A Study Of Advancements In Software Development Process Models And Their Usage In Recent Times.

Sudhir S.Soman

Assistant Professor
Tilak Maharashtra Vidyapeeth Pune.

Abstract

Software development process models are treated as a basis and starting point of software development process. There are various models available for development of software. Many times the models are used in combination. This paper is an attempt to study new emerging as well as upcoming software development models other than the seven basic models. The aim of the study is to find best suitable model according to the type of software to be developed.

Keywords: *Software engineering, software development process, software development process model*

Introduction: In the area of software engineering, software development process models play the most important role. It is considered that softwares in the world are developed by referring at least one process model available out of basic seven process models in software engineering. [1]
These basic seven software development process models are as follows.

1. Waterfall Model.
This model is a sequential approach. The process is plan driven. In this model, next stage cannot start unless and until previous stage is fully completed. This model is relatively rigid. [1]
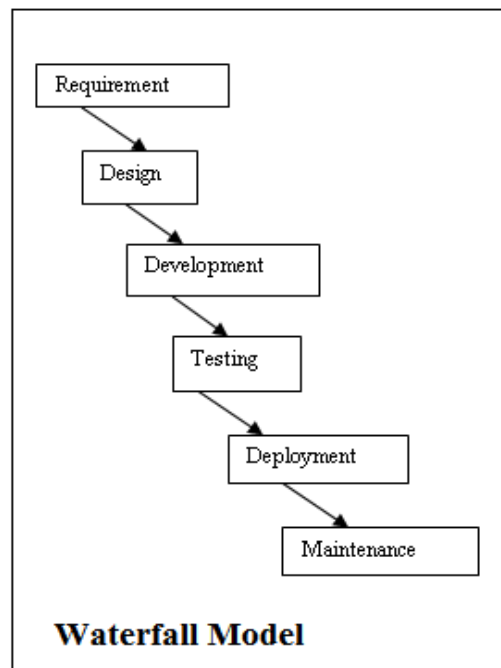

Fig. 1 Waterfall Model. [1]

2. V shaped Model.

V shaped model is linear model where in each stage has simultaneous testing activity. It gives quality control in the workflow of software development. It is one of the most expensive and time consuming models. [1]
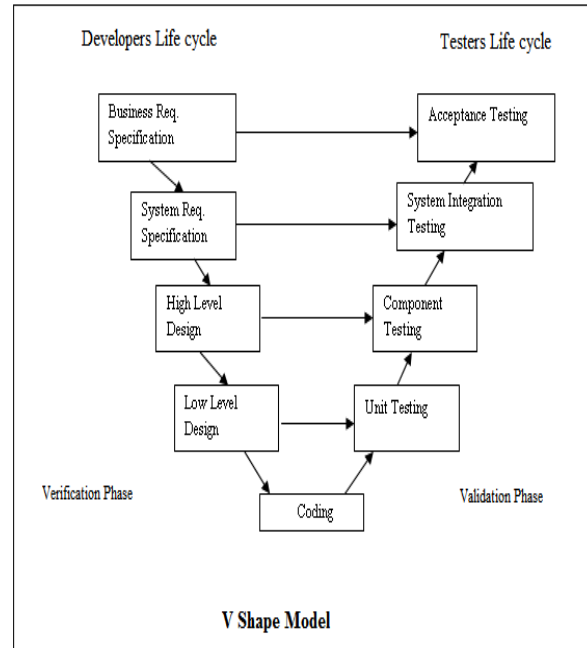

Fig. 2 V Shaped Model [1]

3. Incremental Model

In this model the software development process is divided into many iterations .New software modules can be added in every iteration. In this model the process goes either sequentially or in parallel which may increase the speed of software development. However it may make the project costly and long. [1]
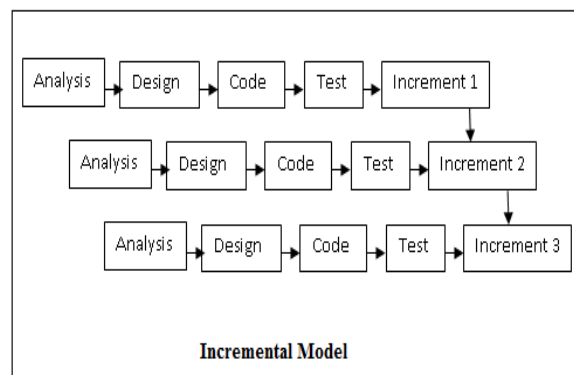

