JOURNAL OF RESEARCH IN SCIENCE, MATHEMATICS AND TECHNOLOGY EDUCATION
DOI: 10.31756/jrsmte.619SI

Easy Coding in Biology: Pilot Workshop Design and Experiences from Block-Based Programming with <colette/> in Secondary Education

Eva Schmidthaler

Johannes Kepler University, Austria

Rebecca Stäter

Goethe University Frankfurt, Germany

Martin Capay

Constantine the Philosopher University in Nitra, Slovakia

Matthias Ludwig

Goethe University Frankfurt, Germany

Zsolt Lavicza

Johannes Kepler University, Austria

Abstract: Block-based programming (BBP) is a way to teach Computer Science (CS) concepts and Computational Thinking (CT) skills to K-12 students. Nowadays, educational applications (learning apps) to teach these CS concepts in STEM subjects are made available, but detailed studies on which and how these can be used successfully in Biology classes are still missing. This mixed-method study aims to close this gap and presents six elaborated exercises of the workshop "*Easy Coding in Biology*" employing two task families (TF), *Building Cubes* and *Drone AR*, of the new learning app <colette/>. The app's TF described in this study utilizes BBP and augmented reality (AR) function to describe several topics in secondary education, including the subject of Biology. After presenting the workshop and task design, preliminary results of the successful use of the app and the tasks are presented. The pilot workshop was carried out with 51 participants at a Slovakian grammar school. Following the implementation in class, 34 students filled in a questionnaire consisting of open-ended and closed-ended questions. Qualitative data shows on the one hand positive feedback on the workshop concept and on the other hand AR malfunction. Quantitative findings indicate that the introduced BBP task family, *Building Cubes* has the potential to successfully convey CS and CT concepts in the subject Biology. Moreover, the tasks encouraged female participants to collaborate as they found these tasks interesting, motivating, and fun. This study lays the groundwork for future research, which will include testing and evaluating the biological tasks in conjunction with BBP.

Keywords: *Block-based Programming, Computer Science in Biology, Computational Thinking, Mobile Coding, Mobile Educational Application, K-12 education*

Introduction

Nowadays, many European STEM educators are faced with challenges and obligations regarding the new curricula to impart their teaching material in a creative way and to use the latest digital media and technologies in their lessons to teach their students the 21st-century-skills (e.g., communication, collaboration, critical thinking) in addition to the technical ones (National Education Association, 2015; Bocconi et al., 2022). Much-discussed competencies and skills in secondary school in recent years are Computational Thinking (CT) and related Computer Science (CS) concepts (e.g., coding, programming) (Polart et al., 2021; Hsu et al., 2018; Bocconi et al., 2022). In the context of this study, six CT competencies, based on Bocconi (2016) can be identified: "*abstraction, algorithmic thinking, automation, decomposition*, *debugging,* and *generalization*". By incorporating these competencies into this study, the researchers aim to explore their application and effectiveness within the context of teaching Life Science content by using block-based programming (BBP) by science teachers during their lessons. In the subsequent sections, the significance of CT and CS concepts in science education is explored, as well as the importance of CS (such as programming) in Science. Additionally, a comprehensive explanation of the app and the <colette/> project is provided.

Computational Thinking and Computer Science Concepts in Biological and Science Education

When someone thinks of CT and CS, Biology or Science might not be the first thing that comes to mind, although several studies have been conducted on how CS and CT concepts can be taught and implemented in biology education in form of student and teacher workshops and courses (e.g., Lockwood & Mooney, 2017; Blum & Cortina, 2007; Qin, 2009; Goldberg et al., 2012). In a teacher workshop emphasizing CT and the many possible ways of CS conducted in 2007, theoretical CT and CS concepts and ideas were combined with practical (biological) examples (e.g., using food). Based on the findings (Blum & Cortina, 2007), all workshop sessions were considered beneficial by teachers, with the practical sessions emerging as particularly popular. Qin (2009) implemented a different type of teacher workshop, which aimed to educate Biology students in Bioinformatics through the integration of Computer Science (CS). To illustrate the connection between CT and Bioinformatics, a specific instance involved the presentation of a biological database, such as the abstraction of protein structure.

Findings showed, the biology students decided to pair up during labs to ensure individual learning, and by stressing CT concepts within the lecture, the participants stated that they observed a common thinking pattern. Furthermore, they had an overall positive feeling regarding the workshop and improving their CS competencies. Some participants even decided to take additional CS lectures after the Bioinformatics course (Qin, 2009).

A student study, conducted in 2014, aimed to teach CT concepts (e.g., algorithmic thinking, abstraction) in Life Science. Their course required a knowledge of CS concepts (such as programming) and afterward concentrated on developing participants' CT skills (Rubinstein and Chor, 2014). Within the study, the participants had to solve biological real-life issues using the text-based coding language Python, concentrating firstly, on CT concepts and not on teaching programming, and secondly, stressing practical employment in Life Science. Rubinstein and Chor (2014) found out that the participants' perceptions of CS had changed after the workshop: CS is not about the actual physical machine but about problem-solving and other CT concepts.

Another research conducted with students (Goldberg et al., 2012) regarding an interdisciplinary lecture aimed to integrate CS-based activities into Art, Biology, and Mathematics, allowing students to discuss CT concepts using CS. Goldberg et al. (2012) first provided information on how CT can benefit the curriculum, and then how CS and CT concepts can be incorporated into K-12 student workshops: a topic such as human health education could be combined with data analysis in the subject of Biology. According to the teachers, the students' understanding of CS increased, as has the students' engagement after being introduced to CT and CS concepts (Goldberg et al., 2012). Another study conducted with students presented the qualitative experiences of five Biology undergraduates taking a CS lecture. CS and CT concepts were taught by utilizing programs and analyzing data, introducing them to various programming tasks. Afterward, they were interviewed, aiming to better understand the participants' thought processes. It was found that the presented visualization tasks in this course relate to CT tasks and provided the participants with useful information to accomplish the programming exercises (Yuen & Robbins, 2015).

More recent research showed that students made Science and computer-assisted learning gains after studying computational modeling as part of a Science unit (Arastoopour Irgens, 2020).

Programming in Science and Biology

One way of teaching CS skills is making use of educational applications (learning apps) which are using BBP. According to current research, block-based programming has a lot of advantages and benefits in comparison to text-based programming concerning the learning process: When programming commands are predetermined, novices can comprehend and memorize them more easily. As a result, the codes are less likely to be subject to (syntax) errors, saving teachers' time and helping them in correcting the students' work (Yamashita et. al., 2017; Shih, 2017; Xu et. al., 2019). Results from a study conducted with high school students (Weintrop & Wilensky, 2017) indicated that all the participants, divided into two groups (either block-based or text-based coding), improved their pre-test results compared to post-test results after having attended a coding workshop. Nonetheless, BBP resulted in better learning outcomes and higher interest in future computing lectures among the participants (Weintrop & Wilensky, 2017).

There have already been several K12-student courses in recent years on how to combine text-based programming (using Python) with biological topics. The latest findings from the perspective of Biology are, for example, "*Programming in Biology*" (University of Zurich, 2018) or "*Biology Meets Programming: Bioinformatics for Beginners*" (University of California San Diego, 2023). Regarding BBP, Gupta et al. (2017), introduced "BioBlocks", a web-based visual development tool, based on *Google's Blockly* and *Scratch* (Blockly, 2022; Blockly Games, 2022; Scratch, 2022), describing and programming biological experimental protocols (Gupta et al., 2017). In a recent study, Researchers found that block-based programming was successfully integrated into STEM school subjects, such as Biology (Anderson et al, 2021). Participating students were able to solve the tasks without prior knowledge of coding. However, because of integrating programming into subjects such as Biology lessons, participating educators were forced to divide their time between "teaching and scaffolding the subject and programming" (Anderson et al, 2021, S.133). In addition to the successful implementation of block-based programming in Biology lessons or with interdisciplinary approaches, there were also issues regarding this topic in other studies: five teachers found BBP in Science class to be an important skill for K-12 students in a 2020 study. However, they raised concerns about teaching this topic without assistance since they were unable to design authentic and interdisciplinary simulation coding activities (Vasconcelos & Kim, 2020). Block-based programming has the potential to assist K-12 students in learning CS concepts and gaining CT competencies, without any requirement for them to type text-based codes and can be successfully used in all subjects. While there have been studies regarding BBP and text-based programming (TBP) up until now (e.g., Weintrop & Wilensky, 2017), studies on BBP in Biology classes are still lacking. Therefore, the new learning app <colette/> is introduced, with which BBP can be easily integrated into Biology classes.

This article aims to discuss how to employ BP with the app to connect biological topics in the subject of Biology with CS concepts to promote CT skills in secondary education.

The <colette/> Project

The tasks described in this paper are based on a newly developed app, which focuses on teaching CT skills called <colette/> (**Co**mputational Thinking **L**earning **E**nvironmen**t** for **T**eachers in **E**urope). In this project, consisting of six partners from five different countries (Germany, Austria, the Netherlands, Slovakia, France), an app (available for free for Android and iOS devices), as well as a web portal were created. <colette/> is making use of the bring-your-own-device-approach (e.g., smartphone or tablet) facilitating the integration of coding into the classroom, as no new expensive equipment must be bought by schools, meaning that students can use their own device of choice (Läufer et. al, 2021; Zender et al., 2014).

A study on the perceptions of teachers and students regarding the implementation of mobile augmented reality (mAR) educational applications (learning apps) in biological education showed that costs are a decisive factor in whether the apps are used or not (Schmidthaler et al., 2023a). The <colette/> app is focusing mainly on deliberately challenging CT skills (Milicic et. al, 2021; Läufer et. al, 2021;), and furthermore, provides material for teachers with a wide range of previous coding experience (from no experience at all to very experienced) as well as additional teacher training. The project is divided into four pillars that interact with each other, (1) the mobile application for students as freeware, (2) the web portal for educators as freeware, (3) teacher training, and (4) a handbook for additional app instructions and implementation suggestions for teachers. The app offers mAR visualization feature which helps students to see what they have coded in real-life and real-time (Läufer et. al, 2021; Schmidthaler et al., 2023b). The web portal is the authoring tool for educators, for example, Biology teachers. On the one hand, the web portal gives access to all the task families, on the other hand, it guides the process of creating learning paths, which can be created by the teachers or other educators themselves (Milicic et. al, 2021; Läufer et. al, 2021; Colette-project, 2021).

