IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Implementation Of Agile Methodology For Distance Education

Ms.Aakanksha Dhuri[1], Mrs. Gunjan Behl[2]

[1] *MCA Final Year Student, Bharati Vidyapeeth's Institute of Management & Information Technology Navi-Mumbai.*

[2] *Professor, Bharati Vidyapeeth's Institute of Management & Information Technology Navi-Mumbai.*

Abstract: *The most common challenges faced nowadays by teachers at academic level is to attract or retain the attention of children and commitment towards education.Thus new and thrilling methodologies were developed to support teachers .The agile teaching/learning methodology is specifically designed for distance education from the field of software engineering and leveraging upon concepts from a agile software methodology. This paper explains the objectives which are behind agile teaching methodology and process architecture of the methodology used in distance learning . Agile methodologies that have pursued educational approaches and to provide a detailed explanation of features regarding educational context .*

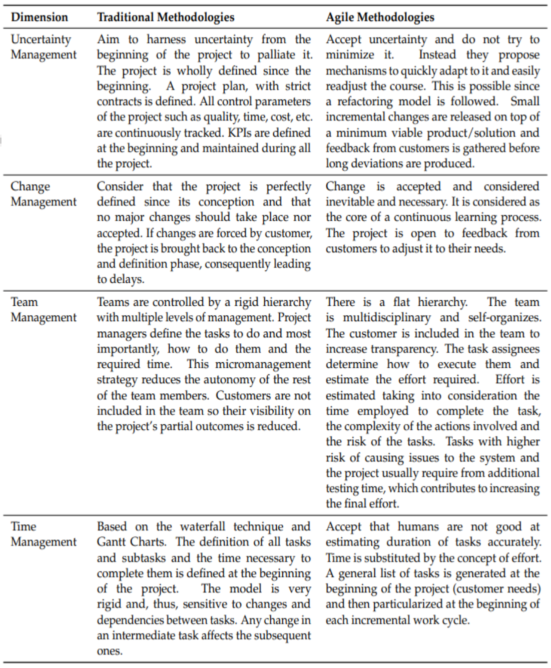
Key Words: Agile Methodology , Extreme programming, Scrum ,Software engineering

1.Introduction

The most used process framework for software development is agile. Agile teaching also promotes a communication, knowledge sharing and the learning process to encourage self learning individually. Agile teaching methodology is balanced methodology that helps both sides of the equation .If the qualities of people working on a project are good enough they can almost use any of the processes and accomplish their projects .In order to reach common goals efficiently agile makes people interact on excellent basis by using their individual talents in team.Software engineering(SE) practitioners have been long aware of the importance of grabbing a well defined methodology for a software development .software development is mostly complex process and it makes sense to have well defined steps, tasks and plans as well as understanding of As with many different techniques, there is no formal publication or work which could be considered as the conception of Agile methodologies. Different experiences with quick incremental definition-execution-evaluation-correction production cycles can be tracked back to the 1950s based on a publication from 2003 by Craig Larman and Victor Basili the dynamics behind the whole development life cycle and when to deal with expected and unexpected changes ,problems and risks.

In the past decade, their has been a slowly trend towards favoring a set of ”light weighted” software development methodologies called agile processes . Several methodologies have been stated within the agile culture for extending and implementing its values and principles i.e. eXtrem programming and Scrum. Learning must be Agile to cope up with changing research ,business and technology environments.This research paper aims to give an overview of those agile methodologies for which some research work is present in literature. The aim is to highlight it’s values, principle apply to improve students learning .

Table 1. Differences between Traditional Project Management and Agile Methodologies.



Dimension	Traditional Methodologies	Agile Methodologies
Uncertainty Management	Aim to harness uncertainty from the beginning of the project to palliate it. The project is wholly defined since the beginning. A project plan, with strict contracts is defined. All control parameters of the project such as quality, time, cost, etc. are continuously tracked. KPIs are defined at the beginning and maintained during all the project.	Accept uncertainty and do not try to minimize it. Instead they propose mechanisms to quickly adapt to it and easily readjust the course. This is possible since a refactoring model is followed. Small incremental changes are released on top of a minimum viable product/solution and feedback from customers is gathered before long deviations are produced.
Change Management	Consider that the project is perfectly defined since its conception and that no major changes should take place nor accepted. If changes are forced by customer, the project is brought back to the conception and definition phase, consequently leading to delays.	Change is accepted and considered inevitable and necessary. It is considered as the core of a continuous learning process. The project is open to feedback from customers to adjust it to their needs.
Team Management	Teams are controlled by a rigid hierarchy with multiple levels of management. Project managers define the tasks to do and most importantly, how to do them and the required time. This micromanagement strategy reduces the autonomy of the rest of the team members. Customers are not included in the team so their visibility on the project’s partial outcomes is reduced.	There is a flat hierarchy. The team is multidisciplinary and self-organizes. The customer is included in the team to increase transparency. The task assignees determine how to execute them and estimate the effort required. Effort is estimated taking into consideration the time employed to complete the task, the complexity of the actions involved and the risk of the tasks. Tasks with higher risk of causing issues to the system and the project usually require from additional testing time, which contributes to increasing the final effort.
Time Management	Based on the waterfall technique and Gantt Charts. The definition of all tasks and subtasks and the time necessary to complete them is defined at the beginning of the project. The model is very rigid and, thus, sensitive to changes and dependencies between tasks. Any change in an intermediate task affects the subsequent ones.	Accept that humans are not good at estimating duration of tasks accurately. Time is substituted by the concept of effort. A general list of tasks is generated at the beginning of the project (customer needs) and then particularized at the beginning of each incremental work cycle.

