Learners’ perspectives on block-based programming environments: Code.org vs. Scratch

Johannes Krugel
Technical University of Munich
School of Education
Munich, Germany
krugel@tum.de

Alexander Ruf
Technical University of Munich
School of Education
Munich, Germany
alexander.ruf@tum.de

ABSTRACT
We report from a comparative study regarding two popular block-based programming environments for the introduction of algorithmic concepts, namely Code.org and Scratch. A quasi-experiment was conducted with five classes (grade 7) from two secondary schools (n = 122). To investigate the effects of both learning environments, the students were divided into two groups.

Following a mixed-methods research approach, we use quantitative and qualitative methods to gain a comprehensive understanding of learner’s perspectives and skills. We measure the learners’ perceived self-regulation and intrinsic motivation. The learners of Code.org show a higher intrinsic motivation compared to the Scratch group. Qualitative analyses of answers to open-ended questions reveal positive and negative aspects of the learning environments. With Code.org, the learners in particular like the fun and the challenges, while for Scratch, they like the feeling of actually programming something and the very free environment.

CCS CONCEPTS
• Applied computing → Computer-assisted instruction; Interactive learning environments; • Social and professional topics → Computational thinking; K-12 education.

KEYWORDS
Interactive learning environments, Block-based programming, Algorithmic thinking, Computational Thinking, Motivation

1 INTRODUCTION
Computational Thinking is considered to be an important ability every student should learn in school [12]. Block-based programming languages are frequently used for teaching nowadays [1, 10]. We focus on web-based tools because they can easily be used in a flipped classroom setting [3] or when learning at scale [4]. This is of special interest in the current COVID-19 (“Coronavirus”) pandemic where students have to learn at home with their own devices.

2 TEACHING INTERVENTIONS
We target learners without any previous knowledge of algorithms and compare teaching with Code.org and Scratch. The learning objectives cover basic algorithmic concepts usually included relatively at the beginning of courses on computational thinking [5]. Due to the COVID-19 pandemic, the students learned at home with detailed instructions prepared by the teachers (both authors of this paper). They received a 1-page working sheet each week with reading instructions, links to explanation videos, and exercises. The format of the tasks varied to include different categories [2]. We did explicitly not make the learning sequence of both groups exactly the same because this would contradict the inherent differences of the learning environments. The learning objectives and the difficulty was, however, very similar. The first experimental group learned on Code.org with the Accelerated course1. The learners program a figure on the screen to, e.g., follow a path in a maze or to draw geometric shapes. The course is based on game-based learning and microlearning, consisting of a series of small games with several levels. The course contains short videos introducing the concepts (in English with German subtitles); we created 13 additional short explanatory videos very similar to the videos of the Scratch group. The second group learned with the web-based Scratch 3.02. Figures can be programmed to move on the screen, perform other actions and react to events. Similar to Code.org, the figures also draw geometric shapes (using the Pen extension). The teaching material is based on a continuing education course for teachers3 and includes 18 explanatory videos of an experienced teacher.

---

1 https://studio.code.org/s/20-hour/
2 https://scratch.mit.edu/projects/editor/
3 https://lehrerfortbildung-bw.de/u_matnatech/informatik/gym/bp2016/fb1/
3 METHODOLOGY
We conducted a quasi-experiment with 5 classes of two secondary schools in grade 7, where Computer Science is a compulsory subject and has 1 lesson (45 minutes) per week. There were 122 students of age 12–14 with a balanced gender distribution, see Table 1. The intervention spans 9 weeks and we collect data at three time points $T_0$, $T_1$, and $T_2$. At the time of writing this paper we had completed the first 5 weeks and data collections at $T_0$ and $T_1$. The research design is visualized in Figure 1. Other similar experiments often focus on quantitative methods [8, 9]. We use a mixed-methods approach to compensate the weaknesses and combine the strengths of quantitative and qualitative methods. We use a pre-test, a post-test, and free text feedback. Additionally we employ the “Academic Self-Regulation Questionnaire” (SRQ-A) which can be used to determine the self-determination index (SDI) [7]. The intrinsic motivation is assessed with the KIM instrument [11] with 4 subscales, see Table 1.

4 RESULTS REGARDING THE MOTIVATION
At the beginning of our study $T_0$, both experimental groups have the same average SDI of 2.21 (see Table 1). At $T_1$, we received 79 responses regarding the intrinsic motivation. In this paper, we analyze the data with a relatively simple comparison based on the arithmetic mean of the Likert items (like [8]). The students of Code.org exhibit a higher intrinsic motivation (with consistently higher average value on the individual subscales Interest/Enjoyment, Perceived Competence and Perceived Choice, and a lower value for Pressure), This was surprising for us because we expected a higher intrinsic motivation for Scratch which offers more freedom to the learners. Our mixed-methods approach proved to be helpful here because the qualitative analyses yields further insights.

5 QUALITATIVE RESULTS
The answers to the qualitative survey were analyzed using an inductive qualitative content analysis based on the methodology of Kyngäs [6]. Each answer could be assigned to multiple categories. We performed several iterations over the data to adapt and refine while maintaining a coding manual. This resulted in 18 positive and 15 negative aspects.

For Code.org, the mentioned positive aspects of the learners frequently fall into the categories Fun and Challenges (e.g. “I had to think but at the same time it was fun.”), as well as Easiness and Usability (e.g. “It is very easy to understand and you only have to pull blocks together”). The most prominent negative aspects is, that some students regard the exercises as too easy.

For Scratch, the by far most frequent category of positive aspects is Programming, followed by Trying out (e.g. “I liked to do my own programming and just try out how that works.”). Also aspects related to the freedom of the learning environment are frequently mentioned (e.g. “With Scratch you can make the character do almost anything.”). As negative aspects, the learners mentioned the difficulty, especially at the beginning of the sequence and some technical issues. This perceived difficulty is probably also due to the fact that Scratch offers a rich set of functionality which might be overwhelming for some beginners; we assume that this is also an explanation for the relatively lower intrinsic motivation.

6 CONCLUSION
A limitation is the number of participants and the non-random assignment to the experimental groups. Additionally, only a fraction of the learners participated in data collection $T_1$. However, at $T_2$ we were able to collect data from the majority of students. The analysis of this data will provide a more detailed view and allows the comparison of the whole teacher sequences, including the achieved learning outcomes.

REFERENCES

Table 1: Quantitative results per learning environment ($T_0$ and $T_1$ denote the respective number of participants).

<table>
<thead>
<tr>
<th>Group</th>
<th>n</th>
<th>Female / Male</th>
<th>$T_0$</th>
<th>SDI</th>
<th>Interest/Enjoyment</th>
<th>Perceived Competence</th>
<th>Perceived Choice</th>
<th>Pressure/Tension</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code.org</td>
<td>47</td>
<td>24/23</td>
<td>43</td>
<td>2.21</td>
<td>3.21</td>
<td>3.21</td>
<td>0.73</td>
<td></td>
</tr>
<tr>
<td>Scratch</td>
<td>75</td>
<td>38/37</td>
<td>67</td>
<td>2.21</td>
<td>3.02</td>
<td>2.80</td>
<td>2.93</td>
<td>0.85</td>
</tr>
<tr>
<td>Total</td>
<td>122</td>
<td>62/60</td>
<td>108</td>
<td>2.21</td>
<td>3.11</td>
<td>2.89</td>
<td>3.06</td>
<td>0.79</td>
</tr>
</tbody>
</table>