UDC 004.91
https://doi.org/10.31891/csit-2022-4-1

IRYNA ZASORNOVA, SERGII LYSENKO, OLEKSANDR ZASORNOV
Khmelnytskyi National University

CHOOSING SCRUM OR KANBAN METHODOLOGY FOR PROJECT MANAGEMENT IN IT COMPANIES

Taking into account the conditions of the functioning of modern business, it is necessary to take into account the constant changes that occur in the process of developing projects in IT companies. Therefore, when creating projects, there is a need for a well-founded choice of project management methodologies, which will allow to quickly solve problems that arise in the process of their creation and will provide an opportunity to ensure high efficiency of project-oriented activities of companies.

The conducted analysis of the methodologies showed that today there are a sufficient number of them, among which there are rigid and flexible ones. Among the hard methodologies, as Waterfall model, V-model, Incremental model, Spiral model, and Iterative model are known. Among flexible methodologies, the Agile model, as well as Scrum and Kanban are most often used. The following flexible methodologies are also known: Lean, eXtreme Programming (XP), Rational Unified Process (RUP), Dynamic Systems Development Model (DSDM), Rapid Application Development (RAD), and Extreme Programming (XP).

It has been determined that most researchers prove in their studies that agile methodologies are more widespread nowadays and are better adapted to the conditions of rapid changes in projects. The only exceptions are those methodologies that apply to certain industries, such as medical, aviation, and others. Such fields involve the use of only brutal methodologies.

The article examines the modern Agile approach to project management, its difference from the traditional approach of the rigid Waterfall methodology or others; the scope of application of Agile; a short story; the advantages and disadvantages of implementation and its use.

The article provides a comparative analysis of the most popular today's flexible Scrum and Kanban methodologies. The roles that are indicated in projects in Scrum and Kanban are considered. The events performed by the teams are also analyzed in more detail.

The article elaborates a block diagram for a justified choice of Scrum or Kanban methodology. The article is based on research on the international and Ukrainian markets of IT companies.

The direction of future research of the authors is to conduct a more detailed analysis and comparison of rigid and flexible methodologies with the aim of using combined forms of project management.

Keywords: IT companies, rigid and flexible methodologies, Waterfall model, V-model, Incremental model, Spiral model, Iterative model, Agile model, Scrum and Kanban, roles in the project, Product Owner, Scrum team, Scrum master, Sprint Planning, Daily Scrum, Sprint Review, Sprint Retrospective, backlog, block diagram, comparative analysis, justified choice of methodology.

ІРИНА ЗАСОРНОВА, СЕРГІЙ ЛИСЕНКО, ОЛЕКСАНДР ЗАСОРНОВ
Хмельницький національний університет

ВИБІР МЕТОДОЛОГІЇ SCRUM АБО KANBAN ДЛЯ УПРАВЛІННЯ ПРОЄКТАМИ В ІТ КОМПАНІЯХ

Враховуючи умови функціонування сучасного бізнесу потрібно враховувати постійні зміни, які виникають в процесі розробки проєктів в ІТ компаніях. Тому, при створенні проєктів виникає необхідність обґрунтованого вибору методологій управління проєктами, що дозволить швидко вирішувати проблеми, які виникають у процесі їх створення, та дасть можливість забезпечити високу ефективність проєктно-орієнтованої діяльності компаній.

Проведений аналіз методологій показав, що на сьогодні їх існує достатня кількість, серед яких є жорсткі та гнучкі. Було визначено, що більшість дослідників доводять у своїх працях, що гнучкі методології є більш широко розповсюдженими в наш час і краще адаптуються до умов швидких змін у проєктах. Виключення становлять лише ті методології, які застосовуються для певних галузей, таких як медична, авіаційна та інші. Такі галузі передбачають використання тільки жорстиких методологій.

У статті розглядається відносно сучасний підхід Agile до управління проєктами, його відмінність від традиційного підходу жорсткої методологі, наприкладі Waterfall, або інших; область застосування Agile; коротка історія; переваги та недоліки впровадження та його використання.

