Date Received: ...............................................



**ASSIGNMENT COVERSHEET for INDIVIDUAL WRITTEN WORK**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STUDENT ID**  **15882307** | **FIRST NAME**  Shuang | | | **SURNAME**  He | | |
| ***Paper Code/ Name:*** | 409232 Software Development Method | | | | | |
| ***Assignment***  ***Name:*** | Assignment one – Case Study | | | | | |
| ***Lecturer’s***  ***Name:*** | Jim Buchan | | | | | |
| ***Assignment***  ***Due Date:*** | 08/28/15 | ***Date***  ***Submitted:*** | 08/27/15 | | ***No. Words/Pages***  ***(not including quotes):*** | words |

Please read the following and  **tick** to indicate your understanding:

|  |  |  |
| --- | --- | --- |
| 1. | I understand it is my responsibility to keep a copy of my assignment. | **√** |
| 2. | I have signed and read the **Student’s Statement below**. | **√** |
| 3. | I understand that a software programme (Turnitin) that detects plagiarism and copying may be used on my assignment. | **√** |
|  | ***Plagiarism and Dishonesty are methods of cheating for the purposes of General*** |  |
|  | ***Academic Regulations (GAR)*** [***http://www.aut.ac.nz/students/regulations.htm***](http://www.aut.ac.nz/students/regulations.htm) |  |

**Student’s Statement:**

This assessment is entirely my own work and has not been submitted in any other course of study. I have submitted a copy of this assessment to Turnitin, if required.

In this assessment I have acknowledged, to the best of my ability:

 The source of direct quotes from the work of others.

 The ideas of others (includes work from private or professional services, past assessments, other students, books, journals, cut/paste from internet sites and/or other materials)

 The source of diagrams or visual images.

**Signature: Shuang He Date:** 08/27/15

Software Development Method

409232

Assignment One – Case Study

**Lecturer: Jim Buchan**

**Student name: Shuang He**

**Student ID: 15882307**

# Abstract 0.5 page

The report is a case study carried out by interviewing an expert in a middle-size local company. The interviewing questions are mainly focused on software development. Through the case study, the report figure out the success factors in their development process. The development team adopted part of Agile development methodology, Iteration, to get quick feedback from customers which can help them keep the software aligned with business needs. Besides that, the team works very closely, short meetings are held quite often for team member to exchange information which can largely eliminate misunderstanding. Regarding challenges, there are mainly three challenges. Firstly, test-first method is not successfully implemented in the team, which wastes some bandwidth of the development team. Secondly, the test management tool is not efficient enough, it wastes QA’s effort to wait the pages to refresh. Thirdly, no continuous integration involved. Consequently, the deployment is not of high efficiency. Recommendations are given based on those issues.

# Introduction and background (1-2 pages)

1. The important characteristics of the fictional organization 2

The company is a health software development company. It has much cooperation with different medical institutes. There are many small groups in the company and each group works on a separate project. Though the group is small, it contains all the necessary roles for a software development life cycle, such as developers, test analysts, business analysts, project manager and delivery engineers. Some of the groups in the company adopt agile development methods while others use hybrid development methods, the mixture of agile and waterfall development method.

1. The nature of the fictional development team including roles and responsibilities 3

I interviewed an expert of a group which uses the hybrid development method. The team does not follow vigorous agile development method, but it adopts part of the agile process, such as development iterations. Each iteration lasts for three to four weeks. At the end of each iteration, they provide a working application to customers. Regarding the working application, customers can provide feedback and raise new requirements. Therefore, the development team can start a new iteration.

