engineering proceedings

MDPI

Proceeding Paper

Design and Implementation of Educational Game Using Crossword Principles †

Atanaska Bosakova-Ardenska * and Dimitar Andreev

Department Computer Systems and Technologies, University of Food Technologies, 4002 Plovdiv, Bulgaria; dimitar.andreev25@gmail.com

* Correspondence: a_ardenska@uft-plovdiv.bg

† Presented at the International Conference on Electronics, Engineering Physics and Earth Science (EEPES'24), Kavala, Greece, 19–21 June 2024.

Abstract: In this paper, we developed an educational game with a graphical user interface that uses crossword principles. The architecture of the game is designed using a three-layer software architecture. We developed algorithms for dynamic words' selection for crossword building. The game is implemented using C#, and it consists of two main modules: one for teachers and the other for students. The developed game is applied in the education of computer engineers in order to assess students' perceptions of basic terms in C/C++ programming. A survey of students' interaction with the developed game is conducted in order to evaluate their perception of the gamification the education processes in the field of computer programming. The results indicate that usage of the computer-based crossword accelerates the educational process during the semester, and the game is a useful tool for motivating study throughout the whole academic year.

Keywords: programming; gamification; educational process; C#; crossword

check for updates

Citation: Bosakova-Ardenska, A.; Andreev, D. Design and Implementation of Educational Game Using Crossword Principles. *Eng. Proc.* **2024**, *70*, 12. https://doi.org/10.3390/engproc2024070012

Academic Editor: Grigor Mihaylov

Published: 29 July 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The advantages of games in education are well known in all stages of human development and throughout our lives, beginning with the appearance of every new human life. Every game has a set of rules that forms the basis for game process organization and which often includes a sequence of steps that have to be performed in order to play the game. Thus, a series of actions could be easily remembered and reproduced as part of the game process [1,2]. A lot of medical and psychological research indicates that games play a positive role in memory enhancement due to their effect on the human brain with respect to the activation of relations among neural centers and the interactions among the player, which improve cognitive functions [3]. The advancement of computer-based technologies involves a lot of hardware and software resources in the gaming industry, which has led to the very fast evolution of computer games; thus, they attract a very large audience. Education in many different sciences uses computer games as a part of the whole educational process in order to motivate students to study. As a tool for the development of knowledge in different sciences, computer-based crosswords are also well known. They are popular in a wide range of educational programs such as medicine [4–6], pharmacology [7], stomatology, dental medicine [8], language courses [9,10], economic disciplines [11], history [12], and many others.

In the last decade, engineering education has become more and more significant regarding its importance for industry development and its application in our everyday life. Thus, engineering education is going to evolve with respect to contemporary problems related to energy consumption and environmental safety, and all engineers of the future will have to implement "green thinking" and sustainable concepts, exploiting a rich base of knowledge [13]. Consequently, all of the achievements of contemporary science

and technology should support engineering education in order to improve the quality of education and to ensure that its positive effects are felt in all technological spheres. Nowadays, computer-based technologies and specialized software products are an essential part of engineering education and practice. Therefore, modern educational technologies are obligatory in all spheres of engineering education. As part of the recent trend towards digitalization, computer games are also becoming popular in engineering education, and especially in the education of computer and software engineers [14,15]. Such games attract young people and support their concentration in studying complex theoretical problems that are a fundamental part of engineering knowledge. In this way, a greater amount of knowledge will be accessible to future engineers, which will extend their opportunities for professional development and success in their chosen work fields, solving future problems for humanity.

The main aim behind the current research is to stimulate students' activity during the semester by applying modern educational technologies and, in particular, by using a recently developed crossword game to motivate the study of the fundamental details of programming as a part of students' education in computer and software engineering.

2. Materials and Methods

The design of the software products is commonly based on patterns that are established in practice, which allow for the design process to have a clear and persistent structure. The most popular patterns for software design are layered due to their logical consistency and opportunities for the easy support and development of the designed products [16]. The preferred software design patterns (models) define an architecture that contains three or four layers, which allow for a logical structure with clearly separated modules for user interactions (user interface), data storage (database), and data processing. Such models continue to be popular nowadays in the context of a wide variety of information systems, frameworks, and software products, such as machine learning applications [17], multi-robot systems with applications in power line inspection [18], web-based systems [19], and many others. A three-layered software design model was chosen for the purpose of the current research. The architecture of the designed educational game, the algorithms for word choice, the data that are used for testing the developed application, and the participants in the survey regarding the developed crossword game are discussed in the next four subsections.

