

Assessment of Complex Problems and Quality Improvement using Agile Software Development

M. Ejaz Akram, Aqsa Arshad, Asif Ashfaq

Department Of Computer Science, University Of Agriculture, Faisalabad Pakistan

ABSTRACT— Agile programming improvement is a substitute approach to conventional project administration utilized as a part of programming advancement. Despite the fact that from the most recent one decade the quantity of agile designers is expanding. All through the advancement procedure agile improvement philosophy gives chances to give the method for a project. Explores neglected to accomplishing the objective identified by programming item. In any case, there is a number of complex issues in programming improvement. The target of this examination to recognize the mind boggling issue emerges at some phase in the agile programming improvement. Amid the product advancement agile technique used to change and get chance less items. Agile programming strategies used to make it adaptable when contrasted with others creating methods. I utilized the deductive strategies to break down the mind boggling issue. At that point checked the issue with little, medium and huge venture. I utilized the agile procedure to conquer these issues and complex free improvement.

Keywords: Complex Problems, Quality Improvement, Software Development, Agile Software Development, Extreme Programming, Software Prototyping, Waterfall Development.

1. INTRODUCTION

The Agile Software bundle Improvement is really delineated appropriate here from the Agile Manifesto and is as often as possible, exceedingly iterative and incremental albeit versatile rather than thoroughly considered and by and large lays eyes upon a great deal less formal procedures and substantially less drawn out models when contrasted and conventional programming program anatomist. Normal Software bundle Executive is really laid out ideal here on the grounds that the absolute best practice programming program anatomist that was embraced before coordinated programming program change went along. It turned out thoroughly considered, profoundly iterative and incremental, and as of now utilized various models of the program procedure as of now being made, as exemplified from the real Rational Unified Method [1].

Traditional teaching, furthermore learning has by and large when contrasted with absolutely not dismissed the general concept of enthusiastic instructing furthermore learning inside instructive climate. The most recent logical tests indicate the utilization of advances, workforce face off regarding furthermore understanding, sharing period has developed favored alternatives to advance energetic learning technique. For a long time, your instructive group might do tests furthermore thinks about as an approach to building understudies' commitments in their gaining from the classroom [2].

The heavyweight strategies, likewise thought to be conventional techniques, more often than not concentrate on

thorough arranging, substantial documentation, and huge plan in advance. In the course of recent decades, numerous product improvement techniques have been made and used in the product business. Every strategy has distinctive elements and qualities that recognize it from different strategies; when all is said done, these techniques can be named either a heavy weight or a light weight technique. Interestingly, the lightweight techniques, otherwise called spry strategies, focus more on people and connections than procedures and apparatuses, more on working programming than far reaching documentation, esteem client coordinated effort more than contract arrangement, and concentrate more on reacting to change than taking after an arrangement [3].

In contrast to standard procedures, everyone is the primary concentrate of the Agile procedures. Agile changeover involves involvement, in addition to the cooperation of all the application practitioners like builders, enterprise experts, project in addition to department managers, elderly managers, consumers, and so on. Various problems in addition to troubles are usually experienced by means of application corporations throughout their Agile changeover. People-centric character these procedures has resulted in a few complications as soon as application corporations are usually changing their own improvement model by standard in order to Agile [4].

Agile software development technique, since specified within the manifesto, features a number of ideals: persons along with relationships, working software, client collaboration, along with answering and adjusting adjust. Your technique is actually

put in place by simply several strategies such as: Scrum, serious selection, along with agile modeling. It helps producing perhaps shippable working software at standard time periods, that helps to supply buyers quality functions eventually. It applies a new greedy-like technique having partial data for picking uses to formulate. Your technique relies upon using behavior, ideas, along with guidelines for building very good software. Your technique, retaining several instructional classes associated with software, including Web software [5].

Expansive scale coordinated programming advancement was the point of a workshop at the International Conference on Agile Software Development (XP2013), and this article gives an account of the exchanges at the workshop that prompt to a proposition for an examination motivation. Coordinated programming advancement strategies were made for little, gathered improvement groups, however, are progressively connected in different settings.

Vast undertakings are currently led using coordinated methods, and regularly these ventures are basic for organizations or even countries. Further, "lithe and extensive undertakings" was voted the "top smoldering exploration

there are a few reviews on the subject, despite the fact that scaling light-footed strategies was distinguished as a testing point in 2003. There are already given some direction on the most skillful method to sort out vast coordinated tasks, for example, Eckstein's book Agile in the substance.