Introduction of the Task Families *Building Cubes* and *Drone AR*

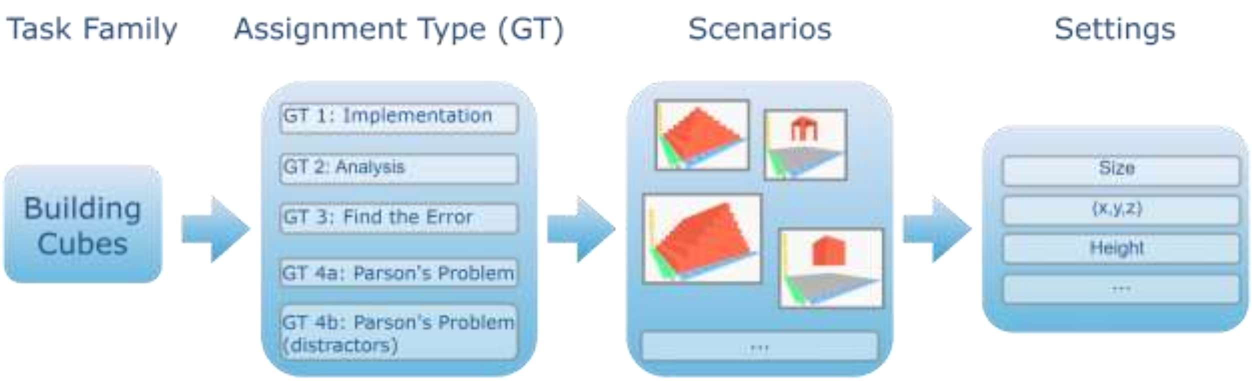
Having seen what the project has to offer, a detailed introduction of the app's task families (TF) is given, starting with the core concept of TF: A *Task Family* can be defined as a "task format", "type of tasks" or "a set of tasks", which have a common core but might have different values for settings, and a TF is a blueprint to generate a multitude of different tasks varying only in specific settings (Stäter et al., 2023a; Stäter et al., 2023b). One of the TF is called *Building Cubes*. All the derived *Building Cubes* tasks have in common that they are in a 3D-grid and the student places (red) cubes on the grid or 3D map to build a specific object (e.g., geometric form, tree, mushroom) using BBP, based on *Google Blockly* code (Blockly, 2022; Blockly Games, 2022). The concept of Generic Tasks (GT) (Milicic et al.,

2020) provides access to several assignment types for coding exercises, which are all enforcing different CT skills. The Generic Tasks for the Task Family Building Cubes are e.g., Implementation (students have to code the desired structure) or find the error (students receive an erroneous code and have to correct it) (*Figure 1*).

Using the mAR view integrated into the app, students can see their coded structure on their smartphone using its camera and the AR marker. Students can walk around the structure and observe what they should have created as a ghost view (Stäter et al., 2023a; Stäter et al., 2023b). This way, students can explore and visualize the structure, understand difficult concepts (Saidin et al., 2015; Ke and Hsu, 2015), and detect possible errors enforcing the skill of debugging and algorithmic thinking (AT), as it was already shown in other pilot studies (Cheli, et al. 2018).

Figure 1

The TF and GT of the <colette/> application: An example of the TF "Building Cubes", its GTs, and Scenarios (e.g., table) is shown.



The second TF *Drone AR* is also based on a 3D-grid but instead of building structures as in *Building Cubes* a flying route of a virtual drone must be coded by the students in such a way that the drone takes pictures of the structure in specific positions. The mAR view allows students to follow the drone executing their code step-by-step, hence, enforcing the CT skills of debugging and AT. The individual TFs, their GT, and the corresponding CT concepts were created "backwards" based on Ziegenfuss' and LeMire's "Backward Design" (Ziegenfuss & LeMire, 2019). Hence, having the learning goal (CT skill) in mind and finding a student exercise that will enforce this skill: TF in combination with GT (e.g., TF Building Cubes and GT Implementation). Combining TF and GT has the potential to provide access to teaching material for all the individual CT skills, and GT can enhance CT skills such as abstraction, AT, automation, decomposition, debugging, and generalization (Milicic et al., 2020).

Furthermore, each of the TFs of <colette/> concentrates on individual CT skills. Following the backward design, the following CT competencies will be enhanced by the individual Task Families as seen in *Table 1*.

Table 1

The CT skills (Bocconi et al., 2016) of the two portal-ready TF of the <colette/> application

Task Family (TF)	What Should Students Do (WSSD)	Main CT Skill	Secondary CT Skills	Additional Skills
Building cubes	Implementation, Debugging	AT	Decomposition, Abstraction, Generalization	Spatial Skills, Coordinate System,
Drone AR	Implementation, Debugging	AT	Decomposition, Abstraction, Generalization	Spatial Skills, Coordinate System

Besides CT skills, more general skills are also enforced, such as coding principles and skills like making block-based code more efficient, testing the code, and possibly creating general solutions. Furthermore, besides working with rather simple code blocks (e.g., setting a cube to a given location), *Building Cubes* also covers more advanced code structures like loops and conditional statements. In addition, the concepts involved in the *Building Cubes* and *Drone AR* tasks are a coordinate system with three dimensions (3D) spatial orientation and visualization as part of spatial skills (McGee, 1979). As the mAR view creates an immersed view of the created structure it builds a bridge between the 2D view of a 3D object and trains, thus, students to interpret a given 2D view as a 3D object (Boon., 2003).

Methodology

In this section, the authors outline the methodology, which details the systematic approach, data sources, and analysis techniques utilized to address the research objectives. Furthermore, it provides a clear understanding of the process undertaken to obtain reliable findings, contributing to the advancement of knowledge in Science education. Additionally, two BBP TF of the <colette/> app are presented. Currently, the implemented tasks are mainly targeting the subjects Mathematics (M) and Computer Science (CS) and were already successfully tested with European students (e.g., *Building Cubes*) and in several student workshops in the summer until winter 2022 during the <colette/> project (Schmidthaler et al., 2023b). As a result, other STEM subjects, such as Biology (B), have not yet been covered, and Drone AR has not been tested yet, emphasizing the importance of this design-based research.

Research Aim and Design

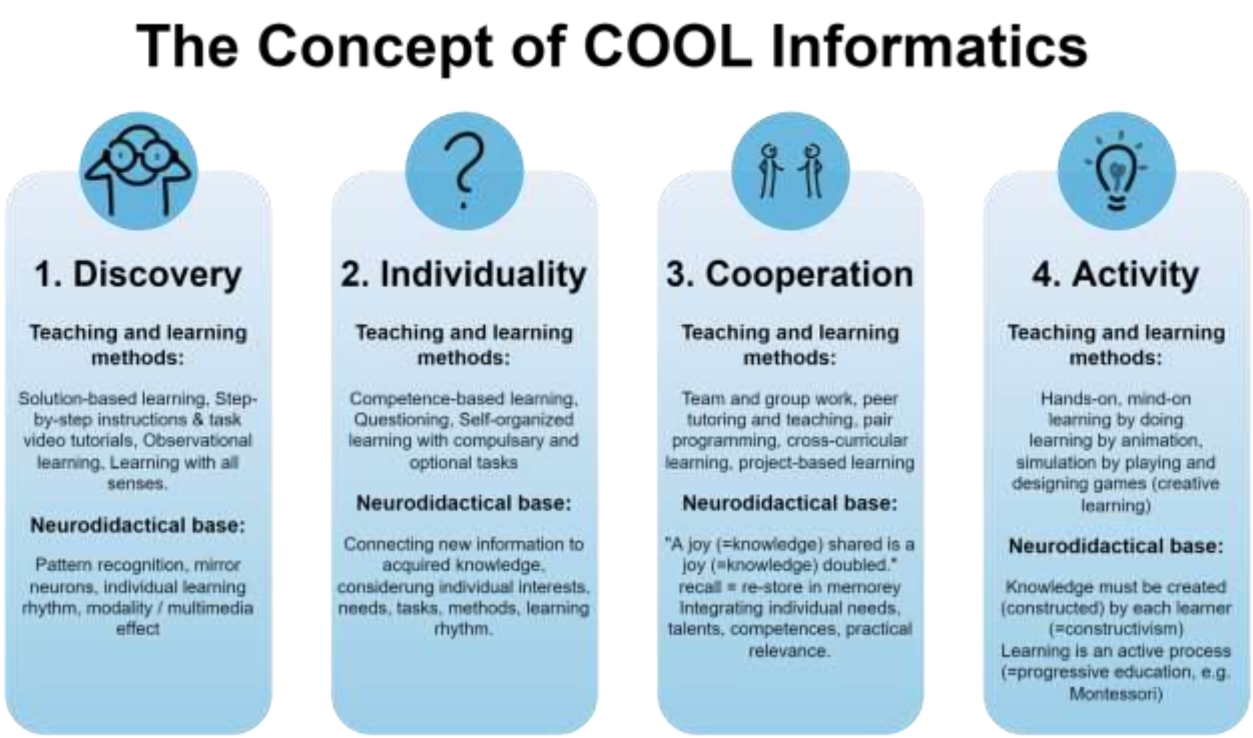
To address the gap in integrating CS and CT concepts into biology classes, a mixed-method research approach was employed. The aim was to integrate the two Task Families, *Building Cubes* and *Drone AR*, of the mAR app into Science education, specifically biology classes. The research design involved the creation of a workshop that combined Science and CS concepts and was facilitated by biology teachers. The workshop included newly developed teaching materials and incorporated the Task Families of the <colette/> app. The methodology consists of a two-step process: In the first step, biology teachers conduct regular classes, delivering the relevant biology content to the students. These classes serve as the foundation for subsequent coding activities, ensuring that students have a solid understanding of the biological principles that would be reinforced through coding exercises. The content covered in these classes aligns with the curriculum guidelines and learning objectives of the biology course. Following the completion of the biology content instruction, the second step involves the introduction of coding tasks within the context of Biology. Students are provided with the BBP app <colette/>, which was specifically designed to facilitate their engagement with CS and CT concepts in a biological setting. The coding tasks are carefully designed to align with the biology content covered in the preceding classes, enabling students to apply their knowledge in a practical coding context. The evaluation of this integration primarily focuses on the CT tasks within the application. The aim was to assess students' understanding and application of CT concepts in the context of biology, with a specific emphasis on the effectiveness of the CS and CT integration facilitated by biology teachers. The evaluation employed both quantitative and qualitative student surveys.

To demonstrate the cross-curricular advantages of the educational application for CT, coding workshops (C-W) were developed and tested with secondary school students (fifth to eighth grade), with a particular focus on engaging young girls. The workshops utilized the mAR app's Task Families and incorporated BBP for visualization and promoting CT. Additional hands-on materials, such as task sheets and paths, were designed to enhance the educational teaching approach based on the "COOL Informatics" method (Sabitzer, 2013a). The pilot workshop with Slovakian grammar school students, titled "Easy Coding in Biology", served as a testbed for the coding materials with the app.

