An activity-based approach to up-skilling teachers who teach software products development

K A Rutskaya[1], **A M Aronov**[2], **V N Rutskiy**[1], **R Yu Tsarev**[1,3], **N V Titovskaia**[3], **K Y Zhigalov**[4,5], **N V Bystrova**[6] **and Younes El Amrani**[7]

[1] Siberian Federal University, 79, Svobodny pr., Krasnoyarsk, 660041, Russia
[2] Moscow City University, 4, Vtoroy Selskohoziajstvenny proezd, Moscow, 129226, Russia
[3] Krasnoyarsk State Agrarian University, 90, Mira ave., Krasnoyarsk, 660049, Russia
[4] V. A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences, 65, Profsoyuznaya street, Moscow, 117997, Russia
[5] Moscow Technological Institute, Block 2, 8, Kedrova, Moscow, 117292, Russia
[6] Minin State Pedagogical University of Nizhny Novgorod, 1, Ulyanova street, Nizhny Novgorod, 603005, Russia
[7] Abdelmalek Essaâdi University, aviation route, BP. 1373, Tangier, Morocco

E-mail: tsarev.sfu@mail.ru

Abstract. Effective up-skilling for teachers involved in teaching software products should take into account the challenges of teaching practice. Despite the long practice of organizing the teaching of computer science in educational organizations in Russia, relevant up-skilling courses are not methodologically supported. The paper presents an active, practice-oriented approach to the organizing of up-skilling courses for teachers who teach the creation of software products in the framework of intensive forms of education and on the model of training for students used in intensive schools for development, design, promotion and management of software products.

1. Introduction

In the modern world, about 13% of companies in Central and Eastern Europe, including Russia, are implementing a digital business transformation strategy and are actively using modern digital technologies [1]. However, one of the main barriers is the lack of qualified IT professionals. According to IDC Microsoft, only 3.5% of employees are fully up-to-date. The highest (35%) is the lack of qualified IT professionals in companies implementing cloud technologies.

The professions that are in great demand in the IT sector are software developers, UI/UX (user interface and experience) designers, QA specialists (Quality Assurance, testing and product quality control), data analysts, Internet marketing specialists, product managers and project managers.

Activities in the IT field and the creation of software products do require not only “hard” skills such as knowledge of specific programming languages but also specific “soft” skills such as involvement, honesty, innovation and creativity; communication skills and the ability to interact productively with members of the project team, the ability to make independent, responsible decisions. At present, the training of such specialists is mainly the task of the specialists themselves, engaged in self-education and in searching for opportunities to obtain the first professional experience in project activities. Accordingly, the success of the training for the IT industry depends on the early specialization and the ability to provide an early professional test.

However, a review of recent studies on the training of teachers, in particular computer science teachers, allows us to conclude that computer science teachers in state educational institutions in Russia do not have experience in creating custom software products [2], [3], [4]. They know the methodology of teaching the subject and, as a rule, they have command of one or two programming languages, often outdated, rarely used in the work of modern IT companies. Sometimes interactive teaching methods are used in the learning process to develop methodological training products, for example, studying in small groups. However, in educational practice, there are merely some descriptions of the use of modern tools or technologies that are in demand in the IT industry, for example, methods of agile development software products, in which “soft” skills are of particular importance. Thus, it can be argued that the training of computer science teachers in Russia is outdated not only in the quality of acquired “hard” skills but also in the quality of “soft” skills.

We have to admit that the formation of experience in developing software products is not a mandatory element of the educational programs of specialists – teachers in the field of computer science. These programs are more aimed at organizing the teaching of computer science. Similarly, the development of software products is not the main subject of training IT specialists in the framework of the corresponding state educational programs implemented in educational institutions in Russia.

In the framework of the national project "Personnel for the Digital Economy" of the federal program "Digital Economy of the Russian Federation", the authors developed and implemented the model for the advanced training of teachers interested in mastering modern technologies for working in small teams for the flexible development of software products according to the Agile methodology. A description of the model is presented in this paper.

The question, therefore, becomes relevant on what the goals, objectives and organizational and pedagogical conditions of advanced up-skilling of teachers could be in the field of software product development. Moreover, the authors have experience in organizing intensive educational practices aimed at mastering various professional activities by students, as well as conducting up-skilling courses, for example, courses for teachers who guide educational and research activities and project activities of students [5]. The problem of understanding and implementing the activity-based approach in education is presented in [6], [9], [10].