The group includes following roles: 1 project manager, 1 test leader, 2 senior test analysts, 4 developers and 1 deployment engineer. These different roles are in charge of different tasks:

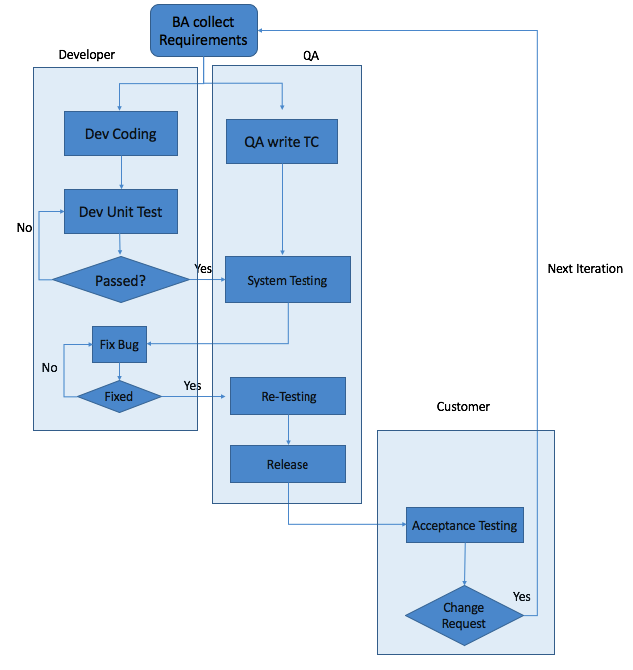
* Project Manager: taking the responsibility of monitoring the development process; communicating with customers; coordinating between different team members; doing internal acceptance testing.
* Business analysts: eliciting requirements from customers; passing the requirements to developers and test analysts; clarifying the ambiguous parts of requirements to other roles;
* Test leader: deploying test version to test analysts; holding partner meeting to check the dependencies between different modules; arrange test priority according abovementioned dependencies. For example, if a module is replied on another model, then the testing work should be arranged in a reasonable order; test leaders also needs to estimate test effort and assign tasks to senior testers.
* Senior test analysts: writing test cases; executing test cases; creating automation test scripts; executing different levels and different types of testing, such as regression testing, system testing, non-function testing.
* Developers: coding for application; implementing unit testing; fix bugs.
* Deployment engineers: deploying application to production environment.

1. A high-level diagram of the software development process with a brief description 3

**Overview of development process:**



**Detailed development process of each iteration:**

****

# Finding and Discussion

1. A description of the methods and supporting tools used throughout the software development process and their purpose. Make the structure of this description very clear. 8
2. Main description of the methodology used in the company

The team used a hybrid development methodology which mixed the agile development methodology with waterfall development methodology. The team adopted iteration in development process which is a concept from agile development methodology. At the same time, in each iteration, they follow the waterfall development method. Regarding the interview results, the team ran very well in such a mechanism. They have already delivered many quite successful applications to various customers.

Iteration is one of the most distinguishing features of Agile development methodology ([Thummadi, Shiv, & Lyytinen, 2011](#_ENREF_1" \o "Thummadi, 2011 #8)). According to the expert interviewed, the most significant benefit of her team to accept iteration in their development process was to elicit feedback from customer as earlier as possible. The life cycle of each iteration was 3 to 4 weeks. During this period, they needed to finish all the development, testing and implementation work. After each iteration, a working application would be provided to customer. Customer would do acceptance testing on it. If the customer found anything unsatisfied , for example, some features were not implemented, then the customer and development team would work together to figure out the reason. If the feature was once raised by the customer, but the development team did not work it out, then a new change request would be raised and the development team could not charge the customer anymore. Otherwise, if it was a new requirement, customer needed to pay for it. Customer raised new requirements and started next development iteration. In addition, if any defects found by customers, they would raise to development team to let the team fix them to meet the business needs. Adopting the short release terms, the development team could ensure that the IT work aligns with business needs. Consequently, it could maximum decrease the project risk. If there was any violation to the requirements or any delay to the whole schedule, the development team can learn from the failures and make up in the next iteration.

