Software Process Improvement

Chilla Kartheek
Department of Software Engineering
9304163778
BTH, KARLSKRONA
chilla.kartheek87@gmail.com

Abstract- The paper mainly focuses on the software Process improvement. The experiences within the software projects are discussed and issues hindrances related to the process improvement are discussed as well. Jacobs's recommendations along with different elements in APIM to perform the improvements are described.

I. INTRODUCTION

The author doesn't have any industrial experience with respect to software projects. The authors however have a very constrained knowledge experience pool within academics in developing software projects and a few experiences related to developing software projects via freelancers. These projects involved both the teamwork and individual work by the author. The examples that are stated below from the experiences of the author, where he felt the need for improvement within the process of the software projects that author developed.

II. EXAMPLES RELATED TO PROCESS IMPROVEMENT

Example1: Delivering the Groceries to the customer via home delivery, using web portal.

Author is currently working in a project of team 8, in which we have to develop working software within time frame. As part of this study project we need to come up with the detailed requirements that are need for the project. The project is one of its kind and unique and no other website have the requirements feature we planned to insert. We had to come up with detailed plan to be followed ahead alongside with the development model that would be followed to implement the study project.

- Author felt the process improvement should have done in selecting the life cycle model within the study project.
- To stick on to the topic of requirements what we planned to insert instead of changing the entire study project plan itself as it is believed that the output is simple.

*Process followed***:** Initially a brainstorming section is conducted where the team members said the requirements of the topic are interesting and it is good to start work on the project by coming up with detailed project plan and the corresponding life cycle model is discussed as well within the team based on the constraints that would affect the project. Everyone gave his or her preferences and opinions. The life cycle model and the product to be developed are selected based on the team member opinion.

*Needed Improvement***:** Author felt the process mentioned above can be improved instead of changing the entire requirements of the project to be less complicated and which is one mistake done in the middle of the project that helps us to complete the project but should have done in the brainstorming phase itself to look into all possible constraints. The constraint here was technical skills, team members if they had opened up in the initial phases we would have opted for alternatives unlike in the middle of development phase. Author also felt there is a need in the process improvement as we are in the middle of the development phase and we already choose the incorrect life cycle model. Everyone should consider the entire requirements and constraints before selecting the model. Choosing the model convenient to the project is very important for adaptability reasons rather than choosing the model in which they fell they are flexible to work on. The actual process could have been to select all possible constraints that are applicable to the given project. Later list of possible life cycle models should be decided based on prioritization. Then if possible a percent showing the suitability of which life cycle model when constraints are given as input the corresponding response of life cycle model is known. This way it uplifts the confidence in selecting right life cycle model.

*Hindrances***:** The possible hindrances for the process improvement that is specified may be:

- Resistance from different people: As we are in the middle of the project in the development phase. Giving more time in selecting the life cycle model by giving detailed plan for it would consume a lot of time team members would not accept that process improvement procedure due to time constraint.

- Different people have different skills and choices: The people and their skills in developing the web applications should have been discussed earlier everyone believed they know programming no one come up with genuine pinion their skill assessment ability. In the middle of the project they were honest in accepting that they don't know the skill. Voting is done to change the task too much simpler and a different project itself instead of what was actually planned. Yes, different people have different opinions and skills but should be assessed in the beginning of the project itself and stick on with it. Unlike put voting in the middle of the project and change the entire project as it is difficult to achieve.

*Example 2: Dynamic Food management system using the foodinfo.org portal***.**

During the requirement analysis phase, it was identified that team members are lacking in the required language skill for the development of the web portal/web application. This means it is observed that there is need for training to be done as part of the project. There are different levels of projects in which team members have different skills. In this project when it comes to language skills only 4 out of 8 members are expertise with php scripting language the other 4 know just the basis stuff. So they thought training would be good choice to bring all the people to one common platform.

Process followed:

- Identify the issues and area in which team need training.

- Provide the resources to the team for practice.

*Needed Improvement***:** above process could have been improved in a better way because in this case the training turned up to be ineffective. Instead of giving training to the team members it would have been nicer to use the pair programming technique in which one expertise team member out 4 who are identified as skilled people would team up with team member who have less experience in developing the project. In this way the time constraint is reduced rather than in training in which it consumed 40 % of the total time in development.

- Identify skilled experts in PHP.

- Team them up with the less experienced once.

- Allocate tasks slowly and once it is finished give them the next task.

- Allocate the resources required for pair programming work.

*Hindrances***:** The possible hindrances would occur.

- Time factor: giving training to the people within the project deadline is hard to schedule. Any up and down would lead to rescheduling.

- Deadlines: unable to meet the last submission deadline which is what happened in this case and lead to second submission.

- Different opinions should be given value: First should listen to all possible ways and strategies to solve the experience level in php problem. Instead of that considering training is the only option is wrong way of assessing. Should give importance to everyone opinion.

Example 3: Static and dynamic analysis of Sudoku.

