

Improvements In Agile Manifesto For Small Software Development Industries

Ratnmala Raval, Haresh Rathod

Computer Department, School of Engineering, RK University, Rajkot, Gujarat, India.

Computer Department, School of Engineering, RK University, Rajkot, Gujarat, India

Abstract

Now a day's business environment is much more dynamic, and organizations are constantly changing their software requirements to adjust to the new environment. They also insist on fast delivery of software products as well as for accepting changing requirements. Agile is swift active and quick to respond, and this is what agile software development methodology refers to. Agile methods are much attracted to small projects, but no agile methods workings glowing as a standalone system so some adoption or customization is needed. The paper describes an agile adoption survey for small industries in Gujarat region. The main objective is to find reasons for less agile adoption. The circulated survey results are analyzed thoroughly and discussed to prove the arguments made during the research time and the recommendations are provided to improve appropriate enhance required changes for the agile principles. Finally, these improvements are implemented by various industries and The analysis of the results showed that the improvements used in the aforementioned cases were effective.

Keywords—Small project, Agile Adoption survey questions, challenges of agile adoption, Improvement theory, process flow improvement

I. Introduction

Agile methods are a unified set of theory, principles and practices to address what most consider the bane of software development: unfortunate and changing requirements, short development cycles, stretched working hours and rapidly increasing system complexity. Assimilating the best ideas of design and development results in a full agile development process that can reduce costs and improve quality through the focused application of agile key principles. These principles have been applied successfully to many different real-time and embedded markets, such as telecommunications, medical, industrial automation, defense and aerospace. This paper first defines challenges in agile adoptions in small software development industries then presents research questions about agile adoption, which is live survey and description of improvements applied in different cases and the results of these improvements.

II. Background

Low cost and time-to-market is the most important drivers of software productivity improvements Small enterprises (with around 50 employees) and very small enterprises (fewer than 25 employees) represent up to 85% of all software companies in countries that have an advanced IT industry [2]. Generally, small companies are very responsive and flexible, and directly/indirectly

involved in many small projects. Managing challenges in needs tight planning and response. Level 1 company does not have defined agile software development process. The main problems at this level relate to overtimes, schedule slips, communication, software quality and development cost. These companies operate in their own unique way and depend on particular people rather than whole team[5]. On the other side in managing challenges larger projects consist more of analyzing the complexity for business areas, technology, risks, and managing a large number of stakeholders. Hence, small projects have a need for a different approach Than larger ones. Here in this section we describe an approach for small software development projects and challenges facing by small software development industries. Here some factors are classified with the help of that we can classify projects as small.

- Project duration is 4-6 months.
- Team members are an average 5-7.
- Small no of skill areas are involved.
- Readily achievable solution with single objective.
- Scope and definitions are limited.
- No external project source is required as automated solution [2].
- Projects are commonly with limited budgets.

A. Agile adoption survey in small software development industries.

published globally on the internet especially in the Gujarat region.

To find out agile adoption in small industries, we prepared following questionnaire and

Table I: Agile adoption questionnaires

1	Since how long your organization works for software development?
2	In which sector your organization primarily in?
3	How many projects your organizations developed yet?
4	What is the approx team size of your project?
5	Which kind of development practice your organizations follow?
6	Have you prepared formal paper based documents for your project?
7	Which kind of testing and quality practice followed by your organization?
8	Does your organization use any kind of testing tool?
9	Do you follow code review?
10	Have you heard about agile?
11	Do you conduct respective meetings while developing software
12	Do you change your developing strategy while changes platform / domain?
13	Do you think estimation time will affect cost development?
14	Do you think change adoption is most challenging in software development?
15	Do you think hires expert developer will affect the cost of development?
16	Do you think developer's perspective will affect the quality of software.
17	Do you believe in self organized team structure will affect the quality of output.

B. Challenges in agile adoption.

- Agile is time consuming for small projects.
- Communication and team cooperation.
- Customer communication, involvement.
- Change resistance.
- Management support and understanding.
- Lack of planning
- Inexperienced project teams.
- The Project manager is responsible for multiple functions.
- Agile requires expertise developers which in not suitable in a small budget.
- Time pressure. Communication within the team and cooperation. Process vs. uncertainty.
- Agile puts people ahead of the process.
- Agile methods depend on strong development they must be amicable, talented, skilled and able to communicate well this is managing risk in agile development.
- As a tester level in agile method such as xp tests are changed before code is modified by the development and the role of tester is significantly reduced.
- Agile planning is an informal process.
- With agile process information is communicated informally & is simply kept as a part of the collective knowledge of the organization. While

reducing amount of documentation can increase productivity, but it does come risk and cost.