2.1. Architecture of the Crossword Game

The architecture of the designed application is based on a three-layer software design model that incorporates all necessary modules to serve two types of users: students and teachers (Figure 1). All administrative modules are available only to users who have teacher accounts. Students can only use crosswords, which are generated dynamically, with a horizontal or vertical orientation of the main word which is crossed by the other words. The user interface and contents of the crossword could be visualized in either Bulgarian or English.

The generation of the crossword is based on the stored data in the files that build the data layer. The words and questions for their guessing that are used for crossword building are stored in two files: one for Bulgarian content and the other for English content. The file for settings contains information about the default language for crossword generation and the user interface, the maximum length of two types of words (long and short), the mode for working (students or teacher), and the number of words for crossword generation (Table 1). As a summary, the words for crossword generation are divided into two groups, the "long words" group and the "short words" group, which depends on setting the minimum length of a word which is of the "long word" type.

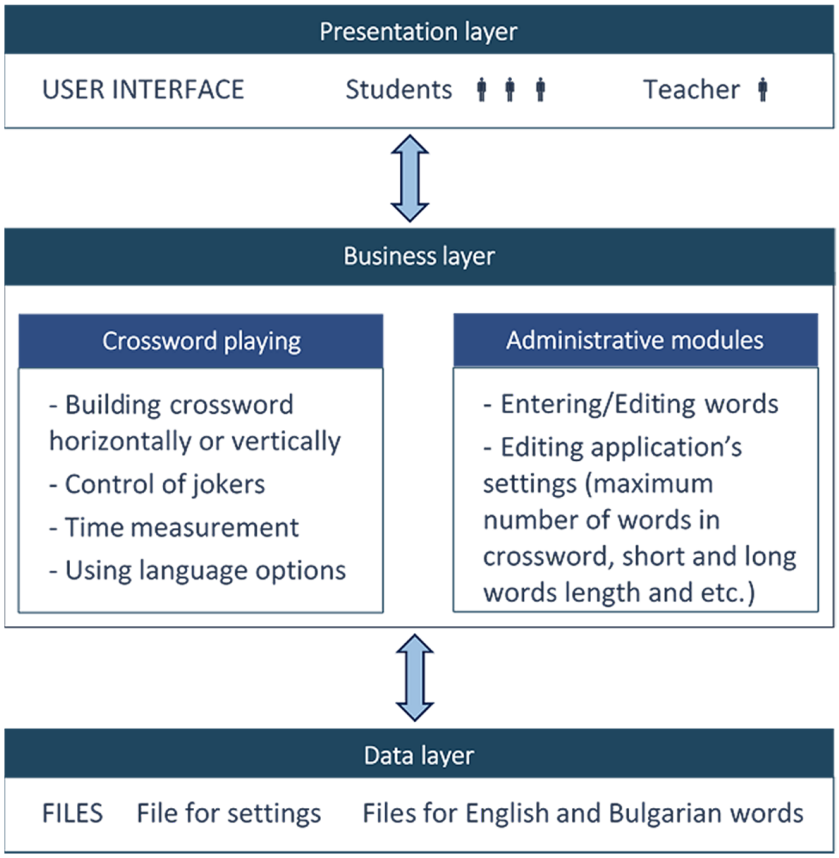


Figure 1. Crossword game architecture.

Table 1. Structure of file for settings.

Parameter	Description
LongWordsMin	Minimum number of characters for long-type words
LongWordsMax	Maximum number of characters for long-type words
ShortWordsMax	Maximum number of characters for short-type words
Mode	Sets the working mode, i.e., for teacher or students
Language	Sets the language for user interface and crossword generation
ShortWordsNumber	Maximum number of short-type words that have to be chosen for crossword generation

The crossword game has a long history and a variety of types; regarding the shape that is used for crossword puzzle arrangement, there are different possibilities, and one of the most popular is a grid containing crossing words, so the beginning of every word is numbered in order to correspond with the appropriate clues [20]. This variant, but containing one main word that is crossed by other words, is chosen for the designed game. When a crossword has to be built, there are two main concepts regarding the constraints that exist for the variables and domains of the words: the first concept is based on the assumption that all variables are numbered and directed and that they could be assigned to some of possible words in the domain; the second concept is based on the assumption that every variable is a word that is built of squares, and every square could contain one of the alphabetic characters [21]. In the current research, we have chosen the second concept and have thus described the settings for the maximum length of both long and short words (Table 1) necessary to define a rectangular area with squares for crossword visualization.

