Teaching and Learning Activities Based on the Priscilla Tool

Beata Zielosko, Małgorzata Przybyła-Kasperek, Kornel Chromiński, Arkadiusz Nowakowski, and Eugenia Smyrnova-Trybulska

1 Introduction

Both online and blended learning poses many challenges regarding the methods of working with students, their activation in the learning process, and the selection of tools to support students and teachers. The chapter describes selected collaborative learning strategies and tutoring methods as important elements in the process of personalized education. An example of a tool supporting the work of both students and teachers is the Priscilla educational platform developed under the FITPED project.

2 Background

The topic of introducing innovative methods, tools and forms to the educational process is very important and actual, in particular in e-learning and blended learning processes and environments. The Covid-19 pandemic, with the accompanying lock-downs and the global transformation of the educational process to online education proved that in times of crises we do not have an alternative to online education and because of this, this mode of teaching and learning should be permanently improved and increased in quality.

B. Zielosko · M. Przybyła-Kasperek · K. Chromiński · A. Nowakowski
University of Silesia in Katowice, Katowice, Poland
e-mail: beata.zielosko@us.edu.pl; malgorzata.przybyla-kasperek@us.edu.pl; kornel.chrominski@us.edu.pl; arkadiusz.nowakowski@us.edu.pl

E. Smyrnova-Trybulska (✉)
Department of Humanistic Education, University of Silesia in Katowice, Katowice, Poland
e-mail: esmyrnova@us.edu.pl

E. Smyrnova-Trybulska et al. (eds.), *Microlearning*,
https://doi.org/10.1007/978-3-031-13359-6_6

From a different perspective, in the framework of the FITPED project (www.fitped.eu), the new educational "model is focused on minimizing the number of students with problems in learning" (Drlik et al., 2019, p. 12). The researchers and partners of the project stressed that increasing the level of knowledge obtained by students in the introductory courses focused on the development of programming skills need for increasing the level of of knowledge of highly specialized IT skills, as well as for training students for LLL. As a conclusion, they stressed, „the features of work-based learning, active learning, collaborative and problem-based learning approaches will be used during this process" (Drlik et al., 2019, p. 12).

Skalka et al. (2021) described the design of the framework from a peda-gogical point of view. Authors emphasize that Bloom's taxonomy is applied to all phases of the process of building knowledge and skills, from a programmer start-ing from zero experience to a professional ready to develop applications in practice within the context of microlearning too, and described the students' shift across the stages, according to Bloom's taxonomy from top to bottom.

There are various definitions of *microlearning*. For example, Theo Hug proposed it thusly: "The term *microlearning* has been used since the beginning of the twenty-first **century** mainly in the context of e-Learning. Commonly it stands for an abbreviated manner of expression for all sorts of short-time learning activities with microcontent." (Hug, 2012). The author also stressed:

> The term is used in many different ways. The spectrum of implicit or explicit definitions ranges from (1) unspecified forms of webspeak about learning by means of digital media and (2) ideological concepts of how learning processes ought to be organized according to the fast-moving world of technology to (3) differentiated conceptualizations of learning processes as related to micro-perspectives in the context of learning, education, and training. (Hug, 2012)

Aldosemani stressed that, "the development of new skills and knowledge requires a variety of teaching methods and learning strategies" (2019). "Students' opinion overviewed through an outline of the various phases of microteaching, implementation aspects, and microteaching content and activities used in this teaching method." (Aldosemani, 2019).

Other researchers "explore[s] [the] current understanding of microlearning as an effective model for professional development. From a theoretical perspective, the authors explore the rationale for microlearning by considering the literature on competency-based education (CBE) and microcredentialing." The authors "argue that microlearning can be a powerful model if the design is appropriate" (Zhang & West, 2020).

Experts Lee et al. (2021) conducted the "formative research to study the mobile microcourse's learning efficacy, defined as effectiveness, efficiency, and appeal… The results indicate that learners of this mobile microcourse had an increase in knowledge, more certainty in decisions about practical applications, and an increase in confidence in performing skills."

