Uladzimir KAVALENKA
Siarhei LAPICH
Maryna NIKALAYUK-RTSISHCHAVA
Brest State Technical University,
Brest, Republic of Belarus

TEAMWORK SKILLS DEVELOPMENT

Introduction

In accordance with modern realities, the industry has special requirements for the training of students, including the skills of teamwork on projects. This skill is relevant at any time, but now, because of the growing popularity of remote work (both in connection with coronavirus, and as a result of natural processes) there is an additional challenge — to develop it through remote learning and interaction.

To begin with, let's consider what skills students should acquire:

- understanding and application by trainees of the principles of division of responsibilities in the team based on the personal and professional qualities of each team member;
- correct setting of intermediate goals and deadlines for their implementation;
- code quality control, test coverage and error handling;
- ability to write code together (pair coding);
- analysis and evaluation of the result (total and stages), adjustment and reassessment of the next stage;

- to get knowledge and try in reality modern methods and techniques of development, testing, launch and maintenance of software products;
- connection between related disciplines.

That is, each student must gain knowledge and experience of the full cycle of software development (from brainstorming at the beginning, to the final presentation) [1]. Also, it is necessary to describe separately the requirements to which the chosen method of achievement of the above-described goals should correspond:

- the teacher's ability to monitor and correct (if necessary) the decisions made by students at each stage;
- evaluate each learner's contribution and make final grades;
- the complexity of the task must be sufficient to handle a multitude of related tasks (debugging, testing, launching, and maintenance);
- the ability of all team members (including the instructor) to work remotely while fully accomplishing the task.

Based on the above, there are two ways to organize students' work:

- complex of special interconnected laboratory works;
- one large-scale project for a long time.

Each of these methods has its own advantages and disadvantages. To determine the best, you must carefully analyze each of them.

Complex of special laboratory works

"Classic" laboratory works, in which each of them is designed to work on one particular aspect or area of knowledge, do not meet modern requirements for the learning process. Losing coherence, and most importantly, in this case it is difficult, or almost impossible, to establish competent teamwork between students because of the small amount of work required to perform a single laboratory. Also, in this case, it is impossible to single out separate roles for testers, devops-engineers, etc., so that everyone would get all the necessary knowledge and do the work [2].

In accordance with modern realities, the industry has special requirements for the training of students, including the skills of teamwork on projects previous. In this case, the above goals will be achieved in a more complete way, as part of the work may be devoted to additional topics.

For example, let us consider the plan of laboratory work shown in Figure 1. It is assumed that by the beginning of its execution, the trainees have completed and have the result of the previous work. Based on their results, the trainees have experience working with the proposed database and the software structure in which it is planned to use it.

It should be noted that, with this approach, it is extremely difficult to create conditions for each learner to fully work out the teamwork on a large-scale project.

A large-scale project for a long time

The second option is a project on which the group (team) of students will work together for a long time (the most obvious option is a training semester, or about 4 months). In the process of working on such a project it is possible to achieve all the goals in full.

It is this format that allows you to fully pass the path of software development from team analysis of the task and division of responsibilities, to the final presentation of the results of work and protection of the project; to identify and develop leadership and other personal qualities, communication skills and teamwork. The instructor can, at each stage, monitor the work and give necessary advice and guidance [3].

It is also possible and necessary to consider the connection of the discipline, within which the project is carried out, with related disciplines studied in parallel or earlier. In such situations, it is necessary to speak about the project component, which belongs to this discipline. When performing such work, students should separately demonstrate and defend the results of their work in each subject.

In addition, when working can be used one of the most common methods of project management, such as SCRUM, Agile, etc., in which

