BUSINESS STRATEGY: USING SHIFT LEFT PRINCIPLES TO MANAGE IT PROJECTS EFFECTIVELY

Venkatesh Jaganathan
Priyesh Cherurveettil
Anna University, Regional Centre Coimbatore, Tamilnadu, India
Thenmozhi Srinivasan
Gnanamani College of Technology, AK Samuthiram, Pachal, Tamilnadu, India
Dr. Balasubramanie Palanisamy
Kongu Engineering College, Perundurai, Tamilnadu, India

Abstract

Customer satisfaction is seen as a key differentiator in a competitive market place. As per Gartner 2012 report, only 75%-80% of IT projects are successful. Customer Satisfaction should be part of the business strategy. As a project manager, the associated project parameters should be pro-actively managed and the project outcome needs to be predicted. There is lot of focus on the end state and in minimizing defect leakage as much as possible. Focus should be to pro-actively manage and shift left in the life cycle. Identify the problem upfront in the project cycle and not wait for lessons to be learnt and take re-active steps. This paper gives the practical applicability of shift left techniques and illustrates use of predictive model in a project to predict system testing defects thus helping to reduce residual defects

Keywords: Project, customer satisfaction, shift left, project management

Introduction

A project is temporary endeavor with defined objectives. Project Management involves managing the project throughout the lifecycle. Project life cycle includes initiation phase, planning phase, executing phase, monitoring and control phase and finally close down phase. The challenge lies in understanding and meeting the project goals with the defined project constraints. Every project is unique and need to be planned well. A project has a defined start and end date. Project management is applicable across industries like production, information technology, textile to name a few. In the Information Technology industry, project management plays a crucial role. Industry experts have highlighted the importance of project management. 20%-25% of IT projects fail due to poor project management. Project Management principles need to be understood well by the project managers. Proactive management is the key for success of any project. Ability to predict project outcomes and take preventive actions will determine the success of the project. The focus on shifting left in the project life cycle is vital.

Shift Left Approach

Shift left is one of the approaches where the focus is to concentrate on the upstream activities. The intent is to reduce the defect leakage upfront such that there is less impact to downstream activities. This approach is applicable for any type of industry. In a software development project, the shift left refers to section of quality management concerned with prevention planning. Designing the shift left strategy is important. Focus should be to improve overall operational efficiency and ensure early defect detection while reducing risks and costs. The process phases for a typical software lifecycle project is define phase, design phase, develop phase, test phase and finally deployment phase. A management layer ensures these processes are followed as planned. Shift left strategy need to be inbuilt in this process so that the project manager can identify defects upfront.

In requirements phase of project life cycle, shift left approach can be implemented. In requirement phase, few process steps that can be added are requirement harvesting, requirements review by the right stakeholders and requirements testability. Requirement harvesting is not only about getting the requirements from the end user and to baseline it, but to deep dive on the requirements. Requirements should be explored and understood in detail to understand the business context and objectives clearly. Requirement review is a process that is crucial and it is important to bring the right stakeholders from the business and user community to validate the requirements. From the given requirements it is important to break it down to testable requirements. Of these requirements, identify those which can be tested individually, as a group, or those that cannot be tested. This phase is the first step and it is crucial to do it right. The focus is to detect defects early in the life cycle. Test driven development plan should be the focus in design phase. New project development models talk about test driven development. Test cases are written, executed and based on the failure of the test cases, codes are developed and tested again. Smoke testing and test environment validation need to be included before testing starts. Root cause analysis at every stage is vital to look at the corrective and preventive actions. These actions need to be implemented to avoid defect leakage and re-occurrence of defects. Project managers need to be aware of these process improvement activities and implement them in the project. Process quality consultants play a vital role in training the project managers with relevant case studies so that they can implement these improvement process steps.