With this systematic review, we aim to analyze the relationships between Agile methodologies and Education that have given birth to Agile Education, as well as the competencies that are empowered through its application and their relation to ESD. We aim to confirm whether Agile Education conforms a valuable educational framework that helps education in modern societies in a more efficient and sustainable way.[4]

1.1 Background

Most of the students are familiar with ADDIE, the linear framework for building effective training solutions. Like anything in this world, ADDIE has its strengths (easy to understand and communicate, straightforward development phases) and its weaknesses (fairly rigid, has challenges incorporating changes in requirements late in the process).Agile has taken root in the eLearning industry because it addresses one of the fundamentally difficult challenges in eLearning development: namely, how to address late-breaking changes in requirements. An Agile approach can increase flexibility to a project while keeping the same deadline and budget, resulting in happy stakeholders and increased morale. However, not every project or organizational culture is a perfect fit for Agile.

1.2 Literature review

Alfonso botia in 2005 introduce Agile methodology were first methodlogy as software engineering courses in which a teacher manages students teams and make them practice real software project.Then it was found that agile methodology was also useful in teaching other subjects for eg.Mathematics(Duvall, Hutchings ,Kleckner,2017) .The aim is to highlight agile values principles and practises applied to improve students learning .This research paper give an overview of those agile methodologies for which research work is done. Stewart et al. presented a first review of the literature aimed at showing how agile methods were applied to education. Moreover, they provided a mapping between values and principles of the Agile Manifesto to specific educational methods and activities[2]

A search of JISE articles reveals that the first article proposing an Agile teaching approach appeared in McBride (2005) who applied the values of eXtreme programming as a model to teach an e-commerce course. In that same year, McAvoy and Sammon (2005) developed an adoption assessment matrix to assist in the selection of the appropriate Agile method for use in specific software projects. A component of the study proposed a pedagogical approach based on active learning to "improve the student's knowledge of the adoption of Agile methods" (p. 409) through participation in critical adoption factors workshops[1]

In 2001, a team of 17 leading software practitioners devised and published what they defined the "Manifesto for Agile Software Development". It is a document that defines the values and principles of the Agile software development movement as a summary of how they found "better ways of developing software by doing it and helping others to do it" (Beck et al., 2001).[2]

As stated by Stewart , "the correspondence between elements seen in the agile software development environment and the academic environment are striking. At first glance, the similarities can be easily seen. Both teaching and software development require detailed planning and scheduling. Each requires management and constant assessment and feedback from all involved. Making sure a course is delivered correctly and on time presents similar difficulties to those encountered in software development projects".[3]

1.3 Objective/Scope

The target users of the study are the Senior High School and Collegiate students who are currently a distance learner. The student models of learning were recognized on the premise of a particular example including those under e-learning study show and those under the conventional examination display. Also, the hypothetical idea of this investigation constrains its immediate pertinence for the instructive praxis. Along these lines, it is trusted that future research may moreover concentrate on how understudy perceptions about learning are affected with regards to ordinary learning conditions. eLearning, software development, and a number of other project-based activities follow a similar path

When you think of it, eLearning is like software development in that we're often building something unique to support the organization. Business sponsors can't articulate all of their needs and desires at the beginning of the project. It's likely that the organization's underlying needs will change along the way, too, and we want to be responsive to those changes. Each of these situations offers a compelling reason to adopt Agile methods.

2. Research Methodology

In this part, the research methodology is described, including the design science research strategy and how it is implemented in this study. A description of the study's methodology is provided, followed by an assessment of its strengths and weaknesses.

2.1 Research approach

2.1.1WHY TO APPLY AGILE METHODOLOGY FOR DISTANCE EDUCATION?

