

UDK 378.14

Vladyslav Kruhlyk⁵Melitopol State Pedagogical University named after Bohdan Khmelnytskyj,
Melitopol, Ukraine**FORMATION OF PROFESSIONAL COMPETENCES OF FUTURE
ENGINEER PROGRAMMERS IN THE PROCESS OF INDEPENDENT
EDUCATIONAL ACTIVITY**

DOI: 10.14308/ite000633

The article is devoted to the issue of the organization of independent educational activity of future engineer-programmers while studies at universities. The organization of students' independent activity is an effective mean of improving the quality of the professional training of future engineers-programmers at higher education institutions. Such activity foresees handling of the complex creative tasks, connected with the real practice of programmers' work at the enterprises. It is aimed at forming professional competences in students, readiness for further self-improvement, ability of making decisions and accepting the responsibility for their consequences, identifying of mistakes, finding the ways of their fixing and minimizing negative consequences, reacting adequately to professional problems. The teachers face the task of choosing the directions, means and content of independent work in such a way that it would be effective and would encourage students for professional activity.

In the article, such methods of organization of independent work for students are analyzed: execution of diploma projects, scientific-research activity, participation in professional olympiads and contests, going through industrial practices. The description of the stages of execution of the diploma projects and the content of the teacher's and students' activity on each of them is provided. Special attention is paid to teamwork on the project and the usage of information technology for organization of such interaction. The effects of usage of students' independent work is characterized during industrial practices, participation in olympiads and contests, conducting research work.

Keywords: *higher professional education; engineer-programmer; diploma project; industrial practice; professional olympiads; independent educational activity.*

1. Introduction.

The definition of an issue in general view and its connection with important practical tasks. The providing with high-quality training of the specialists for the field of information technologies is the major task of higher education institutions that train future engineers-programmers. The authors of numerous researches about the issues of modern IT education, in particular [1] and [2], emphasize this and offer various ways of fulfilling this task.

In this regard, we consider the organization of effective independent educational work of students to be the extremely important direction of the teachers' activity throughout the whole term of studies at the university. To our opinion, the very creative research work facilitates the formation of students' professional competences, the ability of their usage in the conditions of the real IT industry, as well as readiness for further self-improvement and development. During such activity, the student faces the necessity of making his own, substantiated and responsible decisions which are going to influence the further process of work, as he is not supervised by a teacher permanently, is not controlled by him.

The mistakes, which are an integral part of this process, are also independent. Thus, the student learns not only how to apply the received knowledge, but also to make decisions and to accept the responsibility for their consequences, to identify mistakes, to find ways of their fixing and minimization of the negative consequences, to react adequately to professional problems.

Therefore, during the independent activity, the practical experience necessary for further professional activity is formed.

⁵ © Vladyslav Kruhlyk

The analysis of the researches and publications in which the solution to this issue was initiated. The issues of effective organization of independent educational activity of students are covered in the works of many researchers, namely: in the field of professional training of specialists (Asherov A. T., Zyazyun I. A., Savchenko O. Ya., Sysoeva S. O., Shkil M. I. and others); in the field of informatics and methodology of informatics (Bykov V. Yu., Velikhov Ye. P., Glushkov V. M., Gurzhiy A. M., Ershov A. P., Zhaldak M. I., Lvov M. S., Monakhov V. M., Polat E. S., Ramskuy Yu. S., Tseitlin G. E., and others); in the field of methods of teaching of informatics (Apatova N. V., Bilousova L. I., Verlan A. F., Kasatkin V. N., Klochko V. I., Lapchik M. P., Makarova N. V., Morse N. V., Rakov S. A., Spivakovskuy O.V., Trius Yu. V., and others); in the field of actual issues of computer engineering and training of engineer-programmers (Dening P., Knut D., Morozova T. Yu., Osadchuy V. V., Papper S., Seidametova Z. S., and others).

The formulating of the goals of the article (definition of the task). The aim of the article is to determine the main directions of organization of independent educational activity of future engineer-programmers during their studies at universities. For this, such directions of work will be covered: execution of diploma projects, scientific-research activity, participation in professional olympiads and contests, going through industrial practices.

2. The results of the research

2.1. *The independent educational work of future engineer-programmers in the process of execution of diploma projects.*

The diploma projects of future engineer-programmers are often executed as program projects, meaning, the result of work is the research on the specific issue and the prototype of software product. The execution of the diploma project has the significant limitation: shortage of time (less than 10 months). For the large projects, this issue is settled by the creation of the project team - several students are working on different aspects of the issue, creating one product in result. Such execution will demand both additional attention of the scientific supervisor (teacher) and high responsibility from students.

During the execution of the work, students should get maximally close to the real project process, so the project should foresee the necessity of usage of modern technologies, integration of data or services with side developments, design of architecture, organization of interaction between team members, etc.

We identified and characterized the main stages of work on the diploma project, namely [3, p. 91-93]:

1. The choice of the theme of the work. The theme of the diploma project should meet such criteria: actuality; the theoretical or practical significance; the personal interest of students. The themes of works should be coordinated according to the level of complexity and independence.

2. The development of the project concept. At this stage, it is necessary to thoroughly analyze the needs of users for the satisfaction of which the software product is created. Besides, the comparative analysis of existing systems should be carried on with the aim of determination of the parameters of the program that will maximally fully satisfy the requirements and expectations of the targ

3. The organization of work in the group. At this stage, it is necessary to identify the means and approaches that will be used further on in the process of project development, in particular: for the distribution of project tasks and responsibility; for control of the readiness of separate modules and their integration; for organization of common work on the project; for adjusting of co-operation and communication between project participants. Besides, the roles of the individual team members should be defined: leader, tester, designer, interfaces designer, etc.