The files for words (part of the data layer) are encrypted, and they are built from couples: <clues, word>. The clues are given by a text written as a question or as some sentence that has to be completed which also has the meaning of a question, and because of this, their section starts with letter "Q". Every word is the answer to some question (it is part of the couple: <clues, word>); thus, its sections start with the letter "A". Using the words that are stored in the data layer involves the performance of decryption operations in order to access their content in the business layer of the application. Depending on the user mode (student or teacher), decrypted words are presented as a crossword (student mode) or as a list of questions and answers that could be edited (teacher mode). When the student mode is used, the application then deals with data only in the reading mode, but the performance of the teacher mode is related to data reading and writing. Thus, the implementation of the student mode utilizes only decryption operations, but the implementation of the teacher mode utilizes decryption/encryption operations. A simple encryption method is chosen for data encryption due to its effectiveness and fast performance. The implemented encryption algorithm is based on the cipher of Cesar, which uses a substitution of every letter with another one that has a number assigned to the alphabet which decreases with a constant value [22], but it utilizes additional rules depending on the position of every letter in the processed string.

2.2. Algorithms for Words' Selection

There are two algorithms designed that deal with words in order to select words for crossword generation. These algorithms are designed for C# implementation; thus, they use some structures typical for this language, such as the dictionary [23]. The first algorithm is used to select long words and to determine their unique letters (Figure 2). When it starts, a file is opened which contains the questions and answers needed for the crossword. The information from this file is read, decrypted, and written into the fileContent array. After the array is filled, the questions and answers should be added to a previously created dictionary (dicFileContent). It is known that the file alternates between question and answer, filling in the dictionary with odd values (words) from the array as keys for the dictionary and even values (questions) as their values. This is necessary in order that the questions can be easily accessed at a later stage. When finished with completing the dictionary, the program should divide the words into two lists: shortWords and longWords. At each step of the array processing, whether the array element starts with "A" is checked. If the element meets this condition, its length is checked. In cases where the length is less than the LongWordsMin, an element is added to the shortWords list; otherwise, it is added to the longWords list. When the processing of the array is over and the words are distributed across the lists, the next step is the long word choice for the crossword puzzle. This is achieved randomly using the Random class (C#).

When a long word is selected, its unique letters are determined. This is achieved by going through the word and checking for each of its letters to determine whether they are contained in a pre-created dictionary (dicLetters). If the letter is not contained in the dictionary, it is added to it as a key value and take 1 as its value. If the letter is already present in the dictionary, the value is increased by 1.

When the short word selection algorithm is started, the dicLetters dictionary is processed (Figure 3). At each step of the process, the current letter is written as a key value to another ListDict dictionary, and as a value for each key, a list is created.

