

Process Improvement

Sai Sandeep Chikkala
M.Sc. in Software Engineering
Blekinge Institute of Technology
Karlskrona, Sweden
saca14@bth.se

Abstract—This paper is a report of my perceptions on different aspects of software quality. Each section in this report refers to a question regarding software quality which is followed by an answer with reference to my opinions and experience.

I. MY EXPERIENCE IN SOFTWARE PROJECTS

My experience in developing software projects is pretty much limited to few study projects and free lancing projects. I have no industrial experience in this context. I worked in teams and also individually. I worked in both co-located and distributed teams. So, the examples that I state below are from my experiences where I felt a need for improvement in a process that I followed in the projects.

II. EXAMPLES

A. Example1:

Scenario: I am working in a project of team eight, in which we have to develop a working software in stipulated time. This is a study project and we are given the liberty to set the requirements for the software. We need to come up with a detailed project plan ahead and have to specify the lifecycle model we follow for developing the software. We followed a bit vague process to choose the lifecycle model for our project. I felt that the process we followed to choose the lifecycle model could be improved to end up with better results.

Process followed: We held a brainstorming session to decide upon the lifecycle model. We walked through the constraints that would affect the project. Then everyone expressed their opinion and gave their preference. Then we choose the model for which most people opted for.

Needed Improvement: I felt the process mentioned above could be improved because now we are in the middle of development and we already feel that we chose inappropriate lifecycle model. We overlooked few constraints that would have more weight on the project. Everyone opted the one considering their convenience and familiarity with the model rather than its adaptability for this project. I felt that there could be a process where all the constraints that would affect project have to be listed. Then they are all to be given weights i.e. prioritized in context of the current project. Then a list of possible lifecycle models has to be prepared. Evaluate each model with the constraints as input. Its better if there is a

objective measure to show the percent suitability of the model to this project. Then we could confidently choose the model that is more suitable.

Hindrances: The possible hindrances for the specified improvement of this process could be:

- Resistance from team members: Since we had strict deadlines to meet in the project, team may not encourage following this process as it would consume more time. Since this is not a development activity, team may not feel the need for following a detailed process in selecting a lifecycle model.
- Different people can have different opinions: The improved process demands the opinions from every person regarding the prioritization. So, there could be conflicts regarding others opinions.

B. Example2:

Scenario: I was part of a team of three who worked for free lancing projects. Coding and testing are the major activities we did for most of the projects. Customers were different for all the projects. After every project we asked for the customer feedback. Most of the customers commented that there are significant amount of defects in the delivered product which triggered further testing and rework. In order to overcome this problem we chose to follow different approach for coding i.e. Test Driven Development (TDD). But then the results were even worse. We missed the major deadlines. I felt that we did not do the adaption process properly. Improvements can be made to the process we followed to adapt a new methodology.

Process followed: After analyzing the feedback of our customers, we understood that the delivered project must have low defect density. Then we blindly followed an advice from a consultant and decided to work on the next project with TDD approach in order to lower the defects.

Needed Improvement: When we decided to take a shift from current method, I felt the following steps would have been followed to facilitate the adaption process.

- Analyze the customer feedback and identify the need for adapting new method.
- Browse through the list of methods that serves purpose.
- Weigh the Pros and Cons of every method in context of the project environment (Do some research).

- Make a Mock project with the new method.
- Analyze results of mock project and repeat step 3 if needed.

Possible Hindrances:

- Being Lazy : People can be very lazy to do the research that is needed to weigh the Pros and Cons of every method.
- Cost and Time: To make a mock project it takes time and there will be no funding for it. It's a demotivating factor to work on a project with no specific rewards (incentives).
- Resource availability: There may be insufficient research on a method or there are no empirical evidences regarding the success of the method in particular environment.

C. Example3:

Scenario: This example refers to a project in which we identified a need for training the team members. Team members have varying experience and different skills. Team wanted to use php in which all team members lack experience. Also there are other activities in the project in which team members have varying level of expertise. So, we thought to plan training sessions so that all members could reach same level of expertise.

Process followed:

- Identify issues in which team needs training.
- Identify people who can train the team
- Provide resources to the team members so they can practice.

Needed Improvement: Above mentioned process has to be subjected to improvements because the training ended up to be ineffective. Suggested improvements:

- Provide hands on training
- Identify mentors and mentees for every issue of training
- Plan training sessions ahead and provide the relevant resources for mentees to learn and practice.
- Check if all team members reached same level of proficiency. Have some metrics that track training progress or its results.