III. Improvement theory

A software development method is said to be an agile software development method when it is people focused, communication-oriented, flexible (ready to adapt to expected or unexpected change at any time), speedy (encourages rapid and iterative development of the product in small releases), lean (focuses on shortening timeframe and cost and an improved quality), responsive (reacts appropriately to expected and unexpected changes). In addition, such a method should provide an environment for learning (focuses on improvement during and after product development). .Xp and Rup are most popular agile model widely used in various organizational and software industries, Both share common features; they are iterative, customer oriented and role based . RUP comprises 100 artifacts while XP focuses only on code. Similarly, RUP has 40 roles while XP has only 5 roles. These similarities and differences motivated us to analysed and integrate the best features of both models [3].

A. Agile improvement theory for small software development industries.

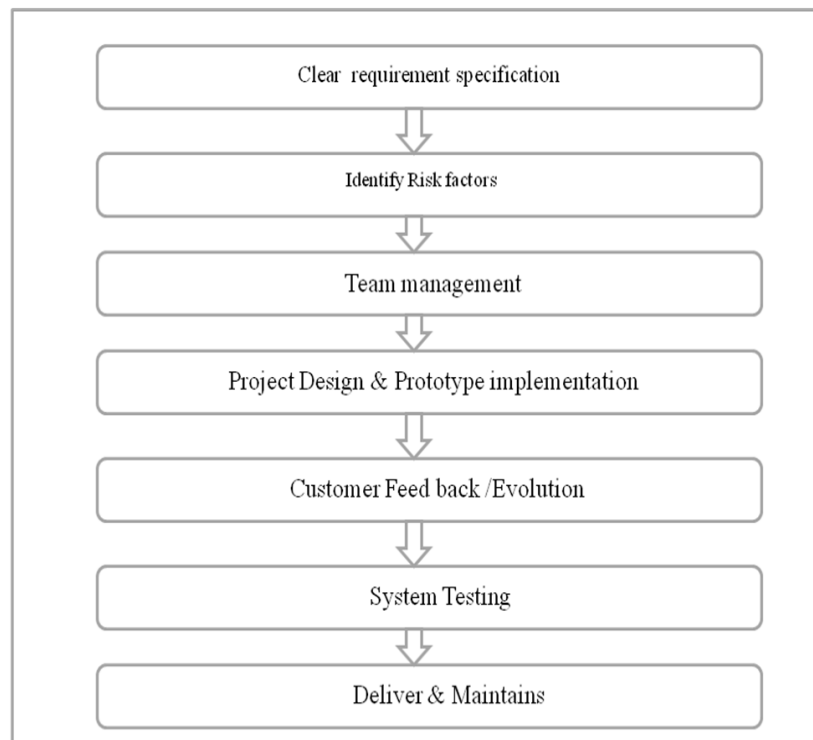


Fig 1. Steps of improvement theory.