Within each iteration, waterfall development method was adopted. At the very beginning, BA negotiated with customers to collect requirements, and then cascaded the requirements to the whole team, both developers and QAs. PM was involved as well, but the PM only focused on a high-level understanding of the requirements. Simultaneously, the developers began to write code and the QAs started to create test plan. They started to work nearly at the same point. When developers finished development work, they would provide a deployment list to QA lead. QA lead was in charge of internal release. He would release the test version 2 to 3 times per week. There was no fixed schedule for the internal release, it depended on the real situation. When QA found defects, they submitted the defects to JIRA defect management system. Developers would fetch defects from the system and began fixing. After fixing, QA did retest. The regression testing would be executed after all the defects were fixed. There was quite a special point in this company, they had both internal acceptance testing and external acceptance testing. For internal acceptance testing, the testing happened within the company. The PM was the main implementer of this work. After PM finished the internal acceptance testing, the application was released to customer’s site by Deployment Engineer. Customer did the external acceptance testing and provided feedback to the development team.

1. There are many supporting tools used in the whole process.

* **Overview of the tools:**

|  |  |
| --- | --- |
| **Development Activities** | **Tools** |
| Coding IDE | Eclipse |
| Maintain Test Case | Xstudio |
| Performance Testing | Jmeter |
| Webservice testing | SoapUI |
| Automation Tools/Framework | Eclipse/Selenium |
| Defect Management Tool | JIRA |
| Communication Tools | Outlook, Skype |

* **Detailed information of abovementioned tools:**

Coding IDE: Eclipse

Eclipse is an integrated development environment (IDE). There are various plug-ins available for the tool. Therefore, many programming languages are support. Such as Java, C, C++, PHP. For the expert’s company, the developers use Java to do programming. Eclipse is free and completely meets their needs.

Maintaining Test Case: XStudio

XStudio is a free test management platform. The tool can handle the complete life-cycle of QA’s work, including tests cases, test reports and defects. However, the company only used it to manage test cases. The expert said that the QAs of her company did not like the tool because of slow response time. They usually needed to wait quite a long time to proceed next step.

Performance Testing: Jmeter

Apache Jmeter is a free load testing tool. QAs use it to do loading testing.

Webservice testing: SoapUI

SoapUI is an open-source web service testing application. QAs use it to do Webservice testing.

Automation Tools/Framework: Selenium/Eclipse

The company uses quite a lot of automation testing in their daily work. They adopted an open-source testing framework, Selenium. Based on the framework, the QA team customises it to meet their needs. Selenium is just a framework, the QA team still needs to find an IDE to support the further development. Because the developer team uses Eclipse to do coding and they give positive feedback on it. Therefore, the QA team also uses Eclipse to do further development on Selenium.

Defect Management Tool: JIRA

JIRA is an open-source issue tracking management tool. The JIRA administrator can configure the functions of JIRA to adjust it to best serve the development team.

Communication Tools: Outlook, Skype, Phone

The team uses various tools to do daily communication. Such as outlook, Skype and phone. Outlook can record all the discussion as evidence. Skype can provide instant communication. Employees can choose any kind of communication tools as they like.

1. The aspect of software development that the team found challenging and want to address. 5

During the development process, the group meets several challenges as below:

* For the new iteration, they mostly follow waterfall development methodology. started with new requirements. BA worked on gathering these new requirements. Developers and QAs learn the requirements and start their work respectively. Part of the testing work happened simultaneously with coding work. For example, when developers are coding, QA will work on creating test cases. However, there are still some limitations which restrict their work efficiency. The problem is that the automation scripts created by QA team are highly relied on a complete application. It means only when the developer team finishes all their coding work and deliver the complete application to testing environment, the QA team can read the attributes of each element on the web page and start writing automation scripts. The effort spent on writing automation scripts is huge. During that period, the development team is relatively idle. It is a waste of bandwidth.
* The test management tool, Xstudio, is not capable enough for the QA team. The expert said that the tool responded too slow. The QA team wasted a lot of time on waiting the page to refresh. Therefore, it hindered the QAs’ initiative to create test cases.
* There was no continuous integration. Therefore, each time, a lot of time was spent on deployment.

