Methodology of Building Agile-Education Processes in Higher Education Institutions

Viktoriia Mironova[a], Mykola Pyroh[a] and Iryna Harko[a,b]

[a] *Taras Shevchenko National University of Kyiv, Bohdana Havrylyshyna St, 24, Kyiv, 04116, Ukraine*
[b] *National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Politekhnichna St, 41, Academic building № 18, Kyiv, 03056, Ukraine*

Abstract

In this paper the technology of using the popular flexible software development methodology – agile – is considered for the development and construction of courses and semantic links of higher education disciplines for training specialists in the field of information technology. The authors analyze the basic principles of the Agile-Manifest and their application for the formation of the curriculum program and the students' training disciplines. The presented article considers the result the flexible agile software development methodology implementation into the process of higher education students teaching. We present approaches to the educational process modernization, educational and methodological support development and approaches to teaching in high school on the example of the discipline "Web technologies and site programming" for students in specialty 122 Computer Science in educational program "Applied Programming". The article describes the results of several years gradual adaptation of Agile technology to the learning process, its introduction into the learning process and the results of the developed methodology approbation. The theoretical bases and advantages of Agile-methodology over the classical scheme of training, which we define as "direct" or "cascade", are considered. In order to effectively study the impact of flexible methodologies on the educational process, approaches to the formation of experimental and control academic students groups are presented. The dynamics of laboratory work timeliness and comparison of the academic groups final results are given. The influence of Agile-methodology on the academic groups success is considered and a statistical comparison with the classical approach to teaching is made. Also, we determined the directions of further research on the topic of Agile-methodology implementation in the educational process taking into account the urgency of the education transformation issue in Ukraine and professional training of specialists in the field of information technology.

Keywords[1]

agile-methodology, scrum, education, Information Technology, training of specialists, professional training.

1. Introduction

Training of specialists in the field of information technologies requires the formation of educational programs and courses using modern technologies, programming languages, libraries and frameworks. However, the rapid growth and changing trends in the IT sector makes it impossible to create such a program even for four years of bachelor's degree [1]. That is, even if at the beginning of the educational cycle, the program was designed taking into account the most relevant approaches and topics, at the time of graduation, it becomes obsolete. To understand the need to adapt to such changes, it is worth estimating the frequency of programming languages standards updates and

IT&I-2020: Information Technology and Interactions, 2-3 December, 2020, Kyiv, Ukraine
EMAIL: vicky.mironova@gmail.com (V. Mironova); mykola.pyroh@yahoo.com (M. Pyroh); garko.iryna@gmail.com (I. Harko)
ORCID: 0000-0002-0878-0967 (V. Mironova); 0000-0003-2588-6066 (M. Pyroh); 0000-0003-0671-6336 (I. Harko)

releases, which are 1-3 years for Java, 3 years for C ++ and from six months to 3 years in C#. And even more, programming libraries and frameworks are released almost each month.

To help higher education institutions to adopt their materials and educational programs, we propose the introduction of an adaptive methodology based on Agile methodology [2]. Such approach should promote the development of future IT professionals' skills, help student to quickly master new material and be flexible at the IT industry fast-paced market [3].

2. Principles of Agile-education processes

Agile management – an iterative method of planning and managing projects and processes [2]. Agile management allocates short product development cycles, providing additional updates depending on changing customer needs. In 2001, Kent Beck, Martin Fowler, Dave Thomas, and 14 other leading software practitioners published the Agile Manifesto, which was a brief description of some of the easy software development technique principles. Agile-Manifesto describes the principles of "flexible" methodology of software development and implementation: priority of customer interests, openness to change, system operation at all times, mobile joint teams "customer-developer-user" with a high level of communication. Flexibility and adaptability are that the Agile client outlines the task, sets the direction, and the team thinks how to solve this problem. The team can change the path to the goal and implement the client's plans in any way convenient for them. According to Agile's principles, the client's business goals, his wishes are more important than the initial conditions and documentation. Another difference between Agile and traditional control schemes is the process algorithm. The traditional management model involves a gradual movement from the stage of analysis to the stage of production. Agile consists of cyclically repeating sprints (Sprint), in other words a number of stages of continuous product implementation and adjustment (fig 1.).