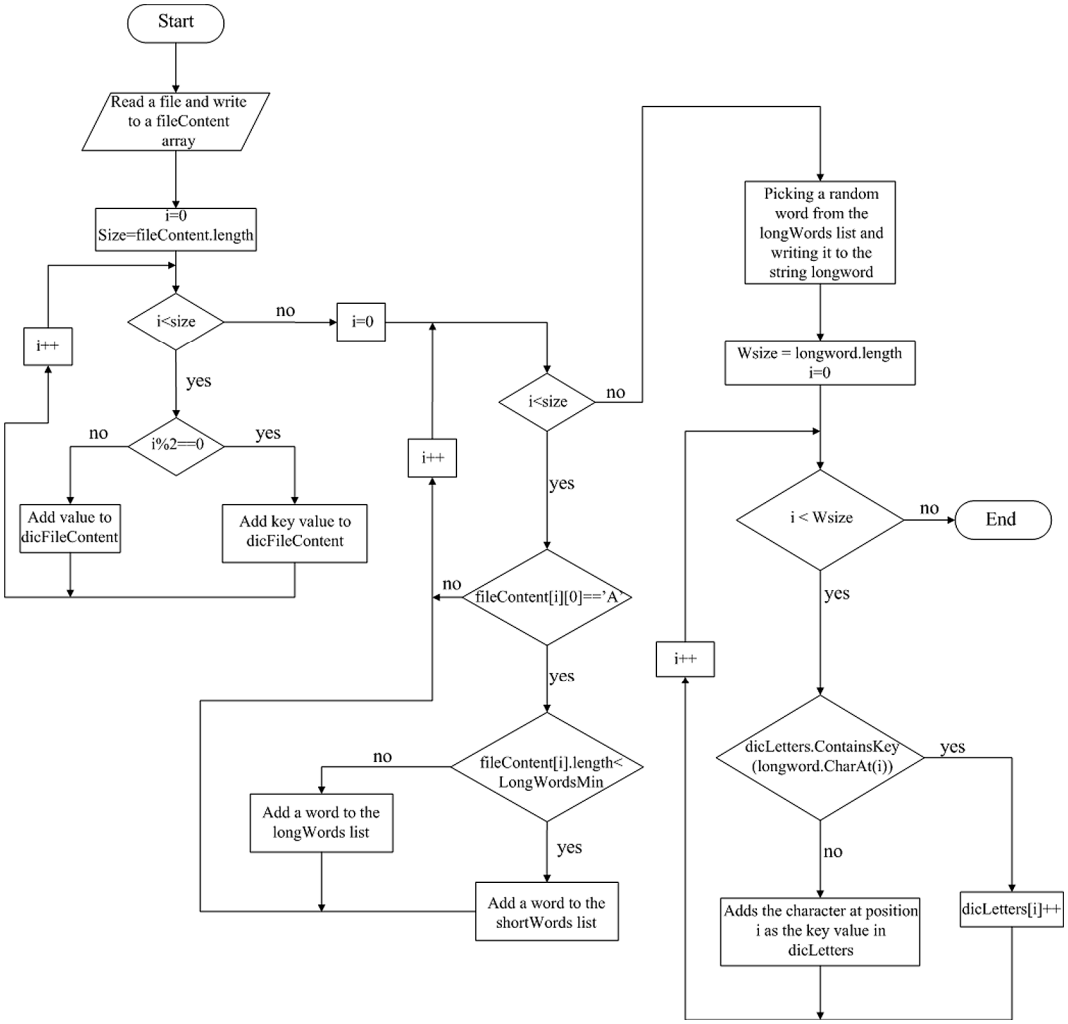


Figure 2. Algorithm for long word choice and unique letters definition.

When the processing is complete, the lists from the ListDict dictionary will be filled with words containing the key (letter) of the given list. This is achieved by processing the shortWords list. While processing it, each word is checked to see if it contains any of the ListDict key values. If the key value is contained in the shortWords, it is added to the key value list. When all the words have been checked and added to the corresponding lists, the algorithm finishes its work. After the algorithm has finished its work, the ListDict is sorted according to the number of words in the lists. This is undertaken in order to select words first for the letters for which the list of words is shorter. The already-sorted dictionary is saved using the name SortedListDict.

Once the dictionary containing all the selected short words that meet the condition of containing one of the unique letter origins from the selected long word is prepared (using the algorithm shown on Figure 3), it is processed, starting from the rarest letter in order to find the necessary number of short words for crossword building. The election of words for every key from the SortedListDict is undertaken using again the Random class. Once the word is selected, it is removed from all lists of words with unique letters in order to prevent multiple selections of the same word. Thus, the process of word selection is dynamic and uses randomization so the content of the generated crossword differs among multiple runs of the game.

