IJCER International Journal of Contemporary Educational Research
Volume 11, Number 1, March 2024, Page 20-28 Article Type: Research Article

The Effect of Digital Game Design-Supported Coding Education on Gifted Students' Scratch Achievement and Self-Efficacy [1]

İbrahim Bozan[2] | | ibrahimbozan7@gmail.com
Ministry of Education, Antalya Science and Art Center, Antalya, Türkiye
Erdal Taşlıdere | | etaslidere@mehmetakif.edu.tr
Burdur Mehmet Akif Ersoy University, Faculty of Education, Department of Basic Education, Burdur, Türkiye

Abstract
Coding tools that use blocks to create programs are popular among kids and play a key role in learning how to code. The effectiveness of the coding courses that are available nowadays depends on how well the tools match the students' needs. The aim of this study is to reveal the impact of digital game design-supported coding education with Scratch on gifted students' Scratch academic achievement and self-efficacy. The research was conducted with a one-group pre-test and post-test experimental design. The sample of the study consists of 40 gifted 3rd grade students studying at a Science and Art Center in Türkiye. The Scratch achievement test and the Scratch self-efficacy scale were used as pre- and post-tests before and after the training. The data from the study were analyzed with a dependent group t-test. The post-test scores of the students obtained from the Scratch achievement and self-efficacy scale showed statistically significant increases compared to the pre-test scores. It was revealed that digital game design-supported education contributed positively to students' scratch achievement and self-efficacy in coding.

Keywords: Coding, Digital game design, Gifted students, Self-efficacy, Scratch

Citation
Bozan, İ. & Taşlıdere, E. (2024). The Effect of digital game design-supported coding education on gifted students' scratch achievement and self-efficacy. *International Journal of Contemporary Educational Research, 11*(1), 20-28. https://doi.org/10.52380/ijcer.2024.11.1.531

Received	08.08.2023
Accepted	29.01.2024
Publication	25.03.2024
Peer-Review	Double anonymized - Double Blind
Plagiarism Checks	Yes - iThenticate
Conflicts of Interest	The author(s) has no conflict of interest to declare.
Complaints	ijceroffice@gmail.com
Grant Support	The author(s) acknowledge that they received no external funding in support of this research.

[1] This study is produced from the first author's PhD dissertation supported by Mehmet Akif Ersoy University Scientific Research Projects Commission with the project number "0681-DR-20" under the supervision of the second author. The study is a developed version of the paper presented orally at the 13th International Congress on New Trends in Education

[2] Corresponding Author

Introduction

Since the beginning of the 21st century, all developed and developing countries in the world have been implementing many practices to improve their education systems in order to benefit more from technology (Sabah, 2020). One of these practices is related to the popularization of coding education. Although coding is not a new concept, it has started to find a place at the pre-school and primary school levels, which are considered the basic steps of education, very quickly in recent years compared to previous periods (Sayın & Seferoğlu, 2016). Although the words coding and programming differ in the curricula of countries in literature reviews, they can be used in the same sense and are defined as performing a specified operation using computer commands (Keçeci et al., 2016).

As technology becomes more integrated in schools, it has become important to be familiar with educational software developed by computer technologies (Beauchamp, 2004). As we are going through a period in which technology is combined with computers and computers are combined with software that is a necessity of the age, coding software directs people to computerized thinking (Balanskat & Engelhardt, 2015). Hence, learning to code has become a must-have skill for the 21st century. The applications and software needed for this have started to be provided, and coding education is carried out with various coding tools in different age ranges (Olsson & Granberg, 2022). Thus, coding education can be started even at an early age, such as preschool.