For the control of the process of the project development can be used free systems, in particular:

- for monitoring and discussing of tasks - Redmine;
- for organization of chats and making common calls - Skype;
- for storing of codes with the opportunity of shared access - Git based on Bitbucket;
- for conducting of design documentation - Wiki;

- for tracking of project errors - Bugzilla.

4. The choice of development methodology. It is reasonable to execute the diploma project on the basis of Agile methodology, creating iterations at the 1st-2nd week, depending on the complexity of the tasks. This approach allows to overcome the main disadvantage of the generally accepted waterfall model - the lack of time for testing of the developed program, fixing of mistakes and creation of a version suitable for implementation.

5. The work on the product interface.

6. The work on the design of the software product.

7. The choice of the development technologies. The aim of this stage is to choose technologies which allow to fulfill the task of maximum quality. In the process of choosing, it is appropriate to pay the main attention to the technologies and tools that are freely available.

8. The work on the quality of the software product. The quality control should be carried on throughout all the software development process. With this aim, it is reasonable to use UnitTests. Besides, it is advisable to draw an experienced programmer to revision of the code.

9. The completion of the project. The completion of the diploma project is the submission of the finished software product and reporting documentation for the presentation.

10. The transfer of the project. The fulfillment of some tasks, which is brought in the diploma design, needs more time than the usual term expected by the curriculum. In such cases, it is reasonable to organize the continuation of work on the project next year with another performers' staff. In connection with this, the problem of transfer of software code and documentation arises. For its settling, information about the project should be preserved using the version control systems, Vicky, etc.

11. The support and continuation of the project. If the project continues and the data transfer is carried out qualitatively, then at the beginning of the project, the participants should study the existing documentation and code. After that the project can be executed according to the usual scenario.

Since the real industrial software development requires the ability of work in a team now, it is necessary to form the experience of the collective work on projects among future engineer-programmers, in particular under the conditions of work on a diploma project. We believe that it is worth for students to understand that software development is more than just typing the program code. It is suggested to use additional tools for this: task and time management systems, bug trackers, project management systems, version control systems, professional development environments, etc.

Let's describe in what way it is possible to use such software means for organization of work on the diploma project.

With the help of Redmine [4] you can monitor the tasks and their discussion. It is a free server web application for managing of projects and tracking of errors. The system includes calendar scheduler, Gantt Chart for visual presentation of project progress and terms of fulfillment, there is an opportunity to use Wiki and forums for each project. The system provides such possibilities: running a few projects; usage of flexible system of access with the use of roles and error tracking; conducting of project news, documents and file management; realization of notifications about the changes via RSS feeds and e-mail; assessment of time spending; adjustment of your own fields for tasks, time spending, projects and users; integration with version control systems (SVN, CVS, Git, Mercurial, Bazaar and Darcs); creation of notes about errors based on received emails; availability of multilingual interface (including Ukrainian language); support of database management system MySQL, PostgreSQL, SQLite.

For the beginning of work with the system Redmine, the students should create the project. This can be done by the user of the system with administrator's rights. Then the students should be added as users. It is necessary to add several registered users to the project and activate them. After activation, the teacher can assign them one or more roles. By default, such roles exist: manager, developer, report generator. These roles influence what each user can do in the frames of each project. It should be noted that the role assignment influences the permission in two different plans:

1) the role will influence the solution of all aspects of the project, for example, the role of "manager" allows the user to create new projects, manage the forum, wiki, repository, etc. mainly within the project frame; 2) the role influences the sequence of actions on the task (new - in process - resolved - closed - denial), for example, the difference between the manager and the developer is that the Manager is allowed to put the status "Reject" on the task, but the Developer is not allowed.

The next step of the work of students with the Redmine system is the creation of tasks. This can be done by a teacher (administrator) or by a student, who is assigned to the role of the manager. It is advisable to draw the attention of future engineer-programmers to the correct filling of the task fields: the type of the task, the title, the description, the priority, the customer, the aim, the deadline of the task fulfilling, estimating time of the task.

Such types of tasks are available in the system:

- Bug - error in the product functioning, false product behavior; is assigned to developer or tester;
- Feature - new improvement, functional, new logics of product functioning;
- Task - the technical task assigned to the immediate executor;
- Research - the task of studying of the possibilities of realization of a new functional;
- Request – the informational request from the project participants;
- Idea – the description of the new idea, is assigned to the manager, supervision;
- Discussion – the discussion within the project is also the type of tasks for internal, non-project tasks.

The title of the task is a short, unique description necessary for a quick search of the task in the system. In the "Task Description" (description) field, the task is described in detail: how it should be done, how the completed task should look like, and so on. The specification of the priority of the task shows its importance compared to other tasks in the current project, which influences the order of execution: the higher the priority, the faster this task will go into development. In the field "Customer" is pointed out the one who is interested in the completion of this task and ordered its execution. He will check the result of the work and evaluate its performance later. "Goal" is the final goal which the completed task should solve. In the field "Deadline" the task manager indicates the date of its delivery to the customer. "Estimated time" - the estimated time of the task completion (is filled in by the executor or a team of executors in the "perfect hours", meaning, how much time any task takes compared with others).