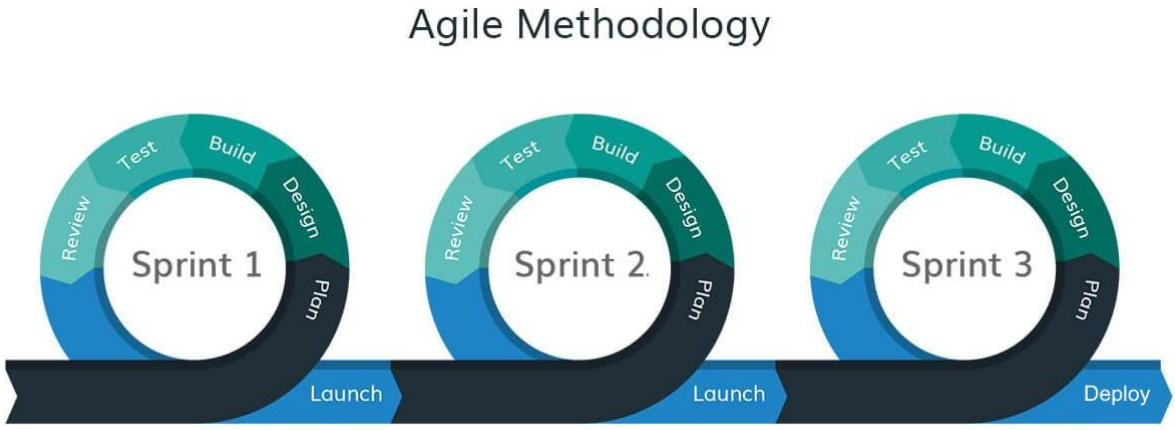

Figure 1: Agile-methodology Scheme

If we consider the process of training a specialist in higher education in terms of project management, and the specialist – as a unique product, the training can be approached from the standpoint of project management and apply to this process all modern and successful project management methodologies. Due to the success of the Agile approach in industry, software design, marketing and business, foreign experts in the field of education have tried to apply similar principles to the field of education, thus forming the Agile Schools Manifesto:

- people and interaction are more important than processes and tools;
- significant knowledge is more important than learning;
- cooperation with stakeholders is more important than complex negotiations;
- readiness for change is more important than following the original plan.

12 agile principles adapted to the school were also formed. The same principles are easy to implement in the field of higher education and training:

1. Education results through continuous learning of the discipline

2. Divide large chunks of study and materials into smaller and achievable tasks for quicker completion and easier mastering the proper skills
3. Adhere to the decided timeframe for the delivery of a completed tasks and homeworks
4. Students and teacher must frequently collaborate to ensure that the education is going in the correct direction
5. Create a supportive environment to motivate students and encouraging them to get the job done
6. Prefer face-to-face communication over other methods
7. Knowledge of theoretical and practical materials is the primary measure of progress
8. Try to maintain a constant pace of learning
9. Maintain the quality of study by paying attention to technical details
10. Maintain simplicity
11. Promote self-organization in the team
12. Regularly reflect on your performance for continuous improvement

EduScrum's [4] theory and methodology are based on the empirical theory of process management or empiricism. Empiricism states that knowledge comes from experience and decision-making based on knowledge and information. eduScrum uses an iterative, consistent approach to optimizing the achievement of learning objectives and risk control. Three bases support each implementation of empirical process control: transparency, verification and adaptation (fig 2).

