the entire project. In order to work efficiently we had to work fast with our assigned work, which meant we had to understand our assignments and trust each others work. This doesn’t mean we didn’t review each others work, but rather had to put more work on delivering our assigned work as best as possible so that we can continue with the next assignments with minimal time wasted.

During the last meeting of week 2, we decided to meet every Monday to review past week progress and to complete each of our team weekly team reports. We believed that by meeting on Monday, we could fully review what we had to work on each week and decrease
our weekly meetings, which we did after week 4. This was done in preparation of other projects team members had to work on later in the semester and to avoid any mishaps that prevent us from meeting any other day. We scheduled meetings in other days only if we felt it necessary however. This is why Monday meetings not only dominate over time, but in length as well.

**Team B’s Meetings**

**Figure 5:**

The first ten weeks of Team B’s project heavily relied on reviewing quantum mechanics and searching and conducting interviews for professionals in the physics field. Our meetings had to rely on the amount of interviews we were able to schedule because without them we wouldn’t be able to decide upon a deliverable that demonstrated quantum computer capabilities with relevance to cybersecurity. We were able to conduct 11 interviews in total, which are shown by figure 5 through the first ten weeks of the semester. I counted the
interviews as meetings because we discussed information during the interviews. Small team meeting times were added on top of those interview times because we just made sure we understood what happened during the interviews during those small meetings. In addition, the following 7 weeks were meetings between team members B1 and B2 because they were the only two collaborators on the final programming and whitepaper deliverables.

Not only did we have to schedule meetings based on our own schedule, but based on the professionals’ schedule for the first ten weeks. However, the last 7 weeks were more thorough since we had already established what our final deliverables were going to be. Despite being able to conduct 11 meetings, not all meetings gave us enough information to proceed with our ideas and, as a result, we could not organize our time as well as we would have liked to until we interviewed as many professionals as we could. Even after we knew what we wanted to do during the tenth week, we could not schedule certain days to meet each week due to other work and projects from other classes. As a result, from week 11 on, members B1 and B2 had to decide when to meet based on availability and progress made.
Team C’s Meetings

Figure 6:

Team C had, by far, the fewest number of meetings done throughout the semester as shown by figure 6. Having small projects with an intermediate complexity helped us a lot during these projects. Project week meetings for project 1 were held on weeks 7 through 9, while only one meeting was held in week 13 for project 2. We had most of the knowledge needed to complete these projects and, therefore, did not have to schedule many meetings. Meetings were only scheduled to go over what we have done and make sure we can explain it to the teacher. It is worth mentioning that these meetings were scheduled based on the availability of both members with little consideration on being ordered.
Average Team Meeting Statistics

Figure 7:

![Team Meetings Conducted](image)

Figure 8:

![Average Team Meeting Length](image)
As shown in figure 8, the average team meeting lengths did not vary much from each other, with all averaging near an hour. The main difference is the amount of meetings conducted per team as shown in figure 7, with Team C being the obvious one. It makes sense that the average meeting time for Team C is the longest since there is more to be discussed within the small amount of meetings. It is also reasonable the average team meeting length for both Team A and B are close, considering the number of meetings conducted are also close. However, these statistics could have easily differed if there were different number of concerns, amount people, situations, and/or project complexities for example and would, therefore, not force a concrete meaning upon them.
Team Communication

Aside from meetings through group voice-calls, each team communicated via group text messages for non-scheduled meetings or concerns on assigned individual work. All messages were viewable by all members in all teams so we can all be updated with individual progress. Most messages pertained to all members of a team or had to do with the progress of an individual, which each team felt should be viewable to all members. Team B was an exception.

Team A’s Communication

Figure 9:

Communication throughout each week did not vary greatly for Team A, with most notable spikes in messages during week 4 and 16. During week 4, we were going over final GUI designs, team report risks, and bug fixes within our Github repository that took some
time to fix. Week 16 was spent going over GUI bugs and requirements for our final presentation and for our final deliverable. Week 5, 7, 10, 11 were weeks of user story assignments, which were when Scrum meetings were held and everyone decided on what user stories to create, voted the difficulty of each, and assigned them to members of the team. Everyone on the team would text their difficulty vote for a user story on the group server for example, which would count for a good amount of the messages those weeks. Any other week consisted of messages that either involved scheduling meetings, asking questions on a particular user story, or bug notifications.
**Team B’s Communication**

The number of messages within Team B was not as stable as those between Team A due to a learning curve. Team B divided the messaging servers to two. The first was for general messages pertaining to progress made, project questions, and meetings and involved all members, while the second focused on programming-related questions for our QKD protocol programs and only involved members B1 and B2.

