Learning While Playing: Introducing Programming Concepts to Children in Minecraft

https://doi.org/10.3991/ijoe.v18i13.26451

Bashaer Alsaadi(✉), Bushra Alsaadi, Athir Alghamdi, Mashaal Alfhaid,
Nedaa Almuallim, Maram Meccawy
Information Systems Department, Faculty of Computing and Information Technology,
King Abdulaziz University, Jeddah, Saudi Arabia
bsaderalsaadi0001@stu.kau.edu.sa

Abstract—Children's interest in playing popular video games could be utilized for learning and educational purposes. The entertaining and interactive learning environments of digital games can facilitate the learning of a variety of challenging subjects. This paper presents the Creative Programmer, which is a Minecraft modification (Mod) that takes an advantage of the popularity of Minecraft game among children to teach them basic computer programming concepts. This mod targets children aged between (5–15). The effectiveness of this Mod was tested using a pre and post quizzes. The results indicate that children who did not have any previous knowledge of programming answered most of the test questions correctly, while the performance level of children who were familiar with programming improved in the post-quiz compared to the pre-quiz. Moreover, it has been observed that the familiarity of the Minecraft environment made learning more enjoyable and enhanced the knowledge gaining process.

Keywords—game-based learning, Minecraft, programming, e-learning

1 Introduction

The introduction of new technologies in our daily activities, has led to a dramatic rise in the number of children who are possess devices and have direct access to the Internet [1]. As a result, the new generation has become more involved in many online activities, including playing video games both online and offline, watching YouTube, and participating in social media networks. Video games attracts millions of children and have become an ubiquitous part of their lives. This phenomenon involves many games such as Fortnite [2], Angry Birds [3], Pokemon [4], and Minecraft [5]. In addition to the time children spend enjoying these games, the way of playing some of these games can improve their cognitive and mental skills that are required in the 21st century such as problem-solving and creativity [6–8]. Moreover, multiplayer games enhance children's emotion regulation, relationship-building abilities, and cooperation [9]. Children spent considerable time playing digital games, for example, children in Saudi Arabia spend an average of four hours daily playing video games [10].

The growing acceptance of these games and the considerable time spent playing them, attracted educator's attention to take advantage of these games for learning purposes. One of these popular games is Minecraft [5]. Minecraft is considered one of the most popular open-ended virtual games for children aged between 6–13 years old. It does not have a standard way of playing. However, it allows players to customize their world and create anything they can imagine which encourage their creativity. The use of Minecraft for learning was investigated by previous studies for different subjects. For example, it was used to teach mathematics concepts [11], artificial intelligence algorithms [12], and computer programming [13]. These studies proved the effectiveness of using Minecraft for education. However, most of them used it as a supportive tool for the traditional class learning to ensure the concepts are understood well or as an assessment method. Moreover, not all of them targeted children.

Therefore, this research aims to utilize children's energy, interest, and time in playing Minecraft to teach them basic programming concepts. In this paper, Minecraft modification which is also known as (Mod) is developed to serve as a stand-alone educational tool and is named as the Creative Programmer. The Mod takes advantage of the advanced levels that children reached in playing Minecraft to learn computational thinking and related skills. This Mod is then tested by seven children within the games age range to evaluate its effectiveness. The rest of the paper continues as follows. Section 2 presents background about gamification, game-based learning and other related concepts, as well as briefly describing some recent related works. Section 3 discusses the related studies in the literature. Section 4 provides detailed description about The Creative Programmer Mod. Section 5 demonstrates the testing for the proposed Mod and discusses the results. Finally, section 6 and 7 conclude this work by presenting conclusion, limitations and recommendations for future work.

2 Background

This section presents an overview of gamification and the role it plays in education. Then, it provides a brief description of Minecraft game. The section also highlights the importance of learning programming for children as well. It also explores recent research studies that used Minecraft as a tool to enhance the learning process.

2.1 Gamification and game-based learning

Gamification tends to have various definitions based on the context perspective, generally, it is described as the process of adopting game dynamics and mechanics in achieving a goal outside the gaming context [14]. Gamification attracts the interest of people from different domains and acquires increasing attention from researchers in both practice and academic fields. That is due to its effectiveness in increasing the level and quality of user engagement [15], boosting interactivity and competitivity among users, and improve the content's usefulness by adopting entertainment [16].

Gamification has many applications in almost every domain, such as health, marketing, and education. In terms of education, teachers are continuously pursuing to enhance student's engagement in the learning process and improve their academic performance by creating an interactive learning environment [17]. Game-based learning and gamification share using game mechanics and dynamics outside the gaming context. However, the first is about designing the learning activities for primary purposes like conveying concepts and acquiring skills in a game-like framework. On the other hand, gamification is more about increasing the participation and boosting competition by applying game elements [16, 18].

Many studies over the years were conducted to explore the effects of adopting gamification and game-based learning in the learning process and define the conditions, features and models that can maximize the effectiveness of this adoption [19]. Overall, those studies resulted in clarifying that gamifying the educational process contributes to increasing student's involvement in learning activities, promoting the interaction, boosting their motivation for learning, embracing the challenge incentive and guiding them towards better academic achievements, besides enhancing decision-making and thinking skills [20]; [21]; [22]; [23]. In this paper, the work falls under game-based learning as it aims to design learning activities in Minecraft game where children can play and learn at the same time.

2.2 Teaching computer programming skills to young children

Programming and computational thinking have become important skills to be taught for children [24]. Many studies approved that children have the ability to learn how to program at an early age, this gives them skills including collaboration, logical thinking and problem-solving in a fun and useful way [25, 26]. Many games have been developed to teach children those vital skills in entertaining ways, some of them are focused on teaching children program thinking, for example, Move the turtle [27], while others focus on teaching programming commands such as Scratch (Scratch, n.d.). The research presented in [28] argues that children should learn how to create their own programs and become digital innovators rather than being only consumers of whatever software that is available in the market. It introduces *The Little Programmer*, which is an application for mobile devices that introduces the basic programming concepts to Arab children aged (8–12) years. It introduces the new concepts through animated videos and let learners later test their knowledge through game-like quizzes. User trials results showed that the application was helpful and allowed its users to identify the concepts of computer programming. In addition, it showed that linking difficult or newer concepts to familiar entities and objects through visualization can have a positive impact on the learning.

