AN AGILE DRIVEN DESIGN AND IMPLEMENTATION OF SALES, INVENTORY, AND LOGISTICS SYSTEM

Roy Ivan A. Cruz
University of Asia and the Pacific
Pasig, Metro Manila, 1603, Philippines
royivan.cruz@uap.asia

Monica T. Rosario
University of Asia and the Pacific
Pasig, Metro Manila, 1603, Philippines
monica.rosario@uap,asia

Giorenzo F. Gomez
University of Asia and the Pacific
Pasig, Metro Manila, 1603, Philippines
giorenzo.gomez@uap.asia

Giuseppe C. Ng
University of Asia and the Pacific
Pasig, Metro Manila, 1603, Philippines
giuseppe.ng@uap.asia

ABSTRACT

Automation is necessary for companies to increase productivity especially when the company has to handle large volumes of data every day. The GMI Company is one of them wherein sales and procurement transactions happen daily. Although GMI has an existing sales and inventory system being used, the researchers found out that some aspects of their business process still revolve around manual methods. Particularly, the organizing and assigning of product deliveries to delivery vans are still handled through pen and paper. With this, the researchers proposed a system that would not only cover the company's existing system functionalities, but also add automation for their logistics process. The Agile Methodology was used to develop a web-based system. This required many meetings between the client and the researchers to evolve the requirements of the client. The researchers were able to design and implement a cross-platform website accessible to different departments of the company can access it. Because it is web-based, the system is accessible through the Internet allowing the company's warehouses 9ito share and sync information easily.

KEYWORDS

Software Engineering, Agile Methodology, Scrum Framework, Web-based System.

1 INTRODUCTION

The GMI Company is a rice dealing business in the Philippines. The company provides rice to different companies in Metro Manila. They also cater to local markets and grocery stores. The GMI Company deals with numerous transactions every day. Their business process begins with the salesperson receiving orders from customers through phone calls, text messages, emails or in person. The ordered products are then verified by the inventory person if stocks are available. Once verified, the customer pays and gives order details such as delivery location and delivery date. At this point, their current IT system is able to record the sales and update inventory data. However, when delivering the products to the customers, their current IT system is unable to share information among their three other warehouses. When product stocks are needed from one of the warehouses, the logistics person has to make a phone call to the warehouse and distribute lists of orders that needs to be delivered. The process of phone calling takes up approximately 3-5 minutes for every order list which hinders the productivity of employees. Moreover, when the sales department needs to create sales reports, they make use of Microsoft Excel because it is able to perform needed mathematical operations which their current system is not capable of doing. Lastly, a manual process consisting of paper forms is used for delivering the products to customers. With this, the logistics department encounters problems in organizing and assigning customer orders to delivery vans. Due to the manual

process, inefficient assignment of delivery routes to delivery vans happens which constitutes to a backlog of 3-7 deliveries a week.

Software Engineering refers to the type of engineering that deals with the detailed development, design, maintenance of a software. It was introduced to systematically solve problems that are encountered in software. It aims to reach all of the requirements by the customer while in time and budget constraints [1]. A good software has its attributes such as Availability, Performance, Reliability, Scalability, Usability, Security, and Maintainability. Availability refers to the time the software is working and functioning properly. Performance is how responsive the software is to do an action in a given time interval. Reliability applies to how well the software can function over time. The ability of the software to scale up or down without damaging its functions is called Scalability. Usability determines if the software meets the requirements of the customer and how well it functions. Security deters unauthorized access and provides protection for all the information stored. Maintainability is the ability of the software to go through changes without any problem. These changes may include upgraded features, fixed bugs and errors, and adding new customer requirements [2].

According to the Standard Glossary of Software Engineering Terminology, Architectural Design is "the process of defining a collection of hardware and software components and their interfaces to establish the framework for the development of a computer system [3]". It is used to represent the overall structure and components of the system and is also used by software engineers to get a bird's eye-view of the system. It allows them to see effectiveness of the design and to see what might go wrong with the design and how it could be solved.

