SEG3902

CO-OP WORK-TERM REPORT (Winter 2014)

**AGILE SOFTWARE DEVELOPMENT**

MediaVantage

By

Samira El-Rayyes

6439366

Presented to Professor Thomas Tran

Software Engineering Academic CO-OP Coordinator

University of Ottawa

May 9, 2014

Table of Contents

Abstract 3

I Introduction 3

II MediaVantage 5

Public Relations 5

What is MediaVantage? 5

III Agile Software Development 5

IV Other Projects 8

Research & Development 8

Code Bugging 8

Quality Assurance 9

V Recommendations 9

MediaVantage 9

CO-OP Program 10

VI Conclusion 10

Bibliography 12

Appendices 13

APPENDIX A: MediaVantage, Log-In 13

Supervisor’s Approval 14

## Abstract

This report will include details about the Agile Software Development process that MediaVantage adopted as their main development process. I will also discuss the general benefits and some challenges of the agile process, and why it is considered one of the most efficient and most popular current development processes. My main duties throughout my work term as a Software Developer at MediaVantage included participating in all of the software development procedures for a release for the company's current primary product, MediaVantage. I was responsible for participating in writing unit tests, debugging, refactoring and investigations for this product all by following the agile development process adopted by the company. Moreover, I performed some other tasks such as assisting in related research and development areas and performing some regression testing. I had the privilege to experience employment with MediaVantage, and I was fortunate enough to see for myself the positive results of a product being developed in an agile environment, all while expanding on my technical software research and development skills. My engineering background has been highly beneficial to me while acquiring new technical knowledge. In addition, I gained interpersonal and team skills by working with influential individuals who allowed me to learn from their expertise, as well as expanding my network to include experienced professionals. This work term was highly critical to my development as an engineer and a growing member of the high-tech community.

# I Introduction

For my third CO-OP work term in Software Engineering at the University of Ottawa, I worked for MediaVantage from January 6th through April 25th, 2014. I was hired as a Software Developer under the supervision of Mark Bason, one of the senior developers of the Development Team. MediaVantage was designed and developed by dna13, a software company based in Ottawa, Canada. CNW Group, a Canadian company that had already been reselling their software solution since 2005, acquired dna13 in April of 2010. The dna13 solution was renamed MediaVantage in September of 2010. A trademark of CNW Group, which is owned in part by US-based PR Newswire, MediaVantage’s main development office is located in Ottawa, Ontario, with other offices in Toronto, Ontario and Montreal, Quebec.

Within my first few days at MediaVantage, I was introduced to the members of the Development team and other teams that they work closely with, such as the Quality Assurance and Support teams. By the end of the first week, I had nearly met every employee in the company and already had most of my workstation set up for my next few months of work. I was also fortunate enough to have to the opportunity to take part in a multiple events throughout my term that focused on developing interpersonal skills such as team-bonding activities, foosball challenges and lunch outings with the company.

My performance goals from the beginning of the work term were to learn to be more efficient in my work while practicing with programming languages. I wanted to accomplish this by learning enough about writing Java code and Linux commands and the system to improve my coding and debugging techniques. I was also aiming to improve my oral and written communication skills during meetings and small discussions by engaging in technical conversations with coworkers from the Development Team via email and in-person discussions. Finally, my third performance goal for the term was to become more confident in my work while contributing to the software development as well as during general software development procedures by writing dozens of unit tests as well as my own code for numerous classes in the project and not shying away when my tests fail, but rather to learn form my mistakes and implement what I have learned in future coding tasks.

As learning objectives for this work term, I wanted to develop a sound competency in using the Linux operating system by executing hundreds of Linux commands and practising the use of the terminal for following logs of the systems, navigating computer files and committing code changes. I also wanted to have enhanced my software development skills by working on the production of MediaVantage's project and participating in the full development process, including test-driven development and regularly working on various coding projects in Java.

In between working on different projects, I had the opportunity to work with many types of people and try various tasks related to software development. My main duties included working on the MediaVantage project. My primary tasks included writing unit tests to maximize code coverage, fixing bugs and perform refactoring on a few classes, among other things. Lastly, I did a bit of research and development and participated in some of its regression testing.

This report will provide a brief description of public relations, a summary of the MediaVantage product and what clients benefit out of its use. Furthermore, an explanation of the advantages of the agile software development process practiced at MediaVantage will be given, as well as a few aspects that may have caused more challenges. I will also briefly explain other projects that I have worked on and the challenges that came with each one. This report will present observations that were made by me, as well as the lessons that I learned through this experience. To conclude, my work term with MediaVantage has significantly expanded my knowledge in thorough and efficient software development by following the agile development process and its possible impacts on a development team if not executed properly.