The process of students' work on tasks happens in this way. Initially, the task has the status "New" and is assigned to the developer, who at the beginning of the work transfers it into "In Process" status, and then to the status "Resolved". The manager checks the received decision and transfers it to the state "Closed" or back into one of the previous states. If the record will not be resolved for any reason - the task receives a status from the manager "Declined".

Redmine provides an opportunity for the teacher to see the brief summary of tasks which displays the statistics about closed, open, and all tasks in different blocks according to trackers, priorities, users to whom the tasks are assigned, the authors of tasks, versions, categories (Fig. 1).

#	TRACKER	STATUS	PRIORITY	SUBJECT
382	Bug	Assigned	Urgent	Add confirmation on delete notes
119	Development	Assigned	Normal	Create console to send emails to users
981	Development	Assigned	Normal	Make site responsive
188	Feature	Assigned	Low	Search in notes by key words
380	Bug	Assigned	Low	Button "Print" prints page 3 times

(1-5/5)

Fig. 1. The summary according to the students' tasks in Redmine

The usage of the Gantt chart in Redmine allows to display the tasks that have the date of the start and the date of the completion or are connected with the version with a date [5] (Fig 2).

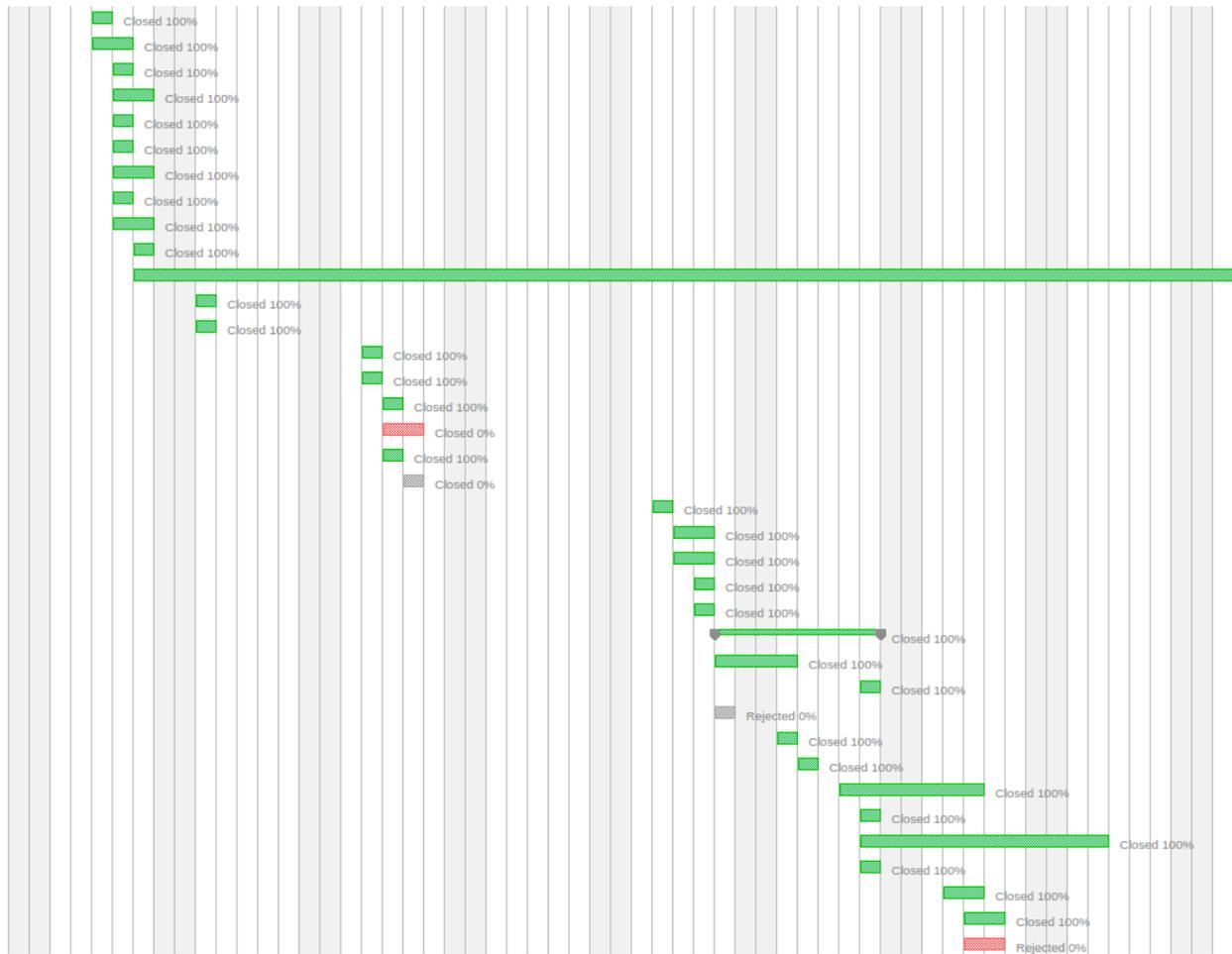


Fig. 2. The usage of the Gantt chart in Redmine

As pointed out in [6] and [7], the usage of Git and GitHub helps not only to track changes in versions, but also with the help of practice-oriented methods to show students the methods of organization and collaboration on the projects. Github [8] is positioned as a web hosting service for projects with usage of the Git version control system as well as the social network for developers. The users can create an unlimited number of repositories for each of which Wiki is provided, the system of tracking usage, there is the possibility to held the review of the code and cooperate on the project, etc.

