Development of a PyGame workshop for extracurricular Computer Science education

Oleksandr O. Pozniakov, Pavlo V. Merzlykin

Kryvyi Rih State Pedagogical University, 54 Universytetskyi Ave., Kryvyi Rih, 50086, Ukraine

Abstract

With the rapid growth of the gaming industry, teaching game development in extracurricular education has become increasingly relevant. The analysis of the recommended extracurricular programs revealed that most of them do not completely cover contemporary game design approaches, suggest the use of outdated or expensive software, and advocate russian resources and services. This study presents a PyGame workshop for extracurricular Computer Science education. The workshop was designed as a step-by-step game creation guide, with each step logically connected to the previous and next lessons, ensuring gradual progress. The Python programming language and PyGame library were chosen for the workshop implementation due to their accessibility, extensive documentation, and suitability for beginners. The workshop was implemented as a website using Google Sites, providing easy access to learning materials, code examples, and resources. The collected feedback indicated high levels of engagement and satisfaction with the learning experience.

Keywords

PyGame workshop, extracurricular education, Computer Science education, game development, Python programming, project-based learning

1. Introduction

In a context where programming is becoming an important tool for solving a wide range of problems, education must provide access to effective and modern teaching methods. In this regard, creating computer games is an ideal combination of technical and creative components that contribute to the comprehensive development of students [1, 2, 3, 4, 5].

Extracurricular education is a flexible tool that allows students to realize their potential through practical activities. In this research, we are trying to examine the interests of students who take part in extracurricular Computer Science events and to test whether they fit what the officially recommended extracurricular programs propose them to study.

As a result of this comparison, we are going to design an extracurricular course that could fill the gap between students' expectations and extracurricular programs' recommendations.

2. Analysis of the current state of extracurricular Computer Science education

2.1. The system of extracurricular Computer Science education in Ukraine

The system of extracurricular Computer Science education in Ukraine is an important component of the educational process that contributes to the development of the scientific and technical skills of young people and their interest in information technologies. It is based on Ukraine's law "On Extracurricular Education" that regulates extracurricular education, as well as methodological recommendations that ensure its quality [6].

The key points of the Ukraine's law "On Extracurricular Education" [6]:

- The structure of extracurricular education includes state, municipal, and private extracurricular education institutions; other educational institutions as centers of extracurricular education; clubs and associations at the place of residence, regardless of subordination, types, and forms of ownership; cultural, educational, health, sports and other educational institutions and establishments; foundations, associations whose activities are related to the functioning of extracurricular education; relevant government bodies and scientific and methodological institutions; other subjects of educational activity providing educational services in the system of extracurricular education.
- Extracurricular education is carried out in a differentiated manner according to the individual capabilities, interests, inclinations, and abilities of students, considering their age, psychophysical features, and state of health in various organizational forms.
- The main tasks of extracurricular education include educating a citizen of Ukraine, promoting the free development of personality, forming socio-civic experience and patriotism, and a respectful attitude towards family, older generations, and cultural traditions.

The Institute of Education Content Modernisation offers recommendations for the development of STEM education, particularly Computer Science, in secondary and extracurricular educational institutions. Special attention is paid to the use of interactive teaching methods, such as organizing competitions, hackathons, and practical classes in robotics and programming, which contribute to the practical skills development and stimulate interest in modern technologies [7].

2.2. Analysis of student's demands from extracurricular Computer Science education

By its nature, extracurricular education is something that students choose to study voluntarily, that is to say, their interests should be appreciated during the extracurricular programs design. It will ensure their motivation and engagement.

The Computer Science extracurricular education demand can be assessed through the students' participation in scientific projects contests [8].

The analysis of the statistics from the Junior Academy of Sciences of Ukraine contests in the Dnipropetrovsk Oblast for 2014–2024 demonstrates the diversity of areas that attract students. The most popular Computer Science section throughout the studied period is "Multimedia systems, educational and game programs" (since 2023 "Educational, game programs and virtual reality"). This section consistently demonstrates a high level of student involvement, with over 10 participants every year (figure 1). For comparison, in other areas, the number of participants usually did not exceed 10 people (figure 2).