Although coding education can be started at a young age, some problems may arise during the education process. The reason for this is often said to be the complexity and difficulty of the programming process and education in young age groups (Kert & Uğraş, 2009). In the literature review, coding studies have emerged in the last few years. The literature mainly focuses on how coding education affects lesson motivation, the views on coding education, and the analysis of documents. For example, Gültepe (2018) interviewed eight teachers who participated in a coding project and found that coding helped children to create ideas and gain self-confidence. Sırakaya (2018) received opinions from 21 middle school students on coding education. As a result of the interviews, students found block-based coding tools fun and interesting. Özbey (2018), in his research on coding education for preschoolers, discovered that coding can enhance children’s cognitive abilities, boost their problem-solving skills, and help them apply it to other domains. These studies show that coding can be beneficial for children from an early age in many ways. For this reason, it becomes important for children to have high achievement and self-efficacy in the coding trainings they participate in.

Coding is a new topic of study, so it is misunderstood, incomplete, or unknown by parents, teachers, and students in the education field (Türker & Pala, 2018; Göncü, 2019). In the ongoing projects, there is a lack of equipment, knowledge, and educators due to this situation (Gültepe, 2018). While providing coding training, attention should be paid to whether the programs are suitable for the age group of children and the characteristics of the period in which they are (Resnick et al., 2009). This study used Scratch, a block-based coding program that is appropriate for elementary school students and has a simple interface for kids. One of the reasons for choosing Scratch is that it is understandable for students and can provide fun work opportunities (Ford, 2017). Hence, it is valuable to use programs like Scratch in coding education.

Scratch

Scratch is a program developed by researchers at the Massachusetts Institute of Technology Media Laboratory to teach coding (Balouktsis & Kekkeris, 2016). It is mostly designed for individuals between the ages of 8 and 16 and is now being actively used in approximately 160 countries. Users have the opportunity to share their projects on Scratch with other individuals. By coding with Scratch, children learn important strategies for problem solving, project design, and reasoning (https://Scratch.mit.edu/about). By sharing their Scratch projects interactively, students improve their problem-solving and creativity skills in an enjoyable way (Balouktsis & Kekkeris, 2016). The Scratch program, which is a block-based coding tool, has code blocks that can be moved with drag-and-drop logic, making coding easier for children. Besides being appropriate for elementary school students (Ford, 2017), the Scratch program can also be used by teachers and other adults without difficulty. Students can turn their work into a project and save it in a file so that they can work on it again later. The program can be used online on the internet as well as offline by installing it on computers. Each of the code blocks in the program means a command (Küçükkurt et al., 2016). By dragging and dropping the code blocks one after the other, the desired algorithm is created, and the designed project is made operational.

In order to develop coding skills in Türkiye, the Ministry of National Education shares educational content on http://scratch.eba.gov.tr and encourages the use of the Scratch program. In the literature (Brown et al., 2013), it is stated that lessons using coding and game design programs such as Scratch as a tool contribute to students' effective learning of coding. At this point, it may be effective to utilize game design in coding education with Scratch to improve students' coding skills (Shute et al., 2011). Therefore, coding education within the scope of this study was supported by digital game design.

Digital Game Design

Digital games are computer programs that became more popular with the rise of desktop computers in the 1990s and are now part of life (Oblinger, 2004). Digital games are fun, challenging, goal- and performance-oriented, competitive, and require skills such as strategy development and decision-making. Therefore, they are attractive to students (Kiili, 2005; Koster, 2005).

Games have the potential to be used in the educational process with all previously stated features (Prensky, 2001). Educators can use students' passion for games to teach programming. In this context, digital games can be considered teaching material (Gee, 2005). Kafai (2006) states that one of the aims of using digital games in education is to enable learners to create digital games in the learning-teaching process. Because students learn the content while designing, they build their own games to achieve educational goals, and learning takes place during the construction of these games (Kafai, 2006). Game design is a process where the learner is active and in charge of his or her own learning process, which engages learners' interest rather than a passive experience (Smeets, 2005). According to Resnick (2007), meaningful learning can only take place when there is full participation and full creation. For this reason, he proposed the process of creative game design. According to Brennan (2011), game design develops computational thinking skills such as mathematics, programming, and algorithmic thinking, as well as skills such as creativity and problem solving.

