

Article

The Effect of Coding Teaching on Students' Self-Efficacy Perceptions of Technology and Design Courses †

Celalettin Özden ^{1,*}  and Murat Tezer ² 

¹ Educational Programs and Instruction, Education Faculty, Near East University, 99138 Nicosia (via Mersin 10 Turkey), North Cyprus, Cyprus

² Department of Mathematics Teaching, Education Faculty, Near East University, 99138 Nicosia (via Mersin 10 Turkey), North Cyprus, Cyprus; murat.tezer@gmail.com

* Correspondence: cozden2001@gmail.com; Tel.: +90-505215222

† Note: This article was based on Celalettin Özden's Ph.D. Thesis.

Received: 25 September 2018; Accepted: 16 October 2018; Published: 22 October 2018



Abstract: The aim of this research was to examine the effect of coding teaching on students' self-efficacy perceptions of technology and design courses. A consecutive-explanatory design in a combined research model was used in this study. The qualitative dimension of the research was carried out through a random experimental design with pre/post-tests with controlled groups, and the qualitative dimension was conducted through interview techniques. The participants were seventh grade students from Şehit Hüseyin Ruso Secondary school in Lefkoşa, North Cyprus, in the 2016–2017 academic year. The quantitative data were collected through the “Technology and Design Course Self-efficacy Perception Scale”, developed by the researcher, and the qualitative data were collected through interview forms oriented around “Coding Teaching and Revised Technology and Design Course Teaching Program”. The quantitative data were analyzed through related and unrelated sampling *t*-test, Cohen's *d* effect size coefficient, and two-way variance analyses, and the qualitative data were analyzed through content analysis. A significant difference was observed between the students, taking coding teaching, and self-efficacy perceptions of Technology and Design courses compared to the control groups. Besides, it was noted that students taking coding teaching courses exhibited a positive attitude towards the course, and they practiced their metacognitive thinking and social skills at top levels. Based on the students' views, it is obvious that coding teaching has a positive effect on their self-efficacy perceptions of the subject.

Keywords: technology and design; curriculum; coding; software teaching; efficiency; scratch; sustainability

1. Introduction

Rapid developments in Information and Technology deeply affect countries' education systems and teaching programs accordingly. Recent developments in the field of technology have urged educational institutions as open systems, to make changes. In order to maintain sustainability and functionality, educational institutions should be sufficient in raising the human resources that are needed for occupational life [1]. In this regard, urgent revision of educational programs or new curricula that are essential in raising human resources is needed [2,3]. Today, the integration of Information and Communication Technologies (ICT) into learning experiences, particularly necessitates different contents and materials in teaching. In this regard, particularly in order to teach the subjects that entail innovative and creative thinking, like technology and design, and to raise the interest of

young people (the Z generation) who are obsessed with technological tools such as tablets and smart phones, integrating computer-supported technology into teaching has become a necessity [3–7].

Studies done on the use of ICT in teaching emphasize the need for coding-oriented activities to raise manpower with problem solving and social skills to respond to the requirements of the 21st century [8,9]. Coding is considered to be a 21st century skill, enhancing problem solving, critical thinking, communication, collaboration-flexibility, computational thinking, and innovative thinking skills [6,8,10]. The development of such skills is among the main aims of a technology and design course. However, in order to achieve this, program development should be an on-going and consistent process.

In several studies into Technology and Design courses, the researchers stressed the need for the development of teaching programs, in-service training for teachers, and learning environments rich in teaching-learning facilities. Such considerations state that students will be able to put their metacognitive skills into practice at high levels, and this will be more effective in achieving the aims of the subject [11–17].

In light of these considerations, the need arises for equipping the course with suitable learning experiences that respond to the core of the subject, and a process-oriented evaluation, rather than an outcome-oriented evaluation. Therefore, the students always have to be active during learning processes. Meanwhile, an environment addressing the expectations of a new generation of students, offering project-based learning and supporting them to adopt and continue with their projects, is essential. Project-based learning is a process that is based on solving problems through team-work by coming up with innovative ideas and a student-centered interdisciplinary approach in which students can have access to the information themselves [18–20]. This approach can be practiced by activating students' problem-solving, logical thinking, and social skills [21–24]. At this point, students need to use digital technologies such as tablets, smart-phones, personal computers, and electronic boards that they are familiar with, in learning environments [3]. One of the ways to make use of this digital technology is coding teaching, which can be referred to starting in primary school to higher educations at every level.

Coding teaching develops students' critical and computational thinking skills, and makes them innovative thinkers and productive individuals, rather than being passive consumers [8]. Coding teaching is basically the teaching of computer programming, which is a process of transferring and computerizing complex information using 1 and 0 symbols through complex applications in computer language. Here, programs such as Scratch and coding teaching have emerged to facilitate the complex processes in programming language [25]. Unlike the topics in universal programming teaching, coding teaching supports students in many ways, rather than just developing their skills in one-way software writing. This makes it easy for students to use their learning outcomes in different fields, and transfer information and skills among disciplines. Coding teaching contains algorithms, which focus on changes and improvements in students' cognitive thinking habits in solving mathematical problems [26].

Coding, including algorithmic thinking, has been integrated not only in the computer sciences, but in many other fields, too. This integration has been cited in several studies dealing with coding teaching in different countries, e.g., technology-supported maths learning [27], gamification through mathematical coding [28], programming teaching to pre-school children [29], academic readiness and success [30], cooperation in the arts [31], leadership and the role of youth in online creative communities [32], shaping the future of Maths teaching [33], coding in Maths learning, interactive curriculum and the use of digital teaching platforms [34]. In order to spread coding teaching, the digital learning environments should renew their curricula and teaching methods to be supported by computers [8].

In a report by European Schoolnet 2014 [35] it is stated that coding teaching in logical thinking, problem solving, and human interaction skills has been introduced, or that preparations are in progress to introduce it into the curriculum in 20 European countries, including Turkey. At present, Bulgaria, South Cyprus, the Czech Republic, Denmark, Estonia, Greece, Ireland, Italy, Lithuania, Poland, Portugal,

and England are part of this curriculum, but its implementation is still at the planning stage in Spain, Belgium, Flandre, Finland, France, Luxemburg, Holland, and Turkey [35].

In today's world, coding computer programs has become as important as being literate. Websites such as Scratch, Khan, Academy, Alice, Code.org, and programs for coding teaching are available to people, free of charge. Demir [36], in an article titled, "Teach Your Child Code Writing", published in the Turkish issue of Harvard Business Review, cited from Job's interview, "Everybody should learn computer programming because it teaches one how to think".

In environments where self-efficacy is taught, there are factors such as intelligence, age, stimulant, and attention as well as self-efficacy perceptions affecting learning processes. Self-efficacy is an individual's skill in performing and applying in a particular case. In a broader sense, it is also an individual's belief in exhibiting certain behaviors [37]. Self-efficacy is an individual's belief in the potency or the scope of his ability to achieve tasks and reach aims [38,39]. Arsari & Khan [40] explain self-efficacy as the belief in one's strength to influence cases. This concept is most effective in one's strength and choices. Bandura [41] argued that self-efficacy beliefs affect people's way of thinking, feelings, how they motivate themselves, and how they behave.

Bandura specified that strongly perceived self-efficacy and more active linear efforts interact with each other. In other words, self-efficacy expectations are crucial determinants for the choice of activities when relevant skills and demands are considered [42]. Self-efficacy perception has four main sources. These are; experiencing a similar behavior directly (complete and proper experiences), experiences adapted from someone (observing someone's success or failure, social modelling), verbal persuasion (by an authority), and psychological and emotional situations [43].

According to social learning theory, the environment is an important factor on the development of self-efficacy perception. Learning environments are the ones in which an individual mostly interacts [44]. In teaching-learning environments, every teaching program includes skills and sufficiencies that are peculiar to it. There are several studies in the measurement of Mathematics, Geometry, Information Technologies etc., that are related to self-efficacy. In these studies, it has been found that individuals with high levels of self-efficacy are more insistent on solving a problem or succeeding in a task [45–52].

Studies, in literature on the effect of coding on self-efficacy, were reviewed. Ihmaid [53] found out in a study in which learning English words and self-efficacy through Scratch were examined, that there were significant differences among self-efficacies related to learning English words. Meanwhile, it was also found out that Scratch was more effective compared to traditional methods, created a better learning environment, raised student motivation and in-class social interaction, and lessened teacher dominance. In their study, Yükseltürk and Altıok [54] came to the conclusion that Information Technologies increased teachers' self-efficacy perception of programming to a significant level after experiencing Scratch use. In another study, Kalokiannakis and Papadakis [6] found out that using Scratch Jr raised preservice kindergarten teachers' self-efficacy in computational thinking, and developed positive behaviors towards using coding in the future. In their study, Giannakos, Hubwiser & Ruf [55], stressed that learning environments such as Scratch, raised students' self-efficacy perceptions of programming. Similarly, Abo Oda stated in a study that Scratch increased students' self-efficacy (Adap. Ihmaid [53]).

In North Cyprus, labor and handwork are still the main tools in running Technology and Design courses, and such an approach naturally does not add to students' daily learning, and to their better perception of self-efficacy in the subject matter. Maviş [56] stated that technology and design courses were supported by Information Technologies (IT) in USA, and in EU countries. In this respect, other countries can make use of IT in learning experiences. In North Cyprus, studies in coding teaching are a new process, and they have been recently introduced into IT. This process can be accelerated by integrating coding teaching into technology and design programs. So far, a study examining the effect of using coding or programming on the perception of self-efficacy in technology and design courses

has not been presented in the literature. Therefore, the subject in question still remains an issue to be researched.

This research investigated the effect of coding teaching on students' self-efficacy perceptions of technology and design courses. Although coding and technology teaching are worldwide current issues, no any satisfactory studies in the development of projects and programs in these fields have been done in North Cyprus. Therefore, it is hoped that this will be a sample study showing that both technology-designing and software courses can interact. In this regard, it is anticipated that this study will also contribute to possible alterations in programs in coding and technology teaching. In addition, it is also hoped that this study will encourage teachers to adopt new techniques, and to receive training in coding teaching.

1.1. Aim of the Research

This research aimed to specify the level of the effect of coding teaching in technology and design courses on students' self-efficacy perceptions of the course. In this respect, answers were sought for the following questions.

In Technology and Design Courses

1. Is there a significant difference between the pre- and post-test self-efficacy scores by the experimental and control groups?
2. Is there a significant difference between the post- and permanency-test self-efficacy scores by the experimental and control groups?
3. How do students, upon receiving coding teaching, feel about the effectiveness of technology and design courses?