Hindrances: The possible hindrances in improving this process could be :

- Participants may not be motivated since training session would not involve any real development.
- Resources: Lack of proper resources to train the people. Less number of mentees having inexperienced teaching skills.

- Time: Hands on training requires more attention and consumes a lot of time which would hinder in avoiding the training process.

III. PROCESS IMPROVEMENTS USING APIM:

I would like to make process improvements to scenario2 and scenario3 using APIM.

1. Improving the process for Scenario 2:

In this example group of three did majorly coding and testing wanted to shift our approach i.e. adopt a new practice to overcome the defect density. We adopted the model and the results were unsuccessful. I think we did not follow correct process to adopt the new approach. I felt the need for improvement in that process and in this section I want to improve the process following the APIM guidelines [1].

Pre-Maturity Phase:

Launch:

In order to clarify what we are going to do, I would first establish the goals for the improvement. In there it specifies why we need to do this process and what we want to achieve by implementing this process. Clear goals can serve as motivation and also provide vision to the process. It should be specified in the goals if there is any need for the consulting expertise. So, for the given scenario the activities that are to be carried in the Launch phase are:

- Set goals to successfully adopt the TDD approach
- Identify the motivation for the shift and give empirical reasoning to it.
- Identify need to consult an TDD practitioner or advocate to get the firsthand views and techniques.
- Have a flexible schedule for the improvement activities.

Planning:

In order to ensure the goals implement smoothly and as expected it is important to compose an action plan with only needed specifications. The plan must be updated iteratively. Any relevant actions i.e. mentioning short term goals, issue resolutions planning, and other risk management planning has to be mentioned. For the given scenario the planning activities are as follows:

- Develop a plan which comprises short term goals i.e. implementations of TDD in pilots
- Include issue resolution and risk management planning.
- Perform Force field analysis which helps in identifying the driving and opposing forces for adopting TDD

Maturity Phase:

Awareness:

Awareness is knowing your current position. You can go where you want to only if you know where you are. So, assessment plans are formed to know where you stand currently. The activities that could be carried out for this section in the current scenario:

- Conduct interviews with the team members to understand what they think about the current coding practices.
- Perform initial mini assessment.
- To record the progress of process improvement use an assessment tracker i.e. index

Triage:

Triage is about sorting the actions that are to be carried based on specific criteria. It's important to sort the actions depending upon its importance since there is limited time factor to perform the process improvement. Basing on three goals i.e. business, project and process goals prioritization can take place. The actions in triage for the scenario:

- Select the criteria that are to be sorted based on the goals
- In here making a pilot study is important and also carrying the force field analysis is critical. So, these actions need to be performed first.

Resolution:

So after triage we have the prioritized tasks and they are to be assigned to team or individual to deal with the task. The task should be traced back to an event in action plan mentioning its characteristics. The activities in this scenario could be:

- The action taken to solve the issue and the progress has to be recorded.

Training:

After resolution there is a solution that can be implemented but has to be tested first. So, training has to be provided which would make developers familiar with the system. The activities in training:

- Offsite training can be provided to developers since most of the time they would be working as distributed team.
- Evaluation of the training session is to be conducted.

Deployment:

This is kind of testing phase. Following are the activities:

- Develop a pilot and execute the solution.
- Study the effects and see if there are improvements.

Trial:

After deployment the effects of the improvement has to be studied. Then decisions can be taken to either accept or refute the improvement.

- See if TDD approach for pilot is effective in reducing defects.
- If yes, deploy the process improvement. If no, document what went wrong and repeat the maturity phase.

Post maturity Phase:

Appraise:

- Perform projects with the TDD approaches and check to see the enhanced process improvements in terms of reduced defects.

Improve:

- After adopting TDD continuous improvements could be performed but has to be performed in small steps. New best practices may arise in the field and

adopting or integrating other practices with TDD can become the subject of further improvement.

2. Improving process for scenario 3:

In this scenario the training has to be given to the project team in order to reach same level of expertise. There was very vague process which is almost ad hoc that has been followed which proved to be not so effective.

Pre-Maturity Phase:

Launch:

- Identify the need for training. So, needed to document the current skills and expertise which serve as base to identify if there is a difference in levels which would induce the need for training the team members.
- Set the goals of the process improvement. i.e. the training should bring everyone to the same level to same grounds.