Educators emphasize that game design activities should be carried out in programming courses in order to adapt to the process, especially for beginners (Gee, 2005; Moreno, 2012; Rajaravivarma, 2005). In this way, it is aimed at increasing the motivation of beginners by designing games and fostering a positive attitude towards programming. With game design, coding lessons, which are generally seen as difficult and boring, can become more enjoyable and fun. By designing games with tools such as Scratch, students can improve their coding skills in a fun way by interacting with their friends (Resnick et al., 2009).

The necessity of actively using coding and game design tools such as Scratch in the education of gifted individuals has recently become more prominent (Kim et al., 2013). This is because gifted individuals' visual intelligence, motivation, and creativity should be supported along with their intellectual abilities (Callahan, 2000). In Türkiye, gifted students' education is provided in Science and Art Centers (SAC). Students who are identified as gifted attend SACs in the evening or at weekends to receive education in addition to their regular schools. Students receive at least four hours of education per week in these institutions. They have the opportunity to receive education in the fields of science, social sciences, mathematics, visual arts, music, and coding.

One of the fields that gifted students are most interested in is coding (Shin et al., 2013). Using the Scratch program and digital game design in coding education for these students can produce important results. In the literature, there is not enough research about coding education via Scratch and digital game design with gifted students. For this reason, it is important to reveal the effect of coding education via Scratch and digital game design on gifted students' coding achievement and coding self-efficacy beliefs.

Purpose of the Study

In this study, it was aimed at revealing the effect of digital game design-supported coding training with Scratch on the achievement and self-efficacy of gifted 3rd grade students. In line with this goal, students received coding and digital game design education using the Scratch program. They improved their coding skills by designing various games. Accordingly, the problems of the research are as follows:

1. What is the effect of digital game design-supported coding training on gifted students' Scratch achievements?
2. What is the effect of digital game design-supported coding training on gifted students' Scratch self-efficacy levels?

Methodology

The study was conducted in line with the one-group pre-test post-test design, which is a weak experimental design. In this design, the effect of the experimental procedure is tested with a single-group study. The measurements of the subjects regarding the dependent variable are obtained as a pre-test before the application and a post-test after the application using the same subjects and the same measurement tools (Büyüköztürk, 2016). The application was completed in a three-week period. The effects of the training on gifted students' Scratch success and self-efficacy were analyzed.

Population and Sample

The sample of the study consists of 40 gifted primary school students studying in the 3rd grade at one SAC in Antalya province in Türkiye. The participants were determined as a sample for convenience. There are five SACs in the same city, and hence the accessible population is 148 students studying at the same grade level in five SACs. The target population is all 3rd grade gifted learners in the country. The number of students in the sample is approximately 27% of the accessible population. When the gender frequencies of the students in the sample were analyzed, 15 (37.5%) students were female and 25 (62.5%) were male.

Data Collection Tools

Two data collection tools were used in the study. These are the Scratch Achievement Test and the Scratch Self-Efficacy Scale. Information about the data collection tools is given below.

Scratch Achievement Test

With the application in the research, it was aimed that the students would be able to code and design digital games with Scratch. Towards this goal, the Scratch Achievement Test (SAT) was used to determine to what extent the given training affected students' coding success. The SAT was developed by Büyükkarcı (2019) and consists of 20 multiple-choice items. The KR-20 reliability coefficient of the test was reported as .89 by the researcher. It was calculated as .71 in the post-test application of the current study. The SAT was used as a pre-test and post-test in the study.

Scratch Self-Efficacy Scale

The other scale used in the study was the Scratch Self-Efficacy Scale (SSES). Through this scale, the impact of the training on students' Scratch self-efficacy was determined. The SSES was taken from Büyükkarcı's (2019) research and has a five-point Likert structure. There are 12 items on the scale. The Cronbach Alpha reliability coefficient of the scale was reported as .95. It was found to be .89 according to the post-test application of the current study. The scale was used before and after the implementation, and data were collected.

