

Bilgisayar ve Öğretim Teknolojileri Eğitimi

Alanında Uluslararası
Araştırmalar - I

International Research in the
Field of Computer Education and
Instructional Technology

Editör: Doç. Dr. Meltem Kurtoğlu Erden

Editör:

Doç. Dr. Meltem Kurtoğlu Erden

Bilgisayar ve Öğretim Teknolojileri Eğitimi

International Research in the Field of
Computer Education and Instructional
Technology

Alanında Uluslararası
Araştırmalar I

EĞİTİM

yayınevi

**BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ ALANINDA ULUSLARARASI
ARAŞTIRMALAR-I**

(International Research in the Field of Computer Education and Instructional Technology)

Editör: Doç. Dr. Meltem Kurtoglu Erden

Genel Yayın Yönetmeni: Yusuf Ziya Aydoğan (yza@egitimyayinevi.com)

Genel Yayın Koordinatörü: Yusuf Yavuz (yusufyavuz@egitimyayinevi.com)

Sayfa Tasarımı: Kübra Konca Nam

Kapak Tasarımı: Eğitim Yayınevi Grafik Birimi

T.C. Kültür ve Turizm Bakanlığı

Yayıncı Sertifika No: 76780

ISBN: 978-625-5997-22-7

1. Baskı, Ekim 2024

Baskı Cilt: Repro Bir Matbaacılık

İvedik OSB Matbaacılar Sitesi 1514 Cad. No: 23-25 Yenimahalle/Ankara

Matbaa Sertifika No: 47381

Kütüphane Kimlik Kartı

**BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ ALANINDA ULUSLARARASI
ARAŞTIRMALAR-II**

(International Research in the Field of Computer Education and Instructional Technology)

Editör: Doç. Dr. Meltem Kurtoglu Erden

VI+130 s., 135x215 mm

Kaynakça var, dizin yok.

ISBN: 978-625-5997-22-7

Copyright © Bu kitabın Türkiye'deki her türlü yayın hakkı Eğitim Yayınevi'ne aittir. Bütün hakları saklıdır. Kitabın tamamı veya bir kısmı 5846 sayılı yasanın hükümlerine göre kitabı yayımlayan firmanın ve yazarlarının önceden izni olmadan elektronik/mekanik yolla, fotokopi yoluyla ya da herhangi bir kayıt sistemi ile çoğaltılamaz, yayımlanamaz.

EĞİTİM
yayınevi

Yayınevi Türkiye Ofis: İstanbul: Eğitim Yayınevi Tic. Ltd. Şti., Atakent mah.
Yasemen sok. No: 4/B, Ümraniye, İstanbul, Türkiye

Konya: Eğitim Yayınevi Tic. Ltd. Şti., Fevzi Çakmak Mah. 10721 Sok. B Blok,
No: 16/B, Safakent, Karatay, Konya, Türkiye
+90 332 351 92 85, +90 533 151 50 42, 0 332 502 50 42
bilgi@egitimyayinevi.com

Yayınevi Amerika Ofis: New York: Egitim Publishing Group, Inc.
P.O. Box 768/Armonk, New York, 10504-0768, United States of America
americaoffice@egitimyayinevi.com

Lojistik ve Sevkiyat Merkezi: Kitapmatik Lojistik ve Sevkiyat Merkezi, Fevzi Çakmak Mah.
10721 Sok. B Blok, No: 16/B, Safakent, Karatay, Konya, Türkiye
sevkiyat@egitimyayinevi.com

Kitabevi Şubesi: Eğitim Kitabevi, Şükran mah. Rampalı 121, Meram, Konya, Türkiye
+90 332 499 90 00
bilgi@egitimkitabevi.com

