ISSN 2348 – 8034
Impact Factor- 3.155

GLOBAL JOURNAL OF ENGINEERING SCIENCE AND RESEARCHES

A COMPARATIVE STUDY OF QUALITY MEASUREMENT IN AGILE AND WATERFALL MODEL

Mukhiar Bano*[1], Tahira Mahboob[2], Arshia Zaffar[3] and Rida Mumtaz[4]
[1,2]Assistant Prof. Department of Software Engineering, Fatima Jinnah Women University, Pakistan
[3,4]Fatima Jinnah Women University, Pakistan

ABSTRACT

With having numerous factors impacting the system, software development is regarded as one of the highest complex field. There are no apparent laws or certainties to build software. Due to which software is almost always sub-optimized. There are many SDLC models present for building softwares, two of these will be discussed in our paper, named Waterfall Model and Agile. The Waterfall model consists of five phases which need to be completed sequentially in order to develop a software solution. Agile caters the shortcomings of the Waterfall process. A sudden change is not mandatory for transitioning to Agile Development; instead it can be accomplished by embedding Agile Methods into an existing process. There are many factors that contribute towards the downfall of a software system, most significant one is not choosing the right methodology for developing your software. For successfully developing software, the right software development process should be chosen to get a high quality end result. A comparative study of two of the most popular development processes has been presented in this paper which provides an insight towards the processes and makes it easy for the developers to choose the right kind of methodology for their system.

Keywords: *Waterfall, Agile, SDLC, Heavyweight, Lightweight, Software Process, Quality.*

I. INTRODUCTION

Several models have been evolved over the past few years that can be used to develop a software system. Each approach takes a variety of different tasks for completing the process. Software development undergoes a variety of different steps to get to its final shape. Project Planning, Feasibility Study, etc. are some of the phases which need to be followed for developing software systems. A successful software system is strongly dependent on the delivery or high quality software. The term 'high quality' here refers to a number of quality attributes which need to be catered by the software being developed.

The most important factor that distinguishes Agile from Waterfall is the involvement of testing throughout the development lifecycle. The two testing techniques i.e. white box and black box testing are made known to the developer. Developers write the unit tests and make use of the automated tools for testing purposes, a separate testing phase for the testers is not present. The basic and most important goal of Agile is the provisioning of good quality software. The use of automated unit testing makes it easy to validate the individual test features which eases the integration process.

In Waterfall Model each phase is independently performed in a linear manner. It involves significant rework due to the presence of premature freezing of requirements. The quality of the software is tested from the user's point of view during the final life cycle phases which include operation and maintenance. The most distinguishing feature of this model is that it does not cope well with change and contains a lot of rework due to the presence of late testing. While in Agile this problem is catered.

In this paper, two of the most known software development processes i.e. Waterfall and Agile have been examined to see which one is a better approach for building software systems. The findings of this

II. WORKING OF AGILE AND WATERFALL IN DIFFERENT SCENARIOS

2.1 Research on the Rule of Evolution of Software Development Process Model (Y. Gao, 2010).

Efficient selection of Software development process plays an important role for ensuring quality in a system. Many development processed have been introduced up till now but its an evolutionary process. The research shows that, the evolution of software development process is formed on Four basic models which includes waterfall model, rapid prototyping model, incremental model and component-based model. There are four directions of evolution in the future, which will weaken complexity, enhance reliability and flexibility. The research results contain the driving

ISSN 2348 – 8034
Impact Factor- 3.155

force of the evolution of model, the basic model that can be used for continuing evolution, the prevailing trends of evolution of software development process model, and the future evolution
direction of model. The findings presented in this paper can help people properly understand the existing software development process models and build new models. [1]

2.2 A Comprehensive Study of Commonly Practiced Heavy and Light Weight Software Methodologies (A. Irshad et. al, 2011)

A. Irshadet. al categories the methodologies of software into heavyweight and lightweight. Waterfall lies under heavyweight commonly called as traditional methodology. Agile is covered under lightweight. According to waterfall model customers are synergic, requirements are highly stable, architecture is designed for negotiable requirements and team size is kept large. Its main aim is to preserve high affirmation and it manufactures the product in big bang form. Whereas, on the other hand agile customers are collimate and highly interactive with developer team, their requirements are emanating. Agile follows the newly specified architecture and their team is small with aim of winning out customer satisfaction. They focus over implementation and delivering simultaneously. This paper ends in conclusion that none of the models can be made a silver bullet selection of any model depends on the organization, product being made and the customer's demand. [2]