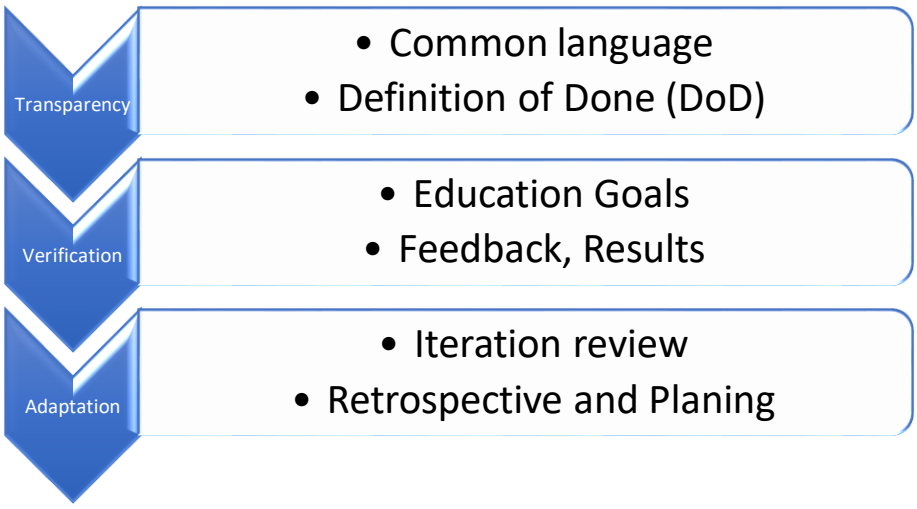

Figure 2: The Education Empirical Methodology Basis

Transparency ensures the visibility of essential aspects of the process for those responsible for the outcome. Transparency requires defining these aspects by a common standard, so observers have a common understanding of what is seen. All participants in the process must communicate in one language and understand all definitions.

Those who do the work and those who accept the work product must have a common definition of "Ready"(DOR – Definition of Ready) and "Done" (DoD – Definition of Done). EduScrum users should frequently check eduScrum artifacts and progress toward learning objectives for unwanted deviations. These inspections are most effective in the case of diligent and systematic performance by both teachers and students in the classroom. If the student (or teacher) determines that one or more aspects of the process threaten to deviate from the defined boundaries and/or the results will be unacceptable, the planning or approach should be adjusted as soon as possible to minimize further deviations. The tools of such adaptation are planning, review of sprints and demonstration of results, personal reflection.

Considering the rapid changes in the IT market and trends in demand for programming languages we propose new Agile methodology of building educational programs in higher education institutions. This methodology is built in concept of branched scheme based on the principles of flexible methodology (fig. 3.) of software development "must do, should do, could do" [5].

The difference of this approach comparing to classical cascade educational scheme is the abandonment of the usual transition from the previous to the next work by dividing each topic into differential tasks. This allow students to perform tasks clearly aware of the "degree of immersion" in the topic. Thus, the part "must be done" is a direct task according to the material covered on a particular topic, part "should be done" – a complex level of the task that provides additional conditions to the functionality of the program within the passed topic of direct work and previous work, the last part "can be done" – a task aimed at the active use of hours set aside for independent work, this includes encouraging students to explore ways to improve the program code, its refactoring and work towards clean code.

Figure 3: MoSCoW prioritization technique

According to the "Regulations on the organization of the educational process" [6] for independent work is allocated from 50 to 67 percent of the time allotted for the study of the discipline.

To effectively use this time, in order to provide quality training for future professionals, we have developed comprehensive tasks aimed not only to deepen students' knowledge, but also to give an understanding of the self-organization of groups during the performance of complex work.

According to Agile Education, groups of 3 to 5 people are formed to perform a complex task, which distribute roles in the group and build relationships according to the eduScrum methodology. In this context, the teacher has the role of consultant-coordinator, who determines the requirements for the functionality of integrated work, guides student groups in their search for information and helps to develop skills of interaction, mutual assistance and mutual learning of group members.

Table 1 presents a fragment of the work program of the discipline "Web technologies and site programming". According to its structure, according to the developed methodology, the plan of work on a complex task is divided into stages (according to eduScrum – sprints), before the first stage students receive tasks and are divided into groups that will work on their complex works. According to the methodology, special meetings are provided to coordinate their actions in the group, where students discuss their work in the group or identify problematic stages.