Core Concept "COOL Informatics" for the "Coding in Biology" Workshop

Figure 2

The four pillars of the core concept of COOL Informatics (Sabitzer, 2013a)



The concept of “engagement, discovery, individuality, cooperation, and activity” can be used by STEM educators to design innovative, engaging, brain-friendly, and supportive teaching methods (Sabitzer, 2013a; Kennedy, 2020). Various guidelines, ideas, and examples are provided for preparing teaching units (planning and developing materials), and for developing and testing special CS competencies (e.g., problem-solving, algorithmic thinking, programming). By incorporating these principles into the design of materials and teaching sequences, it is possible to strengthen learners' motivation, improve their understanding of the content, and increase their success in learning, according to previous research on COOL Informatics, especially CS lessons and within mobile learning (Sabitzer & Pasterk, 2012). *Figure 2* illustrates the four main pillars of COOL Informatics, showing the core teaching and learning methodology, as well as the neuro-didactical foundations, such as pattern recognition, connecting new information to previous knowledge, joy, and constructivism (Sabitzer & Antonitsch, 2012; Sabitzer, 2013a; Sabitzer & Pasterk, 2013b; Piaget, 1971). A central aspect of the concept, along with individuality, cooperation, and collaboration with peers, is discovery learning (Sabitzer, 2013a).

In the designed <colette/> workshops “*Easy Coding in Biology*”, each student can use exploratory learning to solve a wide range of solution-based tasks in their own learning rhythm (discovery) using step-by-step instructions. In addition, with the <colette/> tasks, individual needs, and specific interests of students (e.g., specific biological topics) can be addressed based on competency-based learning (e.g., abstraction, pattern recognition, problem-solving). It is by partner work, along with plenary discussions after the cross-curricular workshop, that cooperation and collaboration within the class are strengthened, togetherness (joy) is conveyed, and knowledge (collaboration) is deepened during the lesson. According to the COOL Informatics concept, the fourth pillar, “activity,” is the most influential: By experimenting with the <colette/> app, the participating students are actively learning basic IT concepts and CT skills, by connecting new knowledge with previous knowledge (Sabitzer, 2013a; 2013b).

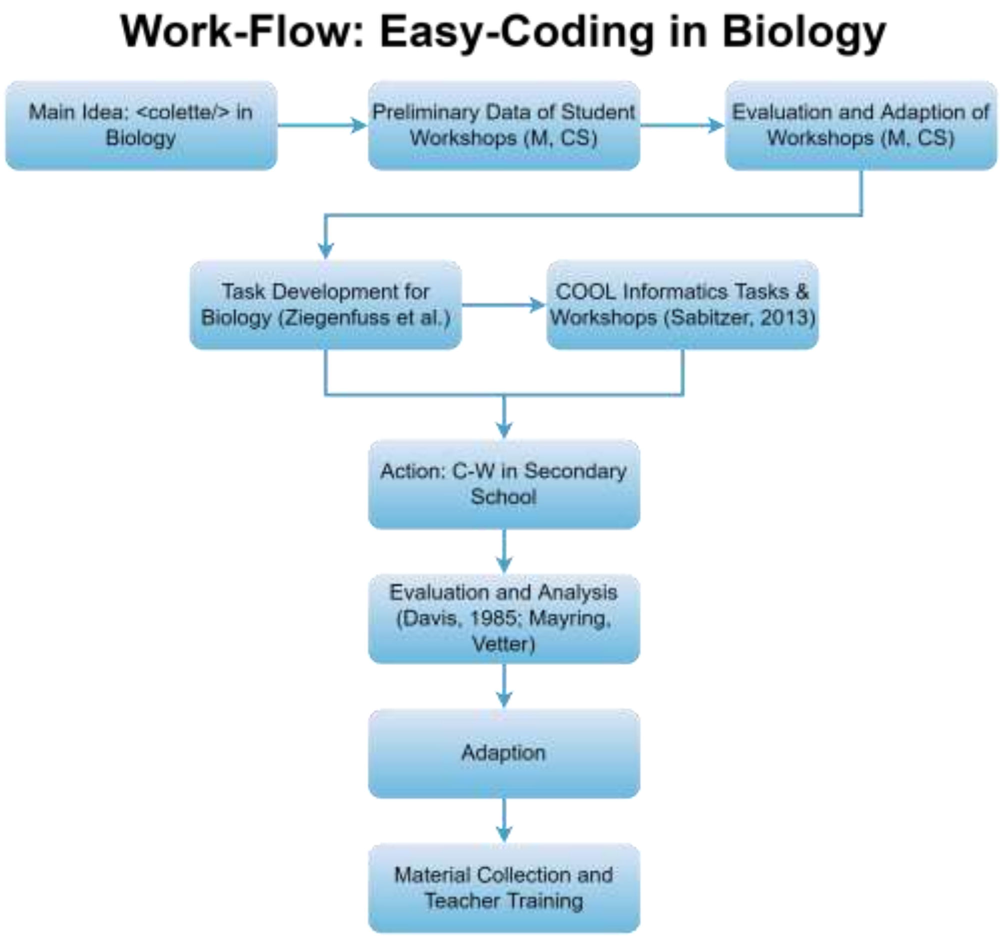
Workflow of Task and Workshop Development: “Easy Coding in Biology”

For the coding workshop used in the mixed-method study was developed based on preliminary findings from several student workshops in 2022 (Schmidthaler et al., 2023b), three Biology curricula (BMBWF, 2018; Ministry of Education, 2022, Prokop et. al, 2007; Ministry of Education, Science, Research and Sports of the Slovak Republic, 2023a), and further research on benefits regarding BBP (Yamashita et. al., 2017; Shih, 2017; Xu et. al., 2019), mAR (e.g., visualization, spatial skills, and conceptual understanding) (Saidin et al., 2015; Ke and Hsu, 2015; Schmidthaler et al., 2023a; Eldokhny, 2021; Marini et al., 2022), and discovery learning (Sabitzer, 2013a).

During the following stages of this design-based-research, the research team will evaluate and adapt the tasks and materials developed for <colette/> (e.g., task sheets), as well as the workshop processes. Once the evaluation is complete, all developed paths, tasks, and materials will be made available as open educational resources (OER) both on the <colette/>-project page (Colette-project, 2022) and the JKU material collection (Materialbörse, 2021). A workflow diagram for the development of workshops and tasks is shown in *Figure 3*, along with the planned evaluation process.

Figure 3

The workflow of the task and coding workshop (C-W) development: "Easy Coding in Biology" is shown.



Sampling of Experimental Workshop

In February 2023, 51 (33=female, 17=male, 1=no gender) Slovakian grammar school students, aged 16-19 years, attended the "Easy Coding in Biology" pilot workshop, which lasted 45 minutes. Only the coding tasks were tested and evaluated within this workshop. Some of the participants worked alone (27 students) the others worked in groups of two (24 students). At the end of the coding lecture, 34 (18 females, 15 males, 1 no gender) filled in the evaluation questionnaire. The participating students tested at least one out of four different tasks from the <colette/> task format *Building Cubes*, as shown in *Table 2*. The students independently chose the tasks they wanted to complete. All participants had prior knowledge of coding, at least "a little bit" (58.82%) (*Figure 3*). 5.88% (two male students) were "very experienced" and 35.29% were "pretty experienced" (e.g., C#, Python, C++, PH, Scratch, and raspiOS). In addition, in a gender-specific comparison, male participants showed more previous experience than female participants.

Figure 4

The prior experience in coding in participating Slovakian grammar school students who attended the pilot workshop "Easy Coding in Biology"

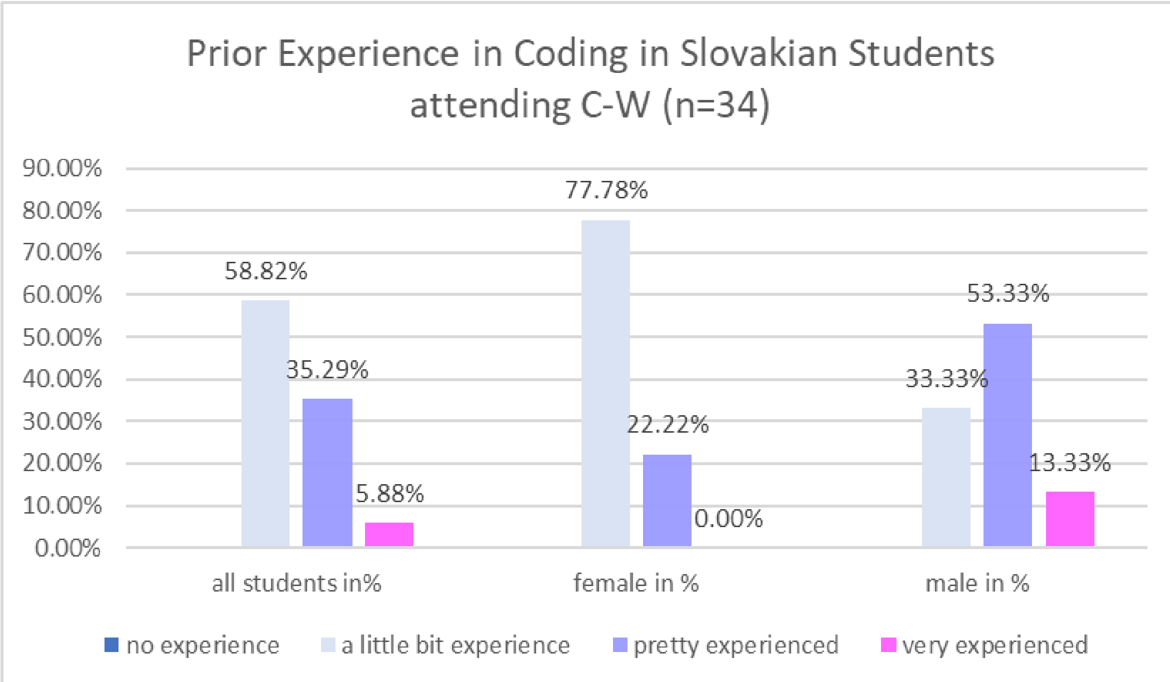


Table 2

Participating students (f) solved the "Easy Coding in Biology" tasks. The numbers of participants are shown according to the four different tasks (in%) (n=34)

Name of the Biological Task:	f	f in %
L-Shape (L-Shaped Plant Bed)	23	67.65%
Pyramid (Fruit, Strawberry, Wine grapes)	8	23.53%
Egg (Square)	16	47.06%
Pyramid (Conifer, Tree)	23	67.65%

Experimental Design, Data Collection, and Processing

Data collection and processing took place in February 2023: In regard to the experimental workshop design, the participants were observed based on the participant observation methodology (Musante & DeWalt, 2010). The instructors documented the students' progress and their final task solutions during and after the observation. Screenshots of the final codes and specific errors on the students' devices, and pictures during the workshop, made by the instructor, were collected, and analyzed afterward. Additional analysis with the *Building Cube* log data (recorded success rates, number of trials, and final code approaches) from the lecture was done, as well as an evaluation of the questionnaire at the end of the coding part of the workshop (duration=15 minutes). The evaluation survey

(LimeSurvey) consists of questions regarding demographic data (age, gender), which tasks have been processed (Multiple Choice MC), and regarding workshop evaluation: thirteen closed-ended (Likert-Scale), and three open-ended based on the Technology Acceptance Model (Davis, 1985). The questions are listed in the *Appendix*. This questionnaire and observation methodology has already been successfully used by the authors in previous <colette/> or "COOL Informatics"-based workshops in 2022 and 2023 (Schmidthaler et al., 2023b; Schmidthaler et al., 2023c). Quantitative data was collected and processed using descriptive statistics (Vetter, 2017). Qualitative data (e.g., students' suggestions for improvement, and errors in the app) were presented as a summary content analysis based on Mayring's qualitative content analysis (Mayring, 2010).