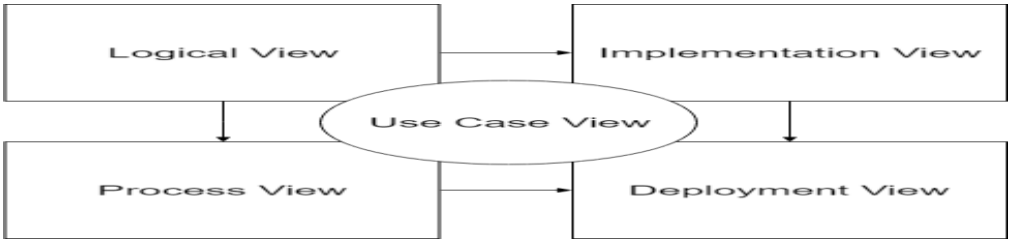

Figure 1. 4+1 Model

Figure 1 shows the 4+1 Model. The 4+1 Model is developed to have a consistent architectural design and structure in the software. It consists of five views. The Use Case View, Logical View, Implementation View, Process View, and Deployment View. The Use Case View are the possible scenarios that may happen in the system. The Logical View is used when an object oriented design method is being used and its description is needed. It is usually used for functionalities needed by the user. The Implementation View refers to how the software's modules and subsystems development are organized. The Process View checks the concurrent processes, performance, and the availability of the system. The Deployment View, also known as Physical View, refers to how the system is going to be deployed and checks the availability, scalability, and reliability of the system [4].

One of the most commonly used methodology in software development today is the Agile Methodology. It focuses on progressive outcomes, team collaboration, and continuous planning. It has its four core values in the "Agile Manifesto". The four values are:

"Individuals and interactions over processes and tools,

Working software over comprehensive documentation,

Customer collaboration over contract negotiation,

Responding to change over following a plan." [5]

The terms on the left are preferred than the terms on the right but it does not mean that it will not be used or will not be needed.

An Agile project has its own framework called Scrum. It is used to organize roles and responsibilities of each person within the team. A Scrum consists of product backlog, sprint backlog, daily/weekly stand-up meetings, and the team.

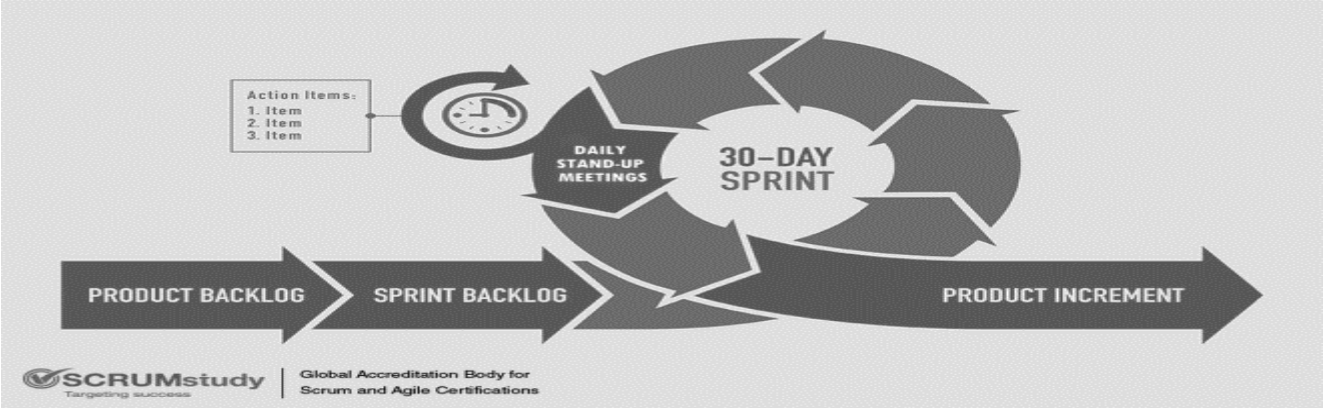

Figure 2. Conceptual Framework of an Agile Scrum Methodology [7]

Figure 2 shows the lifecycle of Scrum. It is incremental, and a complete life cycle, usually lasts 30 days, is called a Sprint. The product backlog is a list of prioritized plans to be done by the team. A Sprint backlog is also a list of prioritized plans but it has to be done until the end of a Sprint. The daily/weekly stand-up meetings are conducted to ensure that the team is in-line with their plans and to discuss the progress of each team member in their assigned tasks [6].

In an Agile Scrum, there are three roles: (1) Product Owner; (2) Scrum Master; and (3) Scrum Team. The Product Owner is the reason behind the product that the team is building. They set the requirements and make sure that the plans are being followed and are up-to-date. The Scrum Master is more of an adviser of team and helps the team when there are issues that could cause delay. The individuals responsible for building the product are the Scrum Team [6]. The Agile Methodology works with a small team consisting of a few members and each member should be self-organizing [8].