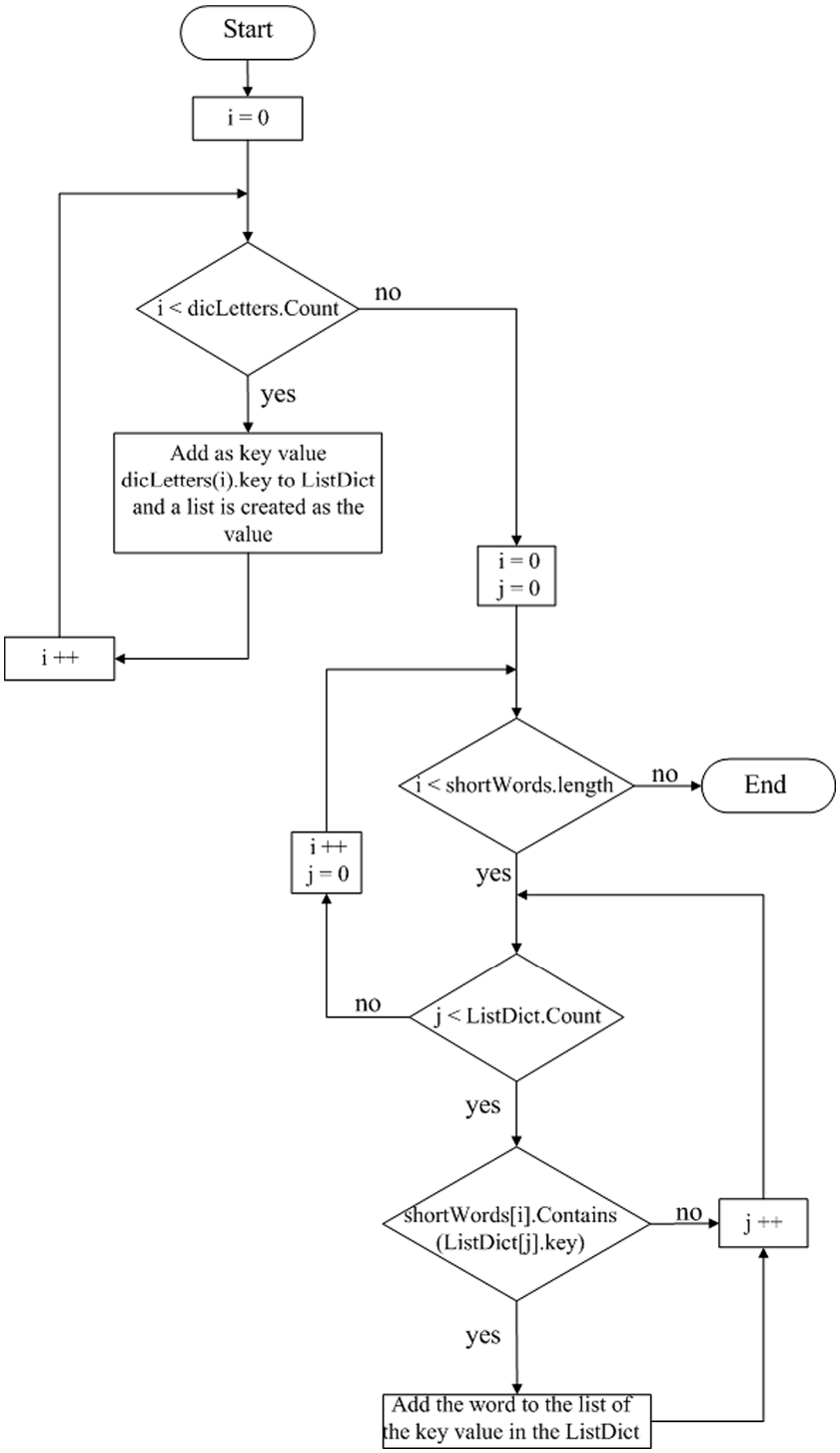


Figure 3. Algorithm for short words preparation.

2.3. Data for Game Verification

As two of the most popular programming languages, C and C++ are widely exploited in different study programs related to computer techniques and technologies. Thus, these languages are included in the teaching program in the course "Synthesis and Analysis of Algorithms", which is part of the bachelor program for computer engineers who study "Computer Systems and Technologies" at University of Food Technologies, Plovdiv (Bulgaria). A set of basis terms related to C/C++ fundamentals and to data structures is formed in order to examine the performance of the developed game. Some of the terms

used are variable, constant, array, pointer, function, increment, and decrement (C/C++ fundamentals), as well as stack, queue, graph, list, and tree (data structures fundamentals).

2.4. Survey Settings

The respondents of the survey are students in their third year specializing in "Computer Systems and Technologies" at the University of Food Technologies. These students passed "Program languages" and "Programming and computer applications" courses that are related to C/C++ programming and, currently, they are studying the "Synthesis and Analysis of Algorithms" course. These students passed the section "Data structures" in theoretical and practical classes before experimental usage of the developed crossword game.

The survey about student experience regarding the interaction with the developed crossword game is performed using a Google Form that contains three questions (Table 2).

Table 2. Survey questions.

Question	Answer Options
Did you participate in solving the Synthesis and Analysis of Algorithms crossword puzzle?	Yes or No
Do you think that solving the Synthesis and Analysis of Algorithms crossword helps your study in the discipline?	Yes; No; I can't decide
Would you like to solve similar crossword puzzles in other disciplines in the field of programming?	Yes; No; I can't decide

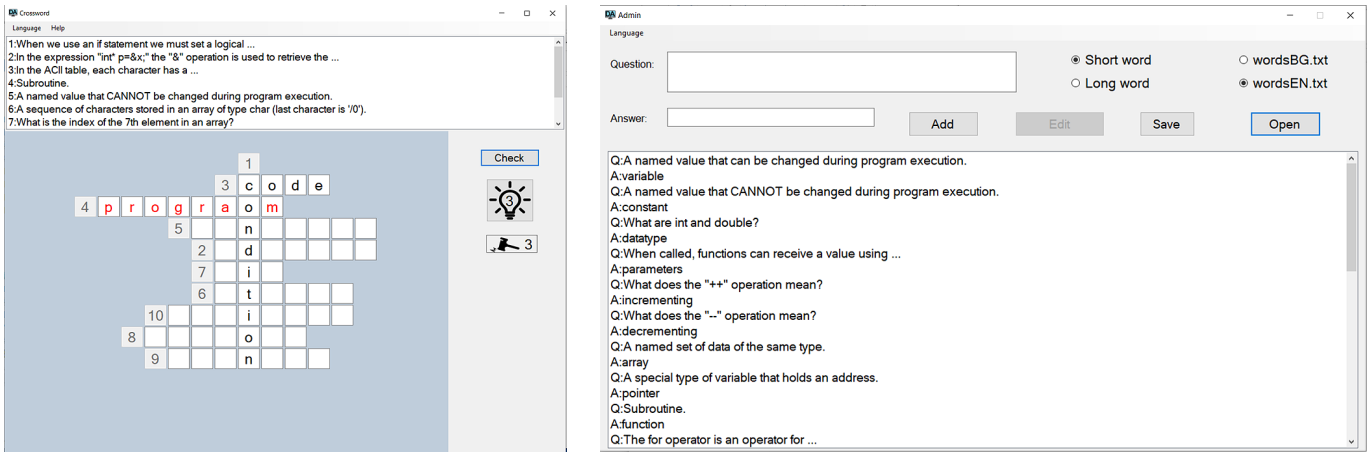
The aim of the survey is to evaluate the overall impression of the developed crossword game on the students. The results of the survey will indicate whether the developed game could be applied to other courses related to programming or not.