During the coding part of the workshop, the grammar school students were supervised by one instructor. At the beginning of the <colette/> workshop (duration=45 min), the students were divided into groups or pairs. After a short explanation regarding the app, the AR marker, and the AR environment, the participants started experimenting with the <colette/> application and AR feature, trying to solve the four tasks, using tablet-PCs.

Course and Task Design

In the following results section, the six tasks developed for the "Easy Coding in Biology" workshop across four different school levels are presented. These tasks encompass exercises, workflow, and planned time management, offering a comprehensive overview of the workshop structure in Science education. The workshops are carefully planned and consist of two main parts: the coding part and the biology content, aligning with the respective curricula. In the subsequent sections, the planned workshops will be described in greater detail, highlighting their specific components and objectives.

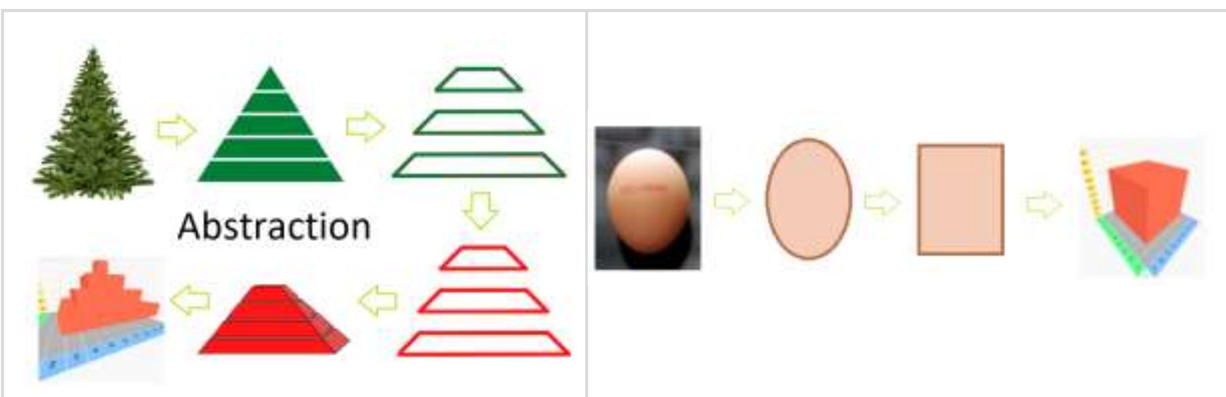
Course and Task Design: "Easy Coding in Biology"

In Austria, the subject of Biology is taught from the fifth up to the eighth grade dealing with the topics of *humans and health*, *animals*, *mushrooms, and plants* as well as *ecology and environment* (BMBWF, 2018; Ministry of Education, 2022). In all four grades, these three main themes serve to structure the lesson content. Additionally, the curriculum also contains suggestions for interdisciplinary work (cross-references to existing topics and impulses for the development of interdisciplinary cooperation). In the following, teaching examples and workshop ideas are discussed to cover these core topics in all four school levels with the <colette/> app, according to the three curricula, shown in *Table 3.* Overall, the app does not claim to cover all biological topics; it is to be used exemplarily at the school level. Therefore, in the following, lesson examples and sequences are presented thematically age-appropriate, regarding the Austrian, German, and Slovakian curricula for secondary biological education (BMBWF, 2018; Ministry of Education, 2022, Prokop et. al, 2007, Ministry of Education, Science, Research and Sports of the Slovak Republic, 2023a).

Integrating the Task Family *Building Cubes* and *Drone AR* into secondary schools, biological topics or living beings (e.g., coniferous tree) are represented in a very simplified way with many individual cubes, for example, an egg is presented as a cube, a coniferous tree shown as a pyramid (Abstraction). In the end, many biological topics (e.g., plants, tree species, fruits, nutrition) can be simplified (e.g., strawberry into a pyramid), and generalized (generalization; "*Program an algorithm for all trees or fruits*"). When solving the task, the students have to work in a solution-oriented manner and must break down the task ("*Create a code for a strawberry*") into individual sub-problems/CT patterns (decomposition), to recognize such CT patterns (pattern recognition, e.g., strawberry consists of similar individual layers of cubes), and to program a CT pattern (automation). At any time, students can experiment with the app and its tasks, communicate, and discuss with their peers (experimentation and tinkering), view or analyze the different approaches, and improve or shorten their codes (using loops; algorithmic thinking), or correct their codes or incorporate other approaches (working in pairs, plenum discussion) into their own (debugging). The mAR feature helps to promote spatial skills in coordinate systems (e.g., viewing their solution of an l-shaped plant bed in 3D) as well, as shown in previous studies (Eldokhny, 2021; Marini et al., 2022).

Figure 5

Abstraction examples of a conifer and an egg in the <colette/> app task format "Building Cubes". The tree (left) and the egg (right) are displayed more and more simplified.



Task and Workshop Development: "Easy Coding in Biology" in 5th Grade and in 6th Grade

The workshop "Easy Coding in Biology: Fifth and Sixth Grade" has been planned and is designed to be conducted as follows: In the first part of the workshop, the biological content is discussed: in the fifth grade, according to the Biology curricula, "ecosystems" are explained, using the example of forests. Selected native representatives from animals, plants, and mushrooms in their construction and function are developed, whereby a basis for age-appropriate understanding of kinship relations is to be laid. Furthermore, positive and negative consequences of human activity in the forest are addressed and questioned, and environmental, natural, and biotope protection can be demonstrated. In the sixth grade the topic "forest ecosystem" is discussed thematically in biology. This includes the students knowing what the difference is between a conifer tree and a leaf tree and being able to name and identify the most important native trees and shrubs. The same applies to spore creatures such as mushrooms. The students should name and describe selected native representatives. The major topic "forest ecosystem", is discussed alongside its most important inhabitants, such as living organisms (plants, animals, fungi), and tree species. In addition, the topic of wood (ecology), nature conservation, sustainability, and the dangers of the (native) forest are discussed.

Table 3

This table shows an overview of biological and CT competencies (Bocconi et al., 2016) of the <colette/> tasks "Easy Coding in Biology" based on Austrian, Slovakian, and German curricula, and findings from the <colette/> project (BMBWF, 2018; Ministry of Education, 2022; Prokop et. al, 2007; Colette-Project, 2021, Ministry of Education, Science, Research and Sports of the Slovak Republic, 2023a)

Curriculum Biology	Humans and Health; Communities of Living Beings (Animals, Mushrooms, and Plants); Ecology and the Environment
Curriculum Digital Education	Coding; Programming; Algorithms
Biological Competencies and Basic Concepts	Compartmentalization; Control and Regulation; Material and Energy Conversion
Digital Education Competencies Computational Thinking (CT) Skills	Problem-Solving; Experimentation and Tinkering; Spatial skills, Coordinate System; Pattern Recognition; Abstraction; Algorithmic Thinking (AT); Automation; Decomposition; Debugging; Generalization *Bocconi et al., 2016*
Implemented Tasks for Building Cubes	Create an Algorithm for (1) an Egg, (2) a L-Shaped Plant Bed, (3) a Fruit (Wine grape, strawberry), (4) a Pyramid-Shaped Plant Bed, (5) a Conifer
Implemented Task for Drone AR	Create an Algorithm for a Drone Flight over a Toadstool
Age & Grade (School)	10-14y (5-8)

The prerequisite for the workshop for fifth and sixth grades is that the mentioned topics are discussed and that the most important characteristics of conifers, leaf trees, and mushrooms are repeated at the beginning of the second part of the workshop (coding). As students repeat the visual indicators, they can better understand and implement the simplified dart positions from the blocks. The basic neuro-didactical idea behind this concept (Sabitzer, 2013a) is that by linking something already known and learned, the students learn something new, in this case, block-based programming, way easier. They can understand and memorize coding better because they already have pre-knowledge, for example, about trees. In addition, the AR function makes it easier for children to understand where to put the blocks so that they can better orient themselves in the coordinate system. As a result, students add new knowledge (CS concepts) to existing knowledge (biological concepts), and link and visualize them, in a fun and creative way (Sabitzer, 2013a).

During the second part of the workshop, the app is installed, and the BBP language is introduced if needed. Afterward, the coding tasks are assessed. During the workshop, the participating students can choose voluntarily to work individually or in pairs. Once every child has the app on their device, the Science teacher starts the actual workshop. During the coding part of the workshop, the participants must solve three different tasks:

(1) Task 1: *Getting to know the App: L-Shaped Plant Bed (Building Cubes)*

(2) Task 2: *Programming a Conifer (Building Cubes)*

(3) Task 3: *Programming a Drone Flight over a Mushroom/Toadstool (Drone AR)* (*Figure 6*).

Figure 6

Pictures of the workshop "Coding in Biology" tasks: L-shaped plant bed in real-life made out of unit-cubes in the Task Family Building Cubes (Task 1) (left), a conifer in real-life (Blume, M., 2019) and through AR Visualization within the <colette/> application (Task 2) (center); and a toadstool (Amanita muscaria) in real-life (Gimmel, B., 2004) and through AR visualization within the <colette/> application (Task 3) (right)

In the first task *"Getting to know the App: Create an L-shaped plant bed with the blocks of the Task Family Building Cubes*", the students familiarize themselves with the functionalities of the app (e.g., blocks, AR feature). The tasks were planned in such a way that they increase in difficulty: The coding part of the workshop starts with a very simple task (Task 1). In the second task: "Program the code of a conifer using Building Cubes", the children must program the shape of a tree (e.g., fir, spruce). In the last task, the participants coded the shape of a mushroom. In all three tasks, the Science teacher provides (at least) three tips or hints: the colors of the coordinates (blue: X, green: Y, and yellow: Z) are indicated, the starting block is revealed in the event of problems, and the possibility of changing the coordinates of the blocks is indicated. In addition, the students are aware that they should not put two or more blocks in the same place. Furthermore, the possibility of shortening the code and thus incorporating loops is noted. At the end of the workshop, an online questionnaire regarding the participating students' perception of the <colette/> app based on the *Technology Acceptance Model* (Davis, 1985) will be distributed to evaluate the tasks, the app, and the workshop itself.

In total (preparation, installation process, repetition of biological concepts (100-120min), coding workshop (50min), questionnaire (15min)), it is suggested that at least a duration of 120-180 minutes is required to successfully execute the whole workshop, depending on the students' prior knowledge of biology and previous coding experience. Optionally, the apps can be installed, and the characteristics of the trees discussed in advance, only two tasks could be solved, or the questionnaire could be given as homework.

