UDC [004.087.4:004.4]:378

APPLICATION OF AGILE PRINCIPLES IN THE ORGANIZATION OF SERVICE UNITS OF HIGHER EDUCATION INSTITUTIONS. PART 1

Olga Olshevska[1], Iryna Zinchenko[2], Oleksii Sakaliuk[3], Oksana Kozub[4], Valery Kolesnyk[5]

[1,2,3,4,5]Odesa National University of Technology
ORCID: https://orcid.org/0000-0002-4512-39151[1], https://orcid.org/0000-0002-5051-518X[3], https://orcid.org/0000-0002-0481-9203[4]
E-mail: olshevska.olga@gmail.com[1] , lib.onaft@gmail.com[2], sakaliuk.olexiy@gmail.com[3], ksyuha.muha@gmail.com[4], colesnik.valerya2016@gmail.com[5]

DOI: 10.15673/atbp.v15i3.2619

Abstract: *The publication deals with the application of Agile principles in organizing the work of service units of higher education. The first part of the publication deals with theoretical aspects of the methodology. Agile does not exist by itself, but is structured and carries out all processes with the so-called Agile Manifesto, consisting of 12 principles. It is found out, to what extent it will be correct to transfer the principles of Agile-manifesto to the system of higher educational institutions. The basic principles, which can guide the administration of the service units of the HEIs, are defined. Agile's definition is articulated in the Agile manifesto of software development, which emphasizes that people and communication take precedence over processes and tools. A functioning product is more important than extensive documentation, and collaborating with customers is more important than negotiating contractual terms. Additionally, being willing to adapt and change is more important than sticking to the original plan. Numerous project management approaches exist to apply the Agile philosophy, with Scrum, Extreme Programming, and Kanban being among the most prevalent. Among project managers, Scrum project management is widely used as one of the most popular Agile methodologies. The second part of the publication identifies the differences and similarities between Agile and Scrum. The third part of the publication examines the application of the methodology in practice. It is important to focus on constantly changing external and internal environment and take into account the feedback from customers and users. This encourages developers and engineers to experiment and look for new solutions without being constrained by rigid frameworks and standards.*

Анотація: *У публікації розглядається застосування принципів Agile в організації роботи сервісних підрозділів вищої школи. У першій частині видання розглядаються теоретичні аспекти методики. Agile не існує сам по собі, а структурований і виконує всі процеси за допомогою так званого Agile Manifesto, що складається з 12 принципів. З'ясовано, наскільки правильно буде перенести принципи Agile-маніфесту в систему вищих навчальних закладів. Визначено основні принципи, якими може керуватися адміністрація обслуговуючих підрозділів ВНЗ. Визначення Agile сформульоване в Agile-маніфесті розробки програмного забезпечення, де підкреслюється, що люди та комунікація мають пріоритет над процесами та інструментами. Функціонуючий продукт важливіший, ніж велика документація, а співпраця з клієнтами важливіша, ніж узгодження договірних умов. Крім того, бажання адаптуватися та змінюватись важливіше, ніж дотримуватися початкового плану. Існує багато підходів до управління проектами для застосування філософії Agile, серед яких Scrum, Extreme Programming і Kanban є одними з найпоширеніших. Серед менеджерів проектів управління проектами Scrum широко використовується як одна з найпопулярніших методологій Agile. Друга частина публікації визначає відмінності та схожість між Agile та Scrum. У третій частині видання розглядається застосування методики на практиці. Важливо орієнтуватися на постійно мінливе зовнішнє та внутрішнє середовище та враховувати відгуки клієнтів та користувачів. Це спонукає розробників та інженерів експериментувати та шукати нові рішення, не будучи обмеженими жорсткими рамками та стандартами.*

Keywords: Agile, Scrum, information technology, manifesto, project, management.
Ключові слова: Agile, Scrum, інформаційні технології, маніфест, проект, менеджмент