This project was run as part of team comprised of 6 members'. The main aim the project is to do static and dynamic analysis of the Sudoku code. This project is part of the verification and validation subject. Use the static code analysis tool anyone to find the defects in the code. Different tools have different capabilities and they give warning within particular loc. Then we need to decide whether they are true positive or false positive. This reviewing of the false positive and true positive is done by the team. The automated static code analysis tool that we used was PMD tool.

*Process Followed***:**

Based on the awareness that we have on the static code analysis tools we found out some tools like soot, PMD, Find bugs, jtest. The task given is to identify the tools that are

useful in static code analysis. Then used the PMD tool, which we believed would give effective results.

*Need for process improvement***:** The actual improvement process could have been to select the code of Sudoku as input and find all possible defects. Thus by giving the code as input to the tool we could have found all the possible different defects and see which tool is efficient by analyzing the results. This would have been an effective method instead of selecting the tool from our previous experience that it will work effectively.

*Hindrances***:** The following hindrances could have occurred in the project.

- Time: Time had an impact on selecting the tools to finish the defect detection fast.
- Complexity: Many team members were in fear to indulge the time in dealing with complex tools that can give efficient defect detecting solutions.

III. PROCESS IMPROVEMENT USING APIM

To implement the APIM for process improvement author would like to use example 1 and example 2 [1].

Process improvement for example 1.

In this example group of three team members were mainly involved in requirements gathering and designing the blueprints for the web portal. The model that our team followed to gather the requirements is wrong, we need to follow new approach i.e., follow new approach to define requirements as we did not follow appropriate process improvement. Author felt the need for improvement in the process and thus using the Jacobs APIM guidelines [1] would like to implement them.

Basically there are three phases namely.

1. Pre maturity phase (PrMP).

Note for all the following process we apply TDD approach (test driven approach)

*Launch***:**

The process or the new approach in order to be implemented it should have a defined motivation. For clear understanding purpose, Author would like to describe the goals for process improvement first. These goals can serve as motivation and vision for the process outcome. There we can specify what are the things that are going to be achieved by implementing this procedure. We should specify as per the experience level whether to consult any expertise in field of requirements to get clarity. The activities that are to be arrived in the launch phase for this example are:

- Set correct goals to adopt to the new life cycle model and the new requirements for the project.
- Study the literature behind success of such projects implemented in given new approach way and provide some empirical reasoning.
- Flexible schedule.
- To consult the requirement analyst if at all needed to know the views and opinions on complexities.

*Planning***:**

For the given example the author would like to provide the plan or action plan and updated it iteratively. If at all having any actions that involve listing short-term goals, new approaches, risk management plan, issues resolving plan can be updated and track down easily to look if everything is going smoothly. For this example, the following activities are used.

- Develop a plan that include short term goals of prioritizing the requirements based on which are important and which are not. At the same time to prioritize which life cycle model to be implemented.
- Include issue resolution and risk management plan.

2. Maturity phase MP.

*Awareness***:** The awareness program is to know where we are in the current project plan. Assessment plans are very important to know where we are. If we don't know where we are in the map, then it is useless in looking for short cuts. For the given example the following activities are:

- To track via maintain index.
- To conduct interview among team members about the current requirements gathered.
- Perform assessment about progress regularly.

*Triage***:** It is important to always separate which is important from those, which are not important. Triage is the process of sorting based on specific importance given to the action. This is important as the time is limited and every project has deadlines. There are usually three ways to prioritize like business, project and process improvement goals. For the current example the following activities are:

- Prioritization based on force field analysis can be implemented.
- Select the criteria or strategy that can be used to prioritize the goals.

*Resolution***.** After prioritizing the task, then the next important step is to distribute them among the team members or individual to implement the task. The progress is tracked down to check if it meets the decision-making or actual purpose.

- Action taken, progress both should be track down.

*Training***.** After finding the solution, it should be implemented. Before implementation it should be tested. All the developers needed to be trained to give them awareness about the new action that is developed. For this example, the activities are.

- Evaluate training progress among individuals.
- Offsite training should be done as most of the time team members work in distributed team. At one location.

*Deployment***.** This is a testing phase where the activities for the example would be.

- Develop a plan and execute the solution.
- See if there is any need for improvement.

*Trial***.** After the deployment phase the improvements need to be studied to accept them or to still improvise them.

- See if the requirements are enough to begin next phase alongside see if the life cycle model is suitable for the development by giving input.
- If yes, the process improvement is correct. If no, then document the things that went wrong. Repeat the maturity phase.

3. Post maturity phase (PoMp).

*Appraise***.** After the process improvement check if the current life cycle model is giving appropriate results in different projects.

*Improvise***.** New practices came every day implanting them long with the current life cycle development model can be done in step by step

1. Process improvement for example 2.

In example 2 the team members are lacking the language skills. Author would like to use the Pair programming here to balance the expertise with the less skilled team member to balance and develop the web portal.