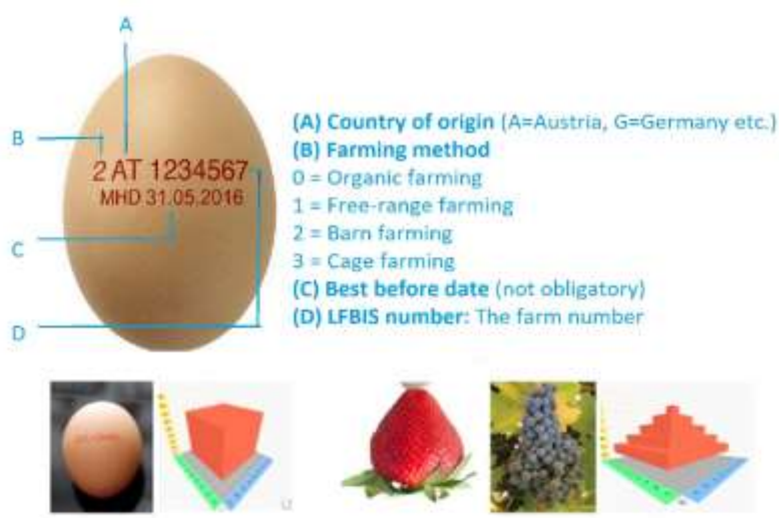
Task and Workshop Development: "Easy Coding in Biology" in 7th Grade and 8th Grade

The workshop titled "Easy Coding in Biology: Fifth and Sixth Grade" has been meticulously organized and is intended to be executed in the following manner: During the first 100-120 minutes of the workshop, the biological content is discussed by the Science teacher: In these year groups, important topics such as sustainability and animal welfare, as well as domestic farm animals and their products (e.g. fur, eggs, meat, milk) are discussed. In addition, the topic of nutrition is addressed. In addition, what constitutes a "healthy" and "balanced" diet is discussed. Furthermore, indigenous useful plants (such as herbs) are explained.

In the seventh grade, a large thematic part of the curricula of the subject Biology is the world of "farm animals" and "farm and useful plants". Students will learn what crops and livestock are, and what resources or products can be made or produced by them (wine, eggs, fruit, vegetables). Participants will learn how to treat (domestic) farm animals in a species-appropriate way (animal welfare) and what the different animal husbandry options are (Egg code, as a sign of origin within the EU; as shown in *Figure 7*).

Figure 7

The EU code for eggs, along with its description (above), is provided. Pictures of a real-life egg (left), an egg made from (red) unit cubes in the Task Family Building Cubes (Task 1) (center left), a *strawberry in real-life (center; Wilson, 2010), a grapevine in real-life (center right; Beaufort, 2007), and fruits made from unit-cubes in the Task Family Building Cubes (Task 2)*



In the Egg Code task, a wide variety of examples of numbers are presented in the biology lesson and their connection with the animal husbandry forms (0-3) is explained. The useful plants and domestic animals should have been discussed at the beginning of the workshop: the students know what these terms mean and are able to list some examples. The topic "Nutrition" also deals with the ingredients (vitamins, minerals, carbohydrates, fats, proteins, fiber). After the unit, the students know which ingredients fruits and vegetables have and why they are essential in the human diet. During the planned coding part of the workshop for the seventh grade, selected crops and products from farm animals are presented with the Task Family *Building Cubes*. The introductory task for using the <colette/> app is therefore chosen differently than in the fifth and sixth grades. After the app installation (optional: parental notification or prior installation process), the "Easy Coding in Biology: Seventh to Eighth Grade" workshop starts with a simple beginner task: Task 1: "Getting to know the App" (function and features of *Building Cubes*): "Code the shape of an Egg".

Figure 8

Screenshot of the pyramid-shaped plant bed task from the <colette/> application: The task description ("Get to know the <colette/>-app"), and the coding area for the Building Cubes Blockly based commands are provided.

Results

According to the observations, the students could collaborate in the pilot BBP workshop through pair programming, code reviews, collaborative problem-solving, sharing resources, and effective communication. The students worked together on the individual four tasks with *Building Cubes,* liked the tasks, had fun during the workshop, and most of the participants could solve the exercises according to the observations of the instructor. According to the log data, there were 42 entries of the 51 participating students, which required a total of f=582 (f=1-40 per participant or pair) trials to solve the four tasks, with around 3-4 attempts to the correct solution per task (arithmetic mean: 13.9 trials/four tasks, median: 13 trials/four tasks). Furthermore, the instructor observed that the students were very annoyed and frustrated when they or the application accidentally deleted their codes; the app deleted their codes during the shift from AR mode to the coding area or the site crashed. In addition, more time should be allowed when practicing BBP with novice students. The participants in this pilot workshop had all prior experience, therefore, future workshops will need to allow significantly more time and assistance. Due to time management, only four out of six tasks could be tested, and the *Drone AR* task was not tested at all. Furthermore, it was also observed that the participating students themselves recognized their errors in the code (e.g., block in the wrong place, wrong variables entered), and successfully correct them (debugging) using the AR visualization (*Figure 11),* and they could incorporate patterns from other codes into theirs, too.

Figure 9

Pictures from the experience workshop "Easy Coding in Biology" with the <colette/> application in a Slovakian Grammar school. The students collaborated to solve the biological tasks, and had fun during the workshop (n=51)

Figure 10

Pictures from the pilot workshop "Easy Coding in Biology" with the <colette/> application in a Slovakian Grammar school. Students were programming biological tasks (Egg and Conifer); AR image (center; right) and parts of the student's code (left) are shown. The correct solution of the coniferous tree is shown in AR mode (center) and students coding the egg is shown (right)

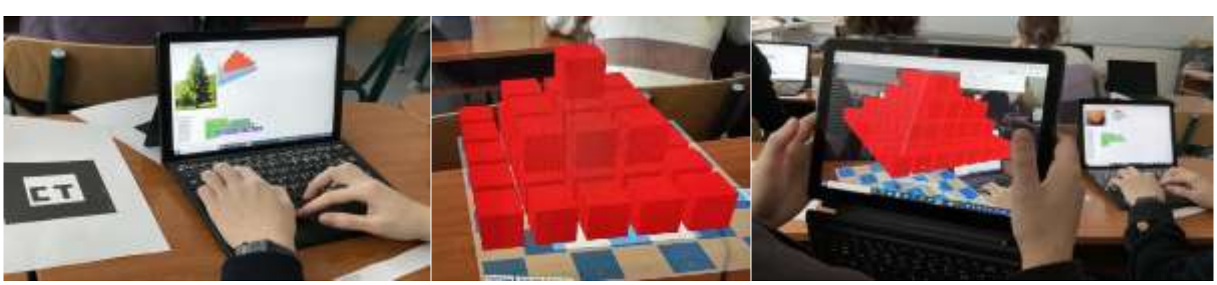


Figure 11

Images from the "Easy Coding in Biology" workshop at a Slovakian grammar school (n=51) demonstrate students programming biological tasks, like an l-shaped plant bed, using the <colette/> application. The app's AR visualization helps students in their debugging (finding/fixing errors): With the AR feature, participants were able to identify that they had specified incorrect variables in their code, resulting in, for example, the incorrect placement of the red cubes of the plant bed within the space.

Regarding their participation in their regular Biology lessons, the participants stated in the evaluation questionnaire after the C-W that 47.05% liked to collaborate in their regular Biology lessons (male students= 40%; female students=50%). When asked how they rate their participation in this "Easy Coding in Biology" workshop, 64.7% of all participants answered with "agree" or "strongly agree" that they liked participating in this course (male=66.67%; female=61.11%). A majority of the students (56%) found the tasks related to Life Science topics interesting. Additionally, 29% of the students found these tasks to be motivating. Among the participants, particularly the female students, 66.67% displayed a high level of interest, while 33.34% demonstrated motivation, surpassing their male counterparts. Regarding task instructions, 80% of the participants found them "easy to understand", and 61.67% perceived the accompanying app as "clear and structured". Notably, most of the girls (83.33%) considered the app to be "well structured", in contrast to 40% of the male students. Approximately 59% of the participants, including 40% of the male students and 77.78% of the female students, found the app "fun to use". Additionally, 50% of the students, with 53.34% male and 52.64% female participants, regarded the app as "easy to work with". About 59% (male=40%, female=77.78%) stated ("agreed/strongly agreed") that the app is "fun to use", and 50% "easy to work with" (male=53.34%, female=52.64%). More than 70% (male=60%, female=83,34%) of the participants would like to use the <colette/> app in school, and 8.82% (male=13.34%, female=5.26%) in their leisure time, as shown in *Figure 12.*
Regarding the qualitative data, most of the students stated that the app, the tasks, and the workshop were fine, no major issues or problems occurred, and they stated no major suggestions for improvement. In the case of problems, the affected students indicated two main criticisms: problems with the AR function (scan or view), and deleting the codes because the program crashed or the students accidentally deleted the code, as shown in *Figure 13*: "I accidentally deleted the whole code, the app should ask for permission if you really want to delete the code."; "The app itself is fine, and the biological workshop concept is really interesting, but the image projections (AR) could use some adjustments". Another student suggested changing the block menu: "When I remove a block from the menu, the menu automatically closes, it would be nicer if it remained open so that I could remove more blocks faster". Furthermore,

students mentioned errors within the task description (e.g., wrong task description), time management (*"app is too time-consuming for me"*), and the coding itself ("I do not understand programming").