У статті виконано порівняльний аналіз найпопулярніших на сьогодні гнучких методологій Scrum і Kanban. Розроблено блок-схему для обґрунтованого вибору методології Scrum або Kanban. Стаття базується на дослідженнях міжнародного та українського ринку ІТ компаній.

Напрямком майбутніх досліджень авторів є проведення більш детального аналізу та порівняння жорстких і гнучких методологійї з метою використання комбінованих форм управління проєктами.

Ключові слова: ІТ компанії, жорсткі та гнучкі методології, Waterfall модель, V- модель, Incremental модель, Spiral модель, Iterative модель, Agile модель, Scrum і Kanban, ролі у проєкті, Product Owner, Scrum команда, Scrum майстер, Sprint Planning, Daily Scrum, Sprint Review, Sprint Retrospective, backlog, блок-схема, порівняльний аналіз, обґрунтований вибір методології.

Introduction

The operating conditions of modern business in the world market are characterized by constant changes. In recent years, these changes have become even more dynamic. Reasonable choice of project management methodologies will allow solving problems that arise in the process of their creation and will provide an opportunity to ensure high efficiency of project-oriented activities of companies.

The variety of concepts and a large number of different applications for the implementation of team projects in IT companies, in particular for the development of software, complicates the choice of the necessary methodology. Therefore, a scientifically based choice of methodology for obtaining high-quality development products in a timely manner is an urgent task. At the same time, the following information must be taken into account: project type, customer requirements, resources, deadlines, tools, team composition, and others.

We will analyze the methodologies used by IT companies. So, today there are different methodologies for managing the implementation of projects in companies. The choice of methodology depends on the type of company and the project it is developing. In modern companies, project management methodologies are usually divided into rigid and flexible, sometimes they use their mixed form [1, 2]. Rigid methodologies include:

– Waterfall model – in this methodology, the stages depend on each other and the next one begins when the previous one is completed, thus forming a progressive (cascading) forward movement. Teams of different stages do not communicate with each other, each team is clearly responsible for its own stage. Advantages: all stages of the project are performed in a strict sequence; the strictness of the stages allows you to plan the deadlines for the completion of all works and the corresponding resources (money and human); requirements remain the same throughout the cycle. Disadvantages: difficulties in formulating clear requirements and the impossibility of changing them; testing begins only in the middle of project development; until the development process is complete, users cannot be sure whether the product being developed is of good quality;

– V-model – involves dividing the project into parts (stages, iterations) and passing the stages of the life cycle on each of them. The use of an iterative model reduces risks and makes it possible to complete the development at the end of each iteration, the set of stages forms the final result. Advantages: strict phasing; minimization of risks and elimination of potential problems due to the fact that testing appears in the early stages; improved time management. Disadvantages: inability to adapt to the customer's changed requirements; long development time (sometimes lasting up to several years) leads to the fact that the product may not be needed by the customer, as his needs change; there are no actions embedded in the risk analysis;

– Incremental model – the cycle is divided into smaller modules that are easy to create. Each module goes through requirements definition, design, coding, implementation and testing phases. The development procedure according to the incremental model involves the release of the product in the first major stage in the basic functionality, and then the sequential addition of new functions. The process continues until a complete system is created. Advantages: the customer can give feedback on each version of the product; there is an opportunity to review risks related to costs and compliance with the schedule; the customer gets used to the new technology gradually. Disadvantages: the functional system must be fully defined at the beginning of the iteration allocation life cycle; with constant changes, the structure of the system may be disturbed;

– Spiral model – using this model, the customer and the development team analyze the risks of the project and execute it in iterations. The next stage is based on the previous one, and at the end of each round - a cycle of iterations - a decision is made on whether to continue the project. Advantages: special attention is paid to risk management; additional features may be added at later stages; the flexible design is possible. Disadvantages: risk assessment at each stage is quite expensive; constant feedback and reaction of the customer can provoke new and new iterations, which can lead to a temporary delay in product development; more applicable for large projects;