Introduction

Due to excessive planning and documentation of software development cycles, companies have lost some basic

provisions of the final work result, such as "usability" and "comfort" of the product. Executives and managers need to apply not only professional expertise, but also applications (web, mobile, and desktop) and planners to the day-to-day organization of work processes due to the intensification of requirements for modern companies and projects. No wonder that during the active development of information technology, various well-proven methodologies, in particular from the IT industry, began to be used to organize activities. Visualization of work processes helps in proper understanding of the planned changes and helps to implement them according to the plan. Every transition between the strata in the flow is monitored, measured and reported on. Actively managing the flow allows you to assess the positive and negative effects of system changes. Until the mechanism or process becomes obvious, it is often difficult or impossible to carry out a discussion of improvement. Without a clear understanding of how things work, any discussion of problems becomes emotional and subjective. With an obvious understanding, one can move on to more rational, empirical, and objective discussions of problems. Improve the process on the project by collaborating, developing experimentally, using models and scientific methods. When teams have a common understanding of theories about work, process, and risk, they are more likely to be able to develop a common understanding of problems and propose improvements that will result in consensus. These points are important to both software developers and customers who want the best results. This paradox contributed to the genesis of the idea to create the Agile Manifesto.

The Agile Manifesto has its origins in a ski resort in Uttah, where 17 representatives of different disciplines met to socialize and find common ground. This meeting was held February 11-13, 2001, with representatives from SCRUM, Adaptive Software Development, Extreme Programming, DSDM, and others present. Subsequently the text of the manifesto became available in more than 50 different languages and signed by hundreds of independent fans.

The main topic of conversation for the Agile Alliance (that's how the like-minded people called themselves) was to create an alternative to heavyweight, documentation-based software development processes. Thus, the Agile Software Development Manifesto was developed.

Agile Manifesto is a document that describes the basic principles on which agile development is based (Manifest hnuchkoyi rozrobki, 2022).

That is, Agile is a system of agile project management, which focuses the team on the needs and goals of customers, while simplifying all the processes and offering work in short cycles. Agile management approach implies that it is not necessary to specify all the parameters that are not subject to change during the course of the project in order to describe the final product of the project. Without going into technical details, this is not a methodology, but a collective name for different management techniques and approaches. In other words, Agile Manifesto has formed a lightweight alternative to template difficult methodologies. On the basis of this system, the popular methods of Scrum, Kanban and others were developed.

The ideas in the described manifesto are applicable in almost all areas. They have the complexity of calculating the final amount of work. Frequent changes and improvements in the final product are possible, which leads to a changing price value of the project. This is why Agile is not suitable for managing those precise projects where a clear estimate is made for the entire job.

The CEO of Scrum.org, Dave West said that Agile principles are not new, they have existed before, but now they are applied differently. These principles of the scientific method have been used since Galileo and Archimedes.

Perhaps one of the best achievements of the Manifesto is the systematization of a way of thinking that has not been used before for software development, which is a great achievement.

It is likely that the Agile environment has an overabundance of methodologies that promise to turn Agile principles into a practical reality. But in the changing world of methodologies, there are no explicitly new techniques yet.

That's why the Manifesto is very relevant today.

As something as culturally significant as the Manifesto emerges, it can be reinterpreted, but none of these interpretations compares with the original. It is also important to find its application in relation to oneself, one's team or organization.

What matters in this situation is not the "tablets of commandments" themselves, which everyone could agree on, but whether a group of people (from the team, to the organization as a whole) can apply the ideas of these "tablets" to a specific situation without missing its true meaning. If done right, unlimited abilities open up.

Theoretical aspects of the formation of Agile principles

Agile is needed to adjust to constantly changing requirements. There are traditional approaches like Waterfall, which dictate total planning. Following them, an employee simply cannot do anything that is not in the plan. However, in any project, whether in an educational institution, government agency, marketing company, or software development, requirements often change or additional opportunities arise. And this is especially true when it comes to large projects, because large projects are often unpredictable. As a result, you either have to ignore new factors that are not good, or squeeze a plan that plunges the work into chaos altogether. With the right approach to planning, the team will be able to anticipate all unknown factors in advance if they build the likelihood of it into the workflow. Practice shows that many of the best ideas come to mind only when the project goes too far. For such tasks, the Agile methodology often comes to the rescue. Prioritization is the most important part of Agile management methods. If you are a project manager, you need to make sure that your team is focused on the most important tasks and achieving the most meaningful result. This aspect

will ensure that the project achieves its goals, i.e. you are guaranteed to achieve the corporate or business goals. Tasks (user stories) in Agile are delivered either continuously or in small cycles called sprints. The general algorithm of the methodology can be represented as follows: "Requirements-Plan-Do-Review-Repeat". Based on the requirements of the project, you should make a list of what should happen. If you forget something, do not worry, because it can be added later.