2. The principles of organization of training in the creation of software products

Let us turn to the grounds that the authors of this article put into the development of the content of a continuing education program for teachers teaching software products. To do this, we describe the functions of a teacher in implementing the practice of creating software products with students, which, in principle, is close to project and research activities in general. If earlier they spoke about the possibility of full-fledged project and research activities of students, now researchers recognize that such achievements as scientific discoveries or the results of professional projects in students' work are more an exception than a rule. The real value of a student’s research or design work lies in the student’s professional test during research or design work, during which the necessary professional “soft skills” are acquired [7].

Most often, the main stages of developing a software product are:

1. Preparation (including discussion of the order with the customer and its description as a task).
2. The project design (including the creation of technical specifications from the task description).
3. Creation (including design, coding, testing, documentation).
4. Support (including implementation and maintenance).

In this regard, in studies on the developing and engineering of software products, the essential aspects of the activities of individuals developing software products are [11]: communication environment (teamwork), purpose (goal) and the intrinsic resources. In this case, the purpose is a solution to the specific task of developing a software product. The intrinsic abilities (resources) of the individual solving the problem are the means, the methods that he possesses, in order to translate the task into the programming language, enter into communication with team members, achieve the goal, receive feedback from the customer and conduct reflection.

In this case, we can talk about three levels of individual thinking when solving problems of programming itself:

1. Creation of an applied formal model and its description in the language of data structures, statement of the problem in terms of a formal model (modelling);

2. Description of the sequence of transformations of the formal model that together solve the problem (algorithmizing);

3. Writing a direct sequence of instructions for the executor (coding; it is this level that is traditionally considered programming in the public mind).

This means that in order to teach students how to develop software products, the teacher needs to live through the situation of software product development. Essential tasks for the teacher are methodological work both with the very process of organizing training (communication among students) and with a logical-subject analysis of the selection of educational material. In this case, the subject matter of the study should be accessible to students and be relevant to their interests, give the opportunity to transform the material on their own. The logical and subject analysis is aimed at highlighting a new basic subject action performed by students, the mastery of which is expected in the course of the studies. Carrying out such an analysis allows one to adapt the material for the age-related capabilities of students, to keep the historical aspect of the formulation and solution of the problem.

When working with students in order to “imitate” communication in the working team developing a software product, it is necessary for the teacher to organize a dialogue among students so that at least 2 positions are presented: the customer and the order executor (“problem solver”). The organization of participants’ understanding of each other in the dialogue becomes the task of the teacher. For example, if the teacher promptly asks the question of who understood what one of the participants was saying, then he or she would replace the traditional role of the all-knowing opponent with the role of an assistant in communication.

With an active approach to the organization of training, the teacher holds the priority of the reflexive function: he or she monitors the “personality-subjective” changes that occur with students at each stage and gradually raise their independence and subjectivity.

The authors' work experience with Computer Science teachers working in universities, secondary vocational education, and schools allows to highlight the key difficulty that teachers face - the lack of their own experience in living through the situation of developing software products as a result of the team and business process (the training situation and real process). Up-skilling courses for computer science teachers are often aimed at introducing authors to teaching methods for motivating students, mastering the teaching methods of the subject, and developing programs for organizing elective classes. However, the content of the technology for developing software products as a process for solving the task (customer’s order), which is implemented by the team, often remains hidden. Teachers lack the significance of conducting a preliminary logical and objective analysis of the problem in order to plan the course of its solution by students as a project; difficulty in allocating time for organizing work with students, and as a result, solving a problem and performing actions instead of a student.

Thus, the difficult position of a teacher teaching integrated software product development, in contrast, is aimed at solving four basic problems:

1. Logical-subject reconstruction of the task content.

2. Management of communication in the students project team and team interaction with the customer of the project task.

3. Observation of the personality-subjective changes of students.

4. Management of the process of product development by students: provision of premises,

supplying the team with hardware, software and materials, monitoring of key stages of product development, etc.

Firstly, the indicated tasks allow us to say that the content of the activities of the teacher who teach the development of software products becomes related to the implementation of organizational and managerial activities [8], since “represents the teacher’s activity in organizing and managing the activities of students”.

Secondly, the logical and subject reconstruction of the material, the organization and management of communication, the generalization of the experience of students causes the greatest difficulties of practising educators.