– Iterative model – the process of creating software, which is carried out in small stages, during which the analysis of the obtained intermediate results is carried out, new requirements are put forward and previous stages of work are adjusted. Does not require the beginning of a complete specification of requirements. The creation begins with the implementation of a part of the functionality, which becomes the basis for determining further requirements. The product is created in such a way that, first of all, the basic functionality that works is designed. Then with each iteration, it is improved, and new features are added. Advantages: risk reduction – early detection of conflicts between requirements, models and project implementation; organization of effective feedback of the project team with the consumer, creation of a product that really meets his needs; rapid release of a minimally valuable product and the ability to bring the product to market and begin operation much earlier. Disadvantages: problems with the architecture and overhead costs - when working with chaotic requirements and without a developed global plan, the architecture of the program may suffer, and additional resources may be needed to bring it to an adequate appearance; there is no fixed budget and deadlines, and strong involvement of the customer in the process is required.

Authors Petersen K., Wohlin C., and Baca D. in the article [3] performed an analytical review of rigid methodologies and the problems that arise when using them. Rigid methodologies are often used in projects where late-stage changes are too expensive or impossible. For example, to create complex engineering structures (aviation, construction, etc.). Rigorous methodologies are also used to develop software in systems for military or medical needs. Also, an overview of the advantages and disadvantages of the Waterfall model is covered by the authors Adetokunbo A.A. Adenowo, and Basirat A. Adenowo in the publication [4].

Among the flexible methodologies, the following are known today:

– Lean – the concept of «Lean production», which is based on the optimization of the company's work, where the processes are focused on the final value (the developed project or the sold product) and the removal of those teams that do not create additional value. According to the conclusions of Grynko T., the advantages of this concept include the following: saving time, which will allow the completion of a larger number of projects; flexibility; involvement of each team member in the optimization process; maximum focus on the consumer. Among the shortcomings, the author includes the need for extremely high qualifications of project managers or other management; does not contribute to increasing the scientific and technological level (R&D); needs a perfect information support system (ISS) and others [5];

– eXtreme Programming (XP) – a software development methodology that differs from other flexible methodologies in that it is used only in the field of software development. It cannot be used in another business. XP Principles: Simplicity, Communication, Feedback, Courage and Respect. Advantages: extreme programming; the customer receives exactly the product he needs; the code always works due to constant testing and continuous integration; pair programming; low risks. Disadvantages: it is difficult to predict the time spent on the project; the methodology works only with senior specialists; not suitable for large projects;

– Rational Unified Process (RUP) – a methodology that involves product development in the following stages: initial; specification; construction; implementation. Each of them includes one or more iterations;

– Dynamic Systems Development Model (DSDM) – a methodology that demonstrates a set of principles, defined types of roles and techniques. The principles are aimed at the main goal - to deliver the finished project on time and within the budget, with the ability to adjust requirements during development;

– Rapid Application Development (RAD) – the methodology of rapid development of applications, which involves the use of tools for visual modelling (prototyping) and development. RAD involves small development teams, deadlines of up to 4 months, and active involvement of the customer from the early stages. This methodology is based on the requirements, but there is also the possibility of changes during the development of the system. This approach allows you to reduce costs and reduce development time to a minimum;

– Extreme Programming (XP) – the methodology focused on constantly changing product requirements offers 12 approaches to achieve effective results in similar conditions. Among them: a quick plan and its constant change; simple architecture design; frequent testing; simultaneous participation of two developers in one task or even at one workplace; continuous integration and frequent small releases;

– Agile model – it is a powerful methodology for software development, which represents certain systems that determine the order of tasks, evaluation and control methods. Since 2000, in the practice of project management, the agile model has been widely used. Since then, methods of its use (Scrum & Kanban, XP, etc.) and software for Agile project management have been developed and implemented. This methodology is so progressive that it is also used in other areas (marketing, education, finance, construction, and others). Advantages: quick decision-making due to constant communications; risk minimization; easier work with documentation. Disadvantages: a large number of meetings and conversations, which can increase the time of product development; it is difficult to plan processes because the requirements are constantly changing; rarely used to implement large projects.

Many researchers are engaged in the study and research of project management methods. For example, the authors Brych V., and Peryt I., believe that in times of global and constant changes, the most effective is the use of flexible methodologies [6].