According to the Agile methodology, this thematic plan can be presented in the form of product requirements and sprint goals (Table 2). After developing the plan, students begin work on the goals and requirements of the product, which is to break the complex work into small tasks, which are distributed among group members. Students determine the degree of "readiness" of each task, ie what its state is considered to meet the goals of the group and can be defined as satisfactory for integration into the product, and determine the priority of tasks according to the above scheme. The results of such work are usually demonstrated in the form of an Agile board (fig. 4). During the sprint, each of the tasks performed is moved on the board in the column "In progress", when the task corresponds to DoD, it is moved to the column "Done". According to our plan, the teacher coordinates the activities of student groups during retrospectives, which summarize the work of the group during the stage. In retrospect, students report on the work done, closed tasks, uncompleted tasks, carry over to the next stage tasks from the category "must be done", if they were not completed during the current stage,

and answer the theoretical questions. Thus, the presentation of the material encourages students to make efforts to self-development in the influence of external, subjective and objective factors such as:

- changes in the academic schedule.
- changes in teaching methods and tools.
- the need to turn to another technology.
- the need to review the functionality of the product.
- unpredictable changes in the development team.

Table 1.
Fragment of the table of the discipline "Web technologies and site programming" thematic plan

Topic Number and Title	Hours		
	Lectures	Practical works	Home works
Basics of HTML/CSS. Information content and structure of the Web-site	4	6	10
Adaptive and Responsive Web Design. Media Queries	2	2	10

Table 2.
Objectives of the first sprint of the discipline "Web technologies and site programming" according to the Agile methodology

Learning objective	Key results
Master the basic principles of web-site creation and html structure	• We know the basic principles and features of HTML. • We can create a simple web-page using HTML/CSS. • Understand the differences of block and inline tags; • We can create semantic web page and validate it according to W3C standard;
Master the methods of creating adaptive web-pages	• We know the structure and methods of using the media queries for web-page creation • We understand the purpose of using viewport and its' characteristics • We know the difference between relative and absolute values and positioning • We can create a web-site that will be working on any devices and screens • We understand the differences between adaptive and responsive web-design technologies

Retrospective is an opportunity for the teacher to understand the needs of each student and student teams, to adapt learning materials to individual perception, to adapt the goals and objectives of subsequent sprints given the speed and quality of knowledge of the material by teams and individual students. In addition, the goals of the sprint and its objectives may change under the influence of external factors: time changes and changes in the schedule; changes in discipline requirements due to changes in the labor market or the irrelevance of technology; changes in teaching methods and tools, etc. The above processes give us a picture of the understanding of learning, which corresponds to the classic scheme of eduScrum [8].

3. Agile-education methodology implementation

Proposed agile-education methodology was used to develop a more adaptive work curriculum of the discipline "Web technologies and site programming" for students majoring in 122 Computer Science educational program "Applied Programming" [7, 8]. Implementing the principles of Agile

Education and own developments in the educational process, a method of teaching this discipline was developed, which without violating the structure and educational norms at the Taras Shevchenko National University of Kyiv [6]. Proposed approach can make the learning process to be more adaptive to changing requirements and each individual group speed of learning and mastering practical skills. This approach to complex tasks was tested by the authors in the 2018/2019 and 2019/2020 academic years. This approach contributes to the interpretation of the usual learning process as a long-term project using Agile-methodology [9]. However, after two years of testing the methodology, the authors decided to adapt this approach to laboratory work [7, 8].

