# HOW <COLETTE/> FACILITATES TEACHING COMPUTATIONAL THINKING

Rebecca S. Stäter, Tim Läufer and Matthias Ludwig
Goethe University Frankfurt, Germany

***Abstract.*** Although the everyday life of students becomes increasingly digitalized, this process does not necessarily lead to a proficient use and understanding of digital tools. Students must be taught how to use them, but the lack of easy-to-use teaching materials hinders teachers to include Computational Thinking in their lessons. <colette/> is an easy-to-use two-component-system to include Computational Thinking in a variety of school subjects. <colette/> guides teachers to create meaningful tasks and gives students an environment to work on those tasks.

*Key words: Computational Thinking, digital teaching, task families.*

## INTRODUCTION

Computational Thinking (CT) has emerged as a skillset which is needed in everyday life and education (Wing, 2006; Bocconi, 2016). For our purposes CT consists of Abstraction, Algorithmic Thinking, Automation, Debugging, Decomposition, and Generalization (see Bocconi (2016) for a more detailed description of these skills). <colette/> is a free to use web portal for educators to teach CT concepts in an engaging way where students will use the app on their smartphones or tablets. A teacher can create tasks and combine them to learning paths (Roth, 2015), which can then be traversed digitally by the students via an app. Additionally, we want to create opportunities in workshops for educators themselves to learn about CT and how to integrate this in their classroom.

## THE <COLETTE/> SYSTEM

The <colette/> system offers a low-threshold approach both for students and educators to learn about and teach CT. The web portal as an authoring tool guides teachers to create meaningful learning paths, the app is user-friendly designed and uses block-based language for coding exercises. As students can work with their own mobile devices like their smartphones, <colette/> makes use of the Bring-Your-Own-Device approach avoiding, hence, the necessity for schools to invest in expensive equipment.

### Creation of Paths with Tasks in the Web Portal.

The images and parts of the text of the web portal and the app are also found in the article by Stäter et al. in these proceedings (Stäter et al., 2023).

The web portal is an authoring and path management tool. Tasks can be created by editing Task Families and then combined to a learning path (Roth, 2015). The handbook, a source of information about the web portal, app and CT, is found in the web portal as well.

A task is created by choosing a Task Family as a template (Figure 1.1), choosing an assignment type (Figure 1.2), selecting a scenario (or variation) of the Task Family

Stäter, R. S., Läufer, T., & Ludwig, M. (2023). 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

(Figure 1.3), editing the scenario's settings (Figure 1.4), and finishing the task data such as the title, problem definition (assignment text), task picture, and hints (Figure 1.5).

This process can be repeated to create multiple tasks which can then be combined to a learning path consisting of a title, a description, and the tasks. When creating a path, a code is automatically assigned. This code is used to identify the path and can be entered in the app so the students can work on this path using the app.