According to the researchers Abrahamsson P., Salo O., Ronkainen J., and Warsta J. agile model makes it possible to adjust the schedule of the project, make certain changes to the project during its implementation, constant improvement and flexible reactions to changes in requirements, potential and understanding the problems that need to be solved, etc [7].

In articles [8, 9], the authors conducted an analysis of the most popular software development methodologies, from which it follows that flexible methodologies better meet business needs and are the most popular among IT companies.

Therefore, the analysis of the methodologies used today in the information technology (IT) market showed that they are chosen based on the direction of the project, its budget, implementation terms and the ability to adapt to flexible business needs. At the same time, due to the lack of clear regulation of actions, the development of projects within the cycle is ensured in different ways. Therefore, the purpose of this article is to perform a comparative analysis of the use of flexible methodologies in the process of the development life cycle and their selection for the creation of a high-quality software product in clearly defined terms.

Performing an analysis of flexible methodologies and developing a block diagram for their selection

In order to achieve high project implementation results in any IT company, it is necessary to carefully plan each stage of its life cycle. For this, it is necessary to make a justified choice of a flexible methodology for each of the stages, which will make it possible to implement the project tasks. To do this, it is necessary to perform an in-depth analysis of the most used flexible Scrum and Kanban methodologies and develop an algorithm for their selection.

Comparing the Scrum and Kanban methodologies, the following criteria can be distinguished: in Scrum, all tasks are performed in sprints that last from 2 to 4 weeks, respectively. The following meetings are usually held: sprint planning, daily scrum, sprint review, and sprint retrospective. Usually, these meetings do not exceed 15 minutes. (table 1). In Kanban, meetings are optional. They can be of the following types: daily meeting, replenishment, deli-very planning meeting, service delivery meeting, operations review, risk review, strategy review.

The release occurs only when the product or some part of it is ready. Usually conducted several times a day or once a week.

Table 1

Scrum events

Events	Inspection	Adaptation

Sprint Planning	– Product Backlog;
	– Commitments Retrospective;
	– Definition of Done
		– Sprint Goal;
		– Forecast;
		– Sprint Backlog

Daily Scrum	– Progress toward Sprint Goal	– Sprint Backlog; – Daily Plan

Sprint Review	– Product Increment;
	– Product Backlog (Release);
	– Market-business conditions
		– Product Backlog

Sprint Retrospective	– Team & collaboration;
	– Technology & engineering;
	– Definition of Done
		– Actionable improvements

In Scrum, a backlog is created with clear tasks to be completed in a sprint. Kanban has a workflow where all the tasks that must be completed are displayed and they are moved from one status to another. At the same time, there is no clear time frame planning.

There can be no changes or additions in a sprint in Scrum. At the beginning of the sprint, it is decided how many and which tasks can be done (sprint scope). Such changes are possible in Kanban. When a task is completed, a new task can be created.

Scrum meetings are usually attended by the product owner, scrum team, and scrum master. In Kanban, project management can be performed by a manager, there is no focus on the distribution of roles, attention is focused on the production of the project. The roles, artefacts, and practices in Scrum are listed in table 2.

Table 2

The roles, artefacts, and practices in Scrum

Roles (Who?)	Artefacts (What?)	Practices (How?)
Product Owner	– Product Backlog	– Sprint
Scrum team	– Sprint Backlog	– Sprint Planning Meeting
Scrum master	– Potentially Shippable Product	– Daily Standup
	– Sprint Burndown Chart	– Sprint Review
		– Sprint Retrospective

In Scrum, task boards are created for each new sprint. The organization of the board has the following parts: «To do», «In progress», «In testing» and «Done».

In Kanban, the board is one and the same. This is an advantage if the team is small because you can see on one board all the tasks, their relationships with each other and the progress of execution. Usually consists of columns «To do», «In progress», and «Done».

Work productivity in Scrum is measured in story points, that is, in the speed of completing tasks per sprint. Assessment of tasks is always available.

In Kanban, productivity is measured in the speed at which a task moves from the «To do» status to the «Done» status. Tasks are not always evaluated. A comparative analysis of Scrum and Kanban methodologies is given in table 3.