In conclusion, the authors stressed that "automated feedback, timed gamified exercises, and interactive real-world content indicate room for improvement to enhance effective learning" (Lee et al., 2021).

Another study (Jahnke et al., 2020) underlines that the results of their research "show a set of 15 principles regarding technical issues, pedagogical usability of micro-content interaction and sequenced instructional flow."

In a further part of the article, some learning strategies and methods using, in particular, Priscilla platform within the framework FITPED project will be presented and analysed.

3 Collaborative Learning Strategies

Collaborative learning is the instructional use of small groups of students, so they can work together to maximize their own and each other's learning activities (Jonhson & Jonhson, 2013). It is an educational approach to enhance the learning process of students through their work together, as a group. There are a lot of benefits of collaborative learning, among them, we can distinguish developing social interaction skills, leadership skills and/or oral communication skills (De Hei et al., 2016). It is clear that this strategy of learning supports an active learning environment and beyond knowledge allows one to acquire and develop skills that are very important nowadays. Below are shortly described popular collaborative learning strategies such as Problem-based learning (PBL), Jigsaw, Role-playing, STAD&TGT, and Peer tutoring.

3.1 Problem-Based Learning (PBL)

PBL is an active way of learning in which students learn about a subject by working in groups to solve real problems. The main goals of the PBL system are orderliness of knowledge (often from various fields) so that it can be used in practice, development of learning skills of students and increasing their motivation to acquire knowledge. It is a method that by assumption makes the student search for solutions by themselves.

The idea behind this strategy can be presented in the form of the following steps (Problem-Based Learning, 2021):

1. Discussion in a group of students the problem that should relate to real situations. This step can be considered as a preparation stage. The teacher should define the time frame and the form in which the solution will be presented.
2. Problem analysis. In the framework of this stage, roles must be assigned to the group. Then the group discusses the directions to solve the problem, as well as possible tasks. It should be identified what the group already knows and what are the potential solutions.
3. Analyzing the results of the brainstorming session. After establishing goals and objectives, students discuss their plans and who is responsible for which activi-

ties (e.g. research, experimentation, literature, interviews, etc.). They also need to formulate learning goals for knowledge that is still lacking.
4. Solution. In this stage, **students** study and follow pre-assigned tasks to solve a problem, combining individual and group work.
5. Discussing the results. Students **summarize** the solution and turn it into a form that can be presented to others. Druing this stage, they can identify new concepts and knowledge acquired during the process of solving a problem.

3.2 Jigsaw

The jigsaw classroom is a research-based cooperative learning technique invented and developed in the early 1970s by E. Aronson and his students at the University of Texas and the University of California. This strategy can be used in different courses where students have to learn a certain part of the material, which can be divided into coherent fragments.

The idea is like in a jigsaw puzzle, where each piece (each student) is essential for the completion and understanding of the final result. The basic principle of the jigsaw method is that each member of the group should become an expert who significantly contributes to the achievements of the entire team. Everyone is important because the results of all depend on his work, each student must use the knowledge acquired by other students and has to help everyone else. This strategy can be presented in the following steps (The Jigsaw classroom, 2021):

1. Dividing students into jigsaw groups, each group contains five to six students.
2. The teacher should divide the content of the lesson into five to six segments, (for example, SQL language contains instructions that can be related to: (i) definition and modification of the table structure (CREATE, ALTER), (ii) selecting data with different conditions (SELECT WHERE), (iii) selecting data with aggregating functions and grouping (GROUP, HAVING), (iv) updating and deleting data (UPDATE, DELETE), (v) granting and revoking permissions (GRANT, REVOKE). Each student from the group has been assigned one segment to learn.
3. Creation of temporary "expert groups". Each such group contains one student from each jigsaw group. Such students have been assigned to learn the same segment of material. Expert groups organize their knowledge relative to the given batch of material, clarify doubts and think about the best method of teaching other members of their jigsaw group.
4. Bringing the students back into their jigsaw groups. Each student presents his segment of material to the group. Each group member should master all the material.
5. At the end of the session, the teacher can give a quiz related to the considered subject. Questions can be prepared by the teacher or by the students (for example, in expert groups—for each batch of material).