2 RESEARCH OBJECTIVES

Due to the size of the team, the researchers opted to use the Agile Methodology. Since the outcome of Agile is to produce a working product every Sprint, the researchers are able to provide the GMI Company the software for them to use while it is being developed. With this, the client is able to test and evaluate the software while operating the business and adding value in the process.

The research objectives of the project are:

1. To identify the issues of the company;
2. To describe the evolution of the system's design; and
3. To present the design and implementation of the inventory, sales, and logistics system.

3 METHODOLOGY

The methodology used is Applied Research. It is a methodology that is often used to solve business, industrial, and medical problems [9]. It can also be used to solve day-to-day problems and to develop new innovations in technology [10]. The results in Applied Research are assessed by the client and are shared through reports or meetings [11].

3.1 Data Gathering

In order for the researchers to understand the problem of the GMI Company, a series of interviews and observations of the company's business process were conducted. An on-site walkthrough on how the company operates was given by the inventory person of the company to the researchers. Questions were asked by the researchers during the walkthrough to clarify and fully understand the business process. In addition, emailing and instant messaging were used by the researchers to get in touch with the client. The information gathered in the interviews and observation were then translated into a requirements document containing diagrams explaining the process of how the system works. From time to time, the requirements document is shown to the client and they give feedback and comments about it. In effect, it ensures that the gathered data and analysis from the interviews and observations are accepted by GMI.

3.2 Prototyping

A prototype of the system was developed by the researchers based on the initial requirements document. The Agile Methodology was implemented during prototyping. Three Sprints were conducted and every Sprint consists of 30 working days (Monday - Friday). Stand up meetings were done daily and consultations with the advisor was done every Wednesday. At the end of each sprint, a client meeting was conducted wherein the researchers gathered feedback and comments from the finance head of the GMI Company. An evaluation form was also used to rate the system for (1) ease of use, (2) pleasant appearance, (3) reliability, (4) security, (5) efficiency, (6) completeness of function, (7) completeness of system, and (8) readiness for deployment. Each of the criteria is scored using the Likert scale and analysis is calculated through the mean score.

4 RESULTS AND DISCUSSION

Table 1 shows the questions of the initial interview conducted by the researchers at the start of data gathering and analysis.

Table 1. Interview Results

Question	Answer
How does your business operate? What are the processes?	• They issue an order form to suppliers which contains information such as the type of rice, quantity, price, etc. • Once approved, they receive it in one of their warehouses to keep in stock which eventually will be delivered to their customers • Their logistics team plans the route of each delivery van on where to deliver the ordered rice.
What is the current software system that is used by the company?	Their current system handles inventory, sales, and delivery. It basically does most of the work needed to run the company. However, their software is quite outdated due to its inefficiency in displaying and encoding information.
Does the current IT system improve company performance? Why or why not?	The current system helped the company with their work but since the current system is outdated, a lot of things in their current system should be improved by upgrading their system to provide better features and user friendliness.
How can the system improve?	By upgrading it to a more modern system which provides optimized and efficient way of handling information.

Based from the interview results in table 1, the researchers identified that personnel from the sales, inventory and logistics department are the ones who will interact with the system.

4.1 Software Context

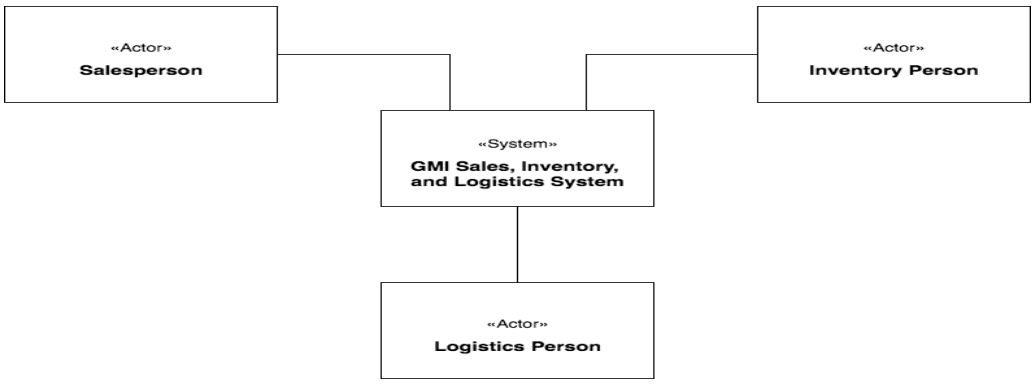

Figure 3. Software Context Diagram

Figure 3 shows the software context and the stakeholders involved in using the system. Each department has different user privileges.