The research presented in [29] believed that children can learn logic and programming concept at an early age even if they were not yet able to read. So, they had presented PiktoMir, which is a software available in Android and Google play with textless environment that supports teaching programming concepts for children at an early age without the need of reading and writing skills. It basically introduces a robot, whose behavior is controlled by the player by using several commands such as forward, loops, conditions etc. The results of the test showed that the game can be considered as one of the ways to support teaching programming concepts to preschoolers. Also, based on the teachers' observation, most of the children found it fun to play with the robot which provides a positive impact on the educational environment.

Another study [25] examined the use of ScratchJr to develop programming concepts and computational thinking in preschool children. It conducts an experiment on 43 preschoolers from both genders, that aims to measure children's' ability to create a complex project using ScratchJr blocks. Before preschoolers make a project, researchers first teach them how to play ScratchJr by providing basic tutorials. The results indicate that ScratchJr has proven its efficiency to teach preschool children programming concepts. Moreover, it found that preschoolers' gender has no effect on learning such concepts.

Many efforts in this area came from the video game industry, as many games have been developed to teach children programming skills. Examples include Box Island [30], Kodable [31], CODESPARK [32], CodeCombat [33] and ScratchJr [34], to name but few.

The results of the analysis of these industrial games show that the majority of them have been designed to be played in a standard way. This means players cannot interact with the games in their own ways. While in the Creative Programmer mod children are free to use their imagination in the way they play, they are also able to do other activities they are interested in, within the same mod, thanks to the nature of Minecraft. Hence, this will enhance their creative thinking skills, ensure enjoyment in learning via the game and reduce boredom of educational games.

All five of the mentioned industrial games are stand-alone games, children have to install them particularly for learning proposes. While Minecraft being a part of a popular game, children will be encouraged to get involved in the mod, since they are not restricted to play this mod only. They can move between several mods and enjoy playing in their preferred and usual way.

2.3 Minecraft

Original Minecraft: Minecraft [5] is a java-based sandbox game having an open-ended virtual world. It can be played using different platforms including Mac, Windows, Androids, IOS and PlayStation. However, there is no standard way of playing this game, in fact, it depends on the player's creativity. This means players have to explore the land by themselves and utilize its resources (e.g., box, trees) for their own purposes. Moreover, the playing style will be affected by Minecraft mods. Several mods exist in Minecraft, *Creativity* and *Survival mods* are an example of main mods developed in this game.

- Survival mod: players' main aim in this mod is to survive. Therefore, they must search and collect food, defense and housing resources. Otherwise, they could die from hunger, environmental damages or even attacks.
- Creativity mod: unlike Survival mod, players in this mod will not face any threats to their survival. They can use their imagination with an unlimited number of resources to build or destroy structures, in addition to their ability to fly.

Minecraft allows players to design and develop their own mod. They can release it to the public, so it can be accessed by everyone. However, there are different third-party tools and languages that can be used in mod development such as Java and Makecode [35].

Minecraft: education edition: In order to improve the learning process and make it more enjoyable for children, Minecraft company launched a special edition that targets the classroom community including teachers and students [36]. The education edition provided many features that support the educational experience such as Classroom Mode and Code Builder. The former enables teachers to monitor students' activities within the game whereas the latter helps the student in learning code skills. Regardless of all the benefits from this edition, children are not able to use it unless they become a part of an official institution that has Minecraft education edition license. Therefore, this paper proposed a solution that provides education within the original version of the game.

3 Related studies

The use of Minecraft in education has recently become an interesting research point, therefore, different studies have been conducted to utilize Minecraft in improving learning outcomes of school courses. It has been adapted in a number of schools in different countries; including American, Swedish and Canadian schools [37].

To increase children's awareness of modern computer science, research in [12] explores how Minecraft could be used in teaching Artificial Intelligence (AI) in a ten-week course. The aim of this study is to make the student aware of the production consequences regarding time and speed while designing and implementing AI algorithms. For this purpose, students were asked to implement their own algorithms through developing Minecraft Mods. By the end of this course, Minecraft proved it effectiveness for the AI course. In terms of math, another Minecraft mod was developed to serve mathematics [11]. Students in this mod have to work cooperatively to pass all the game levels which are Castle, Pyramid, and Rook. Players also have to kill zombies in order to produce numbers. Each level in the game has different learning outcomes such as using mathematical operations, learning how to order numbers and encourage students to cooperate to overcome difficulties. Another study [38]. examined the spatial geometry, which is a topic in mathematics that can be taught in the virtual world by using Minecraft. The study includes two classes; students in first class transfer the objects that they have drawn from the plane to space. While the other class let students scale some objects using Minecraft, where the students could see how the scaling in three dimensions can affect the spatial geometry. The results of the study show that Minecraft is a very effective and useful tool that could be used in teaching and learning mathematics. Likewise, a game-based Japanese course Kotoba Miners was developed for the purpose of teaching a foreign language [39]. It employs Minecraft mods to achieve lessons-defined objectives such as exposing new vocabulary meanings, cooperatively forming questions, conditionals, and sentences. In addition to learning the proper usage of verb tenses and prepositions. It gets positive feedback because it provides actual learning content in a more holistic, interactional, and enjoyable environment. Another study [13] conducted to improve the understanding of programming concepts by the non-programmer, through developing CodeBlock which is a mod within Minecraft. It is a virtual robot that players can control by dragging blocks that show programming commands (e.g. Forward, Backward, Turn Right, Destroy, etc.). These blocks represent instructions for the robot to follow. They found most participants were interested and enjoyed it when the robot was performing the command.

To sum up, the effectiveness of involving Minecraft in school courses to facilitate and enhance the learning process has been proved by the examined studies, regardless of the course content. However, in three of these studies [12]; [38]; [39], students will gain the educational concepts from the traditional classroom learning, and the mod acts only as a supportive tool to ensure the concepts are well understood or as an assessment method. Table 1 summarizes these studies in terms of publication year, publishing journal, the studied field and determines if it includes developing a special mod or not.

Table 1. Summarization of conducted researches about using Minecraft as an educational tool

Source	Year	Publisher	Field	Minecraft Mod
[38]	2017	IEEE Xplore	Spatial Geometry	x
[39]	2015	Peachpit Press	Japanese Language	✓
[11]	2014	Research Gate	Mathematics Concepts	✓
[13]	2013	Semantic Scholar	Computer Programming	✓
[12]	2012	IEEE Xplore	Artificial Intelligence Algorithms	✓

Regarding using Minecraft for teaching programming **[13]** proved its effectiveness in improving the understandability of programming concepts through controlling robot movements or solving the puzzles. CodeBlocks is directed towards non-programmers in general and not specifically for children, and it focuses only on the sequencing concept. However, the proposed mod targets children and is based on animations and original Minecraft game activities, which make it suitable even for children who are non-readers'. In addition, it includes more programming concepts like data types, algorithms, loops, etc.