1.2. Limitations

1. This research was conducted among seventh grade students at Şehit Hüseyin Ruso Secondary school, North Cyprus, in the 2016–2017 academic year.
2. This study was limited to “technology and design, architectural design, mechanical design, renewable energy technologies, technologies facilitating life for the handicapped, advertisement and free-designing”.
3. Personal computers and electronic boards were used in teaching activities in experimental research.

2. Materials and Methods

The research was based on an ordered-explanatory design, one of the mixed methods that were classified by Creswell [57], in which both qualitative and quantitative methods were used at the same time. The quantitative dimension of the study was conducted in a semi-experimental pre- and post-test with the control groups. At the beginning of the research, the experimental and control groups were given pre-tests, and at the end they were given post-tests. Six weeks after the completion of the study, a permanency test was conducted. As for the qualitative dimension of the study, the experimental group of students were given interviews. A content analysis was done for the data collected through semi-structured interview forms. The Scratch program, which can be accessed free of charge, and is used in Turkish and in many other disciplines, was preferred in the coding process. Before proceeding with the study, a contact was established with the Ministry of National Education, and official permission was obtained to carry out this research.

The revised technology and design teaching program with the experimental group and the traditional Technology and Design program with the control group was followed. Both groups were taught by the researcher. The research lasted for eight weeks. During the experimental study, an

electronic board and an equipped classroom was used. All the external disturbing factors were minimized as much as possible. The experiment design is shown in Table 1.

Table 1. Experimental design.

		Pre-Test	Procedure	Post-Test	Interview	Permanence Test	Analyses
Needs Analysis and Teaching	Picking the Experiment group	Self-efficacy application	Practising revised Tech. and design Program	Self-efficacy scale application	Interview with the Students	Self-efficacy Scale application	Analyzing scales and reporting
	Picking Control Group	Self-efficacy Scale application	Practising usual Tech. and design Program	Self-efficacy scale application		Self-efficacy Scale application	Analyzing Scales and reporting

Integrating Coding Teaching into a Technology and Design Teaching Program

1. A needs analysis in technology and design course was done among 613 students, picked through a stratified sampling method from State schools in North Cyprus. The result of the analysis (Cronbach's Alpha 0.91) showed that 68.9% of the seventh grade students expressed the need for a revised technology and design teaching program (Table A1). In a study, Büyüköztürk [58] stated that a 10% sampling size would be sufficient to raise the level of representation and minimize errors. In this needs analysis, the targeted sampling rate for students was 20%. The participants in this research were 3116 seventh grade students. This number was specified in the light of the current information obtained from the Office of Secondary Education.
2. The targeted outcomes of the program were determined according to the items in the needs analysis. The outcomes were classified on the basis of cognitive levels (Table A2).
3. With the help of university experts, Technology and Design course plans consisting of coding, were made and administered to the experimental group to reach the aims (Table A3).
4. Student-centered lesson plans were written and put into practice and the students were kept active, mostly doing group-work activities. The lessons were prepared in an ASSURE teaching design, in which paper, pencils, electronic boards, and laptops were used as teaching tools. Examples of students' projects can be seen in Supplementary Materials.

Experimental Study Process (Program Application and Procedure Steps)

Week 1: Administering a self-efficacy pre-test to experimental and control groups in a Technology and Design course. Discussing with the students the differences and connections between technology and design concepts. Providing various examples from different fields of science. Emphasizing the importance of innovation, design, designing principles, designing processes and line elements.

Week 2: Writing algorithms using pen and pencil. Explaining the order of the process of the algorithms that they wrote. Emphasizing the importance of ordering the process. Assigning students, who could see that one piece of code could affect the whole program, to write application samples of movement, control and the menus through the Scratch program.

Week 3: Practicing changing two-dimensional figures into three-dimensional ones. Presenting three-dimensional figures and various architectural designs on the computer using the Scratch program. Relating the activities to Geometry.

Week 4: Turning characters by certain degrees and moving them forwards and backwards so that they could be seen at different angles. Assigning students to come up with movable mechanical designing ideas. Relating these activities to Mathematics and other subjects. Assigning students to do evaluations among themselves.

Week 5. Students choose stages to fit the characters and draw the figures through different angles. Students come up with ideas about recycling and energy. Students relate these activities to mathematics and other subjects. Students evaluate their own and peer's ideas about algorithms and projects.

Week 6: Forming new characters and designing elements through the Scratch program to suggest solutions to any problems around. Preparing animations using these characters. Producing and presenting ideas to help facilitate life for the handicapped, children, or elderly people. Relating these activities to Geometry, Arts, and Science subjects.

Week 7: Designing characters according to the changes in the images of objects from a distance. Placing these characters on the stage in different locations. Moving characters by special effects. Preparing advertisements for a product of their own choice. Relating these activities to Geometry, Mathematics, Arts and Science subjects. Evaluating their own projects.

Week 8: Independently design products of their own and needs. Animating their products on the Scratch program. Evaluating their products among themselves.

The control group was not interfered with in any way, and the traditional teaching program was followed. All the participants were given pre-post and permanency self-efficacy tests.

2.1. The Participants

The participants were picked through a simple random sampling method, a method in which every unit composing the participants has an equal chance to be in the sampling group [58]. The participants were the seventh grade students from Şehit Hüseyin Ruso Secondary school, North Cyprus, in the 2016–2017 academic year. The age average was 13, and 70% of the students had computers and Internet in their houses. The students came from families with average (above minimum wage) and low level (minimum wage) income. The research was carried out in a state school. All the participants were volunteers. Before forming the study group, the seventh grade students' age ranges and their academic success scores were obtained from the school administration. Among these, the students with the highest and lowest scores were exempted, and the experimental and control groups whose age ranges and academic success rates were close, and were selected by lot. Each group was composed of 30 students, with an age range of 13. In order to specify the balance between the experimental group (17 male, 13 female) and the control group (16 female, 14 male), an independent sampling *t*-test analysis was given to the participants. The result of the test did not show a statistically significant difference in their self-efficacy perceptions. This result revealed that the pre-test results of both groups were well balanced. According to the pre-test results of the self-efficacy scale, the arithmetic mean of the experimental group was $x = 81.6$, the control group $x = 80.6$, and $p = 0.841$, $p > 0.05$, which did not show a statistically significant difference.

2.2. Data Collection Tools

A "self-efficacy perception scale", developed by the researcher for the seventh grade students, was used in collecting quantitative data (Table A4). This scale was developed through gratified sampling method. Out of 3116 specified students, only 515 participants were reached. The scale was delivered in a five-point Likert type. The structural validity of the scale, approved by nine subject and three language experts, was tested through exploratory and confirmatory factor analysis, and its applicability was confirmed. With respect to Exploratory Factor Analysis (EFA), the calculation was as Kaiser-Meyer-Olkin Test (KMO) = 0.935; $X^2 = 6401.599$; $p < 0.01$. The explained total variance was 50.65%. With respect to Confirmatory Factor Analysis (DFA), the calculation was as Chi Square Test (CMIN/DF) = 1.8; Root Mean Square Error of Approximation (RMSEA) = 0.039; Tucker-Lewis Index (TLI) = 0.90; Comparative Fit Index (CFI) = 0.912; Goodness of Fit Index (GFI) = 0.894; Adjusted Goodness of Fit Index (AGFI) = 0.878; Standardized Root Mean Square Residual (SRMR) = 0.07. The result of the analysis was an indication of a balanced calculation of the values [59], and a reasonable distribution of the factor structure. The Cronbach's Alpha value of the self-efficacy scale was 0.93, which showed a high level of reliability. The factors of the scale were as follows; Comprehension

factor (three items), Basic Application factor (six items), Further Application factor (five items), Formal Analysis factor (five items), Further Analysis factor (six items) Synthesis factor (seven items), and Assessment factor (six items). The scale was composed of 38 items, and its validity and reliability calculation was confirmed through SPSS 23 and AMOS 22 package programs.

Semi-structured interview forms consisting of “Qualitative Research Questions in Teaching of Technology and Design Course Program Revised with Coding Teaching” were administered to collect qualitative data. The interview forms included five open-ended questions and their sub-dimensions. Six experts were consulted about the qualitative questions. The interviews were conducted in May, 2017 at the end of an experimental process. For the reliability of the content analysis the qualitative data was coded, categorized, and themed by two expert analysts.

Statistical Analysis: The qualitative data obtained from the pre-, post-, and permanency tests indicated a reasonable distribution. Tabachnick and Fidell [60] state that the skewness and kurtosis values of the variances between -1.5 and $+1.5$ are sufficient for a reasonable distribution. The same distribution was observed in this research. In the light of these findings, parametric tests and analyses were done. In the analysis of the quantitative data a *t*-test, Cohen’s effect size coefficient, for mixed measurements two-way variance analyses SPSS package program was applied. The significance value, $p < 0.05$ was noted. The coding, categorizing, and theming processes were fulfilled through the NVIVO 10 package program.

3. Results

In this part, the quantitative data from the research were analyzed, presented in tables, and interpreted. The quantitative data were subjected to content analysis, and the findings were presented in diagrams, tables, and from direct quotations.

The first research question was to examine any differences between the pre- and post-test scores of the experimental group students in terms of self-efficacy in the Technology and Design course. The *t*-test results are as in Table 2.

Table 2. The experimental and control group self-efficacy pre- and post-test-related samplings *t*-test analysis.

Groups	Tests	N	\bar{X}	sd	df	<i>t</i>	<i>p</i>
Experiment	Pre-test	30	81.63	16.64	29	7.188	0.00
	Post-test	30	114.70	17.21			
Control	Pre-test	30	80.66	20.35	29	−2.961	0.006
	Post-test	30	91.03	15.30			
Experiment	Post-test	30	114.70	17.21	58	5.628	0.00
Control	Post-test	30	91.03	15.30			

The related samplings *t*-test analysis for the experimental group indicated a significant difference in favor of the post-test evaluations after practising the coding revised technology and design course program, and the pre-test evaluations ($t_{(29)} = 7.188, p < 0.05$) before practising the program.

As for the result of the control group related samplings *t*-test analysis, a significant difference between the post-test evaluations after practising the traditional Technology and Design teaching program was observed in favor of the pre-test evaluations ($t_{(29)} = -2.961, p < 0.05$) done at the beginning of the program.

