Bence Gaál, Noémi Bernadett Agócs
Editors

Informatics in Schools

17th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives

ISSEP 2024 Local Proceedings

ELTE | FACULTY OF INFORMATICS

17th International Conference on Informatics in Schools:
Situation, Evolution, and Perspectives, ISSEP 2024
Budapest, Hungary, October 28-30, 2024

Informatics in Schools

17th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives

ISSEP 2024 Local Proceedings

Budapest, Hungary, October 28-30, 2024

Editors

Bence Gaál, Noémi Bernadett Agócs

Responsible publisher

ELTE Informatikai Kar
1117 Budapest, Pázmány Péter sétány 1/C.

ISBN

ISBN 978-963-489-746-0

Project no. 149288 has been implemented with the support provided by the Ministry of Culture and Innovation of Hungary from the National Research, Development and Innovation Fund, financed under the MEC_SZ_24 funding scheme.

ELTE | FACULTY OF INFORMATICS

NATIONAL RESEARCH, DEVELOPMENT AND INNOVATION OFFICE HUNGARY

PROGRAM FINANCED FROM THE NRDI FUND

Preface

The local proceedings contain papers and poster descriptions for presented posters at the 17th International Conference on Informatics in Schools: Situation, Evolution and Perspectives (ISSEP 2024). The conference was held at the Eötvös Loránd University, Faculty of Informatics in Hungary, Budapest, from October 28-30. This Preface and the following Organization section are closely similar to the one published in the Springer proceedings (Lecture Notes in Computer Science, vol. 15228, doi: 10.1007/ 978-3-031-73474-8 978-3-031-73473-1), they complement the proceedings published by Springer and provide a broader overview of the conference's topics.

ISSEP is a significant platform for researchers, educators, and practitioners in primary and secondary school informatics education. This conference offers a unique opportunity for the exchange of ideas and the development of innovative approaches to teaching and learning in the field of informatics. The conference has expanded and progressed since its launch in 2005 in Klagenfurt, Austria, continuing to gather professionals in cities like Vilnius (2006), Torun (2008), Zürich (2010), Bratislava (2011), Oldenburg (2013), Istanbul (2014), Ljubljana (2015), Münster (2016), Helsinki (2017), St. Petersburg (2018), Larnaca (2019), Tallinn (2020), Nijmegen (2021), Vienna (2022), and Lausanne (2023). Each conference has contributed to the development of the field of informatics education.

A Doctoral Consortium was held the day before the main conference, as was successfully done in Lausanne and Vienna in the past 2 years to support young researchers. On October 27, 2024, several doctoral students presented and discussed their research resulting in building relationships, getting constructive feedback from peers and researchers, and enriching their work with new perspectives.

On the first day of the event, local teachers participated in practical workshops and lectures as part of ISSEP-INFOÉRA teachers' conference day. This initiative fosters closer interaction between teachers and researchers, ensuring classroom relevance and providing teachers with insights into the latest developments in the field.

The ISSEP 2024 Program Committee received 42 paper submissions, each was blind reviewed by 3-5 members of the committee. 14 full papers were included in the conference publication this year corresponding to a 33% acceptance rate, and 10 more were selected for the local proceedings. The entire submission, review, and selection process was done using the EasyChair conference management system.

We wish to express our deep gratitude to the members of the Program Committee for the work they have done in reviewing the submissions and providing feedback to the authors. We would like to thank the authors for their high-quality submissions and our colleagues and the local organizing committee for managing the logistics of the physical conference. We would also like to thank Dorottya Vincze for creating the conference design and website. Finally, we thank our partners and sponsors for their generous contributions:

EPAM, the John von Neumann Computer Society, the Webdidaktika foundation and our institution, the Eötvös Loránd University.

October 2024

Bence Gaál
Noémi Bernadett Agócs

Organization

Conference Chairs

Zsuzsa Pluhár	Eötvös Loránd University, Budapest, Hungary
Ágnes Erdősné Németh	Eötvös Loránd University, Budapest, Hungary

Steering Committee

Erik Barendsen	Radboud University and Open University, the Netherlands
Andreas Bollin	University of Klagenfurt, Austria
Valentina Dagienė	Vilnius University, Lithuania
Gerald Futschek	TU Wien, Austria
Yasemin Gülbahar	Teachers College, ColumbiaUniversity, USA
Juraj Hromkovič	ETH Zurich, Switzerland
Ivan Kalaš	Comenius University, Slovakia
Mart Laanpere	Tallinn University, Estonia
Gabriel Parriaux	University of Teacher Education, Lausanne, Switzerland
Jean-Philippe Pellet	University of Teacher Education, Lausanne, Switzerland
Zsuzsa Pluhár	Eötvös Loránd University, Budapest, Hungary
Sergei Pozdniakov	Saint Petersburg Electrotechnical University, Russia

Program Committee

Andreas Bollin	University of Klagenfurt, Austria
Andrej Brodnik	University of Ljubljana, Slovenia
Špela Cerar	Faculty of Education, University of Ljubljana, Slovenia
Christian Datzko	Gymnasium Liestal, Switzerland
Monica Divitini	Norwegian University of Science and Technology, Norway
Ágnes Erdősné Németh	Eötvös Loránd University, Budapest, Hungary
Gerald Futschek	TU Wien, Austria
Bence Gaál	Eötvös Loránd University, Budapest, Hungary
Yasemin Gülbahar	Teachers College, ColumbiaUniversity, USA
Micha Hersch	University of Teacher Education, Lausanne, Switzerland
Juraj Hromkovič	ETH Zurich, Switzerland

Corinna Hörmann	Johannes-Kepler-Universität Linz
Ivan Kalaš	Comenius University, Slovakia
Kaido Kikkas	Tallinn University of Technology, Estonia
Dennis Komm	ETH Zurich, Switzerland
Martina Landman	TU Wien, Austria
Peter Larsson	University of Turku, Finland
Lukas Lehner	TU Wien, Austria
Nina Angela Lobnig	University of Klagenfurt, Austria
Birgy Lorenz	Tallinn University of Technology, Estonia
Tilman Michaeli	TU Munich, Germany
Mattia Monga	Università degli Studi di Milano, Italy
Stefan Pasterk	University of Klagenfurt, Austria
Gabriel Parriaux	University of Teacher Education, Lausanne, Switzerland
Jean-Philippe Pellet	University of Teacher Education, Lausanne, Switzerland
Zsuzsa Pluhár	Eötvös Loránd University, Budapest, Hungary
Giovanni Serafini	ETH Zurich, Switzerland
Vipul Shah	ACM India Pathshala Initiative
Jacqueline Staub	Trier University, Germany
Gabrielė Stupurienė	Vilnius University, Lithuania
Maciej Sysło	Warsaw School of Computer Science, Poland
Michael Weigend	University of Münster, Germany
Markus Wieser	University of Klagenfurt

Additional Reviewers

Smitha Balakrishanan
Ľudmila Jašková
Lisa Kuka
Karolína Miková
Michal Winczer

Table of Contents

Papers

Innovative Approaches in Teacher Training and Educational Technology

Extending the concept of MOOCs for school classrooms 3
Leon Frischauf

Professionalization for Developing Discipline-Specific Language Proficiency Among Learners as Part of a Master's Program in Computer Science Teacher Education......... 11
Jan Strobl, Fatma Batur, and Torsten Brinda

Hands-On vs. Hands-In: A Comparative Analysis of VR and Tangible Learning for Abstract Informatics Concepts 24
David Baberowski, Thiemo Leonhardt, and Nadine Bergner

ICE-T: A Multi-Faceted Concept for Teaching Machine Learning............................... 36
Hendrik Krone, Pierre Haritz, and Thomas Liebig

Understanding Teachers' Motivations for Digital Education Training: A Qualitative Study.. 48
Corinna Hörmann, Lisa Kuka, and Barbara Sabitzer

Educational Strategies and Methods for Teaching Computational Thinking

Process of Creating and Evaluating Workbooks as a Resource for the Structured Teaching of Computational Thinking at Primary School... 60
Taina Lehtimäki, Rosemary Monahan, Kevin Casey, Thomas J. Naughton, and Aidan Mooney

Place of "crafts" in the education of Algebraic and Computational Thinking................ 74
Sarmasági, Pál Sarmasági, Anikó Rumbus, Zsuzsa Pluhár, and András Margitay-Becht

Designing a Computational Thinking Intervention for Kindergarten Students 84
Tobias Bahr

AI and Robotics in Early Education: Engaging Young Learners with Emerging Technologies

Experience AI: Introducing AI/ML to Grade 6–8 students in the UK 93
Jane Waite, Ben Garside, Robert Whyte, Diana Kirby, and Sue Sentance

Children and Robotic Education Children's Persistent Belief in Humanoid Robots, or, Why is your robot crying 107
Corinna Mößlacher, Markus Wieser1, and Andreas Bollin

Poster Descriptions

"Are the Insides of a Smartphone Different from a Desktop?": Study about Preconceptions of 6th and 8th Graders Comparing Hardware Components 122
Anna Yaghobová, Anna Drobná, Marek Urban, and Cyril Brom

Card-Based Activity to Raise People's Awareness about how the Digital World Works 126
Sébastien Combéfis

Development of Interactive Database Courses Using aDBenture, H5P and Storytelling 130
Nina Lobnig, Claudia Steinberger, and Andreas Bollin

DigiFit4All - Advantages and Challenges of Competency-Based and Personalized Open Online Courses (POOCs) 134
Tatjana Angermann, Stefan Pasterk, and Andreas Bollin

DigiTeaMap: A Digital Map to Represent Addressed Competencies in Digital and Computer Science Education in Austria 138
Stefan Pasterk , Corinna Hörmann , and Andreas Bollin

Exploring Physical Computing In Schools: Designing a Longitudinal Study 142
Sue Sentance and Jessie Durk

Knowledge of Cookies and Personalized Ads among Lower Secondary Students: Effects of a Simple Treatment 146
Cyril Brom, Anna Yaghobová, Anna Drobná, Daniel Šťastný, and Marek Urban

Novice Primary School Teachers' Conceptions of Internet Structure: A Qualitative Analysis 150
Anna Drobná, Anna Yaghobová, David Šosvald, Daniel Šťastný, Marek Urban and Cyril Brom

PRIMMDebug: A Tool and Process for Teaching Text-Based Debugging to Beginner Programmers 154
Laurie Gale and Sue Sentence

Towards Generating Bebras Tasks Using Large Language Models: What Criteria to Use for Evaluation? 158
Mohsen Asgari, Filip Strömbäck, and Linda Mannila

Fostering AI Literacy in Primary Education? 162
Gabrielė Stupurienė and Gintarė Perednytė

Papers

Innovative Approaches in Teacher Training and Educational Technology

Extending the concept of MOOCs for school classrooms

Leon Frischauf[0000−0001−7817−7883]

TU Wien, Institute of Information Systems Engineering, Vienna, Austria
`leon.frischauf@tuwien.ac.at`

Abstract. MOOCs (massive open online courses) enable students to study at their own pace, accessing high-quality education regardless of location or financial constraints. They offer flexibility to balance education with other commitments while integrating interactive elements and diverse learning resources.
However, the focus often solely lies on the student's perspective, neglecting the role of teachers in the learning process. Students independently "learn" without teachers having the opportunity to intervene or monitor their progress. This renders MOOCs often impractical in conventional classroom settings (indicating that they are not widely used in Austrian or German K-12 classrooms). Previous research has acknowledged this issue and frequently suggests theoretical solutions but has not offered a practical solution yet.
To bridge this gap, this paper proposes a solution that combines the benefits of MOOCs with the advantages of traditional schoolbooks, aiming to create a learning environment that incorporates the best of both worlds. This solution was developed through a comprehensive literature survey, which explored the potential applications of MOOCs in classroom settings and identified currently missing features.
The identified use cases were then mapped to specific features and compared with standard MOOC offerings, forming the foundation of the proposed solution.

Keywords: MOOCs · digital schoolbooks · digital teaching
MOOC supported digital teaching

1 Introduction

Schools are integrating more and more digital platforms into their lessons, ranging from quiz apps to interactive simulations. The variety of available digital tools is vast, making platforms that give an overview of all the possibilities essential (see [5]).

While it's great to see digital devices have finally made their way into the classroom, there is often a lack of a holistic strategy for integrating these tools effectively. Teachers and students frequently register on numerous platforms, leading to confusion about which tools to use and when. Additionally, the sheer number of available tools can obscure their pedagogical value. Teachers must invest significant time in evaluating and selecting appropriate tools for their students.

As universities have encountered similar challenges, they have developed a partial solution through the creation of MOOCs (Massive Open Online Courses). These courses, designed by experts, integrate various elements like videos and interactive assets, all vetted by the course creators. This ensures that users can rely on the MOOC's quality and use it as a holistic teaching resource.

However, MOOCs cannot be really well integrated into school classrooms for several reasons. They often lack features essential for classroom teaching - such as tools for fostering teacher-student relationships and effective communication between students and teachers ([1] and [7]).

This paper aims to address this gap by proposing a set of user scenarios that school-focused MOOCs need to support (partly based on a literature review).

2 The role of MOOCs in different settings

The predominant form of MOOCs today are content-focused xMOOCs, which allow massive participation and often free access to high-quality education. Coursera, edX, or OpenLearning are three prominent examples of such platforms (see [1]). However, they often lack "social learning or interaction," diverging from the original connectivist ethos of MOOCs (see [2, p.427]). Due to the large number of participants, it is merely not possible to foster student-teacher relations, as "two or three teachers cannot be expected to meet the learning demands of hundreds and even thousands of participants" [1, p. 70]. While this is less of a problem in universities due to the greater independence of the students, it poses a significant challenge in a school setting. Even in the university setting, dropout rates range from 75% to 95% - for non-obligatory courses in a school setting, this would even be higher.

Typical teacher-student interactions in a university setting MOOC have been examined by [4], who list various learning scenarios involving MOOCs in classroom settings. For example, they describe a blended MOOC, which incorporates the blended learning approach, and a flipped MOOC, which includes the "flipped classroom" concept.

The challenge with classroom teaching is that, to keep students motivated, teaching methods need to be flexible and adapted to the topic taught.

According to [7], most MOOC platforms perform well when looking at learner-system interactivity and learner-content interactivity. This is especially important in university-style learning. However, they lack when it comes to learner-learner and learner-instructor interactions. These interactions are key when it comes to incorporating a MOOC into a classroom setting. This statement is also further validated by a statement in [1]: David Chernoff, a professor who taught Cornell University's first MOOC on the edX platform, stated that MOOC platforms lack sufficient technological tools for synchronous and immediate interaction. They do not support face-to-face interaction between teachers and students, and there is also a lack of timely feedback on course activities.

3 Which user scenarios are important to digitize today's school classrooms?

Today's digital platforms typically focus on digitizing specific elements, such as the process of completing homework online (user scenario), rather than replacing traditional textbooks entirely (what a MOOC would do). Consequently, teachers see the advantages of integrating these systems with conventional teaching methods instead of completely changing the traditional setup.

[12] confirms this hypothesis and states that, e.g., the use of digital technologies has been heavily focused on their application in exams. This would be one use case of such an asynchronous system (LMS), which, e.g., allows collecting homework or automatically correcting revisions. In parallel, teachers need a synchronous communication system for student-teacher communication.

In order to be able to introduce fully digital systems to classrooms, one has to look at the current status quo - which platforms with which features are used in today's classrooms.

[5] has analyzed different platforms frequently being used in Austrian classrooms. The following user scenarios (US) could be identified.

- **US1: Student process becomes visible for the teacher**: The teacher gets information about the solving process of the students. This includes, e.g., which examples a student solves, whether he has solved the example correctly or not, and details about the solution of the student. Some platforms even allow the upload of the students' full handwritten solutions. There has to be an easy way for teachers to look into student's solutions. One possible way is to list all the student's solutions aggregated on an example level. Then, the teacher will see the specific examples and the students' solutions connected with them.

- **US2: Student's process is analyzed on competency level (like in the curriculum)**: Student performance shall be evaluated based on competency levels as outlined in the curriculum. Modern curricula are often competency-based, and standardized tests such as the iKM Plus [3, p.83-100] and the Austrian final exams are also structured around a framework of competencies. In 2018/19, the Austrian information technology curriculum was updated to align with this approach, now describing its goals as competencies [9].
  Consequently, the presentation of a student's strengths and weaknesses should align with the competency structure defined by the curriculum.

- **US3: Adaptive algorithm that adapts to the strengths and weaknesses of the students**. Modern (and leading) platforms do not show each student the same content or examples but base this on competence information. [8] proposes an adaptive algorithm that selects content for learners after a pre-test. This ensures students receive material that matches their skill level, avoiding too difficult tasks. By applying theories like Ausubel's Meaningful Learning Theory, the algorithm optimizes the selection of examples to enhance learning outcomes.

- **US4: Organizational features such as class management, electronic class registers, etc.** Students shall be able to focus on learning - and teachers, therefore, not on organizational tasks. The platform, therefore, has to be easy to use and fit into existing communication systems such as Microsoft Teams or Google Classroom (e.g., speak with their API). The platform shall support standards such as LTI to export the results to another platform.