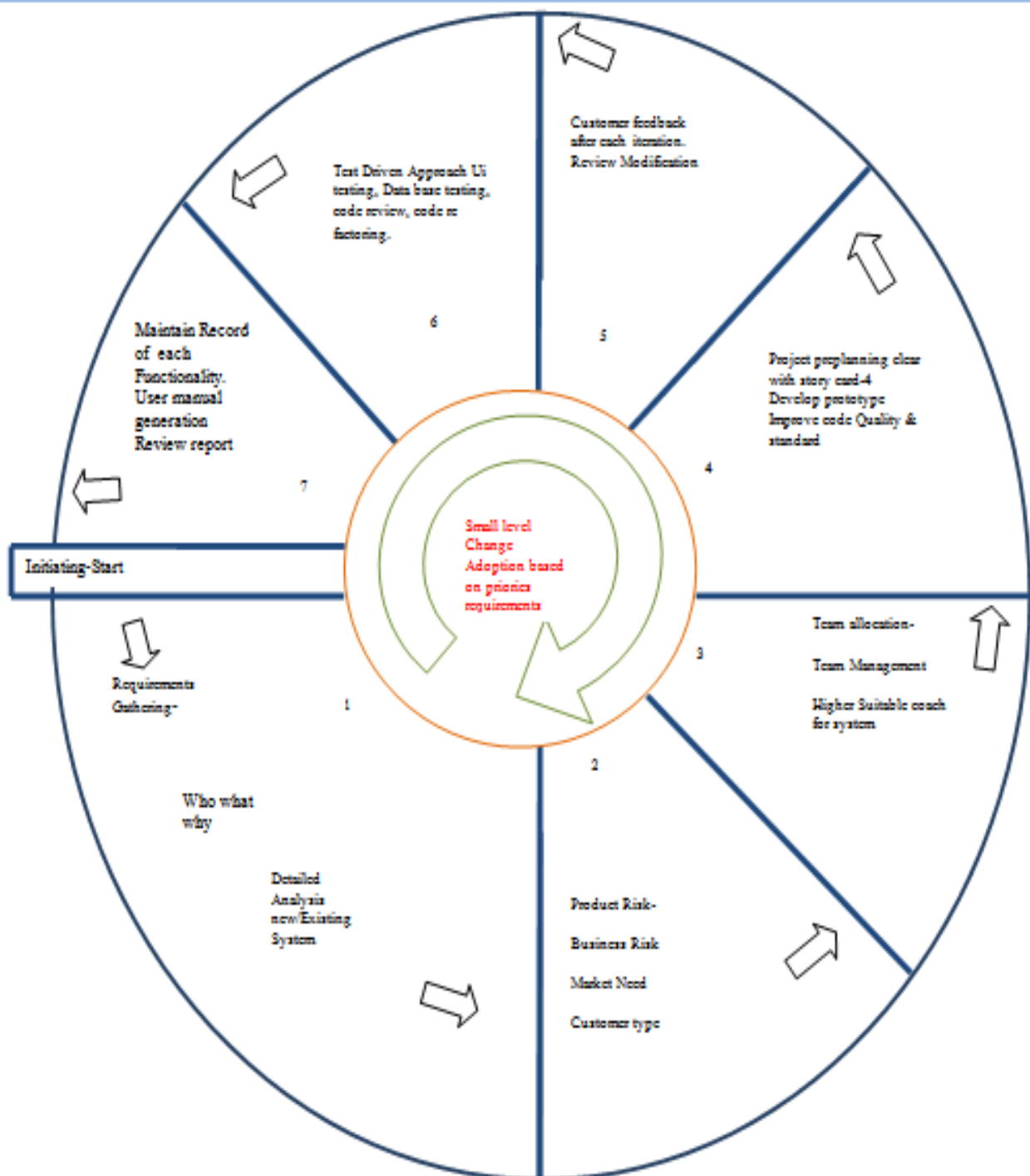


Fig.1 Process flow

Step-1 Clear requirement specification: First, the project team identifies the five requirement specification factors affecting the selection of either agile or plan-driven methods: 1) background on the reason for requirement. 2) Description of resulting action form input. 3) Description of input/user action 3) Owner of requirement 4) Target user. Elements of requirement specifications are Background on the

reason for the requirements, Description of resulting action from input, description of input, owner of requirements, the targeted user[12].

Step -2 Identify risk factors: The Project team should identify following risk factors 1) size: the number of people on the team, 2) criticality: products safety criticality, 3) dynamism: degree of

requirements and change in technology, 4) personnel: the skill and experience of the team[3],5)identify business risk 6) Responds to market needs[12].

leadership team.[11], 3) Hire some expert and some basic level programmer for implementing pair programming.

Step-3 team management: 1) Development team meets at the start of each design iteration to plan work for that iteration.2) Develop transformational

Step 4 Project Design & Prototype implementation / Process frame work.