Course Materials

A computer was provided for each student during the coding process. A smart board was used so that the students could see what the teacher explained through his or her computer. 14 pages of coding papers were prepared for students to use as a resource for coding through Scratch. The papers prepared by the researchers were given to the students before the Scratch training. The coding papers primarily included the Scratch interface. The features and usage of the stage, adding decor, adding puppets, costumes, and sound sections were explained with pictures. Then, the tasks of the block package in nine categories in the Scratch program were included. The block groups are movement, appearance, sound, events, control, perception, operators, variables, and my blocks, respectively. The purpose for which the code blocks under each block group will be used is expressed in pictures.

Training Process

Before starting the coding and digital game design training, the SAT and SSES were administered to the students as pre-tests. After the application of the tests, the training started. The training was completed in a total of 18 hours over three weeks, consisting of six hours per week. The researcher and an expert information technology teacher took part in the implementation. In the classroom where the training was held, computers used by the students were also available, along with equipment such as smart boards, desktop computers, printers, internet connections, etc. At the beginning of the training, 14-page coding papers were given to the students. It was ensured that the students could follow the training. The researcher was also present in all lessons and took the necessary precautions to prevent technical problems, student requests, etc.

During the training process, the interface of the Scratch program was explained, and students were provided with the menus and code blocks they would use while coding. The features and usage of the stage, adding decor, adding puppets, costumes, and sound sections were explained with pictures on the coding sheets. Movement blocks, view blocks, sound blocks, event blocks, control blocks, detection blocks, operator blocks, and variable blocks in the block menu were introduced to the students. Students were shown the variables and codes that they can use while designing games. Students made sample coding applications and game designs on Scratch based on what they learned during the education process. They were also able to follow the properties of all menus and code blocks shown by the teacher during the application process on the coding sheets. After the training, the SAT and SSES were administered to the students as post-tests, and the process was completed by collecting the data.

Data Analysis

Descriptive and inferential statistics were applied to test the problems of the research. Descriptively, the mean, standard deviation, skewness, and kurtosis values of the pre-test and post-test scores of the SAT and SSES applications were calculated. The values found for the pre-test and post-tests were compared. A dependent group *t*-test was used as inferential statistics to find out whether the difference between the pre-test and post-test scores of the students regarding SAT and SSES was significant or not.

Results

Descriptive Statistics Results

The descriptive analysis results of the SAT and SSES are shown in Table 1. According to Table 1, the mean pre-test score of the students on the SAT is 9.70, and the mean post-test score is 17.10. When the scores obtained from the SSES are analyzed, the mean pre-test score of the students is 27.95, and the mean post-test score is 52.50. It is seen that students' average means of the SAT and SSES scores increased by 7.4 and 24.55 points, respectively, from pre- to post-tests.

Table 1. Descriptive Statistics Results of the SAT and SSES

		N	Mean	Sd. Dev.	Skewness	Skewness Stand. Error	Kurtosis	Kurtosis Stand. Error
SAT	Pre-test	40	9,70	3,86	-,24	,37	-1,11	,73
	Post-test	40	17,10	2,60	-1,43	,37	1,48	,73
SSES	Pre-test	40	27,95	9,26	,37	,37	-,10	,73
	Post-test	40	52,50	6,72	-,85	,37	-,05	,73

According to the post-test data obtained from the Scratch achievement and self-efficacy tests, the skewness and kurtosis values are between -1.5 and +1.5. According to these values, the data are almost normally distributed (Tabachnick & Fidell, 2013).

Inferential Statistics Results

A dependent group *t*-test was used to reveal the effect of digital game design-supported Scratch education on students' code success. The data were analyzed to check the statistical significance of the difference between the pre-test and post-test scores of the students regarding the measurement tools. Assumptions regarding the analysis were tested, and no problems were encountered. The values for the dependent group *t*-test are shown in Table 2.