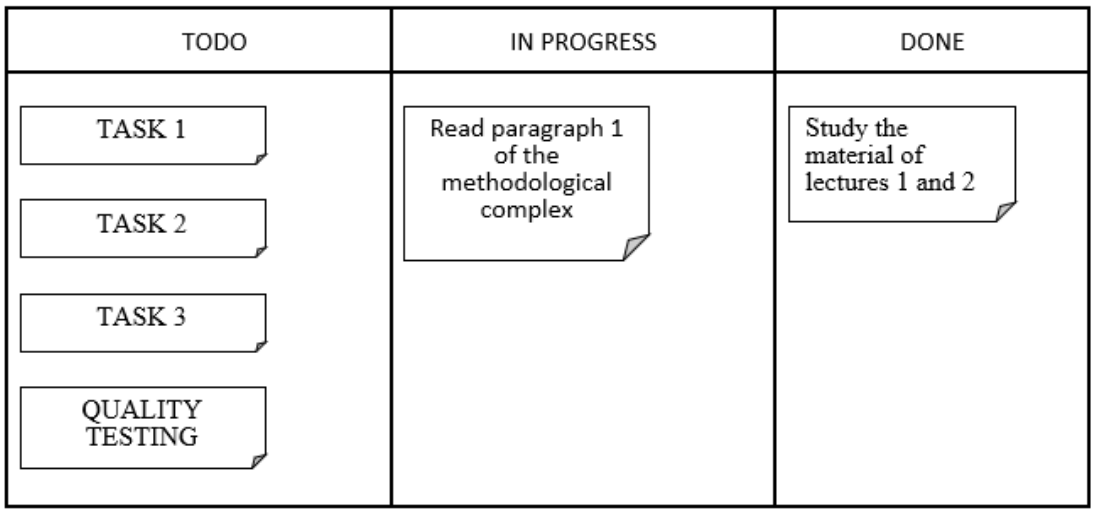

Figure 4: A fragment of the Agile-Sprint discipline board

At the beginning of the article we outlined a certain adaptation of a set of laboratory works to the Agile methodology. Therefore, in the 2019/2020 academic year, it was decided to adapt the methodology of group work for laboratory tasks. In order to assess the effectiveness of the methodology in the full cycle of studying the discipline "Web technologies and site programming" in the presence of two groups (50 students), divided into four subgroups (10-15 students), in the current academic year it was decided to teach two subgroups Agile methodology (experimental groups: EG1, EG2), and two – according to established methodologies (control groups: CG1, CG2) (Table 3).

The discipline "Web technologies and site programming" is studied in the third year in the second semester. The experiment was conducted in the second semester of the 2019/2020 academic year while studying the basics of programming web-sites. That is why, in order to ensure the normal distribution of students into groups according to the level of initial training, the points of their average marks gained during education on the faculty, were used as data.

Table 3.
Distribution of students by groups

	Control group (CG)	Experimental group (EG)	Total
First Group	10	11	21
Second Group	15	14	29
Total	25	25	50

In Agile subgroups (EG1 and EG2) students were divided into groups of 3-4 people working on tasks together. These students received complex tasks, taking into account the teamwork and defended them together, including a package of theoretical questions. In subgroups KG1 and KG2 laboratory work was performed according to the classical "cascade" scheme. According to the results of the exam, the average score of students in all subgroups was calculated (Table 4).

As you can see, students in the control groups who studied in the traditional format, on average, received 14.18 points less than students in the experimental subgroups. It is also worth noting that the average score of students in the second group is 0.93 higher than that of students in the first group.

Similar dynamics is observed in the corresponding control and experimental subgroups: the average score in the experimental subgroup of the first group (EG1) exceeds the average score in the control subgroup of the same group (CG1) by 13.58 points, while for subgroups EG2 and CG2 this difference is 14.78 points. It is also worth noting that students in the second group generally study better and have higher scores. Figures 5,6 show the distribution of final points in the discipline for students of all subgroups.

Table 4.
Average score of students by groups

	Control group (CG)	Experimental group (EG1)	Total
First Group	69.6	83.18	76.39
Second Group	69.93	84.71	77.32
Total	69.77	83.95	76.86

Studying the graphs presented in Figures 2 and 4, we can see that for the experimental groups the distribution of final scores in the discipline is significantly shifted to the right, in the subgroup EG2 the maximum occurs in the range of 90-100 points. In the control subgroups CG1 and CG2 the percentage of students who received a final grade of "excellent" (90 points and above) – 10% and 13.33% respectively, in the experimental subgroups EG1 and EG2 this figure is higher – 36.36% and 35.71%, respectively. At the same time, in the experimental subgroups there is no student with a score below 60, and in the control subgroups the percentage of such students is: CG1 – 20% and CG2 – 13.33%.