It is necessary to distinguish between the quality of the project (educational) software product developed by students and the quality of the organization of training in developing software products. By the quality of students' project work, we understand, on the one hand, the content of academic work (order interpretation, effective choice of a solution, degree of independence in execution), and on the other hand, the quality of the project software product developed (software product functionality, software solution optimality and reliability, convenience of software product interface, compliance with customer requirements in terms of price/quality/speed of product development). The quality of the organization of training in developing software products implies the teacher's choice of the level of tasks in accordance with the abilities of students, the quality of creating conditions for communication, the ability to redesign the process.

3. Implementation of an activity-based approach to up-skilling of teachers teaching development of software products

The authors took the model described in [5] as the basis for up-skilling courses for teachers, and also implemented it in practice during intensive schools of the project “Factory of programming: the path of a professional”. This project has been conducted in Russia and in the Krasnoyarsk Kray for more than 25 years to train university students, colleges students and schoolchildren in development, design and promotion of software products. Let us introduce the basic components of this model that form the basis of teacher up-skilling seminars. The “Research task” element fixes the structure and essence of actions from setting the task to its implementation. The most complete model of programming instruction levels is described in [12]. The content of the component allows you to uncover the challenges a developer faces, what is the methodological vector of organizing seminars.

The element “Professional and personal experience of the teacher” includes a set of ways to guide students, the level of proficiency in methodological tools, and the horizons of the teacher. The need for the introduction of this component is justified by the results of the study [5]. The ability and willingness of the teacher to master new subject content, as well as ensuring the depth of the student’s advancement in solving the problem, the ability to create a contextual field for understanding the essence of the problem, directly depends on the breadth of horizons and the arsenal of methodological methods that the teacher owns. Thus, in order to teach the student how to develop software products, only knowledge of programs and their own individual programming experience is insufficient, it is necessary to test themselves in the team developing software products, and therefore, to enter substantive communication with the development team and the customer.

The communicative environment element has two functions. Firstly, as shown above, modern software products are created in a team from receiving an order to delivering a finished product. Secondly, from the point of view of the quality of the learning process organization, the communicative component provides diagnostics of the material acquiring. L.S. Vygotsky, E.V. Ilyenkov et al. note that the level of mastering the method, the free possession of subject content is observed through the freedom to operate with the corresponding language structures, which gradually enter the human cultural field [6].

The basis of this understanding is the methodological idea of mastering the method of objective action, which includes three levels: action according to the model, the allocation of significant relationships, the functionalization of the method of action (Delta model). Additionally, one can fix the "zero level" - the level of initiative tests, when the teacher acts as an enthusiast. At the “zero level”, the teacher understands the development of a software product as a conversation with the customer of the product (in particular, the teacher himself), a test description of the key tasks of the order, designing the key algorithm of the program code. However, such an independent test and obtaining a meaningful expert opinion from the customer on the basis of the coding allows us to realize the need for mastering the samples and structuring the chaotic work of software products development in order to obtain the desired result.

According to the described structure, mastering the first level means teacher’s mastering of the general meaning and form of the action mode. The observed result of the actions of the teacher as leader is manifested in the ability to describe independently the solution to the problem – to write the technical specifications of the process of software product development for customer order, to decompose it into blocks and to conduct modelling of algorithms that imply the implementation of the functionality incorporated into the product.

The result of mastering the second level is a test of the teacher’s pedagogical ideas using the material presented at the first level. At the second level, there is a need to refer to action models, highlighting significant aspects for designing the development of specific software products and managing (supervising) a group of students engaged in developing a custom software product for a customer. The teacher organizes the solution of tasks of development, design and promotion of the software product and the general organization of project work together with the students' team. This project work is similar in structure to the one that was set as part of the pedagogical workshop. Qualitative changes during the transition to the second level are manifested in the teacher’s questions regarding the logical-subject analysis of the problems of designing, developing and supporting a software product before offering it to a student. Questions are based on examples of the organization of communication in a group of students, mastering terms about the structure and processes of developing software products based on popular templates, libraries of programming objects, etc.

The third level of teachers mastering project management methods for developing software products is aimed at encouraging students to search for "their" specific non-trivial solutions to the problems of developing, designing or promoting software products. Thus, a professional test (functionalization) of the methods of working with the material mastered at previous levels and the principles of leadership takes place. Also, the professional orientation of students takes place. The main difficulties are connected with the choice of material, which directly depends on the personal, professional and communicative experience of the teacher as a leader of the project activity for developing software products. There is an assumption that an expert teacher who demonstrates the development of the third level can act as an expert for projects on developing software products in the process of implementing intensive forms of training (intensive schools, workshops, competitions, hackathons).