Figure 12

Acceptance of Slovakian participants ("agreed/strongly agreed") on collaboration, interest, perceived usage, app's structure, and motivation on the coding tasks during pilot workshop "Easy Coding in Biology" (n=34)

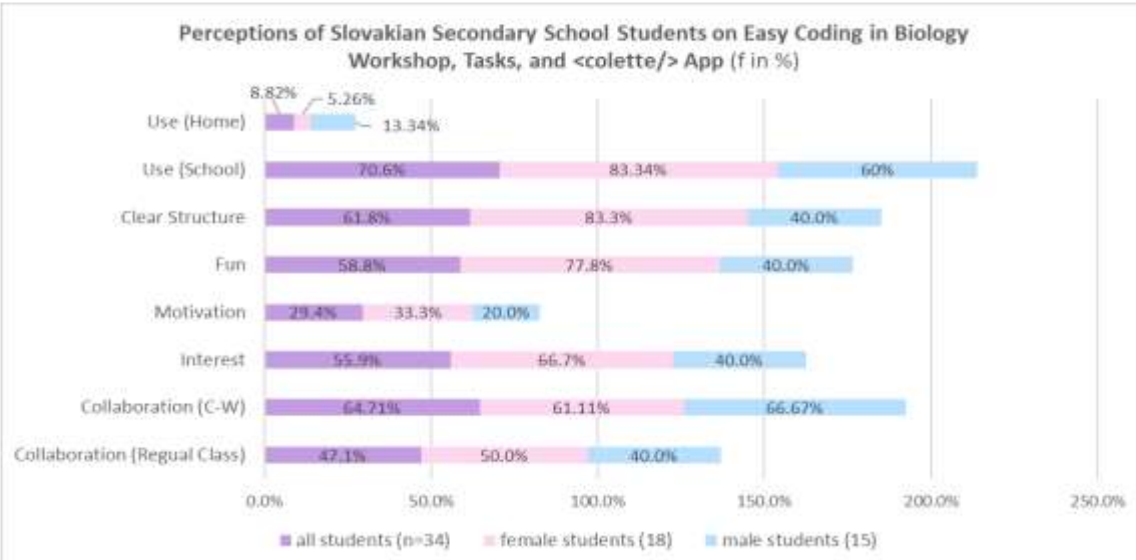
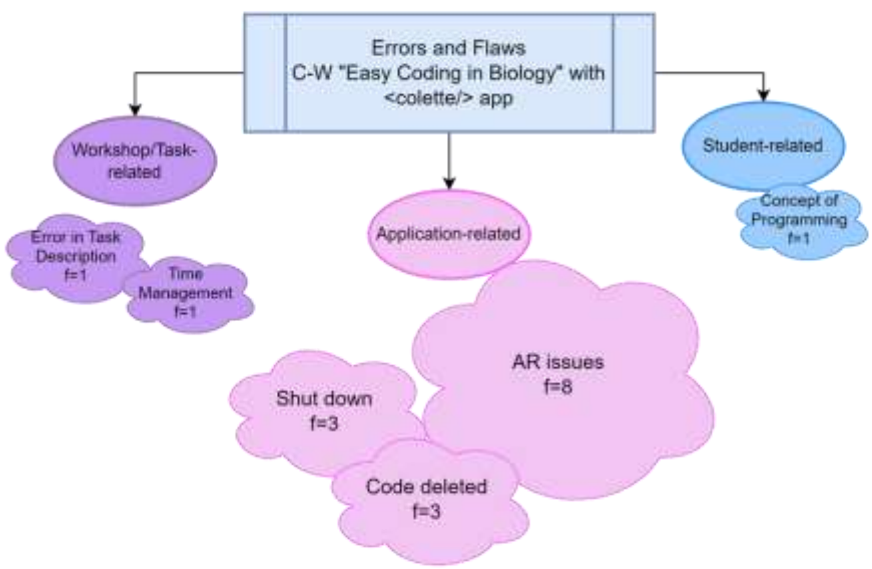


Figure 13

Presentation of qualitative data analysis (Mayring, 2010). Students critique regarding the pilot workshop, coding tasks, and <colette/> app (n=34). The points of criticism are shown in clouds, the more frequently the mention (f) the bigger the presenting cloud.



Discussion and Limitation

It is important to note that the initial pilot study primarily focused on CT tasks and did not directly measure the impact on students' Biology content learning. Future iterations of the project will include comprehensive evaluations that assess both CS and CT learning outcomes, as well as the impact on students' Biology content knowledge. These evaluations will provide a more comprehensive understanding of the effectiveness of integrating CS and CT concepts within Biology classes. The reason for this approach is that the authors first set the CS tasks and wanted to (1) evaluate and test their implementation, (2) they can also be performed by Biology teachers, and (3) the tasks increase slightly in difficulty.

Within the different biological scenarios of *Building Cubes*, one of the introduced TF in this study, there are various shapes and objects that students can recreate using Blockly code. As the TF is coding-based, the assignment types of the Generic Tasks (GT) will be applied (Milicic et al., 2020; Stäter et al., 2023a; Stäter et al., 2023b). Currently, only the assignment type "Implementation" is realized in the portal, further ones will follow within the <colette/> project (Stäter et al., 2023a; Stäter et al., 2023b; Läufer et al, 2021; Colette-project, 2021).

Other limiting factors of this study are that the number of trials per participant for the correct code cannot be assigned to the individual tasks (due to identical naming); and that the workshop "Easy Coding in Biology" was tested only with a small sampling using *Building Cubes*. *Drone AR* was not tested with students at all in the pilot workshop. During this research, only one possible teaching example for *Drone AR* was described. Hence, no general conclusions can be drawn. Due to the same naming of the four tasks in the server code, the number of attempts could not be assigned to the individual tasks, only the total number of calls to the application and the total number of attempts of all tasks per student or pair. In addition, there were issues with the AR marker printing being too large, causing the individual blocks to appear too far apart in the AR visualization, and children having to hold the CT scan far from their devices to make the AR view work. Another issue is that the mobile learning app is currently only available in a beta version for Android and iOS. Therefore, some technical flaws are still present in the application, for example, issues with AR visualization and markers detection, which were also common problems in other studies (Ajit, G., 2021).

However, since the app is still in its beginner stage, the feedback and suggestions of the students help to improve not only the app but also the workshop (e.g., change the menu for blocks, improve AR, fix issues with AR scan, and change time management). Errors within the task description were deleted right after the experimental workshop. The tasks were evaluated and adapted for the 10-14-year-olds. Since the pilot workshop was conducted with older students, more time was calculated for the lower grades to offer the participants efficient support with any of their concerns and questions. Younger novice students often have no previous experience with programming languages, because programming (informatics) is only mandatory in Slovakian primary school, not in Austria and Germany (BMWF, 2023; Deutscher Bildungsserver, 2022; Ministry of Education, Science, Research and Sports of the Slovak Republic, 2023b).

Most of the participants could solve all four tasks, and they stated that they wanted to continue using the app in school because the tasks were motivating or interesting. Striking in the experimental workshop was that the participating students rated their participation significantly higher than in their regular Biology lessons, especially the girls. In addition, the students found the app and tasks motivating and clearly structured. There were clear gender-specific differences that need to be researched in more in-depth studies. However, the students indicated that they did not want to use the app at home. This could be due to the extra help that the workshop instructors had to provide when problems occurred (such as errors in code, AR malfunction, deleted codes, page shutdown, or page freezing) that students may not feel confident about successfully solving it on their own at home.

In addition, the planning of the coding workshop also considers that the teacher has to offer more support in the workshops than in upper school classes. As this is a cross-curricular workshop, it requires Biology teachers who also teach CS, or have excellent or at least advanced skills in CS or block-based programming. To guarantee this issue, it is necessary that teachers are excellently trained in dealing with CT skills, CS concepts, new technologies, and educational tools, which they are not yet. Studies show that the biggest issues in integrating CT into classrooms are a lack of ready-to-use materials and professionally educated teachers (Bocconi et al, 2022; Borkulo et al., 2022). Therefore, future literature and research on <colette/> must aim to stress picking up teachers at their level of knowledge, providing hands-on teaching materials as OER, and training educators in the best possible way, in the form of professional teacher training.

Regarding promoting CT skills, according to the experimental workshop observations, the additional AR visualization within the <colette/> app helps to identify students' own and third-party errors and patterns to incorporate them into their own code and improve or correct it. Therefore, mAR helps students to visualize difficult (STEM) concepts, and assists with problem-solving, pattern recognition, and debugging. This observation has already been made in previous studies or workshops with secondary students (Cheli, et al. 2018; Saidin et al., 2015; Ke and Hsu, 2015; Schmidthaler et al.., 2023a), and in previous <colette/> workshops, held by the authors (Schmidthaler et al., 2023b). The biological tasks are also particularly suitable for abstraction and generalization, this was also confirmed by the observations in the experimental workshop based on the students' reactions. It can be assumed that the 3D visualization, made of unit-cubes, possibly reminds students of the game "Minecraft" (Minecraft, 2023) due to its similar abstraction of objects. Research on Minecraft indicates that the tool has the potential to promote creativity and understanding of concepts as an educational tool (Short, 2012; Cipollone et al, 2014), and combined with 3D printing, can promote the CT skill abstraction (Roscoe et al., 2014). However, to draw conclusions from studies about Minecraft to <colette/>, further studies must follow.

Conclusion and Outlook

This research on BBP tasks in science education shows the possibility of using the educational application <colette/> to teach cross-curricular CS and CT competencies in Biology lessons at the European secondary level. Through selfdeveloped teaching examples in the "Easy Coding in Biology" workshop for grades five up to eight, CS concepts such as algorithms, coding, CT skills such as pattern recognition, abstraction (Bocconi et. al, 2016), problem-solving, and other concepts such as experimentation and spatial imagination (coordinate system) can successfully be taught. The pilot workshop with Slovakian grammar students showed that the participants, especially the girls, liked the workshop and biological tasks very much, and the majority could solve the exercises. Although there are already some nationwide projects and interventions, like "Girls* Only" in Austria (Leitner, et. al, 2023), currently, there is still a shortage of women in STEM professions, especially in technical and IT ones (Bergmann et al, 2021; Lehman et. al, 2016; Han, 2016). However, participation is higher in the field of Biology (Eddy et. al, 2014; Statistics of Promotion, 2021). Therefore, future studies should focus on whether the connection and combination of the subject of Biology and Computer Science or biological topics with CS concepts such as programming via the <colette/> app, bridges this gap and tackles the issue of women lacking in technical professions by raising their interest in this topic to work in this field later in life.

Although some issues occurred in this study due to the AR function and CT scan, most of the participants had fun during the coding workshop, and the tasks increased their motivation and collaboration. Therefore, it can be concluded from the results that the mobile learning app <colette/> can be used successfully as an educational tool in European schools in Biology lessons. Above that, <colette/> has great potential to promote CT, although the app development is not finished yet. Further research, regarding <colette/> teacher training and bigger sampling, will take place in the summer of 2023 in Austria. Moreover, after the task evaluation, several Austrian lower secondary schools will participate in the "Easy Coding in Biology" workshops, using *Building Cubes* and *Drone AR* tasks, until August 2023 in Austria.

Acknowledgments

The project is (partially) funded by the Erasmus+ grant program of the European Union (2020-1-DE03-KA201-077363). Neither the European Commission nor the project's national funding agency PAD are responsible for the content nor liable for any losses or damage resulting from the utilization of these resources.