The teacher, leading several diploma projects of future engineer-programmers, is capable to organize the process of the development in such a way to use maximally these online services. This is reasonable because they provide such opportunities for the teacher and students: to publish tasks and assessments; to review students' listings (program codes), to control the time of their placement and authorship; to set the dead-line, meaning, to prohibit from the definite moment registering of works; to create comments not only to the whole project, but also to separate lines of the program; to point out the drawbacks in the work and place instructions for the tasks completion; to use your own and common repositories at any moment; to receive messages about the work check and the comments of the teacher automatically.

The important feature of Git is that you can work on a local computer with the periodic update of the repository (synchronization) on GitHub. The basic element of the Git-repository is the commit (commit) - the user-registered state of the repository. It is possible to return to any of the commits and see changes and additions in comparison with other commits. This allows the teacher

to point out the drawbacks and errors directly in the program text. Using GitHub, it is easy to find changes in the code, inserted for fixing of previous errors. This allows not to look through each student's data-review once again, but see only those parts that have been changed (Fig. 3).

```

394 394     $getOriginalResult = $s3Storage->getFile($sourceFile, $TmpName);
395 395     } catch (\Exception $ex) {
396 396         $dbLogs->log('imageProcessing', 'Error while resizing image. Could not get file from s3 storage!');
397 + @unlink($TmpName);
397 398         return false;
398 399     }
399 400     if ($createFolder && $getOriginalResult) {
426 427         unlink($TmpName);
427 428         return $result;
428 429     }
429 - @unlink($TmpName);
430 + @unlink($TmpName);
430 431     return false;
431 432 }
432 433

```

Fig. 3. The presentation of the changes in the program code GitHub

The methods of work with Git requires from the teacher such actions: to clearly formulate the task; to put this task to the developer (student); to check student's understanding of the task; to check periodically the result of the task completion in the course of the student's work over it; to return the unsatisfactory result to the student for correction; to note the satisfactory result; to carry out the final check of the software product developed by the student (students); in case of execution, to close the task as completed, otherwise send it for correction.

The functions of the student: to receive from the teacher the task; to proceed to its execution; to present the result in the system; if the task is returned for correction - to elaborate it and send it again in the system.

For the management of tasks Redmine is used, where the documentation on the project is conducted, both about the progress of the software tool development and for the user of the developed software tool that another student can fulfill.

The algorithm of work with Git and the general picture of what is happening in the process of the students' and teacher's work over the project.

The work according to this algorithm is conducted in this way:

- 1) the teacher sets the task for the student (the developer);
- 2) the student is working on the actual branch of the project and makes it a local branch;
- 3) the student solves the task in this branch;
- 4) the student sends the branch with the completed task into the working repository;
- 5) the teacher takes this branch from the working repository and checks it;
- 6) if the task is fulfilled correctly, the teacher combines this branch with the actual branch of the project in the main repository or allows the student to do this.

In this way, the future engineer-programmers learn to work with such technologies which they can use in future professional activity.

As pointed out in [9], due to the problematic features of the modern educational process, such as the use of the outdated model of Waterfall (the sequential method of software development, called so because of the diagram similar to the waterfall [10, p. 58]) in the development of programs, the limitation of academic programs in time, bureaucratic processes in changes with an average cycle of 3-4 years, insufficient attention to the needs of the modern labor market and too large fragmentation and differentiation of education, it is necessary to apply progressive approaches and to organize the process of project management by means of the introduction of modern forms and methods of training.

Agile methodology, in contrast to the Waterfall model, allows, on account of an interactive analysis of requirements to start a project execution quickly without spending of resources on planning and collecting of requirements; on account of short iterations and constant feedback to get

the product that maximally meets the requirements; on account of the maximal aiming of the team on the result and simplification of the formal side of the process development management to obtain the result in a short time; on account of usage of certain engineering practices to receive the result of satisfactory quality [11, p. 347-348].

The special feature of this model lies down in fixation of the successive processes of the software product development. The model of the factory is laid down in its foundation, where the product goes through stages from the plan to the production, then it is passed on to the customer in the form of the final good, where the replacement is not expected, though another similar product can be given [10, p. 58].

The basic principles of usage of Agile in the studies are the following [12]:

1) the satisfaction of needs: the main priority of the system of education should be meeting of the students' needs by the way of providing meaningful education;

2) the dynamic adaptation: willingness to requirements changes even during the educational process in favor of students;

3) the decomposition of the educational process: the division of the curriculum into smaller logical blocks with the duration of a few days to several weeks;

4) the common work on the general goal: the representatives' work of the educational system, the families of students and the students themselves as one team;

5) the providing of the necessary conditions: the creation of projects based on motivated people, providing them with conditions and support their needs, confidence in the independent fulfillment of the task;

6) the live communication: the most effective method of information transfer in the team is the live communication between its members;

7) the meaningfulness: the main measurement of progress is the meaningfulness of education;

8) the duration: the teachers and students should have a possibility of maintaining of the steady pace of education long enough to ensure the sustainable development;

9) the self-organization: the best ideas and initiatives arise in teams that can organize their cooperation independently.

Consequently, the offered approach to the execution of diploma projects helps students to acquire and strengthen key IT competencies. As the content of the diploma project is maximally close to the real tasks, students get the opportunity to go through all the typical stages of development of the commercial product. We note that students who have fulfilled the projects actively at university, later take the key positions in IT companies. At the same time, the organization of the project activity, in particular diploma design, has a number of drawbacks: quality, development processes, documentation, publication, optimization, texts, etc. are often left without attention. The processes in teams of students are often built chaotically too.