İnternet Satış: www.kitapmatik.com.tr
+90 537 512 43 00
bilgi@kitapmatik.com.tr

İÇİNDEKİLER

SUNUŞ..... V

EĞİTİMDE ÇEVİRİMİÇİ TOPLANTILAR: FIRSATLAR,
ZORLUKLAR VE ETKİLERİ..... 1

Cemal Tatlı

EĞİTİMDE YAPAY ZEKÂ DESTEKLİ OYUNLAŞTIRMA:
ÖĞRENCİ MOTİVASYONU, PERFORMANSI VE
KİŞİSELLEŞTİRİLMİŞ ÖĞRENME DENEYİMLERİ 29

Abdullah Alagöz

BÖTE MEZUNU ÖĞRETMENLERDEN BEKLENTİLER
ÜZERİNE BİR DURUM ÇALIŞMASI 45

Meltem Kurtoğlu Erden, Ural İbiş

THE VIEWS OF INFORMATION TECHNOLOGY TEACHERS
ON BLOCK-BASED PROGRAMMING EDUCATION..... 67

Yunus Emre Özenoğlu, İsmail Kara, Okan Alkan

UZAKTAN EĞİTİMDE ÖLÇME VE DEĞERLENDİRME
ARAÇLARI 89

Cemal Tatlı

YAPAY ZEKÂ ARAŞTIRMA TRENDLERİ EĞİTİM BİLİMLERİ
ALANINDA TEZLERİN İÇERİK ANALİZİ.....111

Cansu Şahin Kölemen

SUNUŞ

Teknoloji insanlığın var olduđu ilk andan itibaren gelişmeye devam ederek ve her çağda, o dönemdeki gelişmeler ışığında değışerek bugüne kadar gelmiştir. İçinde bulunduğumuz çağda toplumların gelişmişlik düzeyleri bilim ve teknolojiadaki ilerleme düzeylerine bağılı olarak belirlenmektedir. Toplumların gelişmişliklerini artırmaları da ancak eğitim yoluyla sağlanır. Eğitimin en temel amaçlarından biri de çağa ayak uydurabilen, yeniliklere uyum sağlayabilen, sadece tüketen değil aynı zamanda da üreten bireyler yetiştirebilmektir. Eğitime bilimsel ve teknolojik bir nitelik kazandırmanın zorunlu olduđu bir çağda eğitim ve teknolojinin birbirlerini ne yönde etkilediğini, aralarında ne gibi ilişkilerin bulunduğunu inceleyen bir bölüm olan Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü, kurulduđu dönemden günümüze kadar gelen süreçte, teknolojiyi ve teknolojiyle birlikte hayatımıza giren yeniliklerle birlikte bu yeniliklerin öğrenme-öğretme süreçlerinde kullanılmasına ilişkin birçok araştırmanın yanısıra, dijital araçları kullanmasını bilen, teknolojik gelişmelerden faydalanırken etik kurallara ve kişi haklarına saygılı, çevrimiçi teknolojileri ve dijital platformları bilinçli ve doğru kullanmasını bilen, bilişim etiğinden haberdar, teknoloji ve bilgi okuryazarı bireylerin yetişmesine de olanak sağlamıştır. Bu çalışma kapsamında bizler de yolu BÖTE bölümlerinde çalışmış araştırmacılar olarak, çağın getirdiği yenilikler ışığında Bilgisayar ve Öğretim Teknolojileri Eğitimi Araştırmaları-1 başlığı altında güncel bilimsel çalışmalar sunmayı amaçladık. Öncelikle Eğitim Yayınevine bizlere bu imkanı sunduđu için, sonra da yazarların her birine bu esere katkı sağladıkları için teşekkür ederim.

Meltem KURTOĞLU ERDEN

THE VIEWS OF INFORMATION TECHNOLOGY TEACHERS ON BLOCK-BASED PROGRAMMING EDUCATION

Yunus Emre Özenoğlu¹, İsmail Kara², Okan Alkan³

INTRODUCTION

The living conditions of today's information society have created differences in the competencies individuals need to possess (Aslan, 2019). There is a need for individuals who can access information quickly and easily, keep pace with technological developments, think analytically and produce logical solutions (Duran et al., 2015). In other words, it is expected that individuals of the 21st century not only use information but also produce it (Kalelioğlu, 2015). In this context, the competencies that students today need to have include critical thinking, problem-solving, creative thinking, collaboration, the ability to analyze and synthesize, reasoning skills (Güksoy, 2018), systematic thinking ability (Alkan, 2019), entrepreneurship and effective use of information and communication technologies (Sırakaya, 2018; Yıldızlar, 2013). It is thought that these skills can be developed through programming education using various methods (Saygıner & Tüzün, 2017). Programming education, which has been a topic of debate for a long time, has recently become widespread under the name of "coding education." Coding education is

1 Öğr. Gör., Erzincan Binali Yıldırım Üniversitesi, emre.ozenoglu@gmail.com, orcid.org/0000-0001-5051-4622

2 Öğr. Gör., Atatürk Üniversitesi, ismailkaratr@gmail.com, orcid.org/0000-0002-8097-9006

3 Öğr. Gör., Erzincan Binali Yıldırım Üniversitesi, okanalkan011@gmail.com, https://orcid.org/0000-0002-6526-1645

seen as an important tool by educational institutions in raising individuals who possess social skills. Both in our country and around the world, the importance of coding education has been recognized and incorporated into curricula (Akpınar & Altun, 2014). In Europe and America, programming education, starting from kindergarten, aims to develop students' problem-solving abilities and to train individuals who can solve everyday problems (Kukul & Gökçearslan, 2014). Furthermore, in countries like the UK and South Korea, initiatives have been taken to start coding education at a young age (Demirer & Sak, 2016). In our country, one of the aims of the "Information Technologies and Software Course Curriculum (1st, 2nd, 3rd and 4th Grades)" prepared by the Ministry of National Education is to enable students to use different thinking structures for problem-solving and to be able to design their games and program them (TTKB, 2018). Similarly, the curriculum prepared for 5th and 6th-grade students aims to develop students' problem-solving and cognitive thinking skills and to enhance their collaboration competencies (TTKB, 2017). Information Technology and Software teachers have significant responsibilities in equipping students with 21st-century skills (Pala, 2019).

