

GAMIFICATION IN EDUCATIONAL PROCESS: REALIZATION

Nataliya Shakhovska¹, Krzysztof Rejman², Yuriy Bobalo¹, Volodymyr Brygilevych²¹Lviv Polytechnic National University, Lviv, Ukraine*rector@lpnu.ua, nataliya.b.shakhovska@lpnu.ua*²State Higher School of Technology and Economics in Jarosław, Poland*pwste@pwste.edu.pl, vbrygilevych@pwste.edu.pl*<https://doi.org/10.23939/jcpee2022.01.001>

Abstract: The paper presents an architecture of the education system with gamification. The system for assessing practical knowledge and skills through play is designed to optimize the educational process and the manifestation of students' creative thinking. The main purpose of this system is to simplify and automate the learning process in educational institutions.

Key words: gamification; educational process; web-site; class diagram.

1. Introduction

To date, in higher educational institutions, especially in technical ones, there exists a long-standing system of assessing students' knowledge and skills, which is based on the implementation of standard tasks for a laboratory exercise. Therefore, students do not show their initiative to perform the tasks, but on the contrary, seek a ready solution to such a problem [1].

Despite the popularity of the IT industry and the large number of students, the first years of study are difficult for them because programming and maths seem boring and difficult. All this causes a decrease in motivation for learning and self-development.

Computer games in turn are attracting more and more young people. In 7 out of 10 cases, young people prefer to play a computer game rather than do a lab work. Moreover, they want not only to waste the time playing, but also to create the games. Yes, such global competitions for programmed computer heroes or bots as Topcoder, USACO and others gather hundreds of thousands of participants every year. Therefore, there was a problem to organize tasks for a laboratory work in the form of game [2, 3]. In particular, it is about studying and implementing algorithms and methods of a programming language in the form of competitions to increase the interest in their own tasks, as well as provide an opportunity to express their creative ideas and non-standard solutions to problems. In addition, senior students will be involved in building a portal for the games. They will compose and implement tasks for junior students. This will allow for a continuous learning process and communication between students of different years of study.

This topic is relevant today, as the issues of creativity of the younger generation, non-standard solutions to the problem, as well as an interesting way of learning the material of programming is of great interest for both teachers and students. The development of such portals is also interesting to science, as the collected statistics will allow the cognitive characteristics of students to be taken into account and the learning process to be personalized. For students, it is an opportunity to show their skills, knowledge, share their experiences with others and get advice from others. Therefore, it is necessary to involve university teachers in practicing this method of teaching and assessing students.

The object of the study is an information process of gamification of educational process.

The web portal of the gamification system of the educational process and automatic assessment of the level of knowledge on laboratory work has been developed.

2. State of the art

The term "gamification" [3, 4] quickly began to gain popularity in 2010. It involves the use of game elements in "serious", i.e non-game, processes. The purpose of this process is to make the familiar traditional platform more interesting by adding a huge number of game elements such as achievement, hidden bonuses, competitions, progress, quest or leaderboard that can be used in the process of creating a website or program to solve an algorithmic or mathematical problem. Games are commonly used to visualize and improve actions and skills that are difficult to demonstrate using traditional methods, as well as to encourage students to compete with each other and evaluate their progress. Gaming describes a widespread trend in the use of game mechanics for non-gaming settings such as innovation, marketing, training, staff work, health and social change.

Methods of gamification [5, 6] seek to attract natural human instincts:

- competition;
- achievement;
- status;
- self-expression;

- altruism;
- problem solving.

Competition is an important element that can be used in gaming. It gives all participants the opportunity to see the rewards of others, or pay bonuses to leaders and thus encourage others to perform tasks.

The field of gamification is still relatively young and it lacks structured research and despite the relatively short time of its development, there are already many studies and publications on this issue. Thus, the study of Maciej Laskowski found that along with the increasing role of various social media there is a new trend in software development – gaming, the main purpose of which is to increase user’s activity when working with software, an attempt was made to analyze this phenomenon, positive and negative side [7–9].

Given the great potential of gamification, the following training programs have been developed and are actively used using game elements:

- Codecademy – learning programming in JavaScript, HTML, Python, Ruby;
- Motion Math Games – mobile games make learning maths fun and exciting;
- Mathematics – a program for schools aimed at involving children in mathematics through games and Challenge;
- Khanacademy – free video courses on various topics;
- Spongelab – a personalized platform for science education;
- Foldit – solving scientific problems in the form of puzzles.