3. Results and Discussions

Based on the discussed software architecture and algorithms for words selection, an application on C# is developed and named DACrossword. This application is applied as an additional tool for students' advancement assessment during the winter semester of the 2023–2024 academic year. The appearance of the developed application and its impression on the students are discussed in the next two subsections.

3.1. DACrossword Application

The developed game is implemented using two main windows: one for students mode and second for teachers (Figure 4). The students mode window visualizes the dynamically generated crossword with vertical or horizontal orientation of the long word (in Figure 4a, vertical orientation of the long word is presented), a list of clues that are numbered according to the numbers of the words that have to be guessed (upper part of the window), a button for crossword check, and two types of jokers (located beneath the button), with an opportunity for triple usage. The options for crossword orientation are visualized before crossword generation and vertical or horizontal orientation can be chosen. The jokers are visualized as a lightbulb and as a small hummer, and their action are related to showing a random unguessed letter (wrong letter or unwritten letter, i.e., an empty square filling) and to showing a letter selected by the user that is unguessed, respectively. When the check button is used, all the words entered by the user are compared with the correct answers, and those letters that are incorrect are colored red. After crossword checking, a message about the number of unguessed words and the elapsed time is shown. If the crossword is solved (all words are guessed), then the timer for the elapsed time is stopped.



(**a**) (**b**)

Figure 4. Graphical user interface (GUI) of DACrossword game: (**a**) students mode; (**b**) teachers mode.

The teachers-mode window (Figure 4b) can be visualized only if a teacher account is used, and it contains a list for file content, fields for question and answer, buttons (Add, Edit, Save, and Open), and options for type of word (long or short) and files selection (Bulgarian or English version). Files for words and clues are encrypted, so the action of the Open button includes the decryption of the content for the selected file and visualization of the decrypted content into the list (located at the bottom of the window). The fields Question and Answer can be used for the addition of a new couple <clues, word> or for the addition of the existing couple <clues, word>. The content of the working file can be stored using the Save button or when the teachers-mode window is closing. The process of file saving includes the encoding of the content that is shown in the list of couples <clues, word>.

3.2. User Expirience

The developed game (DACrossword) is used as a part of the practical classes in "Synthesis and Analysis of Algorithms" (Figure 5).

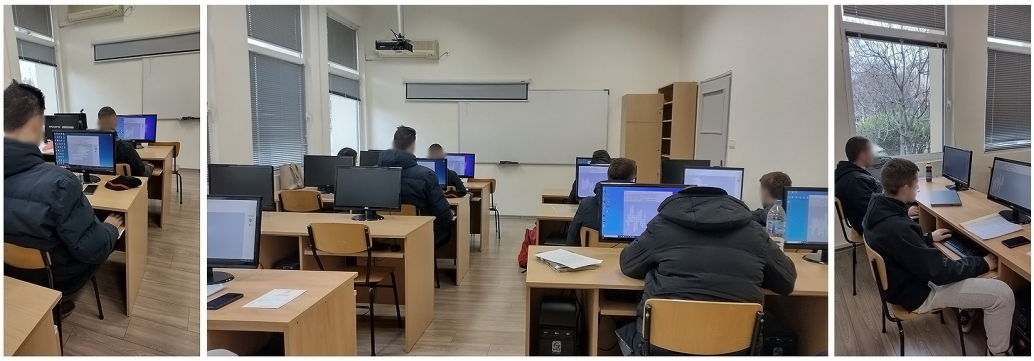


Figure 5. Participants in educational game testing.

The opinions of students are collected using a Goggle Form, and the results are presented in Figure 6. It is observed that 90% of the participants are satisfied with their interaction with DACrossword and would even prefer to use this application in other courses related to programming. Only 10% of students in the discipline "Synthesis and Analysis of Algorithms" could not decide whether DACrossword helped them to study, and these students probably did not participate in crossword solving (10% of respondents said "No" to the first question of the survey, which is shown on Figure 6a).