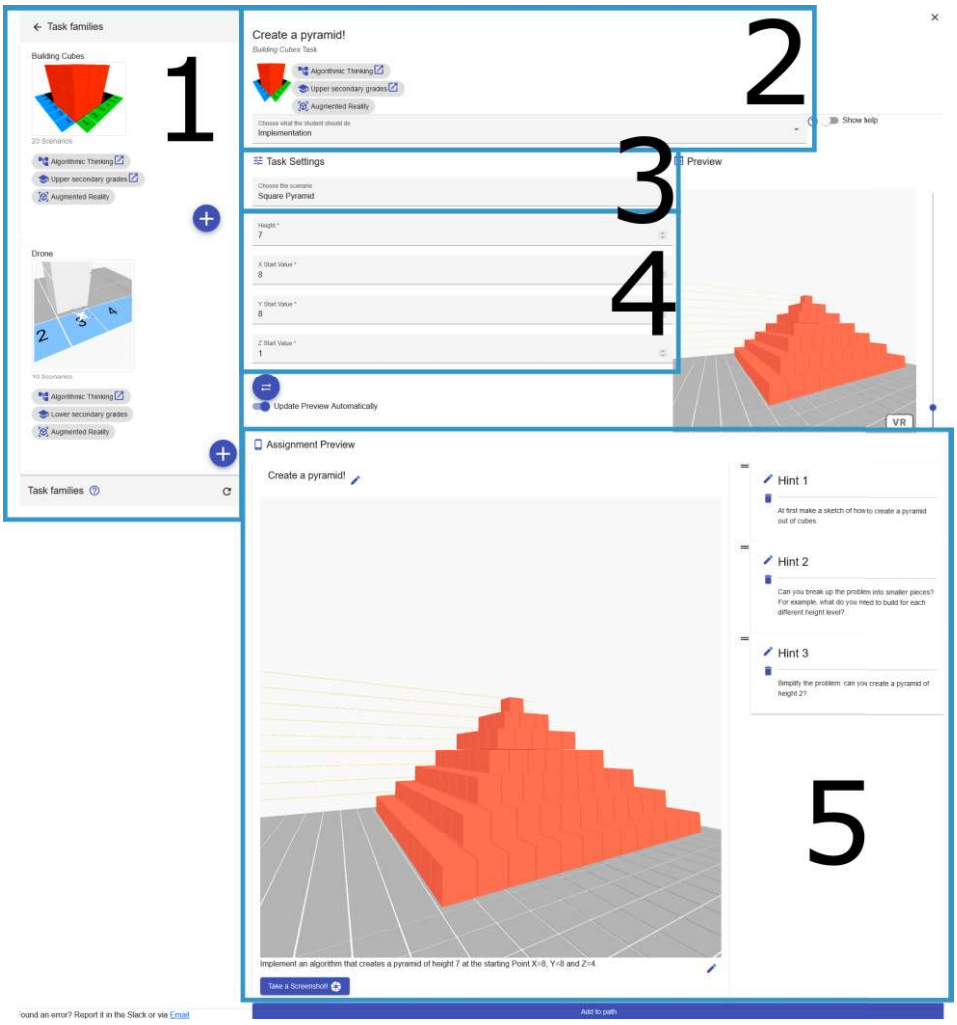


Figure 10: Creating a task in the <colette/> web portal (screenshot). You can see the work flow of an educator going from step 1 to step 5 (labeled by large black numbers) to then save the created task from the "Building Cubes" Task Family. A final path consists of multiple such tasks.

## Viewing and Solving a Path in the <colette/> App

After the educator has created a learning path (Roth, 2015), the student can start working on that path by using the app (Figure 2). The included AR view emerges the code into the reality by showing either the coded figure (in the Task Family "Building Cubes") or the coded route (in the Task Family "Drone"). In this way a student can debug their code and see whether the code did what they expected.