Table 3

Comparative analysis of Scrum and Kanban methodologies

№	Parameters	SCRUM	KANBAN
1	Visualization of the life cycle	+	+
2	Availability of iterations	+	+
3	Presence of backlog	+	-
4	Daily sprints	+	+
5	Big projects	-	+
6	Medium projects with the possibility of making changes	+	+
7	People and interaction come first	+	-
8	Processes and tools come first	-	+
9	Readiness for change	+	+
10	Development speed	+	+
11	Minimization of risks	+	+
12	Repeated sprints of fixed duration	+	-
13	Continuous process	-	+
14	Release at the end of each sprint after manager approval	+	-
15	The flow continues without interruption or at the discretion of the team	-	+
16	Roles: product owner, scrum master, scrum team	+	-
17	Roles: a team led by a manager	-	+
18	The main indicator is the speed of the team	+	-
19	The main indicator is time	-	+
20	During the sprint, changes are undesirable	+	-
21	Changes can happen at any moment	-	+
22	Self-organization of the team regarding the distribution of tasks	+	+

From table 3, it can be concluded that each of the methodologies has its own differences.

Table 3 can be used to create a block diagram of a Scrum or Kanban methodology selection program for a company's work organizations. However, for this, it is necessary to make several transformations with the table 3.

We perform the first transformation by discarding parameters that do not affect the selection. Namely, parameters by numbers: 1, 2, 4, 6, 9, 10, 11, 22. Because the parameters of the methodologies have the same positive value. The result of the first transformation is the reduction of the number of parameters to 14 (table 4).

Table 4

Parameters that affect the choice of methodology

№	Parameters	SCRUM	KANBAN
1	Presence of backlog	+	-
2	Big projects	-	+
3	People and interaction come first	+	-
4	Processes and tools come first	-	+
5	Repeated sprints of fixed duration	+	-
6	Continuous process	-	+
7	Release at the end of each sprint after manager approval	+	-
8	The flow continues without interruption or at the discretion of the team	-	+
9	Roles: product owner, scrum master, scrum team	+	-
10	Roles: a team led by a manager	-	+
11	The main indicator is the speed of the team	+	-
12	The main indicator is time	-	+
13	During the sprint, changes are undesirable	+	-
14	Changes can happen at any moment	-	+

The second transformation is the grouping of parameters according to the priority of selection. Parameter 2 (is the size of the project large) should be the first because a positive answer to it makes the choice of Kanban methodology. Because further choices do not make sense. The second parameter should be 1 (about the presence of a backlog), since a positive answer to it also makes further selection impractical. In addition, it is necessary to reformulate the parameters into questions.

The third transformation is a combination of parameters that are mutually opposite. In parameters 3 and 4, the user must make a choice that for him, people or tools are more important in the organization of the process. It is also necessary to combine parameters 5 and 6 (continuous process, or sprints of fixed duration); 7 and 8 (release at the end of each sprint after approval by the manager, or the flow continues without interruption or at the discretion of the team); 9 and 10; 11 and 12; 13 and 14. After combining the parameters, it is necessary to reformulate the combined parameters into questions in the same way as in the previous transformation.

The fourth transformation is replacing the symbol «-» with «no» and «+» with «yes».

Only after the last transformation, table 4 with parameters takes on a form that can be used to develop a block diagram of the program for choosing the company's work organization methodology (table 5).

Table 5

Data for the development of a block diagram of the methodology selection program

№	Parameters	SCRUM	KANBAN
1	Your project is big	**no**	**yes – end**
2	You will have a list of work tasks arranged in order of importance (backlog)	**yes – end**	**no**
3	In the first place you are:	people and their interaction **(yes)**	tools come first **(no)**
4	The process is:	repeated sprints of fixed duration **(yes)**	continuous teamwork **(no)**
5	The flow is:	release at the end of each sprint after manager approval **(yes)**	continues without interruption or at the discretion of the team **(no)**
6	Roles:	Product Owner, Scrum Master, Scrum Team **(yes)**	a team led by a manager **(no)**
7	The main indicator:	team speed **(yes)**	time **(no)**
8	Changes:	during the sprint, changes are undesirable **(yes)**	changes can happen at any moment **(no)**