The Jigsaw strategy supported by the Priscilla platform (Skalka & Drlik, 2018) was used during database course provided at the University of Silesia. The considered topics were: database modeling and SQL realised in a form of blended learning. Students registered in the Priscilla platform were assigned to the groups created by the teacher. In case of database modeling, the scope of the material has been divided as follows: creating entity relationship diagrams including entities, attributes, relationship between entities, Barker notation, and creating a database implementation diagram, including the primary key, foreign key, rules for transforming a conceptual model into a relational model. The first possibility of verification of knowledge in this area of material was solving short test questions available on the Priscilla platform. The second possibility of analyzing and verification of knowledge involved the creation of entity relationship diagrams and the transformation to the relational model for a given problem description was solving in the form face-to-face. In the case of the SQL language, both the acquisition of knowledge and the execution of commands took place using the functionalities available on the Priscilla platform.

3.3 STAD & TGT

Student Teams-Achievement Division (STAD) is a learning strategy where students work together to learn and they are responsible for the learning process of themselves and their teammates. To test the learning process, students take individual quizzes.

The STAD method consists of the following steps:

1. The teacher presents the material (a lesson).
2. Students work in teams of four to five to make sure that all team members have mastered the material.
3. The quiz on the lesson is taken individually by all students.
4. Students are assigned individual improvement scores.
5. The points (scores) are then summed up to form team scores and teams are recognized for the highest scores.

Teams-Games-Tournament (TGT) is based on the same teams work as in STAD but quizzes are replaced with weekly tournaments where students compete with members of other teams to contribute points to their team score. Tournaments are played at a “tournament table” (three students) against other players with a similar score from the previous week. The winner at each tournament table wins points for their team. Similar to STAD, teams are recognized for the highest score.

TGT is appropriate for the same types of objectives as STAD. There are many studies on STAD and TGT which have shown the positive effects on using these methods in learning math, science, arts, and other subjects (Slavin, 2010).

3.4 Role-Playing

Role-playing is an active pedagogical approach where students engage in relevant scenarios in order to gain cognitive and behavioural understanding. They may "act out" imaginary characters or, in some instances, play themselves. Additionally, scenarios may be contemporary or historical (Golwitzer, 2018). This strategy of activating students during the learning process can be used in different disciplines, such as medicine, history, engineering, geography, international relations and many others. Researchers noted that role-playing can help students make more explicit connections between content, which can lead to the ability to apply knowledge in different contexts. Role-playing allows students to practice skills such as negotiating, making decisions, expressing their opinions and emotions, and communication, and leads to developing students' interests through experience. When a teacher decides to use the role-playing strategy in the class, the following issues should be considered:

1. Getting to know the problem/issue, which should be complex with multiple perspectives and opinions. Roles can be performed by individual students, in pairs, or groups.
2. Developing characters with sufficient detail. Students should receive information pertaining to background, goals, and expectations of their character (for example, an aggressive client).
3. Providing assessment or reflection and discussion about the interactions, such as alternative ways of dealing with the situation.

3.5 Peer-Tutoring

Peer tutoring is generally described as a specific form of collaborative learning in which the tutor (for example an experienced student) offers help and support to one or more tutees (students). There is a wide variation of how peer tutoring can be applied in practice, for example, peer tutoring with small or large groups, same-age or cross-age, online or face-to-face. In the next section, tutoring is described in more detail.

3.6 Tutoring

Tutoring is a very popular method of organizing individual learning. In many publications (Alegre Ansuategui & Moliner Miravet, 2017; Mohamed & Lamia, 2018), we can find arguments that this form of individual support brings very positive effects. It is aimed not only at the achievement/skills, such as learning a specific programming language but also at the general development of the tutti/student. A

good organization of the tutoring process is very important. Initial arrangements should be made and the rules of the meetings should be written down. It is also extremely important to set a goal, which should be demanding, but not exceeding the possibilities of the tutti. In this section, the meaning of tutoring and the difference between mentoring, coaching and tutoring will be described. Guidance will be given on how to plan meetings and some selected tutorial tools will be discussed.