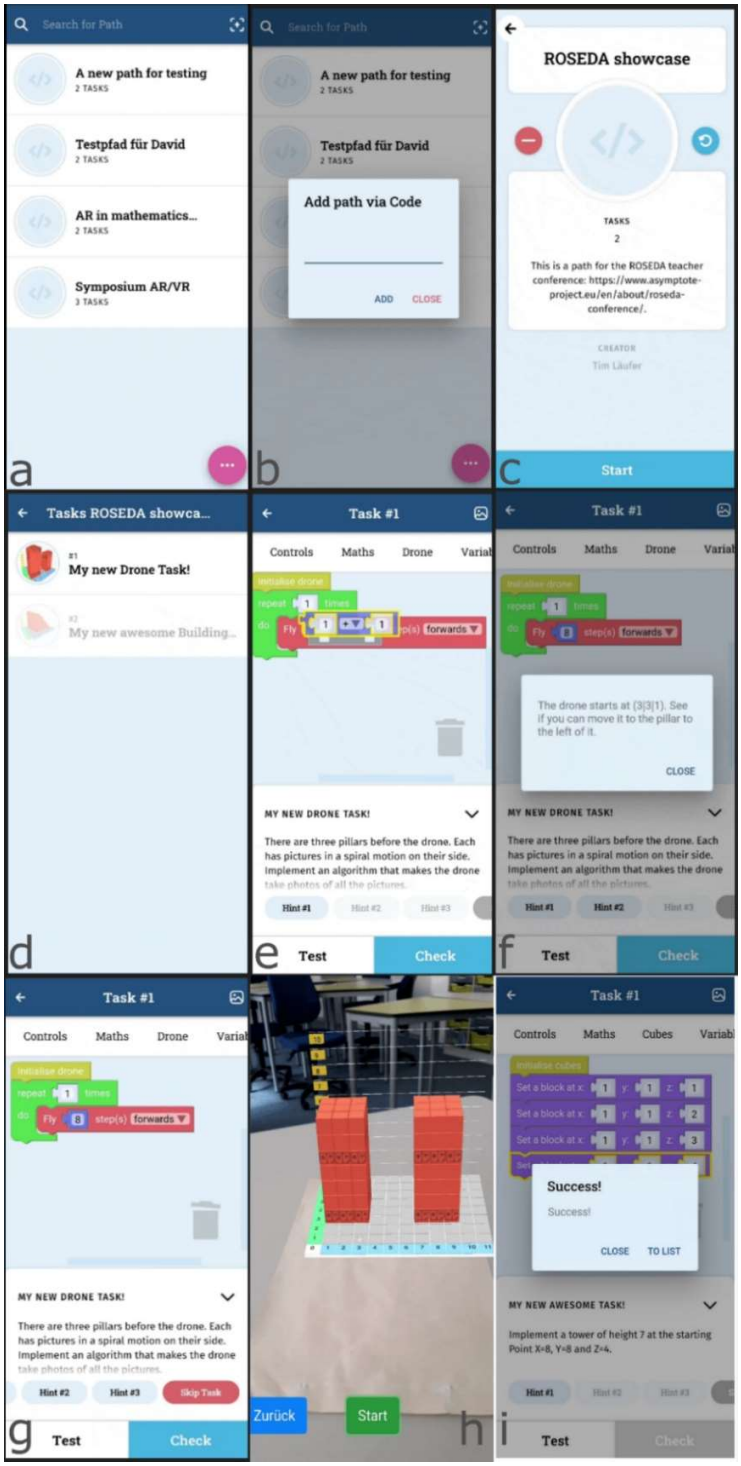


Figure 11: Viewing, solving and reviewing a task with the <colette/> smartphone app (screenshots). Upon starting the app the student can see their already added paths (a), import a new path (b) and start the path while seeing details about the path (c). They will then see the tasks within the path (d) to start working on one (e). Tiered hints will help the student to get to a solution (f) and after having seen all of the hints the student can also skip the task (g). In a AR preview the student can see an animation of the drone flying along the coded route to take pictures (h). When successful a notification is shown (i).

## OUTLOOK: THE WORKSHOP

Within the proposed workshop, the participants will get to know the <colette/> system first from the students' point of view by exploring and testing the app, then from the educators' perspective by using the web portal. They will familiarize themselves with the concept "CT" and the creation of CT fostering tasks.

After a short introduction to CT, the participants experience a path using the app. After a short feedback round they will start focusing on how to create a path from the technical perspective as well as the didactical perspective. Both will be covered in the workshop.

As preparation for the workshop, the participants should download the <colette/> app and should bring their own devices (laptop, smartphone) to the workshop, so that we can start creating a <colette/> path.

### Acknowledgment

The <colette/> project is co-funded by the European Union as part of the Erasmus+ Strategic Partnership Programme (PAD; grant no. 2020-1-DE03-KA201-077363).

### References

Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., Engelhardt, K., Kampylis, P., & Punie, Y. (2016). *Developing Computational Thinking in Compulsory Education: Implications for Policy and Practice.* Publications Office of the European Union.

Roth, J. (2015). Lernpfade – Definition, Gestaltungskriterien und Unterrichtseinsatz. In J. Roth, E. Süss-Stepancik, & H. Wiesner (Eds.), *Medienvielfalt im Mathematikunterricht* (pp. 3–25). Springer.

Stäter, R. S., Läufer, T., & Ludwig, M. (2023). 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.

Wing, J. M. (2006). Computational Thinking. *Communications of the ACM, 49*(3), 33–35.