2.2. The organization of independent scientific-research activity of students

It should be noted that in the work on the diploma project it is necessary not to leave without attention the formation of the research skills of the future engineer programmers. The studying within the frame of the discipline "Introduction to the specialty of the programmer" the basics of scientific activity and strengthening the skills of work on research during the study of the discipline "Fundamentals of scientific activity in the field of computer science" (7th semester) allows to prepare students for this activity. In the result, the future engineers-programmers form research skills, namely: to analyze information, to work with primary sources; to determine the issue and formulate hypotheses for its solution; to generalize, systematize and draw conclusions; to choose the optimal algorithm for completion of the task; to define subtasks in the global task; to compose the necessary informational product from available ready standard blocks; to determine the quality of the software product; to present the data in different ways, to choose among them the most adequate way of presentation of the research results.

The involvement of the future engineer-programmers to the scientific-research activity in the process of their professional training helps for extension, generalization, systematization of their

knowledge, development of the scientific and practical competences of students, formation and development the scientific potential of the department, preparation of competitive graduates who are capable to become the full-fledged scientific and labor potential of the country.

We consider the popularization of the scientific-research activity of the future engineer-programmers to be important, which is reasonable to accomplish the stimulation of students by such means, as rewarding diplomas of honor, diplomas or monetary prizes for the most significant scientific achievements. Besides, no less important factors exist that can motivate students to participate in the scientific-research activity, namely: the possibility of improving their qualifications; the acquiring of practically useful knowledge and skills; the connection with future employment; the expanding of the access to computer technics, equipment and means of communications; the addition of the project to your personal portfolio.

The participation of the future engineer-programmers in the scientific-research activity allows to develop the competencies necessary for the completion of the applied tasks in the future professional activity. For this it is reasonable:

1. To set goals clearly.

In the process of the scientific-research activity, the student can determine the theme and the task of research both independently and with the help of an experienced teacher.

2. To identify the method of the goals achievement.

The work with sources and literature, communication with like-minded people and fellow-workers of the department helps the student to choose the successful approach to solving of the issue, to find the necessary tools and technology for realization.

3. To monitor and manage the realization.

Being the 4th year student, the young researcher takes over the experience of the teacher who gives him the tasks and controls the quality of the performed work.

4. To analyze the threats and counteract them.

By fulfilling of the task, the student gets acquainted with the typical risks of IT projects and the methods of reacting to them. The experience of the tasks fulfilling is accumulated, the knowledge base about the risks is formed, the basic experience of risk management is acquired.

5. To create a team.

While solving the modern issues in the field of computer science individual work is practically impossible to be enough. By conducting of the research in the laboratory, the student learns to work and cooperate in the consolidated team whose members are united with the common deal. Participating in the development, the student can try himself in different roles (developer, designer, test manager, manager), which is going to give him the opportunity to adapt quickly to executing of the complex practical tasks in the future professional activity.

Thus, the participation in the scientific-research activity, besides the development of scientific and research abilities, helps to train the specialist, ready for the applied issues execution with usage of the modern means and approaches to the software development.

2.3. The formation of professional qualities of the future engineer-programmers in the process of participation in professional olympiads and contests.

The olympiads on informatics and programming are one of the important directions of work with gifted youth, the effective means of stimulating their interest to computer modeling and the development of algorithmic thinking as versatile cognitive tools in students. The olympiads are called to form and develop the collaborative skills and teamwork; the ability to perform the standard and non-standard tasks of the professional activity; the ability to use the methods of functional, logical, object-oriented and visual programming, as well as various tools for programs development; the readiness for the logical design; the ability for self-organization and self-education.

The fulfillment of the olympiad tasks foresees the effective use of the skills acquired in the process of the professional training, in particular, such as: execution of decomposition (splitting of the task into smaller subtasks); optimization (the search of the best structures and objects for the

realization of the fast and more reliable algorithm of performance); visualization of the solution (usage of the software products for the graphical display of algorithms); work in a team (generation of common ideas with the following division of functions and completed tasks).

During the olympiads, the students can try themselves in different roles (programmer, algorithmic, mathematician); get acquainted with real projects (including puzzles) and clear criteria of evaluation of the results. It is valuable that the participants of the olympiad can express their thoughts and offer the realizations for the set tasks. Thus, in the process of participation in the olympiad, students not only improve the special skills and abilities, but also develop the important social qualities: communicative skills, ability to cooperate and design their activity. The results of the olympiad can be valuable educational material that can be immediately used in the educational process.

As Krajvanova V. A. and Kryuchkova Ye. N. [13] properly point out, the usage of olympiad approaches in the educational process allows to increase the level of readiness of the future engineer-programmers for professional activity at the moment of graduation from the university. The olympiads allow to realize the transition from training according to the standard curricula to expansion of the educational component, extension of the number and increase of the quality of mastering of the basis and basic skills of the programmer's profession. This is achieved due to such features of the olympiad movement:

1. The olympiads produce the clear style of programming, the ability to test programs, see possible mistakes.

2. Team work during trainings and competitions facilitates the working out of the methods and skills for collective work execution.

3. Success aiming is one of the undisputed values of the olympiads. It is the identical for everyone: both for those who did not win and for the champions. The students learn to work so that in the future they would be able to create their own team and make it successful.