4 The Creative Programmer Mod

This paper presents Creative Programmer Mod, which is an educational tool targeting children aged (5–15) years based on a conducted questionnaire that will be described in (Sect. 4.1). The mod is designed to achieve several objectives, which are to utilize children's playing time in learning by assuring that the academic performance of children is improving while playing. The Mod facilitates the understanding of the complicated concepts of programming. Moreover, it takes advantage of children's familiarity of Minecraft game including its environment and playing rules to teach them new technical skills while they are having enjoyable time.

4.1 Data gathering

A survey was conducted to understand the user need and preferences of using the proposed Mod. The survey was composed of seven questions that targeted the parents who have children playing Minecraft, with some questions to be answered on behalf of their children (For Survey Questions See Appendix A). The aim of using this questionnaire is to: (a) investigate how children interact with games in general; (b) know their preferences when playing Minecraft game; (c) explore their background in playing programming games and the benefits they gained from it. The survey was spread online, and the study sample chosen using snowball sampling, which is one of the non-probability sampling techniques. The sample was collected by firstly sharing the survey link with surrounding parents who have children in the targeted age range, which is between 5 and 15 years old, then those parents were asked to share that link with other parents in their social environment. The survey ended up receiving 87 responses. About 13.8% of investigated children were aged between 5-7, while 42.5% and 43.7% of them aged 8–10 and 11–15 respectively. In first section, parents were asked about how often their children play Minecraft, 48.3% of parents stated that their children played Minecraft most often (see Figure 1). Another question was posted in order to obtain the preferable way of interaction with games. 51.7% of children using textual instructions and animation in dual, while 42.5% preferred to interact with pictures and animated elements only.

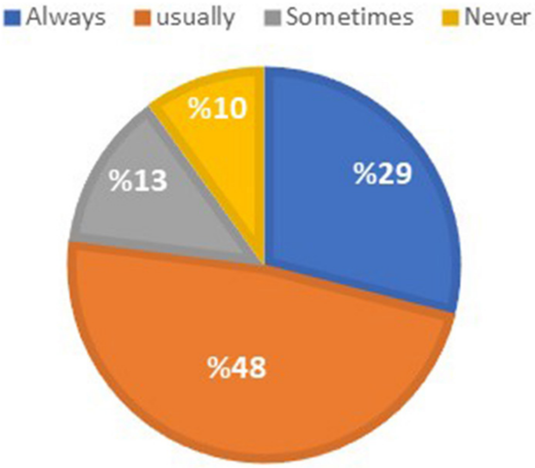


Fig. 1. Time spends playing Minecraft

Then, two questions were designed to define the children's preferences when playing Minecraft, regarding the mod type and their preferred activity. The results show that around half of children (53.8%) are willing to play in both survival and creativity mods. Regarding the playing activities – 42.3% of players like to build creative structures, 38.5% of them love both building, farming and caring of plants activities.

While the last two sections contained an open-ended question about the programming game. Regarding children's experience with programming-related games, 78% of sample children had not tried any programming-related games before (see Figure 2). Children who had tried such games enjoyed and had a good experience, while some benefited from that by developing their skills. A great example of this is that there is a child who started to provide services for the public using his own server and financially benefited from that, and another one started to program for other games

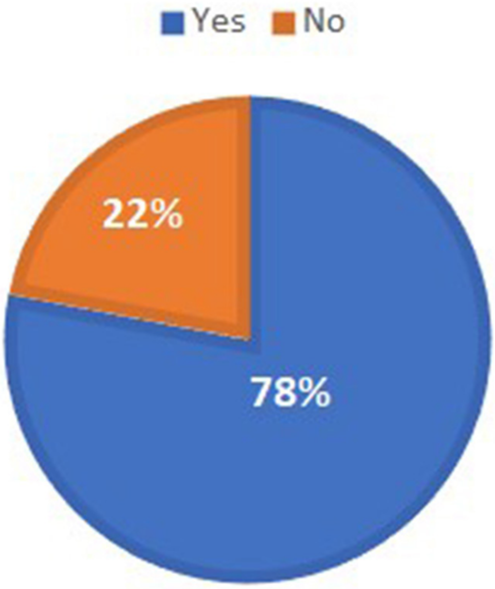


Fig. 2. Knowledge of children in programming-related games

Moreover, parents were asked about their opinion on introducing learning elements in popular games such as Minecraft, Fortnite, etc. Although Answers varied between supporters and opponents, there are more supporters than opponents; they think it is a good idea to use the children's time spent on these games into acquiring knowledge and benefits. Some of them see it as a good opportunity that leads to faster delivery of information, others believe that we are in a digital generation and there is a real need for learning computational skills. On the other hand, opponents justified their opinions by sharing their concerns about the amount of time the children will spend on the game.

After gathering the data, the requirements for developing a child-friendly mod were determined. The Creative Programmer Mod combines the usage of animation and simple textual commands keeping in mind the childrens' preferred activities on Minecraft. Creative Programmer targets children aged between 5 and 15 years, and as most of the children have no experience in programming, Creative Programmer will start from fundamental concepts and pave the way for advancement.

4.2 Programming approach

Teaching new concepts to children is a crucial, complex, and time-consuming process. Teaching beginner's computer programming is one example of such difficulties. It requires a well-defined approach to be followed in order to achieve desired outcomes in a reasonable time. The approach being followed in this research for the Creative Programmer mod is Top-Down Approach (TDA). Since some children are not proficient and skilled in programming concepts while others had never learned it, TDA approach has been chosen due to the shorter time it requires to gain programming knowledge comparing to other approaches such as Bottom-Up Approach (BUA) [40]. Also, it starts by introducing the abstracted programming concepts and emphasizing their usage by samples then going into implementation details [40].

4.3 Mod design

In this Mod, there are three independent tasks including variables' data types, loops and programming commands from which a child can choose what to play first. In addition, another conditional task – if-then is implicit in each of the three independent tasks. Each task achieves one or more programming objective which is based on two programming books [41] and [42]. The workflow of the Creative Programmer Mod is presented in Figure 3.

All the tasks fulfill the logic and conditional (if-then) concepts. The latter is considered one of the essentials and the most used programming concept in children's games [39]. However, the understanding of these concepts is achieved when a child knows why things happen in a certain way, like if the child does action 'X', then he/she knows that outcome 'Y' will happen.