2 RELATED WORK

2.1 Software Development Life Cycle

SDLC is a process of structure or keeping up programming frameworks. Commonly, it incorporates different stages of preparatory advancement examination of post-improvement programming testing and assessment. It additionally comprises of the models and procedures that improvement groups use to build up the product frameworks, which the techniques shape the system for arranging and controlling the whole advancement handle. A product application or a data framework is intended to play out a specific arrangement of assignments. Regularly, this arrangement of assignments that the framework will perform gives very much characterized outcomes, which include complex calculation and handling. It is in this manner an unforgiving and dreary employment to administer the whole improvement procedure to guarantee that the finished result contains high level of trustworthiness and power, and in addition client acknowledgment. Accordingly, an orderly improvement prepares which can stress on the comprehension of the extension and many-sided quality of the aggregate advancement process is fundamental to accomplish the said characteristics of an effective framework. At present, there are two SDLC approaches which are used by most framework engineers, to be specific the customary improvement and dexterous advancement which

clarified in the next session. In area 4 we look into these two techniques in detail and recommended a few upgrades in taking after segment. At last, the conclusion is displayed.

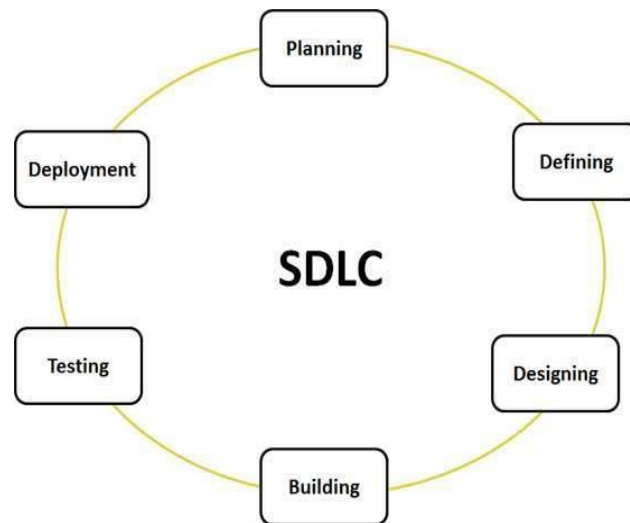


Figure 1: Software Development Life Cycle

2.1.1 Traditional Software Development

Programming philosophies such as Waterfall strategy, RUP and V-Model are called conventional programming improvement techniques and these are grouped into the heavyweight procedures. These approaches depend on a successive arrangement of steps like prerequisites definition, arrangement building, testing and organization. Customary programming advancement procedures require characterizing and recording a steady arrangement of prerequisites toward the start of a venture [6].

There are four stages which are normal for customary programming advancement technique. Once the necessities are laid out, the following stride moves into the outline and structural arranging stage where a specialized foundation is created as charts or models. The initial step is to set up the prerequisites for the project and decide the time span it will take to execute the different periods of advancement while attempting to foresee any issues that may emerge in the venture. These convey to the surface possible problems that the venture confronts it as advances and give a feasible point to designers to actualize.

Improvement is regularly separated into smaller assignments that are appropriate among different groups in light of aptitude. Once the group is happy with the building and configuration arrange, the venture moves into the advanced stage where code is created until the particular objectives are

come to. The testing stage frequently covers with the improvement stage to guarantee issues are tended to right off the bat. Once the nears fruition and the designers are near meeting the venture necessities, the client will turn out to be a piece of the testing and criticism cycle and the venture was conveyed after the client fulfill with it [7].

The customary programming improvement strategies are subject to an arrangement of foreordained procedures and current records which is composed the job advances and aides facilitate advancement. The achievement of a venture which is drawn nearer along these lines depends on knowing the greater part of the prerequisites before improvement start and implies that executing change amid the advancement life cycle can be to some degree risky. Notwithstanding, it likewise makes it less demanding to decide the expenses of the venture, set a timetable and distribute assets appropriately.

2.1.2 Waterfall Development

The waterfall model is a consecutive outline, prepare, utilized as a part of programming improvement firm, in which advance is viewed as streaming relentlessly downwards (like a waterfall) through the periods of origination, start, examination, plan, development, testing creation/usage and upkeep.