4. The olympiads develop the abilities to scientific and technical creativity, independency, ability to orient themselves fast in different situations.

The separate kind of the independent educational activity of the future engineers-programmers are contests. As pointed out in [14], in the frames of the computer programming competitions, it is possible to reproduce separate professional skills and abilities with the help of modeling specific tasks. And in the work [15] it is noted that competitions allow to popularize the theory of programming. About the fact that the contests are the special discursive practice that influences the formation of the professional community is mentioned in [16]. On the basis of what, the following features of the competition are singled out as the discursive practice of professionals: the aim of the competition is to facilitate the development of the professional discipline; the contest has an educational character, setting the direction of growth of the professional community; the involvement of the targeted audience; the competition is built around the principle of the expert knowledge and the demonstration of professionalism.

2.4. The formation of professional qualities of the future engineer-programmers in the process of going through industrial practices.

Not only the work on a diploma project, participation in competitions and olympiads realizes the practice-oriented approach to the professional training of the future engineers-programmers. It is also carried out in the process of going through industrial and pre-diploma practice by the students, which is realized in the conditions of real practical activity connected to the development of applied software. The aim of the practices is the actualization of the professional competences acquired during the training, in particular, on the development of the software product.

The aim of the industrial practice is the formation of the professional practical knowledge in students, skills and abilities necessary for the successful work in organizations and enterprises

which run software, use the informational methods and computers for the modeling of industrial processes.

The aim of the pre-diploma practice is the application and consolidation of the knowledge and skills of students received at higher education institutions during the whole process of the professional training; the testing of the students' capacity for independent work in the real conditions of an enterprise or scientific-research subdivision; the collection of material for the diploma paper.

For achievement of the aim of the pre-diploma practice, the future engineer programmers are directed to one base of the industrial practice for problems identification, settling of which demands the development of the software product, and the preparation of the technical task for its development, according to which the work on the diploma project is carried out.

In the process of going through the industrial practice, the students work on the specification of the separate components of the software, create the software product code on the base of the ready specifications at the module level, set up the program modules with usage of the specialized software tools, test the software modules, accomplish the optimization of the software code of the module, develop the components of the design and technical documentation with usage of the graphic language of specifications, analyze the design and the technical documentation at the level of interaction of the components of the software, perform the integration of the modules into the software system, etc. All this allows to understand the essence and social significance of the future profession, to learn how to organize own activity, to identify the methods and means of the performance of professional tasks, to evaluate their effectiveness and quality, and to realize the search, the analysis and evaluation of information necessary for the setting and fulfillment of the professional tasks, the professional and personal development, to learn how to work in the collective and team, to be prepared for the change of technologies in professional activity.

The work of the future engineer-programmers during going through the pre-diploma practice should contain: 1) the study of the modern information technologies, the mathematical methods, the software and hardware according to the subject of practice; 2) the conducting of the scientific research with the aim of the improvement and simplification of technologies, search of new approaches and methods for the completion of the considered tasks; 3) the conducting of the calculating experiments in comparison with efficiency of the used and offered informational technologies, methods and algorithms; 4) the carrying out of the scientific researches and experiments on the diploma paper.

While going through the pre-diploma practice, the students study [17, p. 6]:

- 1) the organization of work in the enterprise;
- 2) the mathematical methods of the financial activity, which are used at the enterprise, department;
- 3) the state of the modern information technologies and their usage in different spheres of activity;
- 4) the equipment, devices, electronic computing outfitting, which is used in the department;
- 5) the experience of the leading specialists of the department;
- 6) the marketing in the field of the production of software products and informational services.

While going through the practice, the students develop and explore:

- 1) the mathematical models of the computational mathematics, management theory;
- 2) the informational systems in general and their separate modules;
- 3) the informational technologies and software for the completion of the received tasks;
- 4) the automated management systems in general and their separate modules;
- 5) the databases and database management systems;
- 6) the computer networks, Internet and Intranet technologies.

6. Conclusions from the conducted research

The professional activity of the engineer-programmer in the IT industry requires the availability of various competencies that must be formed in the process of the professional training in a higher education institution. In this regard, universities face the task of creation of such conditions when the student works deliberately with the aim of obtaining of such competencies. Among the many ways of achievement of this aim, we distinguish the independent work. We note that the leading role is played by those its varieties which are aimed at the creative, searching activity. With this aim, it is necessary to offer the students difficult tasks that demand the scientific search and the theoretical analysis (scientific-research work), the collective or independent work for a long period of time for the creation of the finished software product (training projects, diploma design, industrial practices, professional contests), the completion of the complex tasks with the usage of non-standard approaches (professional olympiads and contests).

Thus, such types of the independent educational activity of students as the scientific-research activity in the course of execution of the diploma paper, organized on the basis of the project method and team work, the olympiads and contests, the industrial and pre-diploma practices, which also facilitates the development of teamwork, have the important meaning in professional training of the future engineer-programmers, assisting the development of all components of the professional competences in the conditions of the quasi-professional activity.