The *first task* was designed for all targeted ages, and it fulfills the variables concept, which stores a value assigned to it without any form of calculation. The variables in this task are represented by two different boxes, one of them is named" Display 'x' animal" which act as a string data type where 'x' is the animal's name. While the second box is named" Display 'x' animal 'n' times" which represent integer data type where 'n' is the number of 'x' animal. Integer and String data types were chosen because they are one of the basics data types [43] and can be clearly distinguished using animations. The player will be asked to destroy one of the boxes, and the output will be based on the box name, which reflects the instruction that will be performed.

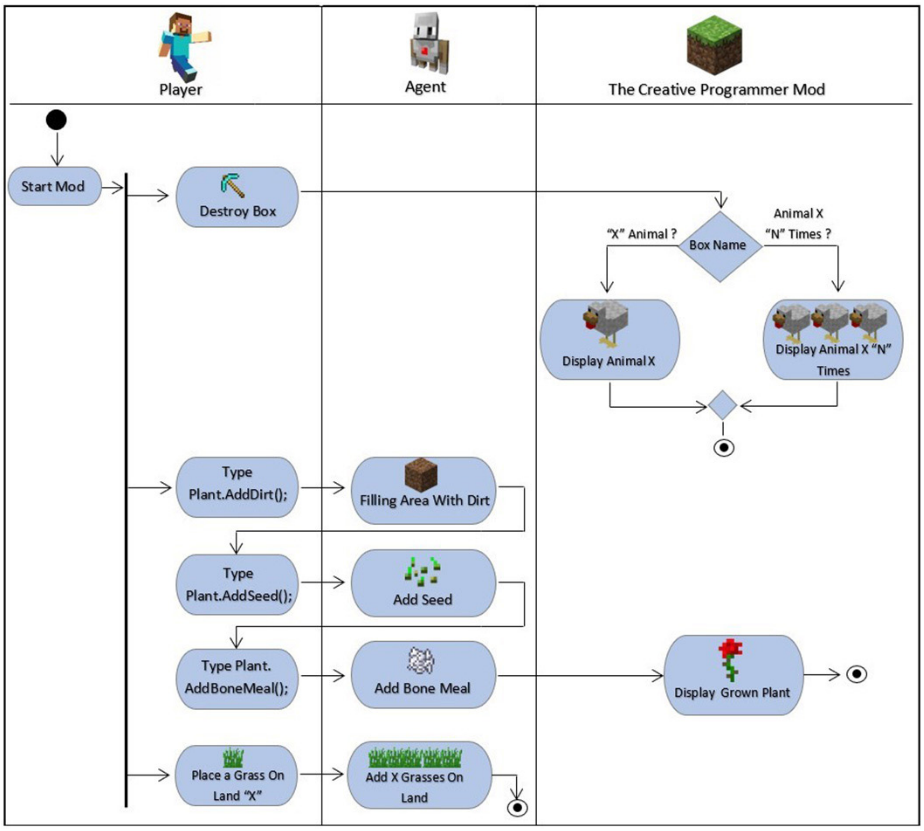


Fig. 3. Creative programmer workflow diagram

The *second task* targets only children who are seven years and older since they will be asked to write commands. It meets the algorithms concept which is a set of instructions that enable computers and humans to do a specific task, in addition to the programming concept in which the instructions are given in a written form. This task will be done with the help of Minecraft Agent [44] which is a *mob* (living mobile game entity) that will be programmed to execute given instructions. The term mob in Minecraft is used for moving objects such as players and animals. A set of commands for growing a plant will be written on a plate. The command will be in JavaScript programming language. In order to get a flower, the child should write these commands one by one via the chat window. Once the child enters a command for example" flower.AddSeed();", Minecraft Agent will move and plant some seeds in the soil.

The *third task* targets the selected age groups (5–15) and it meets the repetition concept which is represented by executing the same command repeatedly, this concept is very similar to the loop concept in programming. This task also uses Minecraft Agent to perform the instructions. In this task, the player will be asked to place a grass block in one of the three existing areas. Each area will have a specific number next to it, this number indicates the number of repetitions for this grass block. For example: if the child puts a grass block once in the land area that has number 'X', the instruction will be repeated 'X' times.

4.4 Programs and tools

This section presents the programs and tools used to develop the proposed Minecraft mod (see Figure 4). First, a Microsoft MakeCode platform [35] is used to program the mod, it provides both block-based and JavaScript editors. The developed mod by this platform can be played only in Minecraft Windows 10 [45] and Minecraft: Education Edition [36]. Second, Minecraft game for Windows 10 is used for running and playing the mod, since the education edition is available only for schools, museums, and libraries. Finally, a Code connection app [46] is used to connect MakeCode with Minecraft game.

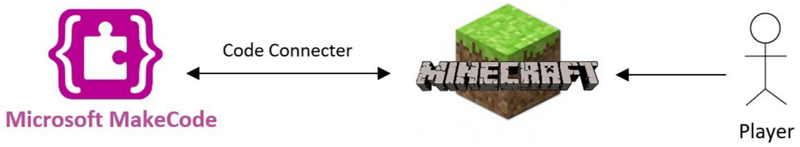
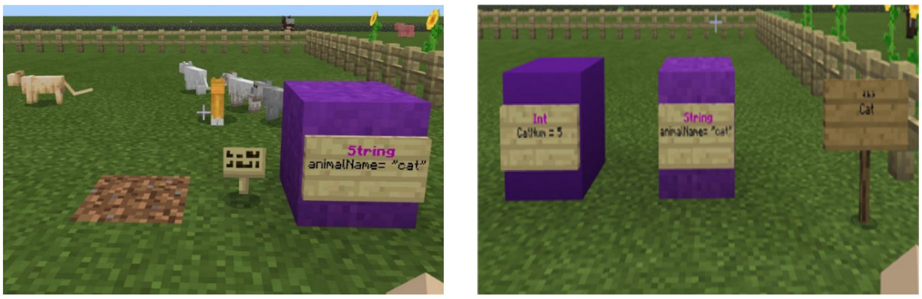


Fig. 4. Overview of the used programs in mod development

4.5 User interfaces

The Creative Programmer consists of three main tasks, each task is designed to deliver specific programming concepts which described previously in (Sect. 4.3). The tasks are independent from each other, which means, the player can choose any one of them to start playing.

The first task is a small fenced area that aims to teach data types and variable concepts (see Figure 5). This area contains three pairs of Minecraft blocks, each have a different color which represent an animal type. A board with the name of each animal type written in both Arabic and English, is placed next to each pair of blocks, to make it easier for the child to differentiate between them. However, each pair of blocks consist of two types of boxes, one represents string data type and the other, integer data type. The role of the player here is to destroy theses boxes to fill the area with animals. While doing that, player will notice the difference between data types based on the output from the destroyed blocks.