2.3. Review of existing extracurricular education programs related to game development

Despite the high demand from students, the analysis of educational programs recommended by the Ministry of Education and Science of Ukraine showed the absence of specialized courses on game development for extracurricular education (table 1). Some programs partially touch on this topic, such as "Fundamentals of Computer Graphics" by Yu. S. Dorofeyeva, Zh. V. Deineko or "Multimedia Systems, Educational, Game Programs" by V. V. Yefymenko, but their content requires significant changes.

Analysis of existing courses revealed several key problems that may affect the effectiveness of learning. Firstly, some programs have inconsistencies between the programming language, compiler, and sources, which complicates the learning process. In addition, some recommended resources are already outdated and may no longer be supported or relevant to current conditions. Some programs contain paid resources with high costs per user, which can become a barrier for students. Another problem is the presence of russian services and literature, which is unacceptable in the current educational situation, as each program contains such sources.

Given the identified problems, the development of a PyGame workshop for extracurricular Computer Science education appears to be relevant. This will help avoid the mentioned shortcomings, providing
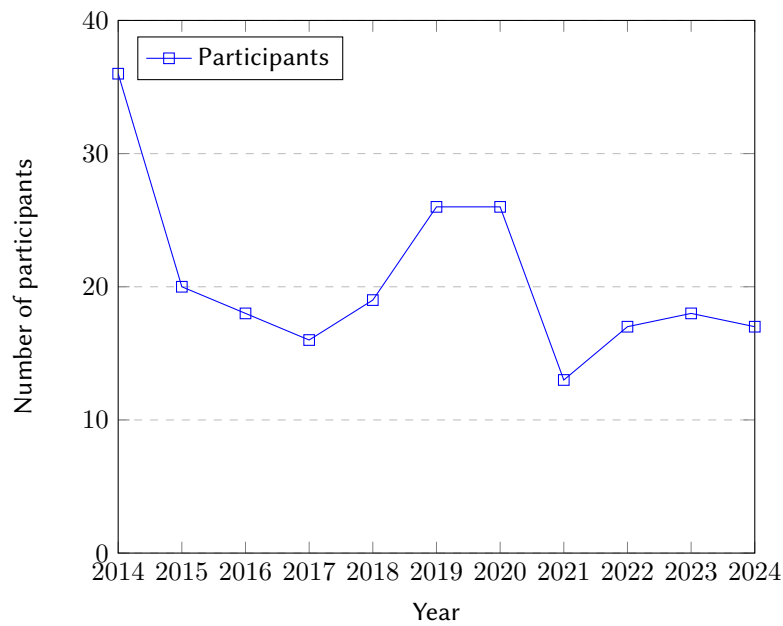
Figure 1: Diagram of the number of participants by year in the field of "Multimedia systems, educational and game programs" (since 2023 – "Educational, game programs and virtual reality").

Figure 2: Diagram of the share of participants by year in the field of "Multimedia systems, educational and game programs" (since 2023 – "Educational, game programs and virtual reality").
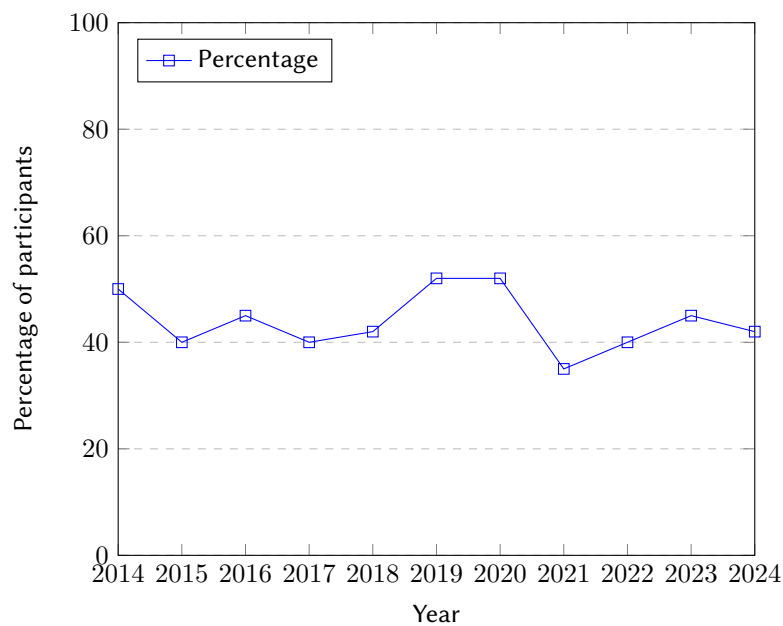
students with modern tools for learning programming and creating games that will meet today's security and relevance requirements.