REFERENCES (TRANSLATED AND TRANSLITERATED)

1. Osadchyj V. V. The analysis of the problem of the professional training of the programmer and the ways to solve it [Online] / V. V. Osadchyj, E. P. Osadchaja // The international electronic magazine "Obrazovatel'nye Tehnologii I Obshestvo (Educational Technology & Society)". – 2014. – Vol. 17, no. 3. – pp. 378-392. – Available: <http://ifets.ieee.org/russian/periodical/journal.html>. [in Russian]
2. Osadchyj V. V. The certification of professional knowledge and skills of students in the field of database as the necessary condition of further successful employment / V. V. Osadchyj, S. V. Sharov // Problemy inzhenerno-pedagogichnoj osvity. – 2015. – No. 46. – pp. 49-56. [in Ukrainian]
3. Kruhlyk V. Diploma Project Team Work Management / V. Kruhlyk // Journal of Information Technologies in Education (ITE). – 2014. – No. 18. – pp. 90-95. [in Ukrainian]
4. Overview – Redmine, Redmine.org [Online]. – Available: <http://www.redmine.org>. [in English]
5. RusGuide – Redmine, Redmine.org [Online]. – Available: <http://www.redmine.org/projects/redmine/wiki/rusguide>. [in Russian]
6. Barvenov S. The experience of using Git and GitHub in conducting classes with students / S. Barvenov, A. Stankevich // Proceedings of the 3rd International Scientific-practical Conference "Web-programming and Internet-technologies WebConf 2015", Minsk, May 12-14, 2015. – pp. 44-46. [in Russian]
7. Shaltunovich A. The organization of joint development of web applications within the frames of social network github / A. Shaltunovich // Vestnik Nizhnevartovskogo gos. universiteta. – 2011. – No. 3. – pp. 86-89. [in Russian]
8. Features For collaborative coding – developers work better, together | GitHub, GitHub [Online]. – Available: <https://github.com/features>. [in English]
9. Plokha O. The Agile methodology introduction in the learning process for students of IT specialties using project-based method / O. Plokha, O. Scherbakov, K. Liman // Actual scientific research in the modern world: XII Intern. Scientific. Conf., April 26-27, 2016. – 2016. – No. 12, p. 3. – pp. 65-71. [in Ukrainian]
10. Lavrishcheva K. Software Engineering / K. Lavrishcheva. – K., 2008. – 319 p. [in Ukrainian]
11. Danchuk V. The specifics of implementation of Agile methodologies for software development projects / V. Danchuk, D. Lutsyuk // Visnyk Natsionalnoho transportnoho universytetu. – 2011. – Vol. 24, no. 2. – pp. 346-350. [in Ukrainian]
12. Highsmith J. Agile Software Development Ecosystems / J. Highsmith. – AddisonWesley Professional, 2002. – 448 p. [in English]

13. Krayvanova V. Olympiad Programming as an Effective Tool of the Training of Professional Programmers / V. Krayvanova, E. Kruchkova // Vestnik NGU. Seriya: Informatsionnyie tehnologii. – 2012. – Vol. 10, no. 4. – pp. 51-56. [in Russian]
14. Salenieks P. Professional skills assesment in programming competition / P. Salenieks, J. Naylor // SIGCSE Bulletin. – 1987. – Vol. 4. – pp. 9-14. [in English]
15. Shilov N. Engaging students with theory through ACM collegiate programming contests / N. Shilov, Y. Kwangkeun // Communications of the ACM. – 2002. – Vol. 9. – pp. 98-101. [in English]
16. Gavra D. Programming Contest as the Discourse Practice of the Professional Community / D. Gavra, N. Ipatova // Vestnik SPbGU. Seriya 12. Sotsiologiya. – 2008. – Num. 4. – pp. 498-503. [in Russian]
17. Osadchyi V. The program of the pre-diploma practice of students of the specialty "Informatics" (educationally qualifictional level: specialist) / V. Osadchyi, K. Osadcha. – Melitopol: MDPU im. B. Khmelnytskoho, 2015. [in Ukrainian]

REFERENCES (TRANSLATED AND TRANSLITERATED)

1. Osadchij, V. V., & Osadchaja E. P. (2014). The analysis of the problem of the professional training of the programmer and the ways to solve it. The international electronic magazine “Obrazovatel'nye Tehnologii I Obshhestvo (Educational Technology & Society)”, 17(3), 378-392. <http://ifets.ieee.org/russian/periodical/journal.html>. [in Russian]
2. Osadchij, V. V., & Sharov, S. V. (2015). The certification of professional knowledge and skills of students in the field of database as the necessary condition of further successful employment. Problemy inzhenerno-pedagogichnoj osvity, 46, 49-56.
3. Kruhlyk, V.S. (2014). Diploma Project Team Work Management. Journal of Information Technologies in Education, 18, 90-95.
4. Overview –Redmine.org. Retrieved from <http://www.youtube.com/watch?v=lqM90eQi5-M>
5. RusGuide – Redmine.org. Retrieved from <http://www.youtube.com/watch?v=lqM90eQi5-M>
6. Barvenov, S., & Stankevich, A. (2015). The experience of using Git and GitHub in conducting classes with students. Proceedings of the 3rd International Scientific-practical Conferece "Web-programming and Internet-technologies WebConf 2015" Minsk, 44-46.
7. Shaltunovich, A. (2011). The organization of joint development of web applications within the frmes of social network github. Vestnik Nizhnevartovskogo gos. universiteta, 3, 86-89.
8. Features For collaborative coding – developers work better, together | GitHub. Retrieved from <https://github.com/features>.
9. Plokha, O., Scherbakov O., & Liman K. (2016). The Agile methodology introduction in the learning process for students of IT specialties using project-based method. Actual scientific research in the modern world: XII Intern. Scientific. Conf., 12(3), 65-71.
10. Lavrishcheva, K. (2008). Software Engineering.
11. Danchuk, V., & Lutsyuk, D. (2011). The specifics of implementation of Agile methodologies for software development projects. Visnyk Natsionalnoho transportnoho universytetu, 24(2), 346-350.
12. Highsmith, J. (2002). Agile Software Development Ecosystems. AddisonWesley Professional.
13. Krayvanova, V., & Kruchkova, E. (2012). Olympiad Programming as an Effective Tool of the Training of Professional Programmers. Vestnik NGU, Seriya: Informatsionnyie tehnologii, 10(4), 51-56.
14. Salenieks, P., & Naylor, J. (1987). Professional skills assesment in programming competition, 4, 9-14.
15. Shilov, N., & Kwangkeun, Y. (2002). Engaging students with theory through ACM collegiate programming contests. Communications of the ACM, 9, 98-101.
16. Gavra, D., & Ipatova, N. (2008). Programming Contest as the Discourse Practice of the Professional Community. Vestnik SPbGU, Seriya 12, Sotsiologiya, 4, 498-503.
17. Osadchij, V. V., & Osadchaja E. P. (2015). The program of the pre-diploma practice of students of the specialty "Informatics" (educationally qualifictional level: specialist). Melitopol: MDPU im. B. Khmelnytskoho, 2015.