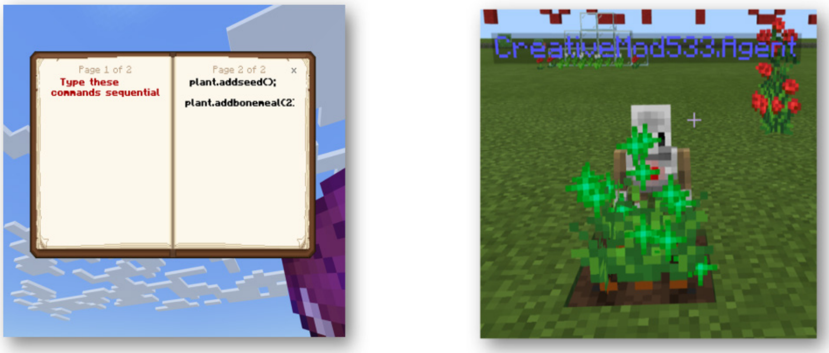


(a) The output of a destroyed integer block

(b) Integer and string blocks

Fig. 5. Task1. data types interfaces

In the second task, the child will learn how to program and write an algorithm to plant a carrot through Minecraft chat commands. The commands were written in a book element to make it easier for the child to see them and then write them in the chat (see Figure 6a). The two commands are: (1) *Plant.Addseeds()*. If this is written by the child, the Minecraft agent will till the soil and put the seeds in their allocated place. (2) *Plant. AddBoneMeal(2)*. The Minecraft agent here will add bone meal twice to speed up the process of carrot growth (see Figure 6b).



Page 1 of 2

Type these commands sequential

Page 2 of 2

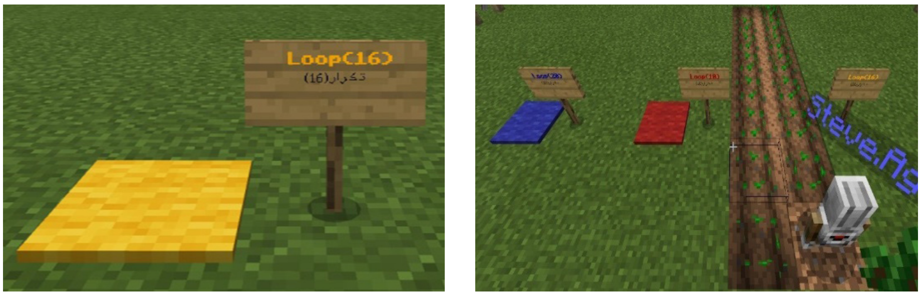
```
plant.addseed();

plant.addboneneal(2:
```

(a) Programming commands written in the book element

(b) After the Minecraft agent implemented the commands

Fig. 6. Task2. programming commands interfaces

In the third task the loop concept (see Figure 7a) is explained, it has three boards, each has a number for a loop written on it with different colors so the child can easily differentiate between them, the childs' role is to destroy the Square on the floor, when it is destroyed the agent in Minecraft will make loops depending on the number that is written on the board to plant the seeds (see Figure 7b).



(a) Loop concept

(b) The output of the loop

Fig. 7. Task3. loop concept

5 Testing and results

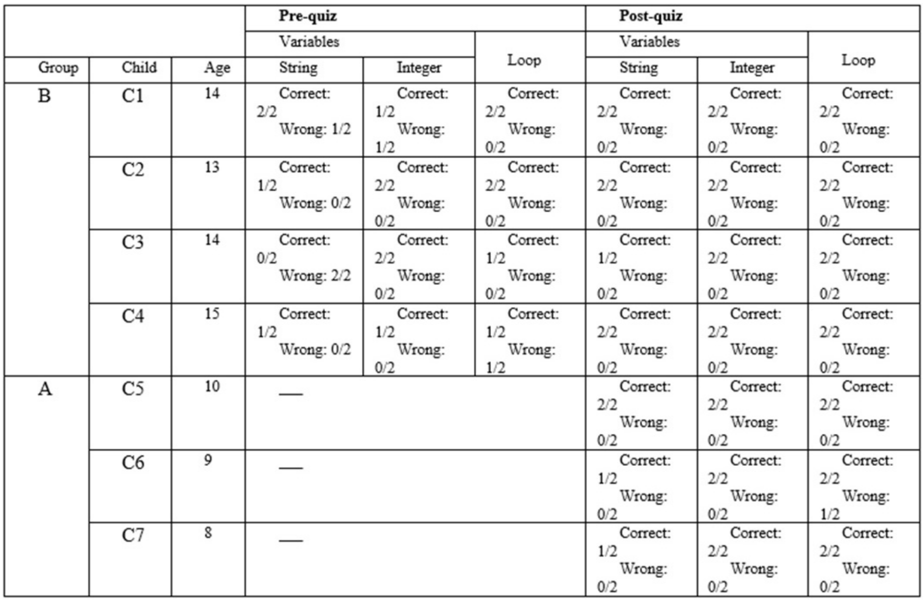
The Creative Programmer was tested by a preliminary study that includes a group of seven children aged between 9–15 years old who had previous knowledge of Minecraft. This test aimed to assess whether the children learn the programming concepts from the mod, or their knowledge of these concepts improved. Moreover, it aimed to observe how their familiarity with the learning environment is affected by their learning process. The testing method for learning outcomes is a quiz that consists of four questions related to the programming concepts designed in the game (see Sect. 4.3). The quiz was designed as the following, three multiple choice questions were used for variable and loop concepts, and one open-ended question was used for testing their ability to write programming commands (see Appendix B for quiz questions). The quizzes included variables, repetition, algorithm, programming, and logic concepts which are discussed in (Sect. 4.3). In the research testing, two approaches have been used (1) qualitative approach, which is used in the open-ended question and observing children's behavior while playing. (2) Quantitative approach, that had been used in the multiple-choice questions to measure the number of children who answered correctly.

Children were divided into two groups; group A and group B. Group A consisted of three children aged between 9–12 who had not learned programming concepts before. Group B consisted of four children between 13–15 who had pre-knowledge in programming concepts. For group B, they took the same quiz before (pre-quiz) and after playing the mod (post-quiz), to measure their improvement in performance. While group A took the quiz only after playing the mod since they did not have pre-knowledge in programming concepts. However, the open-ended question that related to programming commands, was not included in the pre-quiz, only in the post-quiz.