**Figure 10:**

![TEAM B GENERAL MESSAGES](image)

Messages within the first six weeks only consisted of questions about interview scheduling, content, or preparation. We were tasked with watching several lectures on quantum mechanics regarding quantum circuits and the properties of qubits since the beginning of the third week, however, not all teammates were able to make the same progress on understanding the content. Team member B3 was not able to learn enough content by the end of week 6 to help the team brainstorm ideas on how to create QKD
protocol simulations, so team members B1 and B2 had to communicate between each other to begin making progress on the simulations. This was done by members B1 and B2 so not to impose new material on member B3 without him/her understanding the basic material needed or else B3 would become more confused. Despite member B3 not fully understanding all material, he/she was given the task of finding articles recommended by interviewed professionals and other articles on certain topics needed for the programs and give summaries or highlights on the topics members B1 and B2 may need to continue progress on the programs. Messages within the entire team after week 10 were to make sure we were all making progress in our tasks and make sure we are ready for our weekly in-class zoom presentations of our progress.

Figure 11:

![TEAM B PROGRAMMING MESSAGES](image)

As shown in figure 11, the second messaging server was created in week 7, where members B1 and B2 discussed how the programs were going to be handled. By the end of week 10, members B1 and B2 had settled with two QKD protocols to simulate and were
ready to collaborate on creating the programs. The programming lead, member B2, worked on the overall structure and content of both QKD protocol programs during weeks 10 through 13, while referring to member B1 for opinions on his/her progress. Upon reaching week 14, they both worked on specific parts of both protocols and made sure their work didn’t affect each other’s progress. Week 15 was spent mostly discussing the results of both programs and finding and fixing errors. The last two weeks consisted of testing and adding comments throughout both programs.

Team C’s Communication

Figure 12:

Communication within Team C was far less complex than both Team A and Team B. Messages were just to discuss issues we had during our virtual machine installations on UAH’s servers during week 5 through 7, while week 8 and 9 were about going over what we needed to explain to the professor during our project presentation. Our second project,
which involved communication within weeks 11 through 13, were mainly about explaining what every command we implemented meant when creating instances of virtual machines.

We decided to use week 13 to go over all of project 2 due to us running into a small number of complications. Whenever we had meetings that involved us discussing how we were going to explain our projects to the professor, we sent commands back and forth with an explanation on what they did. We did this because most of our work involved us using the Linux command prompt to set up what we wanted. Therefore, we sent each other the commands we used to perform each important step of each project and then tried to explain the parts of the commands that allowed the entire command to work as it should. That is why weeks such as week 8 and week 13 consisted of more messages than the rest of the weeks during their respective project time frame.

_Total Team Messages_

_Figure 13:_
Team B had the highest number of messages sent between each other with a total of 978 messages. The first bar for Team B from a top-down view is the second server set for B1 and B2 to communicate about their QKD protocol programming, while the second is the first/general server established for all team members. As mentioned before, this was done in order to not confuse member B3 with any other information discussed about the programs. The reason that server alone contained so much communication is because it was started around mid-semester and it was about the main deliverable, the programs we had not yet began to work on. Team members B1 and B2 had to constantly talk between each other to be kept in the same page as far as progress and changes that needed to be made to the programs. Not only that, but each of them had to do their own research on some parts of the project and then explain why and how they changed the program to fit their research. The project complexity was high and with two people working on the main deliverable, they had to communicate as much as possible whenever they both could. Since Team A had a semester long project and was able to be more organized and spread work evenly between four members, they had the second most amount of messages sent between them. Since Team B’s projects complexity was not as high as the other teams, they didn’t average as much as the other teams as expected.
Team Member Perspectives

Upon completing their respective projects, each team member was interviewed and asked the following questions about the project:

1. What did you think we excelled in?
2. Was there anything you would have done differently? If so, what would you do differently?
3. Do you think we did anything particularly bad or subpar? If so, could you give some examples?
4. Based on your answers to the previous questions, would you say you are satisfied with how our project(s) turned out?