Nowadays, it is observed that many block-based programming languages have emerged, allowing anyone, especially children, to learn coding (Çatlak et al., 2015). Block-based programming languages rely on combining existing code blocks on the screen in a drag-and-drop manner, like a puzzle, rather than writing long lines of code (Akbay, 2019; Mohamad et al., 2011; Weintrop & Wilensky, 2017). Besides these features, they are also used to build the necessary foundation for learning advanced programming languages (Papert, 1993; Weintrop & Wilensky, 2017). Kim, Choi and So (2012) stated that learners focused more on block-based coding since there were no errors related to the syntax and punctuation found in traditional languages. In addition to the positive features of block-based coding tools, they also

have some negative aspects. The reluctance to transition to text-based programming due to familiarity with block-based programming, its limitations in developing advanced software and its inapplicability to older age groups are mentioned as disadvantages of block-based coding tools (Mohamad et al., 2011).

A review of the literature reveals that studies addressing the advantages and disadvantages of block-based coding tools are limited. This research aims to contribute to the literature by determining the views of information technology and software teachers regarding the block-based coding tools they use in coding education. In this regard, the main research question is “What are the views of Information Technology and Software teachers on block-based coding tools?” and the following sub-questions have been explored:

1. What are the advantages of block-based coding tools?
2. What are the disadvantages of block-based coding tools?
3. What are your views on the usability of block-based coding tools (Scratch, Lego Mindstorms, App Inventor) in the classroom, both from your and your students’ perspectives?

METHODOLOGY

In this section, information about the research design, study group, data collection tool and data analysis is shared.

Research Design

The study employed a case study design, one of the qualitative research methods. According to Yıldırım and Şimşek (2016), the aim of a case study is to reveal results related to a specific situation. In other words, the factors related to a particular situation (individuals, environment, events, etc.) are examined with a holistic approach, focusing on how they influence the situation and how they are affected by it.

Study Group

In the study, a purposive sampling method was used to allow for in-depth examination of cases that are thought to hold rich information. The participants in this study consist of Information Technology and Software teachers ($n=16$) who are employed in public and private schools affiliated with the Ministry of National Education (MEB) and are actively involved in robotic and coding education. The interview questions were sent to 18 Information Technology and Software teachers with 3-4 years of experience using coding tools in both public and private institutions and data were collected from 16 participants. Of the teachers who participated in the study, 12 are female and 4 are male. Additionally, 2 of the teachers work in public schools affiliated with MEB, while 14 are employed in private educational institutions affiliated with MEB. The demographic information of the teachers is presented in Table 1 below.

Table 1: Demographic Information of the Teachers

		f
Gender	Female	12
	Male	4
Employment Status (Affiliated with MEB)	Public	2
	Private	14
Educational Background of the Teachers	Bachelor's Degree	13
	Master's Degree	3
Total		16

Data Collection Tool

In this study, a semi-structured interview form containing open-ended questions, one of the qualitative data collection tools, was used. During the preparation of the interview questions, a literature review was conducted and a pool of 11 questions was created. This pool was then sent to three academics in the field of education to obtain expert opinions. Based on the feedback received, five questions were selected for the interview form. For content validity, the prepared

interview form was reviewed by two experts in the field of Computer Education and Instructional Technology (CEIT) and for Turkish sentence structure and grammar, feedback was obtained from an expert in the Turkish language. A pilot study was conducted with two Information Technology and Software teachers in line with the feedback received. The pilot study identified questions that were either clear or unclear. After incorporating the feedback from field experts and the pilot study, necessary revisions were made and the final version of the interview form was prepared. The following questions were asked during the interview:

- In your opinion, what are the advantages of block-based coding tools?
- In your opinion, what are the disadvantages of block-based coding tools?
- What are your views on the usability of block-based coding software in the classroom from both your perspective and that of your students?

Data Analysis

In this study, the views of Information Technology and Software teachers regarding the use of block-based coding tools were analyzed using the content analysis method based on predefined themes. According to Yıldırım and Şimşek (2016), the main process in content analysis is to gather similar data under certain themes, organizing and interpreting them in a way that the reader can easily understand. To enhance the validity and reliability of the research, various measures were taken and efforts were made to strengthen credibility, consistency, transferability and confirmability (Fraenkel et al., 2012). The measures taken to enhance the validity and reliability are presented in Table 2.