Planning:

- An action plan has to be developed. Inclusion of team communication plans, Risk management plans, issue resolution plan has to be done.
- Make sure the plan incorporates major details i.e. promoting more hands on training.

Maturity Phase:

Awareness:

- Be aware regarding where the team stands in regard with the training sessions. So, record the current position or trends in training and also the needed improvement that has to be done.
- Conduct mini assessments through conducting interviews and obtain data regarding current training process.

Triage:

- Identify the most important actions that has to be carried out for the improvements.
- In here, it is important that the training is made sure to be hands on and also enable a metric to assess the effectiveness of training session.

Resolution:

- Soon after triage we have the most important actions that can be performed. A hand on training is the first concern and the ways in which that can be done effectively is to be decided by performing brainstorming.
- Report the status and issues with the given task.

Deployment:

- Pilot study is made choosing a sample population and a topic for training. Check to see if the implementation of hands on training is efficient by comparing it upon a metric i.e. team performance.

Trial:

- If it shows good result after piloting a decision can be made to deploy and the process could be followed.

Post maturity phase:

Appraise:

- The effectiveness of this improvement can be observed by measuring how effective the new training process is.

Improve:

- The improvement process should be continuous and in small steps.

IV. TOOLS USED FOR THE IMPROVEMENTS:

The tools being used to make perform APIM for the above mentioned three scenarios are:

- **Work-breakdown structure:**
WBS will show the list of all the tasks that has to be performed in the process. And it is hierarchical and helps to maintain coordination among the tasks.
- **Schedule Checklist:**
It is important to check the actual happenings against the planned activities in action plan and check the completed activities in checklist.
- **Overall process improvement action plan:**
This helps in formulating the plan for overall process. This plan will have only the most specific concerns and includes risk management plan, issue resolution plan. This serves as basis for having the team working towards same goal.
- **Major activity action plan:**
This will comprise only major or most prioritized activity and the plans for their execution. This helps in scheduling the most important activity first.

V. APPLICATION OF GQM+ STRATEGIES:

Scenario 2 is improved through GQM+ strategies, they are as follows [2] :

In scenario 2, team has opted a shift in their coding practice from traditional to TDD. They did not obtain perceived results. So, applying GQM+ strategies for process improvement of adaption of TDD.

GQM+ STRATEGIES:

- **Initialize:**
Specify the scope of the problem domain. i.e. development is done by following the TDD approach and the results are measured in terms of defect density. Team is trained for the process improvement in being performed by following the GQM+ strategies.
- **Characterize environment:**
The scope of application environment is for web application project. The initial specified scope is the unchanged.
- **Define Goal, Strategies and Measurements:**
The goal is to improve the process of adoption. Strategies that are relevant to the goal could be

training the team, making mock projects and measure the efficiency. Number of defects before release is the measure taken to validate the efficiency of new approach.

- **Plan Grid implementation:**

In the previous section GQM+ grid is implemented formed and here the plan to implement the grid is described. i.e how to apply the strategies in order to obtain data.

- **Execute Plans:**

Grid plan is implemented here. The various strategies are being implemented here. The strategy is to check the current coding practice in terms of defect density. This helps to know where the organization stands.

- **Analyze outcomes:**

Executing the plans will result in obtaining related data. i.e. data regarding the defect density. Hence it can be analyzed and further improvements to be made can be inferred.

- **Package improvements:**

Revise the plans and the strategies. Execute the suggested improvements and then repeat the cycle from initial so that further improvements can be done.

The GQM strategy template:

Analyze:	Process that is coding practice
To :	Improve project
With respect to:	Defect removal
From the viewpoint of :	Developer
In the following context:	Code and test for freelancing projects

SIMILARITIES AND DIFFERENCES:

Similarities:

- Both can be repeated and continually improved.
- They both have the action plan in the beginning.
- They both have clear mention of how the practices are recorded.

Differences:

- APIM focuses on agile practices whereas GQM focuses on goal and measurement approaches.
- APIM have three different phase of execution for process improvements but the GQM+ has only current phase of execution

VI. REFERENCES:

- [1] D. Jacobs, *Accelerating process improvement using agile techniques*. CRC Press, 2005.
- [2] V. Basili, A. Trendowicz, M. Kowalczyk, J. Heidrich, C. Seaman, J. Münch, and D. Rombach, *Aligning Organizations Through Measurement: The GQM+ Strategies Approach*. Springer, 2014.