4 Differences and Similarities Between Tutoring, Mentoring and Coaching

Tutoring is a method of individual care for the tutti/student, an innovative method of developing creativity and independent thinking. Organized in this way, the teaching process is not only focused on achieving a specific goal, but also on the internal development of tutti—among others, the skills of independence, responsibility, constructive and critical thinking skills, self-discipline can be improved.

Traditional teaching methods are based on mentoring, i.e. an expert in the field/mentor knows, knows how to achieve the specified goal and passes this knowledge to the student. The student is rather passive in such a process, he should listen and take knowledge from the mentor. Tutoring follows completely different rules. The tutor knows, but he does not pass it on. It allows the tutor to independently choose how to achieve the goal. However, he supports the student in this way, by asking questions, he can indicate a wrong direction, but never imposes his opinion.

Coaching is also a modern technique for supporting individual development and achieving goals. A coach is a person who does not have knowledge in a given field, does not know the way to the goal, but is able to support and motivate in achieving the intended goals. The main tool for achieving goals in both tutoring and coaching is the ability to ask appropriate questions and use motivational techniques. Both methods are aimed at supporting the student, not leading or imposing the teacher's will. However, the main difference here is knowledge, the tutor has the knowledge, while the coach does not.

5 First Meeting—Setting Goals and Rules for Meetings

The tutoring process should consist of a minimum six meetings. The first meeting is very important and should be devoted to setting out the rules according to which the entire process will be organized. Such a set of rules should be written in the form of a contract and signed by both sides—the tuti and the tutor. Among these principles, issues such as confidentiality should be addressed. Tutoring meetings should be confidential, honest, friendly and devoted entirely to the tuti. In addition, the tutti has to ensure that they are ready to develop or change, and there is space between

where the tutti is now and where they are going to be. It is also very important to define such basics as the time, form and place of meetings, punctuality and what will happen when an absence is not reported. The goal of the tutoring meetings should also be specified and described in the contract. This target may change, in this case, the contract must be renegotiated.

6 Selected Tutorial Tools

In this section, selected basic tools that could be applied during tutorial meetings will be discussed. A tutor circle can be used in the initial phase of tutoring meetings to explore the goals of the tutti in a given area of life. A values table can be used to make the tutti aware of what values are actually the most important for them and to check whether a given tutti and the tutor can work together—similar values are crucial for them. A matrix of questions is helpful in systematizing and making aware what could happen in case the goal will be achieved or not. It is a motivating tool. The SMART card is a tool that could be helpful in determining the path to the goal, i. e. what tasks must be performed, what steps should be taken to achieve the goal and how to recognize that the goal has been achieved.

6.1 Tutor Circle

A tutor circle is actually many circles with a common centre and different radius that have been divided into parts, Fig. 1. Each part in the tutor circle will reflect one area: competencies, skills, priorities, problems, opportunities or other aspects. This will be determined by the tutti. The tutti fills the part of the circle by giving the degree of a goal's achievement (in some scale e.g. from one to four). Such activities

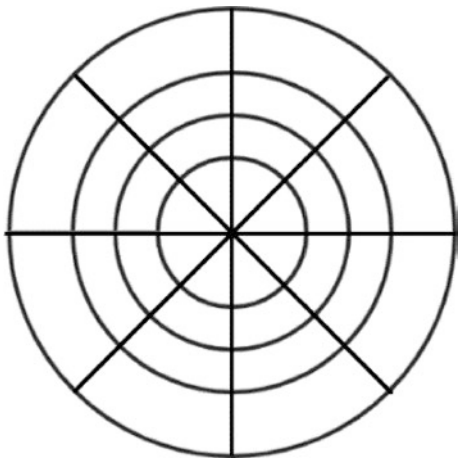

Fig. 1 Tutor circle. Source: Prepared by Małgorzata Przybyła-Kasperek

help the tutti be aware at what stage of achieving the goal they are currently located and in which particular areas they should work harder. Of course, the tutor should support the tutor with constructive questions, for example: Which area do you want to focus on first? Why?, What can you do to be satisfied with the results of this work?