Additionally, authors of [6] have examined these scenarios more from a user-centric perspective for a platform often used in Austrian classrooms. Figure 1 illustrates the relative importance of different scenarios.

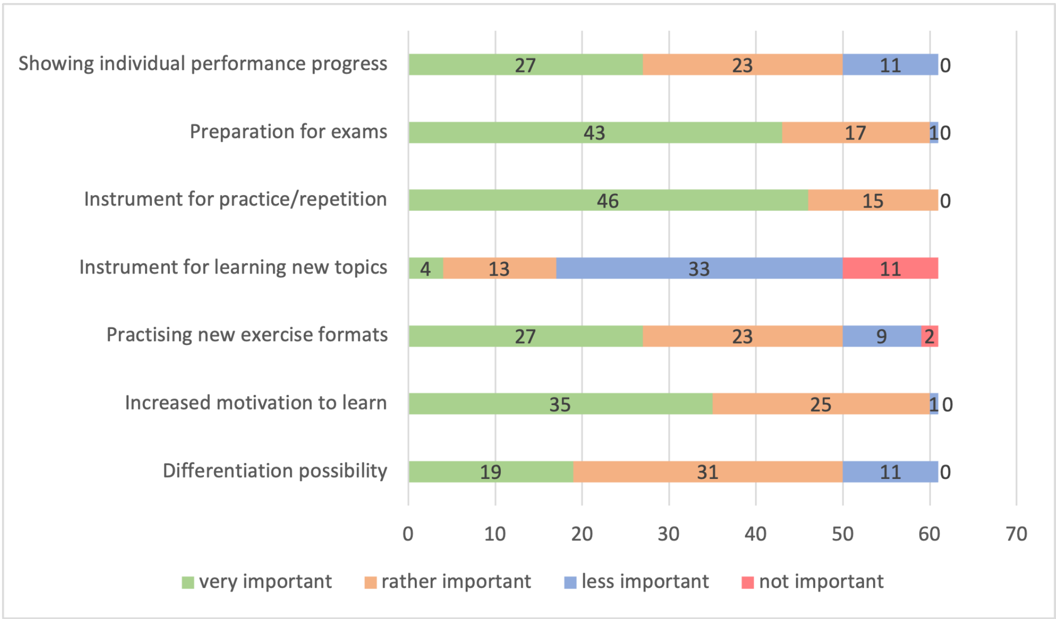


Fig. 1. Importance of different features for teachers [6]

[5] and [6] continue stating the importance of time-saving features for teachers and the benefits they provide to students. Teachers shall not have additional

efforts when introducing such tools into the classroom. Also, in order to be widely used, digital tools have to adapt to existing teaching methods.

4 Future vision

4.1 Extending the concept of MOOCs to be integrated into a classroom setting

From the previous chapter, one sees that the tools are used in addition to traditional teaching methods, but not as a replacement of them. In schools, this means that the conventional textbook remains the primary learning resource.

In contrast, university learning has already begun to integrate digital elements more extensively. Some courses are held entirely online.

[11] describes various learning settings for MOOC participants in learner-instructor environments. They discuss behaviors such as watching instructional videos, submitting homework and tests, and interacting in discussion forums.

While these points are all valid, simply integrating a traditional MOOC into a classroom setting will not work out. The lesson design in universities differs significantly from that in schools - therefore another approach is needed.

This claim was also validated by [10, p.850]. In this paper, a typical classroom setting (traditional face-to-face) was compared with a MOOC, and students were asked about what they experienced with each of the two scenarios. Significant differences were observed in terms of help-seeking abilities between the two scenarios.

The authors suggest that MOOC providers shall "explicitly scaffold students' learning." The authors continue by suggesting that future MOOC courses should focus on highlighting help-seeking avenues and improving the quality of the feedback from instructors and peers as they work on problem-solving. Addressing help seeking at the student or course platform level might help high school students to transition from face-to-face to MOOC computer science courses, thus allowing a more significant number of students in more diverse settings to pursue computer science. ".

This is precisely the direction in which the following user scenarios go.

4.2 User scenarios

This paragraph will conceptually discuss the various elements required for extending the concept of MOOCs for a classroom settings. It will consider the transition process, including incorporating features from existing tools and aligning them with traditional learning scenarios. Note that some concepts of MOOCs does not translate 100 % into classroom settings (such as the "openness concept").

As a teacher, I want to present specific parts of the MOOC to my students. In a teacher variant, the larger screen shall support the teaching in the classroom. One could highlight interactive elements breaking the concept of teacher-centred instruction. There shall be a special version optimized for the larger screen (e.g. showing interactive simulations).

As a teacher, I want to send my students to a specific position on the MOOC. The students can be provided with jump marks that signal to them that special attention is required at this point or that specific material is now available (see Figure 2). This solves the problem that if the teacher wants to discuss a specific topic, the students no longer have to navigate to the specific page but get a notification directly inside the MOOC. A specifically important aspect is the granularity of this approach, as the students would still need to scroll to the specific element otherwise. Therefore, the teacher should be able to set jump marks for an interactive asset, videos, text elements, etc.

As a teacher, I want to add my own content to the MOOC or want to modify existing content. Teachers should be able to modify the MOOC structure to make the MOOC their own. The same is done today with schoolbooks, where teachers only cover parts of the curriculum and not everything. Consequently, teachers should be able to:

- hides certain elements of the MOOC if they prefer alternative explanations. Although these hidden elements will not be visible to students, they will be marked with a symbol indicating their availability, maintaining the integrity of the original content. This feature ensures that students still have access to the entire content if needed while allowing teachers to modify the workflow without disrupting the curriculum designed by the original author.
- add their own elements to the MOOC: they should have the freedom to add their own content to the MOOC. These additions should be clearly distinguished from the original content, for instance, by being labeled as teacher-created elements. This shall allow them to add their own texts, videos, and problems. Intriguingly, teachers often use text elements not only for theoretical material but also for providing instructions or work orders. To accommodate this, there should be a "draft state" option, indicating that a specific element is only visible to the teacher and not the students.

When students solve exercises inside the MOOC, I wish to get immediate feedback. Whenever students solve exercises inside the MOOC, information about the progress of these exercises needs to be available to the teacher. The teacher should be able to see which students have weaknesses and help those. Each chapter should be aligned with specific competencies outlined in the curriculum, and student's progress should be tracked in real time. This way, teachers can see where students stand in relation to these competencies at any given moment. The exercises students complete should contribute to an overall competency score, which updates dynamically.

Teachers should have a live dashboard displaying these scores, allowing them to promptly identify and support students facing difficulties. This is crucial because students frequently do not proactively inform their teachers about any misunderstandings.

As a teacher, I want to select specific examples from a MOOC as homework for my students. There is a necessity for teachers to mark specific elements to be completed at home. This element is completely absent e.g. from xMOOCs as students are learning at their own pace. While this has apparent advantages for non-obligatory subjects, this is not the case in a school setting.

Additional features.

- Own notes: Students and teachers can add their own notes to each chapter. These notes can be shared with their teacher to indicate, e.g., something they did not understand when learning it at home.
- Continue where left: When exiting from the MOOC, an invisible progress marker is automatically set (for the teacher and pupils) to enable re-entry at the same point. If the jump marker is somewhere else, there is a prompt at the top to jump to the teacher's position.

5 Future steps and outlook

As a next step, the goal is to create a MOOC platform prototype that fulfills the requirements listed in the fourth chapter. For this development process, we will build the platform, starting with the content.

Finally, we will conduct a mixed-methods pilot application, involving both qualitative and quantitative studies with students and teachers. Initial results are very promising (to be published in a future paper), indicating that students enjoy working with the digitized version. The interactivity of the platform is particularly advantageous when compared to traditional schoolbooks.

References

1. Atiaja, L., Guerrero-Proenza, R.S.: The MOOCs: origin, characterization, principal problems and challenges in Higher Education. Journal of E-Learning and Knowledge Society **12**, 65–76 (02 2016). https://doi.org/10.20368/1971-8829/1093
2. Baturay, M.H.: An overview of the world of MOOCs. Procedia - Social and Behavioral Sciences **174**, 427–433 (02 2015). https://doi.org/10.1016/j.sbspro.2015.01.685
3. Bundesinstitut für Bildungsforschung, I..E.d.S.: Kompetenzorientierter Unterricht in Theorie und Praxis (2011), accessed on June. 3, 2024. URL: `https://www.bifie.at/system/files/dl/bist_vs_sek1_kompetenzorientierter_unterricht_2011-03-23.pdf`
4. Ebner, M., Schön, S., Braun, C.: More Than a MOOC—Seven Learning and Teaching Scenarios to Use MOOCs in Higher Education and Beyond, pp. 75–87 (01 2020). https://doi.org/10.1007/978-981-15-0618-5$_5$
5. Frischauf, L.: Design process of the learning app Studyly for use in Austrian informatics school classes. Master's thesis, Technische Universität Wien (2023)
6. Gabriel, S., Wallner, J., Überacker, G.: Evaluierung einer Mathematiklernplattform aus Sicht von Lehrenden und Lernenden (final results not published as of date June 2024). Accessed on June. 3, 2024. URL: `https://www.forschungslandkarte.at/studyly/`
7. Gamage, D., Perera, I., Fernando, S.: Moocs lack interactivity and collaborativeness: evaluating MOOC platforms. Int. J. Eng. Pedagog. **10**(2), 94–111 (2020)
8. Kardan, A.A., Ebrahim, M.A., Imani, M.B.: A new personalized learning path generation method: Aco-map. Indian Journal of Scientific Research **5**(1), 17–24 (2014)
9. Micheuz, P., Schwarz, G.: Aktueller Lehrplan INF 5. Klasse AHS - gültig ab 2017/2018. Accessed on June. 3, 2024. URL: `https://www.ahs-informatik.com/informatik-5-klasse/`
10. Sands, P., Yadav, A.: Self-regulation for high school learners in a MOOC computer science course. In: Proceedings of the 51st ACM technical symposium on computer science education. pp. 845–851 (2020)
11. Tian, Y., Sun, Y., Zhang, L., Qi, W., et al.: Research on MOOC teaching mode in higher education based on deep learning. Computational intelligence and neuroscience **2022** (2022)
12. Weigand, H.G.: Was lehrt uns das "Lernen zuhause" im Hinblick auf den (zukünftigen) Einsatz digitaler Technologien im Mathematikunterricht? Mitteilungen der Gesellschaft für Didaktik der Mathematik (109), 63–67 (2020)

Professionalization for Developing Discipline-Specific Language Proficiency Among Learners as Part of a Master's Program in Computer Science Teacher Education

Jan Strobl[0009−0003−3965−8408], Fatma Batur[0000−0002−7377−6585], and Torsten Brinda[0000−0001−6137−4258]

Computing Education Research Group, University of Duisburg-Essen,
45127 Essen, Germany
`https://www.ddi.wiwi.uni-due.de/en/`
{jan.strobl,fatma.batur,torsten.brinda}@uni-due.de

Abstract. The importance of discipline-specific language learning in all subjects is now widely recognized. The fact that computer science has been and is being introduced as a compulsory subject for all students of all types of schools in many places makes it even more important to take the linguistic heterogeneity of the student population into account when teaching computer science. Various core curricula, including for computer science, therefore stipulate that language education and multilingualism must be taken into account. In order to prepare prospective computer science teachers for this task, we have developed a course on this topic in our master's degree program. This course enables students to identify linguistic challenges in computer science lessons and to implement language education approaches in their class, with a particular focus on developing writing and reading skills. The course has been successfully offered several times in recent years. This article describes the design of the course, initial experiences and resulting further integration of the topic into our study program.

Keywords: computer science, language learning, teacher training

1 Introduction

In recent decades, it has been recognized that, in addition to general language skills, discipline-specific language proficiency is a key factor for academic success in all subjects. Although the importance of taking different language backgrounds into account for educational equity already existed, migration and flight have increased public interest in the topic in recent years, as well as the urgency.

As computer science (CS) develops from an elective subject for interested students, often with prior knowledge, to a compulsory subject for everyone, future teachers need to be even better qualified to deal with linguistic heterogeneity. Digital literacy and language skills are interdependent and influential key skills for participation in today's society [8]. To address this problem, the curricula of various school subjects, including CS, provide for the consideration of language education and multilingualism, but subject-specific professionalization of prospective teachers for this task is not offered for all subjects during their studies. For this reason, we have developed and implemented a course for CS student teachers in the master's program to prepare them on how to recognize the linguistic challenges of CS teaching and apply language-sensitive concepts to overcome them, with the aim of giving their learners the necessary language skills for success in CS.

In the context of this paper, *language proficiency* refers to the ability to use natural language in written and oral form to describe, explain or discuss subject content. Unless explicitly stated otherwise, this does not refer to formal or programming languages or other artificial domain-specific languages in the narrower sense.

The rest of the paper is structured as follows. In section 2, we provide the theoretical background of this work, justify the need for language-sensitive teaching of subjects and discuss preliminary work in the context of CS education (CSE). In section 3, we describe the curricular integration of the developed course and in section 4 its structure and selected content. In section 5, we report first experiences running the course and in section 6, we briefly describe further integration of the topic into our study program. The paper ends with conclusions and outlook in section 7.

2 Theoretical Background

2.1 Language Education and Educational Inequality

Teaching in heterogeneous groups, also with regard to language proficiency, is part of every teacher's daily business in many parts of the world (including the Ruhr area in Germany, where the university of the authors of this work is located). Beginning with addressing students who speak German as a second language (in German: *DaZ – Deutsch als Zweitsprache*), language pedagogy research has long called for language-sensitive [18] or academic-language-promoting [11] design of subject lessons in order to overcome known educational inequalities between students with migration backgrounds and so-called monolingual German-speaking students, as regularly shown by studies such as TIMSS[1] and PISA[2]. In addition to the goal of creating a better learning environment for students who speak German as a second language, there is now a consensus that language-sensitive subject teaching is beneficial to learning for all students [20].

Language support in all subjects, including first language support for second language learners, is essential for educational success [10, 29]. If learners' language practice is not sufficiently taken into account, this can result in avoidable disadvantages for all students who did not already practice the required educational language outside school [5, 15].

---

[1] `https://timssandpirls.bc.edu/`

[2] `https://www.oecd.org/pisa/`

The language style required at school and later in educational and academic contexts is usually referred to as CALP (Cognitive Academic Language Proficiency) [6] and is to be distinguished from the language used in private contexts, which is referred to as BICS (Basic Interpersonal Communication Skills). CALP often only develops if it is explicitly learned by students, as they do not automatically acquire it from everyday language skills [8]. Teachers need to explicitly address this in class and need to be trained to do so.

The need for uniform and consistent teaching of discipline-specific language in subject lessons is now widely accepted not only in science, but also at the political and administrative level and has found its way into guidelines and specifications, for example through core curricula for schools.

Most current curricula in North Rhine-Westphalia (NRW) - including that for the compulsory elective subject of CS in grades 9 and 10 [19, p. 8] - explicitly stipulate the consideration of language-sensitive subject teaching, consistent language education and the inclusion of multilingualism, which results in the need to qualify (prospective) teachers for this task. For other STEM subjects, especially – but not only – mathematics, it has been explicitly shown that language skills also have a considerable influence on educational success in the subject in Germany [14, 21] and there are already subject-specific pedagogic concepts for language learning [3, 28] as well as concepts for teacher training [24].

2.2 Research on Language Learning in CS

Teacher training in CS cannot yet draw on such extensive preliminary work. However, it can draw on internationally established concepts for all subjects, such as scaffolding [10] or the genre-based approach [22], which have already been adapted for the German language [13, 16], and apply them to CS lessons, while also benefiting from preliminary work in related subjects.

Nevertheless, the linguistic requirements of CS lessons and suitable concepts for teaching them still need to be systematically developed and evaluated. In NRW, this need is reinforced by the recent introduction of CS as a compulsory subject in all secondary schools in grades 5 and 6. A compulsory subject leads to a significantly more heterogeneous student body than an elective subject, because such a subject is often chosen by students who are particularly interested or have previous knowledge.

So far, there are only a few publications on language education or (natural) language in general in CS lessons, most of them focusing on very specific aspects. In the following, we briefly describe those works that have been regarded to be relevant for our work.

Diethelm et al. [8] provide the most comprehensive presentation of the fundamental necessity of language education in CS classrooms, including the identification of concrete linguistic challenges and possible solutions. The authors highlight that CS (like every subject) “has its own, additional, variety of language, consisting of a specific terminology, specific frequent syntactic constructions, specific semantic conventions and specific communicative routines” [8, p. 210]. On a terminological level, the authors identify multiple meanings of terms in everyday and professional contexts. In particular, they point out that the use of English terms in a non-English-speaking environment can also have advantages, as it is easier to identify them as subject-specific terminology. They also discuss the importance and opportunities of metaphors, but at the same time point out the particular linguistic challenges and risks of developing misconceptions. Central recommendations include enabling meta-discourses in class about the meaning of terms, phrases and metaphors in everyday and professional contexts, empowering prospective teachers to formulate explanations in different ways but establishing consensus among teachers and schools on the meaning of terms.