Table 2: Measures Taken for the Validity and Reliability of the Study

Strategy		Measure
Validity	Internal Validity - Credibility	Interview questions were prepared with expert opinions.
		Interview questions were prepared through a literature review.
		The interview was conducted with high levels of interaction without shortening the duration.
Reliability	External Validity - Transferability	The sample size was kept as large as possible.
		The data collection process was explained.
	Internal Reliability - Consistency	Participants were informed about the process before the interview.
		Interview questions were discussed, and data collection tools were explained with guidelines, ensuring that the number of items did not negatively affect reliability.
Reliability	External Reliability – Confirmability	
	Expert opinions were sought for the synthesis and analysis of the interviews.	

FINDINGS

The data obtained during the research process were generally evaluated under two different categories: positive views on block-based coding tools and negative views on block-based coding tools. Table 2 provides information on the number of responses within these categories and subcategories. In Table 3, the frequency and percentage values of the data obtained in the study are analyzed and presented. A total of 16 teachers participated in the study. In some cases, the same teacher expressed views on multiple categories. Therefore, the same teacher was grouped under more than one category.

Positive Views of Information Technology and Software Teachers on Block-Based Coding Tools

Information Technology and Software teachers were asked about the positive aspects of block-based coding tools and the analysis of the findings is presented in Table 3. The results of the analysis regarding the positive views of Information Technology and Software teachers on block-based coding tools are as follows: 17.54% of the teachers (n=10) considered

the tools easy to use, 5.26% (n=3) stated that anyone can do it, 7.02% (n=4) mentioned the visual aspect, 5.26% (n=3) referred to the drag-and-drop functionality, 3.51% (n=2) noted that it builds students' self-confidence, 3.51% (n=2) mentioned that it enhances student motivation, 5.26% (n=3) indicated that it dispels the prejudice about the difficulty of coding, 7.02% (n=4) stated that it provides the foundation for learning advanced programming, 5.26% (n=3) observed that it develops programming logic in students, 3.51% (n=2) said that it ensures student participation in class, 3.51% (n=2) mentioned that it improves algorithmic thinking, 7.02% (n=4) said that it enables concrete learning for students, 5.26% (n=3) noted that it leads to the creation of digital products, 3.51% (n=2) highlighted that it promotes learning through fun, 3.51% (n=2) mentioned that it fosters an interest in programming, 5.26% (n=3) said that it gives students the feeling of playing a game and 8.77% (n=5) suggested that block-based coding tools can be used with younger age groups.

Table 3: Positive Views of Information Technology and Software Teachers on Block-Based Coding Tools

View	<i>f</i> (57)	% (100)
Easy	10	17.54
Young Age Group	5	8.77
Visual	4	7.02
Basic Infrastructure/Preparation	4	7.02
Concrete Learning	4	7.02
Accessibility for All	3	5.26
Drag-and-Drop Logic	3	5.26
Dispelling Prejudices	3	5.26
Programming Logic	3	5.26
Digital Product	3	5.26
Sense of Playing a Game	3	5.26

Building Self-Confidence	2	3.51
Providing Motivation	2	3.51
Ensuring Class Participation	2	3.51
Improving Algorithmic Thinking	2	3.51
Learning Through Fun	2	3.51
Fostering Interest in Programming	2	3.51

Under the “Ease of Use” category, the views of 10 Information Technology and Software teachers in the “Easy” subcategory are highlighted in the following excerpts:

T₃: “Using more visual representations for younger age groups makes it easier for them to understand the content.”

T₆: “Students can easily code without knowing a language, simply by using the drag-and-drop logic.”

T₇: “Block-based coding facilitates communication. Since it is more concrete to the students, it is easier for them to understand.”

T₈: “Block-based tools provide great ease for beginners.”

T₉: “It helps 6-10-year-old students grasp the logic of software from a simple perspective.”

These views suggest that block-based programming tools are used because they offer ease of use for younger ages, are logically simple and provide more concrete representations.

Under the “Accessibility for All” category, the views of three Information Technology and Software teachers in the “Accessibility for All” subcategory are summarized in the following excerpts:

T₆: “It is suitable for all age levels and everyone can do it.”

T₇: “... Even people who do not know any programming languages can solve problems by reasoning and can build a programming algorithm.”

These views indicate that block-based programming is used because it allows even those unfamiliar with programming languages to code easily and develop algorithmic skills.

Under the “Contribution” category, the views of three Information Technology and Software teachers in the “Overcoming Prejudice” subcategory are highlighted in the following excerpts:

T₉: “... *For those introduced to coding at an early age, the complex-looking lines of code in text-based programming become more understandable through shapes. Thus, they can learn more easily without developing a prejudice against the subject.*”

T₁₀: “*It helps 6-10-year-old students grasp the logic of software from a simple perspective and allows them to create simple applications without assistance, eliminating their prejudices toward programming languages.*”

It is indicated that text-based programming can appear complicated to students, creating prejudices and that block-based programming should be used before transitioning to text-based programming to overcome these prejudices.