6.2 Values Table

The table of values is helpful in determining the set of basic values that are the most important for the tutor. Firstly, this activity will determine the elements and areas which the tutti will not sacrifice in order to achieve the goal (it may be, for example, time spent with the family). The second important application of the values table is to check if the tutor and the tutti have a similar set of values. If there are fundamental differences, there is a high probability that good cooperation between such people will not be possible.

In order to apply this tool, we prepare a table containing different values. There should be a lot of values, e.g. 70. Among other things, there can be authenticity, safety, patience, curiosity, sensitivity, gentleness, pride, stability, flexibility, efficiency, enthusiasm, dignity, close relation, harmony, honour, consequence, creativity, loyalty, wisdom, love, hope, perfection. Then we ask the tutor to cross out the ten least important values, features from the table. Then we ask once again to cross out the next 20 least important values. We repeat these steps until the last five, the most important, values remain in the table. This tool not only allows one to learn about these most important features for a given person but also establishes a certain hierarchy of values.

6.3 Matrix of Questions

As was mentioned previously, the essence of tutoring is to support and motivate the tutti by, among other things, asking the right questions. The questions in different areas should be asked in order to analyze all possible scenarios. The matrix of questions can be helpful in this. It is presented in Fig. 2. The tutti should answer questions in each cell of the matrix, and then the consequences of various situations should be analyzed together.

Fig. 2 Matrix of questions. Source: Prepared by Małgorzata Przybyła-Kasperek

6.4 SMART Card

A SMART card is a tool that helps to learn about the characteristics of the goal. A deeper understanding of the goal affects its specificity and deetermines what steps should be taken to achieve it. In the SMART card, five rows are given, in each, there is one of the goal's characteristics: Specific, Measurable, Ambitious, Realistic, Timely. In the appropriate rows questions that will help the tutti to define the goal are presented. Examples of questions in each characteristic are given below:

S—What exactly do you want to achieve?
M—How will you know when you will achieve this goal?
A—Why is it important?
R—What resources do you need for this?
T—When do you want to end the activities?

Of course, the tutor gives additional questions as appropriate and helps summarize the entire activity.

Teaching in accordance with the principle of tutoring can be carried out using the PRISCILLA application (Skalka & Drlik, 2018). The goal will be, for example, to learn one of the programming languages. PRISCILLA is a tool that allows the student to learn independently, acquire knowledge during microlessons and practise skills in tasks. The tutor should help to define the goal as well as motivate and accompany the student on their way to achieving the goal.

The PRISCILLA application was used during tutoring classes in the project "Masters of didactics—Tutoring for the best students" that was realized at the University of Silesia in Katowice. Students who participated in the program learned programming languages, such as Python, using the PRISCILLA application. In this process, students were supported by a tutor who helped in setting sub-goals and milestones in the learning process. Additionally, gamification elements available in the PRISCILLA application were used. This tool was very motivating for students and was a helpful tool to evaluate the learning progress. Overall, project's participants positively assessed the PRISCILLA application and used it with pleasure.

7 Automated Programming Assignment Based on Priscilla

In the tutoring of programming languages, we can distinguish two learning ways: rote learning and understanding (Mayer, 1981). In general, both methods perform well in learning problem-solving, yet, concerning teaching programming languages, it is essential to use techniques that foster understanding. The PRISCILLA platform (Skalka & Drlik, 2018) allows teaching the former as microlessons and the latter as an automated programming assignment.

Automated programming assignment is a type of exercise/assignment that employs a student to write small programs that are run on a server automatically and verified using premade test cases. This approach is very similar to unit testing, a

method of software testing that tests an individual unit of source code to determine whether it is fit for use. Nevertheless, in terms of teaching, test cases should be prepared not only for verifying the code itself but also to show students flaws in their works alongside descriptive messages, which lead students to the correct solution. As in unit testing, a single automated programming assignment can be defined as three elements: arrange, act, and assert. The first element, "arrange", means all input data from test cases and preconditions like loading external libraries and variables definitions. "Act" is a student's work, but sometimes regarding the expected output, the form of the solution must be provided (e.g. a function with a specific name). The last element, "assert", is the verification of a student program result against expected values that sometimes includes data transformation. To enhance the above description, verify the below example from a PHP course.