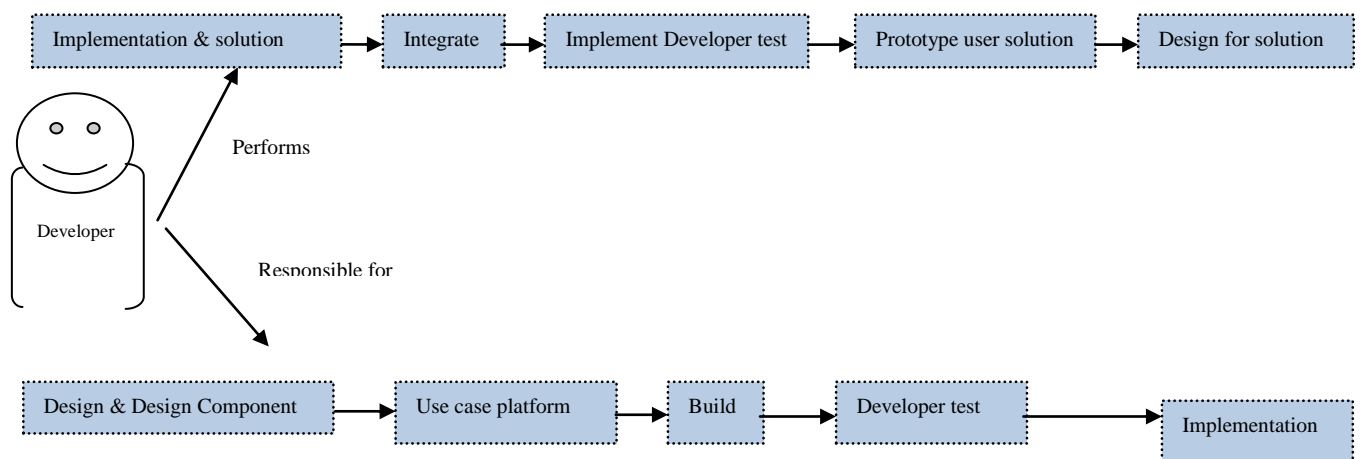


Fig .2 Process design for project implementation

In the phase of process development, visualize workflow manages workflow and make process policies explicit and collaborative. Secondly capability to improve code quality which includes 1) Design verification /testing occurs with each design iteration.2) Iteration review include continues improvement discussion . 3) Iteration review focuses how customer requirements have been met? It is to be advisable that developers must have broad knowledge on all aspects of software development but should also specialize and hone their skills in certain areas.[18]

is required that will not affect the time and cost of the project.

Step- 5 Customer feedback and evaluation: Agile focus on higher customer priorities and insist face to face communication, but in small software development industry level of customer technicality is less. Relatively projects are also in small size, 2-3 times customer meeting is enough or using some online software project management tools to daily communicate with customer but it relate the type of customer.

Step-7 Deliver /Maintains: Agile says the documentation is wasting time. But the researcher who got the idea of documentation have some important glow. To maintain documentation will help while the project moves from one phase to another phase in the six months project some time happens that team member are not remembering the small decision taken at analysis time .So to be up-to-date with documentation is always achieved good quality in software development.

Step -6 System Testing: Use testing tool is increase budgets of small industries. Following test approach should be followed like test driven approach, a testing, database testing, module testing, implement product as much as flexible so if any change adoption

IV. Implementation and Results

This improvement theory is applied in various industries, domain and platform, as steps present in Fig 2 process flow, calculate risk factors time for calculation and cost for development. The final risk calculation follows the original equation: Risk = Probability x Impact Value[4].

Table 2 and Fig 2 show that risk identification is up to perfect level, which was not possible with traditional approaches and a chance to generate risk is very low with early identification. Fig 3-4 shows time for development and cost to development. It shows a vast margin between the traditional approach and improved approach.

A. Risk calculation

Table-II. Risk Calculation

Project	Impact level	Impact value	Risk
1	1	1	0.95
2	1	1	0.95
3	1	1	0.95
4	1	1	0.95
5	2	3	2.85