Agile is a project management methodology used commonly in software development industry .The agile models serves precisely to avoid these waste of resources and principle of dividing the project into the smaller phases .It is perfectly suited to creation of a modules for an online course . Agile has been adopted by e-learning due to its elastic nature and flexibility to changing clients. Agile delivers clients what they need effectively. As you know the process of creating an online course contains several months of work. If the interaction with the client were limited to monitoring meetings or even at the end of the course creation, there would be a risk that all the work done by the instructional designer and his team would not gradually meet the clients needs.

2.1.2BENEFITS

In recent times, Agile has been adopted by e-learning production teams due to its elastic feature, which the gives the permission for development process to respond differently to changing client needs.

Agile differs from the traditional ADDIE model, a more permanent end-toend process where client needs are defined at the starting before development and production takes place, resulting in a singular version of the product released at the end. In contrast, Agile focuses on adaptation, evolving development, rapid prototyping, and recent feedback and evaluation.

1. Responds flexibly to changing client needs

When it comes to corporate projects, nothing is "set in stone." During an e-learning production lifecycle, there are proposal for content additional, processes can differs, or new stakeholders wielding subjective opinions can met the mix. Developers need to be able to utilize variances with ease way and professionalism, and Agile allows them to do just that due its ability to bring and respond to make through frequent iterations.

2. Allows more feedback opportunities

In an Agile e-learning development process, random iterations of the product are released. On each release, a client can give feedback to incorporate into the design along the production way, rather than holding input until set review points.

3. Delivers client wants and needs effectively

In most cases, internal e-learning projects require building something unique for an organization to assist in meeting business needs. Because there is usually not an existing template, there are inherent challenges in successfully interpreting the client vision and infusing it into the incomplete product. With Agile, client involvement is an integral part of the process, bringing it much great to likely make deliver a product that makes bigger client expectations.

4. Supports rapid development

Agile can support fluid, rapid development whereby a developer can make product updates live onsite with the client, greatly making the time less for lengthy review and feedback cycles. Because the product is developed incrementally, both the client and developer can formed on a smaller spectrum of details, making sure these are right before moving on.

5. Increases functionality through frequent testing

In an ADDIE process, testing is usually left until the full version of the product is released, it a making lot more difficult to make correct issues. Alternatively, testing is a key factor of the Agile process, built into each incremental iteration. Because testing is done along the way, by the time the full product is released, is usually tight and there are fewer issues.

6. Promotes communication and collaboration

Agile is a collaborative process conducted through consultative sessions with key stakeholders, including project managers, subject matter experts, project sponsors and developers. The greater in communication Agile promotes leads to lesser misunderstandings and larger team cohesion. Since it greatly increases communication with the client, Agile is also a great approach for making clients back on board if they have previously experienced issues with fixed development processes.

2.1.3 PRINCIPLES

1. The highest priority of agile methodology is to satisfy the need of students and their families through early and healthy learning.

2. Agile welcomes changing requirements even in a late learning cycle for the benefits of the students and their families.

3. Agile delivers significant learning habitually with the preference to shorter time period.

4. Team members work together daily to create learning opportunities for each and every participants.

5. They build projects around motivated individuals and gave them the environment and believe in them to get the job done.

6. They recognise the most effective method of conveying information by face to face interaction.

7. Meaningful learning is the primary way of progress.

8. The best ideas and initiatives comes from self-organizing teams.

3 WHICH METHODS ARE USED ?

3.1SCRUM

Scrum is unquestionably the most used framework in agile methodology .Scrum is characterized by cycles or stages of development by the maximization of time required to develop a software product .It is successfully used in business related context But it is most prominently used in development of software product .

SCRUM is an agile framework of project management that originated in the early 1990's and was created by Ken Schwaber and Jeff Sutherland . Its name comes from the term scrummage (abbreviated scrum) in rugby, where "a group of attacking players from each team who come together with their heads down and arms joined, and push against each other, trying to take control of the ball" [3].

It is a method for restarting play.A simple "Definition of done " is used for validating the requirement of the clients.The dynamics of this methods allows the developers to organised their priorities that have not yet been completed and need more kind of attention. Transparency allows the projects to be followed by all the team member.

Teams are divided into different roles: (i) developers, carrying out the actual development tasks, (ii) a Product Owner which is in contact with the customer and is in charge of creating the project and Sprint Backlogs which are composed of User Stories and (iii) the Scrum Master who acts as a facilitator of the development team, helping to remove all the blockers that may arise and ensuring that the Scrum Framework is followed. Scrum Masters are also responsible for holding daily stand-up meetings and helping the developers understand the definition of done of a User Story.[4]

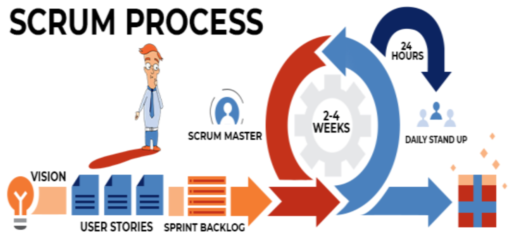


Fig1. Scrum lifecycle

.