After analyzing the existing methods and tools used in educational and scientific activities, it is proposed to use combined approaches with elements of gamification for practical work from educational portals Codecademy, Mathletics and Spongelab.

3. Main material

The game portal “Test My Labs” (“TML”) is developed in the paper.

“TML” performs 4 main functions:

- registers new and authorizes existing users;
- decides on sending and testing it;
- creates new groups and holds group competitions;
- provides information about the list of existing games and instructions for them.

The registration and authorization package consists of 3 classes in terms of backend, and 5 classes in terms of frontend:

- `app / auth / authenticate_user.rb` – a basic class for user authorization and registration;
- `app / models / user.rb` – a user model class with field validation and links to other entities;

- `app / controllers / users_controller.rb` – a user controller class, which contains methods of registration, authorization, output and verification of data;
- `src / app / auth / auth.service.ts` – the main class-service of user;
- `src / app / login / login.component.ts` – the main class of components with authorization methods;
- `src / app / login / login.service.ts` – an additional class-service for authorization methods;
- `src / app / login / login.component.html` – a component class to display the graphical authorization shell;
- `src / app / login / login.component.scss` – a style class for styling the authorization shell.

The package for sending and testing the solution consists of 4 classes in terms of backend, and 8 classes in terms of frontend:

- `app / models / test.rb` – a test model class with links to other entities;
- `app / controllers / tests_controller.rb` – a test controller class, which contains methods for creating a new test, deleting an existing one and obtaining both one and several test records;
- `app / controllers / solutions_controller.rb` – a solution controller class, which contains methods for creating a new solution, deleting an existing one and obtaining both one and several records about the solution;
- `app / models / solution.rb` – a solution model class with connections to other entities;
- `src / app / tests / tests.service.ts` – a test service class of the solution that sends requests;
- `src / app / tests / tests.component.ts` – a basic component class with methods for creating, deleting and obtaining tests;
- `src / app / tests / tests.component.html` – a component class for displaying the graphical shell of tests;
- `src / app / tests / tests.component.scss` – a style class for styling the test shell;
- `src / app / solutions / solutions.service.ts` – a solution service class that sends requests;
- `src / app / solutions / solutions.component.ts` – the main class of components with methods of creating, deleting and obtaining solutions;
- `src / app / solutions / solutions.component.html` – a component class for displaying the graphical shell of solutions;
- `src / app / solutions / solutions.component.scss` – a style class for shell solution styling.

The package for creating groups and group competitions consists of 6 classes in terms of backend, and 8 classes in terms of frontend:

- `app / models / group.rb` – a test model class with connections to other entities;
- `app / models / group_member.rb` – a solution model class with connections to other entities;
- `app / models / group_tests.rb` – a solution model class with connections to other entities;
- `app / controllers / group_members_controller.rb` – a class of controller of the group member;
- `app / controllers / group_tests_controller.rb` – a group test controller class, which contains methods for creating a new test and obtaining both one and several records of group tests;
- `app / controllers / group_controller.rb` – a group controller class, which contains methods for creating, editing and deleting a group, blocking another user by group administrator, inviting other users and receiving both one and several group entries;
- `src / app / shared / groups.service.ts` – a service class of the group that sends requests;
- `src / app / groups / groups.component.ts` – a basic component class with methods for creating, deleting and retrieving groups;
- `src / app / groups / groups.component.html` – a component class to display the graphical shell of groups;
- `src / app / groups / groups.component.scss` – a style class for styling the shell of groups;
- `src / app / shared / groups-tests.service.ts` – a group test service class that sends requests;
- `src / app / groups-tests / groups-tests.component.ts` – a basic component class with methods for creating, deleting and obtaining group tests;
- `src / app / groups-tests / groups-tests.component.html` – a component class to display the graphical shell of group tests;
- `src / app / groups-tests / groups-tests.component.scss` – a class-style for styling the shell of group tests.

The package for viewing the list of available games consists of 2 classes in terms of backend, and 4 classes in terms of frontend:

- `app / models / game.rb` – a game model class with connections to other entities;
- `app / controllers / games_controller.rb` – a game controller class;
- `src / app / shared / games.service.ts` – a class-service of games that sends requests;
- `src / app / games / games.component.ts` – the main class of components for obtaining information about games;

- `src / app / games / games.component.html` – a component class to display a graphical shell of information about games;
- `src / app / games / games.component.scss` – a style class for styling the shell of information about games.

4. Results

A database consisting of 10 tables with specified fields, their type and relationships between the tables has been developed (Fig. 1).