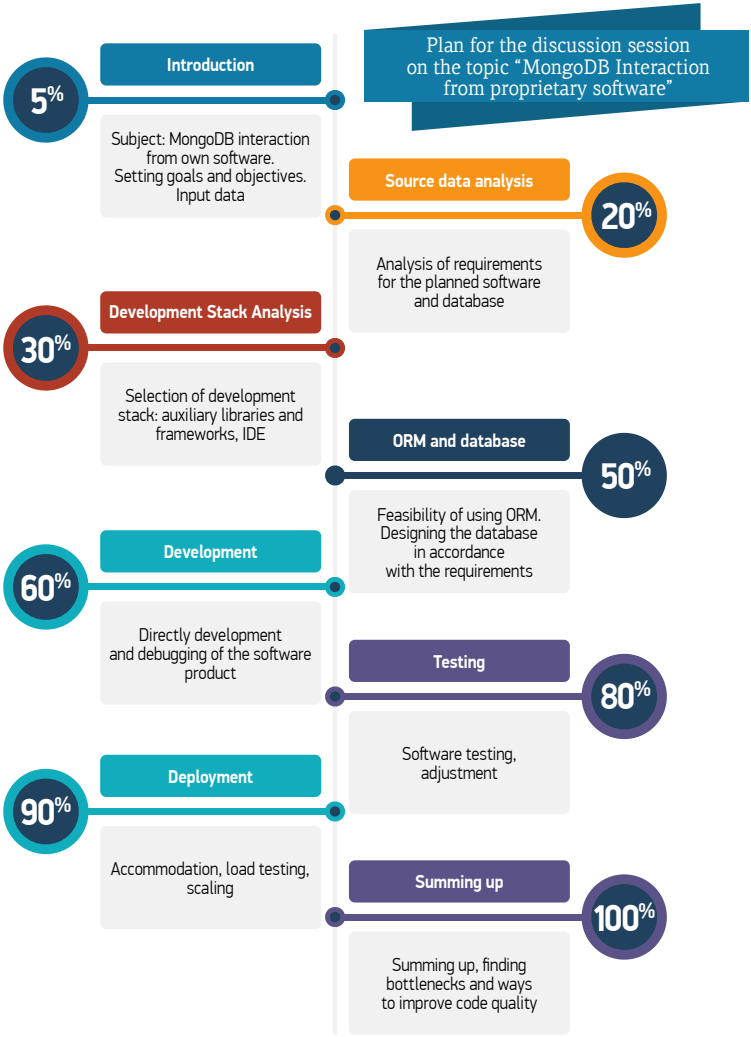

Figure 1 — *Laboratory work plan*

students themselves act as SCRUM Manager, SCRUM Development Team and the teacher act as SCRUM Product Owner.

Let us consider this approach on a concrete example for the discipline "NoSQL database".

Example of a team project in the discipline "NoSQL database"

The project plan is shown in Figure 2. It shows the plan of one of the components of the Magistrate's Team Project for "NoSQL databases" course, was developed at BrSTU for 2020/2021 magistrate students and deployed on the university Moodle portal, which in its turn was created in bounds of the learning process modernization inspired by the international project of EU Program ERASMUS+CBHE, Enhancement of Lifelong Learning in Belarus/BELL(586278-EPP-1-2017-1-LV-EPPKA2-CBHE-JP).

Non-relation database component in a group project of trainees

	Introduction	Configuring the database	Data change	Virtualization	Protection of the project
Discussion	— Setting goals and objectives, expected results; — Proposals for development tools (languages, IDE, version control system, testing, databases, cloud platforms)	Principles of data storage, user authorization	Ability to scale, restore, log all actions	Features of containerization, scaling, virtual machines	Methodology of "bottle necks" search
Controlled work	— Checking implementation opportunities; — Clarification of functionality and cost	Installation and configuration in GNU/Linux, authorization, server development	Interaction with the database from different NPS, CRUD, logging	Development in virtual machine, in Docker, using Mongo Atlas service	Justification of technical decisions taken, development of technical documentation
Results and adjustment	Changes in the stack and development principles	Optimized of requests, correct error handing	Comments, suggestions for optimization	Checking of correctness of work, optimization of requests	Load testing of the developed database

Figure 2 — *Team project plan*

The project goal is formed individually for each group depending on personal and professional qualities and acquired skills of each group. Since the group of trainees will be the team that will do the work, it is an excellent model of a situation where employees have to find an "approach" to other members of the group and will perfectly help develop communication skills. Combined with components from other disciplines, a fully completed project is the best way for a graduate to go the full software development path [4].

It should be noted that the complexity of the final goal of such a project (software) should be sufficient but not exceed some reasonable limit.

For example, the project implemented by one of the groups was a copy of www.booking.com website and service, but subsystems like payment processing and real interaction with hotels are not implemented.

Conclusions

Even a less useful model with a course of related laboratory work arouses additional interest among students, and the implementation of the project itself stimulates additional efforts. Moreover, it is possible to refer to participation in it at further employment.

All of this makes projects the most appropriate form of training for developing team skills, but it also provides additional moral incentives.

Moreover, according to personal experience, in several groups, where laboratory classes were replaced by a long-term project, the motivation and productivity of students increased. The desire to learn new, relevant technologies and techniques that are widely used today was stimulated.

LIST OF REFERENCES

1. Хасанова, Г.Ф., Шагеева, Ф.Т., Иванов, В.Г. (2014). Групповая проектная деятельность студентов как средство подготовки инновационных инженеров в исследовательском университете. *Вестник Казанского технологического университета*, 2014, с. 489–492.
2. Иванов, В.Г, Гурье, Л.И., Барабанова, С.В, Богоудинова, Р.З., Богатова, Л.М. (2012). Теоретические и методические основы инновационной подготовки инженеров в исследовательском университете. — *Казань, ГБУ "Республиканский центр мониторинга качества образования"*, 288 с.
3. Stahl, G. (2004). In What we know about CSCL: And implementing it in higher education. *Kluwer Academic Publishers, Boston, MA*, pp. 53–86.
4. Stahl, G. (2002). Computer support for collaborative learning: Foundations for a CSCL community (Boulder, CO, USA, 2002). Proceedings of CSCL. *Lawrence Erlbaum Associates, Boulder, CO*, pp. 62–71.