The role of metaphors is also discussed by Saathoff et al. [23], who analyze interviews with 23 students to determine which analogies or metaphors they know and use to understand and explain CS concepts. Lampe and Diethelm [17] analyze a transcript of a 10^{th} grade lesson aimed at teaching terms and concepts of data encapsulation and visibility in the context of object-oriented programming (OOP). Among other things, this reveals the teacher's sometimes contradictory use of terms, who is therefore rather unsuitable as a language model, the very low proportion of students' speech and the almost complete lack of students' own use of CS terminology. Batur and Strobl [2] emphasize the importance of text work which increases in higher grades. They demonstrate the application of a genre-based approach to writing in subject lessons using the example of describing class diagrams.

Sentance and Waite [26] focus on oral communication in programming lessons, analyzing the teachers' perspective on the basis of 20 interviews. Among other things, the study examines how teachers use student-teacher dialogues between students and with the teacher as well as explanations in the context of a PRIMM programming lesson. Even though the interviewees do not explicitly address how the students acquire the language skills needed for the discourse of the content, some conclusions can also be drawn with regard to language learning. The authors "have noted that teachers report that students do not find it easy to explain how a program works, or to use a range of linguistic tools to verbalise their reasoning. Programming teachers are not generally trained to facilitate productive dialogue in our experience. Furthermore, many teachers in our study reported delivering whole-class explanations rather than focusing on the ways in which learners could improve their own explanations, [...]" [26, p. 278]. Most authors state that further research is needed on language education in CS lessons and that there is also a need for teacher training and continuing professional development.

3 Teacher Training Program and Language Learning

Teachers for elementary and secondary schools are trained at our university in NRW. Upon completion of a six-semester bachelor's degree, and a four-semester master's program, candidates undergo an 18-month preparatory service (teacher traineeship, in German: *Referendariat*). The university program requires students to combine two subjects as well as mandatory studies in educational sciences, which encompass pedagogy and psychology. Furthermore, practical training is an integral component of the curriculum, which includes a practical semester at a school (in German: *Praxissemester*) in the second semester of the master's program. To complete the degrees, students must write a bachelor and a master thesis.

In NRW, all student teachers must take courses focusing on German for learners with a migration background (in German: *Deutsch für Schülerinnen und Schüler mit Zuwanderungsgeschichte (DSSZ module)*) with at least 6 credit points (CP = ECTS credit) to acquire basic skills for working in linguistically heterogeneous classes and to show them ways of teaching in a language-sensitive way. Evaluations show that these courses arouse interest and raise awareness of the topic, but they can only impart basic knowledge [7]. In particular, students would like to see subject-specific training on the topic of language learning offered in their subject departments [9]. Authors from other federal states [12] also argue in favor of an integrated model in which language learning is offered in the subjects. Corresponding courses already existed in some subjects and also students who have acquired the relevant skills by choosing relevant courses and topics for examinations. However, this was not always visible in the final certificate. To systematize the range of courses and certify the qualification, an additional certificate for language learning (*ZuS*, in more detail in [27]) has been implemented at our university. In order to successfully complete it, courses with a focus on language education, migration and multilingualism must be taken within the regular course of study in the educational sciences (area B) and in the subjects (area C) or thematically relevant final theses must be written. In addition, relevant courses must be chosen to accompany the practical phases (area D). The basic idea of ZuS is to initiate the development of a profile of skills in the field of language education and multilingualism by using the available options in the normal course of study and to document a corresponding focus. Only one additional language course (area A) in a previously unlearned language and the final examination must always be completed on-top.[3]

The CS teacher education program includes both subject-specific and subject-pedagogical modules. Pedagogical modules focus on techniques, methods and approaches for designing teaching-learning situations in CSE as well as on scientific findings on teaching and learning within the field. The bachelor's degree program focuses on subject-specific content for all CS students. Additional courses can be chosen in the master's degree program. The first subject-pedagogical course is located in the 5th and 6th bachelor semesters. In the mandatory subject-pedagogical master course, subject-related pedagogical topics are further expanded upon. In addition to a subject-specific course accompanying the internship semester at school (2nd master semester), students can take various subject-pedagogical elective courses. The course "Language Education in CS Classes" described in the following section is one of the options and can also be credited towards area C of *ZuS*.

---

[3] Translation of a visualization of the structure of *ZuS* and its areas in Appendix A.

4 The Course: Language Education in CS Classes

4.1 General Structure of the Course

The aim of the course is to teach the subject-specific implementation of language education, which is regarded as a necessary supplement to the foundational DSSZ module (see 3). It enables prospective teachers to plan, teach and reflect on the language of instruction and to take on their role as language role models. This task arises not only but also from recent core curricula (see 2.1) and CSE-related publications on language education emphasize the need for teacher professional development on this topic (see 2.2). The one-semester course is regularly held with two hours per week in presence. It is supported by a Moodle course in which course material, literature and tasks are provided and through which students regularly submit their work (e.g., application of learned concepts, analyses of materials) during the semester. For examination, the results of the semester tasks and reflections on them are summarized in a portfolio, whereby a self-selected focus topic is further deepened and developed. The students present this focus individually at a colloquium at the end of the semester.

4.2 Exemplary Topics of the Course

The central goals of the course are to create an awareness of the linguistic challenges of the subject, knowledge of concrete linguistic peculiarities of the technical language, an understanding of conceptual spoken and written language as well as knowledge of strategies for the development of conceptual written language, the handling of texts and types of texts as well as the systematic and gradual development of the writing and reading skills that their future students need, for example to be able to deal with complex texts in final exams.

A central approach in the course is that the students first experience or recognize the essential aspects themselves through discovery or research-based learning before they receive theoretical input and application tasks based on it.

Writing and reflecting subject-specific texts in a foreign language. The course usually starts with a surprising writing exercise using a foreign language learned at school – in Germany in most cases English. In this task, explanatory or argumentative texts should be used, whereby these should pose a certain challenge both on a technical and linguistic level. For instance, the task could be to explain asymmetric encryption in a way that is understandable to a general audience and to justify why encryption is important. This task can be performed on the basis of non-verbal algorithm visualisations such as IDEA[4]. After having written the text, the students reflect on and discuss which aspects of writing the text were particularly difficult and what linguistic support would have helped.

It is desirable that the students realize that the obvious technical terms are not the major issue. In a variant of this task, the central technical terms can

---

[4] `https://idea-instructions.com/`

also be given in the foreign language. However, if English is the target language of the task, then this is more relevant in other subjects than in CS, as many English-language technical terms of the CS field are also used in everyday language. Challenges that are often identified in addition to technical terms are how to get into the writing process, how to structure it, how to connect certain steps, sections or events linguistically, the challenge of simultaneously recognizing, thinking through and verbalizing technical aspects in the illustration.

Analyzing School Books. In order to gain a broader overview of the language challenges and linguistic peculiarities of the subject of CS, students analyze authentic textbook passages. Central questions are whether and how new vocabulary is introduced, which grammatical structures occur frequently, what function they have and whether this becomes visible, which types of texts (genres) have to be read, how the texts are linked and which reference structures exist within and between the texts, whereby so-called discontinuous texts such as tables, diagrams and CS-specific presentations should also be included.

With regard to the tasks within the books, it should be examined to what extent it is clear what results should be produced in what form, whether and what kind of texts should be written and whether there are model texts for this purpose that can be used as a starting point. Each participant works on a different textbook and presents the results in plenary, which are then discussed. This gives all students a broad insight.

Conceptual Written and Oral Communication. An important aspect of language education is the development of conceptually written communication skills, which is a literal translation of the German term “Konzeptuelle Schriftlichkeit” and which is closely related to the term CALP commonly used in English-speaking countries. Important characteristics of such communication are the independence from a temporal or spatial context, an abstraction from a concrete situation and the use of monologue and impersonal expressions.

CS Unplugged. A great way to discuss this issue in the context of CS is in an experimental way using CS Unplugged [4]. We combine a CS Unplugged activity on sorting algorithms[5] with clear communicative roles and language observation [24, p. 46]. The students are provided with a number of identically looking but differently weighted objects (e.g., film cans) and a mechanical scale that can only compare weights but not determine them absolutely. Students then have to put the objects in the correct order according to their weight. Two or three students discuss the procedure and instruct a student to carry it out exactly. This student does not take part in the discussion. One or two other students record the spoken word verbatim. This can be supported by audio recording. In a variant of the task for larger groups, the students, who decide what to do, are not in the same room but are connected to the student

---

[5] `https://classic.csunplugged.org/activities/sorting-algorithms/`

carrying out the task by telephone or even just by text chat. If there are not enough people present, a subsequent task can be a thought experiment on how communication would have changed as a result. To reflect on the task and the communication used, students are asked to identify the search algorithm used and to find a description of it that is as generic as possible. In some cases, the students were unable to find a generally valid, objective description and considered a description generated using generative AI to be more suitable. This description is now compared to the transcripts created. Statements are arranged on a spectrum from *verbal - concrete - context-bound* such as “put it there” or “if it is lighter, it goes to the left” to *written - abstract - context-independent* such as “if the element is lighter than the pivot element, it is sorted into the first row”. This scheme is inspired by Kniffka and Neuer [16, p. 46], who demonstrate this for the introduction to magnetism experimenting with compasses. Finally, the students create small tasks in order to reach the next more abstract written expression.

CS Taboo. Another variant is to use the CS version of the game *Taboo*. Here, several pairs compete against each other, with the aim of explaining as many terms as possible to their partner in a given time, whereby the use of usually four other related terms is prohibited (taboo). Language observation is also used here. The task is well suited to sensitizing students to how difficult it is to explain terms in a clear way. It is also noticeable how few missing words are enough to make communication very difficult. When analyzing the transcripts, it is often apparent that the students initially use definitions that are as correct as possible, but later refer to shared experiences (“we did this in the exercise in the second semester”) or socio-cultural knowledge.

Linguistic Characteristics and Challenges in CS Lessons. Based on the tasks and activities described above, the students then develop an overview of the linguistic peculiarities and challenges of language in CS lessons. The aim is to identify the challenges at word, sentence and text level. They are supported by comparable overviews from other subjects, which they can use to check whether comparable phenomena also exist in CS, if these have not been found already in their own considerations.

It is noticeable here that in the jargon of CS, even at school level, a number of difficulties have to be overcome that cannot be found in any other subject. In the case of identifiers in source code, rules such as the capitalization of nouns in German are set aside. In addition, there are conventions on capitalization that should be observed but do not lead to incorrectness if they are not observed, such as camel-case syntax in Java. Furthermore, the semantics of identifiers have no effect and can theoretically be completely meaningless as long as they are consistent. In extreme cases, the name of a thing can be the opposite of what it is, for example a `hero` class can be the superclass of a `superhero` class.

So called “false friends”, which refers to words or phrases that are familiar from everyday life but have a different or more precise meaning in the subject, play a particularly important role in CS. Examples have also been found where the meaning known from the CS context is the more common one and a supposedly simple real-life example can lead to comprehension difficulties. In one book, for example, databases are introduced using the example of containers in a port, where the text contains the following segment - literally translated: "containers will be deleted". What this actually means is that sea containers are unloaded (the German translation for "to delete" is "löschen", which in the logistics context also has the less well-known meaning of "to unload"). When analyzing natural language texts in the context of modeling, for example by using the Abbott Textual Analysis [1], several technical meanings from different contexts may be necessary at the same time. For instance, when discussing the text itself, attributes can be identified from a linguistic perspective, but these are not necessarily attributes as defined in object-oriented modeling.

Reading and Writing in CS Lessons. By analyzing school-leaving exams, the students realize that they are extremely text-heavy in CS. This allows them to recognize that their future students need to be proficient in systematic reading comprehension of texts. Similarly, writing of longer, coherent texts, as required in exams, must be taught gradually and progressively over a long period of time.

In the course, students therefore learn approaches for the receptive and productive use of different genres and apply these independently. The focus here is on multi-step methods for reading and the genre-based approach to writing.

In the latter, the process of writing specific kinds of texts is always considered in relation to the social context and certain roles (who writes for whom and for what purpose). In addition, the starting point is not a blank page but a model text, which is first deconstructed and further developed with teacher support before the students write their own texts. The approach is quite similar to concepts from CSE such as Use-Modify-Create and PRIMM. In order to develop corresponding lesson plans, students must first develop model texts that are as generic as possible. It is often noticeable that they first have to reach a consensus on how a certain type of text is actually designed and what it must contain - an absolute necessity in order to be able to communicate these requirements transparently to their own students in the future.

5 First Experiences

Based on the course runs to date with usually no more than ten students, it can be said that it has been successful in raising awareness of the topic among students and providing them with tools to consider language in the classroom. Students surveyed stated that they now have a different and much more conscious view of language requirements and are aware of possible solutions or at least know where and what approaches to look for. In addition, students expressed a desire for ready-made teaching materials, which are yet available only to a limited extend. Due to the large time gap to the foundational DSSZ module (see 3) in the bachelor's program on the one hand, but elective options in the other subject or in the educational sciences for language learning on the other, there is often a very diverse level of prior knowledge at the beginning of the semester. Students, who are already completing or have completed their teaching internships at school, often have a special awareness of the need for language learning and can also report on approaches implemented in their schools. Several students decided to continue working on the topic in the remaining part of their studies. Among other things, topics from CS were chosen as a project for the final examination of the additional qualification (see 3), portfolio focuses were further elaborated and published as research papers (e.g., [25]) or related theses were written.

6 Further Integration

As a result of the work in the field that had already been initiated, there was an existing demand and opportunities arose to introduce and present this content in other contexts. In a joint project with other universities, Open Educational Resources (OER) materials are being developed for university teaching on a range of CSE topics. Building on the existing preliminary work, a module on language learning in CS classes is being developed. Some central content of the course has also been integrated into our mandatory CS pedagogy course in the master's program (see 3), so that students become familiar with the aspects even if the specialized elective course is not chosen. The broad interest in the topic is also illustrated by the fact that we have been invited to present our work as part of guest lectures, training courses and seminars and in administrative working groups. This mainly took place in the context of teacher training in CS, but we also had the opportunity to present the subject-specific implementation in the context of interdisciplinary events on language pedagogy. We offer students to write their theses in the field of CS and language learning. Several students have already chosen to do so, either because they were motivated by the seminar or because they were aware of this focus, and have gone on to write successful theses in this area. In several cases, the subject of the work was the analysis and comparison of textbooks. One notable master thesis developed a list of criteria for evaluating materials for language sensitivity based on a review of literature and an interview study with CS teachers.

7 Conclusion and Outlook

We have explained why subject-specific language education in CS is important, how it has been integrated into the curriculum and implemented in practice at our university, and how this has resulted in further offers. Yet research in this area is still in its beginnings. Further research is needed to identify the linguistic difficulties in CS teaching in the field and to establish a consensus on the exact linguistic requirements. Developed approaches require systematic evaluation.

This applies in particular to compulsory CSE for all students, which has recently been introduced in NRW. The resulting increase in the heterogeneity of the student population is obvious. The heterogeneity of the teaching staff is also increasing as a result of meeting the growing demand, including from lateral entrants, and makes teaching practice in the newly introduced subject an important field of research, also with regard to the language of instruction.

Approaches to teaching programming languages and associated supporting and scaffolding methods have a long tradition in CSE. The same is true for questions of dealing with abstraction, precision and context independence. This raises research questions regarding how work from the CS field on learning programming languages and associated scaffolding techniques can contribute to subject-specific language learning approaches (and maybe even vice versa).

Finally, computational thinking and language skills are regarded as core and cross-cutting skills for the digital age. Here, concepts are needed to make computational thinking and its verbalization accessible to all.