As we can see from table 5, the first two parameters allow continuing the choice of methodology, if the answer is «no». The last six do not allow you to switch to the next one, because the program will end when you select any of the six listed items. However, there is a possibility of the so-called evaluation selection of the listed parameters. After choosing one of the two possible answer options, the chosen one will be given a weight equal to «1», and the other – «0». The next step is to count the number of units in the methodology column. The methodology is chosen if the number of positive answers is greater. However, it can be seen from table 5 that the number of positive answers can be equal to three. In this case, you need to choose a mixed methodology for the organization of the company's work.

In the block diagram (fig. 1), the parameters from table 5 are given by numbers.

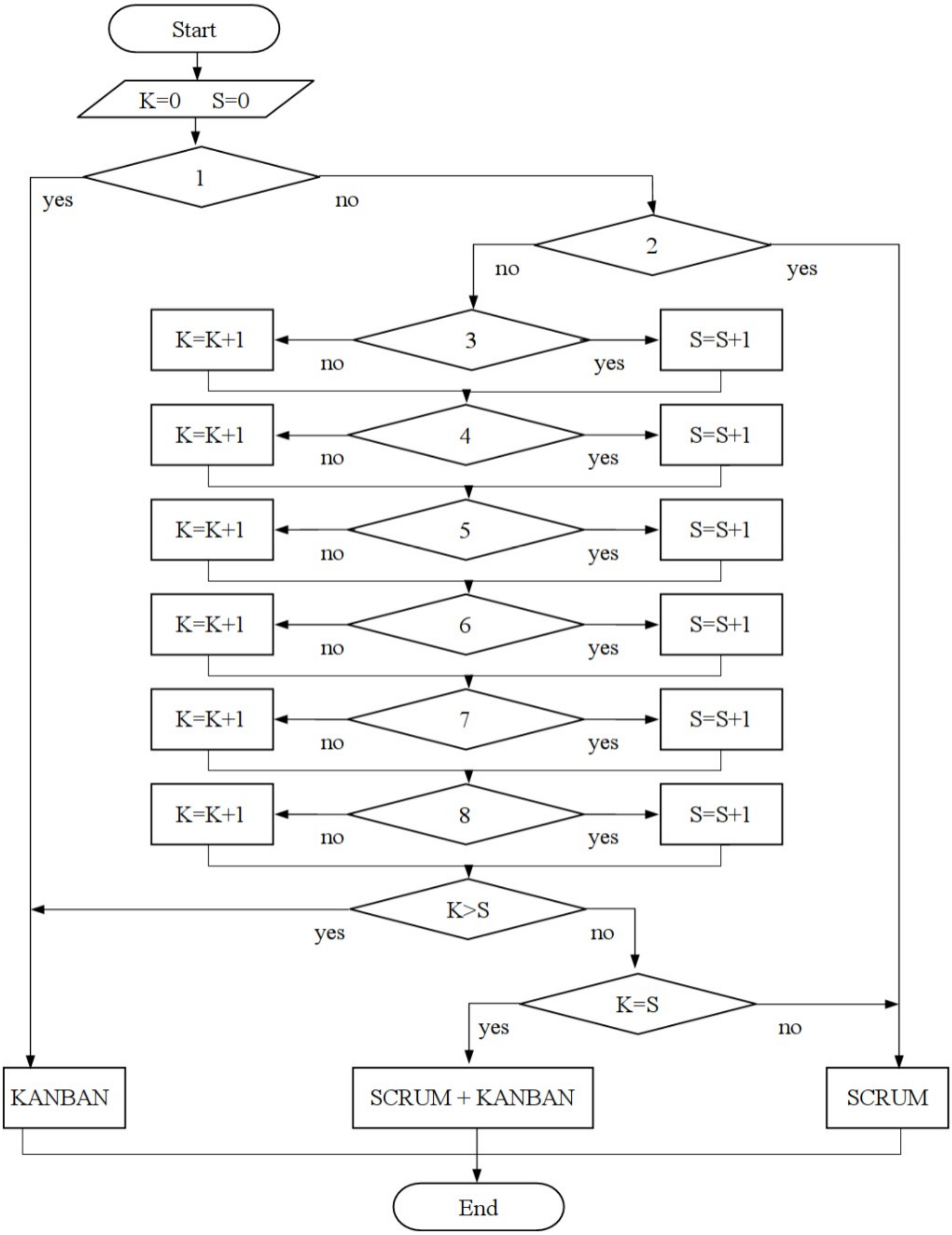

Fig. 1. Block diagram of Scrum or Kanban methodology selection program

Conclusions

Therefore, the analysis of the methodologies has shown that today there are a sufficient number of them, among which there are rigid and flexible ones. It was determined that most researchers prove in their works that flexible methodologies are more widespread nowadays and are better adapted to the conditions of rapid changes in projects.

The article provides a comparative analysis of the most popular today's flexible Scrum and Kanban methodologies. A block diagram has been developed for an informed choice of Scrum or Kanban methodology. The article is based on research on the international and Ukrainian markets of IT companies.

The direction of the authors' future research is to conduct a more detailed analysis and comparison of rigid and flexible methodologies with the aim of using combined forms of project management.

References

1. Scrum Alliance. URL: https://www.scrumalliance.org/.
2. Continuous Improvement Through Enterprise Agility. URL: https://xscion.com/agile/.
3. Petersen K., Wohlin C., Baca D. (2009). The Waterfall Model in Large-Scale Development. In: Bomarius, F., Oivo, M., Jaring, P., Abrahamsson, P. (eds) Product-Focused Software Process Improvement. PROFES 2009. Lecture Notes in Business Information Processing, vol 32. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-02152-7_29.