2.4. Comparison of existing Ukrainian educational courses using PyGame

To develop a PyGame course that meets modern educational needs, it is essential to analyze existing learning programs. This helps identify effective teaching methods, evaluate their strengths and weaknesses, and integrate the best practices into a new curriculum. The following overview examines three Ukrainian educational courses that include PyGame.

Formula [10] offers a PyGame course within its "Graphic Programming" section, following an

Table 1
Review of extracurricular education programs from the collection of educational programs for extracurricular education [9].

Author	Program name	Game Dev related sections	Tools (✓ – relevant, ✗– outdated)	Other remarks
Yu. S. Dorofeyeva, Zh. V. Deineko	Fundamentals of computer graphics	Development of 2D and 3D graphics	✓MS Paint ✓Adobe Illustrator ✓Adobe Photoshop ✗Adobe Flash ✓3D MAX	Presence of Russian-language literature
D. S. Hrebeniuk	Fundamentals of Modern Programming	Description of basic algorithmic structures	✓Microsoft Visual Studio	Inconsistency of programming language, compiler and sources. Presence of Russian-language literature
I. V. Bychkova	Modern Programming Technologies	Basic algorithmic structures, two-dimensional graphics	✓Microsoft Visual Studio ✗Borland C++ Builder	Presence of Russian-language literature
V. V. Yefymenko	Multimedia systems, educational, game programs. Internet technologies and web design	Flash animation	✓Adobe Photoshop ✗Flash	Recommendation of the uCoz tool from a Russian manufacturer. Presence of Russian-language literature

introduction to Python. The course is free and designed STEM education. However, it lacks a clear focus on game development, making PyGame concepts harder to grasp. Additionally, the course structure and navigation are not user-friendly, which may hinder the learning experience.

DystOsvita [11], a Moodle-based distance learning platform, provides a structured PyGame course divided into five modules: a basic game project, event handling, Arkanoid, Mario, and Snake. It is free and suitable for independent learning, with well-organized theoretical explanations preceding practical tasks. However, the course lacks visual examples and has poor text formatting, making content harder to consume. Despite this, its step-by-step approach ensures gradual skill development.

IT Hub [12] offers a comprehensive PyGame course covering Python fundamentals, game development techniques, and project-based learning. It includes teamwork, a demo game, and a final presentation. However, the course is paid, costing 2,400 UAH per month for three months, which limits its accessibility. Without purchasing access, a further detailed evaluation of its content and methodology is difficult.

This analysis highlights key advantages and limitations in existing courses, providing insights for designing an improved PyGame curriculum that combines effective teaching methods while avoiding current shortcomings.

3. Development and approbation of the PyGame workshop

3.1. Choosing the software

For the workshop implementation, the Python programming language and the PyGame library were chosen, which is justified by the following factors:

- Python is one of the most popular programming languages in the world in various areas of development.
- Python is actively used in schools to teach the basics of algorithms and programming, which simplifies the process of integrating a new course into the curriculum.
- Python's easy syntax allows one to quickly master the basics of programming, which is especially important for students who are just starting to learn software development.

- PyGame is a set of Python modules designed for writing games, allowing the creation of fully functional games and multimedia programs.
- PyGame is highly portable and runs on almost every platform and operating system.
- PyGame has a simple core structure, and additional functions are managed by separate modules, which ensures code organization.
- PyGame uses the advantages of modern computer architectures by using multiple CPU cores for faster graphics rendering.

For the development of the course, the Google Sites platform was chosen as a convenient tool for creating a free website with a modern design and necessary functionality.

3.2. Designing the structure and content of the PyGame workshop

The workshop [13] was designed as a step-by-step course for creating the first game in PyGame, where each step is logically connected with the previous and next ones, ensuring a gradual progress. Unlike the discussed extracurricular programs, we decided to implement a project-based learning technique. The central idea was to make students build a functional yet simple game project, which could be used as a core for their further 2D games. We believe that it should increase their motivation and give an idea of how different game design techniques work together. Unlike the school Computer Science course, we could afford more flexible approach in extracurricular education. We expect that our students already know the fundamentals of Python programming.