2.3 Quality Assurance and Its Standards: Importance in Various SDLC Models (A. Gupta, 2014)

In the current era customers are more interested in user-friendly software, which makes
them more satisfied. That is why Software Quality standards should be considered. Ensuring that the software has less number of defects and it achieves the standards is called, Software management. These standards include maintainability, reliability and portability. These standards represent an identification of best practice. Getting reviews from the user and accessing software quality is the best way to check whether software is deliverable or not. A study of how quality of software varies in different SDLC model, has been made in this paper. To ensure consistent quality for both process and product Quality standards can be used. Prior to the delivery of the product, it is necessary to review quality assurance plan template and non-compliance report document. [3]

2.4 Software Process Models and Analysis on Failure of Software Development Projects (R. Kaur et. al, 2011)

The chosen paper highlights some of the root causes of project failure. The major reason can be the chosen methodology, customer, developer and other concerning bodies. The requirement gathering is the most initial step which requires customer, customer lacking communicating skill might not be able to convey their need properly, for this expertise are required who interpret the customer's requirement intelligently. User involvement should be appreciated as it helps in connecting the team members with their work. Developer and the system should be flexible enough to adapt changes because in today rapid changing world requirement can't be made frozen. Next is the time span which is important to follow, it shouldn't be like much long that some other organization launches the product earlier than you. It is recommended to deliver product within terse time extent. [4]

2.5 A Simulation Model for the Waterfall Software Development Life Cycle (Y. Bassil, 2012)

One of the many SDLC models is Waterfall model; it consists of five phases which should be completed sequentially for developing a software solution. But there are some limitations associated with this approach. A simulation model for Waterfall process has been presented in this paper. In this model, all the employees and resources are kept busy all the time to reduce the idle time. The simulation tool used for this purpose is called Simphony.NET. This simulation guarantees that the interval-time between each project arrival is equal to the interval-time between each project production. Conducting some experiments proved that the proposed model is accurate, as it accurately calculated the number of optimal resources required to accomplish a particular software solution based on their utilization metric. [5]

2.6 The Waterfall Model in Large Scale Development (K. Petersen et. al)

Waterfall development is still a widely used way of working in software development companies. But many issues are associated with this model. These problems include difficulty in coping up with change and that the defects are detected too late in the software development process. However, many of the problems identified in the literature are based on beliefs and experiences, and notion empirical evidence. To cater this research gap, a comparative study has been made in which the problems in literature are compared with the results of a case study at Ericsson A Bin

Sweden, with the sole objective of investigating issues in the waterfall model. Validation or contradictory beliefs of the problems in waterfall development have been addressed in this paper. One of the findings of the paper includes that for large-scale development, the waterfall model is not suitable. [6]

2.7 A Framework For Software Quality Assurance Using Agile Methodology (M. Sagheer et.al, 2015)

Iterative and incremental approach is used by Agile methodology, it is widely used in the industry projects as this methodology can cater the changing requirements. Short iterations are used for efficient product delivery. Ensuring quality of a system is a trivial task, and is difficult to achieve, which raises a question that: Does the use of agile methodology assures quality of software to be developed? This research paper presents a solution and sums up that how the use of agile software development assures quality by using different factors. A brief overview of the parameters, which assure quality in agile has been included in this paper. It discusses the effect of different factors and parameters on software quality. Communication with the customer is an essential part in Agile, so that the new features that are added in the system satisfy the customer needs and requirements. [7]

2.8 Adopting Agile Software Development: Issues and Challenges (H. Hajjdiab et.al, 2011)

Adopting agile software development leads to production of better quality software by significantly reducing costs. But adopting this approach has some challenges e.g. for successful outcome organizational changes are mandatory. A case study for adoption of agile software development for a government entity in U.A.E has been presented in this paper and its outcomes are properly analyzed and compared to the other published case studies in this domain. One case study may not be efficient enough reach conclusions on the Scrum agile method in the United Arab Emirate (U.A.E), nevertheless many lessons can be learned from these experiences. [8]