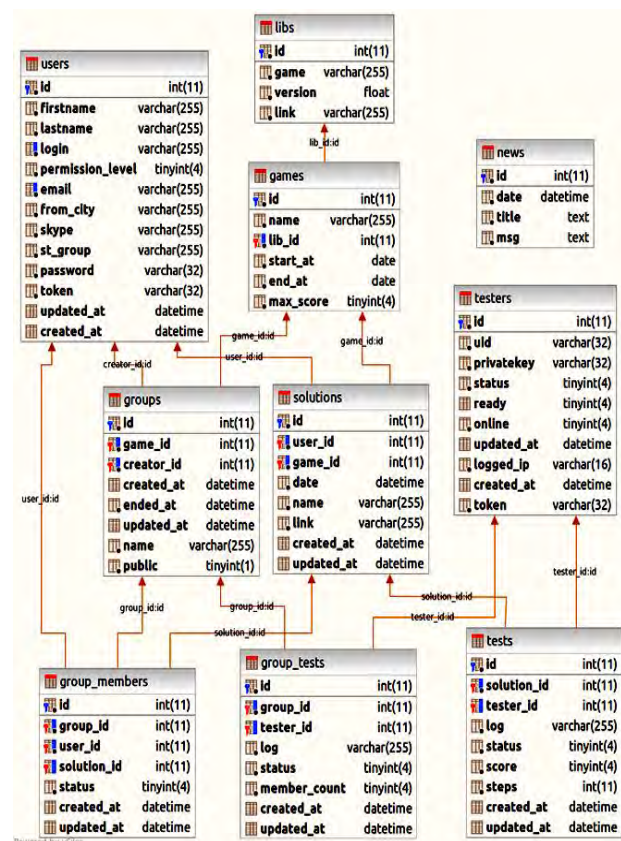


Fig. 1. The database schema

- a `libs` table with `id` fields (identifier) of type “integer”, `game` and `link` of type “text” of up to 255 characters, `version` of type “floating point numbers” and connection 1 to N with table of `games` on field `lib_id` in the latter;
- a `news` table with fields `id` (identifier) of type “integer”, `date` of type “date and time”, `title` and `msg` type “text” without links (isolated);
- table `games` with fields `id` (identifier), `max_score` and `lib_id` type “integer”, `name` type “text” up to 255 characters, `start_at` and `end_at` type “date” with links 1 to N with tables `groups` and `solutions` in the field `game_id` in the latter;
- table `users` with fields `id` (identifier) and `permission_level` of type “integer”, `firstname`,

lastname, login (uniqueness required), email (uniqueness required), from_city, skype and st_group type “ribbon” up to 255 characters, password and token type “tape” up to 32 characters, updated_at and created_at type “date and time” with links 1 to N with tables group_members, groups, solutions in the fields creator_id in the first and user_id in the last;

- table testers with fields id (identifier), status, ready and online type “integer”, uid, privatekey and token type “tape” up to 32 characters, updated_at and created_at type “date and time”, logged_ip type “tape” up to 32 characters 16 characters with links 1 to N with tables group_tests and tests in the field tester_id in the latter;
- table groups with fields id (identifier), game_id, creator_id and public type “integer”, created_at, ended_at and updated_at type “date and time”, name type “ribbon” up to 255 characters with links 1 to N with tables group_members and group_tests in the group_id field in the latter;
- table solutions with fields id (identifier), user_id and game_id type “integer”, date, created_at and updated_at type “date and time”, name and link type “ribbon” up to 255 characters with links 1 to N with tables group_members and tests on the solution_id field in the latter;
- table group_members with fields id (identifier), group_id, user_id, solution_id and status type “integer”, created_at and updated_at type “date and time”;
- table group_tests with fields id (identifier), group_id, tester_id, status and member_count type “integer”, log type “tape” up to 255 characters, created_at and updated_at type “date and time”;
- table tests with fields id (identifier), solution_id, tester_id, status, score and steps of type “integer”, log type “tape” up to 255 characters, created_at and updated_at type “date and time”.