# Recommendations and conclusion (1-2 pages)

1. **Your recommendation for the team, to address the challenge you identified. Make sure you are convincing in your argument. Reference theory or empirical evidence in literature to support your recommendation.** 5
2. Test-first method can be adopted. –Reference needed to back this argument
3. A more efficient test management tool is recommended. –Reference needed to back this argument
4. Continuous integration is recommend to –Reference needed to back this argument
5. **Appendix A: prepared questions you asked the industry expert** 3
6. What important development method and practices do you think I should understand to be part of a successful development team?
7. What roles do you think are needed in a successful software development team?
8. I have been told that it is important to treat testing as important as coding through using methods like automated regression tests, automated builds, and a test-first approach. What do you think?
9. We learn about practices like standup meetings, sprint planning and sprint reviews to keep in touch with other team members and the client in some projects. What would you recommend to keep in touch and get feedback during development?
10. We have been told that you review the team process in retrospective meetings after every sprint and it's ok to experiment with the process and make changes. This is part of continuous learning for the team. What do you think works well to keep the team learning?
11. There are lots of tools to help with software development methods, like continuous integration tools. What do you think the important tools are to support development team over the development lifecycle?
12. What do you think the main success factors are, with respect to methods and tools in developing software?
13. In your opinion, what are the main challenges related to software development methods and tools I should learn about?
14. **Appendix B: Your industry expert’s opinion of the critical success factors and the main challenges in team-based software development.** 3

**Critical success factors:**

1. The team adopted part of the agile development methodology. For example, they provided a working software to customers in every 4 weeks. Accordingly, the development team can get the feedback from the customers immediately. Thus, such practice decreases the risk of creating a software that does not meet customers’ requirements. When customers found any gap between the requirements and the software, they would raise the Change Requests. Those Change Requests helped to correct the software.
2. Different team members closely work together. The information can be exchanged instantly. Therefore, there is very little misunderstanding between team members. In the expert’s company, there are variety of ways to communicate, such as outlook, skype, phone and face to face talk. The group members have the authority to choose the most efficient way to exchange information. Besides those individual communication, the team leaders or PM also held meeting to let the whole team discuss together. Such meetings were quite effective and efficient to keep all the people in the same page.

**Main challenges:**

1. According to the experts, the most challenging part of the development life cycle was the implementation of test-first methodology. Currently, testing, especially the automation test scripts need to wait the whole development finished. It was quite a waste of time. If the test cases and automation test scripts can be generated simultaneously with coding work, the whole development time box can be shortened.
2. Test management tools were not capable enough, it could not respond instantly.
3. There is continuous integration. Regarding each deployment, developer needed to provide a complete list of deployment items, then the test lead needed to follow the list to deploy. It usually took quite a long time to finish a deployment. In addition, if the test lead made some mistake about following the deployment list, the deployment would fail. A lot of efforts would be required to re-do the deployment.
4. **Appendix C: Your answers to each curiosity question based on the consultation with the industry expert** 5

**Questions related to organizational aspects of software development:**

1. How requirements are elicited(discovered).
2. How shared understanding of requirements (elaboration, clarification) is done.
3. What is done for release planning and scheduling.
4. How the order of features to work on(priority) is agreed on?
5. How the expected effort to develop features is estimated for planning (eg planning poker)?
6. How the expected effort to development is monitored?
7. How the team is organized – what roles and responsibilities.
8. How the team keeps in touch with each other?
9. How the team keeps in touch with the clients (Product owner).
10. What the team’s reaction is to changes in features.
11. Do the team experiment with process and practices?
12. Do the team reflecting and continuously learn?

**Questions related to technical development tasks:**

1. How iterative and incremental development is done (eg three-week sprints)?
2. How requirements are documented/represented (eg user stories)?
3. How changes to requirements are handled?
4. How testing is done and what levels of testing – unit, regression, integration, acceptance, performance, load. What testing is automated?
5. Is exploratory testing done?
6. Are any quality metrics (measures) tracked?
7. Is a test-first approach used?
8. How is the build managed – any automation? Frequency of build?
9. How is continuous integration achieved?
10. How are non-functional or quality requirements managed?
11. What programming languages are used?
12. Any other important tools?

**5. Reference**

Thummadi, B. V., Shiv, O., & Lyytinen, K. (2011, 7-13 Aug. 2011). Enacted Routines in Agile and Waterfall Processes Symposium conducted at the meeting of the Agile Conference (AGILE), 2011 doi:10.1109/AGILE.2011.29