References

Andersen, R., Mørch, A.I., Litherland, K.T. (2021). Learning Domain Knowledge Using Block-Based Programming: Design-Based Collaborative Learning. In: Fogli, D., Tetteroo, D., Barricelli, B.R., Borsci, S., Markopoulos, P., Papadopoulos, G.A. (eds) End-User Development. IS-EUD 2021. Lecture Notes in Computer Science, 12724. Springer, Cham. https://doi.org/10.1007/978-3-030-79840-6_8

Ajit, G. (2021). A Systematic Review of Augmented Reality in STEM Education. *Studies of Applied Economics*, 39(1). https://www.researchgate.net/publication/348814721_A_Systematic_Review_of_Augmented_Reality_in_STEM_Education accessed, 13.6.2023

Beaufort, J. (2007). Grapes. https://www.publicdomainpictures.net/en/view-image.php?image=223603&picture=grapes last accessed, 22.1.2023

Bergmann, N., Lachmayr, N., Mayerl, M. & Pretterhofer, N. (2021). Frauen in technischen Ausbildungen und Berufen – Fokus auf förderliche Ansätz. https://www.ams-forschungsnetzwerk.at/downloadpub/AMS_Frauen_in_Technik_LR_oeibf_2021.pdf last accessed, 21.1.2023

Blockly, (2022). Introduction to Blockly. Google Developers.https://developers.google.com/blockly/guides/overview last accessed, 12.1.2023

Blockly Games, 2022. About Blockly Games https://blockly-games.appspot.com last accessed, 12.1.2023

Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., Engelhardt, K. (2016). *Developing computational thinking in compulsory education – Implications for policy and practice*. 10.2791/792158

Bocconi, S., Chioccariello, A., Kampylis, P., Dagienė, V., Wastiau, P., Engelhardt, K., Earp, J., Horvath, M.A., Jasutė, E., Malagoli, C., Masiulionytė-Dagienė, V. and Stupurienė, G. (2022). Reviewing Computational Thinking in Compulsory Education, Inamorato dos Santos, A., Cachia, R., Giannoutsou, N. and Punie, Y. editor(s), Publications Office of the European Union, Luxembourg. 10.2760/126955

Borkulo, van, S., Chytas, C., Drijvers, P., Barendsen, E. & Tolboom, J. (2021). Computational Thinking in the Mathematics Classroom: Fostering Algorithmic Thinking and Generalization Skills Using Dynamic Mathematics Software. In M. Berges, A. Mühling & M. Armoni (Hrsg.), *The 16th Workshop in Primary and Secondary Computing Education*. 1–9. DOI: https://doi.org/10.1145/3481312.3481319

Boon, P., (2003). “Meetkunde op de computer.” *Tijdschrift voor nascholing en onderzoek van het reken-wiskundeonderwijs*, *22(1), pp.17-26*. https://www.fi.uu.nl/publicaties/literatuur/5787.pdf last accessed, 12.1.2023

Blum, L. and Cortina, T.J. (2007). CS4HS: an outreach program for high school CS teachers. *In ACM SIGCSE Bulletin.* Vol. 39(1). 19-23. 10.1145/1227310.1227320

Blume, M., (2019). Conifer in Damgarten. https://commons.wikimedia.org/wiki/File:Conifere,_Ribnitz-Damgarten_%28P1070875%29.jpg last accessed, 22.1.2023

Cheli, M., Sinapov, J., Danahy, E. E., & Rogers, C. (2018, March). Towards an augmented reality framework for k-12 robotics education. *In Proceedings of the 1st International Workshop on Virtual, Augmented, and Mixed Reality for HRI (VAM-HRI)*. https://www.eecs.tufts.edu/~jsinapov/papers/Cheli_VAMHRI2018.pdf

Cipollone, M., Schifter, C. C., & Moffat, R. A. (2014). Minecraft as a creative tool: A case study. International Journal of Game-Based Learning (IJGBL), 4(2), 1-14. DOI: 10.4018/ijgbl.2014040101

Deutscher Bildungsserver. (2022). Lehrplan Grundschule in Deutschland. https://www.bildungsserver.de/lehrplaene-fuer-die-grundschule-1660-de.html

Colette-Project. (2022). Computational Thinking Learning Environment for Teachers in Europe https://colette-project.eu last accessed, 22.1.2023

Davis, F. D. (1985). *A technology acceptance model for empirically testing new end-user information systems: Theory and results* (Doctoral dissertation, Massachusetts Institute of Technology).

Eddy, S.L., Brownell, S.E., Wenderoth, M.P. (2014). Gender Gaps in Achievement and Participation in Multiple Introductory Biology Classrooms. *Cell Biology Education*, 2014; 13 (3): 478 DOI: 10.1187/cbe.13-10-0204

Eldokhny, A. A. (2021). Effectiveness of augmented reality in online distance learning at the time of the COVID-19 pandemic. International Journal of Emerging Technologies in Learning, 16(9): 210–213. https://doi.org/10.3991/ijet.v16i09.17895

Federal Ministry of Education (BMBWF). (2018). RIS. Curriculum for Secondary School. https://www.ris.bka.gv.at/GeltendeFassung.wxe?Abfrage=Bundesnormen&Gesetzesnummer=10008568 last accessed, 1.2.2023

Federal Ministry of Education (BMBWF). (2023). RIS. Curriculum for Primary School. RIS - Lehrpläne der Volksschule und der Sonderschulen - Bundesrecht konsolidiert, Fassung vom 20.02.2023 (bka.gv.at) last accessed, 1.2.2023

Gimmel, B., (2004). Fliegenpilz - Amanita muscaria. https://commons.wikimedia.org/wiki/File:Fliegenpilz_Amanita_muscaria.jpg last accessed, 1.2.2023

Goldberg, D.S., Grunwald, D., Lewis, C., Feld, J.A. and Hug, S., (2012). Engaging computer science in traditional education: the ECSITE project. In Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education. *(ITiCSE '12). Association for Computing Machinery, New York. NY. USA.* 17(1). 351-356 . https://doi.org/10.1145/2325296.2325377

Gupta, V., Iimia, J., Pau, I., Rodríguez-Patón, A. (2017). BioBlocks: Programming Protocols in Biology Made Easier. ACS Synthetic Biology. *American Chemical Society*. 1230-1232. https://doi.org/10.1021/acssynbio.6b00304

Han, S. W. (2016). National education systems and gender gaps in STEM occupational expectations. International Journal of Educational Development, 49, 175–187. https://doi.org/10.1016/j.ijedudev.2016.03.004

Hsu, T.C., Chang, S.C., Yu-Ting Hung, Y.T. (2018). How to learn and how to teach computational thinking: Suggestions based on a review of the literature. *Computers & Education*. 126(1). 296-310. https://doi.org/10.1016/j.compedu.2018.07.004

Ke, F., & Hsu, Y.-C. (2015). Mobile augmented-reality artifact creation as a component of mobile computer-supported collaborative learning. The Internet and Higher Education, 26, 33–41. https://doi.org/10.1016/j.iheduc.2015.04.003

Läufer, T., Stäter, R., Ludwig, M. (2022). Das Projekt <colette/> Computational Thinking (auch) im Mathematikunterricht. *Digitales Lehren und Lernen. GDM-Mitteilungen 113*. https://www.researchgate.net/publication/364598597_Das_Projekt_colette_Computational_Thinking_auch_im_Matheunterricht#fullTextFileContent last accessed, 22.2.2023

Lehman, K. J., Sax, L. J., and Zimmerman, H. B. (2016). Women planning to major in computer science: Who are they and what makes them unique? *Computer Science Education*, 26(4), 277–298. DOI: 10.1080/08993408.2016.1271536

Leitner, S., Hinterplattner, S., Sabitzer, B., and Schmidthaler, E. (2023). STEM Promotion to Empower and Support Girls and Women: A Cross-University Project in Austria. *International Journal for Cross-Disciplinary Subjects in Education (IJCDSE)*. 14(1). 1415-1420. 10.20533/ijcdse.2042.6364.2023.0580

Lockwood, J., & Mooney, A. (2018). Computational Thinking in Secondary Education: Where does it fit? A systematic literary review. International Journal of Computer Science Education in Schools, 2(1), 41-60. https://doi.org/10.21585/ijcses.v2i1.26

Materialbörse. (2022). JKU COOL Lab. https://www.cool-lab.net/materialboerse/ last accessed, 22.2.2023

Marini, A., Nafisah, S., Sekaringtyas, T., Safitri, D., Lestari, I., Suntari, Y., ... & Iskandar, R. (2022). Mobile Augmented Reality Learning Media with Metaverse to Improve Student Learning Outcomes in Science Class. *International Journal of Interactive Mobile Technologies*, *16*(7). https://doi.org/10.3991/ijim.v16i07.25727

Mayring, Ph. (2010). Qualitative Inhaltsanalyse. Grundformen und Techniken. 11. Aufl. Weinheim: Beltz.

McGee, M. G. (1979). Human spatial abilities: Psychometric studies and environmental, genetic, hormonal, and neurological influences. *Psychological Bulletin*, *86*(5), 889–918. https://doi.org/10.1037/0033-2909.86.5.889

Milicic, G., Wetzel, S., Ludwig, M. (2020). Generic Tasks for Algorithms. *Future Internet*. 12(9):152. https://doi.org/10.3390/fi12090152

Milicic, G., van Borkulo, S.P., Medova, J., Wetzel, S., Ludwig, M. (2021). Design and Development of a Learning Environment for Computational Thinking: The Erasmus+ COLETTE Project. *Proceedings of the Conference EduLearn 2021*. 1(1). 7376-7383. DOI: 10.21125/edulearn.2021.1495

Minecraft. (2023). http://minecraft.net/ last accessed, 22.2.2023

Ministry of Education, Science, Research and Culture of the state of Schleswig-Holstein, (2022). Curriculum for Secondary School. https://fachportal.lernnetz.de/files/Fachanforderungen%20und%20Leitf%C3%A4den/Sek.%20I_II/Lehrpl%C3%A4ne/Biologie_%28HS%2CRS%2CRegS%2CGym%2CGemS%29.pdf last accessed, 1.2.2023

Ministry of Education, Science, Research and Sports of the Slovak Republic. (2023a). Curriculum for Secondary Education. https://www.minedu.sk/data/files/5645_rup_a_vs_vzdel_na_zisk_nsv_21122015.pdf last accessed, 26.1.2023

Ministry of Education, Science, Research and Sports of the Slovak Republic. (2023b). Curriculum for Primary Education. https://www.minedu.sk/data/files/6948_priloha_2_rup.pdf last accessed, 26.1.2023

National Education Association. (2015). Preparing 21st century students for a global society: An educator's guide to the "Four Cs". Retrieved from http://www.nea.org/assets/docs/A-Guide-to-Four-Cs.pdf last accessed, 22.2.2023

Schmidthaler E., Andjic, B., Schmollmüller, M., Sabitzer, B. and Lavicza, Z. (2023a). Mobile Augmented Reality in Biological Education: Perceptions of Austrian Secondary School Teachers, Journal on Efficiency and Responsibility in Education and Science, 16(2). 55-64. http://dx.doi.org/10.7160/eriesj.2023.160203, in press.

Schmidthaler, E., van Borkulo, S., Cápay, M.; Kristinsdóttir, B., Stäter, R., Läufer, T., Ludwig, M., Hornsby, D., Skogø, J. and Lavicza, Z. (2023b). Design and Evaluation of Computational Thinking Tasks in the Project: Experiences Gained from Workshops with Secondary and Grammar School Students in Austria, the Netherlands, and Slovakia. In Proceedings of the 15th International Conference on Computer Supported Education. 1. 297-304. https://10.5220/0011974700003470

Schmidthaler, E., Schalk, M., Schmollmüller, M., Hinterplattner, S., Hörmann, C., Anđić, B., Rottenhofer, M., Lavicza, Z. and Sabitzer, B. (2023c). The interdisciplinary implementation of poly-universe to promote computational thinking: Teaching examples from biological, physical, and digital education in Austrian secondary schools. Front. Psychol. 14:1139884. doi: https://10.3389/fpsyg.2023.1139884

Piaget, J., (1971). Psychology and Epistemology: Towards a Theory of Knowledge. New York: Grossman.