4.2 Initial Architectural Diagram

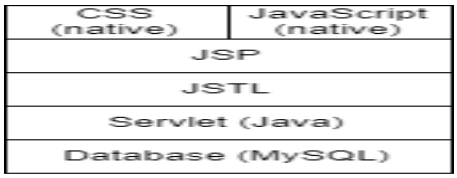

Figure 4. Initial Architectural Diagram

Figure 4 shows the architectural diagram created in the initial analysis based on the interviews and observation. The idea was to develop a Java-web based system with the use of Java Server Pages Standard Tag Library (JSTL) as a means of showing data from the backend to the Java Server Page (JSP). The use of native JavaScript and Cascading Style Sheets (CSS) was also chosen for designing the frontend.

4.3 Entity Relationship Diagram

Figure 5 shows the Entity Relationship Diagram (ERD) of the system. Products are in the Inventory and are replenished when the Suppliers deliver their supplies and for every Supplier Order there is a Procurement. Products are diminished every time there is a Customer Order. The Salesperson records every customer order transaction represented by Sales. Customer Orders have a Customer Order Status in order to distinguish if the orders are Pending, On-going, or Delivered then a Customer Order is assigned to a delivery van. The details of the assignment of Customer Orders to delivery vans are in Customer Delivery. The ERD is based on the initial design and has not changed after developing the system.