The next step is to evaluate each stage of development, produced by hours or by story points. Story points are relative estimates, which are put in comparison with other stories. It should be taken into account that there is a possibility of inaccurate results, that is, you will not be able to get a clear idea of exactly how much time it will take to implement the project.

From a management and project management perspective, it is important to prioritize tasks, putting the most important ones before the queue starts. In this environment, things are usually very dynamic and constantly changing, so don't forget to check priorities more often. Kanban responds well to this frequency. But Scrum is based mostly on fixed cycles, i.e. sprints that last 2 weeks. Scrum and Kanban can be called the two main frameworks for Agile.

Like any methodology Agile does not exist by itself, but is structured and implements all the processes with the so-called Agile manifesto, consisting of 12 principles.

1. Customer satisfaction with the early delivery of the software. In the agile method, we place a high priority on customer satisfaction. The goal here is to provide the customer with an early and continuous delivery of valuable software. There will be periodic interaction with the customer about the ongoing software development cycle.

2. Accept a change request, even at a later stage of development. When the team is working on final deliverables and a new client request or change appears on the current development, the agile program helps us use that request and accept the changes invited by the client for the current development. This helps the client achieve the goal more easily and the team adapt to those changes. It's hard when we have to adapt to changes in software development, but changes can be good if we can respond to them quickly.

3. Delivering software frequently. When we can deliver a product periodically, it helps to identify necessary changes and misunderstandings with the customer. Rather than offering development in which the developer cannot accept any changes requested by the customer. In this method, we can give step-by-step development and entertain any changes requested by the client. Let's move on to the next agile principle.

4. Daily collaboration between business people and developers. Daily interaction with business people and developers helps developers work in the right direction and adapt to any changes the customer asks for. And the customer will be aware of the team's progress. This makes the work smooth and transparent.

5. Projects are built around motivated people, who should be trusted? When people who are motivated to work are given an environment and support, they should complete the task. Then both the client and the developers will benefit from getting the task done. Trusting the developer and having support around them will help them work comfortably. There will be no need to micromanage people.

6. Personal conversation is the best form of communication. In a face-to-face conversation, we can convey information effectively and as efficiently as possible. Although developers will sometimes be in different locations, an effort should be made to communicate face-to-face if possible.

7. Project work, not planning. Working on the software should be the primary goal. When working on it, developers will only have to focus on development. But if you concentrate on executing the plan, the developer will be distracted by activities such as documentation, etc.

8. Sustained development that can maintain a steady pace. Agile helps developers maintain a steady pace during development. This is called constant speed, and we can measure the potential of the team. So going forward, we can get to work based on the speed of the team. Let's move on to the next agile principle.

9. Constant attention to excellence and better design. The ability to pay constant attention to development is a sign that the team is trying hard to satisfy customers. It is the ability to give the best to the owners.

10. Maximizing performance efficiency. Agile is about reducing costs that don't provide value. Being simple rather than complicating unnecessary things is a specialty of Agile.

11. the best design, the best solution and architecture arose from self-organization. A cross-functional team can be called a self-organizing team. These people are the best in the business; they will help the customer achieve the goal with the best advice, the best work, and the best solution.

12. Regular meetings to discuss team improvement. This is a major part of an agile practice team. Once the project is completed, there will be a review of the project. This will help the next client approach differently. Another aspect will be that at the end of each scrum there will be an event called RETROSPECTIVE to see what needs to be improved and what the team has learned about itself from it.

Since the methodology has existed for quite a long time, it has divided into separate methodologies or new components became subspecies of the main.

Agile Modeling is a set of concepts, principles, and techniques (practices) that make modeling and documentation in software development projects quick and easy. Does not include detailed instructions on how to design, does not contain descriptions of how to build diagrams in UML. Main goals: effective modeling and documentation; but does not include

programming and testing, does not include project management, deployment and system maintenance. It does, however, include verification of the model by code.