Launch

- Identify the whether all the people are ok with the pair programming procedure. Identify if there is a real need among the team members to implement the pair programming.
- Set the goals for the process improvement. That is see that every pair have one expertise and one lesser skilled team member.

Planning

- An action plan that tell team collaboration, any misunderstand, risk management plan. Solutions related to plan risk are stated clearly.
- Make sure everybody accepts this process improvement and document him or her in detail.
- Give them or each pair short-term goals and task to be implemented and all the tasks are documented.

Maturity phase.

*Awareness***.**

- Be aware of where the team pair stands in terms of short-term goals. If any fluctuations within pair, try to resolve them before moving forward.
- Conduct interviews and bring awareness or get an idea about how the pair programming is whether helping them to develop the product or any difficulties'.

*Triage***.**

- Identify most important task or code lines that has to be developed with more precession.
- In here the competition climate should be developed among pair teams to assess the effective team performance to know their collaboration and output results perfection dues to implementing the process using pair programming.

*Resolution***.**

- After the triage phase is done we have the important actions and team who can perform well. Discuss and conduct brainstorming section on how to implement the project web portal.
- Report the issues that arise within the pair further if any.

*Deployment***.**

- Check if the team pair is efficient in implementing the task. Conducting a pilot study does this.

*Trial***.**

- If the results are good in terms of working together in collaboration and meeting the task requirements. Then the process improvement model is good.

Post maturity phase

*Appraise***.**

- The effectiveness of this team can be implemented in using the same pair in new different project task and assess the pair performance.

Improvise.

- This process should be continuous and in small steps.

IV. TOOLS FOR IMPROVEMENT

The tools that are being used to implement the APIM strategies are.

Work break down structure: This is useful in coordination among team as the structure is hierarchical and tracking down is easy. It mainly comprises of tasks that has to be performed in the process.

Checklist: It is important to have a checklist about the tasks that are remaining and the tasks that are completed.

Major activity action plan: this help to know the most important activity of all and to implement it first among the prioritized list.

Overall process improvement status plan: This help to track down the risk that are raised and the solution to the issues that arise. Overall process improvement progress and assessment that can be mutually benefited among team pair mainly.

V. APPLICATION OF GQM+ STRATEGIES

Generally, there are seven GQM+ strategies that can be implemented. I considered example 3 to implement the GQM+ strategies [2]. The following GQM+ strategies are used for process improvement.

Initialize: Firstly, defining the scope of the problem domain is very important. The scrum master usually defines the scope of the project. Scrum master allocates the team members tasks individually. The team is devoted to improve the organizational strategies Here the scope of the problem is to find the defects in the Sudoku code. So, TDD approach where all the relevant tools are to be gathered to perform the test for finding the defects. The team should perform the process improvement by following the GQM+ strategies are given below.

Characterize environment: The scope of the project is to find the defects in the code by using the static code analysis. The strategies required to develop the project are discussed here by conducting meeting. In place for the meeting and the required resources are discussed among team as well.

Define goal, strategies and measurement. Here the goal of the project is to find the defects in the code using the static code analysis. The scrum master defines the goal. The hardware and software required for development is also considered. The main goal here is the process improvement. Thus strategy is to break down all the available static tools to estimate the defects and find the efficient one based on the tool evaluation. Measurements of the previous number of defects are to be noted t identify and compare with the defects detected in the new approach.

Plan Grid implementation. Here the scrum master describes the characteristics of the project The main motive here is in previous section the GQM+ grid plan that is all the tools that are available to test for finding the defects are identified. Here the GQM+ grid plan is updatesd as well using the online open source tools. The individual contribution improvement can also be done using these tools.

Execute plan. The grid plans, which are discussed in initial phase, are implemented. Here the strategy is implemented to find the defects that are present in the code. All the tools are used as base and considering all the tools does testing. The time log sheet is also maintained alongside with documentation (data collection) to know the work hours of individual person.

An**alyze outcomes:** All the data that is gathered is analyzed in here. Crosschecking is done with respect to the process improvement results with the number of defects it can identify.

Improvements: The data code detection can be stored in the GitHub. All the data collected in the defect detection can be stored in the excel sheets. Thus the team members can change or compare the tools and their corresponding the number of defects it can identify to find the efficient once.

VI. SIMILARITIES AND DIFFERENCES BETWEEN TEO APPROACHES

Similarities:

- Both are iterative in nature, both stress on current status of the project.
- Both help to improve the quality of the process.
- They have the action plan at the beginning.

Differences:

- GQM+ strategy use GQM concept whereas APIM use agile.
- APIM have three-phase execution whereas GQM+ has only one.

REFERENCES

[1] D. Jacobs, *Accelerating process improvement using agile techniques*. 2004.

[2] V. Basili, J. Heidrich, M. Lindvall, J. Münch, M. Regardie, and A. Trendowicz, "GQM+Strategies - Aligning business strategies with software measurement," *Proc. - 1st Int. Symp. Empir. Softw. Eng. Meas. ESEM 2007*, pp. 488–490, 2007.