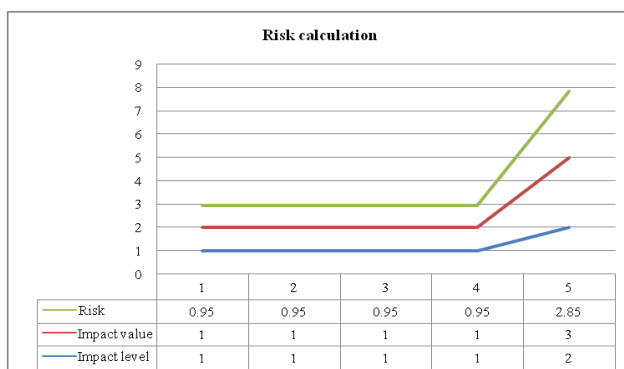


Fig.3 Risk Calculation

B. Cost and time for development analysis

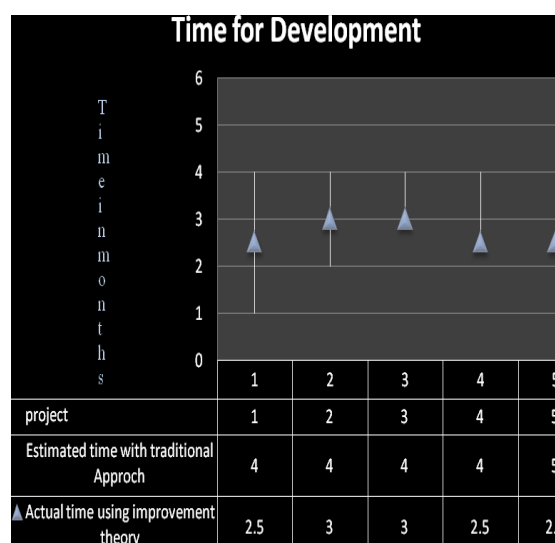
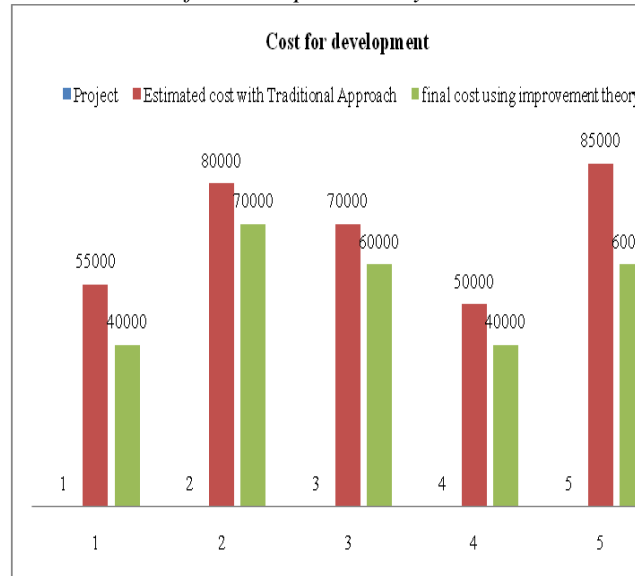


Fig. 4 Cost and time for development analysis

Table-3. Implementation Analysis.

Parameters	Case-1	Case-2	Case-3	Case-4	Case-5
Project type	Task Management Application	Business Application	Process management Application	Business Application	B-to B portal
Duration	2.5 months	3 months	3 months	2.5 months	2.5 months
Platform	Xcode - java	. Net	PHP	PHP	PHP
Domain	Mobile Application	Online race	Business	Business	Business
Success of pair programming(one Expert+ one Basic)	High	High	Medium	Medium	High
Customer feedback	2-3 days	2-3 days	2-3 days	Daily	Module wise
Quality of Product Improvements	Low	Low	Low	Low	Medium
Level of customer satisfaction	Very good	Very good	Good	Good	Very good
Success ratio compared to traditional approaches	91%	95%	75%	85%	91%

V. Conclusion

The objective of present theory is to develop high quality small to medium application within clear risk identification ,time constraint, low cost and higher customer satisfaction. Table2 and Fig 3 shows risk calculation , Fig-4 shows time and cost varies between the traditional approach and improved approach and table -3 shows another parameter measurement, though all these results provide evidence that present theory improve productivity ,performance, time to development and other attribute

For further research on the agile model in software development, researcher should focus on the following aspects. Apply the presented process flow in various small software industries and find out efficacy, outcomes and cost calculation can be improved with various cocomo methods. . In this research our industries is mainly developing web based applications for further research, researcher should focus on another window based, ERP ,embedded system development. .

VI. Acknowledgment

We are gratified to the R.K University Department Of Computer Engineering who have contributed towards the development of this research. We are thankful all industries who support and guide us throughout this research.