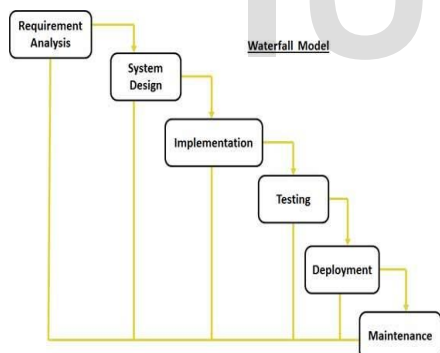


Figure 2: Water Fall Model

The exercises of the product advancement handle spoke to in the waterfall show. There are a few different models to speak to this procedure. The waterfall model is successive improvement technique, through a few stages, regularly in which development is viewed as streaming relentlessly downwards (like a waterfall).

- Requirements investigation, bringing about a product prerequisites determination
- Software outline
- Implementation
- Testing

- Integration, if there are numerous subsystems
- Deployment (or Installation)
- Maintenance

2.1.3 Software Prototyping

• It is the improvement approach of exercises amid programming advancement, the production of models, i.e., deficient forms of the product program being created.

- The essential standards are.
 - Not a stand alone, finish advancement strategy, but instead a way to deal with handle those parts of a bigger, more conventional improvement philosophy (i.e. Incremental, winding, or quick application improvement (RAD)).
 - Attempts to decrease the characteristic venture, chance of breaking a venture into littler sections and giving more simplicity of-progress amid the advancement procedure.
 - User is included all through the advancement procedure, which improves the probability of client acknowledgment of the last usage.
 - Until the model advances to meet the clients' prerequisites. Small-scale ridicule ups of the framework are created taking after an iterative alteration handle
 - It's conceivable sometimes to advance from model to working framework. While most models are created with the desire that they will be disposed of.
 - An essential comprehension of the basic business issue is important to abstain from taking care of the wrong issues.

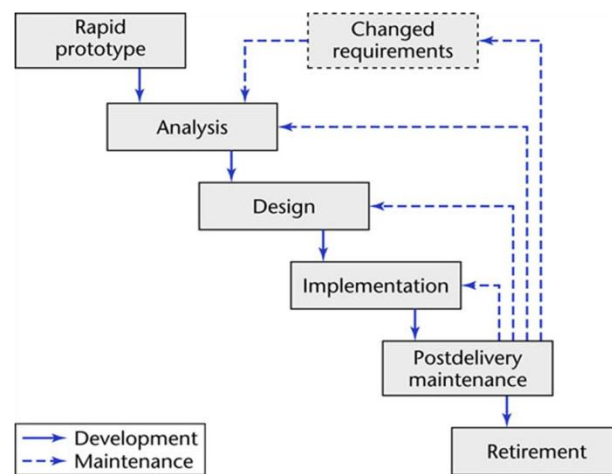


Figure 3: software prototyping

2.1.4 Incremental Development

Different techniques are worthy of joining straight and iterative frameworks advancement systems, with the essential target of each being to lessen the characteristic venture hazard by breaking a venture into littler sections and giving more simplicity of-progress amid the improvement procedure.

The essential standards are

- Overall, necessities are characterized before continuing to transformative, small scale Waterfall advancement of individual additions of a framework.
- All periods of the Waterfall are finished for a little part of a framework, before continuing with the following addition. A collection of smaller than standard Waterfalls are performed.
- The beginning programming idea, necessities investigation, and outline of engineering and framework center are characterized by means of Waterfall, trailed by iterative Prototyping, which comes full circle in introducing the last model, a framework working.

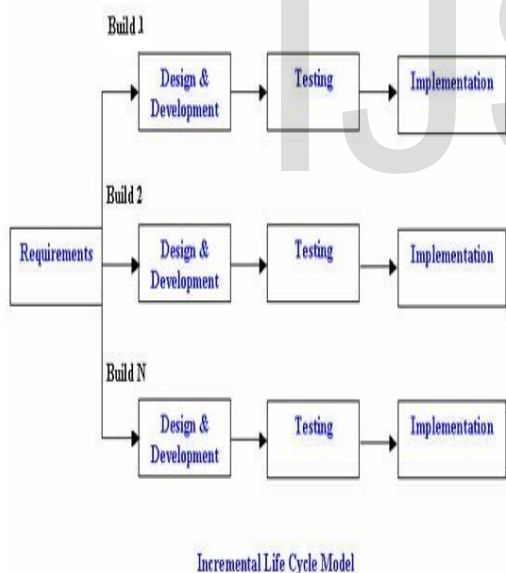


Figure 4: Incremental Life Cycle model

The spiral model introduced in 1986, and this model promised to deal with the risk factor in waterfall and prototype models, it is a more incremental development model that's why it became an integral of the development industry. It allows the project owner's to refine the prototype for the ultimate product. The refined prototype is then used during the process. It can be costly due to high amount of risk assessment. The risk assessment is the integral part of this model which requires high expertise because project success is completely based on risk