Agile Unified Process is a simplified version of IBM Rational Unified Process (RUP) developed by Scott Ambler, describing a simple and straightforward approximation (model) for creating software for business applications.

Agile Data Method is a group of iterative software development methods in which requirements and solutions are achieved through the collaboration of different cross-functional teams.

DSDM is based on the concept of Rapid Application Development (RAD). It is an iterative and incremental approach, which emphasizes long-term involvement of the user/consumer in the process.

Feature driven development is a function-oriented technology. The notion of a feature or property of the system used in FDD is quite close to the notion of a use case used in RUP, the essential difference being an additional restriction: "it is required that every feature be capable of implementation within a maximum period of two weeks". That is, if the use case is small enough, it can be considered a function. If it is large, however, it must be broken down into several relatively independent functions.

Getting Real is an iterative approach without functional specifications used for web applications. In this method, the program interface is developed first and then the functional part.

OpenUP is an iterative-incremental method of software development. It is positioned as a light and flexible version of RUP. OpenUP divides the project lifecycle into four phases: initial phase, refinement phase, design phase, and transfer phase. The project lifecycle ensures that stakeholders and team members are provided with points of insight and decision-making throughout the project. This allows for effective monitoring and decision making on the acceptability of the results. The project plan defines the project lifecycle, and the end result is the final application.

Scrum establishes the rules for managing the development process and allows existing coding practices to be used by adjusting requirements or making tactical changes. Using this methodology allows you to identify and eliminate deviations from the desired outcome early in the development of a software product.

Aglie and Scrum methodologies, similarities and differences

Agile is a project methodology or framework that employs an iterative method for accomplishing projects (as shown in Figure 1).

According to the Project Management Institute, the primary aim of the Agile approach is to achieve a measurable return on investment by delivering product features in a defined iterative manner, particularly during the early stages.

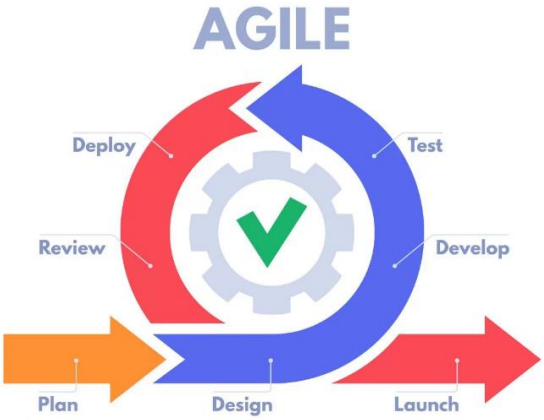

Fig.1. Iterative approach of Agile methodology (Metodologiya CI/CD, 2021)

Given the iterative nature of Agile methodologies, consistent client participation is imperative to ensure that expectations are in sync and to enable project managers to adjust to changes as the process unfolds.

Joseph Griffin, who serves as an associate professor in Northeastern's master's program for project management, asserts that "Agile is, above all else, a project management methodology that centers on specific principles and values. Consider Agile as a standard for our approach to project-related activities." (Stobierski, T., 2021).

One characteristic that sets Agile apart is its core principles and values, which can be employed across various methodologies that are specific to a given context.

Joseph Griffin explains, "Suppose you adopt the Agile approach for project management. In that case, you should aim to maintain consistent communication with the client and/or end-users, seek a flexible scope that can evolve based on end-user feedback, and take an iterative approach to defining scope." Agile consists of only four core values that capture its essence, which are (Merehead, 2022):

- Prioritizing direct human communication over tools is of greater significance.
- Giving precedence to an operational product over a comprehensive specification is essential.
- Giving more importance to direct communication with customers as opposed to making changes to contracts is crucial.

- Sticking to the original plan is nonsensical since everyone must be prepared for alterations in events and requirements..

Agile has slightly more principles, 12. You can read them on the official website. Here is a list of what really makes Agile so popular. The first and most important is that changes can happen on a daily basis, so you have to be dynamic and ready to react quickly. That said, minimal changes involve documentation. It is worth noting right away that, given this principle, Agile for the most part is not suitable for projects with tight deadlines and budgets. Second, strong communication increases the quality of work and makes the task predictable. It is recommended to hold daily rallies and report on results, problems. They discipline the work and allow you to clearly plan your work schedule.