References

- [1] Claudia Melo*, Daniela S. Cruzes†, Fabio Kon* and Reidar Conradi† Department of Computer Science, University of Sˆao Paulo, Brazil Department of Computer and Information Science, NTNU, Trondheim, Norway. Agile Team Perceptions of Productivity Factors- 2011 Agile Conference.
- [2] Seiyong Lee* and Hwan-Seung Yong** J Inf Process Syst, Vol.9, No.1, March 2013 <http://dx.doi.org/10.3745/JIPS.2013.9.1.069> Agile Software Development Framework in a Small Project Environment
- [3] Ghulam Rasool1*, Shabib Aftab1, Shafiq Hussain2, Detlef Streitferdt3 eXRUP: A Hybrid Software Development Model for Small to Medium Scale Projects *Journal of Software Engineering and Applications*, 2013, 6, 446-457
- [4] Robert W. Ferguson A Project Risk Metric The Journal of Defense Software Engineering April 2004
- [5] Chetankumar Patel and Muthu Ramachandran Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices Int.J. of Software Engineering, IJSE Vol.2 No.1 January 2009
- [6] Real-time agility- IBM Software Technical White Paper December 2011
- [7] Michael Coram and Shawn Bohner Department of Computer Scienc Virginia Polytechnical Institute and State University- Proceedings of the 12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'05) 0-7695-2308-0/05 \$20.00 © 2005 IEEE
- [8] Shvetha Soundararajan, James D. Arthur and Osman Balci A Methodology for Assessing Agile Software Development Methods- 2012 Agile Conference- 978-0-7695-4804-3/12
- [9] M.A.S.C.Manamendra, K.N.Manathunga, K.H.D. Perera and N.Kodagoda, Improvements for agile manifesto and make agile applicable for undergraduate research projects The 8th International Conference on Computer Science & Education (ICCSE 2013) April 26-28, 2013. Colombo, Sri Lanka 978-1-4673-4463-0/13/
- [10] Theodore D. Hellmann, Abhishek Sharma, Jennifer Ferreira, Frank Maurer- Agile Testing: Past, Present, and Future Charting a Systematic Map of Testing in Agile Software Development2012 Agile Conference 978-0-7695-4804-3/12
- [11] Claudia Melo*, Daniela S. Cruzes†, Fabio Kon* and Reidar Conradi†- Agile Team Perceptions of Productivity Factors 2011 Agile Conference 978-0-7695-4370-3/11
- [12] Michelle Boucher implementing an agile methodology to improve time and quality- september 2012 ,Aberdeen group A hertic hance company
- [13] Gayane Azizyan Miganoush Katrin Magarian- Ericsson AB Stockholm, Sweden, Mira Kajko-Mattson- Royal Institute of Technology Stockholm, Sweden- Survey of Agile Tool Usage and Needs- 2011 Agile Conference 978-0-7695-4370-3/11
- [14] Andrew Griffiths- Agile Improvement Method Lamri Ltd 2011
- [15] Naresh Kumar Nagwani, Pradeep Singh Department of Computer Sc. & Engg. National Institute of Technology, Raipur- An Agile Methodology Based Model for Change-Oriented Software Engineering International Journal of Recent Trends in Engineering, Vol 1, No. 1, May 2009.

- [16] Hongyu Pei Breivold (Industrial Software Systems-ABB Corporate Research) Daniel Sundmark, Peter Wallin and StigLarsson (Mälardalen University) Västerås, Sweden-What Does Research Say About Agile and Architecture?2010 Fifth International Conference on Software Engineering Advance
- [17] Ashwini Mujumdar, Gayatri Masiwal, P. M. Chawan / International Journal of Engineering Research and Applications Analysis of various Software Process Models International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com Vol. 2, Issue 3, May-Jun 2012, pp.2015-2021
- [18] Kieran conboy and Sharon coyle,"People over process: Key Challenges in agile development" IEEE Software.
- [19] D. Duka Ericsson Nikola Tesla, Split, Croatia, Agile Experiences in Software Development Ericsson Nikola Tesla, Split, Croatia MIPRO 2012, May 21-25,2012,
- [20] Scott w ambler ,metthew holitza.Compliments of ibm limited edition Agile for dummies .
- [21] Roger S. Pressman, Ph.D.R.S. Pressman & Associates, Emerging Trends in Software Engineering Inc. Boca Raton, Florida USA January, 2009