User information is presented in JSON-file (Fig. 2).

```

JSON
- data: {...}
  id: 2
  firstname: Nicolas
  lastname: Stakhiv
  login: Nicolas
  permission_level: 4
  email: mykola.stakhiv@gmail.com
  from_city: Lviv
  skype: mykola.skype.ru
  st_group: KN-48
  updated_at: 2018-02-13T22:41:45.000Z
  created_at: 2016-07-27T18:35:29.000Z
  solutions_count: 9
  tests_count: 16
  error: []

```

Fig. 2. The description of student representation

Next, the game rules are presented in html-format (Fig. 3).



```

OGCNMVF003L.txt x
<?xml version="1.0" encoding="utf-8"?>
<xml>
<round>
  <bot>31_31</bot>
  <find>33</find>
  <start>70</start>
  <steps>
    <step>
      <move>up</move>
      <cords>
        <current>70</current>
        <parent>71</parent>
        <left>-1</left>
        <right>-1</right>
      </cords>
    </step>
    <step>
      <move>up</move>
      <cords>
        <current>71</current>
        <parent>72</parent>
        <left>70</left>
        <right>-1</right>
      </cords>
    </step>
    <step>
      <move>up</move>
      <cords>
        <current>72</current>
        <parent>69</parent>
        <left>71</left>
        <right>74</right>
      </cords>
    </step>
    <step>
      <move>up</move>
      <cords>
        <current>69</current>
        <parent>63</parent>
        <left>66</left>
        <right>72</right>
      </cords>
    </step>
  </steps>
</round>

```

Fig. 3. The description of the game rules

The main page is presented in Fig. 4. This page is used for the authorization too.



Fig. 4. The main page of the web-portal

Then, the user is able to choose the game. The game having been chosen, the rules page is activated (Fig. 5).

After that, the user can upload the solution. The assessment is organized automatically. The results of students’ knowledge assessment are presented in Fig. 6.

5. Conclusion

The game portal “Test My Labs” is developed and described in the paper. It allows students to try

themselves as a player in the performance of a task. This web resource allows students to understand the basic algorithms taught in the course “Theory of Algorithms”. Besides, they have access to task instructions and personal accounts with the ability to view their rating and performance. In addition, it is an opportunity to challenge others and evaluate students’ capabilities.

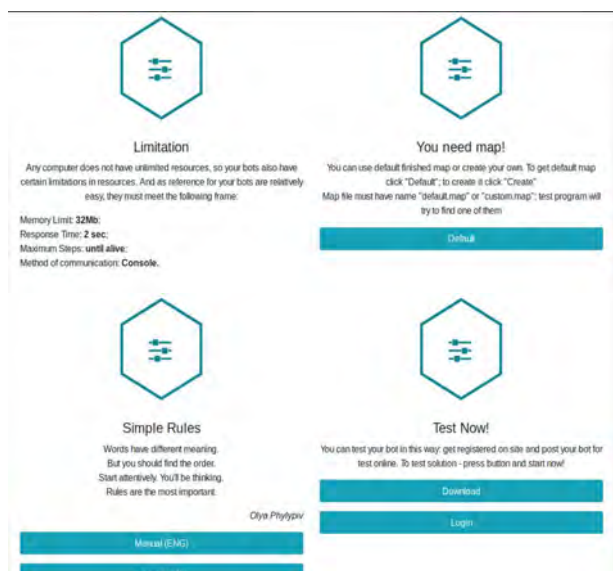


Fig. 5. Game rules and description page

#	Member	Solution	Status	Action
1	Stepan		✘ Refused	
2	Nicolas		👤 Owner	
3	natalya233		🔍 Not reviewed	
4	Ramfan		🚫 Banned	👤 Login 🗑️ Delete
5	Alaska		✔ Accepted	🚫 Ban
6	christina		🔍 Not reviewed	

Fig. 6. The web-page with students results

The user of such a system needs to log in to be able to submit his/her solution. The solution (programmed game logic) is sent by the user as an executable file. Each user-written program (hereinafter bot) can be tested in two ways.

The first way is the usual competition with other users' bots. The competition means the performance of the task set in the laboratory work for the time being. Accordingly, the winner will be the bot with the least computational complexity. Also, the task for a laboratory work involves restrictions on computational complexity (response time and number of steps).

The mode of single testing is an important link between the implementation of the bot and direct participation in competitions. Since the way of solving the game is individual, there is a problem of proper interaction and uniform conditions for all participants. To do this, with the terms and conditions of the game, each user also receives a description of the input/output format. The essence of the unit mode is to check how correctly implemented not only the direct logic of the player (his/her bot), but also the interaction of his/her algorithms with the environment itself.

It should be noted that unauthorized users have limited access rights: they are only allowed to view the main page of the website, game genres and instructions on how to perform the tasks (English and Ukrainian versions are available).