Prediction Model

Prediction models should help the project manager to predict the project outcomes. Project outcomes include schedule, costs and defect parameters. A project manager has to consider the project requirement, project context and project constraints, to manage the project successfully. A project manager needs to be proactive in tracking project goals instead of being reactive. Prediction models are statistical and simulative in nature. These models should help in simulating project scenarios and help in determining outcomes. It can model the different variation factors. These models help the manager with the predicted range or the variation of its outcomes. Based on the different project scenarios, project manager needs to perform 'What-if' analysis. This analysis will help the manager to change the project parameters based on different scenarios and select the best option.

Client objectives should be translated into quantitative objectives. These objectives should then be converted to project goals. For each of the goal based on the contract, the service level agreements should be tabulated. Project manager can review these goals and go with client goals or if organizational goals are even stringent, manager can set that as the project goal. After the goals are clear, operational definition for each of this goal should be documented and agreed upon. This should also include the metric, measures, collection mechanism and frequency of collection. Process and product measures need to be considered. Influencing factors for these measures should then be identified. Based on the measures and influencing factors, predictive models can be built to predict project outcomes. Prediction models can be used to predict interim and final outcomes. Influencing factors can then be modified to analyze the impact and determine actions to be taken.

Process quality managers play a key role to ensure projects adhere to quality standards. In an organization, the quality group should focus on developing standard prediction models for basic project parameters. A prediction model for project schedule variance, project effort variance and project defect density would be a good start. Process teams can build these models using rich organization project data. These models can then be used as reference for the project managers. When a project manager starts a project, organizational prediction models can be inherited and then customized to the project context. The influencing factors and project constraints might vary from project to project.

Defect Prediction Model

System testing is an important phase in project development life cycle. At this phase, systems are tested extensively. This also includes the integration of systems. For a project manager, system testing defect density is an important parameter to track. The number of defects identified during system testing determines the quality of the development. Project subject matter experts identify the different parameters that influence the system testing defect density through brainstorming session. Based on the influencing factors, the key ones that impact the system testing defect density is chosen. Then the operation definition, metrics and measures are arrived at. Project data was collected for these parameters. Linear regression was performed against the data to find out the key variables that influences the system testing defect density. After many trial and error methods the below two variables were established as the x factors.

1. Y – System Testing defect density – No of defects identified in system testing of the project against effort spent during system testing phase of the project
2. X1 - Technology experience – Average relevant technology experience of the team, in person months
3. X2 – Requirement Clarification Index – Number of requirements clarification resolved against number of requirement clarification raised during requirement phase of the project

The project data collated for the x and y factors are as shown in the Table 1. Data points from 25 projects in an organization were collected and considered for analysis. Projects factored in were similar in nature. The null hypothesis considered is that X1 and X2 have no influence over Y. Technology experience and requirement clarification index does not impact system testing defect density.

Y	X1	X2
System Testing Defect Density	**Technology Experience(in months)**	**Requirement Clarification Index**
0.085	35	80.000
0.092	31	75.000
0.092	33	80.000
0.123	35	70.000
0.114	25	50.000
0.113	21	60.000
0.124	22	65.000
0.132	33	70.000
0.115	35	75.000
0.141	18	65.000
0.095	34	80.000
0.070	40	90.000
0.071	42	90.000
0.078	48	90.000
0.083	39	90.000
0.084	41	90.000
0.131	34	75.000
0.133	28	80.000

0.131	25	55.000
0.342	4	50.000
0.620	3	30.000
0.390	5	20.000
0.089	43	95.000
0.095	41	95.000
0.520	6	40.000