analysis. Building prototype in traditional style includes planning, designing, and building before finishing software. Companies now want results in a short time period. Therefore the spiral method is accepted very quickly in the industry because the drawback involves in waterfall methods has overcome. The reduction of high-level of risk is primarily focused and achieved in the earlier stage of development. The attraction of this model is that developer's can customize it for a particular project type.

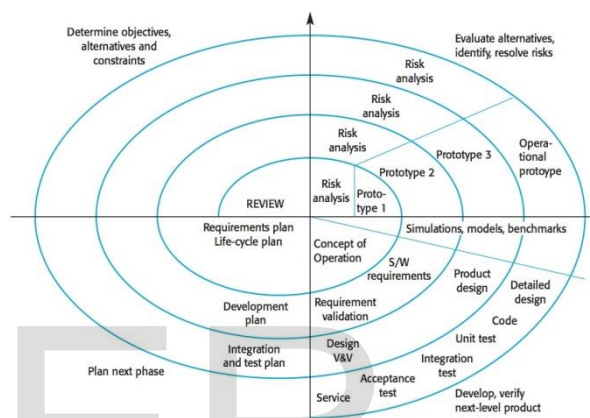


Figure 5: Spiral model

2.2 Extreme programming (XP)

As a type of dexterous programming advancement, it advocates visit "discharges" in short progress cycles, which is proposed to enhance profitability and present checkpoints at which new client prerequisites can be received. XP is programming process advancement which is expected to enhance programming quality and responsiveness to changing client necessities. The philosophy takes its name from the possibility that the gainful components of traditional programming structure practices are taken to "extraordinary" steps. Different components of extraordinary programming include: programming in sets or doing a broad code audit, unit testing of all code, abstaining from the programming of elements until they are really required, a level administration structure, effortlessness and clarity in code, expecting changes in the client's prerequisites over the long haul and the issue is better comprehended, and incessant correspondence with the client and among software engineers. For instance, code audits are viewed as a helpful practice; taken to the outrageous, code can be inspected consistently, i.e. the act of combining programming.

3 PROPOSED WORK

3.1 Quantitative Research and Qualitative Research

There are two concepts of research methodology that has been practiced most commonly 'Qualitative' research and 'Quantitative' research.

3.1.1 Quantitative Research

Quantitative research concern itself with 'judgment' based on statistics while qualitative research is 'judgment based on words' [8]. In quantitative research the methods used for collecting data are: online survey, interviews, and online poll. The methods used in quantitative are more structured than used in qualitative methods the advantage of quantitative research it is more objective because conclusion based on evidence, therefore minimizing bias views or prejudgment. The drawback of this method, it is possible the data collected from users is not accurate and do not represent the true answer.

3.1.2 Qualitative Research

Qualitative research focused on acquisition of peoples views and provide quality data that support the research question. Qualitative research is based on assumptions rather than numbered data. It is very useful in identifying trends in different opinion and very helpful in testing and developing further hypothesis. Qualitative method is exploratory because it reflects viewpoint of employees of a particular area. Qualitative research there is less emphasis placed on the typical causes and effect mentality while more on the experience involved. Typically interviews are used for data collection questionnaire data also used for gaining people's views.

3.2 Inductive and Deductive approaches

The deductive approach is often called "Top down" research approach. As is not agreeing with that and said this method" entails the development of a conceptual and theoretical structure prior to its testing through empirical observation. It is clearly stated that deduction research firstly comes with a theory and then later theory is being tested and observed which prove or disprove a given hypothesis. In this way project managers and project team evaluate a hypothesis based on a theory which is often based on problems found in development life cycle. The next stage is to test that hypothesis in a different way and at last the result either confirms the given hypothesis or reject which based on the result. Sometime it is elements of the hypothesis is modified or retested [9].