In accordance with the tasks of mastering the “hard” and “soft” skills of organizing the development of software products, teacher up-skilling courses consist of three blocks:

1. Lecturing and seminar block: teaching teachers the content of activities when developing software products, mastering or improving knowledge of the necessary programming languages, software products for design, ways to market software products and manage their development and launch, the formation or improvement of “hard skills” in developing software products.

2. Project block: developing a software product by order of an external customer, during which the formation or improvement of teachers’ “soft skills” in developing a software product takes place.

3. Methodological block: pedagogical task to create a curriculum for a course aimed at organizing and managing groups of students when they develop custom software products.

Up-skilling courses are held within and in parallel with the training of students (students of universities, colleges, schools) at the intensive 7, 10 or 14-day schools aimed at developing software products by students in accordance with real product orders from partner organizations of the Project

“Programming Factory: The Path of the Professional”.

During intensive sessions of the School, both students and teachers undergo a brisk casting to identify their professional inclinations and the opportunity to work in training "IT firms" within the framework of their professional positions chosen ("programmer", "designer", "advertiser and IT marketer” or “IT company manager”). Further, students and teachers not only receive the necessary “hard” professional knowledge and skills in the chosen educational and professional positions but also master the “soft” skills of professional communication, well-coordinated teamwork, developing creativity, increasing responsibility, learning to achieve a team result in the process of developing various digital products based on needs of real market customers - from the creation of computer games, webservers, websites and medical computer simulators to services that are in demand by enterprises, IT companies, educational institutions.

The format of the educational work of the participants of the School “Programming Factory” is based on the principles of the flexible SCRUM methodology of software development and the implementation of the SMART methodology for setting goals in IT startups. During the work, real orders from specific IT companies and partner organizations are used for developing digital products. The path of a professional. ” In the course of the work and the studies, participants actively use such popular programming languages as C#, Python, Unity, Java and technologies that are in demand in the global IT industry. Participants also develop their own elements of the corporate identity of educational “IT firms” (name, logo, slogan, trademark, etc.), make independent decisions on choosing projects from the set of real tasks of partner companies, select methods and tools for their design “packaging” using 2D and 3D design and UI / UX design, as well as tools for promoting and advertising software products using various means of Internet marketing and the creation of advertising materials (landing pages, commercials, teasers, etc.).

At the same time, throughout the entire intensive session of the School, participants not only attend relevant lectures and expert seminars, which are conducted in an interactive form, but also put the knowledge gained into practice, and implement their projects under the supervision of IT curators. Curators are not managers for training "IT firms". They do not solve all the important tasks of designing, developing and promoting digital products instead of their students. Their main mission is to help participants independently achieve their goals, to help them with the selection of the necessary tools and techniques, to explain the erroneous and successful decisions in the implementation of their projects, to reveal the important secrets of professional competencies in practice. The results of the work of the School participants under the guidance of curators are demonstrated to representatives of organizations during the defence of projects. It is partner organizations that order software products before the beginning of an intensive session of the School "Programming Factory" and assess finished products at the end intensive session.

The most important result of participation in the project “Factory of programming. The path of a professional” is a professional test of students and teachers in the digital economy, which begins during the work at intensive sessions of the School and defences of projects aimed at developing software products by order of the partner organizations of the “Factory of Programming”. The professional test continues during the internship at the partner organizations of the School motivated to hire some of the interns.

The internship of the best participants of the School “Programming Factory” begins with a preliminary interview, during which the representative of the partner organization finds out the level of professional training and erudition of the candidate, career guidance of the candidate based on his experience and the results of project work at the School, learns his personal preferences and potential for professional development. Further internships are planned taking into account the project work schedule in the partner organization where the internship is held in the full-time or remote format under the supervision of the School's curators and experts.

Based on the results of the project work during the internship, the internship leader, together with the curators and experts of the School, give the intern recommendations for further professional development. The curators and experts of the School also offer a further training program and together with the internship manager from the partner organization determine the conditions for further professional cooperation with the partner organization up to possible employment of intern or launching his or her own project.

Thus, teachers are simultaneously involved both in the direct creation and solution of the development and business tasks of software products development and in the methodological pedagogical workshop. The purpose of the workshop is to create a project of a cycle of training sessions for the organization of extracurricular activities or sessions implemented into the program of the course.