Task description: *Assume you have the array $arr with numerical keys; print the keys (comma-separated) that values containing the value: 3.*

Example (a part of the description):

```
input : [1,2,3,3,5]
output: 2,3
The content of an assignment file:
<?php
// your array, do not remove this line
$arr = json_decode(trim(fgets(STDIN)),true);
// write your code here
```

The expected file includes the input transformation as test cases are written in plain text. Finally, after code verifying, a student receives points calculated based on the number of passed test cases.

Using automated unit testing in teaching programming languages is efficient. Barriocanal et al. report that all students who used unit testing in their assignments improved their code quality in the experiment (Barriocanal et al., 2002). However, for some cases like web-development programming languages, an automated programming assignment requires additional work as tutors need to prepare a mockup of the environment, e.g. a browser for testing DOM API in JavaScript. These bothersome cases can extend the time of creating teaching materials and introduce confusing assumptions.

7.1 A Modern Approach to Learning programming Languages on the Priscila Platform

Learning one of the programming languages is one of the basic skills that an IT specialist must have nowadays. The ability to create your own programs is also useful in many other professions. Nowadays, there are many opportunities to learn the programming language of your choice by yourself, from video tutorials to

interactive platforms. One example of a platform that allows interactive learning of programming languages is the Priscila platform. Within the platform, it is possible to learn several programming languages, including Java, C++ and Python.

By learning through the Priscila platform, the learner has the opportunity to learn the syntax, commands and how to develop software in an accessible way, without having to install the development environment on their computer.

Learning particular issues in programming languages is based on three stages, as presented in the Fig. 3. In carrying out each topic, the learner is first introduced to the problem in the form of a theoretical introduction with examples. In the next step, the learner can verify the acquired knowledge by taking a test. The final stage of each topic is a set of programming exercises covering the discussed issue. The implementation of programming exercises takes place through the website of the platform.

Learning programming languages using Priscila platforms begins with introducing the basics of a given language, and then in the next steps, expanding the knowledge of the learner more and more, up to advanced programming (see Fig. 4). After reading a topic and practical exercises performed by confirming that the learner comprehends the material, they have the opportunity to move to the next question. For each well-solved task, the learner receives additional points, which can be exchanged for hints about tasks, which additionally allows them to better understand the problem.

Programming exercises are designed in such a way as to best teach the learner to use the aspects of a given programming language discussed in a given topic. For the implementation of programming exercises, the platform user is not required to have a programming environment installed locally on their computer. The entire code written as part of the exercise is entered directly on the platform's website. Each task has a description of the problem to be solved, as well as an example input and expected output for the program. After writing the program code, the user can run it directly from the Priscila platform, as an example, the Python programming task is shown in Fig. 5. After running the code written by the user, test cases checking the correctness of the written solution are evaluated, and the user is informed about the result of this check.

Each task from the administrative side that creates the course, apart from the content of the task itself, also contains a solution that the platform user can display

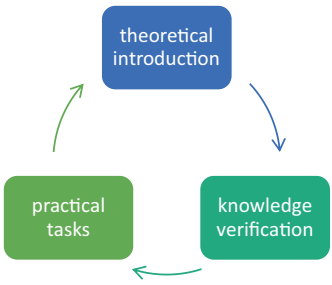

Fig. 3 The stages of teaching particular issues of programming languages on the Priscila platform. Source: Prepared by Kornel Chromiński

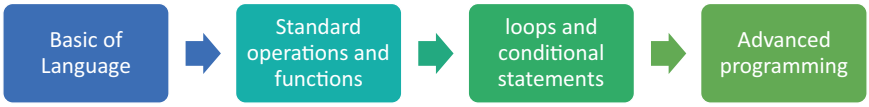

Fig. 4 the stages of introducing topics from programming languages. Source: Prepared by Kornel Chromiński