References

1. Abbott, R.J.: Program design by informal english descriptions. Communications of the ACM **26**(11), 882–894 (1983)
2. Batur, F., Strobl, J.: Discipline-specific language learning in a mainstream computer science classroom. In: Proc. of the 14th WiPSCE Conference on Primary and Secondary Computing Education Research. pp. 1–4. ACM, New York (2019)
3. Beese, M., Kleinpaß, A., Krämer, S., Reschke, M., Rzeha, S., Wiethoff, M.: Praxishandbuch Sprachbildung Biologie. Ernst Klett Sprachen, Stuttgart (2017)
4. Bell, T., Vahrenhold, J.: Cs unplugged—how is it used, and does it work? In: Böckenhauer, H.J., Komm, D., Hromkovic, J. (eds.) Adventures between lower bounds and higher altitudes, pp. 497–521. LNCS Sublibrary: SL1 - Theoretical Computer Science and General Issues, Springer, Cham, Switzerland (2018)
5. Bransford, J.D.: How people learn: Brain, mind, experience, and school. National Academy Press, Washington, DC, expanded ed., 7. print edn. (2000)
6. Cummins, J.: Linguistic interdependence and the educational development of bilingual children. Review of Educational Research **49**(2), 222–251 (1979)
7. Darsow, A., Wagner, F.S., Paetsch, J.: Konzept für die empirische Untersuchung der Berliner DaZ-Module. In: Becker-Mrotzek, M., Rosenberg, P., Schroeder, C., Witte, A. (eds.) Deutsch als Zweitsprache in der Lehrerbildung, pp. 187–202. Sprachliche Bildung, Waxmann, Münster (2017)
8. Diethelm, I., Goschler, J., Lampe, T.: Language and computing. In: Sentance, S., Barendsen, E., Schulte, C. (eds.) Computer science education, pp. 207–219. Bloomsbury Academic, London (2018)
9. Döll, M., Hägi-Mead, S., Settinieri, J.: Ob ich mich auf eine sprachlich heterogene Klasse vorbereitet fühle? – Etwas! In: Becker-Mrotzek, M., Rosenberg, P., Schroeder, C., Witte, A. (eds.) Deutsch als Zweitsprache in der Lehrerbildung, pp. 203–215. Sprachliche Bildung, Waxmann, Münster (2017)
10. Gibbons, P., Cummins, J.: Scaffolding language, scaffolding learning. Heinemann, Portsmouth, NH (2002)
11. Gogolin, I., Hansen, A., McMonagle, S., Rauch, D.P. (eds.): Handbuch Mehrsprachigkeit und Bildung. Handbuch, Springer VS, Wiesbaden (2020)
12. Goschler, J., Montanari, E.: Deutsch als Zweitsprache in der Lehramtsausbildung: ein integratives Modell. In: Becker-Mrotzek, M., Rosenberg, P., Schroeder, C., Witte, A. (eds.) Deutsch als Zweitsprache in der Lehrerbildung, pp. 27–36. Sprachliche Bildung, Waxmann, Münster (2017)

13. Gürsoy, E.: Genredidaktik: Ein Modell zum generischen Lernen in allen Fächern mit besonderem Fokus auf Unterrichtsplanung (2018), `https://www.uni-due.de/imperia/md/content/prodaz/guersoy_genredidaktik.pdf`
14. Gürsoy, E., Benholz, C., Renk, N., Prediger, S., Büchter, A.: Erlös = Erlösung? Sprachliche und konzeptuelle Hürden in Prüfungsaufgaben zur Mathematik. Deutsch als Zweitsprache (1), 14–24 (2013)
15. Heath, S.B.: Ways with words: Language, life, and work in communities and classrooms. Cambridge University Press, Cambridge (1983)
16. Kniffka, G., Neuer, B.: Sprachliche Anforderungen in der Schule. In: Günther, H., Kniffka, G., Knoop, G. (eds.) Basiswissen Lehrerbildung, pp. 37–49. Klett Kallmeyer, [S.l.] (2017)
17. Lampe, T., Diethelm, I.: Transkriptanalyse einer Informatik-Unterrichtsstunde. In: Butler, M., Goschler, J. (eds.) Sprachsensibler Fachunterricht, pp. 203–219. Springer Fachmedien, Wiesbaden (2019)
18. Leisen, J.: Handbuch Sprachförderung im Fach: Sprachsensibler Fachunterricht in der Praxis. Ernst Klett Sprachen, Stuttgart (2013)
19. Ministerium für Schule und Bildung des Landes Nordrhein-Westfalen: Wahlpflichtfach Informatik: Kernlehrplan für die Sekundarstufe I Gymnasium in NRW (2023)
20. Prediger, S., Wessel, L.: Brauchen mehrsprachige Jugendliche eine andere fach- und sprachintegrierte Förderung als einsprachige? Zeitschrift für Erziehungswissenschaft **21**(2), 361–382 (2018)
21. Prediger, S., Wilhelm, N., Büchter, A., Gürsoy, E., Benholz, C.: Language proficiency and mathematics achievement. Journal für Mathematik-Didaktik **39**(S1), 1–26 (2018)
22. Rose, D., Martin, J.R.: Learning to write, reading to learn: Genre, knowledge and pedagogy in the Sydney School. Equinox textbooks and surveys in linguistics, Equinox, Sheffield (2012)
23. Saathoff, A., Pancratz, N., Diethelm, I.: Von Paketen, Klötzchen und Wolken – Auf der Suche nach geeigneten Sprachbildern für den Informatikunterricht. In: Butler, M., Goschler, J. (eds.) Sprachsensibler Fachunterricht, pp. 101–114. Springer, Wiesbaden (2019)
24. Schacht, F., Guckelsberger, S.: Sprachbildung in der Lehramtsausbildung Mathematik: Konzepte für eine sprachbewusste Hochschullehre. Springer eBook Collection, Springer, Berlin, Heidelberg (2022)
25. Schnieders, M., Batur, F., Brinda, T.: Die Befähigung zum Darstellungswechsel im Informatikunterricht als Aufgabe der Fach- und Sprachförderung?! In: Hellmig, L., Hennecke, M. (eds.) INFOS 2023 - Informatikunterricht zwischen Aktualität und Zeitlosigkeit, 20.-22.09.2023 Würzburg, Deutschland, pp. 177–186. Gesellschaft für Informatik, Bonn (2023)
26. Sentance, S., Waite, J.: Teachers' perspectives on talk in the programming classroom: Language as a mediator. In: Ko, A.J., Vahrenhold, J., McCauley, R., Hauswirth, M. (eds.) Proc. of the 17th ICER. pp. 266–280. ACM, New York (2021)
27. Strobl, J., Herberg, S., Gürsoy, E., Reimann, D.: Qualifizierung von Lehrkräften für das Unterrichten in sprachlich heterogenen Klassen. In: Leuders, T., Christophel, E., Hemmer, M. (eds.) Fachdidaktische Forschung zur Lehrerbildung, pp. 87–98. Waxmann, Münster (2019)
28. Tajmel, T.: Naturwissenschaftliche Bildung in der Migrationsgesellschaft: Grundzüge einer Reflexiven Physikdidaktik und kritisch-sprachbewussten Praxis. Springer Fachmedien, Wiesbaden (2017)
29. Thomas, W.P., Collier, V.: School Effectiveness for Language Minority Students. National Clearinghouse for Bilingual Education, Washington, DC (1997)

A Appendix A - Structure of the additional qualification *ZuS* (Translated)

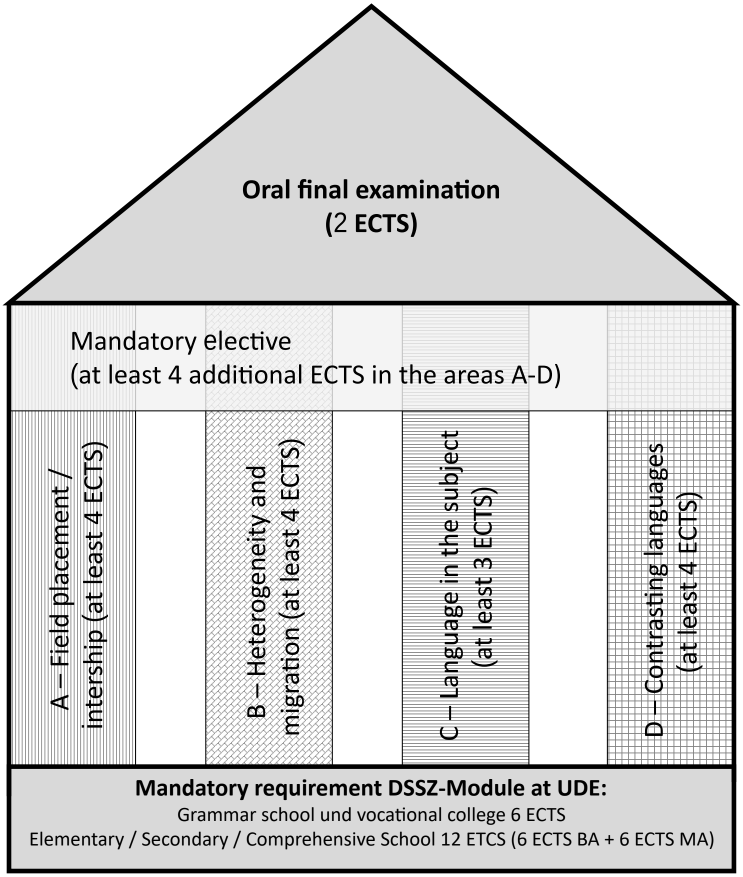


Fig. 1. Structure of ZuS (Translation of `https://www.uni-due.de/daz-daf/zus/zusvoraussetzungen.php`)

Hands-On vs. Hands-In: A Comparative Analysis of VR and Tangible Learning for Abstract Informatics Concepts

David Baberowski[1][0000−0001−6308−4334], Thiemo Leonhardt[2][0000−0003−4725−9776], and Nadine Bergner[2][0000−0003−3527−3204]

[1] TUD Dresden University of Technology, Germany
[2] RWTH Aachen University, Germany

Abstract. Understanding abstract concepts is a common challenge in computer science. Immersive and interactive media such as virtual reality (VR) have great potential to help learners with abstract concepts. However, the feasibility of VR is often questioned, especially when compared to hands-on activities that facilitate direct interaction between learners and abstract concepts.

This paper presents a study that compares the learning experience and learning effect between a VR learning environment and an equivalent physical tangible (TG) clone. The learning applications are designed to help learners aged 13 to 15 gain a better understanding of the process of routing in a home network. The educational design of the learning application is detailed, explaining the process to ensure comparability between the VR and TG versions. Tasks in a pre-post design and an established flow scale are used for evaluation. The results are consistent with recent research that is showing mediating effects of flow on learning when comparing immersive media. The study shows similar flow and learning effects in VR and TG due to the identical educational design.

Keywords: VR · Serious Games · Routing · Educational Design

1 Introduction

Informatics education includes many aspects of theoretical informatics, depending on national and international curricula [9][5]. Beginning in the early grades, students are exposed to many abstract concepts, including automata, formal languages, networks, and logic. Since concepts that are inherently abstract benefit from the use of analogies and metaphors [14], it is essential to explore ways to design and use such analogies effectively. Two possibilities for interactive learning with analogies and metaphors are virtual reality (VR) based serious games [17, 10] and tangible (TG) artifacts [18]. This study aims to contribute to a better understanding of the relationship between learning effects, learning experiences, and the potential of VR by comparing a VR learning environment with a TG replica.

The study addresses the concept of routing. The Informatics for All coalition has defined 11 core topics for the development of informatics curricula in Europe [5]. While these are not direct recommendations for curricula in specific age groups, routing is covered by two of these core topics (Computing Systems and Networks and Communications). This is in line with suggestions for computer science curricula in Germany, for example, which include routing and networks for secondary schools [9].

The utilization of virtual reality (VR) as a learning medium has been demonstrated to positively impact both learning engagement and process learning [12]. In contrast, TG environments have the potential to influence learning outcomes [11]. They can facilitate learning by scaffolding concept understanding, reducing cognitive load, and increasing learning activity. They can also change learning behavior by increasing attention, control, and expression. Furthermore, they can enhance learning emotions by making learning more engaging, immersive, and enjoyable. Mulders posited that flow is an essential moderating factor in VR learning applications [13]. However, it remains unclear whether this phenomenon can be directly compared to tangible learning.

2 Related Work

VR is an umbrella term for various technologies that aim to immerse the user in a virtual environment and make them feel as if they are physically present [19]. Examples of these technologies include PC-based VR, tablet-based VR, and CAVE systems. In this paper, VR specifically refers to head-mounted displays combined with controllers that allow the user to look around, move about, and interact with the environment.

Research on the benefits of VR for education suggests positive effects on learning engagement [6] and especially on process learning [12]. As a result, there are many applications that transport learners to distant or dangerous locations or allow them to practice processes in a safe and controlled environment. In a literature review of current trends in educational VR gaming, the majority of applications were found to be in areas such as healthcare, safety training, and pilot training (19 out of 30) [16]. When designing real-world-based learning scenarios, many aspects of the virtual environment, such as important objects, aesthetics, and interactions, are predetermined by the subject. When designing metaphor-based learning environments for abstract concepts, all of these decisions must be made consciously during the design process. At the same time, VR allows educators to realize any metaphor for a learning game because they have full control over the environment and behavior of the virtual environment. This makes research into effective design strategies for VR learning applications important.

Although there are legitimate criticisms, the vast majority of studies in this area are media comparison studies, which often do not clarify the underlying educational design for all conditions [4]. These studies do not allow generalized principles to be derived or theories to be formed. Looking at the learning experience, for example, seems to be a better approach, showing that the effect of the medium on learning is strongly moderated by flow [13]. The flow state is a psychological construct derived from motivational psychology. It is defined as a state of optimal experience characterized by a balance between one's perceived abilities and the perceived difficulty of the task at hand. This balance is achieved through a sense of coherence, concentration on a limited stimulus field, a change in temporal experience, and a merging of self and activity. In addition to the definition of flow as a balance between ability and challenge, the perceived importance of the activity and the individual's motivation to perform are important components. According to [8], additional factors are likely and flow should be measured in its multidimensionality in the future.

An alternative to interacting with VR-based analogies when learning abstract concepts are tangible learning objects. An example of this is unplugged activities, first introduced by Bell et al. [3]. The idea of computer science unplugged is to deliberately exclude digital technology from the learning process in order to focus on the fundamental principles of computer science concepts and their validity outside of digital systems. However, when objects need to exhibit complex behavior, tangible artifacts can be extended with embedded systems that no longer fit the definition of the unplugged approach. There is a large body of research on Tangible User Interfaces [18] and their application in learning contexts. A TG environment was used in this study, as this enabled the creation of a comparable counterpart to the VR environment in which as many aspects of the learning experience as possible, such as interaction mode and immersion, could be retained.

3 Educational Game Design

To illustrate the underlying educational design for the applications in this paper, the learning content and objectives related to the concept of routing are first defined. Then the metaphor used and the game design are described.

3.1 Learning Content

The underlying informatics content concerns the routing of data packets in a typical home network, where a router connects to local clients and the Internet Service Provider (ISP). The router must route data packets according to their destination IP address, which is a unique address for each network component. In a home router environment, there are three common cases for these packets:

- **Internal packets** have both a source and a destination within the local home network. The router has a direct connection to the destination client and can therefore match the packet by comparing these addresses. This type of packet occurs when local clients exchange data directly, such as screen sharing from a smartphone to a TV.

- **Outgoing packets** originate from a local client but are destined for an IP address outside the local network. In this case, the router does not have a direct connection to the destination client and forwards the packet to the ISP. The packet is then routed through the Internet, which is outside the scope of this learning design. This type of packet occurs whenever a client sends data to a server on the Internet, such as uploading a photo to Instagram.
- **Incoming packets** come from the Internet through the ISP. Because the router has only one external public IP address and the local clients are not visible from the outside, all incoming packets are addressed to the router's public IP address. To route the packet to the correct local client, the router must perform Network Address Translation (NAT). The router uses ports, an additional piece of metadata, to identify the destination client. Each time a connection is made between a local client and a server, packets on that connection use a specific port, which is recorded in a NAT table. When an incoming packet arrives, the router can use this NAT table in combination with the port to determine which local client the packet belongs to.

There are details and aspects such as the structure of IP addresses, the formation of subnets, and the use of ports outside of NAT that are relevant to the topic but are not discussed here. This is because the educational design does not cover these aspects in the learning activity.However, these aspects can and should be covered by the instructor for integration into the informatics lessons.

3.2 Learning Goals

The learning goal for the educational design is for students (age 13-15) to understand how the routing of data packets works. Since interaction with the learning object is at the core of VR and TG learning environments, students should take an active role in the learning process. This can be achieved by refining the learning objective as follows: "Students will perform the routing of data packets to and from clients in a home network by interpreting IP addresses and applying Network Address Translation (NAT)." This learning objective requires 7 sub-competencies (see table 1).

All these together fulfill the learning objective. The sub-competences are interdependent as shown in figure 1. By analyzing these competencies in the context of the routing process, 4 stages can be identified. While competencies A and B are necessary to gather all important information, competency C enables students to choose the right strategy by identifying internal (strategy uses competency D), outgoing (strategy uses competency E) and incoming (strategy uses competencies F and G) packets. An additional skill that is not necessary for the routing decision and requires knowledge transfer is H).

The tasks to evaluate the learning effect were constructed for the specific sub-competencies in a process of three stages. First different ideas for tasks where constructed and matched to the competences. Then the tasks were reviewed by a group of experts (5 people with a background in computer science education and university teaching). In this step, the experts rated the correspondence of the

Table 1. Sub-Competencies

A	Students will identify the important information (IP address and port) of a data packet.
B	Students know that a typical home network is built around a central router that connects clients to the ISP.
C	Students will assign IP addresses to the internal or local network and to the external network or Internet.
D	Students match internal packets to their local destination.
E	Students will match outgoing packets according to their external destination.
F	Students identify incoming packets from the Internet as a use case for NAT.
G	Students will perform NAT on an incoming packet.
H	Students will identify the complete path of a packet through the network from sender to destination.