Under the “Basic Infrastructure/Preparation” category, the views of four Information Technology and Software teachers are highlighted in the following excerpts:

T₂: “*Introducing younger age groups to text-based programming.*”

T₆: “*It helps students build a foundation in basic algorithm structures such as loops, conditional commands and randomness.*”

T₈: “... *Block-based tools provide great ease for beginners. Therefore, since learners can easily create applications (compared to text-based programming), I believe that block-based tools play a significant role in fostering a love for programming and teaching basic programming logic to beginners.*”

Block-based programming tools are seen as essential for teaching basic programming concepts such as variables, loops and conditional structures.

Under the “Concrete Learning” category, the views of four Information Technology and Software teachers are summarized in the following excerpts:

T₆: “... *It provides more concrete learning experiences for students... Since it is supported by visually rich content (such as a robot, puppet, or electronic board), it offers more concrete learning experiences for students.*”

T₇: “... *It is easier for students to understand because it is more concrete to them.*”

The teachers emphasize that block-based programming fosters more concrete learning due to its visual richness, making it easier for students to grasp concepts.

Under the “Digital Product” category, the views of three Information Technology and Software teachers are highlighted in the following excerpts:

T₇: “... *Instead of writing code line by line, digital products can be created by dragging and dropping blocks.*”

T₁₄: “... *The more blocks used, the higher the quality of digital products that can be made.*”

These views highlight that digital products can be created using block-based programs, even if they involve a large number of blocks.

Under the “Fostering Interest in Programming” category, the views of two Information Technology and Software teachers are summarized in the following excerpts:

T₄: “*Since these tools can be used when the student’s readiness level for coding is low, they quickly help students develop an interest, curiosity and self-confidence in coding.*”

T₈: “... *I believe that block-based tools play a significant role in fostering a love for programming and teaching basic programming logic to beginners.*”

These views suggest that even when students have a low readiness level for coding, using block-based programming tools contributes to developing interest, curiosity and self-confidence in coding.

Under the “Sense of Playing a Game” category, the views of three Information Technology and Software teachers are summarized in the following excerpts:

T₅: *“Since block-based coding creates story-appropriate blocks (e.g., Scratch, code.org), it gives students a sense of learning something new or playing a game each time, thereby increasing their participation in class.”*

T₁₅: *“... Since it gives the feeling of playing a game while coding, implicit learning takes place and students learn coding without feeling the pressure of learning responsibility.”*

These views emphasize that block-based programming gives the feeling of playing a game, facilitating implicit learning.

Under the “Young Age Group” category, the views of five Information Technology and Software teachers are summarized in the following excerpts:

T₉: *“... The advantage of block-based coding tools is that for those introduced to coding at an early age, the complex-looking lines of code in text-based programming become more understandable through shapes.”*

T₁₄: *“Block-based coding should definitely be used for younger age groups.”*

Overall, block-based coding applications are preferred due to their ease of use, accessibility to individuals without prior programming knowledge and their more concrete nature compared to text-based applications. It is emphasized that basic programming skills are easier to learn through block-based applications, leading to the creation of digital products. Additionally, participants noted that block-based applications

contribute to students developing a love for, curiosity about and self-confidence in programming.

Information Technology and Software Teachers' Negative Opinions on Block-Based Coding Tools

Information Technology and Software teachers were asked about the negative aspects of block-based coding tools and the findings are analyzed in Table 4.

Table 4: Negative Opinions of Information Technology and Software Teachers Regarding Block-Based Coding Tools

View	<i>f (18)</i>	<i>% (100)</i>
Limited-Restricted	9	50.00
Insufficient	1	5.56
Decreases Motivation	2	11.11
Reluctance to Transition to Text-Based Programming	2	11.11
Inability to Develop Advanced Software	2	11.11
Not Widely Used	1	5.56
Addictive	1	5.56

Table 4 presents the analysis of the negative opinions of Information Technologies and Software teachers regarding block-based coding tools. According to the analysis, 50% of the teachers (n=9) stated that block-based coding tools have limited and/or restricted use, 5.56% (n=1) found them insufficient, 11.11% (n=2) mentioned they lower student motivation, 11.11% (n=2) indicated that students do not want to transition to text-based programming, 11.11% (n=2) emphasized the inability to develop advanced software, 5.56% (n=1) indicated that they are not widely used and 5.56% (n=1) believed they could lead to addiction.