According to the results of the experimental and control groups, self-efficacy was unrelated to the sampling post-test analysis ($t_{(58)} = 5.628, p < 0.05$) a significant difference was observed in favor of the experimental group. The experimental group had a higher arithmetic average compared to the control group. The Cohen’s *d* value was calculated as ($d = 1.45$). The $0.8 < 1.45$ effect size can be assumed as a strong effect. The results of the self-efficacy post-test revealed a stronger effect of the revised technology and design course programs on self-efficacy perceptions compared to traditional programs.

The second question was to examine any significant differences between the scores by both control and experimental groups in post- and permanency self-efficacy tests in Technology and Design courses. The results are shown in Table 3 below.

Table 3. The related samplings *t*-test analysis between post- and permanency self-efficacy tests given to the control and experimental groups.

Groups	Tests	N	\bar{X}	Ss	df	<i>t</i>	<i>p</i>
Experiment	Permanency	30	113.96	15.72	29	0.216	0.831
	Post-	30	114.70	17.21			
Control	Permanency	30	93.40	16.85	29	−0.699	0.490
	Post-	30	91.03	15.30			
Experiment	Permanency	30	113.96	15.72	58	4.885	0.00
Control	Permanency	30	93.40	16.85			

A statistically significant difference between the self-efficacy post and permanency test ($t_{(29)} = 0.216, p > 0.05$) given to the experimental group was not noted. In such a case, it can be assumed that the permanency test evaluation levels are close to the post-test levels. In the result of the analysis, a statistically significant difference was not observed between self-efficacy post- and permanence tests ($t_{(29)} = -0.699, p > 0.05$). Thus, the permanence level of the control group taught through traditional education programs remained close to post-test evaluations.

The unrelated samplings *t*-test analysis ($t_{(58)} = 4.885; p < 0.05$) administered to the self-efficacy permanence test scores of the experimental and control groups showed a significant difference in favor of the experimental group.

In order to compare the evaluation results of the related and unrelated samplings, a two-way variance analysis was done for the mixed measurements. At the end of the analysis, the calculations were as Box's *p* ($0.306 > 0.05$); Wilk's Lambda's *p* (< 0.01); Mauchly's *p* ($0.009 < 0.05$); Sphericity Assumed *p* ($0.00 < 0.01$). According to these values, the students' self-efficacy perceptions of Technology and Design courses ($F_{(2-116)} = 8.859, p < 0.01$) differ significantly in favor of the experimental group.

The diagram shows the results of pre-, post-, and permanency tests given to the experimental (group 1) and control (group2) groups. As it can be seen in Figure 1, the difference (measurement 2) between the post-tests of both groups was equal at the beginning of the study, but it significantly raised in favor of the experimental group at the end of the experimental process. The difference (measurement 2) in the permanency tests for both groups is quite high in favor of the experimental group. The difference in post and permanency tests in favor of the experimental group indicates a significant effect of technology and design teaching program on students' self-efficacy perceptions of the course.

The third question examined the students' views about the effectiveness of Technology and Design courses.

The quantitative findings are grouped under four themes; cognitive output, affective output, external output, and skills. Every theme has its sub-themes. The result of the content analysis in qualitative data is as follows.

As it can be seen in the Figure 2, four main themes and their sub-themes have formed a model. The distribution of the frequencies in the formation of this model is as in Table 4.

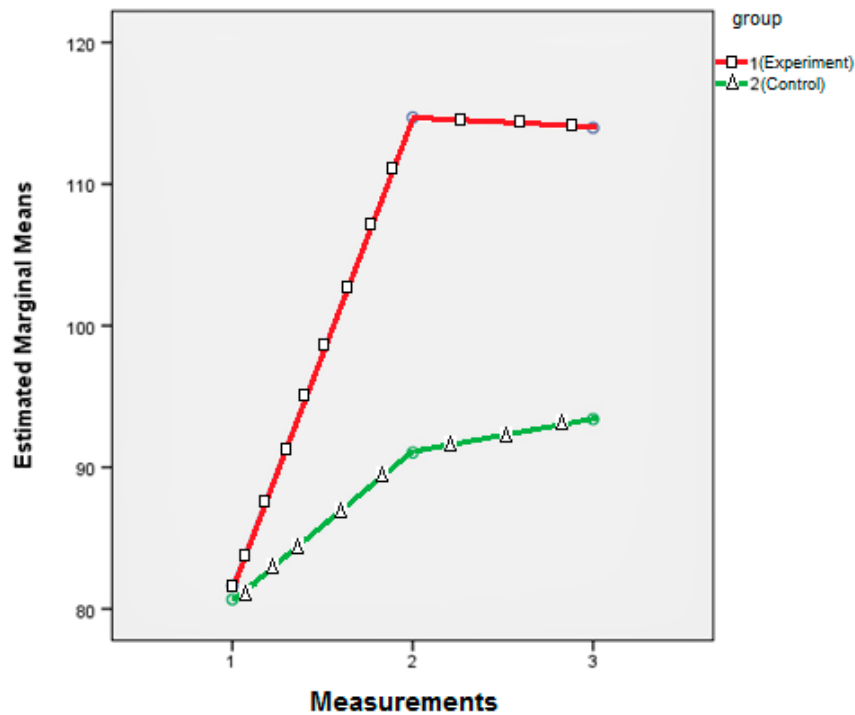


Figure 1. A Two-way variance diagram for self-efficacy mixed measurements.

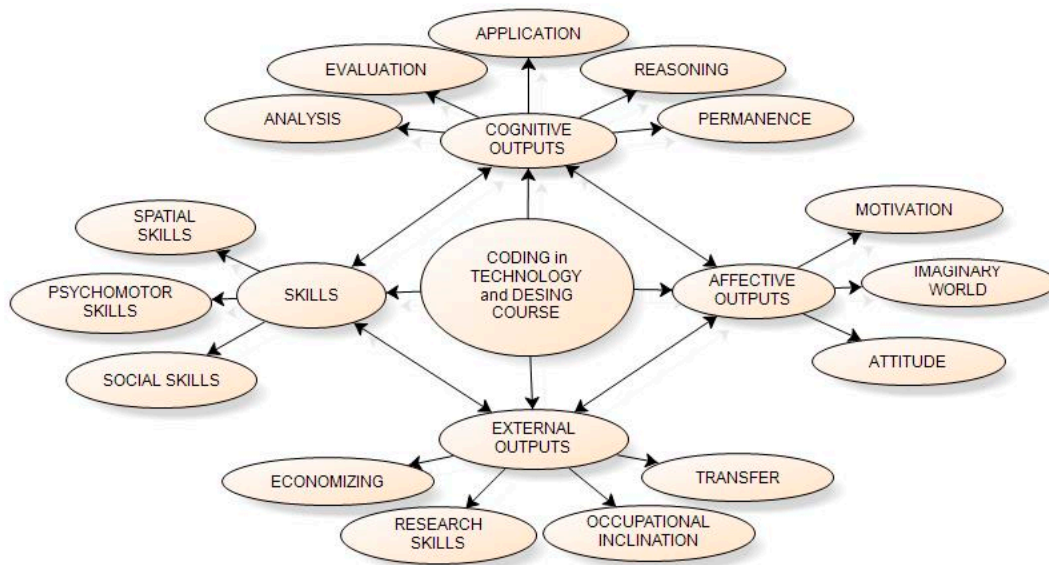


Figure 2. Coding model in the Technology and Design Course.

Every single sub-theme of the qualitative model was supported by direct quotations. The participants were identified in numbers from 1 to 30 (P1, P2... P30). As can be seen in Table 4, the theme of “Cognitive outputs” received the highest frequency distribution (37.68%). The “skills theme” is in the second place with a 23.16% frequency distribution. In the third place is the “Affective output” with 21.98% frequency distribution and the “External Output” is in the fourth place with 17.17% frequency distribution.

The Cognitive output theme consists of five sub-themes. In terms of frequency load, they are, “application”, “reasoning”, “evaluation”, “analysis”, and “permanency”.

Table 4. The participants' emphasis on the themes related to the qualitative questions.

Themes	Frequencies	%
Cognitive output		
Application	133	13.05
Evaluation	55	5.40
Analysis	36	3.53
Reasoning	131	12.86
Permanency	29	2.85
Total	384	37.68
Skills		
Use of spatial skills	81	7.95
Social skills	146	14.33
Psychomotor skills	9	0.88
Total	236	23.16
Affective output		
Attitude	125	12.27
Motivation	51	5.00
Imaginary world	48	4.71
Total	224	21.98
External output		
Transfer	70	6.87
Research skill	33	3.24
Economizing	49	4.81
Occupational inclination	23	2.26
Total	175	17.17
Overall	1019	100

In the Application theme, a big majority of the students stated that in class activities, they did coding, wrote projects, did technical drawing, animations, produced three-dimensional images, and gave concrete examples of their imaginations and explanations related to their projects etc., using the Scratch program. In these activities the students participated in doing and learning and this raised their self-efficacy perceptions. From this information it is assumed that coding has a significant effect on other themes as well. The students' views indicate that the more coding activities are done, the more active they become in their metacognitive skills.

P26 explained, "We play games by giving codes, create our own games, prepare advertisements and animate cartoons. We did a lot of things".

"Coding in Technology and Design courses made it different from other subjects, because we do not do any applications with other subject teachers" said P17.

"I now know that we can do many things by trying, but not in one try" added P6.

Reasoning is the theme in which students question the problems around them, and come up with solutions, produce a new innovative product of their own, an idea or a project, and develop their decision-making skills. As the participants stated, the reasoning theme interacted with other themes. While expressing views about reasoning, they raised ideas about the "application" theme as well.

P20 explained, "I couldn't come up with ideas of my own in other subjects as I could do in technology and design course. I would only try to find answers. Now I can think and produce. I have developed my skill in this respect".

"While design coding, we thought creatively and came up with new ideas" added P13.

“Using coding in technology and design courses helped me produce new and different designs....” admitted P16.

“We can think and fix something broken or out of order. We can think of quicker solutions to problems after receiving coding courses” remarked P11.

“It is fun to learn by coding. For example, I can reason differently in Maths lessons when I try to solve a problem” admitted P21.

The Evaluation theme mostly includes self-evaluation, peer assessment, and views about specifying the suitability of goods to certain criteria. It has been observed that students improved social communication through group work.

“I can do self-evaluation. I can evaluate a study of my own. I can find out my strengths and weaknesses” said P2.

“...I have a different way of criticizing myself and others” explained P26.

P17 stressed saying, “I can criticize a project in terms of its aims”.

The Analysis theme mostly deals with geometric shapes and the analysis of their measurements, and views about the solution of a problem.

“We now know how to explain procedures step by step...” stated P9,

“I can tell the width-height-depth of objects.” explained P15.