3.2 EXTREME PROGRAMMING XP

The origin of eXtreme Programming (XP) started in 1996 from Kent Beck, one of the 17 Agile practitioners that signed the Agile Manifesto. At that time, Beck was handling the Chrysler Comprehensive Compensation System (C3) project in the Chrysler Corporation aiming at replacing several payroll applications with a single system. In 1999, he collected all the experience at Chrysler in a book called "eXtreme Programming Explained: Embrace Change" (Beck, 1999).[2]

. It is based around the idea of simplest thing that will work without putting to much weight on product view It emphasises values like communication simplicity feedback courage and respect .Moreover customer satisfaction over everything else . Teamwork is extremely important in Xp since if any problem is arises it is resolved by a group of users bringing them to promote conversations which help to breakdown the barrier of conversation .In XP software is tasted from day one to improve the development .It promote activities with strong testing component.It also contributes in lifting up team talent.

XP is based on 5 values:

- simplicity: simple solutions are considered as cheaper and faster to be implemented than more complex solutions;

- communication: the documentation is always after the direct communication. All the team members should intensively communicate with each other through personal dialogue, aiming at avoiding misunderstandings;
- feedback: the customer is interviewed using very short feedback loops. The development progress is frequently shown, and mistakes can be thus avoided. The feedback also comes from the tests;
- respect: every member of the team is valuable, even if it is just enthusiasm;
- courage: using the XP values requires a great deal of courage. It is important to tell the truth about progress and estimates. Also, XP does not allow fear of refactoring old code or throwing it away if needed for changes.[2]

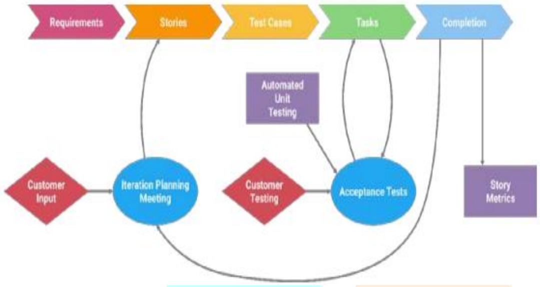


Fig2.EXtreme Programming lifecycle

In Fig.2, It shows the XP lifecycle. XP allows frequent releases in short development cycles, improving the productivity of the team, and at the end of which new customer requirements can be adopted. XP employs user stories and associates acceptance tests to them that need to be successfully passed before the stories can be considered as done. Moreover, the programmer is expected to write tests for the individual tasks that contribute to a story. Indeed, XP puts tests before code, and each piece of code is expected to be associated with a test, or it should not be integrated.[2]

6. Conclusion

The reported experiences explains that Agile can be effectively work conveniently , especially where active and project-based teaching can be applied. Not only can Agile be simulated in software engineering courses, but also it can be rigidly applied to learn other subjects. Applying Agile methodologies to learning and teaching changes from knowledge transfer to knowledge generated from rich collaboration and experience. Teachers who become facilitators, coaches, and inspirational servant leaders for students that are self-developed learners. The focus is not on fix plans, rather flexibility is confirms to take into account students' feedback and their different abilities, interests, difficulties, problems, and experiences, focusing on unlocking their hidden strengths and passions. The emphasis is on passing the highest value, in terms of both discipline specific teaching outcomes and soft skills such as organization, planning, collaboration, and teamwork. This review of the literature shows that there is a increasing interest in studying, but even more, forming Agile learning methodologies to make students to work together in an energetic, targeted, and effective manner. There is also an actively effort from researchers in formalizing the agile methodologies in the education context.

References

[1] Sharp, J. H. & Lang, G. (2018). Agile in Teaching and Learning: Conceptual Framework and Research Agenda. Journal of Information Systems Education, 29(2), 45-52

[2] Agile Methodologies in Education: A Review: Bringing Methodologies from Industry to the Classroom

[3] Implementation of Agile Methodologies in an Engineering Course Jana Pócsová *, Dagmar Bednárová, Gabriela Bogdanovská and Andrea Mojžišov

[4] A Systematic Review of the Use of Agile Methodologies in Education to Foster Sustainability Competencies Abigail López-Alcarria 1, Alberto Olivares-Vicente 2,* ,‡ and Fátima Poza-Vilches

[5] The Agile Teaching/Learning Methodology and its e-Learning Platform by Andy Hon Wai CHUN

[6] New Directions on Agile Methods: A Comparative Analysis by Pekka Abrahamssona , Juhani Warstab , Mikko T. Siponenb and Jussi Ronkainen