# II MediaVantage

## Public Relations

Public relations (PR) are the professional maintenance of any form of public image, whether it is a famous company, organization or a person. PR may include the company, organization or person gaining popularity and/or exposure to audiences by means of topics of public interest and news items that are not paid for by the public. A company’s PR has usually one main goal, which is to persuade the public, investors, partners, employees, and other stakeholders to maintain a certain point of view or opinion (usually a positive one) about said company, its products or business. The most popular forms of PR activities include press releases, conference presentations, receiving industry-level recognition, working with the media outlets, and employee communication.

## What is MediaVantage?

MediaVantage is a web-based application that monitors media mentions and online conversations, aligns corporate teams and measures communications success. Public relations, investor relations and marketing communication professionals around the world rely on MediaVantage to help manage their brands and protect their corporate reputation. Sharing the name of the company, the MediaVantage software solution provides real-time access to TV, radio, print, Internet and social media content. It gives communicators the tools they need to both plan and fine-tune communications strategies, synchronize corporate messages and engage with key stakeholders.

# III Agile Software Development

Agile software development is one of the most common types of software development today among small development teams. Although it is a very precise process with many deadlines, my experience on an agile team has shown me that there are many benefits to using this process and that the results of this process can be quite impressive.

One of the main purposes of using an agile software development process is that it allows quality development while helping others do the same. To quote the *Manifesto for Agile Software Development*, it primarily values “individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to change over following a plan”. The highest priority for an agile team is to satisfy the customer through early and continuous delivery; all while welcoming changing requirements, including changes that need to be made late in development. Business people, stakeholders and developers must work together daily throughout the project in order to deliver working software frequently. The agile process includes providing the environment and support that everyone involved needs and the trust that they will get the job done. Thanks to these principles, there is almost always work that can be done on the project, meaning that there is continuous progress towards an end result by continuously making available new work, whether they be new features, bugs, testing, that ultimately contribute to an end-result of the project.

Agile is heavily dependent on proper communication between all developers, managers and any stakeholders in the project. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation. In the agile development process, the first priority is to get software working right. This is the main measure of progress, which is to say that if a user story is working as expected and there are other user stories that need attention, a new story is often picked up as opposed to making changes to something that already works, of course depending on time constraints. Moreover, there is consistent and continuous attention to technical excellence and good design. Simplicity is key, where simplicity refers to the “art of maximizing the amount of work not done”. An agile team is self-organized, which often results in the best architectures, requirements and designs. The team would also commit to regular intervals of reflections and updates about completed work as well as the work in progress, and remains adaptable to structure changes that would potentially lead to more effective results.

MediaVantage follows a Time-boxed style of the agile approach. Each release of a product is split into a number of iterations, usually somewhere between six or nine iterations. An iteration lasts one week, starting on a Wednesday and ending on the Tuesday of the following week. This is classified as time-boxed working since the focus is on prioritizing and setting date limits on certain features. Contrarily to feature-boxed methods, which evaluate how long a certain set of requirements will take to complete, the time-boxed method is more concerned with what the most important features are and how much can be completed and accepted within the time available. There are three main time-boxes: a release, an iteration, and a day. At the end of every release, new software is deployed to end-users, the next release is planned and the team reflects on the past release in a retrospective to improve the next release. A release is composed of iterations, where at the end of each iteration the team will have working software that is available to be deployed, at least to a testing environment. Similar to an end of release but for short-term, the team reflects on the past iteration and plan the next one.

There may be many challenges associated with agile development with regards to relying on self-organized teams with time constraints. Sometimes, members of the team do not communicate effectively, which can lead to more than one issue. The goal of regular intervals of reflections and updates, such as daily stand-up meetings and weekly iteration planning meetings (IPM), is not only to discuss the progress of the project but it is also an opportunity for developers to ask for assistance if the user story that is assigned to them is too much for them to handle. If information such as that is not conveyed appropriately to the team, more time than necessary could be wasted on this story. It is also possible that there are another developers waiting on the story’s completion in order to work on their own story or to conduct some testing, which could delay the project further. This could potentially lead to delays in the development process, extend the work schedule, and cause unnecessary frustration to not only the developer working on it but the team as a whole.

A release will plan and deliver business value through themes, also known as features, which will be provided by a number of user stories. By the end of each iteration, a number of user stories will be completed and delivered. Each story may contain a number of tasks, usually referring to discrete human activities, which need to be completed before marking the story as complete.