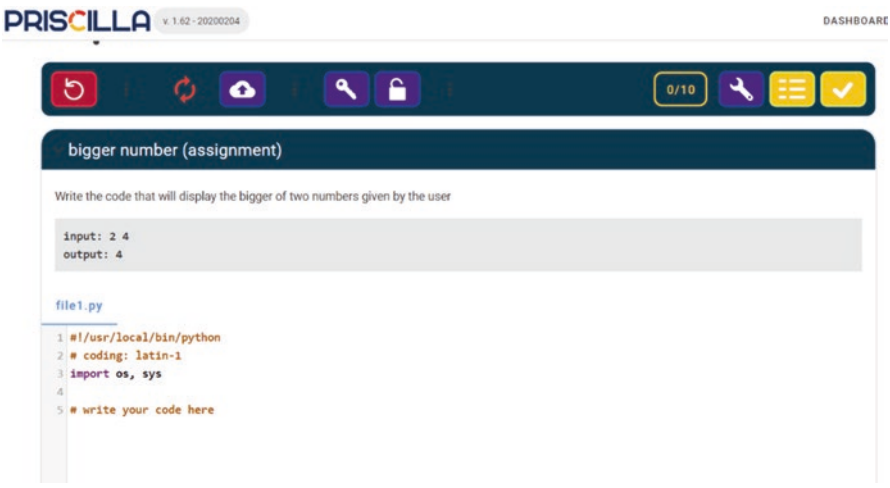

Fig. 5 Examples of programming tasks in Python language. Source: https://priscilla.fitped.eu/courses/3

if they have a problem with solving a given task, and a set of test cases containing sample inputs for the program and expected outputs (see Fig. 6).

This modern approach to learning programming languages allows a student to learn independently and verify their progress. The issues discussed in the framework of learning programming languages are divided into small blocks so as not to provide too much knowledge at one time, and to focus on practical tasks confirming the learning of a given part of the material. For example, for the Python language, the course is divided into 39 main thematic blocks, each block consists of several sub-topics, as well as practice questions and programming tasks. The Python course includes a total of 270 test questions and 225 programming tasks adapted in the best way to the presented topic.

The Priscila platform can also be a great tool to support the teacher in the process of teaching programming languages. As part of the platform, it is possible to create class groups and assign students to them, and to track their progress in the implementation of the course. It allows also to provide discussions on a given topic among students as well as modertate discussion by a teacher. Thanks to information about tasks solved by students and scored points, the teacher can follow the students' activity and progress. The implemented on the platform gamification module ensures the integration of elements of competitiveness and motivations for

Authors solution:

```
#!/usr/local/bin/python
# coding: latin-1
import os, sys

# write your code here
a = int(input())
b = int(input())
if a>b:
  print(a)
else:
  print(b)
```

Test cases (358 chars)

```
Case = Test1
input = 4
2
output= 4
Case = Test2
input= 6
5
output= 6
```

Fig. 6 Author's solution and a list of test cases for a programming task. Source: https://priscilla.fitped.eu/courses/3

surname	name	score (1860)	%	score (960/900)	Basic terminolo... (60/0)
[illegible]	[illegible]	1190	91.7/34.4%	880/310	60/0
[illegible]	[illegible]	60	06.3/00.0%	60/0	60/0
[illegible]	[illegible]	908	62.3/34.4%	598/310	48/0
[illegible]	[illegible]	900	70.8/24.4%	680/220	60/0

Fig. 7 Summary of students' activity within the class group. Source: https://priscilla.fitped.eu/courses/3

achieving goals. Based on the observation of students' work results, the teacher can apply an individual approach, assigning students tasks with various levels of difficulty adapted to their needs and skills. Thanks to the possibility of working in a group, the teacher can also use the chosen technique of colaborative learning, e.g. jigsaw, by dividing the group of students into smaller teams.

Self-learning or supporting the process of learning programming languages using Priscila platforms allows a student to master a given programming language in an easy and accessible way.

8 Conclusions

Blended learning is one of the modern teaching methods that combine face-to-face learning and online learning. In the framework of FITPED project, a platform named Priscilla was created. It can be used as a platform, for e-learning classes concentrated around programming languages. This chapter describes how an automated programming assignments were created and a case study for using Priscilla in the e-learning process for Python and how other programming languages were implemented.