Some of the opinions from the 9 Information Technologies and Software teachers under the Limited-Restricted subcategory of the Use category are provided below:

T₄: *“When coding, individual differences in cognitive abilities emerge and block-based tools reduce these differences compared to text-based tools, limiting the codes students can use.”*

T₅: *“... I believe that block-based coding is more limited in terms of creativity, as the coding education is restricted to the blocks available in the coding tool.”*

T₇: *“While block-based coding tools are still programming tools, they are more limited in terms of intervening in events or algorithms compared to a real programming language. You are forced to use the existing code blocks.”*

T₉: *“The disadvantage is that the actions you can perform are limited.”*

T₁₀: *“Compared to text-based coding, it is more limited and can restrict originality in students.”*

T₁₁: *“Advanced software cannot be developed. At some point, it limits the person doing the coding.”*

T₁₂: *“The tasks to be performed can sometimes be limited.”*

T₁₃: *“... If the number of blocks is low, you can only do basic coding, but it won't be the coding you envision. It will be limited.”*

According to these opinions, block-based programming tools are considered limited in terms of revealing individual differences and creativity skills compared to text-based tools.

The opinion from the 1 Information Technologies and Software teacher under the Insufficient subcategory of the Use category is provided below:

T₁₆: *“It may be insufficient for in-depth coding education.”*

According to this opinion, block-based tools are considered insufficient for programming education beyond the basics.

Some of the opinions from the 2 Information Technologies and Software teachers under the Decreases Motivation subcategory of the Teaching category are provided below:

T₄: *"... It reduces the motivation of students who have developed high-level thinking skills and want to do coding."*

T₆: *"... There can sometimes be a decrease in motivation when transitioning from block-based tools to text-based tools."*

According to these opinions, it is emphasized that block-based programming is much easier than text-based programming, leading to a decrease in motivation when transitioning to text-based programming. It is also noted that block-based tools fall short in meeting the needs of individuals with advanced thinking skills, which can further decrease motivation.

Some of the opinions from the 2 Information Technologies and Software teachers under the Reluctance to Transition to Text-Based Programming subcategory of the Teaching category are provided below:

T₆: *"... The visual richness and ease of use of block-based tools make students inclined toward simplicity and reluctant to transition to text-based programming."*

T₇: *"The visual appeal and ease of use lead students to get used to simplicity, making them reluctant to move on to text-based programming."*

T₁₁: *"Advanced software cannot be developed."*

T₁₅: *"It develops coding logic but does not allow for the production of all types of software directly."*

According to these opinions, it is stated that the visual richness and ease of use of block-based programming tools, which are seen as easier compared to text-based tools, make the transition to text-based programming more difficult and, as a result, hinder the development of advanced software.

The opinion from the 1 Information Technologies and Software teacher under the Not Widely Used subcategory of the General Opinions category is provided below:

T₁₀: *"Block-based coding tools are not as widely used compared to other tools."*

According to this opinion, it is stated that block-based programming tools are not as widely used as desired.

The opinion from the 1 Information Technologies and Software teacher under the Addictive subcategory of the General Opinions category is provided below:

T_6 : “In some cases, when used excessively, attachment to these tools increases.”

According to this opinion, it is noted that the excessive use of block-based programming tools, which attract students’ interest, can lead to dependency on these tools.

Opinions of Information Technologies and Software Teachers on the Usability of Block-Based Coding Tools (Scratch, Lego Mindstorms, App Inventor) in the Classroom

Information Technologies and Software teachers were asked the question: “What are your opinions on the usability of block-based coding tools in the classroom, both for yourself and your students?” The responses to this question and the frequency values of similar opinions are presented in Table 4.

Table 4: Opinions of Information Technologies and Software Teachers on the Usability of Block-Based Coding Tools

View	<i>f (30)</i>	<i>% (100)</i>
Simple and Practical	7	23,33
Suitable for Younger Age Groups	5	6,67
Interface Richness	2	6,67
Visual Design - Visual Programming	2	10
Robot - Puppet Control/Movement	2	6,67
Language Support	3	10
Compatibility Across Platforms	2	6,67
Up-to-date - Ability to be Updated	3	23,33
Entertaining	2	16,67
Integration with Other Subjects	2	6,67

According to Table 4, 7 teachers indicated that the tools were simple and practical, 5 teachers found them suitable for younger age groups, 3 teachers highlighted the presence of language support and the ability to be updated and 2 teachers mentioned features such as rich interface, visual design, robot-puppet control, entertainment, integration with other subjects and compatibility across platforms.

The opinions particularly emphasized the practicality and suitability of block-based tools, especially Scratch, for younger groups.

T₅: *"I actively use Scratch because I mostly teach younger age groups. I use Scratch Jr. for grades 1-2 and Scratch 2.0 or 3.0 for grades 3-4. The younger the student group, the more visually appealing it needs to be to capture their attention, which is why I use it. I also think Scratch is more practical because it interacts with other applications like Arduino and LEGO."*

T₅: *"The fact that the program has Turkish language support is a significant advantage,"* and Teacher 9: *"Having Turkish language support provides ease of use."* Both teachers expressed that the language support of block-based tools positively influences their usability.