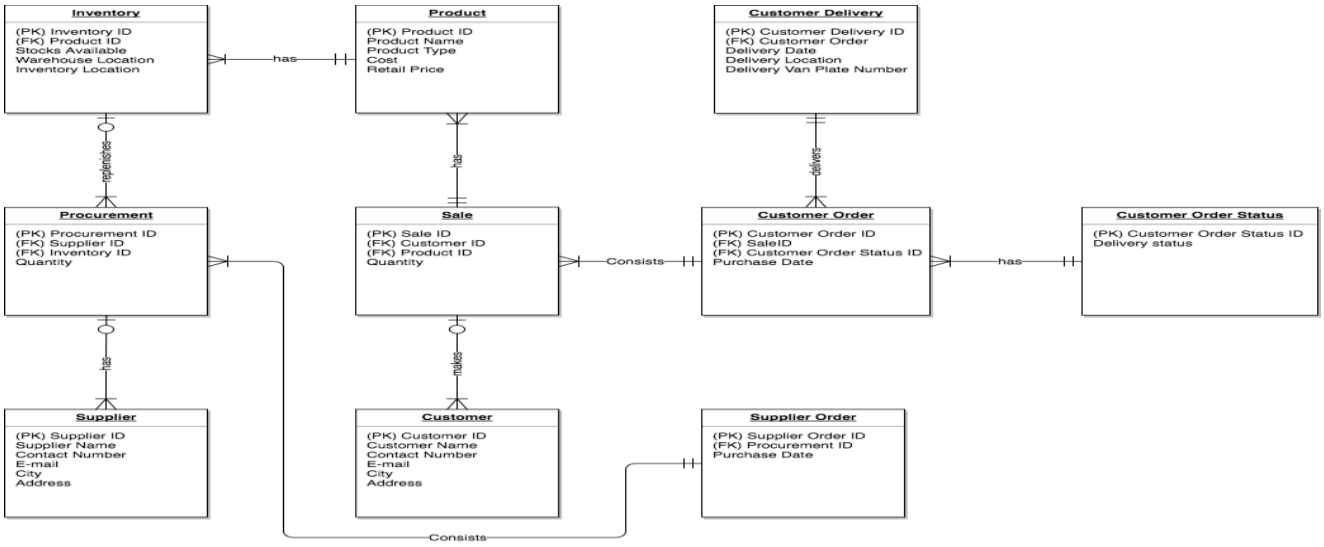

Figure 5. Entity Relationship Diagram

4.4 Class Diagram

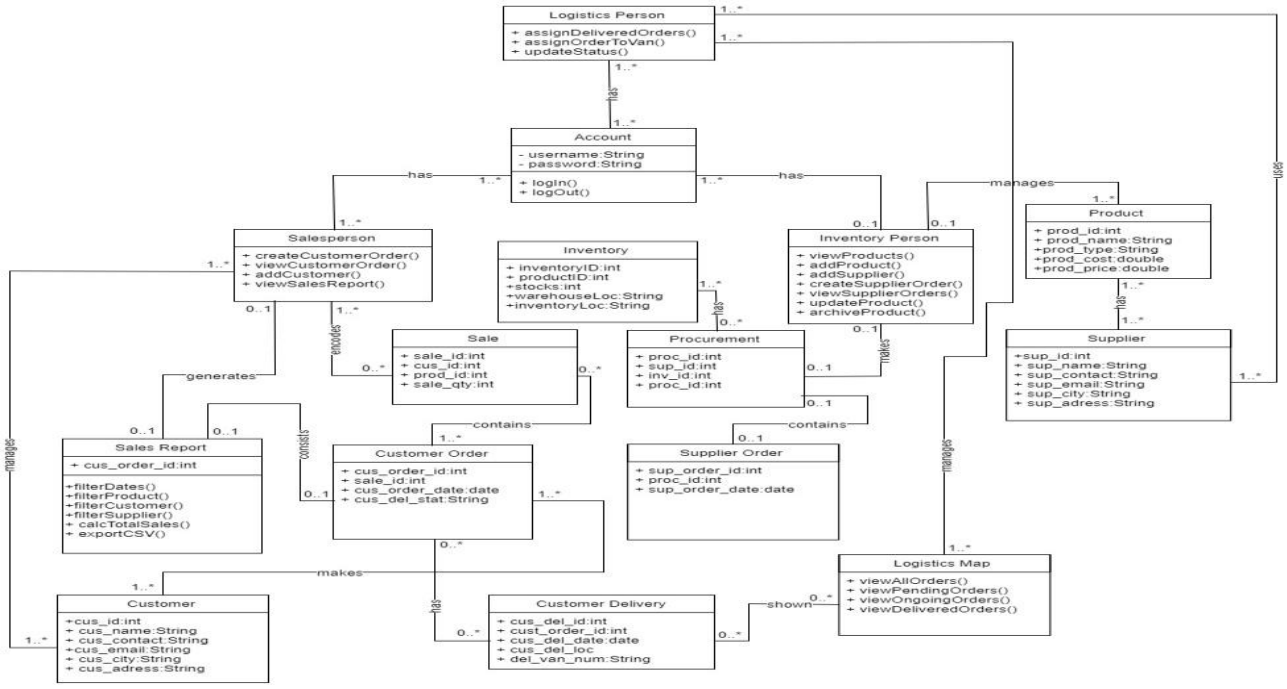

Figure 6. Class Diagram

Figure 6 shows the class diagram designed for the system which relates to the logical view in the 4+1 Model. It is based from the entity relationship diagram found in figure 5. Each entity is linked together to form associations which describe how data is processed from one to the other. The initial design of the class diagram was used and served as a useful guide for implementing functionalities to the system.

4.5 Client Modifications

Table 2. Modifications of the Client

Sprint	Additions/Modifications
1	1. Add a manager account that can view everything. 2. Viewing of Customer Profiles. 3. Viewing of Supplier Profiles. 4. Add credit limit for customers. 5. Add product cost and markup price. 6. Product stock limit. 7. Notification of low stocks. 8. Remove archive of products.

2	1. Terms of Payment of Customer 2. Limit of Customer Order 3. Show Price in the Customer Order
3	1. Input of delivery address should be in ‘add customer’. 2. Stocks should not be editable in ‘update products’.

Table 2 shows the client modifications per sprint. At the end of each sprint, the client gives feedback and suggests modifications to the system.

Based from the feedback of the client from the sprints, the design and implementation of the architectural diagram was necessarily changed. The initial design was insufficient and was limited wherein it was unable to produce the additional features or modifications from the client.

4.6 Revised Architectural Diagram

Figure 7 shows the revised architectural diagram after conducting an agile implementation during software development. Since the agile methodology promotes discovery of new ideas [12], it has helped the researchers to develop a much more maintainable and scalable system with the use of third party libraries such as GSON, Bootstrap, AngularJS, and jQuery. Google Maps API was used for the visual representation of delivery locations for the logistics department. With the use of Google Maps, the developers were able to produce a functioning map for logistics in a matter of weeks. Furthermore, the researchers found out that JSON is a much more standardized data-interchange format when compared to JSTL. With that, the GSON library was used to generate JSON data format to interact with AngularJS and jQuery for displaying information. Functionalities were also split into different packages for maintainability and scalability of the software.