Polat, E., Hopcan, S., Kucuk, S., Sisman, B. (2021). A comprehensive assessment of secondary school students' computational thinking skills. *British Journal of Educational Technology.* 52(5). 1965-1980. https://doi.org/10.1111/bjet.13092

Prokop, P., Prokop, M. & Tunnicliffe, S.D. (2007) Is Biology boring? Student attitudes toward Biology, *Journal of Biological Education*. 42(1). 36-39. DOI: 10.1080/00219266.2007.9656105

Qin, H. (2009). Teaching computational thinking through bioinformatics to Biology students. SIGCSE Bull. 41(1). 188–191. https://doi.org/10.1145/1539024.1508932

Rubinstein, A. and Chor, B., 2014. Computational thinking in life science education. PLoS Comput Biol, 10(11), 10(11):e1003897. DOI: 10.1371/journal.pcbi.1003897

Roscoe, J.F., Fearn, S. and E. Posey, E. (2014). Teaching Computational Thinking by Playing Games and Building Robots. *International Conference on Interactive Technologies and Games*, Nottingham, UK, 14(1). 9-12, doi: 10.1109/iTAG.2014.15

Sabitzer, B. (2013a). A neurodidactial approach to cross-curical open learning. cool informatics. Habilitation. Alpen-Adria-University Klagenfurt.

Sabitzer, B., Antonitsch P., (2012). Of Bytes and Brain. In-formatics Meets Neurodidactics. In: L. Gómez Chova,I. Candel Torres, A. López Martìnez (Eds.): INTED2012 Proceedings. Barcelona: IATED, pp. 2003-2012. https://library.iated.org/view/SABITZER2012OFB last accessed, 1.2.2023

Sabitzer B., Pasterk, S. (2014). Mobile Learning For COOL Informatics. COopertive Open Learning in a Vocational High School. CSEDU. https://www.academia.edu/7587771/CSEDU2014_Mobile_Learning_For_COOL_Informatics last accessed, 26.1.2023

Sabitzer, B., Pasterk, S., (2013b). Informatic is COOL: cross-curricular concepts for COmputer-supported OpenLearning in secondary schools. ICERI conference paper. 6(1). 2739-2749. https://library.iated.org/view/SABITZER2013INF last accessed, 26.1.2023

Saidin, N., Noor, A.H. & Noraffandy, Y. (2015). A Review of Research on Augmented Reality in Education: Advantages and Applications. International Education Studies. 8(13). 10.5539/ies.v8n13p1

Statistics of Promotion. German Ministry. (2021). https://www.destatis.de/DE/Themen/Gesellschaft-Umwelt/Bildung-Forschung-Kultur/Hochschulen/Publikationen/_publikationen-innen-hochschulen-promovierende.html last accessed, 26.1.2023

Scratch. (2022). Scratch - Imagine, Program, Share. https://scratch.mit.edu/ last accessed, 26.1.2023

Shih, W.C. (2017). Mining Learners' Behavioral Sequential Patterns in a Blockly Visual Programming Educational Game. In *International Conference on Industrial Engineering, Management Science and Application (ICIMSA)* (pp. 1-2). doi: 10.1109/ICIMSA.2017.7985594

Short, D. (2012). Teaching scientific concepts using a virtual world-minecraft. *Teaching Science.* 58(3). 55-58. https://www.researchgate.net/publication/236587414_Teaching_Scientific_Concepts_using_a_Virtual_World_-_Minecraft#fullTextFileContent last accessed, 26.1.2023

Stäter, R. S., Läufer, T., & Ludwig, M. (2023a). How <colette/> Facilitates Teaching Computational Thinking. In M. Ludwig, S. Barlovits, A. Caldeira, & A. Moura (Eds.), Research On STEM Education in the Digital Age. Proceedings of the ROSEDA Conference (pp. 197 200). WTM. https://doi.org/10.37626/GA9783959872522.0.25

Stäter, R. S., Läufer, T., & Ludwig, M. (2023b). Teaching Computational Thinking with <colette/>. In M. Ludwig, S. Barlovits, A. Caldeira, & A. Moura (Eds.), Research On STEM Education in the Digital Age. Proceedings of the ROSEDA Conference (pp. 123 130). WTM. https://doi.org/10.37626/GA9783959872522.0.15

University of California San Diego. Biology Meets Programming. Bioinformatics for Beginners. (2023) https://www.coursera.org/learn/bioinformatics last accessed 22.2.2023

University of Zurich. Programming in Biology. (2018). https://mnf.openedx.uzh.ch/courses/course-v1:MNF+BIO134+HS2018/about#:~:text=Programming%20in%20Biology%20is%20an,including%20examples%20from%20recent%20research. last accessed 22.2.2023

Vasconcelos, L., Kim, C. Preparing preservice teachers to use block-based coding in scientific modeling lessons. *Instr Sci* 48, 765–797 (2020). https://doi.org/10.1007/s11251-020-09527-0

Vetter, T.R. (2017). Descriptive Statistics: Reporting the Answers to the 5 Basic Questions of Who, What, Why, When, Where, and a Sixth, So What? *Anesth Analg.* 125(5). 1797-1802. doi: 10.1213/ANE.0000000000002471

Weintrop, D., Wilensky, U. 2017. Comparing Block-Based and Text-Based Programming in High School Computer Science Classrooms. *ACM Transactions on Computing Education*. 18(1). 1-25. https://doi.org/10.1145/3089799

Wilson, B. (2010). Strawberry. https://www.flickr.com/photos/billy_wilson/12958149123 22.1.2023

Xu, Z., Ritzhaupt, A.D., Tian, F., Umapathy, K. (2019). Block-based versus text-based programming environments on novice student learning outcomes: a meta-analysis study, *Computer Science Education, 29*(2-3), 177-204. https://doi.org/10.1080/08993408.2019.1565233

Yamashita, S., Tsunoda, M., Yokogawa, T. (2017). Visual Programming Language for Model Checkers Based on Google Blockly. In Felderer, M., Méndez Fernández, D., Turhan, B., Kalinowski, M., Sarro, F., Winkler, D. (Eds.), *Product-Focused Software Process Improvement. PROFES. Lecture Notes in Computer Science*, 10611. 597–601. Springer, Cham. https://doi.org/10.1007/978-3-319-69926-4_49

Yuen, T.T. and Robbins, K.A., (2015). A Qualitative Study of Students' Computational Thinking Skills in a Data-Driven Computing Class. *ACM Transactions on Computing Education (TOCE)*. 14(4). 27. DOI:10.1145/2676660

Zender, R., Höfler, J., Wolfien, P. & Lucke, U.(2014). ScratchDrone - Systematische Programmierung von Flugdrohnen für den Informatikunterricht. In: Trahasch, S., Plötzner, R., Schneider, G., Sassiat, D., Gayer, C. & Wöhrle, N. (Hrsg.), DeLFI 2014 - Die 12. e-Learning Fachtagung Informatik. Bonn: Gesellschaft für Informatik e.V. 109-120. https://dspace.gi.de/handle/20.500.12116/2991 last accessed, 1.2.2023

Ziegenfuss, D. H., & LeMire, S. (2019). Backward Design: A Must-Have Library Instructional Design Strategy for Your Pedagogical and Teaching Toolbox

Appendix

Table 4

Questions about students' app perceptions on "Easy Coding in Biology" workshop and <colette/> app based on the Technology Acceptance Model (Davis, 1985)

No	Question	Answer Type
1	Age	open-ended
2	Gender	SC: female, male, non-binary, no gender, no answer)
3	Which task(s) did you work on? Please tick the appropriate answer(s)	MC: Create an Algorithm for an Egg, Create an Algorithm for a L-Shaped Plant Bed, Create an Algorithm for a Fruit (Wine grape, strawberry), Create an Algorithm for a Pyramid-Shaped Plant Bed, Create an Algorithm for a Conifer)
4	It is enjoyable to collaborate in my regular Biology (Science) classes	Likert scale: Strongly disagree - Disagree - Neutral - Agree - Strongly agree
6	It was enjoyable to collaborate in this Biology coding workshop	Likert scale: Strongly disagree - Disagree - Neutral - Agree - Strongly agree
7	The biological tasks were very interesting	Likert scale: Strongly disagree - Disagree - Neutral - Agree - Strongly agree
8	It was easy to understand the instructions.	Likert scale: Strongly disagree - Disagree - Neutral - Agree - Strongly agree
9	It took a long time to learn to use the app.	Likert scale: Strongly disagree - Disagree - Neutral - Agree - Strongly agree
10	This app is difficult to use.	Likert scale: Strongly disagree - Disagree - Neutral - Agree - Strongly agree
11	The app is clear.	Likert scale: Strongly disagree - Disagree - Neutral - Agree - Strongly agree
12	This app is fun to use.	Likert scale: Strongly disagree - Disagree - Neutral - Agree - Strongly agree
13	The app easily does what I want.	Likert scale: Strongly disagree - Disagree - Neutral - Agree - Strongly agree
14	I would like to use this app in school.	Likert scale: Strongly disagree - Disagree - Neutral - Agree - Strongly agree
15	I would like to use this app outside of school.	Likert scale: Strongly disagree - Disagree - Neutral - Agree - Strongly agree
16	The app has apparent faults.	Likert scale: Strongly disagree - Disagree - Neutral - Agree - Strongly agree
16a	If so, please explain why.	Open-ended

17	Did you have experience with programming before this workshop (e.g., Scratch, Blockly, Python)?	Likert scale: No experience - A bit of experience - Pretty much experienced - A lot of experience
17a	If yes, describe your experience (e.g., what language, at school, at home, what purpose,...)!	Open-ended
18	Do you have tips and tops for us?	Open-ended

Corresponding Author Contact Information:

Author name: Eva Schmidthaler

Department: School of Education/STEM Didactics

University, Country: Johannes Kepler University in Linz/Austria

Email: eva.schmidthaler@jku.at

ORCID: 0000-0001-9633-8855

Please Cite: Schmidthaler, E., Stäer, R., Capay, M., Ludwig, M., Lavicza, Z. (2023). Easy Coding in Biology: Pilot Workshop Design and Experiences from Block-Based Programming with <colette/> in Secondary Education. *Journal of Research in Science, Mathematics and Technology Education*, 6(SI), 177-206 DOI: https://doi.org/10.31756/jrsmte.619SI

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Ethics Statement: If this study involve human/animal participants please indicate you reviewed and approved your study.

Author Contributions: All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

Received: March 1, 2023 ▪ Accepted: June 13, 2023