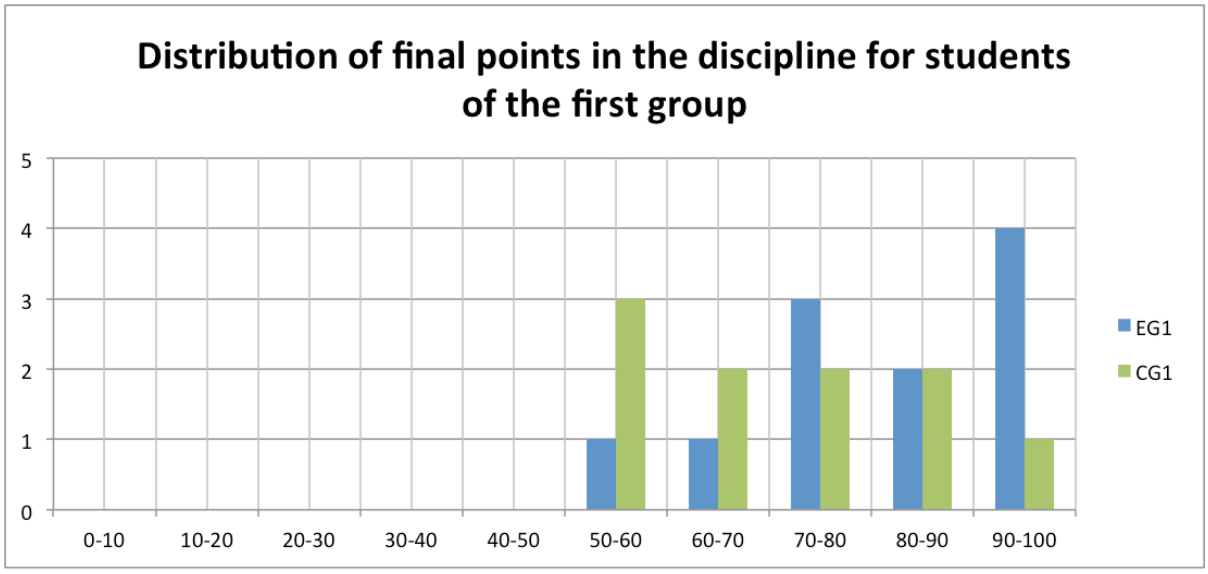

Figure 5: Distribution of final points in the discipline for students of the first group by subgroups

For a more detailed analysis of the effectiveness of the implementation of Agile-methodology and to identify the features of students' learning in different learning technologies, an equally important indicator was considered – the timeliness of students' laboratory work. A positive feature of the use of Agile-methodology in experimental subgroups is that students within microgroups (3-4 people) had a motivational influence on each other, which prompted them to perform all tasks with deadlines, which was reflected in the dynamics of timeliness of their laboratory works (Fig. 7).

Thus, we see that the percentage of students who performed laboratory work on time in the experimental subgroups ranges from 79-90%, while this figure for the control subgroups remained at 54-62%. In addition, in the control subgroups there was a violation of the deadlines for laboratory work for 3 or more weeks in 19-20% of students, in contrast to 2-5% in the experimental subgroups (Table 5). According to the results of the study, we can state that Agile-groups submitted materials with better compliance with deadlines, and group work encouraged students to help each other in the studied theories, which was a condition for successful delivery of laboratory work.