The basic content of the first session of the methodological pedagogical workshop is associated with the difference in the learning situation and the production situation of the development of a software product. Teachers get an idea of the difference between “soft” and “hard” skills when creating a software product in an educational situation. “Soft” skills are associated with self-organization when working within a team, initiative, creativity, responsibility for solving part of the tasks of the project team, etc. "Hard" skills are more associated with the development of specific methods of programming, promotion, design and management of software products. At the workshop, teachers also discuss ways to monitor students' personal changes (“soft” skills).

At the second session of the methodological workshop, teachers reconstruct the stages of work on the creation, design, promotion and organization of a software product. The greatest difficulties for teachers at the same time are working with the content of the concepts “scrum”, “sprint”, “market”, “target audience”, “business processes” etc., due to the lack of practice in conducting logical-subject reconstruction, in fixing the subject of actions when developing a software product.

The content of the third block provides teachers with the opportunity to design a project of their own series of training sessions on the development of software products. When working in groups, they need to perform a logical-subject analysis of one of the proposed production situations in which the development of a software product is required, and to suggest the wording of the assignment for students, as well as describe the differences in educational and pedagogical reality: the position of the student and the teacher-leader of the group of students in the studying process of the lesson. The separation of the pedagogical position (what does the teacher do?) And the teaching (what does the student do?) allows the teacher to become an observer in relation to the ongoing process of developing a software product and manage it, to perform a “reflective function”, and in some cases, to start their own pedagogical research.

The first three blocks create a basis for describing the content of pedagogical support when developing a software product. In addition, there is a basis for developing the content of the educational program of extracurricular activities to develop software products at the level of an educational institution. Communicating to participants about federal and regional requirements for the content and outcome of educational programs allows educators to complete work on the curriculum of the courses. The key parameters for assessing the quality of the curriculum are the description of the content of the task to be solved by students, ways to manage communication between groups of students, the description of how to observe and record personal and objective results, and the quality of management of the software product development.

One of the success factors of intensive courses on the development of software products and related up-skilling courses for teachers is the completion of each participant’s qualification test - designing individual projects, which include: a project for designing a software product and an educational program on the basics of project activity, focused on working with a group of students to develop software products.

Based on the results of intensive sessions of the School “Programming Factory”, regular surveys of school participants (schoolchildren, students and teachers) are also conducted. In particular, at a number of recent Schools, it was found that very few participants ever took part in the real process of development or in training in software products development, even educational products.

For schoolchildren, the IT field is represented by bloggers, all who support the work of the Internet, and programmers. Schoolchildren have an idea that IT professionals work online, among separate areas they distinguish SMM, programmers, web designers, most often they designate “IT people”. This category of respondents notes that work in IT is in demand now (70%), but they see prospects for themselves in other areas of employment.

“Say, of course, I heard about it. I spend a lot of time on the Internet, using technology. I didn’t really go deep into how it works, how it develops. I’m, let’s say, a person who uses the products of the sphere, and does not develop them. ”

As for the survey of students at universities and colleges, the majority of students (90%) already identify the field of IT technologies as the most popular and significant, and, in their opinion, it will remain the same in the next 5-10 years. Along with the IT sphere, the significant areas noted by students are medicine, teaching, and bioengineering. 85% of respondents are interested in gaining competencies in the IT field, but no one has taken targeted courses and so far does not connect their future only with the IT industry.

What kind of specialists are needed to create software products, according to School participants’ opinion? The following response of the respondent is illustrative, revealing the widespread attitude to the IT sector as a team activity: “Yes, diverse, in fact. In general, to make high-quality content, you need various specialists. Only a professional team can record any video, produce professionally beautiful high-quality content on a virtual platform, for example, on YouTube. And this is not always related to IT. The person who sets up the light on the site, yes, let's say a video for the same YouTube, is not connected with IT but is also part of this large team”.

“First of all, an idea is needed, and then specialists are needed: programmers, testers, designers, advertisers who create the product and then promote it. Projects live through many stages: creating an idea, developing, debugging, during which many changes take place, and already at the output, we have a finished product. ”

“To begin development, a technical task is defined in which the main functionality is determined, after which a development team is determined, the size of which depends on the team’s skills.”

Teachers note that along with the demand for teachers, doctors, service specialists, now and in 5-10 years perspective, first of all, IT specialists will be in demand. When describing how specialists work when developing software products, they list specific competencies. For example, “specialists who could analyze very well, understand and know a lot because when you try to advertise a certain area, I think that you need to understand it, and you need to be able to analyze for this, to highlight the key points”, “specialists who can write and edit good texts, I don’t know, people related to computer programs”, “people who have been trained well in computer science, respectively, are aware of various algorithms well, even mathematical formula, and should understand electronic circuits, how, what, why, where and what kind of processor is needed”.