Acknowledgements This research was funded by European Commission under the ERASMUS+ Programme 2018, KA2, grant number: 2018-1-SK01-KA203-046382 "Work-Based Learning in Future IT Professionals Education".

References

Aldosemani, T. I. (2019). Microlearning for macro-outcomes: Students' perceptions of telegram as a microlearning tool. In T. Väljataga & M. Laanpere (Eds.), *Digital turn in schools—Research, policy, practice. Lecture notes in educational technology*. Springer. https://doi.org/10.1007/978-981-13-7361-9_13

Alegre Ansuategui, F. J., & Moliner Miravet, L. (2017). Emotional and cognitive effects of peer tutoring among secondary school mathematics students. *International Journal of Mathematical Education in Science and Technology, 48*(8), 1185–1205.

Barriocanal, E. G., Urbán, M. Á. S., Cuevas, I. A., & Pérez, P. D. (2002). An experience in integrating automated unit testing practices in an introductory programming course. *ACM SIGCSE Bulletin, 34*(4), 125–128.

De Hei, M. S. A., Strijbos, J.-W., Sjoer, E., & Admiraal, W. (2016). Thematic review of approaches to design group learning. *Educational Research Review., 18*, 33–45. https://doi.org/10.1016/j.edurev.2016.01.001

Drlik, M., Smyrnova-Trybulska, E., & Szczurek, A. (2019). Nowy projekt IT. *Gazeta Uniwersytecka [New IT project. University of Silesia Magazine]. 6* (266) marzec 2019. Retrieved 12–13, from http://gazeta.us.edu.pl/node/424333

Golwitzer, B. (2018). *Role-play*. Retrieved from http://www.teachingtoolbox.us/role-play/

Hug, T. (2012). Microlearning. In N. M. Seel (Ed.), *Encyclopedia of the sciences of learning edition*. Springer. https://doi.org/10.1007/978-1-4419-1428-6_1583

Jahnke, I., Lee, Y. M., Pham, M., et al. (2020). Unpacking the inherent design principles of mobile microlearning. *Technology, Knowledge and Learning, 25*, 585–619. https://doi.org/10.1007/s10758-019-09413-w

Jonhson, D. H., & Jonhson, F. P. (2013). *Joining together: Group theory and group skills*. Wilfrid Laurier University.

Lee, Y. M., Jahnke, I., & Austin, L. (2021). Mobile microlearning design and effects on learning efficacy and learner experience. *Educational Technology Research and Development, 69*, 885–915. https://doi.org/10.1007/s11423-020-09931-w

Mayer, R. E. (1981). The psychology of how novices learn computer programming. *ACM Computing Surveys (CSUR), 13*(1), 121–141.

Mohamed, H., & Lamia, M. (2018). Implementing flipped classroom that used an intelligent tutoring system into learning process. *Computers & Education, 124*, 62–76.

Problem-Based Learning. (2021). *Problem-based learning.* Retrieved from Maastricht University. https://www.maastrichtuniversity.nl/education/why-um/problem-based-learning

Skalka, J., & Drlik, M. (2018) Priscilla—Proposal of system architecture for programming learning and teaching environment. In *Proceedings of the 2018 IEEE 12th international conference on application of information and communication technologies (AICT)*, 17–19 Oct 2018.

Skalka, J., Drlik, M., Benko, L., Kapusta, J., Rodríguez del Pino, J. C., Smyrnova-Trybulska, E., Stolinska, A., Svec, P., & Turcinek, P. (2021). Conceptual framework for programming skills development based on microlearning and automated source code evaluation in virtual learning environment. *Sustainability, 13*, 3293. https://doi.org/10.3390/su13063293

Slavin, R. E. (2010). Co-operative learning: What makes group-work work? In R. E. Slavin (Ed.), *The nature of learning using research to inspire practice*.

The Jigsaw Classroom. (2021). Retrieved from https://www.jigsaw.org/#overview

Zhang, J., & West, R. E. (2020). Designing microlearning instruction for professional development through a competency based approach. *TechTrends, 64*, 310–318. https://doi.org/10.1007/s11528-019-00449-4