2.9 Agile Software Methodologies: Strength and Weakness (A. H. Mohammad et. al, 2013)

Every approach has its advantages and disadvantages. The chosen paper highlights the agile methods that have been proved to be best but at the same time has some drawbacks as well. Agile method deals with high level of customer satisfaction and instant response to bring change in a process. It is a flexible approach having short development cycle. Agile method core is customer who is considered as one member of the development team. Team members and management engage at same level enhancing collaboration. Project is developed and delivered side by side. At the other end paper extracts out the weakness of agile in a way that when we say its core is customer interaction, what if the customer is not well aware of his requirement and is stubborn to understand developer proposed solutions. What if customer doesn't have enough time to have meeting with the team members and it also requires that team and customer are located in same place which is not possible in many cases. Lastly, agile lacks for long term planning and has weak documentation record because of fluctuating requirements. [9]

2.10 Agile Software Development and Challenges (K. Jammalamadaka et. al, 2013)

Agile nowadays winning out customer's satisfaction, lowering company's workload and helping to cope up with changeable requirements. The paper classifies the software models as predictive and adaptive. Predictive is the waterfall model as it involves assumptions that requirements are clearly understood and will notchanged. Whereas, adaptive name is suggested for agile which obeys the rule of KIS (keep it simple), projects are delivered and accepted in chunks, involves customer feedback. Customer can have face-to-face communication with the developers. Any change can be accommodated in any part of the project anytime just before delivering the final product. But still agile cantbe considered as a silver bullet for the software development as it has his own challenges and needs which should be fulfilled. It is not feasible approach where there is communication gap, issues of time zones and language barriers among the team members and the customers. [10]

2.11 Agile Software Development: Impact on Productivity and Quality (A. Ahmed et al, 2010)

According to author to cope up with dynamic software progress agile approach should be followed as it fulfills majority of the quantitative measures. Agile promotes self-formulating teams, customer participation, higher quality and reduced time to market. Whereas, while adopting waterfall chances are that many projects may fail because of rigid rule of not changing requirements and lack of customer involvement. Agile methodologies emphasizes on a strong customer and developer relationship. It focuses on working release product rather than forming large documentation. Agile is a flexible approach which compensate changes while implementing hence encouraging the customer interaction and increasing productivity. [11]

ISSN 2348 – 8034
Impact Factor- 3.155

2.12 Agile testing with Scrum- A Survey (C. Malhotra et. al, 2013)

As opposed to other conventional approaches like waterfall, agile methodology is more flexible and adoptable, as here the process of testing is started as early as the requirement and planning phase. Each iteration in Scrum consists of short time boxes. It enhances reliability, repeatability, comprehensiveness and efficiency, which is the basic building block of agile. The survey conducted in this paper concludes that due to the iterative nature of agile and its methodologies including Scrum, Automation is a viable solution. In this paper researchers can get a view of automation in agile testing and various techniques in which it can be done. For the management of activities of Scrum, there are many scrum management tools available. [12]

2.13 Software Quality and Assurance in Waterfall model and XP - A Comparative Study (S. J Khalaf et. al, 2008)

Waterfall models falls under sequential process, is effective for large-scale projects but possess some major drawbacks. Waterfall process works in a linear manner firstly, requirements are gathered then design is formed third step is to implement the design and lastly testing is done. Each phase output becomes the input of the next phase so no two phases can be worked on, in parallel. Customer are just involved at the time of gathering requirements and when delivering the product. On the other hand Agile overcomes the huddles inherited by Waterfall model it promotes customer meetings with developers, focuses on short span deliverables and acceptance testing is done many time to satisfy the customers. The mind-set of involving customer on site enables to upgrade productivity, quality and strength of agile methods. [13]

2.14 Escape the Waterfall: Agile for Aerospace (S. H. VanderLeest et. al)

The paper discusses the comparison of waterfall approach and iterative approach into a traditional DO-178B process for aerospace projects. In a traditional DO-178B waterfall process, initially plan is developed and standards are set. Next step involves gathering stable requirements and then forming an architecture, which is further implemented, no later change is welcomed in waterfall process. Lastly, analysis is done and the certification of the project is made. The working with waterfall results out that it works efficiently for projects having pre-defined unchangeable requirements with less complexity. However, it leads to late integration, erratic estimates and time. Agile is considered as an approach, which highly motivates iteration. Projects are made in form of chunks all relics are initiated and upgraded during iteration. Agile supports changeability at any time, encourages customer involvement and can bring forth better results in aerospace projects. [14]