In the Permanency theme, the students compared their learning experiences in technology and design courses with other subjects with regards to remembering. They stated that this course was fun, but not merely memorizing, and added that one could express himself freely. They also admitted that it facilitated remembering because it was based on application.

“Coding applied technology and design courses are different from other subjects with regard to remembering, because we become more aware of everything around us and we start thinking about new changes” explained P24.

P15 admitted saying, “I could remember the topics in technology and design courses better than the topics in Maths and Turkish”.

The Skills theme contains three sub-themes. They are, in terms of their frequency loads, “social skills”, “practicing spatial skills”, and “psychomotor skills”. Social skills have quite a high effect on the development of students’ self-efficacy in technology and design courses.

In the Social skills theme, the students made common decisions and expressed their ideas freely, which developed their communication skills. In this theme, they mostly expressed views about “group work”, “self-expression”, “communication”, and “democratic decision-making”.

“We shared ideas and produced common projects. I can meet at a common idea with my class-mates” explained P7.

P13 said, “We shared tasks as we did group work. We had responsibilities. We finalized and presented our task”.

“Each of us did our part of the project in group work” added P14.

“This was the subject I could express myself without any worries. I had never experienced this before. Now, I can express my thoughts. In the past I was rather unsociable, but not anymore” admitted P10.

“Since I’ve started using coding in technology and design courses, I can express myself better. I was a bit shy, but I feel more comfortable now” P16.

“We get on well as friends and help each other” P1.

“Communication is not only necessary for the group, but for the whole class. I feel more confident when I talk to my friends and this adds to my life” explained P27.

Remarks by the participants about practising spatial and psychomotor skills.

“I wouldn’t be able to draw three-dimensional pictures if I hadn’t taken this course” P20.

“Drawing helps us develop our manual skills” P1.

The Affective output theme has three sub-dimensions. In terms of their frequency loads, they are; “attitude”, “motivation”, and “imaginary world”.

In the Attitude theme, the students expressed that they developed a positive attitude towards Technology and Design courses. Although this was one of the most well-coded themes, it closely interacted particularly with the Cognitive theme. The students’ revealed that particularly student-centered applied teaching strategies helped them develop positive attitudes. It has also been observed that the level of self-efficacy perceptions of the experimental group was high compared to the control group, and this was because the experimental group perceived coding as necessary issue and they were willing to use it.

P1 expressed views saying, “The new system, coding, made me enjoy the course more. It’s really interesting”.

“I’d feel missing a lot of things without this course” admitted P12.

“I love to be doing something on my own. Coding is fun” P18.

“Everybody should learn coding. Technology and Design courses would be boring without it” P10.

The Motivation theme deals with students’ views about their interest in the course, or the level of their willingness. This theme closely interacts with the attitude theme. Motivation is one of the determinants of self-efficacy.

P18 stated, “We presented our project through animation and I loved this. It was our first experience”.

“I enjoy learning by coding” admitted P21.

“Coding added a lot to technology and design courses. A boring activity turns into fun and is easier by coding” explained P30.

In the Imaginary world theme, students used their imaginations in new designs. They admitted that developing, expressing, and fulfilling their imaginations affect their self-efficacy perceptions.

P5 explained, “I can fulfill what I imagine by coding in technology and design courses. I couldn’t express my thoughts freely and do what I wanted before”.

“This should be a must course. Without it, I wouldn’t be able to fulfill my imaginations, but I can do it now by coding” said P30.

The External output theme includes four sub-themes. In terms of their frequency load, they are; “transfer”, occupational inclination”, “research skill”, and “economizing”.

The Transfer theme is related to one of the main aims of teaching programs, in which students use their learned outcomes in other fields effectively. In this context, the transfer theme explains student views about transferring experiences in technology and design courses fluently into teaching–learning processes, as well as other subjects.

“... I can use coding in Maths, Geometry, or Drawing lessons. I can apply it as word-game in English classes ...” explained P7.

“...for example, we can apply these codings on robots and make them move...” said P13.

“Coding can be used in other subjects as well, for example in friction force (Physics). We could clean rough surfaces and do animations by coding.” added P18.

“... I prepared games for my brother. He can learn letters and colors” said P27.

The participants expressed views about occupational inclination, research skills, and economizing saying;

“...it is for our good to receive coding teaching if we decide to study computer engineering” P13.

“I can surf the internet faster after taking this course.” P18.

“We can publish our outcomes...” P2.

“... I prefer coding when I work on a project. It’s easier and isn’t costly. I don’t pay anything for the material. I can correct my mistakes in my project by coding. Mistakes on paper cannot be corrected...” P16.

“... it would take a month to do a model in the past, but now we can do it on the computer by coding and it takes less time, about two weeks...” P1.

4. Discussion

This research aimed to investigate the effect of coding in technology and design courses on the seventh grade students’ self-efficacy perceptions. In this regard, a self-efficacy scale for the Technology and Design course will be included in the scope of the literature. The discussion of each research question is presented below.

The first question indicated a significant difference between the pre- and post-tests given to both the experimental and control groups. With respect to this indication, coding teaching in technology and design courses, and traditional technology and design courses have an effect on developing self-efficacy perceptions. However, when the post-tests scores of the groups are compared, it was observed that the experimental group had a higher score. This is because the coding teaching program worked better compared to traditional methods. As can be seen in Table 4, in the qualitative data, “application”, “reasoning”, “social skills”, and “attitude” themes have the highest frequencies. The participants stated that they developed positive attitudes towards the course by doing and learning, which indicates a development in their self-efficacy perceptions. The participants’ views support the results of the post tests in this research. Similarly, Tok [61], Eymur & Çetin, [62], Kaya [63], Çoban & Sanalan [64], and Şensoy & Aydoğdu [65] found out that in experimental studies, applied teaching activities based on true experiences rather than theory developed students’ self-efficacy perceptions significantly.

The reflection of effect rate of the programs applied to experimental and control groups on their self-efficacy perceptions has also been revealed by the coefficients of effect-size. The Cohen’s d

effect-size coefficient was found to be higher than of traditional programs. Thus, the qualitative and quantitative data showed a consistent unity.

In literature, the results of the studies done in the effects of Scratch and coding on self-efficacies match up with the results of this study. In a study, Ihmaid [53] Scratch raised students' self-efficacy in learning English words. In their study, Yükseltürk & Altıok [54] and Giannakos, and Hubwieser & Ruf [55] came to the same conclusion (Abo Oda, 2010 adapt. Ihmaid [53]). Calao, Correa, Leon & Robles [66], Choi, Jung & Beak [27], Flannery, Kazakoff, Silveman, Bers, Bonta & Resnick [67], Kukul, Gökçearsan, & Günbatar [68], Kordaki [69], Numanoglu & Keser [70], Taylor, Harlow & Forret [33] and Wang Huang & Hwang [71], studied the effect of coding on problem solving and reasoning, and obtained positive findings.

The second research question did not indicate any significant differences between the permanency tests and the post-tests scores. That is to say, the self-efficacy perception scores in the post-tests remained at the same level with the self-efficacy perception scores in the permanency tests. However, when the post-test scores of the groups were compared, the experimental group had a higher level than the control group. In this case, at the end of the experiment, the experimental group exhibited a higher level of self-efficacy perception, which indicates that the experimental group had more loads with a significant effect of the permanency of self-efficacy perception. In their study, Yüksel & Gündoğdu [72] stated that learning experiences through programs with coding, affected students' attitudes toward the subject, their academic success, and permanency of learning outcomes positively. In their research, Saraçoğlu & Çelik [73], came to the conclusion that applied learning activities affected students' active participation, attitudes, and permanency positively. In their findings, Özyurt & Özyurt [74] stated that in visual programming subjects, students gained experiences related to the business world by writing codes, and their learning became permanent. Karaduman & Emrahoğlu [75] observed in their study that computer-based learning experiences affected the permanency of students' learning outcomes positively. The findings of these studies related to permanency are in parallel with the findings in this research.

While several studies emphasize the positive effects of coding, some studies argue the opposite. Quielle & Bergin [76] in their experimental study, came up with findings revealing that although students enjoyed Scratch, it did not affect their self-efficacy perceptions and success in program applications. In another experimental study by Korkmaz [77], a significant success with Scratch on C++ computer programming language was determined, but a significant difference was not noted in students' self-efficacy perceptions of programming. In a study by Kalelioğlu & Gülbahar [78] with the fifth grade students, it was found out that teaching programming through Scratch did not exhibit any significant differences in their problem solving skills, but, as they stated, the students enjoyed programming. In another study, it was noted that Code or coding teaching did not show any differences between the reflective skills of the experimental and control groups in problem solving. However, as Kalelioğlu [79] emphasized, the qualitative data showed that students exhibited positive behaviors towards programming.

In the third research question, the qualitative data from the students' views were grouped under four main themes; "cognitive output", "affective output", "external output", and "skills". These themes are similar to, as Bandura [43] stated, the sources of self-efficacy such as "experiencing a similar behavior directly", "experiences adapted from others", "verbal persuasion", "physiological and emotional cases". In our research, as can be observed in Table 4, the participants mostly expressed views about the "application" theme. An intensive and active coding application course created a significant difference in favor of the experimental group post-test self-efficacy perception scores, compared to the control group (Table 3). The "Application" theme matched well with "experiencing behavior", one of self-efficacy sources defined by Bandura [43]. In a study by Şeker & Erdoğan [51], it was found out that the post-test evaluations in geometry, taught through GeoGebra software, were significantly different from the pre-test evaluations, and this indicated that the experimental group actively participated in both the computer use and learning processes. Ötürk [80] and Önen &

Muşlu Kaygısız [81] stated in their studies that self-efficacy could be developed positively through theoretical and applied teaching. Similarly, Bahçeçi & Kuru [82] and İlhan Beyaztaş [83] pointed out that practising background knowledge increased self-efficacy positively. Kılıç, Keleş & Uzun [84] found out that practice with laboratory materials significantly increased the participants' self-efficacy beliefs. The connection between application and self-efficacy specified in this research matches up with the other studies in literature. The "Reasoning" theme is one of the themes with the most views. In teaching activities through coding applications it has been observed that students spent a long time thinking of coming up with new ideas, approaching problems with different points of view, and making decisions to design a product. The students came up with ample views about the "Evaluation and Analysis" theme.