4. Adetokunbo A.A. Adenowo, Basirat A. Adenowo. Software Engineering Methodologies: A Review of the Waterfall Model and Object-Oriented Approach. International Journal of Scientific & Engineering Research, Volume 4, Issue 7, July-2013.
5. Grynko T. Advantages and disadvantages of the «Lean production». URL: http://www.confcontact.com/2017-ekonomika-i-menedzhment/4_grinko.htm.
6. Brych V. Basics of business management of domestic economic entities / V. Brych, I. Peryt // Scholarly notes of the «KROK» University. Chapter 6. Management and marketing. - No3 (55). - 2019. - Pp. 82-93.
7. P. Abrahamsson, O. Salo, J. Ronkainen, J. Warsta. Agile Software Development Methods: Review and Analysis. URL: https://arxiv.org/abs/1709.08439.
8. Vavilenkova A. Analysis of flexible methodologies of software development for implementation in team projects / A. Vavilenkova // Visnyk NTU «KhPI». - №1 (7). – 2021. – Pp. 39-46.
9. Korzh B. Flexible models of teamwork management of engineering projects / B. Korzh // Bulletin of Taras Shevchenko National University of Kyiv. Economics, 2019; 6(207): Pp. 21-27.

Iryna Zasornova Ірина Засорнова	PhD, Associate Professor of Computer Engineering & Information Systems Department, Khmenlnytskyi National University https://orcid.org/0000-0001-6655-5023 e-mail: izasornova@gmail.com	Доцент кафедри комп'ютерної інженерії та інформаційних систем, Хмельницький національний університет
Sergii Lysenko Сергій Лисенко	DrSc (Engineering), Professor of Computer Engineering & Information Systems Department, Khmenlnytskyi National University https://orcid.org/0000-0001-7243-8747 e-mail: sirogyk@ukr.net	Доктор технічних наук, професор кафедри комп'ютерної інженерії та інформаційних систем, Хмельницький національний університет
Oleksandr Zasornov Олександр Засорнов	PhD, Associate Professor of Computer Engineering & Information Systems Department, Khmenlnytskyi National University https://orcid.org/0000-0001-9931-4154 e-mail: azasornov@gmail.com	Доцент кафедри комп'ютерної інженерії та інформаційних систем, Хмельницький національний університет