One teacher explained that block-based tools are particularly suitable for young students in the field of robot programming, while text-based programming would be more challenging:

T₁₄: *"Middle school students must use them. Text-based programming would tire students out to control the robot and cause them to get stuck. I consider block coding more important in these tools."*

Teachers also offered opinions recommending block-based coding for developing programs suitable for different platforms:

T₇: *"App Inventor allows beginners to easily develop Android applications, even without knowing any programming"*

language, which increases its usability. The fact that students can use their own apps and games on their mobile phones boosts their interest and motivation for the class. Creating a digital product enhances students' self-confidence."

T₃: *"Nowadays, students grow up surrounded by technology devices such as tablets and phones. However, they have no idea how the applications on these devices are developed. With App Inventor, students not only gain insights into the development of these applications, but they also get the chance to test their small applications on a mobile device."*

CONCLUSION, DISCUSSION AND RECOMMENDATIONS

The aim of this study is to reveal the opinions of Information Technology and Software teachers regarding block-based programming education. In this context, a qualitative research approach was adopted and a semi-structured interview form was used as the data collection tool. Of the participating teachers, 75% (n=12) were female and 25% (n=4) were male. A total of 12.5% (n=2) of the teachers worked in public schools affiliated with the Ministry of National Education, while 87.5% (n=14) worked in private schools also affiliated with the Ministry of National Education.

The research findings revealed that Information Technology and Software teachers consider block-based coding tools as easy to use, accessible to everyone, foundational in establishing a basic infrastructure for students, effective in facilitating concrete learning, suitable for primary and middle school students and enabling the development of digital products. When examining similar studies in the literature, the research results align with findings that Information Technology and Software teachers consider block-based coding tools to be easy to use (Çatlak et al., 2015; Kalelioğlu & Gülbahar, 2014; Yükseltürk & Altıok, 2016), suitable for the age groups of primary and middle school students (Sırakaya, 2018) and

foundational for learning advanced programming languages. The tools are also viewed as facilitating concrete learning and enabling students to learn through enjoyment (Çağltay & Topalli, 2012; Genç & Karakuş, 2011; Yükseltürk & Altıok, 2016). The study concluded that block-based applications, particularly Scratch, increased student motivation. Similar studies have also noted that Scratch enhances motivation (Erol, 2015; Ruf, Mühling & Hubwieser, 2014; Ortiz-Colon & Romo, 2016).

On the other hand, it was found that Information Technology and Software teachers perceive block-based coding tools as having limited or restricted use. Teachers noted that students were reluctant to transition to text-based programming, that the tools did not allow for advanced software development and that they could decrease student motivation.

It was also concluded that Information Technology and Software teachers recommend block-based coding tools like Scratch and Code.org for teaching programming and tools such as Lego Mindstorms, Coding software and/or MakeBlock software for teaching programming with robots. For mobile programming compatible with Android platforms, they recommend the App Inventor application developed by the Massachusetts Institute of Technology. Based on the study's results, it is recommended to use Scratch for its practicality, suitability for primary and middle school levels, ability to enhance class efficiency, language support, entertainment value, facilitation of interdisciplinary communication and other advantages such as being free. For coding instruction involving robots, depending on the budget, Lego Mindstorms, which supports learning opportunities, project/product creation, concrete learning and increased class participation, can be preferred at the middle school level. However, due to its high cost, alternative robotic coding tools may also be used. For mobile software development, the MIT App Inventor software, which is available as both a web and desktop application, may be preferred as a block-based coding tool.

In this study, opinions were gathered from 16 Information Technology and Software teachers teaching at the primary and middle school levels. By increasing the sample size and collecting views from teachers working at different educational levels, more findings could be obtained.