The "Skills" theme has three sub-dimensions; social, spatial, and psychomotor skills. The "Skills" theme has the most frequency load among other sub-themes. This is because students mostly did group work. "Democratic decision-making", "self-expression", "communication", and "group work" are the most frequent views under this theme. In the light of this theme, it can be assumed that students' self-efficacy is affected by observing others' success or failure. When the experimental and control groups are compared, the scores by the experimental group are significantly different from the scores by the control group (Table 2). Making democratic decisions, expressing themselves, participating in group work, and establishing effective communication in class are indications of the effect of social skills on students' self-efficacy perceptions. Lewis [85] emphasized similar findings in a study saying that using Scratch affected collaboration and group work positively. Schunk [50] quotes to Zimmerman & Ringle's study) and states that when the participants observe a model for some time confidently, their self-efficacy increases when doing the same work. Parallel to this finding, it was noted in our research that students' positive evaluations of each other and contributions in writing projects increased their self-efficacy perceptions of the subject. Schunk and Zimmerman [86] emphasized that learning by observation by peer-modelling has a direct effect on skills and continuity. This finding matches with the results of permanency-tests in this research. In their study, Çilingir & Artut [87] found out that experimental group students' visual math self-efficacies developed after group work. The findings in the positive effect of group work on self-efficacy show similarities with the findings in the studies mentioned above. In a similar study by Shin, Park & Bae [88] in the "social skills" theme, it was found out that use of Scratch coding increased the level of developing friendship among highly gifted students. In their studies, Theodorou & Kordaki [89]; Taylor, Harlow & Forret [33] stated that Scratch coding and collaborative learning approaches had positive effects on students' social skills. Motivating each other through verbal persuasion, making common decisions, and guiding each other towards certain targets, while writing projects is another way in self-efficacy issue. This finding matches well with Bandura's [43] views about self-efficacy.

Students' views about transferring their outcomes in coding applications are grouped in the "Transfer" theme, the main theme of "External output". The students' views reveal that experiences are transferred into English, Maths, Geometry, and Drawing. The significant differences in the scores by the experimental group, as observed in Tables 2 and 3, match up with the students' views in the main theme of External output. The "Transfer" theme is in parallel with results from different studies in the literature. In a study by Bong [45] on English, Korean, Maths, and learning in school, a correlation was specified among these disciplines, related to self-efficacy beliefs. In their study in the scope of self-efficacy, Flannery, Kazakoff, Silverman, Bers, Bonta & Resnick [67] argue that coding by Scratch will be a guide to interdisciplinary benefits, from Literature to Mathematics. In another study by Sanjanaashree, Anand & Soman [90], it was found that Scratch used in different disciplines had a positive effect on language learning. Gülten & Soytürk [46] pointed out that the development in self-efficacy in geometry raised success in other subjects also. As it can be observed, self-efficacy that is developed in a certain field, can be transferred to other fields.

“Attitude”, “Motivation”, and “Imaginary world” sub-themes are grouped under the main theme of “Affective output”. The “Attitude” theme is one of the four with the most frequencies. It can be assumed that an intensive use of metacognitive skills and positive social interaction is a positive effect on their interest to the subject. These findings match up with the results in the studies by Yükseltürk, Altıok & Üçgül [9]; Dasgupta, Hale, Hernandez, Hill [91]; Wang, Huang & Hwang [71]; Choi, Jung & Beak [27]. Lee argued in a study [92] that Scratch helped students create more meaningful activities when they imagine and do their homework. The content analysis in “Motivation” theme indicated that coding made students more willing and interested in the subject. This finding matches up with the results of the studies in the literature by Calao, Correa, Leon & Robles [66]; Ouahbi, Kaddari, Darhmaoui, Elachqar & Lahmine [93]; Osman, Loke, Zakaria & Downe [94]; Özmen & Altun [95]; Ruf, Mühling & Hubwieser [96]; Çağıltay & Fal [25]; Erol & Kurt [97]; Fengfeng [98]; Diseth [47].

5. Conclusions and Suggestions

The results of the post- and permanency tests given to the experimental group in terms of their self-efficacy perceptions of coding teaching showed a significant difference compared to the control group. This qualitative data obtained from the participants’ views support this result. It was also determined that the effect-size of the program applied to the experimental group was higher than the program for the control group. In this regard, it can be assumed that running Technology and Design courses with coding applications are more effective than traditional programs in developing students’ self-efficacy perceptions.

This study offers a teaching program based on a needs analysis, which will be a reference for the development of the prevailing traditional Technology and Design course teaching in North Cyprus. Technology and Design courses should be restructured with other subjects to respond to the requirements of the domestic market and 21st century skills. The necessity of coding in technology and design courses is crucial to the sustainability of the subject. A program responding to the interest and expectations of the Z group should be designed. Technology and Design courses should interact with Information and Communication courses. Technology and Design course applications and coding teaching should start during primary education. It is advised that this research is referred to in designing programs for Technology and Design courses, and Information and Communication technologies. In order to confirm the effect of coding on self-efficacy, research should be done at advanced levels. Different programs can be designed using different coding programs to support this study done through Scratch, and the results reported

Supplementary Materials: The supplementary materials are available at <http://www.mdpi.com/2071-1050/10/10/3822/s1>.

Author Contributions: C.Ö. and M.T. created the conceptualization; C.Ö. designed the methodology, C.Ö. carried out the investigation; C.Ö. and M.T. made the analyses, validation, and data curation, C.Ö. wrote the paper, and M.T. contributed to reviewing and editing all sections; M.T. supervised the work.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. Needs Analysis Questionnaire in the Technology and Design Course.

	I Need It Too Much	I Really Need It	I Need It But Not Much	I Don't Need It Much	I Can Do Without It
NEEDS IN GENERAL					
Technology and Design Course					
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
NEEDS IN DESIGN					
Technology and Design Course					
13					
14					
15					
16					
17					

Table A1. Cont.

		I Need It Too Much	I Really Need It	I Need It But Not Much	I Don't Need It Much	I Can Do Without It
18	Utilizing the connection between shape and ground among objects					
19	Placing an object on the ground according to the length of distance					
20	Envisaging three dimensions of an object					
21	Drawing an expanded geometric object with depth					
22	Thinking of the vision of geometric figures from different angles					
23	Drawing geometric figures in different angles					
24	Placing the object on coordinated grounds					
25	Presenting animations of designed projects on the computer					
NEEDS IN PROGRAM CODING						
Technology and Design Course						
26	Writing algorithms					
27	Telling the similarity between the algorithmic and mathematical order					
28	Drawing figures with width–height–depth coding by programming					
29	Making design matching in shape and ground					
30	Placing an object by coding on coordinated ground					
31	Drawing an object through different angles by coding					
32	Expanding geometrical objects in mind by coding programs					
33	Drawing expanded geometrical objects with depth by coding					
34	Changing the angle of depth image of figures by coding					
35	Drawing a figure according to its distance by coding a program					
36	Making a picture move by coding					
37	Drawing a three-dimensional animation of imaginary objects by coding a program					
38	Sharing a program designed by coding on the Internet					

Table A2. Table of Contents.

	Targets	Target Steps					
		Knowledge	Comprehension	Application	Analysis	Evaluation	Synthesis
1	Explains the perceptions of Technology and Design	X	X				
2	Explains the principles of design	X	X				
3	Explains the process of basic design	X	X				
4	Explains the importance of line element in design	X	X				
5	Provides samples in using computer in technology and design	X	X				
6	Recognizes computer programs to be used in class	X	X				
7	Explains simple algorithms	X	X				
8	Writes simple algorithms			X			
9	Masters the order of codes in algorithms	X	X				
10	Knows that a single instruction will affect the whole program	X	X				
11	Draws two-dimensional geometric figures using algorithm and lines			X			
12	Turns two-dimensional figures into three-dimensional ones by algorithm and lines			X			X
13	Turns two-dimensional figures into three-dimensional ones by lines			X			X
14	Changes ready-made characters into three-dimensional shapes			X			X
15	Places the characters onto the stage			X			
16	Places the three-dimensional characters onto the coordinated ground			X			
17	Makes out the vision of the characters from different angles	X	X				
18	Moves the characters in different angles			X			
19	Directs the characters into different directions			X			
20	Evaluates own algorithmic setups					X	
21	Decides on the suitable stage for the selected/drawn characters			X			
22	Designs the stage to suit the selected/drawn characters			X			X
23	Draws an object from different angles by coding			X	X		X
24	Evaluates peer's algorithmic setups					X	
25	Draws expanded geometrical figures with depth			X	X		

Table A2. Cont.

	Targets	Target Steps					
		Knowledge	Comprehension	Application	Analysis	Evaluation	Synthesis
26	Changes the direction of depth images of figures			X	X		X
27	Animates self-prepared characters/designs			X			X
28	Uses logical variables			X	X		X
29	Forms close and distant perspectives of objects			X	X		X
30	Animates close/distant/perspectives of objects			X			X
31	Provides products for independent needs			X			X
32	Explains the designed product by coding			X			
33	Shares the project written by coding on the Internet			X			

Table A3. Academic (Yearly) Plan.

Academic Plan Revised by Coding Technology and Design Course					
PERIOD	UNIT 1. CODE TEACHING AND ITS APPLICATION IN TECHNOLOGY AND DESIGN COURSE				
	AIMS/OUTCOME	TOPICS	ACTIVITIES/PROJECTS	TECHNIQUES IN LEARNING/TEACHING METHODS	RESOURCE AND MATERIAL
Week 1	1.1. Explains the Technology and Design concept		Activity 1. Slide show	STEM-based/Project-based learning	Interactive board, laptop, projection
	1.2. Explains design principles	What is Technology and design?	Activity 2. Does drawing activities on electronic board by lines	Problem solving, logical reasoning, critical thinking	Interactive board, laptop, projection
	1.3. Explains basic design process	Innovative thinking	Activity 3. Draws and designs with pencil and paper.	Brain gymnastics	Interactive board, laptop, projection
	1.4. Explains the importance of line in design		Activity 4. Forms shapes, configuration and direction with specific length of lines		Interactive board, laptop, projection

Table A3. Cont.

