

Manoj Vasa

Dr. Jody Paul

CS 4250 - 02 Software Engineering Principles

11 December 2017

Executive Summary

Even through the variations of the experience that each member in a team has with different tools and applications, communication is essential for efficient progress for these projects. Following a process model like SCRUM Agile provides some support in establishing some clear channels of communication and stating the expectations of each team member. Although some team members are more capable of accomplishing their goal than others, all members must be ready to assist those that are more unfamiliar with the tools and applications used. Through this clear communication within the team and with the stakeholder, the project can be successful.

Introduction

Every new team allocates time to develop their trust and channels of communication. The beginnings of projects consist of these activities that lead teams to collaborate effectively. Understanding the expectations of a project is crucial to the communication within the team. After this phase, my team performed their tasks to progress this project following a process model without deviating from its core values. These core values consisted of clear and constant communication both with the client and with the members of the team. This collaboration resulted in the deliverable that we submitted as a team. We struggled separately in the beginning, but rose from the ashes as we developed our reliable channels of communication and clear documentation that guided our progress for this project.

Software Development

The first week of this project proved to be difficult as we struggled to follow any process model to guide us through the project. We failed to meet some requirements that needed to be fulfilled when developing this product. As a group, we decided to follow the software development life cycle methodology of scrum agile. This lead us in a better direction. After constructing user stories and validating them with the stakeholder we noticed that this was an essential phase of the process. This ensures that the right product is being built. Without this communication with the stakeholder, the product created may not have the functionalities that the stakeholders originally desired. We've noticed this exact scenario when we created a product without validating any requirements with the current stakeholder. We were creating the wrong product. The scrum meetings that were held every week contributed heavily to the progress that was made for the project. These meetings often lead to working together to finish a task rather than allocating various pieces of the task amongst us and separating instantly. I've further noticed that meeting with the team face-to-face produces the best results. I began noticing the effects of clear communication within our team and with our client.

Although these artifacts weren't present in the final deliverable, the design of the project was accounted for by the construction of UML diagrams that displayed the methods and classes handling the suite's menu and its tools. This design consists of the framework for the intended development of the rest of the features of the product. Through this phase, we were in the process of writing acceptance tests that tested the obvious paths of the code. This consisted of checking the existence and actions produced by invoking the extensions installed in the suite. These acceptance tests assessed the quality of the product. These assessments ensured that the suite offered what was promised in the documentation of the product. Through these acceptance

tests we can be sure that the suite at least does what it is intended to do with consistent results. We assessed the product through black-box testing. In this way, we could ensure that the product was operating just as specified by the requirements regardless of the manner of the implementation. The lack of assessment of boundary conditions may prove to be problematic to future users. Assessing the quality of the product results in confirming the correctness of the product thus far.

Team Dynamics

Experience within a team varies among everyone. Some of these team members may be more experienced in the tools and the applications being used in a project than others. Working on these projects requires these individuals to assist the rest of the team in understanding how to use these tools. I've witnessed this exact scenario in my team for this project. One of my team members was far more comfortable with using build environments like Gradle. This team member took their time to create a video and post to our communication channel to assist others in the team that may be having difficulty in using this build for this project. Another aspect of team dynamics that is of consideration is the division of work among the team. I often notice that some team members receive and deliver larger amounts in deliverables for each sprint than others. Some team members carry more weight than others. This weight is difficult to distribute evenly among other team members due to certain circumstances. For example, the major contribution to the development of the GUI was apparent from only one individual from my team. This member seemed to have the most amount of experience in developing GUIs. This uneven distribution may cause some team members to carry a heavier burden than others.

Nonetheless, communication was key in performing well in these projects. Although we didn't meet in person too often, my team communicated through slack and through the issues

assigned in Redmine. Even here, I've noticed variations in the contributions by the various team members. Each person seemed to be active to a different degree. One of the people in my team was very active, but others didn't see the message until the next day. Some other team members communicated through the channel once a week even though they were meeting their goals each sprint. These unique variations in activity will be present in all team projects. However, having some communication before meetings proved to be completely essential to the progression of the suite project.

Example of Significant Contribution

My biggest contribution to this project consisted of the requirements engineering performed somewhat late in the project. I contributed to the user stories that lead to our project finally making progress in terms of the deliverables we had every sprint. Developing the user stories and assessing the importance of each story gave our group a sense of direction and a goal for each sprint. The process of creating the user stories seemed to consume most of the time for my group. The development phase that followed afterwards produced some deliverables that added some features for the product. With the initial design and creation of the user stories, my team could finally assign tasks for each sprint for the rest of the project. The user stories assisted us in knowing what to do for the little time we had left. Such tasks required researching into the functionality of some of the tools we were going to implement into the suite and creating a user-friendly menu. This GUI improvement and tools analysis lead to further progress on what we need to accomplish on the project. Before we developed our user stories we seemed to wander aimlessly as each of us tried to accomplish our individual tasks that contributed minimal progress every meeting. The importance of requirements engineering became abundantly clear to me as I took on contributing to this task.

Conclusion

Through the course of this project and many others I've worked on in the past, I've been exposed to various tools, applications, and problems that contribute to the accumulating knowledge that'll surely help me in the future. Having this experience with the tools and applications associated with this project will reduce the amount of time it'll take me to find my ground in other projects. Furthermore, the team dynamics assessments have informed me of the importance of understanding that every team is unique and every project will consist of various approaches. Developing clear communication within the team certainly assists the team in collaborating in a project. The disorganization in our team before our creation of requirements revealed to me the importance of requirements engineering. This essential step in every project contributes to establishing clear communication between both your team and with your stakeholders.

Works Cited