REFERENCES

- Akçay, A., & Çoklar, A. N. (2018). Bilişim teknolojileri ve yazılım dersi öğretmen adaylarının programlamaya ilişkin algılanan öz yeterliklerinin farklı değişkenler açısından incelenmesi. *Kastamonu Education Journal*, 26(4), 2163-2176.
- Akpınar, Y., & Altun, A. (2014). Bilgi toplumu okullarında programlama eğitimi gereksinimi. *İlköğretim Online*, 13(1), 1-4.
- Alkan, A. (2019). Özel yetenekli öğrencilerin programlama dili öğretiminde Kodu Game Lab yazılımının problem çözme becerileri düzeyine etkisi. *Mehmet Akif Ersoy Üniversitesi Eğitim Fakültesi Dergisi*, 50(1), 480-493.
- Aslan, S., & Duruhan, K. (2019). Bilişim teknolojileri ve yazılım dersinde probleme dayalı öğrenme yaklaşımının öğrencilerin akademik başarılarına etkisi. *Çukurova Üniversitesi Eğitim Fakültesi Dergisi*, 48(1), 32-72.
- Çatlak, Ş., Tedal, M., & Baz, F. Ç. (2015). Scratch yazılımı ile programlama öğretimi durumu: Bir doküman inceleme çalışması. *Journal of Instructional Technologies & Teacher Education*, 4(3), 13-25.
- Demirer, V., & Sak, N. (2016). Programming education and new approaches around the world and in Turkey. *Eğitimde Kuram ve Uygulama*, 12(3), 521-546.
- Duran, M., Özdemir, F., & Kaplan, A. (2015). A research on the use of problem-based learning approach: Teaching of probability sample. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, 6(2), 250-284.
- Erol, O. (2015). *Scratch ile programlama öğretiminin bilişim teknolojileri öğretmen adaylarının motivasyon ve başarılarına etkisi* [Doktora tezi, Anadolu Üniversitesi]. <http://libra.anadolu.edu.tr/tezler/2015/385086.pdf>
- Fraenkel, J. R., Wallen, N. E., & Hyun, H. H. (2012). Internal validity. In *How to design and evaluate research in education* (pp. 166-183). New York: McGraw-Hill.
- Genç, Z., & Karakuş, S. (2011, September). Learning through design: Using Scratch in instructional computer games design. In *5th International Computer & Instructional Technologies Symposium* (pp. 22-24).
- Göksoy, S., & Yılmaz, İ. (2018). Bilişim teknolojileri öğretmenleri ve öğrencilerinin robotik ve kodlama dersine ilişkin görüşleri. *Düzce Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 8(1), 178-196.
- Kalelioğlu, F. (2015). A new way of teaching programming skills to K-12 students: Code.org. *Computers in Human Behavior*, 52, 200-210.
- Kalelioğlu, F., & Gülbahar, Y. (2014). The effects of teaching programming via Scratch on problem solving skills: A discussion from learners' perspective. *Informatics in Education*, 13(1), 33-50.
- Kim, H., Choi, H., Han, J., & So, H.-J. (2012). Enhancing teachers' ICT capacity for the 21st century learning environment: Three cases of teacher education in Korea. *Australasian Journal of Educational Technology*, 28(6), 965-982.
- Kukul, V., & Gökçearslan, Ş. (2014). Investigating the problem-solving skills of students attended Scratch programming course. In *8th International Computer & Instructional Technologies Symposium*, Trakya University Edirne.

- Mohamad, S. N. H., Patel, A., Latih, R., Qassim, Q., Na, L., & Tew, Y. (2011, July). Block-based programming approach: Challenges and benefits. In *Proceedings of the 2011 International Conference on Electrical Engineering and Informatics* (pp. 1-5). IEEE.
- Ortiz-Colon, A. M., & Romo, J. L. M. (2016). Teaching with Scratch in compulsory secondary education. *International Journal of Emerging Technologies in Learning (iJET)*, 11(2), 67-70.
- Pala, F. K., & Türker, P. M. (2019). Öğretmen adaylarının programlama eğitimine yönelik görüşleri. *Kuramsal Eğitimbilim Dergisi*, 12(1), 116-134.
- Papert, S. (1993). *The children's machine*. Manchester, NH, England: Technology Review.
- Ruf, A., Mühling, A., & Hubwieser, P. (2014, November). Scratch vs. Karel: Impact on learning outcomes and motivation. In *Proceedings of the 9th Workshop in Primary and Secondary Computing Education* (pp. 50-59).
- Saygıner, Ş., & Tüzün, H. (2017, May). Programlama eğitiminde yaşanan zorluklar ve çözüm önerileri. In 11. *Uluslararası Bilgisayar ve Öğretim Teknolojileri Sempozyumu* (pp. 24-26). Malatya: İnönü Üniversitesi.
- Sırakaya, M. (2018). Kodlama eğitimine yönelik öğrenci görüşleri. *Ondokuz Mayıs Üniversitesi Eğitim Fakültesi Dergisi*, 37(2), 79-90.
- Talim Terbiye Kurulu [TTK]. (2017). *Bilişim teknolojileri ve yazılım dersi öğretim programı*. Ankara: MEB.
- Talim Terbiye Kurulu [TTK]. (2018). *Bilişim teknolojileri ve yazılım dersi öğretim programı*. Ankara: MEB.
- Weintrop, D., & Wilensky, U. (2017). Comparing block-based and text-based programming in high school computer science classrooms. *ACM Transactions on Computing Education (TOCE)*, 18(1), 1-25.
- Yıldırım, A., & Şimşek, H. (2016). *Sosyal bilimlerde nitel araştırma yöntemleri* (10. bs.). Ankara: Seçkin Yayıncılık.
- Yıldızlar, M. (2013). *Öğretim ilke ve yöntemleri*. Ankara: Pegem Akademi.
- Yükseltürk, E., & Altıok, S. (2015). Bilişim teknolojileri öğretmen adaylarının bilgisayar programlama öğretimine yönelik görüşleri. *Amasya Üniversitesi Eğitim Fakültesi Dergisi*, 4(1), 50-65.