Academic Plan Revised by Coding Technology and Design Course					
PERIOD	UNIT 1. CODE TEACHING AND ITS APPLICATION IN TECHNOLOGY AND DESIGN COURSE				
	AIMS/OUTCOME	TOPICS	ACTIVITIES/PROJECTS	TECHNIQUES IN LEARNING/TEACHING METHODS	RESOURCE AND MATERIAL
Week 2	2.1. Gives samples of computer use in the Technology and Design Course	Algorithm	Activity 1: Algorithm games are played		Paper and pencil, Interactive board, projection
	2.2. Acquires the computer programs to be used in class		Activity 2: Design algorithm on paper and animating it in Scratch		
	2.3. Explains simple algorithms		Activity 3: Knowing basic codes	Algorithm on paper	
	2.4. Writes simple algorithms		Activity 4: Positioning characters, swapping among dots	Movement and control menus, Pencil menu	Paper and pencil, Interactive board, projection
	2.5. Masters the order of codes in algorithms				
	2.6. Knows that a single instruction will affect the whole program				Notifies the effect of constant/delete instructions
Week 3	3.1. Draws two-dimensional geometric figures in algorithm using lines (width–height).	Algorithm, depth	Drawing architectural designs in the environment	Adding to the character or having it drawn by the character. Sound and vision. Relates it to Mathematics	Paper, pencil, Interactive board, laptop, projection
	3.2. Turns two-dimensional figures into three-dimensional ones (width–height–depth)		Activity 1. Two-dimensional geometric figures of a chosen character are drawn	Drawing characters, using lines	
	3.3. Transfers three-dimensional figures into two-dimensional ones using lines (Reverse procedure)		Activity 2. Observing three-dimensional car design	Adding characters, using lines	Interactive board, laptop, projection
	3.4. Turns ready-made characters into three-dimensional ones		Activity 3. Figures drawn on the electronic board are turned into three-dimensional ones		
	3.5. Places characters onto the stage				
Week 4	4.1. Places the three-dimensional characters onto the coordinated plane	Algorithm, location, direction	Developing a product (Moving mechanical designs)	Relates it to Mathematics, connection between coordinate plane and the stage	Pare-pencil, laptop, interactive board, projection
	4.2. Makes out the images of characters from different angles		Activity 1. Practices moving a character to the left or right as in a watch in Scratch	Turning in degrees or...into...directions	
	4.3. Turns the characters into different angles		Activity 2. The image of a cube from different angles	Turning in degrees or...into directions	
	4.4 Directs characters in different angles		Activity 3. Students draw characters in the image direction they wish	Turning in degrees or...into direction, moving forwards...steps	
	4.5. Evaluates self and peer's algorithm settings				

Table A3. Cont.

Academic Plan Revised by Coding Technology and Design Course					
PERIOD	UNIT 1. CODE TEACHING AND ITS APPLICATION IN TECHNOLOGY AND DESIGN COURSE				
	AIMS/OUTCOME	TOPICS	ACTIVITIES/PROJECTS	TECHNIQUES IN LEARNING/TEACHING METHODS	RESOURCE AND MATERIAL
Week 5	5.1. Decides on the stage to suit the characters	Algorithm, stage, ground	Writing projects on renewable energy	Balance between figure-ground, uses ready-made stage	Interactive board, laptop, projection
	5.2. Designs the stage for the selected/drawn character	Depth of the ground	Activity 1. Stage alteration behind the fixed character	Figure-ground harmony, arranges a genuine stage	
	5.3. Draws a figure from different angles by coding 5.4. Evaluates self and peer's algorithm settings 5.5. Produces animations with self-prepared characters/designs		Activity 2. Skin diver Activity 3. Car on the road and stage motion	Starting with geometrical figures draws objects in different angles, interprets visions	
Week 6	6.1. Draws expanded geometric figures with depth		Designs products to facilitate life for the handicapped		
	6.2. Alters the images of depth in figures 6.3. Produces animations with self-prepared characters/figures		Activity 1. Lights and alarm of an approaching train	Draws the image of the same figure with depth from different angles	Interactive board, laptop, projection
	6.4. Evaluates self and peer's algorithm settings 6.5. Evaluates self and peer's algorithm settings (Evaluation)		Activity 2. "If there is someone, opens the door, if not don't" Coding Activity 3. Writing a code for the priority of an ambulance on the road	Perception, if, variable	
Week 7	7.1. Forms distant-close images of an object	Algorithm, perspective	Designs advertisement and animation projects	Changes dimensions according to distance	Interactive board, laptop, projection
	7.2. Forms distant-close images of a self-formed object		Activity 1. Flying bats on the wall and changing the dimensions of the bat according to the distance of the wall	Algorithm, if, variables	
	7.3. Evaluates self and peer's algorithm settings		Activity 2. Students design own advertisements or animations.		Interactive board, laptop, projection
8. Week	8.1. Designs independent, needed products	Project	I'm designing my own product		
	8.2. Explains the own-designed product by coding		Activity 1. Preparing advertisements or projects by animation		
	8.3. Shares the program designed by coding on the internet		Activity 2. The self-produced products are presented in class and suggestions are noted		Interactive board, laptop, project
	8.4. Evaluates self and peer's algorithm settings				

Annual (Yearly) Academic Teaching Plan of a Technology and Design Course Revised by Coding.

Table A4. Technology and Design Course Self-Efficacy Scale.

		Absolutely I Don't Agree	I Don't Agree	Partly I Agree	I Agree	Absolutely I Agree
COMPREHENSION						
In the Technology and Design course						
1	I can explain line elements, one of design elements					
2	I can explain design principles in project making					
3	I can explain technical drawings of a certain project					
BASIC APPLICATION						
In Technology and Design course						
4	I can take responsibilities in a Project Development Team					
5	I can develop my manual skills					
6	I can fix a product through instructions					
7	I can implement safety conditions in workshops					
8	I can use the apparatus provided in workshops					
9	I can search the internet for design					
ADVANCED APPLICATION						
In Technology and Design course						
10	I can design on the computer					
11	I can replace an object on plane coordinate					
12	I can do an animated presentation of my project					
13	I can design mobile mechanical tools					
14	I can do a product according to its ergonomic use features					
FORMAL ANALYSIS						
In Technology and Design course						
15	I can distinguish the measurements of objects					
16	I can distinguish geometric figures in objects					
17	I can distinguish width-length-depth in figure					
18	I can distinguish the declination of a geometric figure from others					
19	I can distinguish figure-ground					

Table A4. Cont.

		Absolutely I Don't Agree	I Don't Agree	Partly I Agree	I Agree	Absolutely I Agree
FURTHER ANALYSIS						
In Technology and Design course						
20	I can see objects from different angles					
21	I can do needs analysis for a new project					
22	I can analyze problems I face					
23	I can design steps in production					
24	I can relate my learning outcomes to Technology and Design Courses					
25	I can compare different areas to use a product					
SYNTHESIS						
In Technology and Design course,						
26	I can recycle waste material					
27	I can develop projects in saving energy					
28	I can design products for the handicapped					
29	I can invent a new product in workshop					
30	I can write instruction for use					
31	I can apply the project plans in other lessons					
32	I can develop product by myself independently					
EVALUATION						
In technology and Design course,						
33	I can evaluate a product according to its features					
34	I find myself sufficient in design a product					
35	I can do self-criticisms about my own designs					
36	I can renew the product in the light of criticisms					
37	I can criticize my weakness and strengths in T-D courses					
38	I can comment positively on products developed by my friends					

References

- Atasoy, R.; Cemaloğlu, N. Evaluation of Quality Policies on Education in Turkish Education System. *Univ. J. Educ. Res.* **2018**, *6*, 1504–1518. [CrossRef]
- Hameed, I.; Khan, M.B.; Shahab, A.; Hameed, I.; Qadeer, F. Science, technology and innovation through entrepreneurship education in the United Arab Emirates (UAE). *Sustainability* **2016**, *8*, 1280. [CrossRef]
- Visvizi, A.; Lytras, M.D.; Daniela, L. Education, Innovation and the Prospect of Sustainable Growth and Development. In *The Future of Innovation and Technology in Education: Policies and Practices for Teaching and Learning Excellence, Emerald Studies in Higher Education, Innovation and Technology*; Visvizi, A., Lytras, M.D., Daniela, L., Eds.; Emerald Publishing: Bingley, UK, 2018; pp. 297–305.
- Ardıç, E.; Altun, A. Dijital çağın öğreneni. *Uluslararası Sos. Bilgilerde Yeni Yaklaşımlar Derg (IJONASS)* **2017**, *1*, 12–30.
- Göksel, A.; Güneş, G. Kuşaklar arası farklılaşma: X ve Y kuşaklarının örgütsel sessizlik davranışı bağlamında analizi. *Gazi Üniv. İktisadi ve İdari Bilim. Fak. Derg.* **2017**, *19*, 807–828.
- Kalogiannakis, M.; Papadakis, S.T. Proposal for Teaching Scratch Jr Programming Environment in Preservice Kindergarten Teachers. In Proceedings of the 12th Conference of the European Science Education Research Association (ESERA), Dublin, Ireland, 21–25 August 2017; pp. 2095–2105, ISBN 978-1-873769-84-3.
- Tuncer, M. Ağ toplumunun çocukları: Z kuşağının kişilerarası iletişim becerilerinin çok boyutlu analizi. *Atatürk İletişim Derg.* **2016**, *10*, 33–46.
- Kanaki, K.; Kalogiannakis, M. Introducing fundamental object-oriented programming concepts in preschool education within the context of physical science courses. *Educ. Inf. Technol.* **2018**, *23*, 2673–2698. [CrossRef]
- Yükseltürk, E.; Altıok, S.; Üçgül, M. Oyun programlamanın ilköğretim öğrencilerinin problem çözme becerilerine etkileri: Yaz kampı deneyimleri2. In Proceedings of the 4th International Instructional Technologies & Teacher Education Symposium 2016, Elazığ, Turkey, 6–8 October 2016.
- Sayın, Z.; Seferoğlu, S.S. Yeni Bir 21. Yüzyıl becerisi olarak kodlama eğitimi ve kodlamanın eğitim politikalarına etkisi. In *XVIII Akademik Bilişim Konferansı-AB*; A Dnan Menderes Üniversitesi: Aydın, Turkey, 2016; pp. 3–5.
- Demirci, A.; Aykurt, G. Teknoloji ve tasarım dersinde öğretmen ve öğrencilerin karşılaştığı sorunlar. *Milli Eğitim Derg.* **2014**, *44*, 170–192.
- Erden, O.; Keskin, N. Point of views of technology and design teachers regarding computer assisted teaching and practicability of computer assisted teaching in the lessons of the technology and design. *J. Int. Soc. Res.* **2015**, *8*, 642–650. [CrossRef]
- Kaya, Ö. Temel Eğitimde Uygulanan Teknoloji ve Tasarım Dersi Öğretim Programı ve 7. Sınıf Öğretim Programı Uygulamalarının Öğretmen Görüşleriyle Değerlendirilmesi. Master's Thesis, Gazi Üniversitesi/Eğitim Bilimleri Enstitüsü, Ankara, Turkey, 2008.
- Safkan, Ş.İ. Öğretmenlerin Teknoloji ve Tasarım Dersindeki Sorunlara Yaklaşımları. Master's Thesis, Yakın Doğu Üniversitesi/Eğitim Bilimleri Enstitüsü, Lefkoşa, Cyprus, 2009.
- Serdar, C.; Ceylan, E.; Dalkıran, M. Teknoloji ve tasarım öğretmenlerinin özel alan yeterlik düzeylerinin belirlenmesi. *Int. J. Act. Learn.* **2016**, *1*, 38–54.
- Tuğcuoğlu, Y. Ortaöğretimde teknoloji ve tasarım dersi için yeni bir model oluşturulması. *Sanat Tasar Derg.* **2013**, *1*, 31–34.
- Yalçın, Z. İlköğretim II. Kademe Teknoloji ve Tasarım Dersine Öğretmen ve Öğrenci Yaklaşımları. (Yayınlanmamış Yüksek Lisans Tezi); Selçuk Üniversitesi/Sosyal Bilimler Enstitüsü: Selçuk, Turkey, 2007.
- Birinci Konur, K.; Yıldırım, N. Examples and implications of prospective science teacher' on project based learning practices. *Int. J. Eurasia Soc. Sci.* **2017**, *8*, 2043–2063.
- Kırbağ Zengin, F.; Yucasu, Ş. The effects of project based learning method on the preservice science teachers' environmental literacy. *Int. J. Eurasia Soc. Sci.* **2017**, *8*. Available online: http://www.ijoes.com/Makaleler/645441062_36-%20DCXVIII-DCXXXIII%20-%20%20fikriye%20k%C4%B1rba%C4%9F.pdf (accessed on 5 August 2018).
- Uysal, M.P. Yazılım mühendisliğinde deneysel bir çalışma:proje tabanlı ve proje destekli yöntemlerin ürün ve akademik başarıya etkileri. *Pamukkale Üniv. Mühendis. Bilim. Derg.* **2018**, *24*, 226–237.
- Calder, N. Using scratch: An integrated problem-solving approach to mathematical thinking. *Aust. Prim. Math. Classr.* **2010**, *15*, 9–14.