Numerous project management methodologies are employed to execute the Agile philosophy, among which Kanban, Extreme Programming, and Scrum are some of the most frequently used.

Among project managers, Scrum project management is a highly favored Agile methodology.

Although Agile is a philosophy or mindset, Scrum is a distinct methodology for project management that defines how to determine, assign, perform, and complete the work." Scrum offers a systematic approach to defining the work, designating the individuals responsible for it, specifying how it will be done, and determining when it will be finished.

Scrum project management involves a project team that comprises the project manager, the product owner, the Scrum master, and other members with cross-functional expertise. The product owner's role is to optimize the product's value, whereas the Scrum master is responsible for ensuring the project team adheres to the Scrum methodology.

Short phases, or sprints, are a defining feature of the Scrum methodology (as shown in Figure 2), wherein project work is carried out. During sprint planning, the project team identifies a small portion of the scope that must be accomplished in the upcoming sprint, which usually spans two to four weeks.

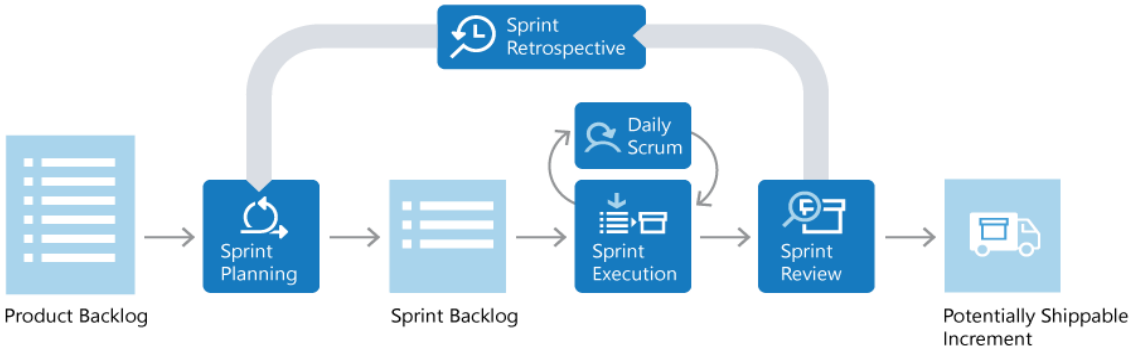

Fig.2. Scrum life cycle (Mijacobs, 2021)

Upon completion of the sprint, the resulting work is expected to be in a state of readiness for delivery to the client. The sprint concludes with a review and retrospective analysis of the sprint's performance, encompassing the lessons learned. This process is repeated throughout the project life cycle until the complete scope is delivered.

In several aspects, this methodology is similar to traditional project management. However, one major difference is that the "shippable" components of the project are developed progressively, rather than delivering everything at the project's completion. This approach enables the client to perceive the project's value throughout the process, rather than having to wait until the project's conclusion to witness the outcomes.

Initially, it may be challenging to distinguish between Agile and Scrum because they share common elements such as iterative processes, regular customer engagement, and collaborative decision-making. Nonetheless, the crucial distinction between Agile and Scrum is that Agile is a project management philosophy grounded on a fundamental set of principles, whereas Scrum is a specific Agile methodology aimed at enabling a project.

Agile and Scrum also differ in several other significant ways.

Differences:

- While Scrum is a type of Agile methodology, Agile itself is a philosophical approach to project management.
- Unlike Agile, which delivers everything at the end of the project, Scrum adopts shorter sprints and smaller deliverables as a way to manage the project in a more iterative and incremental manner.
- Agile typically involves members from different cross-functional teams, whereas the Scrum project team includes specific roles, such as Scrum master and product owner.

It's crucial to keep in mind that although Scrum is an Agile methodology, not all Agile approaches involve Scrum. There are several methodologies that take an Agile approach to project management. While both Agile and Scrum aim to enhance process management performance, they are distinct concepts. Agile represents a set of values, whereas Scrum is a specific methodology. By applying Agile values, you can increase flexibility to the Scrum methodology.