2.15 Analysis of Software Development Models (Dr. D. Jamwal, 2010)

Dr. D. Jamwal is of view that no software model is perfect they might posses factor of uncertainty. The processes differ in their methodologies, techniques, cost, schedule, design implementation and do propose some challenges that makes them imperfect. However, these methods are considered appropriate according to the organization and customer's requirements. The paper represents comparison of Waterfall and Agile models with respect to their development stages. In waterfall, requirements are gathered at every initial point that are unchangeable throughout the process of development whereas agile can fit-in any change anytime. Waterfall relies over large team but low cost whereas;a small team can perform agile task but expertise required which definitely results into high cost. Waterfall contains risk of success, agile grantees success. Further, waterfall product is deliverable in big-bang form i.e. after whole of its development. On the other side agile delivers products in prototype form and get customers feedback in parallel to work. [15]

III. ANALYSIS

Waterfall and Agile are best in their own ways. The survey papers emphasis over mean time failure, defect density, customer satisfaction, customer problem, cost-effectiveness, flexibility, complexity and on-time delivery which are explained below in tabular form. Also some more standardized ISO 9126 Software Quality Characteristics are mentioned in tabular form in table: I and elaborated in table II, III.

ISSN 2348 – 8034
Impact Factor- 3.155

paper can form the basis for selecting a suitable process for high quality software development.

I) EVALUATION CRITERIA FOR STANDARDIZE ISO 9126 QUALITY MEASEUREMENT IN AGILE & WATERFALL

PARAMETERS	DEFINITION	VALUE
Efficiency	The ratio of the useful work performed by a machine	Y, N, N.D
Availability	The proportion of time a system is functional	Y, N, N.D
Performance	Response of the system within a specific time span	Y, N, N.D
Reliability	Ability of system to work properly without failing	Y, N, N.D
Security	Capability to prevent malicious actions	Y, N, N.D
Integrity	State of a system performing its intended functions without being disruption by its internal or external environment	Y, N, N.D
Usability	How well system meets the needs of user	Y, N, N.D
Reusability	System or sub-system able to integrate with other applications	Y, N, N.D
Scalability	Able to handle load without effecting the system performance	Y, N, N.D
Maintainability	Undergo any change without bringing adverse effects	Y, N, N.D
Manageability	Ease of system to handle the work efficiently	Y, N, N.D
Testability	Measure of how easy it is to generate test criteria for a system and its components, execute these tests in order to determine if the criteria are met	Y, N, N.D
Supportability	Ability of the system to provide information helpful for identifying and resolving issues when system fails to work correctly	Y, N, N.D
Interoperability	Ability of a system to successfully exchange information with other external systems	Y, N, N.D

II) ANALYSIS OF STANDARDIZE ISO 9126 QUALITY MEASEUREMENT IN AGILE & WATERFALL

S. no	Techniques	Efficiency	Availability	Performance	Reliability	Security	Integrity	Usability
1	Y. Gao, 2010	Y	N.D	N.D	Y	N.D	N.D	N.D
2	A. Irshad et. al, 2011	Y	N.D	N.D	Y	N.D	N.D	Y
3	A. Gupta, 2014	Y	Y	N.D	Y	Y	N.D	Y
4	R. Kaur et. al, 2011	N.D	N.D	Y	N.D	N.D	N.D	N.D
5	Y. Bassil, 2012	N.D	Y	Y	Y	Y	N.D	N.D
6	K. Petersen et. al	N.D	N.D	Y	Y	N.D	N.D	N.D
7	M. Sagheer et.al, 2015	Y	N.D	Y	Y	Y	N.D	N.D
8	H. Hajjdiab et. al, 2013	Y	N.D	Y	N.D	N.D	N.D	N.D
9	Dr. A. H. Mohammad et. al, 2013	N.D	N.D	Y	Y	N.D	N.D	Y
10	K. Jammalamadaka et. al, 2013	Y	N.D	Y	Y	N.D	N.D	Y
11	A. Ahmed et. al, 2010	Y	N.D	Y	Y	N.D	N.D	Y
12	C. Malhotra et. al, 2013	Y	N.D	N.D	Y	N.D	N.D	N.D
13	S. J Khalaf et. al, 2008	N.D	Y	Y	N.D	N.D	N.D	Y
14	S. H. VanderLeest et. al	Y	N.D	Y	Y	Y	N.D	N.D
15	Dr. D. Jamwal, 2010	Y	Y	N.D	N.D	N.D	N.D	Y