22. Ford, L.J. *Scratch Programming for Teens 1st*; Course Technology Press: Boston, MA, USA, 2008.
23. Maloney, J.; Burd, L.; Kafai, Y.; Rusk, N.; Silverman, B.; Resnick, M. Scratch: A sneak preview. In Proceedings of the Second International Conference on Creating 2004, Connecting and Collaborating through Computing, Kyoto, Japan, 30 January 2004; pp. 104–409.
24. Shin, S.; Park, P. A study on the effect affecting problem solving ability of primary students through the scratch programming. *Adv. Sci. Technol. Lett.* **2014**, *59*, 117–120.
25. Çağıltay, E.N.; Fal, M. *Scratch ile Programlamayı Öğreniyorum*; ODTÜ Yayıncılık: Ankara, Turkey, 2015.
26. Fee, S.B.; Minkley, A.M.H.; Lombardi, T.E. *New Directions for Computing Education Embedding Computing across Disciplines*; Springer: Berlin, Germany, 2017.
27. Choi, B.; Jung, J.; Beak, Y. In what way can technology enhance student learning? A preliminary study of technology supported learning in mathematics. In *Society for Information Technology & Teacher Education International Conference 2013 Mar 25 (pp. 3–9)*; Chesapeake, V.A., Ed.; Association for the Advancement of Computing in Education (AACE): Jacksonville, FL, USA, 2013.
28. Serpe, R.L. Gamification through Algebraic Coding. Master's Thesis, College at Brockport, State University of New York, Education and Human Development, New York, NY, USA, 2017.
29. Portelance, D.J.; Strawhacker, A.L.; Bers, M.U. Constructing the scratcher programming language in the early childhood classroom. *Int. J. Technol. Des. Educ.* **2016**, *26*, 489–504. [[CrossRef](#)]
30. Mitchel, R. ScratchJr: Computer programming in early childhood education as a pathway to academic readiness and success (NSF DRL-1118682). In *Poster Presented at DR K-12 PI Meeting*; Massachusetts Institute of Technology: Cambridge, MA, USA, 2011.
31. Dasgupta, S. From surveys to collaborative art: Enabling children to program with online data. In Proceedings of the 12th International Conference on Interaction Design and Children, New York, NY, USA, 24–27 June 2013; pp. 28–35.
32. Roque, R.; Rusk, N.; Blanton, A. Youth roles and leadership in an online creative community. In Proceedings of the Computer Supported Collaborative Learning Conference, Madison, WI, USA, 15–19 June 2013; International Society of the Learning Sciences: Cambridge, UK, 2013.
33. Taylor, M.; Harlow, A.; Forret, M. Using a computer programming environment and an interactive whiteboard to investigate some mathematical thinking. *Procedia Soc. Behav. Sci.* **2010**, *8*, 561–570. [[CrossRef](#)]
34. Weiss, D. The effect of combining 1:1 computing, interactive core curriculum and digital teaching platform on learning math: The case of a charter school in New York City. In *Handbook on Digital Learning for K-12 Schools (s. 341–354)*; Springer International Publishing: Berlin, Germany, 2016.
35. European Schoolnet. *Computing Our Future Computer Programming and Coding-Priorities, School Curricula and Initiatives Across Europe*; Durando, M., Ed.; European Schoolnet: Brussels, Belgium, 2014.
36. Demir, A.Ö. Çocuğunuza kod yazmayı öğretin. Harvard Business Review Türkiye. 2017. Available online: <https://hbrturkiye.com/blog/cocugunuza-kod-yazmayi-ogretin> (accessed on 28 March 2017).
37. Bandura, A. *Self-Efficacy: The Exercise of Control*; Freeman: New York, NY, USA, 1997.
38. Arseven, A. Öz yeterlilik: Bir kavram analizi. *J. Turk. Stud.* **2016**, *11*, 63–80. [[CrossRef](#)]
39. Ormord, J.E. *Educational Psychology: Developing Learners (5 b)*; Pearson Prentice Hall: Upper Saddle River, NJ, USA, 2006.
40. Ansari, M.; ve Khan, S.A. Self-Efficacy as a predictor of life satisfaction among undergraduate students. *Int. J. Indian Psychol.* **2015**, *2*, 5–11.
41. Bandura, A. Self-Efficacy. In *Encyclopedia of Human Behavior*; Ramachaudran, V.S., Ed.; Academic Press: New York, NY, USA, 1994; Volume 4, pp. 71–81.
42. Bandura, A. Self-efficacy: Toward a unifying theory of behavioral change. *Psychol. Rev.* **1977**, *84*, 191–215. [[CrossRef](#)] [[PubMed](#)]
43. Bandura, A. *Exercise of Personal and Collective Efficacy in Changing Societies. Self-Efficacy in Changing Societies*; Cambridge University Press: Cambridge, UK, 1995.
44. Özdemir, C.; Erdoğan, T. Sınıf öğretmenleri adaylarının ilköğretilere ve yazma öğretilerine ilişkin öz yeterlilik inançlarının belirlenmesi. *Abant İzzet Baysal Üniv. Eğit. Fak. Derg.* **2017**, *17*, 314–331.
45. Bong, M. Academic Motivation in Self-Efficacy, Task Value, Achievement Goal Orientations, and Attributional Beliefs. *J. Educ. Res.* **2010**, *97*, 287–297. [[CrossRef](#)]
46. Çağırğan Gülten, D.; Soytürk, İ. İlköğretim 6. Sınıf Öğrencilerinin Geometri Öz Yeterliliklerinin Akademik Başarı Not Ortalamaları ile İlişkisi. *Mehmet Akif Ersoy Üniv. Eğit. Fak. Derg.* **2013**, *1*, 55–70.