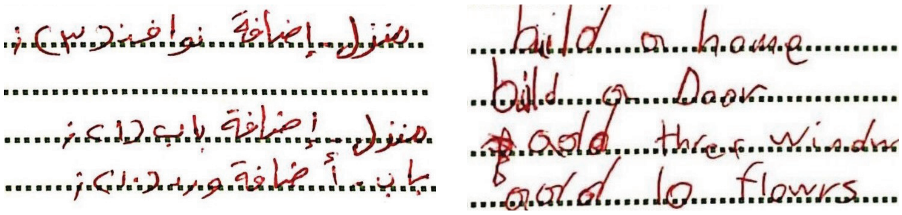
During the test, children played the Creative Programmer mod individually, using a laptop. There was no time constraint for each child, but no one took more than 30 minutes, due to the small number of tasks. The result of the multiple-choice questions is summarized in Table 2. For group A, after playing the mod, children recognized all integer concepts, by answering the question related to this concept correctly. But for the loop concept, only two children recognized all correct answers, and one child added a wrong answer. While for the string concept, only one child recognized all of them, however, two children identified only one correct answer. As for group B, the result of the pre-quiz shows that children of this group did not recognize all the correct answers for each concept. However, after they played the Creative Programmer and took the post-quiz, all children in this group answered all questions correctly. This indicates that all children noted an improvement in their performance in the test after playing the mod, and thus their level of knowledge in programming concepts also improved.

For the open-ended question, both groups were asked to write a few programming commands to build a house. As a result, all the children in both groups were able to write programming commands in clear steps but in their own way, according to their age and understanding. For example, some of them preferred to answer this question orally, due to writing difficulties. Moreover, few children only considered the programming rules while writing the commands, such as the semicolon at the end of the command (see Figure 8a). While the rest of the children wrote the commands in a narrative way (see Figure 8b).

Table 2. Results of pre- and post-quiz



			Pre-quiz			Post-quiz		
			Variables		Loop	Variables		Loop
Group	Child	Age	String	Integer		String	Integer	
B	C1	14	Correct: 2/2 Wrong: 1/2	Correct: 1/2 Wrong: 1/2	Correct: 2/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2
	C2	13	Correct: 1/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2
	C3	14	Correct: 0/2 Wrong: 2/2	Correct: 2/2 Wrong: 0/2	Correct: 1/2 Wrong: 0/2	Correct: 1/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2
	C4	15	Correct: 1/2 Wrong: 0/2	Correct: 1/2 Wrong: 0/2	Correct: 1/2 Wrong: 1/2	Correct: 2/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2
A	C5	10	—			Correct: 2/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2
	C6	9	—			Correct: 1/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2	Correct: 2/2 Wrong: 1/2
	C7	8	—			Correct: 1/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2	Correct: 2/2 Wrong: 0/2



منزل - إضافة نوافذ (٣) ؛

منزل - إضافة باب (١) ؛

باب - أضافة ورد (١٠) ؛

(a) Writing command task in Arabic

bild a home

bild a Door

add thre windw

add 10 flowrs

(b) Writing command task in English

Fig. 8. Results of writing command task

During the test, the authors observed the effect of children's familiarity with the learning environment, since all the children had played Minecraft before. The familiarity of the learning environment brought some benefits while learning, children knew the basic rules of Minecraft game including how to play it; testers did not need to teach them how to play; children took no time to explore and adapt to the learning environment. In other words, the learning process ran swiftly and smoothly. In addition, the children enjoyed themselves while learning, they did not stick to what they were asked to do all the time; they adapted to the game environment and started playing around. They did various activities once they started the game, for example, some of them modified the Creative Programmer appearance, where one added a gate to the farm fence, another built a hut, and one smashed the fence because he did not like the idea of isolating the animals from the surrounding environment. Others surprised us by swimming in the river, trying to ride the boat, killing cows and eating the meat! (see Figure 9). They were also surprised by the agents' character and would like to have one on their mod to execute their instructions.

Fig. 9. One player eating meat

Overall, the proposed Creative programmer mod, which was developed within the Minecraft game, is designed to achieve the goal of teaching the children basic programming concepts like variables' data types, loops, and programming commands in an entertaining learning environment. This developed mod proved its effectiveness; where the results of the testing indicated that children were able to acquire computational thinking skills by learning the programming concepts and most of them answered the quiz completely correct. Moreover, children showed a positive reaction towards the game and they enjoyed it, according to **[20]**, the fun part in game-based learning makes the learning process more effective.

6 Conclusion

In today's highly digital world children are using technology at a very young age, they use laptops and tablets for many daily activities such as watching YouTube, playing games and much more. Some video games get rising attention from children all around the world to the point it becomes a part of their real life, they spend many hours not only playing it, but they also talk about it and even impersonate its characters and buy their clothes. The engagement of children in different technology platforms such as games do not just give them a joyful time, they are actually socializing, acquiring intangible skills and learning new concepts. These benefits can be achieved only if the time spent on such platforms are utilized for doing well-guided activities. Consequently, developing an environment that includes interactive learning content within children's favorite games will result in a highly encouraging learning process. Programming is one of the main concepts that people became aware of the importance of learning it, and its impact on children's skills and abilities has become more evident. Children do not only learn how programs work or how to write a piece of code; they also acquire skills that benefit them in other aspects of their life, like problem-solving and computational thinking. Thus, children with such knowledge become more able to innovate and solve practical problems.

This paper intended to utilize the advanced level that children reached in using different technology' platforms especially games, the time they spent on it and their stimulus to discover new things to facilitate the learning process of new complex concepts. The results of this paper showed that children who were unfamiliar with programming concepts gained the knowledge successfully and were able to answer most of the post-quiz questions correctly after playing the proposed mod. Moreover, the performance of the children with prior knowledge of programming was improved, as they were able to answer all the questions correctly in the post-quiz. Overall, the game-based environment facilitated the learning process and made it more enjoyable for all children by attracting their attention and maintaining their focus throughout the game.

7 Limitations and future work

This study has a few limitations; one of them is the number of children tested. The Creative Programmer Mod was tested on only seven children; hence, a larger number of children may give more reliable results. However, running a test with a small number of participants allowed the researchers to pay full attention to each child and assess his/her learning progress during this experiment. Another limitation is that the tested children were aged between 8–15, and the Mod is designed for a wider range of ages (from 5 to 15), therefore, the test did not cover the full age range of the Mod. As a future work, researchers aim to develop a full programming course that starts from basic programming concepts and then is scaled up to complicated ones. The course content will be validated and adopted by the assistance of educational experts in teaching school-age children science and technology.