ISSN 2348 – 8034
Impact Factor- 3.155

III) ANALYSIS OF STANDARDIZE ISO 9126 QUALITY MEASEUREMENT IN AGILE & WATERFALL

S. no	Techniques	Reusability	Scalability	Maintainability	Manageability	Testability	Supportability	Inter operability
1	Y. Gao, 2010	N.D	N.D	N.D	N.D	N.D	N.D	N.D
2	A. Irshad et. al, 2011	N.D	N.D	Y	Y	Y	Y	N.D
3	A. Gupta, 2014	Y	N.D	Y	N.D	Y	N.D	Y
4	R. Kaur et. al, 2011	Y	Y	Y	Y	Y	Y	N.D
5	Y. Bassil, 2012	Y	Y	Y	N.D	Y	N.D	N.D
6	K. Petersen et. al	N.D	N.D	N.D	N.D	N.D	N.D	N.D
7	M. Sagheer et.al, 2015	Y	Y	Y	N.D	Y	N.D	Y
8	H. Hajjdiab et. al, 2013	N.D	N.D	N.D	N.D	N.D	N.D	N.D
9	Dr. A. H. Mohammad et. al, 2013	Y	Y	N.D	Y	Y	Y	Y
10	K. Jammalamadaka et. al, 2013	Y	N.D	Y	Y	Y	N.D	N.D
11	A. Ahmed et. al, 2010	Y	N.D	Y	Y	N.D	Y	N.D
12	C. Malhotra et. al, 2013	N.D	N.D	N.D	N.D	Y	N.D	Y
13	S. J Khalaf et. al, 2008	N.D	N.D	Y	Y	N.D	N.D	N.D
14	S. H. VanderLeest et. al	N.D	N.D	Y	Y	N.D	Y	N.D
15	Dr. D. Jamwal, 2010	Y	Y	N.D	N.D	Y	Y	Y

IV) PARAMETER SELECTION (SCOPE OF RESEARCH)

PARAMETERS	DEFINITION	VALUE
Mean time to failure	The length of time a product is expected to last in operation.	Y, N, N.D
Defect density	The number of confirmed defects detected in software during a defined period of development	Y, N, N.D
Customer satisfaction	Measure of how products surpass customer expectation.	Y, N, N.D
Customer problem	Documentation about a problem with a product or service.	Y, N, N.D
Cost effective	Producing good results without costing a lot of money.	Y, N, N.D
Flexibility	The ability to be easily modified.	Y, N, N.D
Complexity	Hard to understand.	Y, N, N.D
On- Time Delivery	Measure of process and supply chain efficiency which measures the amount of finish goods or services delivered to customers on time and in full.	Y, N, N.D

I.

V) ANALYSIS OF RESEARCH

S. no	Techniques	Mean time to failure	Defect density	Customer satisfaction	Customer problem	Cost effective	Flexibility	Complexity	On- Time Delivery
1	Y. Gao, 2010	N.D	N.D	N.D	N.D	N.D	Y	Y	N.D
2	A. Irshad et. al, 2011	N.D	N.D	Y	Y	Y	Y	Y	Y
3	A. Gupta, 2014	N.D	N.D	N.D	N.D	Y	N.D	N.D	N.D
4	R. Kaur et. al, 2011	N.D	Y	Y	Y	Y	N.D	Y	N.D
5	Y. Bassil, 2012	N.D	N.D	N.D	N.D	N.D	N.D	Y	N.D
6	K. Petersen et. al	N.D	N.D	N.D	N.D	N.D	N.D	Y	N.D
7	M. Sagheer et.al, 2015	N.D	N.D	Y	N.D	Y	Y	N.D	N.D

8	H. Hajjdiab et. al, 2013	N.D	N.D	N.D	N.D	Y	Y	N.D	N.D
9	Dr. A. H. Mohammad et. al, 2013	N.D	N.D	Y	N.D	N.D	Y	Y	Y
10	K. Jammalamadaka et. al, 2013	Y	N.D	Y	Y	Y	Y	N.D	Y
11	A. Ahmed et. al, 2010	N.D	Y	Y	Y	N.D	N.D	Y	Y
12	C. Malhotra et. al, 2013	N.D	N.D	N.D	N.D	Y	N.D	N.D	Y
13	S. J Khalaf et. al, 2008	Y	N.D	Y	Y	Y	N.D	Y	N.D
14	S. H. VanderLeest et. al	N.D	Y	Y	Y	N.D	Y	N.D	N.D
15	Dr. D. Jamwal, 2010	N.D	N.D	Y	N.D	Y	Y	N.D	Y

NOTE: [Y: Yes, N: No, N.D: Not Defined]

IV. CONCLUSION

The paper ends in conclusion that we see that the waterfall approach is basically predictable and Agile aims on adaptability. So we say, Agile is successors possessing all the advantages of waterfall model and has also rectifies its flaws. Agile gifts with facilities of backward scalability, bug free solutions, rapid development and is adaptive to changing environment.