Table 2. Dependent Groups t-test Results for the SAT

	Mean	S. Dev.	t	sd	p	d
Pre-test Post-test	-7.40	3.48	-13.43	39	.000	2.12

According to Table 2, a statistically significant difference was found in favor of the post-test scores as a result of the dependent group *t*-test conducted on the pre-test and post-test data of the SAT (*p*=.000). The post-test scores obtained by the SAT were significantly higher than the pre-test scores. Considering the calculated effect size (*d*=2,12), the value is at a high level (Green & Salkind, 2005). These results reveal that the training had a significant effect on students' coding success.

Likewise, a dependent group *t*-test was applied to reveal the effect of digital game design-supported coding education on students' self-efficacy. Assumptions regarding the analysis were checked, and no problems were encountered. The values for the dependent group *t*-test are given in Table 3.

Table 3. Dependent Groups t-test Results for the SSES

	Mean	S. Dev.	t	sd	d	d
Pre-test Post-test	-24.55	8.71	-17.81	39	.000	2.81

According to Table 3, a statistically significant difference was found in favor of the post-test scores as a result of the dependent group *t*-test conducted on the pre-test and post-test data of the students' SSES (*p*=.000). The post-test scores obtained by the students from the SSES are significantly higher than the pre-test scores. The calculated effect size (*d*=2.81) shows that this difference is at a high level (Green & Salkind, 2005). These results reveal that the training had a significant effect on students' self-efficacy levels.

The results obtained by the students from the SAT and the SSES reveal that the coding training was successful and that the training was completed as targeted. Thus, it can be said that the students received an effective education in terms of coding and digital game design with the Scratch program.

Conclusion and Discussion

The results of the study revealed that the coding achievement of gifted 3rd grade students increased descriptively from pre- to post-tests. It was also supported that the increase in the average means was statistically significant based on the result of the dependent group *t*-test. Accordingly, it can be stated that digital game design-supported coding training was effective in increasing the scratch achievement of the participants. This result overlaps with those of the previous research (Büyükkarcı & Taşlıdere, 2021; Çağıltay, 2007; Rizvi et al., 2011; Westcott, 2008). In their study, Büyükkarcı and Taşlıdere (2021) report that scratch education increased 4th grade primary school students' coding achievement. Likewise, Rizvi et al. (2011) announced that game design activities with Scratch in the programming course were effective on the coding achievement of first-year university students. In addition, Westcott (2008) stated that game design training with Scratch had an effect on programming achievement. The research conducted by Çağıltay (2007) reports that students who played games with Scratch had higher coding achievement than the students who did not. In addition, the results of the current study are also similar to the previous ones (Bishorp-Clark et al., 2007; Cooper et al., 2003; Howland & Good, 2015), who report that digital game design trainings using visualization tools such as Alice and Flip increase coding achievement.

Another result obtained in the current study is that students' Scratch self-efficacy scores increased significantly. This finding supports various research results in the literature (Armoni et al., 2015; Nikou & Economides, 2014; Rizvi et al., 2011; Serim, 2019). In the study of Armoni et al. (2015), the use of Scratch in coding increased students' self-efficacy levels. Nikou & Economides (2014) reported that Scratch was effective in learning programming in K–12 educational environments and that students' self-efficacy increased. Rizvi et al. (2011) found that students who used Scratch in programming courses had high self-efficacy. In Serim's (2019) study, coding education with a gamification approach positively affected students' self-efficacy perceptions towards coding. These studies show that the use of Scratch contributes to students' self-efficacy.

Scratch is a block-based coding tool with colorful content that students can easily use (Sırakaya, 2018). In the implementation process, the realization of digital game design-supported education in a fun way increased students' motivation. Students' high motivation may have positively affected their self-efficacy (Bandura, 1977). According to the study conducted by Saez-Lopez et al. (2016), the fact that 5th grade students had high motivation ensured their desire and commitment to the lesson. However, according to the results obtained in Korkmaz's (2016) research, the fact that students have positive results about Scratch is due to the features of Scratch. Korkmaz (2016) conducted his study with university students, in which he stated that the negativities experienced in coding education are a decrease in motivation, attention, and perception. The current study was conducted with gifted students in the 3rd grade of primary school. The difference between these results may be due to the fact that the students in the studies were in different educational and age groups. In addition, the gifted students in the current study participated in coding and digital game design training for the first time without any previous experience in coding, which may have led to positive results.