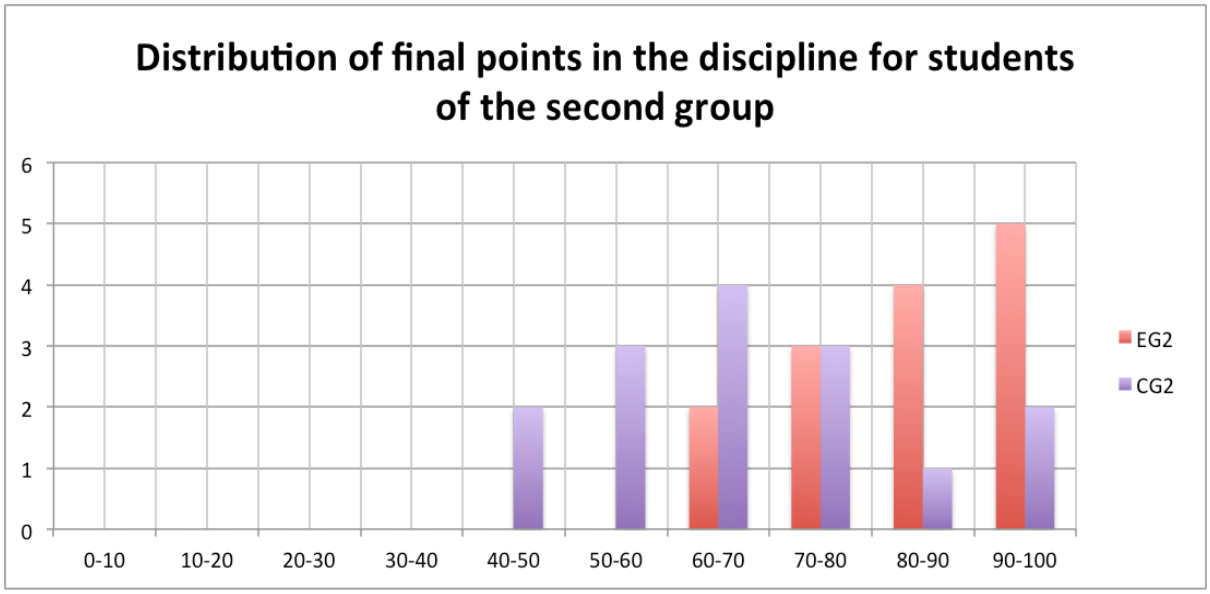

Figure 6: Distribution of final points in the discipline for students of the second group by subgroups

Table 5.
Dynamics of laboratory work timeliness by groups

	CG1	EG1	CG2	EG2
Timely execution	54%	79%	62%	90%
Deadline violation – 1 week	14%	12%	11%	6%
Deadline violation – 2 weeks	11%	5%	9%	3%
Deadline violation – 3 weeks	12%	1%	9%	1%
Deadline violation – more than 3 weeks	8%	4%	10%	1%

4. Conclusions

The introduction of flexible methodologies in the educational process optimizes approaches in the training of future IT professionals. The implementations of the flexible methodology of software development – Agile – in the process of development of educational and methodical support and the proposed approaches in teaching in higher education contribute not only to better understanding and mastering by students of theoretical and practical educational material.

This approach also helps students understand the production processes in IT projects that are set up according to the Agile methodology, set themselves to work within Scrum, gives an initial idea of the complexity of the task, mandatory and additional product requirements – their knowledge of the discipline. Agile and Scrum methodologies allow to develop and implement a modern, adapted to the requirements of the labor market training program for IT professionals and helps to develop skills to find the necessary information and interact with peers, which will also help further training and improve their "soft skills". The introduction of Agile-methodology in the educational process radically changes the attitude of students to learning and stimulates their systematic educational work, which in turn affects the quality of work and compliance with deadlines. This allows for a more objective and differentiated assessment of student performance and develops opportunities for a personal-activity approach to learning. We are convinced that the introduction of flexible methodologies in the educational process of training students in technical specialties should become an integral part of the transformation of higher education.

In addition to the above, the proposed approach is more successfully combined with distance learning, which is dictated to us by the realities of today. With the beginning of the quarantine period of COVID-19 in higher educational institutions of Ukraine, the authors adapted the developed approach to the online mode of conducting lectures and practical classes. Maps have been developed to study the entire discipline and connections of each module, so that students in isolation, even

without constant Internet access, can go independently on the map of the module and occasionally adjust their work according to the requirements of the team and the teacher. Agile-methodology of studying disciplines in distance format will be developed in further research and presented together with the results of the proposed adaptation to distance learning in the following articles.