Table 1 – Project data values

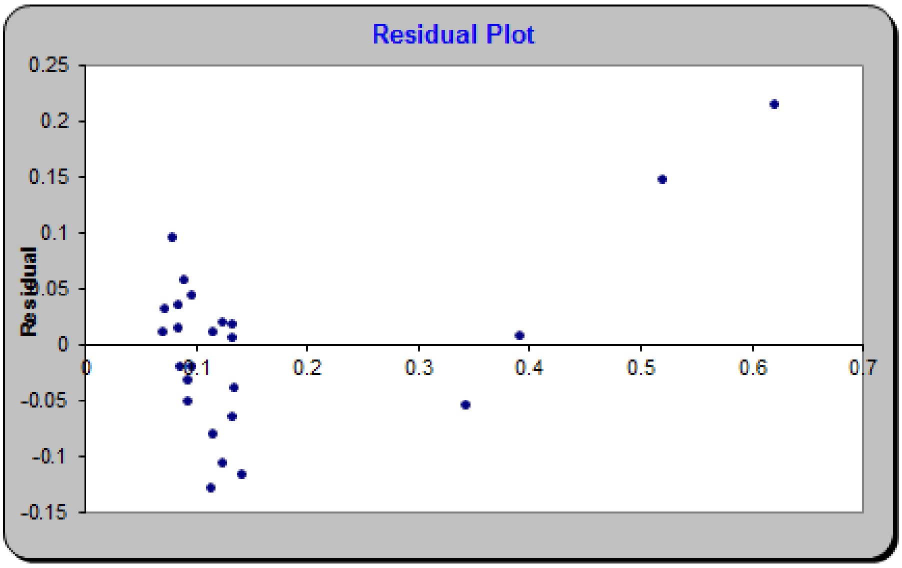

Figure 1.1 – Residual Plot

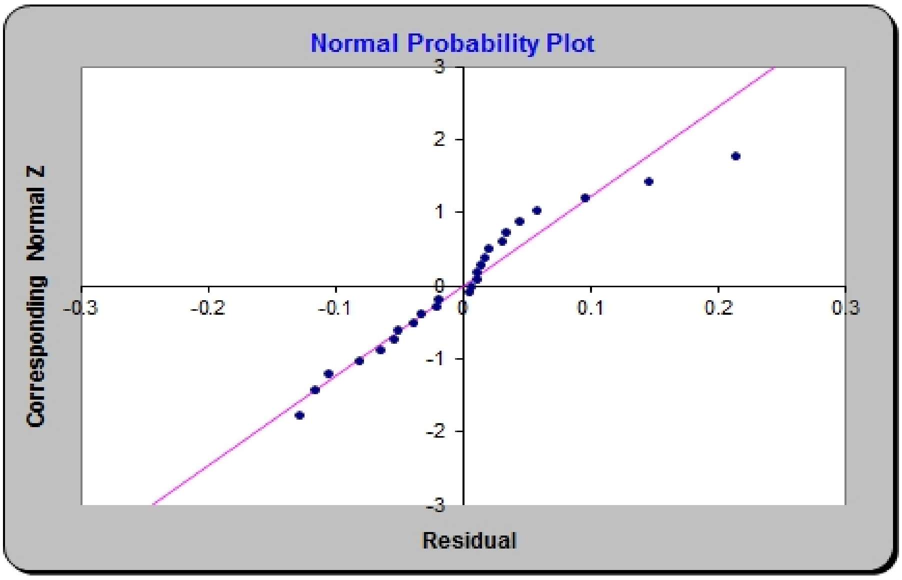

Figure 1.2 – Normal Probability Plot

Mirror pattern is not found in Figure 1.1, Residual Plot and hence no heteroscedasticity is found. The normal probability plot in Figure 1.2, is approximately linear. This would indicate that the normality assumption for the errors has not been violated.

Looking at the p value, since it is 0.001 which is < 0.05, null hypothesis is not valid, which means the variables selected have an impact to Y. Technology Experience and Requirement clarification index have an impact to system testing defect density.

Intercept	Technology Experience(in months)	Requirement Clarification Index
0.47301	-0.0073	-0.0014

Table 2 – Regression Equation

As shown in Table 2, technology experience has a negative influence on overall defect density. As the team's technology experience increases, there are less project defects and hence the system testing defect density is reduced. The influence of requirement clarification index is negative. This means that when the values of requirement clarification index are high the system defect density will be low and vice versa. The more requirements clarifications are resolved, lesser the probability of defect injection.