The point of the interviews was to see how everyone felt while working together and if they had any opinions regarding what we lacked on as a team, what we did right, or how we could improve.

Team A’s Member Perspectives

Member A1:

1. “As a team, I think we excelled in communicating. We picked up the slack of one another and did a great job of constantly updating each other on our progress.”
2. “Yes, I probably would have delegated a bit more of the SCRUM process to [Member A2] since he was the SCRUM master.”
3. “No, I don’t think we did anything badly or sub par.”
4. “Yeah, I am very satisfied with the result of our project. I think it turned out very well.”

Member A2:

1. “I think we did great at communicating where we were at with the project at each stage.”
2. “We just barely dodged a bullet on our second presentation because we weren't paying attention to the timelines. I think we would need to in our first meeting go down the list and double check all of the timelines.”
3. “I think we could've made a better effort at practicing or running through each of our presentations. The only time I think we sufficiently went though and had a good game plan was the final presentation. Fortunately for us, the presentations weren't graded that harshly and they weren't a big part of our grade.”
4. “I am very satisfied with the quality of teammates I had. In the past, almost all of my group members did not do entirely what they were supposed to do, and I would have to make up for it. In this group I did not feel that way at all and I don't think any member really held us back.”

Member A3:

1. “I believe we did a great job of communicating and keeping ourselves in track throughout the semester.”
2. “I would have probably tried to tackle the saving and loading task with JSON files in order to become more familiar with the concept.”
3. “I don't think we did anything particularly bad. Commenting the code was probably something we should've worked on more.”
4. “I was very satisfied with how our project turned out and had a great experience with the team.”

Member A4:

1. “I feel like we did a good job of communicating while working on our user stories and keeping each other updated on our progress.”

2. “We could have probably spent a little more time optimizing the performance of our code subroutines.”

3. “We struggled to schedule each team member’s user stories to not depend on the progress of others’ user stories, but we managed to work through it.”

4. “Overall, I was very satisfied. We all stayed constantly in touch with each other discussing progress and did not let ourselves fall behind.”
Team B’s Member Perspectives

Member B1:

1. “I think we did a pretty good job learning enough about quantum physics and quantum computation to pull this off. Other than that, nothing comes to mind that we excelled in.”

2. “I think I should have made more of a point of determining whether or not [Member B3] understood the project at an earlier point instead of drawing that process out.”

3. “I don't think there was anything that went particularly poorly. There were some communication issues we had about what needed to be done, but those are common and were easily correctable. I also wish we'd been able to split the workload better across three people, but that's another one of those things that just doesn't work out sometimes.”

4. “I'm pretty satisfied with how the project went. There was more last-minute stuff than I would've hoped for, but everything seems to have come out fine in the end.”

Member B2:

1. “I don’t think we necessarily excelled in something throughout the semester aside from learning enough quantum mechanics to create our project. The entire project was a learning process, which I believe we did really good at, but not necessarily excel at.”
2. “I would have liked to ask more questions to the professionals we interviewed, however, I didn’t have those questions until we almost had our project done.”

3. “I believe we struggled understanding and combining our ideas together at first, however, we did a good job at progressively building upon them.”

4. “I was pretty satisfied with how our project turned out considering we didn’t know any quantum mechanics when we began.”

Member B3:

1. “I think we have excelled in the protocol implementations and our research on the subject as we gather as much information for the project.”

2. “Something I could have done differently is to help out more in the coding process to give myself more in depth on the implementation of protocols while I helped out on the researching process of the project.”

3. “I think we did really well in the project since we put in a lot of time and effort on it.”

4. “I am very satisfied of how it turned out as an end result because we each did our best to make sure we do our parts to understand and gain knowledge on the particular subject of quantum physics.”
Team C’s Member Perspectives

Member C1:

1. “I think we did several things very well. First, I think we effectively used our knowledge from previous classes, such as that of Linux and networking, to complete our project. Also, I think we did a good job of starting the project early which allowed us to overcome unforeseen challenges. Lastly, I think we did a good job of attending the professor’s office hours and asking meaningful questions to get her help with the project."