Thus, on the one hand, in the responses to the survey of the School’s students, one can trace high motivation for mastering the competencies of developing software products and a shift in focus towards development of knowledge and skills useful for a wider range of professions, not only in the IT field. On the other hand, teachers demonstrate a greater commitment to the present tasks than to future tasks, and the motivation to improve their skills and apply knowledge and skills directly in the IT industry.

4. Conclusion

The study showed that despite the popularization of “digitalization” and the practice of using IT products, training teachers in organizing software products in Russia and in the Krasnoyarsk Kray is not enough. It is impossible for a teacher who does not have his or her own experience in participating in business processes to create software products to teach students modern demanded competencies of working in a developers’ team. In this regard, the fastest and most effective, least expensive way to train teachers is to simulate teamwork in developing software products – in parallel with the learning processes of students.

A theoretical analysis of the activity on developing software products allowed us to describe the functional tasks of teachers who organize training in software products development – organizing and managing communication in a group of students, a logical-subject analysis of the material, and the management of training students. As it turned out, it was these tasks that caused the greatest difficulties for teachers in practice, as well as in pedagogical activity in general.

The practice of coordinating the development of software products requires teachers' up-skilling courses. The activity aspect of these courses is ensured by:

1. The content of courses and assignments, which are designed according to the tasks that determine the position of the teacher, focused on the development of software products.
2. The position and actions of the course leaders who actively use the existing competencies of teachers in programming in various languages, the experience of creating websites; designing educational programs in computer science and programming.
3. Implementation of methods for collaboration of participants, ensuring the productivity of joint interaction and self-realization of participants in the educational process.
4. The orientation of teachers towards the development of an author's product — a series of classes embedded in an existing program or the creation of a holistic educational program, the purpose of which is to organize students' experience in developing software products.

The form and content of up-skilling courses contributed to the successful passage of qualification tests in the form of development and design of products by studies participants. Students and teachers presented their products developed to potential employers and customers, and teachers developed methodological materials for implementation in their teaching activities. Note that thanks to the built-in communication at the personnel school, teachers were able to offer a number of significant proposals, which were reflected in the work of the organizers of intensive training schools in the development of software products.

References

[1] IDC FutureScape: Worldwide IT Industry 2019 Predictions. Report US44403818 https://www.idc.com/getdoc.jsp?containerId=US44403818, last accessed 2020/02/10

[2] Sakharova V I 2014 Training of a research teacher in a continuing education system *Vocational education in Russia and abroad* **4(16)** 106–9

[3] Leontovich A V 2018 The Approaches to the assessment and examination of student research or design work *Researcher* **3 (23)** 118-22

[4] Popov A A, Ermakov S V and Remorenko I M 2015 Project "Assessment of competency-based results a achievements" *Collection "Open education as a practice of self-determination"* (Moscow, Russia: Non-profitpartnership "Author'sClub")

[5] Bazhenova K A 2014 The model for training teachers to guide the research activities of students *Pedagogical magazine of Bashkortostan* **2 (51)** 84–9

[6] Remorenko I M, Elkonin B D et al. 2018 Activity-based approach to education Non-profit partnership to promote scientific and creative intelligentsia in the integration of world culture (Moscow, Russia: Non-profitpartnership "Author'sClub")

[7] Yudina Y G 2007 Pedagogical support for the formation of individual educational trajectories of students (Krasnoyarsk, Russia: Siberian Federal University)

[8] Shchedrovitsky G P 2015 Organizational thinking: ideology, methodology, technology (lecture course) (Moscow, Russia: Moscow Publishing house of the studio Artemy Lebedev)

[9] Tsarev R Y, Yamskikh T N, Evdokimov I V, Prokopenko A V, Rutskaya K A, Everstova V N and Zhigalov K Y 2019 An approach to developing adaptive electronic educational course *Advances in Intelligent Systems and Computing* **986** 332-41

[10] Bazhenova K A 2014 Designing educational situations for bachelor students in Pedagogics *Moscow University Bulletin* **3**73–83

[11] Brhel M, Meth H, Maedche A and Werder K 2015 Exploring principles of user-centered agile software development: a literature review *Information and Software Technology* **61** 163-81
[12] Bazhenova K A and Znamenskaya O V 2017 The level model of subjective action acquisition on the example of programming *Computer Science and Education* **5** 18-22