The induction or inductive research is reverse order of deductive research by In this type of research it starts with specific observation and end with a particular theory.

An example of this in the context of agile development methodology, with the hypothesis is that it is more suitable for small and medium size project. In order to develop a theory, hypothesis could be drawn from the observation and research will base on those hypotheses to develop a theory. Personal views are very important in this type of method.

3.3 Research Tools

3.3.1 Survey

Survey research involves selecting a representative based and impartial sample collected from the group you wish to study. The researcher will target the right people the subject chosen in this research. The common methods used for that are face to face, interviews, telephone. Survey research would allow researchers to collect the high amount of data in a very cost effective manner. There are several online tools used to conduct surveys like survey monkey, Google forms and few more online sites. Google form is used to create a survey question which is best and ore suitable choice for this area of research. The survey is conducted in two important stakeholder of software development project. The Questionnaire is created separately for these two stakeholders and sent to different software houses. There are two types of surveys descriptive and analytical survey. Descriptive survey concerned with identifying the frequency of a particular response of the survey group, whereas analytical survey, investigate and analyses the relationship between different elements in the sample group.

Advantage	Disadvantage
Cost effective	People may not feel encouraged and comfortable to provide true answers
No interviewer, respondents are more willing to share information	People may not be fully aware of their reasons for any given answer due to lack of knowledge.
More flexibility and reliability	Easier to ignore
Broad range of data can be collected	Issues with validity

3.4 Extreme Programming (XP)

XP, initially portrayed by Kent Beck, has risen as a standout amongst the most mainstream and questionable agile systems. XP is a trained way to deal with conveying brilliant programming rapidly and consistently. It advances high client contribution, fast criticism circles, constant testing, nonstop arranging, and close cooperation to convey working programming at extremely visit interims, ordinarily every 1-3 weeks. The first XP formula depends on four straightforward qualities effortlessness, correspondence, input, and bravery and twelve supporting practices:

In XP, the "Client" works intimately with the advanced group to characterize and organize granular units of usefulness alluded to as "Client Stories". The improvement group gauges, arranges,

and conveys the most noteworthy needy client stories through working, tried programming with an emphasis by-cycle premise. So as to boost profitability, the practices give a strong, lightweight system to control a group and guarantee superb programming.

Extrema programming endeavors to lessen the cost of changes in prerequisites by having different short improvement cycles, as opposed to a long one. In this regulation, changes are a characteristic, certain and attractive part of programming improvement extends, and ought to be made arrangements for, rather than endeavoring to characterize a steady arrangement of necessities. Outrageous Programming described portrays extraordinary programming as a product improvement teach that sorts out individuals to create higher-quality programming all the more constructively. Extraordinary programming likewise presents various essential qualities, standards and practices on top of the agile programming system.

3.6 Activities

3.6.1 Survey

Overview strategy concentrates the examining of individual units from a populace and the related review information accumulation strategies, for example, poll development and techniques for enhancing the number and precision of reactions to studies.

3.6.2 Requirement Elicitation

Necessities elicitation is the act of gathering the prerequisites of a framework from clients, clients and different partners. The practice is additionally once in a while alluded to as "prerequisite social affair".

The term elicitation is utilized as a part of research to raise the way that great necessities cannot simply be gathered from the client, as would be shown by the name prerequisites gathering. Necessities elicitation is non-trifling since you can make sure you never get all prerequisites from the client and the client by simply asking them what the framework ought to do (for Safety and Reliability). Prerequisites elicitation rehearses in corporate meetings, polls.

3.6.3 SPSS Tool

SPSS is a broadly utilized program for factual examination. It is utilized as a part of organizations, government, training specialists. The first SPSS manual depicted when you gather information from the association. The numeric esteem entered in

The SPSS apparatus In the type of table. In the wake of making tables recommend a diagrammatic system and draw the chart through SPSS for permitting standard analysts to do their own factual investigation notwithstanding measurable examination.

3.6.4 Testing

Unit tests figure out if a given component fills in as planned. A considerable measure of testing can take out numerous more defects. Astonishing programming methodology is that if a touch of testing can kill a couple imperfections.

3.5 Goals

Each bit of code that is composed is tried before proceeding onward to the following component. Engineer composes the greatest number of mechanized tests as they think about that may "break" the code; if all tests run effectively, then the coding is finished.