The data proves that the system testing defect density is influenced by requirements sub process and technology experience. Project quality group in the organization shares the base line data for these variables. For each sub process, based on the type of project, the organization values can be considered. The project manager can then use these reference values to determine the upper specification limit and lower specification limit for the project. These values will be available for each of the sub process parameter. The project manager can determine which sub process he would like to control and select the threshold values based on that. For example, in requirement management process, requirement clarification sub process, organization value might recommend 85% as the baseline value. Project manager might decide to refer this and make it more stringent and use 90% as the baseline value. Based on the current project context, the other sub process parameters and technology experience values can be considered for prediction.

Based on the selected threshold values, what-if analysis is performed. Going by the process sub parameters and their values, system testing defect density is predicted. The predicted outcome is then compared with the thresholds. Based on the gap, sub process parameters are further tweaked to understand the variation. It is important to note that while changing the parameters, project manager should understand the practical implication in the project. It is not only about the mathematical model, but about how it can be put in practice. For example, if, based on the prediction model, the team average technology experience is expected to be around 60 months. Then the project manager has to look at the project constraints. To on-board a team of experienced developers, the cost impact and the impact to on-boarding schedule need to be considered. So though the prediction model considers the key influencing factors to predict the system defect density, the influencing factor values might have an impact to project schedule and project cost, which need to be analyzed.

Project manager might decide to adjust the predicted outcome based on the project constraints. At every step, the manager should document the assumptions, risks and mitigation plans. Detailed defect prevention plan should be in place. At every step, the defect, its type, cause, preventive action, owner and target date should be documented. A simple example of a defect at requirement phase would be that the business analyst would have assumed that the requirement has no impact to a particular business segment. The requirement clarification raised, was probably ignored and team went ahead with the design phase. This is a typical case of defect leakage. It is important to understand why business analyst didn't think through the current production scenario. Why the requirement clarification got missed and analysis phase didn't consider the pending requirement clarifications. Thus at every step the quality gates are important. Quality gates need to be defined at every phase in the project life cycle. The quality gates need to be reviewed and approved by the project manager. Process quality manager should also review these quality gates and suggest improvements as need be. Different root-cause analysis techniques like 5-why can be used to pin down the root cause. After identifying the root cause the next steps in terms of corrective and preventive actions should also be thought through. It is recommended to have process quality experts review these plans so that they can bring in their experience and highlight any improvements in these plans.

For a project manager to manage the project parameters, access to the right tools and techniques is important. Project managers should learn to manage project pro-actively than being reactive. Shift left techniques should be adopted. Defect injection at the initial stages of project life cycle should be controlled. The focus of shift left is to arrest defects up in the life cycle so that defect leakage is minimized or eliminated in the downstream process. Capability Maturity Model recommends usage of quantitative models. These models help in building prediction models. Though Capability Maturity Model highlights quantitative models at length, the practical implementation in industry is minimal. Focus on quality should be a culture in the organization. It is about doing it right at the first time. Regular trainings and awareness of process quality and product quality would help in building the culture in the organization.

Building quality gates in project life cycle

A typical software project development life cycle will go through requirements phase, design phase, development phase, unit testing phase, system testing, user acceptance testing and finally implementation phase. Quality is an important project parameter at every phase in the life cycle. In the requirement phase, project manager need to ensure that the scope of the requirements are thoroughly documented and signed off. Requirement traceability matrix need to be created. Every requirement should be traceable as the project moves from one phase to the next phase. Adherence to change management process throughout the process is vital. Project manager should ensure that change control board is formed during the project initiation phase itself. Requirement clarifications need to be tracked and closed. Requirement reviews should be formally signed off. During the design phase, design prototypes need to be prepared and validated with the user groups.

Design standards, templates and tools need to be decided and used in the project to ensure compliance. Traceability of design modules to requirements is important. Build sequence need to be factored as part of design process. Project manager need to ensure that coding standards are followed. Configuration management needs to be implemented. Organization configuration tool or client configuration tool can be utilized. Detailed configuration management process should be laid down. Project manager need to ensure all the relevant stakeholders participate in design and code review process. It is recommended to use code review tools. Defect prevention plans need to be implemented.