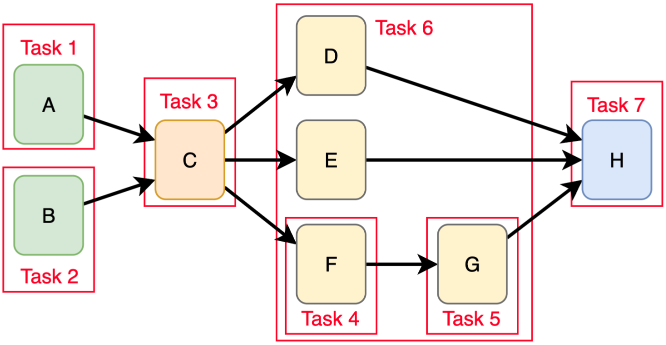


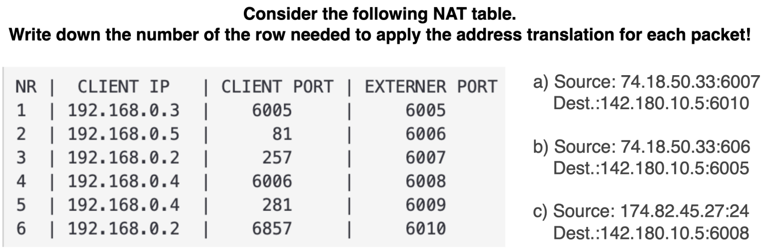
Fig. 1. Structure of the learning objective from sub-competencies. The tasks used to evaluate the learning effects are also shown and assigned.

tasks to the target competencies and had the opportunity to comment on general aspects of the task design. The instrument was then refined by incorporating the feedback and eliminating tasks that did not match the competences, resulting in 7 tasks (see Fig. 2) whose relationship to the competences is shown in Fig. 1.

3.3 Metaphor

To visualize the learning environment, a pneumatic tube post system was chosen, where different tubes represent connections to different clients, and data packets are represented by capsules traveling through these tubes. The metaphor satisfies the three requirements proposed by [2] for analogies for learning abstract concepts:

- The metaphor is **coherent** with the learning content by correctly representing the active components (clients - tubes, packets - capsules) and their relationships.



Consider the following NAT table.
Write down the number of the row needed to apply the address translation for each packet!

NR	CLIENT IP	CLIENT PORT	EXTERNER PORT
1	192.168.0.3	6005	6005
2	192.168.0.5	81	6006
3	192.168.0.2	257	6007
4	192.168.0.4	6006	6008
5	192.168.0.4	281	6009
6	192.168.0.2	6857	6010

a) Source: 74.18.50.33:6007
Dest.:142.180.10.5:6010

b) Source: 74.18.50.33:606
Dest.:142.180.10.5:6005

c) Source: 174.82.45.27:24
Dest.:142.180.10.5:6008

Fig. 2. Task 5, that tests the application of the NAT process on packets (sub-competency G).

- The metaphor is **reduced** by neglecting unnecessary details, such as the contents of capsules or other uses of port numbers.
- The metaphor is **familiar** in the sense that students may not know how pneumatic tube systems work, but they do know how addresses relate to packets in a mail system.

In order to have comparable environments, not only the metaphor and learning content, but also the visuals, interface and mechanics of the environments have to be as similar as possible (see Fig. 3). To allow students to focus on the learning activities, external stimuli should be reduced to a minimum. This was achieved by eliminating all unnecessary details within the environment. The important and active components are highlighted with spotlights to further direct the learner's focus. This means that the learner is transported into a learning environment filled with only five tubes and the NAT table. Within this environment, the learner receives feedback through multiple channels. If a packet is placed correctly, the corresponding tube should light up green and an acceptance sound is played. Then the next packet will come out of the next tube. If the player chooses the wrong tube, a red light and a reject sound will indicate that there is a problem and the packet will bounce back from the tube to the player. In this way, a packet can be played over and over until the correct tube is found.

3.4 Design of the Prototypes

The implementation of the VR environment as well as the TG environment use a sorting action to let the user assign packages to the tubes [1]. To maintain a level of activity and embodiment, the tubes were placed in a semicircle 2m in diameter. In this way, students had to actively move around to place the packages into the tubes (see Fig. 3). The VR environments use hand controllers that register a grasping gesture to allow the user to interact with the packages.

The TG environment was built with the goal of implementing the educational design as close as possible to the VR version. Therefore, the 3D models of the VR implementations were used to 3D print all objects (capsules and tubes). Since the

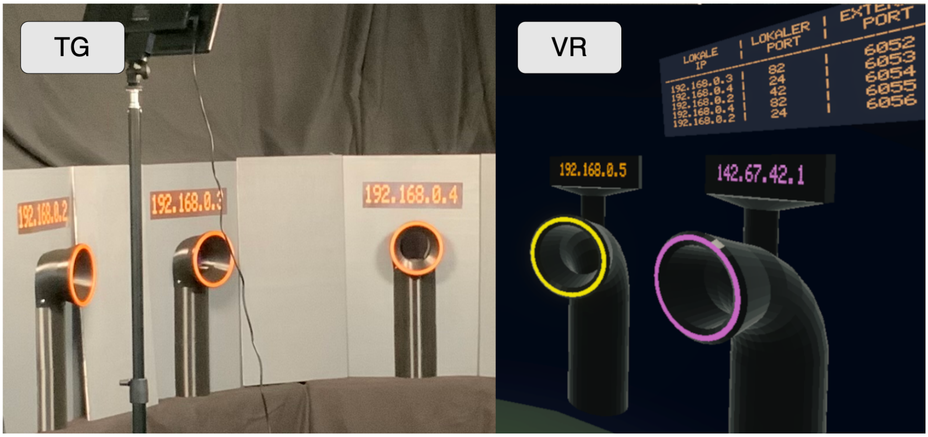


Fig. 3. TG (left) and VR (right) implementations of the pipe-post system metaphor: each pipe is labeled with an IP address, and the NAT table is placed above the pipe connecting to the ISP

actual and perceived size of objects in VR can differ significantly [15], different scaled-down versions of the objects were tested to preserve the subjective feeling while interacting with the object and at the same time keep the production effort (3D printing time and material) reasonable (see Fig. 4).

Fig. 4. Comparison of game objects in the VR and TG environment

The tubes were built as autonomous units using microcontrollers. Each tube unit consists of a distance sensor to detect the insertion of a capsule and a motor to move a conveyor belt to move a capsule into or out of the tube. All the tube units are connected to a laptop computer, which stores the current state of the game and sets the behavior of each tube (accepting or rejecting capsules).

All these details were not visible to the player because they were hidden behind black cloth. Spotlights were used to create the same lighting conditions as in the VR version. From the learner's perspective, only the game assets were visible, and because the tube units could provide instant feedback, the illusion of interacting with a machine or system was created. In reality, a game supervisor was needed behind the scenes to insert the capsules before they could be ejected to the learner. The supervisor could also see the state of the game and upcoming packets via the central laptop.

4 Study Results

The study used a pre-post design to assess learning and flow. The former was measured by the tasks described in the Educational Design section 3, while the latter was quantified using the Flow Short Scale [8]. First semester university students (N=23, convenience sampling) were randomly selected to play with one of the two environments (VR and TG). To collect additional data, participants played the second version after the post-test, this time answering only the flow items for the second learning environment as well. The full dataset for this study is available under a Creative Commons License (CC-BY) on the Open Science framework[3].

The participants were university students, some of whom had a background in computer science. As the pedagogical design is intended for use in the hands-on phase of the classroom and therefore builds on prior knowledge, the study began with a short video clarifying important definitions and explaining the concepts of routing and NAT. Then the participants were given the pre-test (learning effect). They then played one of the two versions of the learning game. For the study, a fixed sequence of capsules was used to avoid introducing different levels of difficulty by chance. After all capsules were placed correctly, the participants were given the post-test. Since each task is designed to assess a different sub-competency, the scores are not combined but considered separately. The normalized result for each task can be seen in table 2.

In the VR condition, participants' mean scores were higher in the post-test than in the pre-test for each task. A one-tailed Wilcoxon signed-rank test was used to test for significance, which was found for Task 4 (p=0.037) and Task 6 (p=0.004). Calculation of effect sizes using Pearson's r revealed large effects for both tasks (Task 4: r=0.613, Task 6: r=0.897) according to Cohen's classification [7]. In the TG condition, participants' mean posttest scores were higher than their pretest scores in tasks 2, 4, 5, and 6. The Wilcoxon signed-rank test showed significance for Task 6 (p=0.038), while Pearson's r showed a large effect (r=0.703).

---

[3] Private Review Access at `https://osf.io/2kvnm/?view_only=f20f7ba2e02547df9ef6997516e95d64`

Table 2. Results for each task in both environments VR (N=11) and TG (N=12) for pre and post scores (normalized).

Task	VR (pre)	VR (post)	change sign.	TG (pre)	TG (post)	change sign.
1	0.91 ± 0.30	0.98 ± 0.08	p=0.500	0.93 ± 0.15	0.90 ± 0.29	p=0.715
2	1.00 ± 0.00	1.00 ± 0.00	-	0.83 ± 0.39	0.92 ± 0.29	p=0.500
3	0.84 ± 0.20	0.93 ± 0.16	p=0.102	0.88 ± 0.17	0.85 ± 0.13	p=0.725
4	0.66 ± 0.34	0.88 ± 0.14	p=0.037*	0.78 ± 0.22	0.81 ± 0.22	p=0.338
5	0.76 ± 0.34	0.88 ± 0.17	p=0.135	0.78 ± 0.38	0.89 ± 0.22	p=0.186
6	0.50 ± 0.26	0.74 ± 0.17	p=0.004*	0.51 ± 0.36	0.68 ± 0.32	p=0.038*
7	0.88 ± 0.20	0.91 ± 0.11	p=0.356	0.85 ± 0.17	0.85 ± 0.23	p=0.367

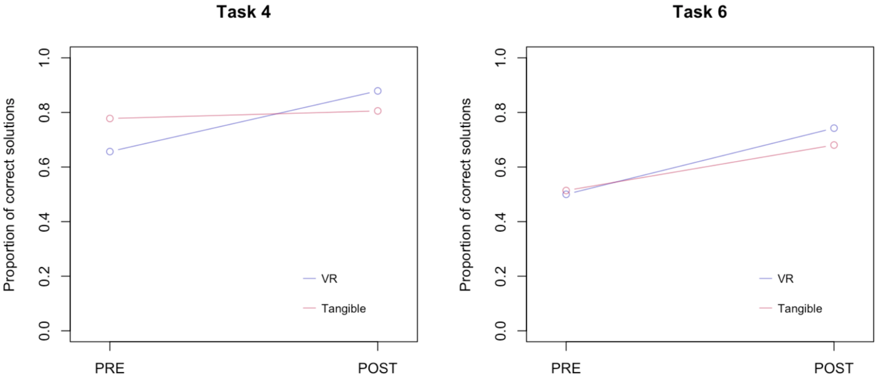


Fig. 5. Comparison of pre and post scores in tasks 4 and 6 for both environments

Flow was measured on a 7-point scale (0 to 6) using the Flow Short Scale [8]. There were no significant differences regarding the order in which the two games were played, which is why the measurements can be evaluated together regardless of the order. The scores for VR (4.8) and TG (4.6) are not significantly different (N=23, Wilcoxon signed-rank test, $\alpha = 0.25$, $1 - \beta = 0.86$). Although there may be an effect of game version on flow, we were unable to measure it due to the small sample size.

5 Discussion

The assumption of flow as a moderating factor, as proposed by Mulder [13] aligns with the findings for VR. The VR and TG variants of the game exhibited comparable flow values. While no significant difference was observed, it cannot be ruled out that there was an effect, though it was probably not substantial. This lends support to the hypothesis that flow is also an important moderating factor in TG learning situations. The comparable flow and learning effects indicate that the flow factor may exert a stronger influence on the learning situation than the medium.

The VR environment was better in sealing off the participants from the outside world, reducing distractions and maybe supporting flow. Since the TG version relied on a human behind the scenes to prepare the next packet, quick players sometimes had to wait a second for the next one in comparison to the VR environment in which packets were always sent immediately. However, all differences relate to the implementation and not to the content of the learning applications, since both use the same educational design and implement equivalent game mechanics.

Both environments show significant learning effects in the tasks most similar to the actual game activity, demonstrating that the instructional design has an effect on learning. Although the differences are not significant, it is noticeable that there are only increases (and one constant value) in the VR version when comparing pre and post scores, while for the TG version Taks 1 and 3 showed worsening. This can be explained in three ways:

The first explanation is due to the small sample size. Increasing the number of participants would make it possible to measure smaller effects and therefore making statements about differences between VR and TG possible. The second explanation is due to the the match between the participants and the target group. Since the participants were older, the tasks were probably too easy, explaining the high scores in the pre test. This causes saturation effects, reducing the ability to measure learning effects further. For this explanation speaks, that the tasks that showed significant effects, where exactly the ones with the lowest pre scores. Since participants in the TG group scored higher in most pre tasks, this saturation would effect this group more. Reproducing the study with students of the target audience could increase the sensitivity of the instrument. A third explanation would be, that there are differences in learning with VR and TG, possibly stemming from differences in immersion and sensory perception.

In the end, the results at hand do not allow conclusions about the superiority of one environment over the other in terms of learning effects; both can help learners with abstract concepts in computer science through hands-on activities. While flow is a moderating factor for learning with different media, VR opens up new possibilities when designing learning activities. This in turn could support learning experience, especially flow. In order to achieve this, more research in design principles for VR learning games is needed.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Baberowski, D., Keerl, N.M., Lilienthal, L., Rieske, M.E.J., Leonhardt, T.: Inside the router. eleed (2022)

2. Baberowski, D., Leonhardt, T., Bergner, N.: Inside the System - Designing VR Serious Games for Computer Science Education. In: Dondio, P., Rocha, M., Brennan, A., Schönbohm, A., De Rosa, F., Koskinen, A., Bellotti, F. (eds.) Games and Learning Alliance (2024)
3. Bell, T., Vahrenhold, J.: CS Unplugged—How Is It Used, and Does It Work? In: Böckenhauer, H.J., Komm, D., Unger, W. (eds.) Adventures Between Lower Bounds and Higher Altitudes, vol. 11011, pp. 497–521. Springer International Publishing, Cham (2018). https://doi.org/10.1007/978-3-319-98355-4_29
4. Buchner, J., Kerres, M.: Media comparison studies dominate comparative research on augmented reality in education. Computers & Education **195**, 104711 (Apr 2023). https://doi.org/10.1016/j.compedu.2022.104711
5. Caspersen, M.E., Diethelm, I., Gal-Ezer, J., McGettrick, A., Nardelli, E., Passey, D., Rovan, B., Webb, M.: Informatics Reference Framework for School (Feb 2022)
6. Chen, J., Fu, Z., Liu, H., Wang, J.: Effectiveness of Virtual Reality on Learning Engagement: A Meta-Analysis. International Journal of Web-Based Learning and Teaching Technologies **19**(1), 1–14 (Dec 2023). https://doi.org/10.4018/IJWLTT.334849
7. Cohen, J.: Statistical Power Analysis for the Behavioral Sciences. Psychology Press, New York, NY, 2. ed., reprint edn. (2009)
8. Engeser, S., Rheinberg, F.: Flow, performance and moderators of challenge-skill balance. Motivation and Emotion **32**(3), 158–172 (Sep 2008). https://doi.org/10/crx5gj
9. Gesellschaft für Informatik e. V.: Grundsätze und Standards für die Informatik in der Schule: Bildungsstandards Informatik für die Sekundarstufe I (2008)
10. Hakulinen, L.: Using serious games in computer science education. In: Proceedings of the 11th Koli Calling International Conference on Computing Education Research. pp. 83–88. ACM, Koli Finland (Nov 2011). https://doi.org/10.1145/2094131.2094147
11. Li, Y., Liang, M., Preissing, J., Bachl, N., Dutoit, M.M., Weber, T., Mayer, S., Hussmann, H.: A Meta-Analysis of Tangible Learning Studies from the TEI Conference. In: Sixteenth International Conference on Tangible, Embedded, and Embodied Interaction. pp. 1–17. ACM, Daejeon Republic of Korea (Feb 2022). https://doi.org/10.1145/3490149.3501313
12. Mayer, R.E. (ed.): Multimedia Learning. Cambridge University Press, Cambridge (2021)
13. Mulders, M.: Beyond Comparing Learning Technologies: Experiencing Flow in Virtual Reality. In: Sampson, D.G., Ifenthaler, D., Isaías, P. (eds.) Smart Learning Environments in the Post Pandemic Era, pp. 39–56. Springer Nature Switzerland, Cham (2024). https://doi.org/10.1007/978-3-031-54207-7_3
14. Norman, D.A.: Some Observations on Mental Models. In: Gentner, D., Stevens, A.L. (eds.) Mental Models, pp. 15–22. Psychology Press, New York (1983). https://doi.org/10.4324/9781315802725-5
15. Ogawa, N., Narumi, T., Hirose, M.: Distortion in perceived size and body-based scaling in virtual environments. In: Proceedings of the 8th Augmented Human International Conference. pp. 1–5. ACM, Silicon Valley California USA (Mar 2017). https://doi.org/10.1145/3041164.3041204
16. Oyelere, S.S., Bouali, N., Kaliisa, R., Obaido, G., Yunusa, A.A., Jimoh, E.R.: Exploring the trends of educational virtual reality games: A systematic review of empirical studies. Smart Learning Environments **7**(1), 31 (Oct 2020). https://doi.org/10.1186/s40561-020-00142-7

17. Pirker, J., Dengel, A., Holly, M., Safikhani, S.: Virtual Reality in Computer Science Education: A Systematic Review. In: 26th ACM Symposium on Virtual Reality Software and Technology. pp. 1–8. VRST '20, Association for Computing Machinery, New York, NY, USA (Nov 2020). https://doi.org/10.1145/3385956.3418947
18. Shaer, O.: Tangible User Interfaces: Past, Present, and Future Directions. Foundations and Trends® in Human–Computer Interaction **3**(1-2), 1–137 (2009). https://doi.org/10.1561/1100000026
19. Slater, M.: A Note on Presence Terminology. Presence Connect **3** (Jan 2003)

ICE-T: A Multi-Faceted Concept for Teaching Machine Learning

Hendrik Krone[1,2], Pierre Haritz[1,2], and Thomas Liebig[2,3]

[1] *Both authors contributed to this work equally.*
[2] TU Dortmund University, Dortmund, Germany
[3] Lamarr Institute for Machine Learning and Artificial Intelligence, Dortmund, Germany
`firstname.lastname@tu-dortmund.de`

Abstract. The topics of *Artificial intelligence (AI)* and especially *Machine Learning (ML)* are increasingly making their way into educational curricula. To facilitate the access for students, a variety of platforms, visual tools, and digital games are already being used to introduce *ML* concepts and strengthen the understanding of how *AI* works. We take a look at didactic principles that are employed for teaching computer science, define criteria, and, based on those, evaluate a selection of prominent existing platforms, tools, and games. Additionally, we criticize the approach of portraying *ML* mostly as a black-box and the resulting missing focus on creating an understanding of data, algorithms, and models that come with it. To tackle this issue, we present a concept that covers intermodal transfer, computational and explanatory thinking, *ICE-T*, as an extension of known didactic principles. With our multi-faceted concept, we believe that planners of learning units, creators of learning platforms and educators can improve on teaching *ML*.