Fig. 3 Incremental Model [1]

4. Iterative Model

In this model, software development evolves and grows with each loop. The Software design remains consistent as each phase gets developed based on previous phase. Small changes in requirements, design

1658

and coding are possible in between the processes. One of the drawbacks in this model is many times major changes made in the system become an issue to the quality of the final product. [1]
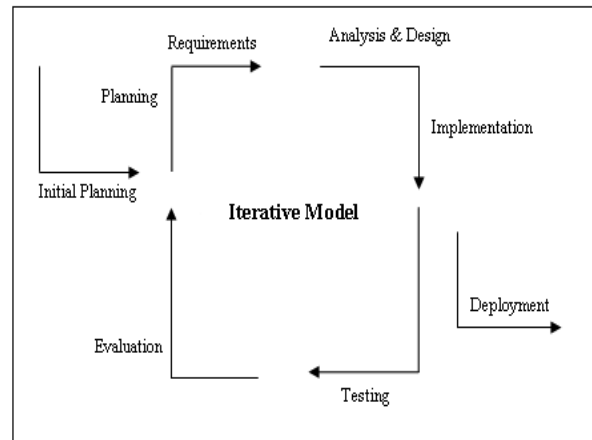

Fig. 4 Iterative Model [1]

5. Rapid Action Development (RAD) Model
This model is adaptive in nature. This model uses incremental prototyping approach so that the end user can give better feedback examining live system rather than only documents. However this model may evolve unexpected overruns in process cost and time schedules. [1]
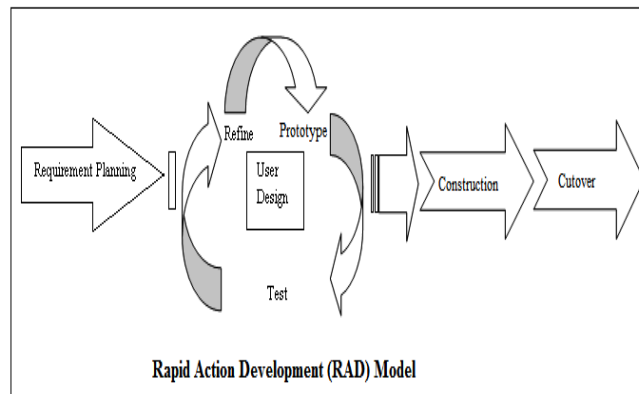

Fig. 5 Rapid Action Development Model [1]

6. Agile Model.
Agile model uses iterative development, greater communication and customer feedback ahead of schedule. This model targets quick delivery of working part of the software rather than detailed product documentation, product specification and product architecture description. The emphasis is more on testing. Agile model makes the development team and the customers to work closer and share feed back after every iteration. Agile model now comes in three subtypes too. These are namely Scrum, Extreme Programming, and Kanban. [1]
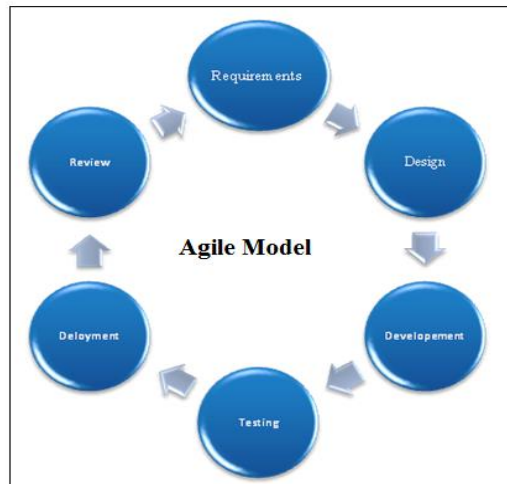
Fig. 6 Agile Model [1]