Recommendations

In this study, coding and digital game design training with Scratch were given to gifted 3rd grade students in primary school. As a result of this training, it was determined that students' academic achievement and self-efficacy in coding increased. Therefore, it may be recommended to conduct scratch-based training with student groups at different levels. Since coding and digital game design activities with Scratch can attract students' interest, they can be integrated into teaching models and used in teaching content in different fields. Qualitative research can be conducted on the reasons for the increase in students' success in coding education. In future studies, students' views on the use of digital game design in coding education can be examined. In addition, students' perspectives on the effectiveness of the Scratch program can be examined. It was concluded that the method used in the study improved students' self-efficacy. The effect of this method on different variables such as attitude, motivation, and creativity can be investigated. In order to be able to use coding education supported by digital game design at all times, classes supported by information technologies can be created in SACs and all kinds of state-owned schools. Similar future studies would be conducted with gifted or non-gifted students, and their results should be compared with those of the current research.

Limitations

This research was conducted in a science and art center in the province of Türkiye. The study was conducted with 40 gifted students, and two measurement tools, the SAT and SSES, were administered. Therefore, all the results in the study are based on the data collected from these students and the tests.

Author (s) Contribution Rate

Both researchers contributed at every stage of the research.

Ethical Approval

Ethical Approval (15/10/2020-79673485-302.08.01-E.45707) was obtained from Burdur Mehmet Akif Ersoy University for this research.

References

Armoni, M., Meerbaum-Salant, O., & Ben-Ari, M. (2015). From scratch to "real" programming. *ACM Transactions on Computing Education (TOCE), 14*(4), 1-15. https://doi.org/10.1145/2677087

Balanskat, A., & Engelhardt, K. (2015). *Computing our future, computer programming and coding - Priorities, school curricula and initiatives across Europe.* Retrieved from http://fcl.eun.org/documents/10180/14689/Computing+our+future_final.pdf/746e36b1-e1a6-4bf1-8105-ea27c0d2bbe0.

Balouktsis, I., & Kekkeris, G. (2016). Learning renewable energy by Scratch programming. *Journal of Research in Education and Training*, *9*, 129-141. https://doi.org/10.12 681/jret.8916

Bandura, A. (1977). Self-efficacy: toward a unifying theory of behavioral change. *Psychological Review*, 84(2), 191-215. https://psycnet.apa.org/doi/10.1037/0033-295X.84.2.191

Beauchamp, G. (2004). Teacher use of the interactive whiteboard in primary schools: Towards an effective transition framework. *Technology, Pedagogy and Education*, *13*(3), 327-348. https://doi.org/10.1080/14759390400200186

Bishop-Clark, C., Courte, J., & Howard, E. V. (2007). A quantitative and qualitative investigation of using Alice programming to improve confidence, enjoyment and achievement among non-majors. *Journal of Educational Computing Research*, *37*(2) 193-207. https://doi.org/10.2190/J8W3-74U6-Q064-12J5

Brennan, K. (2011). *Creative Computing: A design-based introduction to computational thinking*. Scratch Curriculum Guide, Harvard Graduate School of Education. http://scratched.media.mit.edu/sites/default/files/CurriculumGuide-v20110923.pdf.

Brown, Q., Mongan, W., Kusic, D., Garbarine, E., Fromm, E., & Fontecchio, A. (2013). *Computer aided instruction as a vehicle for problem solving: Scratch programming environment in the middle years classroom*. Proceedings of the ASEE Annual Conference and Exposition. http://www.pages.drexel.edu/~dmk25/ASEE_08.pdf

Büyükkarcı, A. (2019). *The effect of 5E model enriched with coding on 4th grade mathematics achievement, permanence and attitude* (Unpublished doctoral dissertation). Burdur Mehmet Akif Ersoy University.