8 References

[1] H. K. Kabali et al. "Exposure and use of mobile media devices by young children", *Pediatrics,* vol. 136, no. 6, pp. 1044–1050, 2015. https://doi.org/10.1542/peds.2015-2151

[2] M. M. Crawford and G. Wright. *Delphi Method*. Wiley StatsRef: Statistics Reference Online, 2016.

[3] J. Evans. "Scrum revealed training book", *International Scrum Institution*.

[4] Pokemon. [Online]. Available: https://www.pokemon.com/us/. [Accessed: October 29th 2019].

[5] Minecraft. [Online]. Available: https://www.minecraft.net/en-us/. [Accessed: September 18th 2019].

[6] K. J. Hewett, B. C. Pletcher, and G. Zeng. "The 21st-century classroom gamer", *Games and Culture,* vol. 15, no. 2, pp. 198–223, 2020. https://doi.org/10.1177/1555412018762168

[7] R. Zviel-Girshin, A. Luria, and C. Shaham. "Robotics as a tool to enhance technological thinking in early childhood", *Journal of Science Education and Technology,* vol. 29, no. 2, pp. 294–302, 2020. https://doi.org/10.1007/s10956-020-09815-x

[8] A. S. Drigas and M. A. Pappas. "On line and other game-based learning for mathematics", *International Journal of Online and Biomedical Engineering (iJOE),* vol. 11, no. 4, 2015. https://doi.org/10.3991/ijoe.v11i4.4742

[9] D. E. Salonius-Pasternak and H. S. Gelfond. "The next level of research on electronic play: Potential benefits and contextual influences for children and adolescents", *Human Technology: An Interdisciplinary Journal on Humans in ICT Environments,* 2005. https://doi.org/10.17011/ht/urn.2005123

[10] Sabq, Survey on the use of Saudi children for smart devices and electronic games. [Online]. Available: https://cutt.us/bcVJZ/. [Accessed: September 9th 2019].

[11] R. Al-Washmi et al. "Design of a math learning game using a Minecraft mod", In European conference on games based learning, vol. 1, p. 10, 2014.

[12] J. D. Bayliss. "Teaching game AI through Minecraft mods", In 2012 IEEE International Games Innovation Conference, pp. 1–4, 2012. https://doi.org/10.1109/IGIC.2012.6329841

[13] C. Zorn, C. A. Wingrave, E. Charbonneau, and J. J. LaViola Jr. "Exploring Minecraft as a conduit for increasing interest in programming", In FDG, pp. 352–359, 2013.

[14] S. Deterding, D. Dixon, R. Khaled, and L. Nacke. "From game design elements to gamefulness: defining 'gamification'", In Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments, pp. 9–15, 2011. https://doi.org/10.1145/2181037.2181040

[15] J. Looyestyn, J. Kernot, K. Boshoff, J. Ryan, S. Edney, and C. Maher. "Does gamification increase engagement with online programs? A systematic review", *PloS One,* vol. 12, no. 3, p. e0173403, 2017. https://doi.org/10.1371/journal.pone.0173403

[16] M. Sailer and L. Homner, "The gamification of learning: A meta-analysis", ed: Springer, 2020.

[17] M.-C. Li and C.-C. Tsai. "Game-based learning in science education: A review of relevant research", *Journal of Science Education and Technology,* vol. 22, no. 6, pp. 877–898, 2013. https://doi.org/10.1007/s10956-013-9436-x

[18] T. Anastasiadis, G. Lampropoulos, and K. Siakas. "Digital game-based learning and serious games in education", *International Journal of Advances in Scientific Research and Engineering (ijasre),* vol. 4, no. 12, pp. 139–144, 2018. https://doi.org/10.31695/IJASRE.2018.33016

[19] S. Erhel and E. Jamet. "Digital game-based learning: Impact of instructions and feedback on motivation and learning effectiveness", *Computers & Education,* vol. 67, pp. 156–167, 2013. https://doi.org/10.1016/j.compedu.2013.02.019

[20] M. Prensky. "Digital game-based learning", *Computers in Entertainment (CIE),* vol. 1, no. 1, pp. 21–21, 2003. https://doi.org/10.1145/950566.950596

[21] J. Hamari, D. J. Shernoff, E. Rowe, B. Coller, J. Asbell-Clarke, and T. Edwards. "Challenging games help students learn: An empirical study on engagement, flow and immersion in game-based learning", *Computers in Human Behavior,* vol. 54, pp. 170–179, 2016. https://doi.org/10.1016/j.chb.2015.07.045

[22] K. Song and L. Burton. "Gamification in learning and education", Ed: Springer Nature, Cham, Switzerland, 2018.

[23] G. T. Jackson, L. K. Varner, C. Boonthum-Denecke, and D. S. McNamara. "The impact of individual differences on learning with an educational game and a traditional ITS", *International Journal of Learning Technology 25,* vol. 8, no. 4, pp. 315–336, 2013. https://doi.org/10.1504/IJLT.2013.059129

[24] M. U. Bers. "Coding as another language: A pedagogical approach for teaching computer science in early childhood", *Journal of Computers in Education,* vol. 6, no. 4, pp. 499–528, 2019. https://doi.org/10.1007/s40692-019-00147-3

[25] S. Papadakis, M. Kalogiannakis, and N. Zaranis. "Developing fundamental programming concepts and computational thinking with ScratchJr in preschool education: A case study", *International Journal of Mobile Learning and Organisation,* vol. 10, no. 3, pp. 187–202, 2016. https://doi.org/10.1504/IJMLO.2016.077867

[26] D. J. Portelance, A. L. Strawhacker, and M. U. Bers. "Constructing the ScratchJr programming language in the early childhood classroom", *International Journal of Technology and Design Education,* vol. 26, no. 4, pp. 489–504, 2016. https://doi.org/10.1007/s10798-015-9325-0

[27] Move The Turtle. [Online]. Available: https://www.movetheturtle.com. [Accessed: October 29th 2019].

[28] M. Meccawy. "Raising a programmer: Teaching Saudi children how to code", *International Journal of Educational Technology,* vol. 4, no. 2, pp. 56–65, 2017.

[29] I. Rogozhkina and A. Kushnirenko. "PiktoMir: Teaching programming concepts to preschoolers with a new tutorial environment", *Procedia-Social and Behavioral Sciences,* vol. 28, pp. 601–605, 2011. https://doi.org/10.1016/j.sbspro.2011.11.114

[30] K. Schwaber and J. Sutherland. *The Scrum Guide*. Scrum.org, 2017.

[31] N. C. Rashmi Popli. "Cost and effort estimation in agile software development", presented at the *International Conference on Reliability, Optimization and Information Technology*, Faridabad, India, 2014.