The workshop begins with a "Main" page, which serves as an introductory one. It presents general information about the course, including the target audience and expected learning outcomes.

The next important part is the "Course" page, where users can familiarize themselves with the PyGame library in more detail. This page presents popular games developed using PyGame, demonstrating the library's capabilities and motivating students to learn. From this page, users can proceed either to the first step or directly to the second if they already have a configured development environment.

The first step of the workshop is devoted to setting up the development environment. It describes in detail the process of installing or updating Python and the PyGame library.

The second step focuses on creating the game world. At this stage, users learn the basic structure of project files and write the initial game code.

In the third step, characters are added to the game. Users learn to create a player and three different types of enemies: fire, ice, and poison.

The fourth step is devoted to creating game objects. At this stage, various elements are added to the game: lava blocks, first-aid kits, and ammunition.

The fifth step is the most complex and covers the implementation of game mechanics. At this stage, users learn to create movement animation, program interaction with objects, implement a shooting system, and manage the health of characters.

The sixth step is the final stage of game creation. Here, users add win and loss menus, improve the game world, and add final touches to the project.

In addition to the main content, the workshop contains a "Resources" page, where all the materials used in the course are arranged: images, videos, and full game code on GitHub [14].

3.3. Approbation of the course

To verify the feasibility, relevance, and effectiveness of the developed course, access to its materials was provided to first-year students of group I-24 who study Python within the "Introduction to Programming" course.

After completing the workshop, the participants completed a feedback survey, which helped gather information about their impressions, the complexity of the materials, the level of interest, and satisfaction with the final results. The survey consisted of two parts: quantitative and qualitative indicators for evaluating the course.

The quantitative assessment on a five-point scale (where 1 is the lowest indicator, 5 is the highest) included the following:

- Compliance of the course with expectations (average score 4.6).
- Difficulty of the educational material (average score 2.6).
- Quality of demonstration examples (average score 4.7).
- Level of interest in the process of creating one's own game (average score 4.7).
- Satisfaction with the result (created game) (average score 4.5).
- Level of interest in working with Pygame (average score 4).
- Desire to continue studying game development in Pygame (average score 4.4).

The qualitative assessment through open questions included the following:

- The most positive aspects of the course.
- Elements of the course that need improvement.
- The most useful components of the course.
- Components that can be optimized or excluded.
- General impressions and suggestions for improvement.

According to the results of the survey, most of the students positively evaluated the course. High scores were given to the quality of demonstration examples, the interest in creating their own game, and satisfaction in working with the Pygame library. Participants noted that the course met their expectations and became a useful and interesting experience for them.

Although some constructive suggestions were received for improving certain aspects of the course, the overall results of the approbation demonstrate the high effectiveness of the workshop. Students noted the practical value of the course, especially in the context of creating their own game projects, which indicates the achievement of the main educational goals.

The conducted approbation confirmed the feasibility of introducing the developed course into the educational process and its compliance with the modern educational needs of students in the field of game development using PyGame.

4. Conclusions

The analysis of educational programs from the Ministry of Education and Science of Ukraine reveals a lack of specialized game development courses for extracurricular education, despite high student interest. Existing programs require significant revisions and suffer from issues like inconsistencies in programming resources, outdated materials, high costs, and the inclusion of russian literature, which is inappropriate in the current context.

We found that Python and the PyGame library are the optimal solution for learning game development. Ease of use, a large number of educational materials, support for different platforms, and wide functionality make PyGame accessible for beginners and convenient for the implementation of game projects.

The development of the workshop structure was based on the principle of step-by-step learning, where each step is logically connected with the previous one. The structure of the course takes into account the interests of students, providing a practical orientation and accessibility of materials such as code, images, and instructions.

Approbation of the course confirmed its relevance and effectiveness. Students positively assessed the accessibility of the material, the quality of demonstration examples, and the opportunity to create their own game. The course met the expectations of most participants, interested them in further studying game development, and became a useful practical experience.

The practical significance of the work lies in the fact that the course can be integrated into coding clubs, groups, or individual training, helping schoolchildren develop logical thinking, creativity, and basic knowledge of game development.