47. Diseth, A. Self-efficacy, goal orientations and learning strategies as mediators between preceding and subsequent academic achievement. *Learn. Individ. Differ.* **2011**, *21*, 191–195. [[CrossRef](#)]
48. Motlagh, S.E.; Amrai, K.; Yazdani, M.J.; Abderahim, H.A.; ve Souri, H. The relationship between self-efficacy and academic achievement in high school students. *Procedia Soc. Behav. Sci.* **2011**, *15*, 765–768. [[CrossRef](#)]
49. Phan, P.H. The Development of English and Mathematics Self-Efficacy: A Latent Growth Curve Analysis. *J. Educ. Res.* **2012**, *105*, 196–209. [[CrossRef](#)]
50. Schunk, D.H. Self-efficacy, motivation and performance. *J. Appl. Sport Psychol.* **1995**, *7*, 112–137. [[CrossRef](#)]
51. Şeker, B.H.; Erdoğan, A. GoeGebra Yazılımı ile Geometri Öğretiminin Geometri Ders Başarısına ve Geometri Öz-Yeterliğine Etkisi. *Uluslar. Toplum Araşt. Derg.* **2017**, *7*, 1–16.
52. Zimmerman, B.J. Self-Efficacy and educational development. In *Self-Efficacy in Changing Societies*; Cambridge University Press: Cambridge, NY, USA, 1999; pp. 202–231.
53. Ihmaid, M.K. The Effectiveness of Using Scratch Application in Developing Sixth Graders' English Vocabulary, Its Retention, and Self-Efficacy. Master's Thesis, The Islamic University/Faculty of Education, Islamabad, Pakistan, 2017.
54. Yükseltürk, E.; Altiok, S. An investigation of the effects of programming with scratch on the preservice IT teacher's self-efficacy perceptions and attitudes towards computer programming. *Br. J. Educ. Technol.* **2016**, *48*. [[CrossRef](#)]
55. Giannakos, M.N.; Hubwieser, P.; Ruf, A. Is self-efficacy in programming decreasing with the level of programming skills? In Proceedings of the 7th Workshop in Primary and Secondary Computing Education, Hamburg, Germany, 8–9 November 2012; pp. 16–21.
56. Maviş, İ.E. Avrupa Birliği Uyum Sürecinde Teknoloji ve Tasarım Eğitimi. Master's Thesis, Gazi Üniversitesi/Fen Bilimleri Enstitüsü, Ankara, Turkey, 2010.
57. Creswell, J.W. *Research Design Qualitative, Quantitative, and Mixed Methods Approaches (4 b.)*; SAGE: Thousand Oaks, CA, USA, 2014.
58. Büyüköztürk, Ş.; Çakmak, E.K.; Akgün, Ö.E.; Karadeniz, Ş.; Demirel, F. *Bilimsel Araştırma Yöntemleri*; Pegem Akademi: Ankara, Turkey, 2016.
59. Engel, K.S.; Moosbrugger, H.; Müller, H. Evaluating the Fit of Structural Equation Models: Test of Significance and Descriptive Goodness-of-Fit Measures. *Methods Psychol. Res. Online* **2003**, *8*, 23–74.
60. Tabachnick, B.G.; Fidell, L.S. *Using Multivariate Statistics*, 6th ed.; Pearson Education Limited: London, UK, 2013.
61. Tok, Ş. Mikro öğretim yönteminin öğretmen adaylarının özyeterlik inançlarına ve mesleğe yöneliktutumlarına etkisi. *Milli Eğit.* **2016**, *212*, 5–25.
62. Eymur, G.; Çetin, P.S. Argümantasyon tabanlı sorgulayıcı araştırma yönteminin öğretmen adaylarının fen öğretimi öz yeterlik inancına etkisi. *Erzincan Üniv. Eğit. Fak. Derg.* **2017**, *19*, 36–50.
63. Kaya, Z. Koro eğitiminde yapılandırmacı yaklaşımın tutum, öz yeterlik algısı ve akademik başarıya etkisi. *İnönü Üniv. Eğit. Bilim. Enst. Derg.* **2014**, *1*, 52–62.
64. Çoban, A.; Sanalan, A. Fen bilgisi öğretimi dersinde özgün deney tasarım sürecinin öğretmen adayının öz yeterlik algısına etkisi, *Erzincan Üniv. Eğit. Fak. Derg.* **2002**, *2*, 1–10.
65. Şensoy, Ö.; Aydoğdu, M. Araştırma soruşturma tabanlı öğrenme yaklaşımının fen bilgisi öğretmen adaylarının fen öğretimine yönelik öz-yeterlik inanç düzeylerinin gelişimine etkisi. *GÜ Gazi Eğit. Fak. Derg.* **2008**, *28*, 69–93.
66. Calao, L.A.; Correa, H.E.; Leon, J.M.; Robles, G. Developing mathematical thinking with scratch an experiment with 6th grade students. In *Design for Teaching and Learning in A Networked World*; Springer International Publishing: New York, NY, USA, 2015; pp. 17–27.
67. Flannery, L.P.; Silverman, B.; Kazakoff, E.R.; Bers, M.U.; Bonta, P.; Resnick, M. Designing ScratchJr: Support for early childhood learning through computer programming. In Proceedings of the 12th International Conference on Interaction Design and Children, New York, NY, USA, 24–27 June 2013; Interaction Design Foundation: New York, NY, USA, 2013.
68. Kukul, V.; Gökçearslan, Ş.; Günbatır, M.S. Computer Programming Self-Efficacy Scale (CPSES) for Secondary School Students: Development, Validation and Reliability. *Educ. Technol. Theory Pract.* **2017**, *7*, 158–179.
69. Kordaki, M. Diverse categories of programming learning activities could be performed within scratch. *Procedia Soc. Behav. Sci.* **2012**, 1162–1166. [[CrossRef](#)]

70. Numanoğlu, M.; Keser, H. Programlama öğretiminde robot kullanımı-Mbor örneği. *Bartın Univ. J. Fac. Educ.* **2017**, *6*, 497–515.
71. Wang, H.Y.; Huang, I.; Hwang, G.J. Effects of an integrated scratch and project-based learning approach on the learning achievements of gifted students in computer courses. In Proceedings of the International Conference on Advanced Applied Informatics (s. 382–387), Kitakyushu, Japan, 31 August–4 September 2014.
72. Yüksel, S.; Gündoğdu, K. Scratch Öğretiminde Ayrılıp Birleşme Tekniği Kullanımının Derse Yönelik Tutuma Akademik Başarıya ve Kalıcılığa Etkisi. *Ege Eğit. Derg.* **2018**, *19*, 245–261. [[CrossRef](#)]
73. Saracaloğlu, A.S.; Çelik, B. Web tasarımı ve programlama dersi öğretiminde proje tabanlı öğrenme yaklaşımının öğrenci görüşlerine göre değerlendirilmesi. *Kast. Educ. J.* **2018**, *26*, 323–333.
74. Özyurt, H.; Özyurt, Ö. Görsel programlama dersinde proje tabanlı öğrenme deneyimine ilişkin öğrenci görüşlerinin incelenmesi. *Turkish J. Comput. Math. Educ.* **2017**, *8*, 244–260.
75. Karaduman, B.; Emrahoğlu, N. Maddenin tanecikli yapısı. ünitesinin öğretiminde, bilgisayar destekli ve bilgisayar temelli öğretim yöntemlerinin, akademik başarı ve kalıcılığa etkisi. *Kast. Eğit. Derg.* **2011**, *19*, 925–938.
76. Quille, K.; Bergin, S. Does scratch improve self-efficacy and performance when learning to program in C#? An empirical study. In Proceedings of the International Conference on Engaging Pedagogy, Dublin, Ireland, 13–14 December 2016.
77. Korkmaz, Ö. The effects of scratch-based game activities on students' attitudes, self-efficacy and academic achievement. *Int. J. Mod. Educ. Comput. Sci.* **2016**, *1*, 16–23. [[CrossRef](#)]
78. Kalelioğlu, F.; Gülbahar, Y. The effects of teaching programming via scratch on problem solving skills: A discussion from learners' perspective. *Inform. Educ.* **2014**, *13*, 33–50.
79. Kalelioğlu, F. A new way of teaching programming skills to K-12 students: Code.org. *Comput. Hum. Behav.* **2015**, *52*, 200–210. [[CrossRef](#)]
80. Ötürk, Ö.F. Bilim-Teknoloji-Toplum Hakkındaki Görüşler ile Fen ve Teknoloji Okuryazarlığı Öz Yeterlik Algısı Üzerine Bir Çalışma. *Necatibey Eğit. Fak. Elektron. Fen ve Mat. Eğit. Derg.* **2016**, *10*, 1–31.
81. Önen, F.; Muşlu Kaygısız, G. Fen Bilgisi Öğretmen Adaylarının 6–8 dönemler Arasındaki Fen Öğretimine Yönelik Öz Yeterlik İnançları ve Bu İnanca İlişkin Görüşleri. *Kuram ve Uygul. Eğit. Bilim.* **2013**, *13*, 2435–2453.
82. Bahçeci, D.; Kuru, M. Portfolyo değerlendirmesinin üniversite öğrencilerinin öz yeterlik algısı ve yaşam becerileri üzerine etkisi. *Ali Evran Üniv. Kırşehir Eğit. Fak. Derg. (KEFAD)* **2008**, *9*, 97–111.
83. İlhan Beyaztaş, D. Sınıf eğitimi öğrencilerinin ingilizce öğretimine yönelik öz-yeterlik inançları. *Erzincan Üniv. Eğit. Fak. Derg.* **2017**, *20*, 265–280.
84. Kılıç, D.; Keleş, Ö.; Uzun, N. Fen Bilimleri Öğretmenlerinin Laboratuvar Kullanımına Yönelik Öz Yeterlik İnançları: Laboratuvar Uygulamaları Programının Etkisi. *Erzincan Üniv. Eğit. Fak. Derg.* **2015**, *17*, 218–236. [[CrossRef](#)]
85. Lewis, M.C. Is pair programming more effective than other forms of collaboration for young students? *Comput. Sci. Educ.* **2011**, *21*, 105–134. [[CrossRef](#)]
86. Schunk, D.H.; Zimmerman, B.J. Social Origins of Self-Regulatory Competence. *Educ. Psychol.* **1997**, *32*, 195–208. [[CrossRef](#)]
87. Çilingir, E.; Artut, P.D. Gerçekçi Matematik Eğitimi Yaklaşımının İlkokul Öğrencilerinin Başarılarına, Görsel Matematik Okuryazarlığı Öz yeterlik Algılarına ve Problem Çözme Tutumlarına Etkisi. *Turk. J. Comput. Math. Educ.* **2016**, *7*, 578–600.
88. Shin, S.; Park, P.; Bae, Y. The effects of an information-technology gifted program on friendship using scratch programming clutter. *Int. J. Comput. Commun. Eng.* **2013**, *2*, 246–249. [[CrossRef](#)]
89. Theodorou, C.; Kordaki, M. Super Mario: A collaborative game for the learning of variables in programming. *Res. Gate* **2010**, *1*, 111–118.
90. Sanjanaashree, P.; Anand Kumar, M.; Soman, K.P. Language Learning for Visual and Auditory Learners Using Scratch Toolkit. In Proceedings of the 2014 Computer Communication and Informatics International Conference on (s. 1-5), Coimbatore, India, 3–5 January 2014.
91. Dasgupta, S.; Hale, W.; Hernandez, A.M.; Hill, B.M. Remixing as a Pathway to Computational Thinking. A. f. Machinery (Dü.). In Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing, San Francisco, CA, USA, 27 May 2016.
92. Lee, Y.J. Scratch: Multimedia Programming Environment for Young Gifted Learners. *Gift. Child Today* **2011**, *34*, 26–31. [[CrossRef](#)]

93. Ouahbi, I.; Kaddari, F.; Darhmaoui, H.; Elachqar, A.; ve Lahmine, S. Learning Basic Programming concepts by creating games with scratch programming environment. *Procedia Soc. Behav. Sci.* **2015**, *191*, 1479–1482. [[CrossRef](#)]
94. Osman, M.A.; Loke, S.P.; Zakaria, M.N.; Downe, A.G. Secondary students' perfectionism and their response to different programming learning tools. In Proceedings of the 2012 IEEE Colloquium on Humanities, Science and Engineering (CHUSER), Kota Kinabalu, Malaysia, 3–4 December 2012.
95. Özmen, B.; Altun, A. Undergraduate Students' Experiences in Programming: Difficulties and Obstacles. *Turk. Online J. Qual. Inq.* **2014**, *5*, 9–27. [[CrossRef](#)]
96. Ruf, A.; Mühling, A.; Hubwieser, P. Scratch vs. Karel: Impact on learning outcomes and motivation. In Proceedings of the 9th Workshop in Primary and Secondary Computing Education (s. 50–59), Berlin, Germany, 5–7 November 2014.
97. Erol, O.; Kurt, A.A. The effects of teaching programming with scratch on pre-service information technology teachers' motivation and achievement. *Comput. Hum. Behav.* **2017**, *77*, 11–18. [[CrossRef](#)]
98. Fengfeng, K. An implement of design-based learning through creating educational computer games: A case study on mathematics learning during design and computing. *Comput. Educ.* **2014**, *73*, 26–39.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).