V. FUTURE WORK

In future, the manager can mingle with waterfall and agile benefits for outshining results. It is no wrong to say that agile hailed by many ways towards perfect solution for all types of SDLC related issues. Developers can fruitfully amalgamate agile in a waterfall domain to enhance outcome, making it cost effectiveness, coping with rapid changes, winning out customer satisfaction thus leading towards eventual success of an organization.

REFERENCES

1. *Y. Gao. "Research on the Rule of Evolution of Software Development Process Model", IEEE Conference, 2010, On page(s) : 466 - 470*
2. *A. Irshad, R. J. Qureshi & U. Ali. "A Comprehensive Study of Commonly Practiced Heavy and Light Weight Software Methodologies", International Journal of Computer Science, Issues, Vol. 8, Issue 4, No 2, 2011, On page(s) : 441 - 450*
3. *A. Gupta. "Quality Assurance and Its Standards: Importance in Various SDLC Models", International Journal of Advance Research in Computer Science and Management Studies, Issues 12, Vol 2, 2014, On page(s) : 143 - 148*
4. *R. Kaur, Dr. J. Sengupta. "Software Process Models and Analysis on Failure of Software Development Projects", International Journal of Scientific & Engineering Research , Issues 2, Vol 2, 2011, On page(s) : 1 - 4*
5. *Y. Bassil. "A Simulation Model for the Waterfall Software Development Life Cycle", International Journal of Engineering & Technology, Vol. 2, No. 5, 2012*
6. *K. Petersen, C. Wohlin& D. Baca. "The Waterfall Model in Large Scale Development", Book Title: Product-Focused Software Process Improvement, 2009, On page(s) : 386 - 400*
7. *M. Sagheer, T. Zaffar& M. Sirshar. "A Framework For Software Quality Assurance Using Agile Methodology", International Journal of Scientific & Technology Research, Issues 2, Vol. 4, 2015, On page(s) : 44 - 50*
8. *H. Hajjdiab& S. Taleb. "Adopting Agile Software Development: Issues and Challenges", International Journal of Managing Value and Supply Chains, Vol. 2, No. 3, 2011, On page(s) : 1 - 10*
9. *Dr. A. H. Mohammad, Dr. T. Alwada'n& Dr. J. Ababneh. "Agile Software Methodologies: Strength and Weakness", International Journal of Engineering Science and Technology, Vol. 5, No. 3, 2013, On page(s) : 455 - 459*

ISSN 2348 – 8034
Impact Factor- 3.155

10. *K. Jammalamadaka& V. R. Krishna. "Agile software development and challenges", International Journal of Research in Engineering and Technology , Issues 8, Vol. 2,2013, On page(s) : 125 - 129*
11. *A. Ahmed, S. Ahmed, Dr. N. Ehsan, E. Mirza& S. Z. Sarwar. "Agile Software Development: Impact on Productivity and Quality", IEEE Conference, 2010, On page(s) : 287 - 291*
12. *C. Malhotra& A. Chug. "Agile testing with Scrum- A Survey", International Journal of Advanced Research in Computer Science and Software Engineering", International Journal of Advanced Research in Computer Science and Software Engineering, Issues 3, Vol. 3, 2013, On page(s) : 452 - 459*
13. *S. J Khalaf& M. N. Al-Jedaiah. "Software Quality and Assurance in Waterfall model and XP - A Comparative Study", Book Title: Wseas Transaction on Computers, Issues 12, Vol. 7 , 2008, On page(s) : 1968 - 1976*
14. *S. H. VanderLeest& A. Buter. " Escape the waterfall: agile for aerospace", IEEE Conference, 2011, On page(s) : 6.D.3-1 - 6.D.3-16*
15. *Dr. D. Jamwal. "Analysis of Software Development Models", International Journal of Computer Science and Technology , Issues 2, Vol. 1, 2010, On page(s) : 61 - 64*