[32] L. M. Marek Majchrzak. "Factors influencing user story estimations: An industrial interview and a conceptual model", *Central and Eastern European Journal of Management and Economics,* vol. 4, pp. 261–280, 2016.

[33] R. J. Adam Trendowicz. *Software Project Effort Estimation*. Springer 2014.

[34] Scratchjr. [Online]. Available: https://www.scratchjr.org. [Accessed: October 29th 2019].

[35] R. P. N. C. H. Sharma. "Prioritising user stories in agile environment", 2014.

[36] Minecraft Education Edition. [Online]. Available: https://education.minecraft.net/. [Accessed: October 29th 2019].

[37] T. Karsenti and J. Bugmann. "The educational impacts of Minecraft on elementary school students", in *Research on e-Learning and ICT in Education*: Springer, 2018, pp. 197–212. https://doi.org/10.1007/978-3-319-95059-4_12

[38] K.-T. Foerster. "Teaching spatial geometry in a virtual world: Using minecraft in mathematics in grade 5/6", In 2017 IEEE Global Engineering Education Conference (EDUCON), pp. 1411–1418, 2017. https://doi.org/10.1109/EDUCON.2017.7943032

[39] A. Wilson, T. Hainey, and T. M. Connolly. "Using scratch with primary school children: An evaluation of games constructed to gauge understanding of programming concepts", *International Journal of Game-Based Learning (IJGBL),* vol. 3, no. 1, pp. 93–109, 2013. https://doi.org/10.4018/ijgbl.2013010107

[40] D. Saito and T. Yamaura. "A new approach to programming language education for beginners with top-down learning", In Proceedings of 2013 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE), pp. 752–755, 2013. https://doi.org/10.1109/TALE.2013.6654538

[41] Primo Toys. *Beginning Computer Programming for Kids Book*. 2017.

[42] Y. D. Liang. *Introduction To Java Programming, Comprehensive Version, 7/E*. Pearson Education India, 2009.

[43] S. Nakov and V. Kolev. *Fundamentals of Computer Programming with C#: The Bulgarian C# Book*. Faber Publishing, 2013.

[44] Minecraft Wiki. [Online]. Available: https://minecraft.gamepedia.com/Agent. [Accessed: October 29th 2019].

[45] Minecraft for Windows 10. [Online]. Available: https://minecraft.makecode.com/setup/minecraft-windows10. [Accessed: October 29th 2019].

[46] E. A. B. Coelho, Anirban. "Effort estimation in agile software development using story points", *International Journal of Applied Information Systems (IJAIS),* vol. 3, 2012. https://doi.org/10.5120/ijais12-450574

9 Appendix A

Questions	Answer Choices
What is the age of your child?	a) 5–7 b) 8–10 c) 11–15
What is your child's favorite way to interact with the games?	Direct interaction with visible objects Interact by typing in the command panel Both a and b
Is your child playing Minecraft?	a) Yes, A lot b) Sometimes c) Scarcely d) No, Never
Which Mod does your child prefer to play in?	a) Survival Mod b) Creativity Mod c) Both
What type of activities your child prefers to do in Minecraft?	a) Farm building and take care of animals and plants b) Constructing buildings and create innovative designs c) Both
Did your child learn any programming concepts through respective games?	a) Yes b) No
While the answer is (Yes) for the previous question, describe your child's experience with it. Did he\she enjoy it? How far did he\she benefit from it?	Open ended question
What do you think about introducing educational content such as the basic programming concepts in popular games that children spend long periods playing it? Such as Minecraft, Fortnite or others.	Open ended question

10 Appendix B

Programming Concept	Question Type	Question	Question Choices (if any)
Programming commands and Algorithms	Short answer	Help Minecraft friends and ***write programming commands*** that enables them to build a wooden house with three windows and one door, then decorate the door with beautiful flowers.	–
Variables	Multiple choice	If you want to build a house by using programming language, which of the following you should define it as a ***String***?	1- House type 2- Number of houses 3- Window colour 4- Number of doors
		If you want to build a house by using programming language, which of the following you should define it as an ***Integer***?	
Loop	Multiple choice	If you want to build a house by using programming language, in which situation you should use ***Loop concept***?	1- Plant one tree 2- Building a fence around the house 3- Plant 5 trees 4- Choose the colour of the fence

11 Authors

Bashaer Alsaadi received her master's degree of Computer Information Systems at King Abdulaziz University in Jeddah, Saudi Arabia. She is working as a research assistant at the Faculty of Computing and Information Technology, King Abdulaziz University. Previously, Bashaer was working as a software developer at King Abdulaziz University. She is interested in the field of software engineering and improving scrum team experience.

Bushra Alsaadi received her master's degree in computer information systems at King Abdelaziz University in Jeddah, Saudi Arabia. She worked as a lecturer at the College of Computer Science and Engineering, university of Jeddah. She also worked as a research assistant at King Abdulaziz University. Her research interest focuses on Health informatics, Artificial Intelligence, and Human-centered design.

Athir Saeed Alghamdi, a student doing her Master's in Information Systems at King Abdulaziz University in Jeddah, Saudi Arabia. She is interested in technology adoption in education, E-systems, and health informatics. This published work aims to help native Arabic children in acquiring introductory programming skills and computational thinking within an enjoyable environment. Athir is expected to submit her thesis about measuring the effectiveness of an Arabic gamified intervention in combating childhood obesity by the end of 2021. She like solving puzzles and reading in her free times.

Mashaal Abdulrahman Alfhaid is doing her Master's in Computer Information System at King AbdulAziz University, she graduated in 2018 with a bachelor's degree in Computer Information System. Mashaal undergraduate research focused on big data analytics and she also interested in field of software engineering and e-learning.

Nedaa Abdulelah Almuallim is doing her master's in information system at King Abdulaziz University, she expected to submit her thesis in 2021. She is interested in mobile Internet and mobile learning.

Dr. Maram Meccawy is currently working as an Associate Professor at the Information Systems department, Faculty of Computer Science and Information Technology at King Abdulaziz University in Jeddah, Saudi Arabia. Dr. Meccawy received her Ph.D in Computer Science from the University of Nottingham in the United Kingdom. Following her graduation, she worked in the industry in London and Jeddah before joining the university as an Assistant Professor in 2014. Her research interests include: Technology Enhanced Learning, Adaptive Hypermedia, User Modeling, MOOCS and Distributed Systems.

Article submitted 2021-08-23. Resubmitted 2022-09-12. Final acceptance 2022-09-12. Final version published as submitted by the authors.