7. Spiral Model

In the spiral model more emphasis is placed on risk analysis. This model has four phases-
a. Planning
b. Risk Analysis
c. Engineering
d. Evaluation.
In this model a software project repeatedly passes through spirals .Baseline spiral, starts from the planning phase, requirements are gathered and risk are identified. Every next spiral builds on the baseline spiral. Firstly requirements are gathered in the planning phase. A process is undertaken to identify risk and alternate solutions in the risk analysis phase. In the end of the risk analysis phase, a prototype is produced. In the engineering phase, software is produced along with testing at the end of the phase. The customer evaluates the output of the project before date in evaluation phase. Then the project continues to the next spiral. [1]
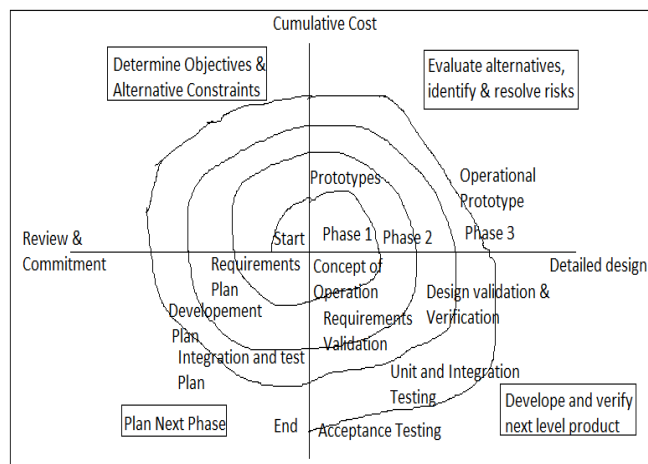

Fig. 7 Spiral Model [1]

Survey:

Recently, I carried a survey of a group consisting of some developers and computer science faculties using a Google docs questionnaire [11] to know whether software developers think SDLC is really important and they really need a model to follow; the results of the survey are as follows.

All people agree that S.D.L.C. i.e. System development life cycle is the basis of software development process. [11]

1. Do you agree that S.D.L.C. i.e. System development life cycle is the basis of software development process?
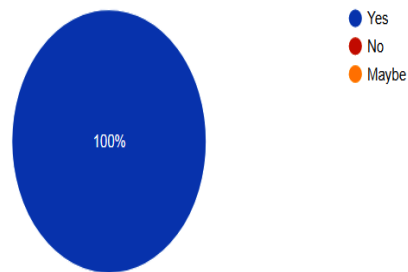
83.3% people agree that at least one of the process models like waterfall, spiral, V shaped, Agile etc. should be followed for software development. [11]

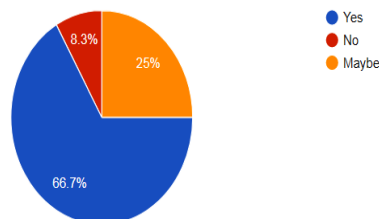2.Do you agree that at least one of the process models like waterfall, spiral, V shaped, Agile etc. should be followed for software development?
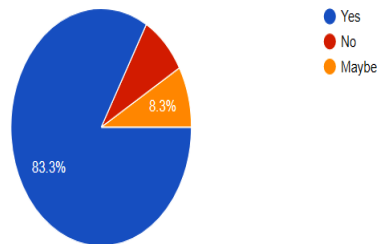

66.7 % said that we should use a combination of process models for software development. [11]

3.Do you think that instead of using a single process model you should use a combination of process models for software development?

58.3% people agree that if there is new idea of software design, there is a need of new model for software development for the improvement of the quality of software development process. [11]

4.Do you agree that if there is new idea of software design, there is a need of new model for software development for the improvement of the quality of software development process?
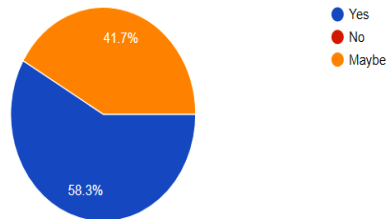

83.3% people agree that extensive changes in Software process modeling in the last three decades impacted the software process' structure, degree of control, degree of visualization, degree of automation and integration. [11]

5.Do you agree that extensive changes in Software process modeling in the last three decades impacted the software process' structure, degree of control, degree of visualization, degree of automation and integration?
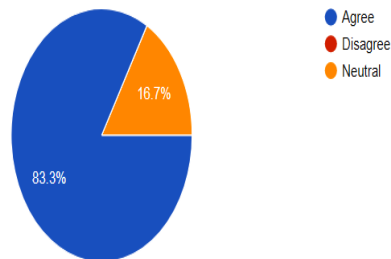
50% people think that in agent oriented software engineering, little attention has been given to the disciplining of the execution of the different phases involved in the software life-cycle. [11]

6.Do you think that in agent oriented software engineering, little attention has been given to the disciplining of the execution of the different phases involved in the software life-cycle?
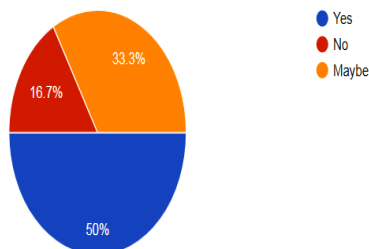
100% people agree that to fulfill every project specific requirement in software development and produce a software system with desired functionality and quality attributes, adequate development and management processes should be designed, adopted, implemented, operated and then upgraded as and when required. [11]