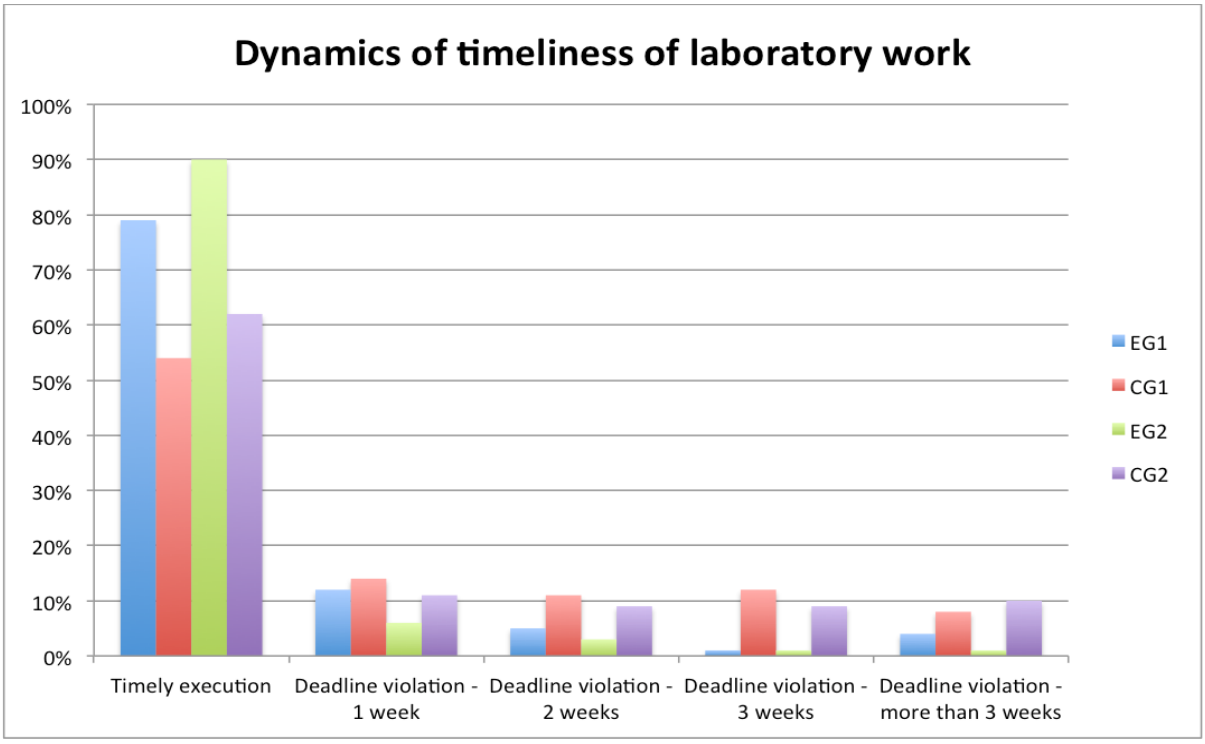

Figure 7: Dynamics of timeliness of laboratory work for students of all subgroups

5. References

[1] D. Semyonov, “Ukrainian higher education is dead. Stay away from it”, Medium, 2016 [Online]. Available: https://cutt.ly/Mymj2R5.

[2] Agile-software development manifesto, 2001 [Online]. Available: https://cutt.ly/SyEftwh.

[3] V. Seminozhenko, “Modern educational policy will cause a lack of IT specialists”, Euro Education, 2014 [Online]. Available: https://cutt.ly/Jymzx4M.

[4] P. Salza, P. Musmarra, and F. Ferrucci, “Agile Methodologies in Education: A Review. In Agile and Lean Concepts for Teaching and Learning”, Springer, pp. 25–45, 2019. Available: https://cutt.ly/Niu2nzk.

[5] MoSCoW Prioritisation, Coley Consulting [Online]. Available: https://cutt.ly/ryB5n4S.

[6] Regulations on the organization of the educational process at the Taras Shevchenko National University of Kyiv, Taras Shevchenko National University of Kyiv, 2018 [Online]. Available: https://cutt.ly/5yEhQf6.

[7] V. Mironova, and M. Pyroh, "AGILE methodology in education as the main tool for building modern educational disciplines of higher education", Economics and Management, №4 (76), pp. 124–130, 2017.

[8] V. Mironova, M. Pyroh, and I. Harko, "Agile-methodology and informatization of the educational process on the example of project work in the discipline "Algorithmization and basics of programming"", in VII International. scientific and technical conf. Problems of informatization, Cherkasy, 2019, P. 30.

[9] J. Stewart, C. DeCusatis, K. Kidder, J. Massi, and K. Anne, “Evaluating Agile Principles in Active and Cooperative Learning”, in Student-Faculty Research Day, CSIS, Pace University, P. B3, 2009. Available: https://cutt.ly/piuBBUI.