The results of the work demonstrate that the development of such courses is an important step for improving the system of extracurricular education, its adaptation to the interests of students, and trends in the development of information technologies.

Declaration on Generative AI

During the preparation of this work, the authors used Claude 3.7 Sonnet to enhance content and improve writing style. After using this tool, the authors reviewed and edited the content as needed and took full responsibility for the publication's content.

References

[1] F. Berger, W. Müller, Towards an Open Source Game Engine for Teaching and Research, in: Z. Pan, A. D. Cheok, W. Müller, M. Chang, M. Zhang (Eds.), Transactions on Edutainment VIII, volume 7220 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 69–76. doi:10.1007/978-3-642-31439-1_7.

[2] S. Jacobs, Building an education ecology on serious game design and development for the one laptop per child and sugar platforms: A service learning course builds a base for peer mentoring, cooperative education internships and sponsored research, in: 2nd International IEEE Consumer Electronic Society Games Innovation Conference, ICE-GIC 2010, 2010. doi:10.1109/ICEGIC.2010.5716882.

[3] O. M. Haranin, N. V. Moiseienko, Adaptive artificial intelligence in RPG-game on the Unity game engine, in: A. E. Kiv, S. O. Semerikov, V. N. Soloviev, A. M. Striuk (Eds.), Proceedings of the 1st Student Workshop on Computer Science & Software Engineering, Kryvyi Rih, Ukraine, November 30, 2018, volume 2292 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2018, pp. 143–150. URL: https://ceur-ws.org/Vol-2292/paper16.pdf.

[4] O. O. Katsko, N. V. Moiseienko, Development computer games on the Unity game engine for research of elements of the cognitive thinking in the playing process, in: A. E. Kiv, S. O. Semerikov, V. N. Soloviev, A. M. Striuk (Eds.), Proceedings of the 1st Student Workshop on Computer Science & Software Engineering, Kryvyi Rih, Ukraine, November 30, 2018, volume 2292 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2018, pp. 151–155. URL: https://ceur-ws.org/Vol-2292/paper17.pdf.

[5] N. V. Moiseienko, M. V. Moiseienko, V. S. Kuznetsov, B. A. Rostalny, A. E. Kiv, Teaching computer game development with Unity engine: a case study, in: O. Y. Burov, S. H. Lytvynova, S. O. Semerikov, Y. V. Yechkalo (Eds.), Proceedings of the VII International Workshop on Professional Retraining and Life-Long Learning using ICT: Person-oriented Approach (3L-Person 2022), Virtual Event, Kryvyi Rih, Ukraine, October 25, 2022, volume 3482 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022, pp. 237–251. URL: https://ceur-ws.org/Vol-3482/paper330.pdf.

[6] Verkhovna Rada of Ukraine, Pro pozashkilnu osvitu: Zakon Ukrainy 07.12.2000r. No. 1841-III, 2000. URL: http://surl.li/tvnoyr.

[7] Instytut modernizatsii zmistu osvity, Normatyvna baza pozashkilnoi osvity ta vykhovnoi roboty, 2025. URL: http://surl.li/tizvjl.

[8] Dnipropetrovske viddilennia Maloi akademii nauk Ukrainy, Konkurs-zakhyst, 2024. URL: http://surl.li/mdabsj.

[9] V. O. Bykov, I. V. Bychkova, D. S. Hrebeniuk, Z. V. Deineko, Y. S. Dorofieieva, V. V. Yefymenko, D. S. Onishchenko, O. M. Pomazun, Navchalni prohramy z pozashkilnoi osvity. Doslidnytsko-eksperymentalnyi napriam. Kompiuterni nauky, Natsionalnyi tsentr "Mala akademiia nauk Ukrainy", Kyiv, 2021. URL: http://surl.li/puzhjb.

[10] Formula - brand of the publishing house 'Lingvist', Pygame: more than just games, 2024. URL: http://surl.li/cpqeil.

[11] DystOsvita, Python and pygame course, 2024. URL: http://surl.li/wsrmzf.

[12] IT Hub, Game development in python, 2024. URL: http://surl.li/owhxsi.

[13] Workshop, Pygame course, 2024. URL: https://sites.google.com/view/kurspopygame/home.

[14] Alexandr019, Project on github - pygame, 2024. URL: https://github.com/Alexandr019/pygame.