7.Do you agree that to fulfill every project specific requirement in software development and produce a software system with desired functionality and quality attributes, adequate development and management processes should be designed, adopted, implemented, operated and then upgraded as and when required?
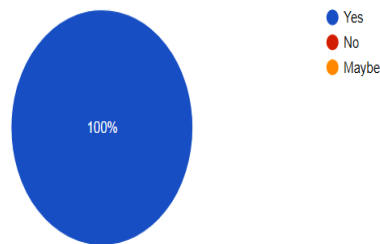
66.7 % agree that small software companies find major challenges in software process improvement model as these companies normally focus on time to market and innovation. [11]

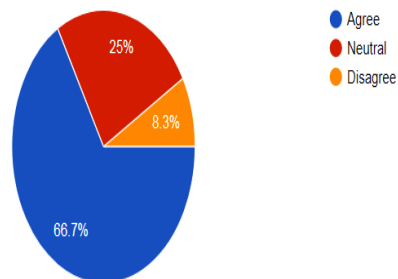8.Do you agree that small software companies find major challenges in software process improvement model as these companies normally focus on time to market and innovation?

58.3 % people agree that small software companies often ignore Software Process Improvement models such as CMM and CMMI. [11]

9.Do you agree that small software companies often ignore Software Process Improvement models such as CMM and CMMI?
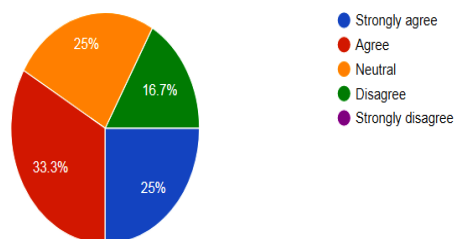

50% people agree that predefined approaches of software process modeling are not automatically adaptive to other software projects. [11]

10.Do you agree that predefined approaches of software process modeling are not automatically adaptive to other software projects?
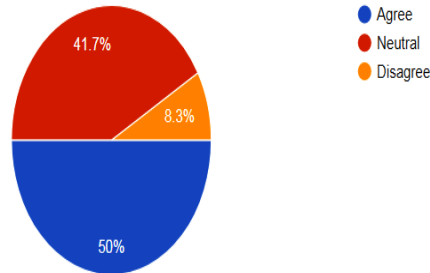
91.7 % people think Process modeling is a suitable method to be used in identifying problems and potential points of improvement. [11]

11.Do you think Process modeling is a suitable method to be used in identifying problems and potential points of improvement?
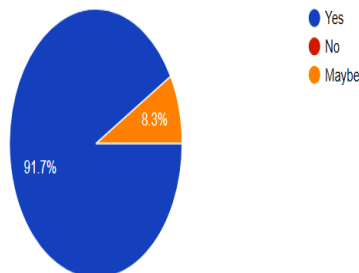
Literature Review:

In the paper "A Survey of Software Development Process Models in Software Engineering" the authors discuss about various software developments process models. A model is the sequence of processes in a software engineering development activity. Authors' objective is to represent various models of software development process and various aspects of every model to help the developers to select a particular model in given situation while considering customer demand at the same time. Authors' have made comparison of Waterfall, V-shaped, Incremental, RAD, Iterative spiral and Agile models on the basis of their advantages and disadvantages. They say that these comparisons will help the developers to choose a particular software development process model. [1]

In the article "A content centric development process model" the authors have discussed about video game development .They talk about the idea of teaching through various interactive videogames. They say that Videogame development is a complex task which requires specialized skills in areas of graphics processing, animation, sound, and buffering. Video game developers have to take different approaches to deal with the complexities which are present in programming compared to traditional programming. Authors have discussed about the cultural clash between instructors, game writers and developers. They say that this is common to the development of other content intensive applications such as educational programs and hypermedia. For developing content intensive applications, there is a need of collaboration

1664

between experts in the subject content and the software programmers. Authors have developed E-adventure process model in collaboration with the Spanish National Center of Information and Educative Communication (CNICE). The base of E-adventure process model is spiral or iterative process models such as the Rational Unified Process (RUP) which divides the process into several stages which can have iterations as and when required.[2]