Стаття надійшла до редакції: 25.05.2017

Круглик В.С.

Мелітопольський державний педагогічний університет імені Богдана Хмельницького, Мелітополь, Україна

ФОРМУВАННЯ ПРОФЕСІЙНИХ КОМПЕТЕНЦІЙ МАЙБУТНІХ ІНЖЕНЕРІВ-ПРОГРАМІСТІВ В ПРОЦЕСІ САМОСТІЙНОЇ НАВЧАЛЬНОЇ ДІЯЛЬНОСТІ

Статтю присвячено проблемі організації самостійної навчальної діяльності майбутніх інженерів-програмістів під час навчання в університетах. Організація самостійної діяльності студентів є ефективним засобом підвищення якості професійної підготовки майбутніх інженерів-програмістів у вищих навчальних закладах. Така діяльність передбачає вирішення складних творчих завдань, пов'язаних з реальною практикою роботи програмістів на підприємствах. Вона спрямована на формування у студентів професійних компетенцій, готовності до подальшого самовдосконалення, здібностей приймати рішення і нести відповідальність за їх наслідки, виявляти помилки, знаходити шляхи їх виправлення та мінімізації негативних наслідків, адекватно реагувати на професійні проблеми. Перед викладачами стоїть завдання вибрати напрямки, засоби і зміст самостійної роботи таким чином, щоб вона була ефективною і сприяла залученню студентів до професійної діяльності.

У статті проаналізовані такі способи організації самостійної роботи студентів: виконання дипломних проектів, науково-дослідна робота, участь в професійних олімпіадах і конкурсах, проходження виробничих практик. Наведено опис етапів виконання дипломних проектів та змісту діяльності викладача і студентів на кожному з них. Особлива увага приділяється колективній роботі над проектом і використанню інформаційних технологій для організації такої взаємодії. Охарактеризовані ефекти застосування самостійної роботи студентів під час виробничих практик, участі в олімпіадах і конкурсах, виконання дослідної роботи.

Ключові слова: повна вища освіта; інженер-програміст; дипломний проект; виробнича практика; професійні олімпіади; самостійна навчальна діяльність.

Круглик В.С.

Мелітопольский государственный педагогический университет имени Богдана Хмельницкого, Мелітополь, Україна

ФОРМИРОВАНИЕ ПРОФЕССИОНАЛЬНЫХ КОМПЕТЕНЦИЙ БУДУЩИХ ИНЖЕНЕРОВ-ПРОГРАММИСТОВ В ПРОЦЕССЕ САМОСТОЯТЕЛЬНОЙ УЧЕБНОЙ ДЕЯТЕЛЬНОСТИ

Статья посвящена проблеме организации самостоятельной учебной деятельности будущих инженеров-программистов во время обучения в университетах. Организация самостоятельной деятельности студентов является эффективным средством повышения качества профессиональной подготовки будущих инженеров-программистов в высших учебных заведениях. Такая деятельность предполагает решение сложных творческих задач, связанных с реальной практикой работы программистов на предприятиях. Она направлена на формирование у студентов профессиональных компетенций, готовности к дальнейшему самосовершенствованию, способностей принимать решения и нести ответственность за их последствия, выявлять ошибки, находить пути их исправления и минимизации негативных последствий, адекватно реагировать на профессиональные проблемы. Перед преподавателями стоит задача выбрать направления, средства и содержание самостоятельной работы таким образом, чтобы она была эффективной и способствовала привлечению студентов к профессиональной деятельности.

В статье проанализированы такие способы организации самостоятельной работы студентов: выполнение дипломных проектов, научно-исследовательская работа, участие в профессиональных олимпиадах и конкурсах, прохождение производственных практик. Приведено описание этапов выполнения дипломных проектов и содержания деятельности преподавателя и студентов на каждом из них. Особое внимание уделяется коллективной работе над проектом и использованию информационных технологий для организации такого взаимодействия. Охарактеризованы эффекты применения самостоятельной работы студентов

во время производственных практик, участия в олимпиадах и конкурсах, выполнения исследовательской работы.

Ключевые слова: высшее профессиональное образование; инженер-программист; дипломный проект; производственная практика; профессиональные олимпиады; самостоятельная учебная деятельность.