Applying the Agile methodology case study in practice

Kanban boards are currently one of the most popular and requested ways to organize the work process. The first attempts to use Kanban mechanisms began in the 1950s on the production lines of Toyota Corporation. Successful attempt of application promoted transfer of these mechanisms to activity of both small and large companies, representatives of

business. Limitless flexibility of the practice and its potential for self-organization of staff allowed to achieve efficiency where other approaches did not work.

No wonder that with the active development of information technology sphere Kanban boards were used to organize activities. The principle of organizing activities is incredibly simple for each process or task created cards. They contain a due date, a description and the name of the performer. Such cards are attached to the column among which may be: Beclog - tasks to be completed, tasks currently under development, tasks completed but not yet handed over to testers, tasks ready to be handed over to the testing department passing project management (PM) tasks, completed tasks.

The above-mentioned order is not mandatory, and depending on the specifics of the project improvised columns may be added. Such columns may include specify (to clarify parameters) and execute (to take up work).

Conclusion

The theoretical aspects outlined in this paper are the next stage in the formation of a methodology and algorithmization of the transition to new standards of higher education institutions, in particular scientific libraries, as components of such organizations. One of the stages and methods has already been published in (Zinchenko, 2021). The information in this paper is a systematization of existing open data and allows to form models of the work of service structural units. The second part of the work will analyze the practical stock of this research, which is being tested in real conditions.

References

1. *Manifest hnuchkoyi rozrobki* (2022) *Wikipedia*. Wikimedia Foundation. Available at: https://uk.wikipedia.org/wiki/Маніфест_гнучкої_розробки (Accessed: August 29, 2022).
2. Stobierski, T. (2021) *Agile vs. Scrum: What's the difference?, Northeastern University Graduate Programs*. Available at: https://www.northeastern.edu/graduate/blog/agile-vs-scrum/ (Accessed: March 15, 2022).
3. *Metodologiya CI/CD: kak razvivatsya* (2021) *ITG BY*. Available at: https://itglobal.com/ru-by/company/blog/methodology-ci-cd/ (Accessed: February 19, 2022).
4. *What is the difference between Scrum and agile methodology* (2021) *Merehead*. Available at: https://merehead.com/blog/scrum-vs-agile-difference/ (Accessed: March 18, 2022).
5. Mijacobs (21AD) *What is Scrum? - azure DevOps*, *Azure DevOps | Microsoft Learn*. Available at: https://docs.microsoft.com/en-us/devops/plan/what-is-scrum (Accessed: March 21, 2021).
6. ZINCHENKO, I. *et al.* (2021) "Methodology for the implementation of web standards to the activities of academic libraries," *Revista Română de Informatică și Automatică*, 31(4), pp. 45–54. DOI: https://doi.org/10.33436/v31i4y202104.

Отримана в редакції 05.08.2023. Прийнята до друку 05.09.2023. Received 05 Augast 2023. Approved 15 September 2023. Available in Internet 12 September 2023.

UDC 621.382

METHOD OF INCREASING EFFICIENCY, MEAN OF CONTROL AND TOOL OF ANALYSIS OF THE USE ENERGY OF POWER SOURCE PULSE LASER RANGE FINDER

[1]Pavlo Bratiuk, [2]Leonid Ozirkovskyy
[1,2]Lviv Polytechnic National University, Lviv, Ukraine
ORCID: [2]https://orcid.org/0000-0003-0012-2908
E-mail: [1]pavlo.bratiuk.tr.2019@lpnu.ua, [2]leonid.d.ozirkovskyi@lpnu.ua

DOI: 10.15673/atbp.v15i3.2620

Abstract. *The latest results of a series researches of the revealed paradox of energy balance disturb and its manifestations in electrical circuits are presented in order to increase the efficiency of the conversion of energy voltage direct current into pulsed energy of other forms, on the example of improving the use energy of the power source of pulsed solid-state laser range finder. It is show, that the efficiency of simultaneous conversion in a load with active linear or nonlinear resistance of the energy, which accumulated in the inductive and capacitive storage, can reach the efficiency of its accumulation. For this the first time offered the aperiodic mode of the transient process in the oscillating circuit*