Büyükkarcı, A., & Taşlıdere, E. (2021). The effect of coding education on students' efficiency and scratch achievements. *Journal of Educational Technology*, *18*(2), 43-54.

Büyüköztürk, Ş. (2016). *Deneysel desenler: öntest-sontest kontrol grubu, desen ve veri analizi* [Experimental designs: Pretest-posttest control group, design and data analysis]. Ankara, Türkiye: Pegem Akademi Yayıncılık.

Cooper, S., Dann, W., & Pausch, R. (2003). Using animated 3D graphics to prepare novicesfor CSI. *Computer Science Education*, *13*, 3-30. https://doi.org/10.1076/csed.13.1.3.13540

Çağıltay, N. E. (2007). Teaching software engineering by means of computer-game development: Challenges and opportunities. *British Journal of Educational Technology*, *38*(3), 405-415. https://doi.org/10.1111/j.1467-8535.2007.00705.x

Ford, M. (2017). *Coding across the curriculum*. Retrieved from https://www.edutopia.org/article/codingacross-curriculum.

Gee, J. P. (2005). Learning by design: Good video games as learning machines. *E-Learning and Digital Media, 2*(1), 5-16. https://doi.org/10.2304/elea.2005.2.1.5

Göncü, A. (2019). *Perceptions of information technologies and software course teachers toward coding education* (Unpublished master's thesis). Abant İzzet Baysal University.

Green, S. B., & Salkind, N.J. (2005). *Using SPSS for windows and macintosh: Analyzing and undestanding data.* New Jersey: Pearson.

Gültepe, A. (2018). Make coding teaching by Ict teachers eye "students are encoding". *International Journal of Leadership Training*, *2*(2), 50-60.

Howland, K., & Good, J. (2015). Learning to communicate computationally with Flip: A bi-modal programming language for game creation. *Computers & Education*, *80*, 224-240. https://doi.org/10.1016/j.compedu.2014.08.014

Kafai, Y. B. (2006). Playing and making games for learning instructionist and constructionist perspectives for game studies. *Games and Culture*, *1*(1), 36–40. https://doi.org/10.1177/1555412005281767

Keçeci, G., Alan, B., & Kırbağ Zengin, F. (2016). Educational computer games assisted learning coding attitude scale: validity and reliability study. *Education Sciences*, *11*(3), 184-194.

Kert, S. B., & Uğraş, T. (2009, May). *Simplicity and fun in programming education: Scratch example*. Paper presented at the First International Congress of Educational Research, Çanakkale, Türkiye.

Kiili, K. (2005). Digital game-based learning: Towards an experiential gaming model. *The Internet and Higher Education*, *8*(1), 13-24. https://doi.org/10.1016/j.iheduc.2004.12.001

Kim, S., Chung, K., & Yu, H. (2013). Enhancing digital fluency through a training program for creative problem solving using computer programming. *The Journal of Creative Behavior*, *47*(3), 171-199. https://doi.org/10.1002/jocb.30

Korkmaz, Ö. (2016). *The* effect of scratch-based game activities on students' attitudes, self-efficacy and academic achievement. *International Journal of Modern Education and Computer Science, 8*(1), 16-23. https://doi.org/10.5815/ijmecs.2016.01.03

Koster, R. (2005). *A theory of fun for game design*. Scottsdale: Paraglyph Press.

Küçükkurt, M., Akça, M., & Turan, İ (2016). *Temel kodlama eğitimi* [Basic coding training]. Ankara, Türkiye: Pusula Kitapçılık.

Moreno, J. (2012). Digital competition game to improve programming skills. *Educational Technology & Society*, *15*(3), 288–297.