Keywords: Artificial Intelligence · Machine Learning · Intermodal Transfer · Computational Thinking · Explanatory Thinking

1 Introduction

Machine Learning (ML) is currently finding its way into more and more curricula due to its increasing significance. Despite the availability of initial materials for classroom instruction, there is still a shortage of suitable resources for effective teaching. In recent years, we have seen new platforms, tools, and games address the topic of *Artificial Intelligence* (AI) in early education (primary school to high school). Studies have demonstrated that digital game-based learning can boost students' motivation and engagement, enrich their cognitive and emotional development, and thereby enhance their learning efficiency [2]. Using digital games for teaching *ML* offers the advantage that students can learn and experiment in a playful manner. This not only enhances their comprehension of these complex subjects but can also spark their interest in the topic.

Visual tools are a popular way to teach *ML* concepts. The variety of representations plays an important role in enhancing the learning experience. In addition, these formats can help to include diverse teaching methods, making complex concepts more accessible to a wider range of students.

Learning platforms play a crucial role in supporting educators. There is currently a significant gap in teacher training for *AI* skills [15], indicating that many computer science teachers are unfamiliar with *ML* techniques and find it challenging to teach them. Using learning platforms enables them to access resources and materials tailored to the needs of students, contributing to more effective *ML* education.

For education, various guidelines, frameworks and models exist. The Technological Pedagogical Content Knowledge (TPaCK) framework describes the knowledge required by educators to create technology-enhanced learning environments, emphasizing the interplay between content knowledge, pedagogy, and technological literacy [11]. DPaCK strengthens digital literacy through subject-specific content, while AI-PaCK provides a structured description of AI teacher education [8]. But didactic principles have not been sufficiently explored to develop a comprehensive and systematic approach for creating new tools, games, and platforms dedicated to teaching *ML* content. To fill this gap, we have answered the following research questions as our main contribution in this paper:

1. *RQ1*: To what extent do existing games, digital tools and platforms for teaching machine learning implement the facets of intermodal transfer, as well as computational and explanatory thinking?
2. *RQ2*: How can a process model for teaching Machine Learning be derived from a Data Mining process model?
3. *RQ3*: How can the presented didactic principles be combined and applied?

To answer these questions, we first introduce three facets of thinking and the related work on the topic. Then, we compare the tools, games, and platforms and take a close look at the process model for Data Science. Finally, we present our proposal, *ICE-T*, a multi-faceted concept for teaching ML.

2 Background & Related Work

In this section, we will give an introduction to three facets of thinking and present other related work.

2.1 Intermodal Transfer

Intermodal transfer refers to the ability to apply knowledge or skills learned in one modality or context to a different modality or context. It involves the transfer of learning from one domain to another. In the context of education and cognitive development, intermodal transfer can be observed when individuals successfully apply what they have learned in one mode (e.g., visual, hands-on) to another mode (e.g., symbolic, abstract).

The principle of *EIS* (Enactive, Iconic, Symbolic) by Bruner [4] is a framework used in educational psychology, specifically in the context of learning and cognitive development. The principle is visualized in figure 1. This framework describes three levels of representing information that learners advance through as they acquire knowledge and skills. The *EIS* principle is closely related to the *spiral approach*, in which students progress through a subject with ever-increasing complexity [4]. In the case of the *EIS* principle, intermodal transfer can be evident in the progression from enactive to iconic to symbolic representation. Learners develop the ability to transfer knowledge and skills across different modalities, for example:

- *Enactive to Iconic:* Learners transfer knowledge gained through physical experiences (enactive) to mental images or visual representations (iconic). For example, a child who has learned to balance on a bicycle (enactive) can transfer this knowledge to create a mental image of bike riding.
- *Iconic to Symbolic:* As learners move from iconic to symbolic representation, they transfer their understanding from visual or sensory images to abstract symbols or linguistic forms. This involves the ability to connect concrete visual representations to more abstract and conceptual symbols.

The intermodal transfer perspective emphasizes the flexibility and adaptability of cognitive processes, highlighting that learning in one modality can enhance performance or understanding in another. This aligns with the broader concept of intermodal thinking, where individuals seamlessly integrate and transfer knowledge across different modes of thought or disciplines.

In summary, intermodal transfer is an important aspect of the learning process, and the *EIS* principle provides a framework that showcases how learners progress through different modalities, facilitating the transfer of knowledge and skills between them.

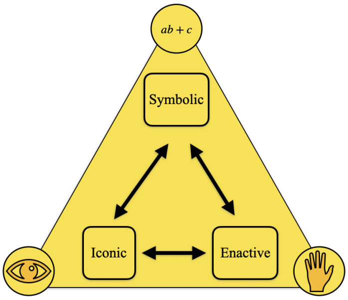


Fig. 1. Intermodal Transfer via the EIS principle: Enactive - Iconic - Symbolic.

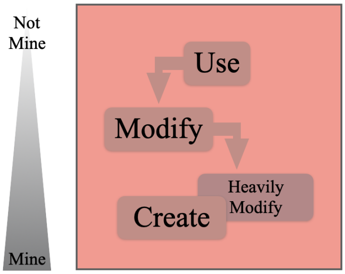


Fig. 2. Computational Thinking via UMC: Use - Modify - Create.

2.2 Computational Thinking

Computational thinking is an approach that includes problem-solving and problem formulation aspects based on concepts and methods from computer science

[6]. It involves breaking down complex problems into smaller parts, and then using algorithms and logical thinking to devise solutions. Computational thinking is not exclusive to computer scientists; rather, it's a fundamental skill that can be applied across various disciplines and everyday life [17].

The principle of *UMC*, which stands for *Use-Modify-Create*, is often associated with computational thinking, especially in the context of learning to program, modeling, or working with software [9], making it more suitable for machine learning where models and similar components play a crucial role. In contrast, for example PRIMM [14] places a stronger focus on programming.

The *UMC* principle, shown in figure 2, aligns with the iterative nature of computational thinking. Individuals start by using existing solutions, move on to modifying them based on their needs or understanding, and eventually progress to creating entirely new solutions. This approach fosters a mindset of continuous improvement, adaptability, and creativity in the realm of computational problem-solving.

- **Use:** When individuals *use* existing algorithms or code, they are applying computational thinking to understand and leverage the functionalities provided. This can be using a programming library or software application to accomplish a specific task that involves understanding how to apply pre-existing solutions.
- **Modify:** *Modifying* existing code or algorithms requires a deeper understanding of how they work. This process involves decomposition, pattern recognition, and algorithmic thinking, for example tweaking the parameters of an existing algorithm or adapting a piece of code for a different purpose.
- **Create:** *Creating* original solutions, algorithms, or code is at the core of computational thinking. It involves decomposition to understand the problem, abstraction to focus on essential details, and algorithmic thinking to design a solution. This step is sometimes less strict with regards to the learners creativity and therefore can be more of a **Heavily Modify** task, which internally combines *Create* and *Modify* subtasks.

In educational contexts, integrating *UMC* with computational thinking provides a structured approach for learners to engage with and master the principles of programming and problem-solving using computational concepts.

2.3 Explanatory Thinking

Explanatory thinking refers to the cognitive process of constructing and providing explanations for various phenomena, events, or concepts. It involves the ability to analyze information, identify patterns, and articulate coherent and meaningful explanations for why something happens or how it works. Explanatory thinking is a crucial aspect of critical thinking and problem-solving, allowing individuals to make sense of the world around them and communicate their understanding effectively. In the context of *AI/ML*, this would mean following a process of learning how to understand the task, working with the data and equip the students with the ability of being able to articulate (and thereby justify) modeling choices. By reasoning about science, students get opportunities for knowledge building [7] and are more likely to attend to the interactive, invisible components in the model to explain such a process [13].

2.4 Related Tools & Games

Wangenheim et al. [16] compiled visual tools for teaching *ML* in the K-12 education sector and identified three key aspects of visual tools for teaching ML in K-12 education. First, they emphasize "learning by doing" through hands-on activities and model development. Second, they are based on constructivism and constructionism, promoting active knowledge construction and artifact creation. Third, they highlight the importance of adapting tools to local contexts to motivate students with relevant and interesting problems.

Exploring the use of concepts in education, Marques et al. [10] examined 30 instructional units (courses, workshops, activities, etc.) primarily used in high schools in their work. They emphasized that, given the complexity of ML concepts, several units cover only the most accessible processes. For example, some units only teach data management, present model learning and testing at an abstract level, and hide some of the underlying *ML* processes.

In [1], Alam found that digital games could improve programming and computational thinking skills. They improve problem-solving through complex puzzles, introduce algorithmic thinking with rule-based gameplay, and require logical and strategic thinking. Additionally, these games foster creativity by allowing players to design elements. A critical aspect is the immediate feedback provided by the games, which supports iterative problem-solving, a fundamental component of computational thinking.

3 Methodology & Results

In this section, we present and examine existing tools, platforms and digital games according to evaluation criteria based on the aforementioned didactic facets. We furthermore propose a process model for promoting explanatory thinking based on an industry-standard process model for *Data Mining* tasks. Finally, we present our concept, *ICE-T*, as a combination of three principles.

3.1 RQ1: *To what extent do existing games, digital tools and platforms for teaching machine learning implement the facets of intermodal transfer, as well as computational and explanatory thinking?*

In order to address the didactic principles used in current games, digital tools, and learning platforms for teaching machine learning, we conducted an analysis of prominent tools, platforms, and games cited in the research of Wangenheim et al., Ashraf, and Zhou et al., as presented in table 1.

Our investigation focused on the extent and manner in which these resources implement the use-modify-create principle as a component of computational thinking. Additionally, we examined the presence and transfer between enactive, iconic, and symbolic forms of representation. For the explanatory facet, we examined whether machine learning algorithms, as well as data handling and the applied processes are taught.

	Computational			Intermodal			Explanatory		
Tool/Platform/Game	use	mod.	create	en.	ico.	sym.	ML alg.	process	data
code.org (AI for Oceans)[4]	✓			✓				✓	✓
Google Teachable Machine[5] [5]	✓	✓	✓	✓	✓			✓	✓
Machine Learning For Kids[6]	✓	✓	✓	✓		✓*		✓	✓
mblock (Face recognition system)[7]	✓	✓	✓			✓*		✓	✓
Minecraft Education: Unit 9 (AI)[8]			✓	✓		✓*		✓	✓
orange3[9]	✓	✓	✓		✓			✓	✓
SnAIp (block-based platform)[10]	✓	✓	✓			✓	✓	✓	
Tensorflow Playground [11]	✓	✓		✓	✓			✓	✓
MIT APP Inventor[12]	✓	✓	✓			✓*	✓	✓	✓
While True: Learn()[13]			✓	✓	✓			✓	✓

Table 1. Evaluation and comparison of digital tools, platforms, and games that emphasize computational thinking through use, modify, and create capabilities, intermodal transfer across enactive, iconic, and symbolic representations, and explanatory thinking using ML algorithms, processes, and data. Checkmarks signify that the respective concept plays a major role. *Remark:* * *(Model deployment)*

After the analysis from Zhou et al. they recommend the use of the *use-modify-create* method as a suitable approach [20]. Our in-depth analysis shows that apart from *code.org (AI for Oceans)*, *Tensorflow Playgrounds* and the two games *Minecraft Education (Unit 9)* and *While True: Learn()*, all tools and platforms have integrated the whole concept.

The advantage of using multiple forms of representation for explanation has not yet been widely adopted. While *Google Teachable Machine*, *Tensorflow Playgrounds* and *While True: Learn()* cover enactive aspects mentioned by Wangenheim [16], they also exploit the potential of iconic explanation. At the symbolic representation level, many tools and platforms use the model deployment ap-

---

[4] https://code.org/oceans

[5] https://teachablemachine.withgoogle.com

[6] https://machinelearningforkids.co.uk

[7] https://planet.mblock.cc/project/234478

[8] https://education.minecraft.net/de-de/lessons/hour-of-code-generation-ai

[9] https://orangedatamining.com

[10] https://snap.berkeley.edu/search?query=snaip

[11] https://playground.tensorflow.org

[12] https://appinventor.mit.edu

[13] https://luden.io/wtl/

proach, which focuses on how to use a model and lets the ML-algorithm remains a black-box.

As Marques et al. found in their analysis of workshops, courses, and activities, we also found in all the tools, games, and platforms we studied, that machine learning is always taught by focusing on the processes [10]. Some tools focus on more accessible processes; for example, *code.org* uses a training phase in which users distinguish fish from garbage, followed by a testing phase in which users watch the computer apply what it has learned. This is followed by the decision-making process, where the user decides whether model should be trained further. *Orange 3* has a strong emphasis on the process, with a focus on workflow and the iconic representation of results, e.g. through the visualization of trees, box plots, and scatter plots.

Teaching machine learning can benefit from a combination of didactic principles. The MIT suggests using *Google´s Teachable Machine* to develop a model before using it with their *APP Inventor* platform. This combination could provide multi-faceted support for teaching ML concepts.

3.2 RQ2: *How can a process model for teaching Machine Learning be derived from a Data Mining process model?*

To answer this research question, we introduce *CRISP-DM*, before proposing a scheme based on the process model for the context of designing iterative learning units for *Machine Learning* students in education. Other applications have been highlighted by Schroer et al. in [12].

Due to its better flexibility and universality [3] compared to other process models (e.g. *SEMMA*, *KDD*), the *CRoss Industry Standard Process for Data Mining* [19] appears to be most suited for adaptation into an educational context. This process model for *Data Mining* or *Data Science* tasks is widely used in industry.

Figure 3 shows the six phases of the process:

1. *Business Understanding* focuses on the requirements and objectives of a use case, which are then converted into a Data Mining (or Data Science) problem.
2. *Data Understanding* describes a phase that consists of acquiring initial data and understanding its characteristics. From this, the goal is to identify potential issues and challenges.
3. *Data Preparation* covers the construction of the data set that will be used in the modeling phase. In this phase it can be necessary to remove, clean, engineer or transform the raw data into a format suitable for analysis.
4. *Modeling* means selecting modeling techniques and algorithms and then building predictive or descriptive models based on the prepared data.
5. *Evaluation* is usually done during, and after the training process. Here, the models' quality gets measured and compared to the criteria set during the business understanding phase.
6. *Deployment* is the final step to integrate the *Machine Learning* model into an operational environment, as an application.

CRISP-DM is an iterative process. Even after the deployment phase, if there are additional insights to be gained or if business objectives evolve, the process may cycle back to earlier phases, such as refining the business understanding or adjusting the data preparation and modeling steps.

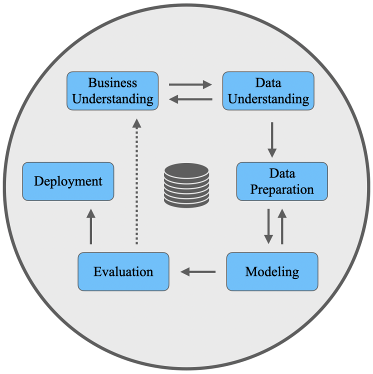


Fig. 3. CRoss Industry Standard Process for Data Mining.

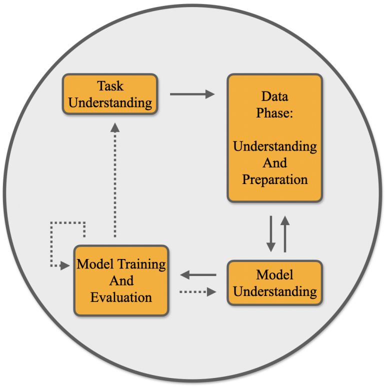


Fig. 4. Promotion of Explanatory Thinking Standard Process for Machine Learning.