- By the developers fulfill the client's genuine prerequisites. Acceptance tests confirm that the necessities as comprehended

In any case, framework wide joining testing has been diminished, two weeks after week, or less regularly, contingent upon the dependability of the general interfaces in the framework. Framework wide combination testing was energized, at first, as an everyday movement, for early discovery incongruent interfaces, to reconnect before the different segments veered general usefulness.

3.6.5 Designing

From the perspective of straightforwardness, obviously one could state that framework improvement needn't bother with more than coding, testing and tuning in. On the off chance that those exercises are performed well, the outcome ought to dependably be a framework that works. By and by, this won't work. One can make considerable progress without planning yet at a given time one will stall out. The framework turns out to be excessively perplexing and the conditions inside the framework set up to be clear. One can maintain a strategic distance from this by making a plan structure that sort out the rationale in the framework. Great plan will stay away from loads of conditions inside a framework; this implies transforming one a player in the framework won't influence different steps of the framework. Another esteem, regard, was included the second version of the Extreme Programming explanation. Values XP at first perceived four values in 1999: correspondence, effortlessness, input, and valor.

4 RESULTS

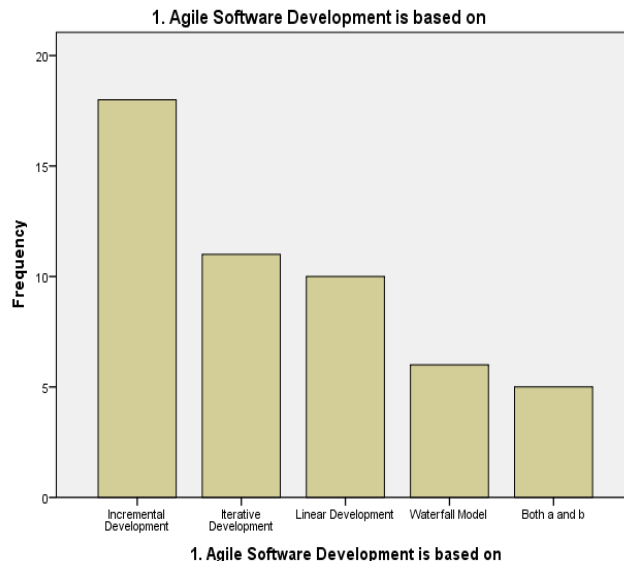
This part gives a detail examination of the data assembled in the midst of the data collection time of research. The data assembled from two basic regions. The one was semi-sorted out gatherings and the other was online survey; one study was followed in wander executives and second surveys in wander aggregate. These two people are the essential accomplice of any item wanders. An authoritative, objective was to gather information from wandering gathering and wander director in light of the fact that in composed wander these two people need to speak with each other all through the headway method.

The last data gathering system was considered in research was semi-sorted out gatherings. Each one of the all inclusive community was incorporated had an association in composing change technique. The fact of the matter was to look into the danger and challenges from routine change style to move in agile. Danger organization is a colossal issue in coordinated venture supervisor and regarded successful for this specific issue. Some question was raised to the developers and get suggestions which show in the form of graphs.

Table 1

Agile Software Development is based on:

	Frequency	Percent	Valid Percent	Cumulative Percent
Incremental Development	18	34.6	36.0	36.0
Iterative Development	11	21.2	22.0	58.0
Linear Development	10	19.2	20.0	78.0
Waterfall Model	6	11.5	12.0	90.0
Both a and b	5	9.6	10.0	100.0
Total Missing System	50	96.2	100.0	
Total	52	100.0		



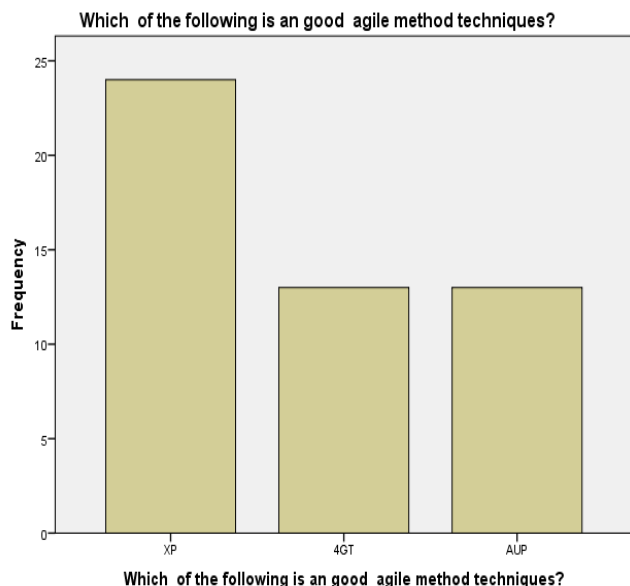
1. Agile Software Development is based on