A user story is “a short, simple description of a feature told from the perspective of the person who desires the new capability, usually a user or a customer of the system.” MediaVantage uses a strict system with regards to the organization of user stories by making use of index cards or sticky notes and arranged on a white board. The white board is separated into three main columns: in-progress, dev-complete, and pm-approved. It is the individual developer’s responsibility to ensure that there are cards created for stories that they are working on and that their statuses are updated accordingly. These index cards are used for more than just organization as they also facilitate planning and allow for more discussion about the feature during daily meetings. A user story can be classified as a “good” user story if it is independent, negotiable, valuable, estimable, small and testable. If a user story posses these qualities, it maximizes the efficiency of the group’s workflow as a whole.

User stories need to be independent, that is to say that any user story can be started and completed before any other user story. They should also be able to bounce across system and component boundaries, and can be prioritized in whatever order as needed in order to produce the most business value in each iteration. It is important that user stories only describe what a user can do, and not how the user will do it. The latter should remain open for discussion. The value of a user story is how much it aligns with specific business goals and how it helps to achieve these business goals. The ability to estimate a user story refers to how well the story can be sized relative to other stories. It is important to be able to measure how estimable a story is based on its title, so a proper estimation on the amount of time and effort required finishing the story could be made during the planning stages of an iteration or release. Finally, user stories need to be small enough to be completed within a single iteration (one week), and can be demonstrably determined to be done with specific acceptance criteria.

Agile software development teams are one of the most efficient types of software development teams because they apply practices that allow them to repeatedly deliver working software, and also permit software to be changed in the future. Theoretically, agile practices would allow a team to deploy their software at any moment in time, while guaranteeing that it would be in a working state. Agile software development practices emphasize simplicity, incremental progress and reproducibility while focusing on clean design, routine refactoring, automated testing and continuous integration. These are the fundamentals of the agile process that, when executed correctly and communicated effectively, can lead the team to much productivity and successful products.

I was personally able to witness the product of such a work process with the CloudCity release of the MediaVantage product. Throughout this term, I was able to accomplish all of my work-term objectives by simply participating in the agile development. The required communication forced me to improve and become more comfortable with my oral and written communication skills during stand-up meetings and IPMs, which led to me learning how to work in a large team on an even larger project. Technical conversations with coworkers also lead me to become more confident in my work while contributing to the software development, which was also one of my learning objectives. Through everyday practices, I learned to be more efficient in my work while practicing with programming languages by learning about test-driven development and by adapting to Linux platform to improve my coding and debugging techniques. Since all software developers at MediaVantage work in Linux, I had to learn how to install, use and maintain a version of the Linux operating system on my desktop computer. This being my first challenge on the job, I was able to learn how to use it quickly enough and over the term, I adapted to new shortcuts and operations that I discovered along the way.

# IV Other Projects

During my four months at MediaVantage, I had the chance to explore other areas of work as part of the Development team.

## Research & Development

Although I was technically a software developer with MediaVantage, there was quite a few times where I would not be developing software, in fact. One of the tasks that I participated in a lot during my term with MediaVantage was research and development related (R&D), which almost never required writing any code.

I very much enjoyed participating in R&D tasks with other developers because I liked getting to dig into our project code and system architecture to investigate the pros and cons of certain changes, or to evaluate the amount of effort that would possibly be required to introduce something new in the project.

## Code Bugging

As mentioned above, taking care of bugs in the software is a recurring task, no matter where in the development process the team is, and it was one of the tasks that I performed fairly often. The bugging process at MediaVantage is followed pretty strictly and the Development and Quality Assurance teams are both involved in a regular routine. When someone finds a problem, whether they are on the Development or Quality Assurance team or the problem was reported by the Support team, a bug is filed using Bugzilla, which is Mozilla’s popular Web-based bugtracker and testing tool. The bug reporter must include a precise and descriptive title, steps to reproduce, as well as observed and expected behaviour. Once filed, the bug goes into a the stack, and every Monday at 1pm a Bug Scrub meeting takes place where the QA team as well as a few senior developers go over new issues found by developers, QA and in the field, and set the priority and target of its fix. The priority can range from P1 to P3, P1 being of highest priority and needs to be addressed before the release. The target refers to the release we intend to fix the bug in. Once the bugs are prioritized, developers can assign themselves (or others) to the bug, and turn it into a user story. At that point, the bug is treated similarly to a regular user story, until there is either a fix is implemented or it is deemed “not worth fixing”, based on the amount of effort required, the cost of implementing the fix, the value of its solution and time constraints.

## Quality Assurance

During the final week before the CloudCity release, which was the second of two hardening iterations, we found that there were no more development tasks that we could do, and since the team was not ready to start working on the next release, I had the opportunity to help out the Quality Assurance team in their Regression Testing. This was a very interesting experience for me, as I was able to look at and use the software from a user perspective as opposed to a developer’s perspective.