Since our goal is to incorporate a principle that promotes explanatory thinking, we adapt the *CRISP-DM* process model to an educational setting. In contrast to CRISP-DM, the Business Understanding phase is intended to provide a given problem/task description. Here, concise formulation is a requirement. The next phase, *Data Understanding*, can be part of the descriptive text, but can also be integrated into tasks such as exploratory analysis (e.g. read/plot data).

Another option is to combine *Data Understanding* and *Data Preparation* and directly design tasks for modifying data after a given description of the data, and most likely constraints of the method that is to be applied. We denote the set of phases 2 and 3 as the *Data phase*. We postulate that designing tasks for this phase is a mandatory step for learners to be able to get a deeper understanding of the data and in turn of the actual method. The Model Understanding phase follows. To promote an understanding of *AI*, we believe it is important to not treat the method/algorithm as a black-box, but instead design tasks in a way that aim to teach how these methods make choices.

Only now should the *Model Training* phase start, and with it tasks that concern the actual method. As a last phase, *Evaluation* metrics are applied and the model is evaluated. Depending on the method, and due to the iterative design of the process, it is possible to revisit any of the previous phases.

The *Deployment* phase plays a bigger role in practical applications (CRISP-DM), but can likely be omitted in K-12 educational cases. We note that we still believe that it has its merits if the specific goal is to support experimental learning opportunities, e.g. through group projects.

In consequence, we define the phases of the Promotion of Explanatory Thinking Standard Process for Machine Learning (*short: PETSP-ML*), depicted in figure 4 as such:

1. **Task Understanding** can be defined as the phase where students gain a comprehensive understanding of a given academic task or assignment. This phase involves students actively engaging with the requirements, objectives, and expectations of the task to ensure clarity and alignment with their learning goals. In this context, it is especially important to convey why *ML* helps with a given problem.
2. **Data Phase** consists of Data Understanding & Data Preparation. Learners should have the task of dealing with the data in a practical way by visualizing the data and being able to transform it in a guided environment.
3. **Model Understanding** refers to the phase where individuals seek to comprehend the inner workings of a machine learning model. This phase is crucial for gaining insights into how the model makes decisions, understanding its limitations, and ensuring that the model aligns with the intended goals.
4. The **Model Training and Evaluation** phase is a crucial step where students make a model and learn from the provided data to make predictions or perform a specific task. They should gain insight into its behavior and predictions through evaluating the training. This step can also be iterated multiple times on its own if necessary.

3.3 RQ3: *How can the presented didactic principles be combined and applied?*

We finally present our own learning concept, called *ICE-T*, which combines the principles of *Intermodal Transfer*, *Computational Thinking* and *Explanatory Thinking*, and which is illustrated in figure 5. We argue that in order to fully understand and use *ML* methods, it is necessary to cover all of these areas when designing learning units for school curricula, learning platforms, etc. To support our claim, we will refer to figure 6 and provide an example:

We consider the situation in which an educator plans to teach how to use *Decision Trees* [18] as a classifier for animals (**1:**). In this example, the data consists of a set of the animals *Lion, Shark, Eagle, Penguin*, each with information about their attributes (*Has feathers?, Can fly?, Eats meat?, Has fins?*). The task (*"Classify different animal species by asking as few questions about their characteristics as possible"*) is presented in a problem-centered manner, where students should discuss ideas to come up with solutions - independent of the algorithm that will be taught. To understand the data, learners first examine tables of animals and their characteristics (**2:** *Iconic + Use*). Students can also do some of the preparation by expanding to the table and adjusting the data (**3:** *Iconic + Modify*). To promote a deeper understanding, they can visualize the data in different ways, e.g., by plotting (**4:** *Enactive, Iconic + Use*).

To understand the model, it can be helpful to first play a "question-answer" game, in which players can decide the order in which the questions will be

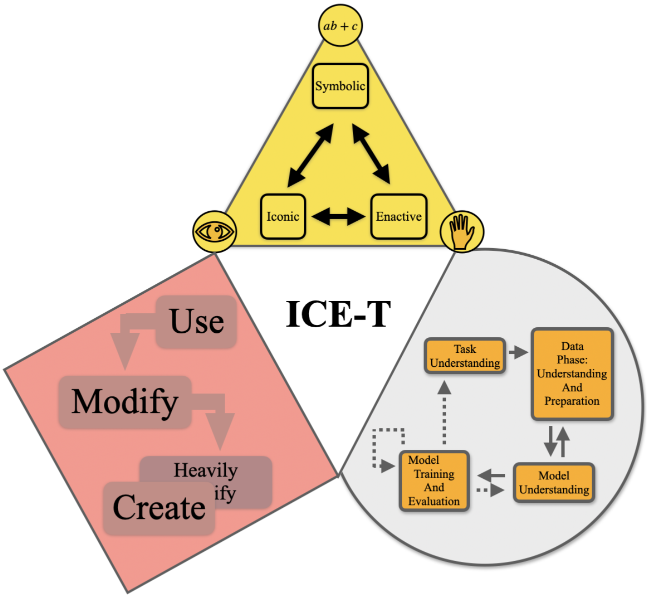


Fig. 5. The three facets of the ICE-T principle: Intermodal Transfer, Computational Thinking, and Explanatory Thinking.

answered, with the goal of asking as few questions as possible in total (**5:** *Enactive + Use*). Additionally, after each question, a tree-like structure could show how the question splits the data depending on the chosen question. Playing multiple rounds could also allow for comparison of resulting trees (**6:** *Iconic + Use*).

For training, students could change model parameters as a programming task (**7:** *Symbolic + Use and Modify*) and evaluate the immediate result according to certain rules (**8:** *Iconic + Use*). Here, it might also be helpful to show the effect of trying to classify data that was not part of training the classifier.

In further iterations of the process (spiral approach), complexity should increase. As an example, the data could now also be generated by the students, and they should experience the algorithmic construction of the tree and model selection through the effect of splitting the data before training different models, or experiment with node splitting criteria (**second iteration:** *Iconic + Symbolic + Use + Modify + Heavily Modify/Create*).

4 Conclusion

We examined didactic principles that get employed for teaching computer science and maths, defined didactic criteria and, based on those, evaluated a selection of existing platforms, tools and games. Additionally, we highlighted the issue of portraying *ML* mostly as a black-box and hence the lack of focus on creating

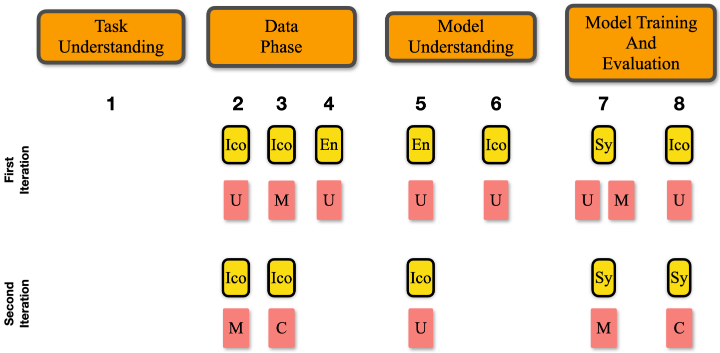


Fig. 6. The iterative process of teaching the use of decision trees for animal classification is illustrated using all elements of ICE-T. The diagram shows four phases of PETSP-ML (Task Understanding, Data Phase, Model Understanding, and Model Training and Evaluation) over two iterations. Each phase indicates different activities and elements based on the EIS and UMC principles at each step.

an understanding of data, algorithm and model that comes with it. To address this, we presented our multi-faceted concept, *ICE-T*.

Our concept includes different representation facets, the utilization of the use-modify-create framework, and our *PETSP-ML* model for educational needs based on the *CRISP-DM* model. The concept aims to provide teachers, planners of learning units, and learning platform developers with guiding didactic principles to enhance the understanding of *ML* and facilitate teaching in this important area. Given the growing importance of *ML* in today's world, it is crucial to introduce students to the fundamentals of this complex technology.

We hope that this paper will contribute to the dialogue about how to teach *ML* and that planners of learning units, creators of learning platforms and educators can benefit from being guided by our concept.

Acknowledgments. *This work was supported by the Federal Ministry of Education and Research of Germany and the state of North-Rhine Westphalia as part of the Lamarr-Institute for Machine Learning and Artificial Intelligence.*

References

1. Alam, A.: A digital game based learning approach for effective curriculum transaction for teaching-learning of artificial intelligence and machine learning. In: 2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS). IEEE (Apr 2022)
2. Anastasiadis, T., Lampropoulos, G., Siakas, K.: Digital game-based learning and serious games in education. International Journal of Advances in Scientific Research and Engineering **4**(12), 139–144 (2018)
3. Azevedo, A., Santos, M.F.: Kdd, semma and crisp-dm: a parallel overview. IADS-DM (2008)

4. Bruner, J.S.: Toward a theory of instruction. No. 59 in Harvard paperback, Harvard Univ. Press, Cambridge, Mass. [u.a.], 8. pr. edn. (1978)
5. Carney, M., Webster, B., Alvarado, I., Phillips, K., Howell, N., Griffith, J., Jongejan, J., Pitaru, A., Chen, A.: Teachable machine: Approachable web-based tool for exploring machine learning classification. In: Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems. CHI '20, ACM (Apr 2020)
6. Kalelioglu, F., Gülbahar, Y., Kukul, V.: A framework for computational thinking based on a systematic research review. Baltic Journal of Modern Computing **4**(3), 583 (2016)
7. Kenyon, L.O., Zemmer, J.D., Penry, B.A.: Explanatory thinking using mobile devices in a nonscience majors biology course. Journal of College Science Teaching **49**(1), 12–18 (2019)
8. Lorenz, U., Romeike, R.: What Is AI-PACK? – Outline of AI Competencies for Teaching with DPACK, pp. 13–25. Springer Nature Switzerland (2023)
9. Lytle, N., Cateté, V., Boulden, D., Dong, Y., Houchins, J., Milliken, A., Isvik, A., Bounajim, D., Wiebe, E., Barnes, T.: Use, modify, create: Comparing computational thinking lesson progressions for stem classes. In: Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education. p. 395–401. ITiCSE '19, Association for Computing Machinery, New York, NY, USA (2019)
10. Marques, L.S., Gresse von Wangenheim, C., Hauck, J.C.R.: Teaching machine learning in school: A systematic mapping of the state of the art. Informatics in Education pp. 283–321 (Jun 2020)
11. Mishra, P., Koehler, M.J.: Technological pedagogical content knowledge: A framework for teacher knowledge. Teachers College Record: The Voice of Scholarship in Education **108**(6), 1017–1054 (Jun 2006)
12. Schröer, C., Kruse, F., Gómez, J.M.: A systematic literature review on applying crisp-dm process model. Procedia Computer Science **181**, 526–534 (2021)
13. Schwarz, C.V., Reiser, B.J., Davis, E.A., Kenyon, L., Achér, A., Fortus, D., Shwartz, Y., Hug, B., Krajcik, J.: Developing a learning progression for scientific modeling: Making scientific modeling accessible and meaningful for learners. Journal of Research in Science Teaching: The Official Journal of the National Association for Research in Science Teaching **46**(6), 632–654 (2009)
14. Sentance, S., Waite, J.: Primm: Exploring pedagogical approaches for teaching text-based programming in school. In: Proceedings of the 12th Workshop on Primary and Secondary Computing Education. p. 113–114. WiPSCE '17, Association for Computing Machinery, New York, NY, USA (2017)
15. Sperling, K., Stenberg, C.J., McGrath, C., Åkerfeldt, A., Heintz, F., Stenliden, L.: In search of artificial intelligence (ai) literacy in teacher education: A scoping review. Computers and Education Open **6**, 100169 (2024)
16. Gresse von Wangenheim, C., Hauck, J.C.R., Pacheco, F.S., Bertonceli Bueno, M.F.: Visual tools for teaching machine learning in k-12: A ten-year systematic mapping. Education and Information Technologies **26**(5), 5733–5778 (May 2021)
17. Wing, J.M.: Computational thinking. Commun. ACM **49**(3), 33–35 (mar 2006)
18. von Winterfeldt, D.: Structuring decision problems for decision analysis. Acta Psychologica **45**(1-3), 71–93 (1980)
19. Wirth, R., Hipp, J.: Crisp-dm: Towards a standard process model for data mining. In: Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining. vol. 1, pp. 29–39. Manchester (2000)
20. Zhou, X., Van Brummelen, J., Lin, P.: Designing ai learning experiences for k-12: Emerging works, future opportunities and a design framework (2020)

Understanding Teachers' Motivations for Digital Education Training: A Qualitative Study

Corinna Hörmann[1] [0000−0002−4770−6217], Lisa Kuka[1] [0000−0002−0000−59154], and
Barbara Sabitzer[3] [0000−0002−1304−6863]

Johannes Kepler University, Linz, Austria
`corinna.hoermann@jku.at`
`lisa.kuka@jku.at`
`barbara.sabitzer@jku.at`

Abstract. As the digital revolution continues to reshape educational landscapes, understanding how educators engage with digital training has become crucial for fostering effective teaching methods. This paper delves into the level of engagement among Austrian teachers in digital education training. Using a comprehensive survey distributed to every secondary public school teacher, the study collected both quantitative and qualitative data to examine various aspects of digital education training. Teachers provided qualitative data through comments alongside traditional survey responses, offering more profound insights into their experiences, preferences, and challenges related to digital education training. By analyzing these comments, the study uncovered nuanced perspectives on the preferred training modes, the support teachers need, and the motivations that drive them to pursue professional development. The qualitative data revealed a range of opinions, from positive perceptions to specific concerns, providing a richer context to the quantitative findings. This qualitative analysis highlighted the factors contributing to teachers' engagement in digital education training and underscored the importance of flexibility, efficiency, and tailored support in professional development programs. Overall, the study suggests that teachers are open to engaging in digital education training, especially when it is responsive to their unique needs and time constraints.

Keywords: Digital Education · Teacher Training · Teacher Engagement · Professional Development

1 Introduction

Teacher professional development is essential in the educational landscapes across Europe, with varying degrees of obligation across different nations. In Europe, continuing professional development (CPD) is mandated for teachers in numerous countries, including the Czech Republic, Germany, Austria, and the UK. In contrast, it remains optional in others like Greece and Italy. The extent and rewards of CPD vary widely across regions. For instance, in Lithuania, Slovenia, and Finland, teachers are entitled to up to five days of CPD annually, while the Czech Republic provides twelve. Notably, only a few nations offer salary increments as motivations for completed teacher training [9].

In Austria, the commitment to teacher training differs by contractual terms, with requirements ranging from zero to fifteen hours per year. Austrian educators maintain autonomy over the choice and scope of their CPD activities, often selecting programs based on personal interest, significantly influencing their CPD decisions [3, 10].

The quality of instruction and the effective integration of computer science into school curricula are hindered by insufficient teacher training [6]. The demand for qualified educators underscores the necessity for training programs, as the need for computer science graduates relative to market needs continues to challenge the education system. To address this, over half of the educational systems in Europe provide retraining programs for existing teachers to acquire additional qualifications, and more than a third offer alternative certification paths for those without formal teaching credentials [8].

Before entering teacher training programs, motivational factors play a crucial role, with intrinsic motivation and interest in the content being significant drivers for participation. Additionally, the practical relevance of the training content, the expertise of the trainers, and the intellectual challenges posed during training sessions are vital in enhancing intrinsic motivation among teachers [1, 15, 2].

This paper is organized as follows: The introduction provides a brief overview of the topic, laying the groundwork for a detailed examination. Section two discusses the background, offering an overview of digital education in Austria and insights into the Austrian in-service teacher training landscape. The study's methodology is detailed in section 3.1, and the findings are discussed in sections 3.2 and 3.3. These results are then discussed in section 3.4. The paper concludes with a final section that summarizes the main insights and implications of the study.

2 Background

Since its introduction as an independent subject in the Austrian educational system in 1985, "Computer Science" has been assigned two hours per week at the 9th grade level in Academic Secondary Schools. With the introduction of "Digital Education" as a compulsory subject in lower secondary education in 2022 a new topic was installed into Austria's school system.

2.1 Digital Education in Austria

Austria has progressively integrated digital literacy into its curriculum since introducing computer science as a compulsory subject in ninth grade in 1985. Recent reforms have further emphasized digital literacy, mandating digital education as a standalone subject for students in grades five through eight. The curriculum, developed in collaboration with educational specialists, focuses on four key competencies: critical thinking, creativity, collaboration, and communication. This approach aims to prepare students effectively for the digital demands of the 21st century, emphasizing the practical application of digital skills in various educational settings [5].

The curriculum is based on a two-dimensional competence model [5]. The topics included in the "Frankfurt Dreieck" are listed in the vertical classification under the corresponding subject headings: (T) technical-media – structures and features of digital, IT, and media systems; (G) social-cultural – social interactions through the use of digital technologies, and (I) interaction-related – interaction in the form of usage, action, and subjectification [7]. The following competencies compose the horizontal line: (1) orientation – analyzing and reflecting on social aspects of media change and digitalization; (2) information – responsible handling of data, information, and information systems; (3) communication – communicating and cooperating using media systems, (4) production – creating and publishing digital content, designing algorithms, and creating software programs, (5) interaction – responsible use of offers and options of a digital world.