Nikou, S. A., & Economides, A. A. (2014). Transition in student motivation during a scratch and an app inventor course. *Proceedings of the IEEE Global Engineering Education Conference (EDUCON)* (pp. 1042-1045). IEEE. https://doi.org/10.1109/EDUCON.2014.6826234

Oblinger, D. (2004). The next generation of educational engagement. *Journal of Interactive Media in Education*, *8*, 1–18.

Olsson, J., & Granberg, C. (2022). Teacher-student interaction supporting students' creative mathematical reasoning during problem solving using Scratch. *Mathematical Thinking and Learning*, 1-28. https://doi.org/10.1080/10986065.2022.2105567

Özbey, T. (2018). *Pre-school coding education and coding tools.* İstanbul Ticaret University Graduate School of Foreign Trade, İstanbul.

Prensky, M. (2001). *Fun, play and games: What makes games engaging?*. New York, USA: McGraw-Hill.

Rajaravivarma, R. (2005). A games-based approach for teaching the introductory programming course. *ACM Inroads - The SIGSCE Bulletin*, *37*, 98-102. https://doi.org/10.1145/1113847.1113886

Resnick, M. (2007). Sowing the seeds for a more creative society. *Learning and Leading with Technology*, *35*(4), 18-22.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., . . . Kafai, Y. (2009). Scratch: Programming for all. *Communications of the ACM, 52*(11), 60-67. https://doi.org/10.1145/1592761.1592779

Rizvi, M., Humphries, T., Major, D., Jones, M., & Lauzun, H. (2011). A CS0 course using scratch. *Journal of Computing Sciences in Colleges*, *26*(3), 19-27.

Sabah , N. M. (2020). Motivation factors and barriers to the continuous use of blended learning approach using Moodle: students' perceptions and individual differences. *Behaviour & Information Technology*, *39*(8), 875-898, https://doi.org/ 10.1080/0144929X.2019.1623323

Saez-Lopez, J., Roman-Gonzalez, M., & Vazquez-Cano, E. (2016). Visual programming languages integrated across the curriculum in elementary school: A two year case study using "Scratch" in five schools. *Computers & Education*, *97,* 129-141. https://doi.org/10.1016/j.compedu.2016.03.003

Sayın, Z., & Seferoğlu, S. S. (2016, February). *Coding education as a new 21st century skill and effect of coding on educational policies.* Paper presented at the Academic Informatics Conference, Aydın, Türkiye.

Serim, E. Ü. (2019). *Examination of students' computational thinking skills and self-efficacy perceptions about coding with coding education designed with gamification method* (Unpublished master's thesis). Balıkesir University.

Shin, S., Park, P., & Bae, Y. (2013). The effects of an information-technology gifted program on friendship using scratch programming language and clutter. *International Journal of Computer and Communication Engineering*, *2*(3), 246-249. https://doi.org/10.7763/IJCCE.2013.V2.181

Sırakaya, M. (2018). Student views on coding education. *Ondokuz Mayis University Journal of Education, 37*(2), 79-90.

Smeets, E. (2005). Does ICT contribute to powerful learning environments in primary education. *Computers and Education*, *44*(3), 343–355. https://doi.org/10.1016/j.compedu.2004.04.003

Shute, V. J., Rieber, L. & Van Eck, R. (2011). Games... and... Learning. *R. Reiser & J. Dempsey (Eds.), Proceedings of the Trends and Issues in Instructional Design and Technology.* (pp. 321–332), Boston, USA: Pearson.

Tabachnick, B. G., & Fidell, L. S. (2013). *Using multivariate statistics (6th edition)*. United States: Pearson Education.

Türker, P. M., & Pala, F. K. (2018). Opinions of secondary school students, teachers and parents about coding. *Elementary Education Online*, *17*(4), 2013-2029. https://doi.org/10.17051/ilkonline.2019.506939

Westcott, S. (2008). *Effectiveness of using digital game playing in a first-level programming course* (Unpublished doctoral dissertation). Pace University.