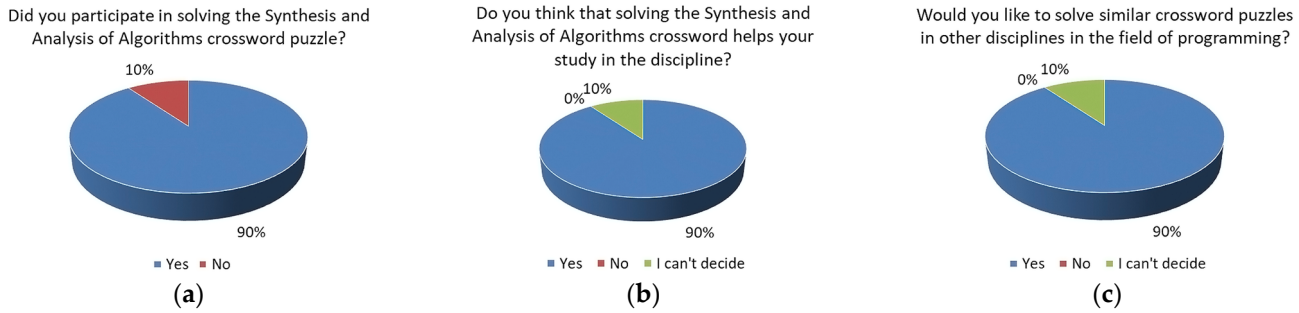


Figure 6. Survey results.

It could be concluded that the students' overall impression of the developed educational game is positive, and such innovations have to be introduced in other courses for computer engineers.

4. Conclusions

Gamification of the educational process has a long history in a variety of scientific areas due to its advantages related to easy memory terms, a series of actions, and the activation of corresponding brain centers. Because computers play a very significant role in everyday life, they are also playing a leading role in the game industry, and especially in educational game development. Thus, current research corresponds to contemporary trends in education. The advantages of the developed game are its fast execution (it is compiled into an executable file), its ease in supporting program logic due to its design based on multilayer architecture, the opportunity to edit the content in crossword generation, the diversity of crosswords based on dynamic generation, the protection of the content thanks to the usage of an encoding algorithm, etc. The developed application (DACrossword) has also been experimentally evaluated by students, and the results are encouraging. This educational game has the potential to be an estimation tool due to the protection of its content, and especially if a large amount of questions and answers are stored, which significantly increases the amount of crossword variants.

As to future works related to the discussed educational game, the opportunity for statistical integration that will present information concerning hard words (terms), average time taken to solve the crosswords, etc., has been noticed. A detailed evaluation of user interaction with jokers is also important for overall user experience enhancement. Another direction for future work could be the performance of a survey of teachers' experience regarding the game's teachers mode.

Author Contributions: Conceptualization, methodology, software, validation, formal analysis, investigation, A.B.-A. and D.A.; resources, A.B.-A.; data curation, A.B.-A.; writing—original draft preparation, A.B.-A. and D.A.; writing—review and editing, A.B.-A.; visualization, A.B.-A. and D.A.; supervision, A.B.-A.; project administration, A.B.-A.; funding acquisition, A.B.-A. All authors have read and agreed to the published version of the manuscript.

Funding: The APC was funded by the University of Food Technologies (Science Fund).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article and can be provided by authors on request.