Detailed test strategy and test plan should be laid down for system testing. Defect management process should be actively followed. Test case coverage is vital. Defect Triage meetings is recommended for faster turnaround of defects. Entry and exit criteria at every testing phase should be called out and followed. Project manager need to ensure that all business scenarios are covered as part of test coverage. Clearly defined standards and procedures should be implemented. Periodic quality reviews and audits are recommended. Project manager need to conduct retrospective meeting and implement the preventive actions. A quality mindset with the project team and all relevant stakeholders is important.

Conclusion

In any business, customer defines the business. Customer Satisfaction is vital for survival. In an IT organization, the customer satisfaction index is pivotal. Customers can make or break business. Customer can in turn act as brand ambassadors and recommend new business for the organization. The main parameters that influence customer satisfaction are faster time to market, quality and cost. Based on the customer context, the priority of these parameters will change. Few additional parameters can get added. But overall, cost, quality and schedule sets the base. These three parameters are again related to each other. As a project manager it is important to understand the customer expectations and ensure that these critical parameters are managed well. Prediction models help in managing these parameters effectively. A manager can predict desired project outcome, perform scenario based analysis and take right decisions to meet project goals. The practical case study demonstrated how project manager can predict system testing defect density considering technology experience and requirement clarification index. The usage of prediction model, what-if scenario analysis, corrective and preventive actions, assumptions and risks, all these process steps help the manager to meet the project outcomes. Project managers need to be trained on prediction models and should be comfortable to use these models effectively in projects. Shift left techniques determine the project success. Prediction models is not a one time activity, it should be continuously used during the project life cycle.

References:

CMMI® Product Development Team: CMMI for Development, Version 1.2, CMMI-DEV, V1.2. Carnegie Mellon University, Pittsburgh, PA., 2006.

Stephen H. Kan: Metrics and Models in Software Quality Engineering 2nd ed., 2002.

Richard D. Stutzke: Estimating Software-Intensive Systems: Projects, Products, and Processes, Addison-Wesley, 2005.

Robert W. Stoddard: CMMI Process Performance Models and Reliability, Carnegie Mellon University, IEEE Reliability Society Annual Technology Report, 2007.

Umesh Kumar Mishra, K. Harihara sudhan, Shalu Gupta: Establishing Process Performance Baselines and Models for Statistical Control of Software Projects, Proceedings of ASCNT-2011, CDAC, Noida, India.

Tim Kasse: Practical Insight into CMMI, 2nd ed., Artech House, 2008.

Bill Self, Greg Roche and Nigel Hill: Customer Satisfaction Measurement for ISO 9000: 2000, 2002.

Michael Johnson and Anders Gustafsson: Improving Customer Satisfaction, Loyalty and Profit, 2000.

Margaret K. Kulpa and Kent A. Johnson: Interpreting the CMMI: A Process Improvement Approach, 2nd ed., 2008.

Tim Kasse: Practical Insight into CMMI, 2nd ed., 2008.

Don Peppers and Martha Rogers: Managing Customer Relationships: A Strategic Framework, John Wiley & Sons, 2004.

Forrest W. Breyfogle III: Implementing Six Sigma: Smarter Solutions Using Statistical Methods, John Wiley & Sons, 2nd ed., 2003.

Dr. J. Venkatesh, Priyesh Cherurveettil, Thenmozhi. S, Balasubramanie. P: Predicting test effectiveness using performance models in Life Science IT projects, Life Science Journal 2012;9(4):96-100 (ISSN:1097-8135).

Margaret K . Kulpa and Kent A . Johnson: Interpreting the CMMI (R), A Process Improvement Approach, Auerbach Publications 2003.

Dr. J. Venkatesh, Priyesh Cherurveettil: Using Application Performance models to achieve improved customer satisfaction, IRACST- International Journal of Research in Management & Technology (IJRMT), ISSN: 2249-9563, Vol. 2 No.2, April 2012.