Authorized users have the opportunity to create a group, invite other members of the portal and create their own competition for a certain category of games. They can also view the news of the site in their profile, their statistics: the number of games played, the number of groups in which the user is registered, the number of solutions that are downloaded.

This topic is relevant today, as the issues of creativity of the younger generation, non-standard solutions to the problem, as well as an interesting way of learning the material of programming is of great interest to both teachers and students. The development of such portals is also interesting to science, as the collected statistics will allow the cognitive characteristics of students to be taken into account and the learning process to be personalized. For students, it is an opportunity to show their skills, knowledge, share their experiences with others and get advice from others. Therefore, it is necessary to involve university teachers in practicing this method of teaching and assessing students.

References

- [1] A. Stott and C. Neustaedter, “Analysis of gamification in education”. Surrey, BC, Canada, 8, 36, 2013.
- [2] Jenni Majuri, Jonna Koivisto, and Juho Hamari, “Gamification of education and learning: A review of empirical literature”, in *Proc. 2nd international GamiFIN conference, GamiFIN 2018, CEUR-WS*, 2018.
- [3] Roy Rob Van, and Bieke Zaman, “Need-supporting gamification in education: An assessment of motivational effects over time”. *Computers & Education*, no. 127, pp. 283–297, 2018.
- [4] A Hallifax, Stuart, et al. “Adaptive gamification in education: A literature review of current trends and developments”. European conference on technology enhanced learning. Springer, Cham, 2019.
- [5] Manzano-León, Ana, et al. “Between level up and game over: A systematic literature review of gamification in education”. *Sustainability*, vol. 13, no. 4, p. 2247, 2021.
- [6] Ofosu-Ampong, Kingsley, “The shift to gamification in education: A review on dominant issues”, *Journal of*

Educational Technology Systems, vol. 49, no. 1, pp. 113–137, 2020.

- [7] Jakub Swacha, “State of research on gamification in education: A bibliometric survey”, *Education Sciences*, 11.2, p. 69, (2021).
- [8] Ana Paula Lopes, et al. *Gamification in education and active methodologies at higher education*, pp. 1633–1640, 2019
- [9] Ofosu-Ampong, Kingsley, et al. “Are we ready for Gamification? An exploratory analysis in a developing country”, *Education and Information Technologies*, vol. 25, no. 3, 1723–1742, 2020.



Yuriy Bobalo – Rector of Lviv Polytechnic National University.

Scientific interests are concentrated on the sphere of reliable development of radio electronic equipment. Y. Bobalo is the author of over 130 publications, among which there is a monograph “System Analysis of Quality of Precise Radio Electronic Equipment Production” (1996), manuals “The Basics of Radio

Electronic Devices Reliability” (1998), “The Basics of Radio Electronics” (2002), “Methods of Radio Electronic Devices Reliability Calculation” (2005), a textbook “The Basics of Radio Circles Theory” (2006). 4 PhD theses have been defended under his supervision.



Nataliya Shakhovska – Head of Artificial Intelligence Department of Lviv Polytechnic National University.

She is the author of more than 200 papers. Scientific interests are concentrated on the sphere of artificial intelligence, machine learning, data integration.

ГЕЙМІФІКАЦІЯ У ОСВІТНЬОМУ ПРОЦЕСІ: РЕАЛІЗАЦІЯ

**Наталія Шаховська, Кшиштоф Рейман,
Юрій Бобало, Володимир Бригілевич**

У статті наведено архітектуру системи освіти з гейміфікацією. Система оцінювання практичних знань і умінь через гру покликана оптимізувати навчально-виховний процес та прояви творчого мислення студентів. Основна мета цієї системи – спрощення та автоматизація процесу навчання в закладах освіти.



Krzysztof Rejman, PhD, DSc, Associate Prof. – Rector of State Higher School of Technology and Economics in Jarosław, sociologist, author and co-author of numerous scientific publications in national and foreign publishing houses, editor and scientific co-editor of many works in the field of social sciences, a participant of many foreign research academic trainings

and research projects. He has actively participated in numerous domestic and foreign seminars and scientific conferences. His research topics largely concern the sociology of organizations, local and regional communities and the broadly understood cross-border cooperation between Poland, Ukraine and Slovakia.



Volodymyr Brygilevych – PhD, Associate Professor of the Institute of Technical Engineering at the State Higher School of Technology and Economics in Jarosław, Poland.

Research interests: mathematical modeling of processes in mechatronic systems, diagnosis of electronic systems.

Received: 14.12.2021. Accepted: 17.01.2022