# V Recommendations

## MediaVantage

I sincerely believe that MediaVantage was a great place for a CO-OP placement. Everyone that I have met or have worked with at this placement was extremely welcoming, encouraging and ready to help and mentor me with anything I needed throughout the term.

In terms of support, I owe a lot of thanks to mainly the senior developers of the team. During the first couple of weeks, and throughout the term, various employees, mainly developers, spent many hours training me and teaching me everything I needed to know to get started, and eventually I gained enough confidence to participate in most of the development tasks alongside the rest of the team. It was a great environment to learn important skills such as how to communicate and collaborate in teams on one large project as well as more technical skills such as how to write unit tests, using Linux, Git, debugging tools, and of course, how to work in an agile development team.

I believe that this was a critical position for me to have as a CO-OP student, as it is very much related to my program and it exposed me to so much of the real word of a professional in the field. I was constantly challenged with new tasks that would push the limits of my skills, and I found that my knowledge of good software development was always constantly increasing as my experience with the company grew. This having been my first development job, I was able to look at software at a completely different angle than I had at my other terms, and really focus a few of the functionalities of the MediaVantage product. For that reason, I would recommend this placement to other junior software engineering students.

## CO-OP Program

For my third work term, I was able to use the resources of the CO-OP office to secure my placement. The CO-OP office made the process of finding exactly the job that I was looking for with MediaVantage very simple and they were very encouraging. Previous help with preparing for an interview helped when it came to all of my interviews with the different candidate placements.

One suggested that I have for the CO-OP office to communicate a little more with their students throughout the work term. The advantage to doing so would be that students will keep in mind that CO-OP is a credited program and it would keep them up-to-date with what’s new in CO-OP as well as with upcoming deadlines.

In general, I found the CO-OP office very reliable for students. Every time I had a question, I called in and got an answer immediately. Since my start with the program, I have highly recommended it!

# VI Conclusion

In conclusion, the Agile Software Development process is clearly one of the most effective processes for developing software when executed efficiently and in an organized manner. After four months of exposure to this process, including a full release cycle, I have come to understand the real importance of proper execution of the agile process and its impact on the software quality and delivery. With the future advancements in new processes and development techniques, it is my opinion that agile will remain a common method with a good chance of minor modifications to the method that would result in even more efficient and effective software development because of its flexibility to requirement modification and its dedication to maintaining a workable status.

Throughout my work term, I was able to relate some concepts and knowledge learned through my classes and personal experience to my daily tasks. Through new challenging projects and assignments, I am able to pick up many new lessons and will incorporate such information into future applications. I had the chance to experience many aspects of software development for one of the company's current major products that not only exercised but also expanded my software engineering skills. I would recommend this position to any other software engineering, computer engineering or computer science student looking for a work term, as it provided an in-depth and hands-on experience with software engineering as a developer.

I would like to sincerely thank all employees of MediaVantage for granting me the opportunity to be a part of their group and to learn from their expertise. I gained valuable hands-on experience with the development of the CloudCity release of MediaVantage as well as I greatly expanded my knowledge about good general software development practices. Most importantly, I gained interpersonal and team skills by working with a brilliant and collaborative team who took the time to train my CO-OP and me peers. I valued my time at MediaVantage and hope to be able to carry on all that I have learned into my future projects

# Bibliography

[1] “Growing Object-Oriented Software Guided by Tests”, Steve Freeman and Nat Pryce, *Addison-Wesley Professional*, 2009

[2] “Manifesto for Agile Software Development”, http://agilemanifesto.org/, visited April 30, 2014

[3] Mark Bason, MediaVantage Senior Software Developer

[4] “MediaVantage – About Us”, http://mediavantage.com/about.html, visited April 30, 2014

[5] “Object-Oriented Software Engineering – Practical software development using UML and Java”, Timothy C. Lethbridge and Robert Laganière, *McGraw-Hill Education*, 2001

[6] Steven Lockey, MediaVantage Senior Software Developer

[7] Tim Bacon, MediaVantage Senior Software Developer

[8] “User Stories and User Story Examples”, Mike Cohn, http://www.mountaingoatsoftware.com/agile/user-stories, visited April 30, 2014

# Appendices

# APPENDIX A: MediaVantage, Log-In

# 

# Supervisor’s Approval

As supervisor of CO-OP student **Samira El-Rayyes**, I, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, certify that, to the best of my knowledge, this report is entirely the student’s work and is free of confidential information to the extent that it can be read by university faculty members.

Signature \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_