4.6 Client Evaluation

Table 3 shows the mean score for the three sprints during the development phase of the system. The finance head of the company evaluates the system at the end of each sprint. The criteria adhere to the attributes of a good software. Ease of use and pleasant appearance falls under the attribute of usability. Reliability, completeness of function, and completeness of system is for the reliability attribute. Efficient pertains to the performance attribute. Readiness for deployment relates to the attribute of availability. Lastly, the attribute of security is scored as well.

Based from the scores from the client, it is evident that the developed software was successful and the use of the agile methodology significantly helped in the software development.

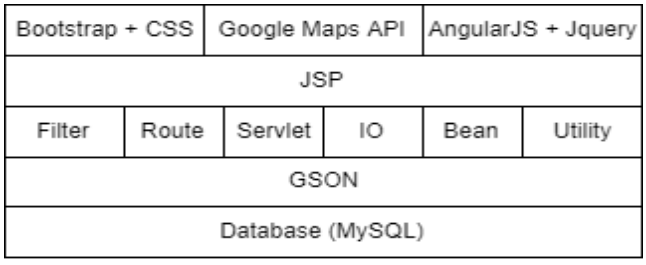

Figure 7. Revised Architectural Diagram

Table 3. Mean Score of the 3 Sprints

Criteria	Mean Score
Ease of Use	4.33
Pleasant Appearance	3.67
Reliability	4
Security	4.67
Efficiency	4.67
Completeness of Function	4
Completeness of System	4.33
Readiness for Deployment	4

5 CONCLUSION

The researchers were able to identify problems in the company with the help of interviews. These problems mostly came from their current IT system. The current system of GMI is unable to share information between different warehouses. This system also lacks the creation of sales reports which makes the employees use third party applications such as Microsoft Excel. Lastly, the current system does not have features that assist the logistics team when it comes to managing the routes of deliveries. With the help of the agile methodology, the researchers were able to track all the changes and modifications to meet customer requirements. After the third sprint, the researchers were able to complete a prototype of the system. The evaluation of the client proved the prototype system to be a success

REFERENCES

[1] CSE. Computer Science and Engineering. Retrieved from https://cse.ucsd.edu/faculty-research/software-engineering

[2] Microsoft. Chapter 16: Quality Attributes. Retrieved from https://msdn.microsoft.com/en-us/library/ee658094.aspx

[3] IEEE Std 610.12-1990 - IEEE Standard Glossary of Software Engineering Terminology. 1990 Retrieved from http://standards.ieee.org/findstds/standard/610.12-1990.html

[4] Philippe Kruchten. 1995. The 4+1 View Model of Architecture. Retrieved from http://www.ics.uci.edu/~andre/ics223w2006/kruchten3.pdf

[5] Microsoft. Aaron Bjork. What is Agile?. Retrieved from https://www.visualstudio.com/learn/what-is-agile/

[6] Microsoft. Gregg Boer. What is Scrum?. Retrieved from https://www.visualstudio.com/learn/what-is-scrum/

[7] SCRUMStudy. An Introduction to SCRUM Framework. Retrieved from http://blog.scrumstudy.com/an-introduction-to-scrum-framework/

[8] Ambysoft. Roles on Agile. Retrieved from http://www.ambysoft.com/essays/agileRoles.html

[9] YourDictionary. Examples of Applied Research. Retrieved from http://examples.yourdictionary.com/examples-of-applied-research.html
[10] Cherry K. 2018. What is Applied Research? Retrieved from https://www.verywellmind.com/what-is-applied-research-2794820
[11] Belmont. 2018. Public Relations: Academic and Applied Research. Retrieved from https://belmont.libguides.com/c.php?g=65860&p=424912
[12] Anon. Characteristics of Agile Development Success. Retrieved from https://www.versionone.com/agile-101/agile-development-success/

ACKNOWLEDGEMENTS

The researchers would like to acknowledge Dr. Mark Anthony Sabili and Mr. Brylle Bae for their overwhelming support and guidance. The researchers would also like to thank the company, GMI Corporation, for giving the researchers the privilege to work with them in making this research possible.