In the paper "A modest but practical software process modeling technique for software process improvement" the authors have discussed about one of the main problems of software engineering which occur due to the difficulties in evaluating and improving software processes. They present a technique to improve software processes through modeling and evaluation. They propose an approach to improve the software development by omitting unnecessary phases from the process, by introducing new phases to the process and by improving the existing phases of the process. Authors state that the process modeling got positive reactions and it was seen as a way to improve the understanding of the process and the effort used for it was well spent. This Process modeling was considered a suitable method to be used in identifying problems and potential points of improvement. [3]

In the article "A multilateral negotiation method for software process modeling" authors state that in current scenario there are predefined approaches of software process modeling. These approaches are not automatically adaptive to other software projects and they do not provide good support to development team formations well as task and resources allocation in real time. They propose an agent-based multilateral negotiation model MNMPA for dynamic software process modeling and ease the work of team formation. Authors propose that the model MNM-PA will bring two advantages that the software processes will not be predefined and the software processes will be for current projects, development teams and allocated tasks and task constrains only. Authors conclude that their method provides an un-predefined software modeling and decision making support of team formation with task and resource allocation for different software projects in reality as compared to the traditional software process modeling. MNM-PA supports the both cooperative and competitive negotiation among the process agents by integrating negotiation with the one-time biding of the classic CNP. [4]

In the article "Adept: A unified assessment method for small software companies", authors have talked about some small software companies which find major challenges in software process improvement model. Newly started software companies often ignore software process improvement models such as CMM and CMMI. These companies normally focus on time to market, innovation etc. These newly started companies are attracted to agile methods which have shorter development schedules and greater delivery flexibility. Authors have studied the Irish software industry. In this Irish industry more than 60 percent of the Irish software firms employ10 or less people.[10] Authors state that the Irish software industry needs to educate its managers in software process improvement as well as in quality agenda. Marketing of Environmental Protection Agency (EPA) and Automated Application Development (AAD) has raised awareness of software Process improvement. Authors have performed assessments which raised the education level of the organizations' employees. [5]

In the paper "Building a hybrid process model for a complex software system integration", the authors of the paper state that to fulfill every project specific requirement in software development and produce a software system with desired functionality and quality attributes, adequate development and management processes should be designed, adopted, implemented, operated and then upgraded as and when required. Authors have proposed a hybrid process model for project management and software development in agile environment. In the conclusion authors have stated that the paper has discussed different potential methodologies for complex software system integration, with the real-life project case study described. Their experiences from the case study and lessons learned have served as a base for a new hybrid process model for complex software system integration. [6]

In the article "Category theory centric systems science and software systems engineering pursuing theoretical insights into systems engineering beyond Sysml / UML", authors discuss about functional agency approach based on category theory, Gurevich abstract state machines, homotopy type theory and Dagger theory including schemas theory as a basis for software systems engineering architectural design. The article states briefings on Dagger theory and Software Design Minimal Methods. Authors were attempting to describe categorical orientation or category theory centric approach to systems architectural design at the Software Engineering as well as Systems Engineering levels. And they found that category theory is not enough and it may take a patchwork of techniques, methods, approaches and orientations to take them into synergetic engineering from where they are and which is trying to design Sysml #2 and making UML ever more complex like C++ language has become.[7]

In the paper "Process models for agent based development" the authors have observed that very little attention has been given to the disciplining of the execution of the different phases involved in the software lifecycle. Authors focus on process models for software development and relate it with current researches in agent oriented software engineering. In this paper authors have introduced the key concepts and issues related to software processes and presented various process models which are already adopted in mainstream software engineering. They have surveyed the characteristics of various agent oriented methodologies and focused on identifying key limitations of current methodology centered researches. They found that in industrial projects the waterfall model should be replaced by flexible and iterative approaches like evolutionary and spiral ones. No single general purpose process model is effective for all projects. Different engineering and commercial and need different process models and the authors' further state that software processes need to be continuously assessed and improved. The duty of a software engineer is to apply his expertise to identify the most appropriate process model for any specific situation and implement that model in practice. [8]

In the paper "Tailoring the software process model to project requirements" the authors have shed light on extensive changes in Software process modeling in the last three decades which impacts the process structure, degree of control, degree of visualization, degree of automation and integration. Due to this, it resulted in a diversity of models that tended to suit software project requirements from various angles. Authors while taking efforts on building taxonomy of super classes of process models and their interrelationships have investigated the main streams of process models. They have carried an eight step evaluation process based on this taxonomy. Author further states that companies have became better integrated with their software applications and those applications have to be mapped in process modeling along with business targets. [9]