2. “The main thing I can think of that I would’ve liked to have done differently is I would’ve liked to have spent more time exploring all of the code that we used for the second project. This code was given to us by the instructor and we used it successfully to complete the project, but if we had more time, I would’ve liked to have played around with it a little bit more."

3. “The only thing that I think we did not do enough of was reading the book before the project. We very much had a hands-on approach to completing the project and a “figure it out as you go” attitude. Later, I realized that the book was a very useful resource, and it may have helped to study it more deeply before approaching the project."

4. “Overall, I was very satisfied with our project.”
Member C2:

1. “I think we did a great job at explaining both our projects to the professor and within ourselves.”

2. “I would have probably tried to stretch out the times we met throughout the weeks we worked on our projects a little bit, but that’s just a minor change that I don’t mind much. I would have just been a little more organized.”

3. “I don’t think we did anything bad, considering we met all criteria for our projects within a good time frame.”

4. “I was very satisfied with how our project turned out considering we barely had any mishaps.”
Thoughts on Perspectives

It seems that all members in Team A believe that communication was a great strength within the team. Team member A3, however, pointed out situations worth mentioning that the other members forgot to mention. Presentations were not of the essence for us and is something that will be very important in the workforce. Knowing exactly when and how to do a presentation will always be essential, considering it is the customer who has the biggest say on a project and should understand a team’s progress to decide what needs to be altered.

Team B encountered more problems compared to the other teams and acknowledge it well. Sometimes there will be some members of a team that do not fully understand their assignments or a project as a whole and it is the responsibility of the team to ensure all members are in the same page since the very beginning to avoid issues in the long run. The workload was not very well divided between all members of Team B as a result, with some doing more work than others.

Despite Team C not having major problems throughout their projects, a slight change in communication can make an experience better. If it were not for all members understanding what had to be done and the fact there were only two members, the lack of meetings could have been a major disturbance. Also, as member C1 indicated, a further in depth look of what was provided to us by the instructor, along with her book, would have enriched our knowledge of our projects and of cloud computing in general.
Conclusion

Given the different situations faced by each team and the number of different approaches towards each project, the results did not vary much.

Figure 10:

Figure 10 shows the resulting grades of all the projects for all teams with Team C having an extra grade due to them having two projects. The resultant grades for all team projects were very good and varied little between each.

A comparison between grades of different types of projects may not seem like an adequate comparison, however, the grades themselves are meant to show that each computer science related project does not require to follow the same or similar procedures to deliver good deliverables. Whether a project is related to software development, research, networking, or any field one is familiar with, one should not always rely on approaching them the same way, especially within different team makeups. Due to everyone having different
opinions, experiences, tendencies, and time, one should rely on taking past or concurring experiences and adapt what worked then to what could work in other projects in the future, keeping in mind that an experience will never be the same. For example, one characteristic shared within each team was good communication, which should be established early on during project development to prevent any mishaps.

Team methodology, meetings, communication, and perspectives are topics I consider all teams should discuss throughout the entirety of a team project. As I previously mentioned, no experience is the same and I would say that is due to not having everything predetermined. A team’s early decisions are subject to change throughout the course of the project to accommodate a team’s progress. For example, Team A’s communication throughout the project became increasingly more efficient, which eventually allowed them to conduct less meetings than expected, Team B had to adjust how they were going to proceed with their QKD protocols due to a learning curve, and Team C had to conduct meetings based on each other’s progress and availability instead of having decided when to have them before doing any work. Having a methodology or a plan for conducting a project is always a good idea, but a team’s progress, whether good or bad, will more than likely involve some changes on behalf of all team members. Such changes can debunk hypotheses made on how a project was going to turn out and, in turn, may give valuable insight, as it did for me regarding the importance of efficient communication over meeting length. In summary, one should learn how to adapt to each situation for the better good of the team as soon as possible, acknowledging team progress and each team members’ perspective on a project, not only after finishing a project but during. In addition, asking for the perspectives of one’s peers can have an impact on one’s approach with other people and future projects.
As I approach the end of my undergraduate career and on to the workforce, I hope to understand what it takes to succeed within different team makeups under different types of projects. The 2020 Fall semester has given me a good insight on how the workforce can be and experiences I hope to build upon.