This graph shows that the data send to the different organization to ask that Agile Software Development is based on the many techniques, but which one is better the developers replied that incremental Development is best Techniques. Data collected from 50 developers and 18 developers suggest that incremental Development is best Techniques 11 developers chose iterative development 10 developers chose Linear Development 6 developers suggest a waterfall model and 5 developers chose the both incremental Development and iterative development.

Table 2

Which of the following is a good agile method technique?

	Frequency	Percent	Valid Percent	Cumulative Percent
XP	24	46.2	48.0	48.0
4GT	13	25.0	26.0	74.0
AUP	13	25.0	26.0	100.0
Total	50	96.2	100.0	
Missing				
Total	52	100.0		



This graph shows that the data send to the different organization to ask that Agile Software Development is based on the many techniques. But which one is better the developers replied that Extreme programming is best Techniques. Data collected from 50 developers and 24 developers suggest that Extreme programming is best Techniques 13 developers chose 4GT and 13 developers chose AUP techniques.

5. CONCLUSION AND FUTURE WORK

Agile procedures are being acquainted in organizations to give a superior situation to programming advancement, where incremental work, visit input and adjustment to change are grasped values. While lithe strategies have been examined in a scope of productions, the creation and support of programming documentations once in a while shrouded in them. This may be brought on by the second explanation of Agile Manifesto working programming over extensive documentation", which is frequently translated in the way that dexterous improvement does not require documentation. While this might be valid in a few undertakings, quality specialized documentation is as a rule some portion of effective programming items. Kentico Software has transitioned from conventional to light-footed improve, yet a few sections of the executed system are still indistinct and flawed. One of these defects is the way toward making and keeping up specialized documentation of the created usefulness.

Complex issues in programming have a tendency to gather amid the improvement procedure by breaking down the enormous errands into smaller assignments, and by institutionalizing diverse phases of advancement, the main drivers of imperfections ought to be found and disposed of. The bigger the framework, the more associations that are included in its

advancement and the more parts are created by various groups, the more prominent the significance of having all around characterized connections between various merchants, so as to deliver a framework with easily interfacing segments. Amid a more drawn out time of improvement, a more grounded subcontractor system is much more helpful than transient benefit streamlining, which does not empower win-win connections.

ER

6. References

- [1] Aitken, A and V. Ilango. 2013. A Comparative Analysis of Traditional Software Engineering and Agile Software Development, System Sciences (HICSS), 46th Hawaii International Conference on, IEEE, pp. 4751-4760.
- [2] Cubric, M. 2013. An Agile Method for Teaching Agile in Business Schools. The International Journal of Management Education, pp: 119-131.
- [3] Schwaber, K. and B. Mike. 2012. "Agile Software Development with Scrum", Prentice Hall, pp. 89-94.
- [4] Badampudi, D., S. Fricker and A. Moreno. 2011. Perspectives on productivity and delays in large-scale agile projects. In Baumeister, H., Weber, B., eds.: 14th Int'l Conf. on Agile Processes in Software Engineering and Extreme Programming (XP '13). 149:180-194.
- [5] Ambler, S.W. 2009. The Agile Scaling Model (ASM): Adapting Agile Methods for Complex Environments, IBM,
- [6] Nikiforova, O., V. Nikulsins and U. Sukovskis. 2009.: Integration of MDA Framework into the Model of Traditional Software Development. In: Frontiers in Artificial Intelligence and Applications, Databases and Information Systems V, IOS Press, Amsterdam 187:229- 239.
- [7] Cohen, S. A. 2010. Software System Development

- Life Cycle Model for Improve Stakeholders' Communication and Collaboration. Int.J. of Computers, Communications & Control. 1: 20-4.
- [8] Digaali, M. 2014. An investigation into factors that contribute to I.T projects failure and successes. sheffield: sheffield hallam university .
- [9] Gill, A. Q. 2015. Agile enterprise architecture modelling :Evaluating the applicability and integration of six modelling standards. Information And Software Technology.

IJSER