In the paper "Research process on software development model", the authors had analyzed and summarized twelve types of software development models which are applied in software development and they have even surveyed the characteristics of every model as well. They have analyzed Waterfall model, Incremental model, Spiral model, Fountain model, XP (Extreme Programming) model, RUP (Rational Unified Process) model, RAD (Rapid Application Development) model, WINWIN model, Intelligent model, Parallel model, Component based model, Architecture based model. Authors have observed that various researchers and software developers are making required improvements in existing development models according to the need of actual software projects in certain application fields. They are redesigning the processes or activities for new development models, and getting good results. Authors also state that whenever there is a new idea of software design, there is a need of new model for software development for the improvement of the quality of software development process. [10]

Conclusion:
For any new project Software development process should follow at least one process model. Now a day's depending on the software type to be developed a combination of two or more models can be used.

It is found that these combinations of models speedup the process of software developments by using advantages of each other and at the same time minimizing the drawbacks of each other.

References:

1. Iqbal H. Sarker, Faisal Faruque, Ujjal Hossen and Atikur Rahman , " A Survey of Software Development Process Models in Software Engineering", International Journal of Software Engineering and Its Applications, Vol. 9, No. , Issue 11 (2015), pp. 55-70. http://dx.doi.org/10.14257/ijseia.2015.9.11.05
2. Published by the IEEE Computer Society, 0018-9162/08/$25.00 © 2008 IEEE,"A Content-Centric Development Process Model", http://www.e-ucm.es/publications/articles.html for updated citation information.
3. Jarmo J. Ahonen, Marko Forsell, Sanna-Kaisa Taskinen, "A Modest but Practical
4. Software Process Modeling Technique for Software Process Improvement", Softw. Process Improve. Pract. 2002; 7: 33–44 (DOI: 10.1002/spip.152).
5. Nao Li1,, Qing Wang, Mingshu Li, Shuanzhu Du, and Junchao Xiao," A Multilateral Negotiation Method for Software Process Modeling", Wang, D. Pfahl, and D.M. Raffo (Eds.): ICSP 2007, LNCS 4470, pp. 147–158, 2007. © Springer-Verlag Berlin Heidelberg 2007.
6. Fergal Mc Caffery, Philip S. Taylor, Gerry Coleman, "Adept: A Unified Assessment Method for Small Software Companies" Published by the IEEE Computer Society 0740 - 7459 / 07 / $ 20.00 © 2007 IEEE, January / February 2007.
7. K. Smojver, H. Belani and Ž. Car," Building a Hybrid Process Model for a Complex Software System Integration", 10th International Conference on Telecommunications, July 2009, 147 - 153.
8. Kent Palmer, "Category Theory Centric Systems Science and Software Systems engineering", https://www.academia.edu/36446262/Category_Theory_Centric_Systems_Science_and_Software_Systems_Engineering, 2018.
9. Luca Cernuzzi, Massimo Cossentino, Franco Zambonelli, "Process models for agent based development", Engineering applications of artificial intelligence, volume 18, issue 2, march 2005, pages 205-222.
10. Osama Eljabiri and Fadi P. Deek, "Tailoring the Software Process Model to Project requirements", https://www.semanticscholar.org/paper/Tailoring-the-Software-Process-Model-to-Project-Eljabiri-Deek/e21d3b46bc78f60758ce06e561ce168754a658fe, 2005.
11. Jiujiu Yu, "Research Process on Software Development Model", IOP Conf. Series: Materials Science and Engineering (2018), 032045, doi:10.1088/1757-899X/394/3/032045, page 1-8.
12. 11.https://docs.google.com/forms/d/1vdJBC8hxb4PVL6Atz3j2l0iBvZ25Ji874YTRFuyXw0A/edit#responses
13. M. Crone, "The Irish Indigenous Software Industry: Explaining the Development of a Knowledge-Intensive Industry Cluster in a Less Favored Region," E,rsa 2002 Dortmund: 42nd Congress European Regional Science Assoc., Economic Research Inst. Northern Ireland, 2006; www.erini.ac.uk/publications/PDF/NiercConf2.pdf.
14. http://mrbool.com/what-are-different software development models/30120. 3.48 P.M.15-12-2020
15. Tilak, G. (2020). Artificial intelligence: A Better and innovative technology for enhancement and sustainable evolution in education system.
16. Tilak, G. (2020). Utilization of Cloud Computing in Higher Educational Institutions.