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Chou, M.J. Board Games Play Matters: A Rethinking on Children's Aesthetic Experience and Interpersonal Understanding. *Eurasia J. Math. Sci. Technol. Educ.* **2017**, *13*, 2405–2421. [CrossRef]
2. Martinez, L.; Gimenes, M.; Lambert, E. Video games and board games: Effects of playing practice on cognition. *PLoS ONE* **2023**, *18*, e0283654. [CrossRef] [PubMed]
3. Al-Thaqib, A.; Al-Sultan, F.; Al-Zahrani, A.; Al-Kahtani, F.; Al-Regaiey, K.; Iqbal, M.; Bashir, S. Brain Training Games Enhance Cognitive Function in Healthy Subjects. *Med. Sci. Monit. Basic Res.* **2018**, *24*, 63–69. [CrossRef] [PubMed]
4. van Gaalen, A.E.J.; Brouwer, J.; Schönrock-Adema, J.; Bouwkamp-Timmer, T.; Jaarsma, A.D.C.; Georgiadis, J.R. Gamification of health professions education: A systematic review. *Adv. Health Sci. Educ. Theory Pract.* **2021**, *26*, 683–711. [CrossRef] [PubMed]
5. Agarwal, H.K.; Singhal, A.; Yadav, A.K. Crossword puzzle: An innovative assessment tool to improve learning of students in Forensic Medicine. *Med.-Leg. Update* **2020**, *20*, 18–22. [CrossRef]
6. Zamani, P.; Biparva Haghighi, S.; Ravanbakhsh, M. The use of crossword puzzles as an educational tool. *J. Adv. Med. Educ. Prof.* **2021**, *9*, 102–108. [CrossRef] [PubMed]
7. Singh Matreja, P.; Kaur, J.; Yadav, L. Acceptability of the use of crossword puzzles as an assessment method in Pharmacology. *J. Adv. Med. Educ. Prof.* **2021**, *9*, 154–159. [CrossRef] [PubMed]
8. Nirmal, L.; Muthu, M.S.; Prasad, M. Use of Puzzles as an Effective Teaching-Learning Method for Dental Undergraduates. *Int. J. Clin. Pediatr. Dent.* **2020**, *13*, 606–610. [CrossRef] [PubMed]
9. Bella, Y.D.; Rahayu, E.M. The Improving of the Student's Vocabulary Achievement through Crossword Game in the New Normal Era. *Edunesia J. Ilm. Pendidik.* **2023**, *4*, 830–842. [CrossRef]
10. Rahayu, A.C.; WL, M.Y.; Sophia, T.C. The use of crossword puzzle in teaching English vocabulary. *Teach. Lit. Linguist. (ETERNAL)* **2021**, *1*, 271–283.
11. Saputri, A.; Sukirno, S. Students' Perceptions of Crossword Puzzles Media Implementation in Accounting Learning. In Proceedings of the International Conference of Ethics on Business, Economics, and Social Science (ICEBESS 2018), Yogyakarta, Indonesia, 31 August–1 September 2018; pp. 108–112. [CrossRef]
12. Zirawaga, V.S.; Olusanya, A.I.; Maduku, T. Gaming in education: Using games as a support tool to teach History. *J. Educ. Pract.* **2017**, *8*, 55–64.
13. Diaz Lantada, A. Engineering Education 5.0: Continuously Evolving Engineering Education. *Int. J. Eng. Educ.* **2020**, *36*, 1814–1832.
14. de Almeida Souza, M.R.; Constantino, K.F.; Veado, L.F.; Figueiredo, E.M.L. Gamification in Software Engineering Education: An Empirical Study. In Proceedings of the IEEE 30th Conference on Software Engineering Education and Training (CSEE&T), Savannah, GA, USA, 7–9 November 2017; pp. 276–284. [CrossRef]
15. Ivanova, G.; Kozov, V.; Zlatarov, P. Gamification in Software Engineering Education. In Proceedings of the 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 20–24 May 2019; pp. 1445–1450. [CrossRef]
16. Savolainen, J.; Myllarniemi, V. Layered architecture revisited—Comparison of research and practice. In Proceedings of the Joint Working IEEE/IFIP Conference on Software Architecture & European Conference on Software Architecture, Cambridge, UK, 14–17 September 2009; pp. 317–320. [CrossRef]
17. Vámossy, Z.; Romhányi, Á. Benefits of Layered Software Architecture in Machine Learning Applications. In Proceedings of the International Conference on Image Processing and Vision Engineering (IMPROVE 2021), Budapest, Hungary, 28–30 April 2021; pp. 66–72. [CrossRef]
18. Silano, G.; Bednar, J.; Nascimento, T.; Capitan, J.; Saska, M.; Ollero, A. A Multi-Layer Software Architecture for Aerial Cognitive Multi-Robot Systems in Power Line Inspection Tasks. In Proceedings of the International Conference on Unmanned Aircraft Systems (ICUAS), Athens, Greece, 15–18 June 2021; pp. 1624–1629. [CrossRef]
19. Raunak, M.S.; Kuhn, D.R.; Kacker, R. Combinatorial Testing of Full Text Search in Web Applications. In Proceedings of the IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C), Prague, Czech Republic, 25–29 July 2017; pp. 100–107. [CrossRef]
20. Feigenbaum, L.J. Crosswords at a Crossroad: The Puzzle Turns 100. What Is the Clue to Its Survival? Master's Thesis, Old Dominion University, Norfolk, VA, USA, 2013. [CrossRef]
21. Poole, D.L.; Mackworth, A.K. *Artificial Intelligence: Foundations of Computational Agents*; Cambridge University Press: Cambridge, UK, 2010; pp. 113–154.
22. Churchhouse, R.F. *Codes and Ciphers: Julius Caesar, the Enigma, and the Internet*; Cambridge University Press: Cambridge, UK, 2002; pp. 1–12. [CrossRef]
23. Microsoft Learn .NET API Browser System.Collections.Generic C#. Available online: https://learn.microsoft.com/en-us/dotnet/api/system.collections.generic.dictionary-2?view=net-8.0 (accessed on 21 May 2024).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.