The content of the curriculum for secondary school has been split into the following four grades [11,12]:
5th grade:
(T) IPO model, search engines, personal data protection, algorithms, hardware
(G) Understanding digital vs. analog, personalized searches, online teamwork, content presentation, media evolution
(I) Evaluate personal internet use, research skills, source assessment, data management, basic data calculations, text processing, problem-solving with help systems

6th grade:
(T) Tech accessibility, data management, internet basics, coding, hardware/software, networking
(G) Media production, software selection, digital communication, opinion dynamics, copyright
(I) Digital life balance, social media, content creation, digital health/ecology

7th grade:
(T) Tech applications in society, AI, cloud systems, Computational Thinking, embedded computer systems
(G) Media behavior, search routines, privacy vs. openness, digital culture, digitalization's ecological impact
(I) Everyday tech reflection, efficient information search, pattern recognition, crowd-sourcing, digital identity, adapting software, cybersecurity

8th grade:
(T) AI's limits, data security, network protocols, software development, software/hardware differentiation, encryption

(G) Digital attitudes, data privacy, manipulation, civil society digital participation
(I) Digital norms, content management, ethical communication, programming, technical limitations awareness, digital autonomy

Currently, the assignment of educators to teach digital education is managed by school administrators, mainly due to the absence of formally certified digital education teachers. This practice is also common in other subjects facing teacher shortages, particularly in "Mittelschulen" (Compulsory Secondary Schools), although it is less prevalent in "Allgemeinbildende Höhere Schulen" (AHS, Academic Secondary Schools). The initial cohorts of teachers enrolled in university college training programs tailored to digital education are expected to complete their studies in the upcoming years. Additionally, universities and university colleges have recently launched Bachelor's and Master's programs for teacher training in computer science & digital education.

2.2 In-Service Teacher Training in Austria

In Austria, the responsibility for ongoing and further teacher education primarily rests with the 14 university colleges of teacher education. These institutions offer comprehensive training courses across various levels, including primary, secondary general, and vocational education. They also provide courses for career changers to qualify as teachers, often in collaboration with public universities.

A specific focus is placed on the in-service training program for digital education, which is available at eleven of the university colleges [4]. This four-semester program contains 30 credits from the European Credit Transfer and Accumulation System (ECTS), blending theoretical and practical learning to cover various topics relevant to digital education. The curriculum, developed collaboratively by universities and colleges, includes both guided and self-study components, ensuring a thorough educational experience. Graduates of this program are expected to gain a deep understanding of the ethical, social, and practical aspects of digital media and technologies. The training aims to equip teachers with skills in programming, computational thinking, and the pedagogical use of digital tools, enhancing their ability to manage diverse and inclusive classrooms effectively. Moreover, the program highlights the importance of data management, network communication, and digital content creation, all critical in fostering a competent digital education environment [16].

The five modules that make up the teacher training program in digital education are: (M1) *Understanding and shaping your own media use*, (M2) *Digitality and Society*, (M3) *Programming*, (M4) *Computer Systems*, and (M5) *Application*. Module one offers eight ECTS in two classes titled *Understanding Media, Shaping Its Use 1* and *2*. Module two, including six ECTS, is divided into two parts: *Social Influences Through Digital Media* and *Project Work on Socially Relevant Influences of Digital Media*. Module three consists of four parts with a total of seven ECTS: *Programming – Basics 1 & 2*, *Programming – Didactics*, and *Programming – Project Work*. With four ECTS, module four covers *Computer Systems: Fundamentals* and *Specialization*. The final module consists of five ECTS worth of courses: *Applied Computer Applications* and *Applied Media Design Including Project Work* [16].

3 Study

3.1 Methodology

The main goal of this study was to explore the landscape of digital education in Austria from the perspective of secondary school teachers. It seeks to uncover their training, needs, preferences for professional development, and the broader context of their demographics and professional motivations.

Initially, the study confirmed whether the participants taught digital education in the current school year to exclude those who did not. A subsequent question assessed whether teachers joined the course by choice or were required to do so. The next part of the survey explored teacher training to determine if participants were already enrolled in a course. The section after that focused on the teachers' perceptions and requirements regarding training. The final part of the survey gathered demographic information such as gender, age group, years of service, school type, subjects taught, and state.

The question "I would also like to say the following" allowed participants to add their opinions. Seventy-four people completed this section. This work focuses on analyzing the qualitative survey data, whereas previous articles have examined the quantitative survey data.

The "LimeSurvey" online tool was used to collect data, provided at no cost by the university and compliant with the General Data Protection Regulation (GDPR). This platform offers full data export capabilities and built-in data analysis features [14].

A qualitative content analysis was conducted on the collected qualitative data, following the seven-step model outlined by Kuckartz and Rädiker (see Figure 1) [13]. This standard provides a comprehensive approach to structured qualitative content analysis. The text is analyzed, organized, and summarized in the initial phase. The next step involves identifying key categories, leading to the first coding round based on these categories. If necessary, sub-categories are created, and a second coding round is performed. The following steps allow for additional analyses, while the final step entails documenting the process and results. This spiral process can be restarted at any point, allowing for iterative refinement [13].

The analysis was done with the help of "MAXQDA" and AI tools. At first, ChatGPT helped translate the comments into English. Then, the comments on the most common words were investigated using MAXQDA. Furthermore, the statements have been categorized into topics, although one comment may be related to multiple categories. Four key codes were established to determine

Fig. 1. Sequence of a qualitative content analysis in seven phases [13] (edited by the authors)

the primary topics teachers mentioned: "Training Quality", "Additional Training Burden", "Curriculum Issues", and "Equipment Shortages". Moreover, the statements were categorized into three codes, reflecting the attitude of the comment: "positive", "neutral", and "negative".

The fourth and final part of the review involved finding potential arguments by fusing qualitative and quantitative analyses.

3.2 Quantitative Results

Although this paper's primary focus is on analyzing qualitative results, it is essential to summarize the quantitative results currently in publication. This integration is crucial as it provides a comprehensive basis for the subsequent discussion, synthesizing insights from the study's quantitative and qualitative aspects.

The quantitative results show that secondary public school instructors in Austria enjoy teaching digital education, primarily due to individual choices. Most educators believe integrating digital literacy into the curriculum is essential, underlining their responsibility to prepare students for an increasingly digitally connected world. The provision of accessible and adaptable training modalities that respect teachers' professional and private responsibilities is a

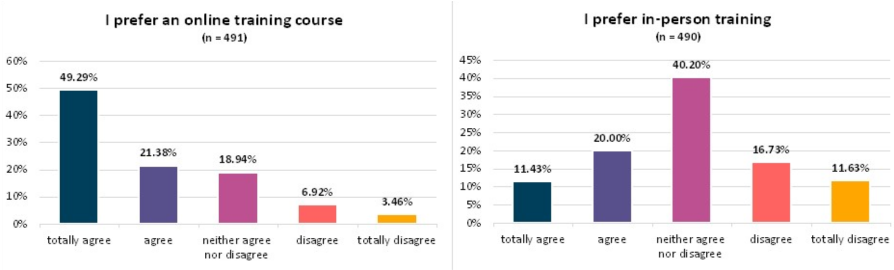


Fig. 2. I prefer online training vs. I prefer in-person training (n = 490)

prerequisite for their desire to engage in extended professional development (see Figures 2 and 3).

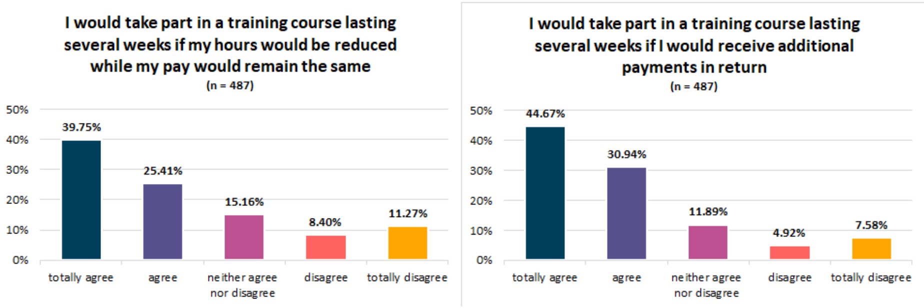


Fig. 3. I would take part in a training course lasting several weeks if my hours would be reduced while my pay would remain the same, I would take part in a training course lasting several weeks if I would receive additional payments in return (n = 487)

3.3 Qualitative Results

The voluntary comment section was completed by 73 out of 578 (12.6%) participants. The text area had an average character count of about 278; the most extended text was 224 words, and the shortest comment was only six words long.

Taking a look at the category-based analysis of the comments (see Figure 4 left), "Training Quality", positive and negative, was mentioned most often with 26 comments (48.1%). Teachers named "implementation is sometimes inadequate", "course is far too theoretical and not practical enough", "make working life as a teacher easier", "not in the least useful for lessons", "time-consuming course", "many units seemed boring", "good and bad training courses", "no

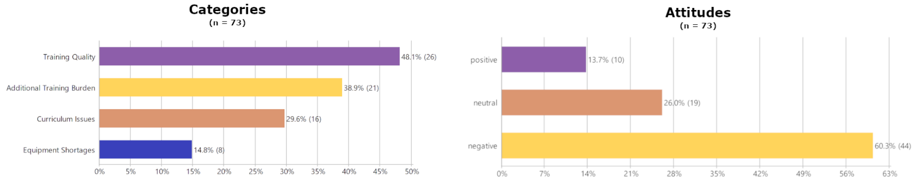


Fig. 4. Left: category-based analysis of comments (n = 73), right: assigned attitude of the comments (n = 73)

didactics in course", "no preparation of the speakers", "absolutely unsatisfactory", "speakers and the content supported me . . . and prepared me extremely well", "lack of practical and differentiated training", "prevailing chaos" amongst others. Twenty-one (38.9%) stated "Additional Training Burdens" that represent the extra workload or stress on individuals due to training requirements. Comments like "all training days take place during the time when there are no lessons", "I absolutely need the weekend to recover", "university is very far", "teachers can't just do everything for free", "too much is already required of me", "poor compatibility with my teaching" was added. Among the 73 comments, 16 (29.6%) mentioned concerns related to the content, structure, or relevance of the curriculum, stating "not yet age-appropriate or too complex", "too many subject areas", "curriculum is rather vague", "curriculum did not take into account the resources", "digital education should start later", "curriculum is basically just a lot of blah blah without any real content", "highly recommended to revise the curriculum", amongst others. The last category, "Equipment Shortages", indicates a lack of necessary tools or technology in school and was found eight (14.8%) times, highlighting topics like "I don't even have WiFi", "we sometimes have to teach without equipment", "no projector or whiteboard, no WiFi", "one projector available for 14 classes", "lack of financial resources".

Considering the overall attitude of the comments, ten (13.7%) could be identified as "positive", 19 (26%) as "neutral", and 44 (60.3%) as "negative" (see Figure 4 right).

The code-relations-graph in Figure 5 highlights these issues' deep interconnectedness. For example, the topics "Additional Training Burden" and "Training Quality" strongly connect (28 and 36) to negative associated comments. The node "Curriculum Issues" is linked to nearly all other ones, underscoring that curriculum problems could have widespread implications on all other aspects of the training environment. "Equipment Shortages" is directly linked to "Training Quality" and "Additional Training Burden", indicating that the lack of proper equipment in schools could be a critical bottleneck affecting the overall quality of training and resource utilization. Overall, this model serves as a diagnostic tool to visualize the complex interplay of multiple factors within the training of teachers.

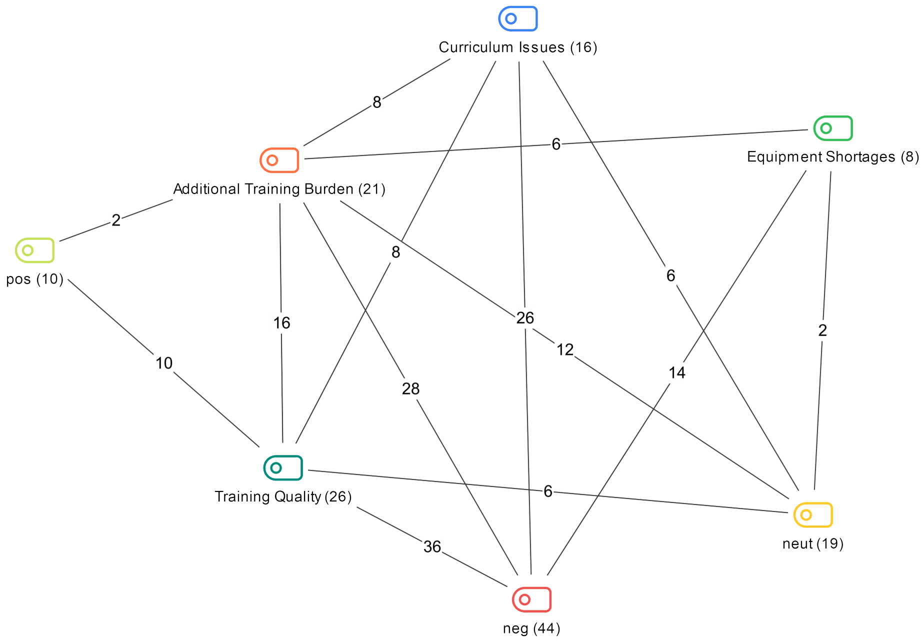


Fig. 5. Code-Relations-Model (n = 73)

3.4 Discussion

The quantitative data highlights a strong preference among teachers for engaging in digital education, emphasizing the need for accessible and flexible training modalities. This preference underscores the importance of developing training programs that respect teachers' professional and personal responsibilities. For instance, many educators expressed a willingness to participate in training if it were more aligned with their schedules and provided actual motivations such as reduced teaching hours or additional compensation. On the qualitative side, the analysis of open-ended responses reveals deeper concerns and specific challenges that teachers face. The most frequently mentioned issue was the quality of training, with many educators criticizing the lack of practical relevance and the excessive theoretical focus of current programs. Comments such as "course is far too theoretical and not practical enough" and "lack of practical and differentiated training" illustrate the need for more hands-on content in training.

As the survey relied on volunteer samples, this method comes with the risk that some potential participants might choose not to participate. Furthermore, it is known that most people fill out the comment section of surveys when they have something negative to add, which could also influence the fact that most comments have been categorized as "negative".

Many teachers reported insufficient technological resources in their schools, such as WiFi, projectors, and other essential tools. This lack of equipment hinders the effective delivery of digital education and adds to the frustration of teachers trying to implement new training methodologies without adequate support.

4 Conclusion and Outlook

The presented study provides a comprehensive understanding of the current state of digital education training among Austrian secondary school teachers. Integrating quantitative and qualitative data highlights the preferences, motivations, and challenges educators face in adapting to the new curriculum.

The qualitative mainly enrich the quantitative data by understanding why specific training modalities are favored – such as the need for practical, relevant training that respects teachers' time – and what specific aspects of digital education are valued or seen as lacking. Furthermore, the qualitative comments offer a narrative of the quantitative findings related to the challenges in training. While the quantitative data might indicate dissatisfaction or preference trends, the qualitative data voice specific frustrations, highlighting areas such as curriculum issues, training quality, and shortages of necessary equipment. These narratives provide context to the numbers, explaining why dissatisfaction exists and where improvements are most needed.

Still, understanding these connections could help policymakers or administrators prioritize areas for intervention. For educators and trainers, insights into the relationship between training quality and additional training burden might help them design more efficient training sessions that are perceived as less burdensome by participants.

In summary, while teachers are clearly enthusiastic about digital education training, significant barriers need to be addressed. Improving the practical relevance of training, reducing additional burdens, addressing curriculum concerns, and ensuring adequate equipment are essential steps towards enhancing the effectiveness of digital education training programs for teachers. These improvements will increase teacher satisfaction and contribute to better educational outcomes for students.

References

1. Andreitz, I.: Motivation und Lehrer/innen/fortbildung. In: Martinek, D., Hofmann, F., Müller, F. (eds.) Motivierte Lehrperson werden und bleiben, pp. 123–156. Waxmann (2018)
2. Andreitz, I., Müller, F., Wieser, M.: Die Bedeutung der Motivation für die Fort- und Weiterbildung von Lehrkräften. In: Kreis, I., Unterköfler-Klatzer, D. (eds.) Fortbildung Kompakt. Wissenschaftstheoretische und praktische Modelle zur wirksamen Lehrer/innenfortbildung, pp. 103–118. StudienVerlag (2017)

3. BKA: Landesvertragslehrpersonengesetz – Dienstpflichten (2023), `https://www.ris.bka.gv.at/NormDokument.wxe?Abfrage=Bundesnormen&Gesetzesnummer=10008213&Artikel=&Paragraf=8&Anlage=&Uebergangsrecht=`
4. BMBWF: Pädagogische Hochschulen (2018), `https://www.bmbwf.gv.at/Themen/schule/fpp/ph.html`
5. BMBWF: Projekt Lehrpläne NEU - Fachlehrplan Digitale Grundbildung (2022), `https://www.ahs-informatik.com/app/download/10356599585/22-03-14_Fachlehrplan_Digitale+Grundbildung_sek+I.pdf?t=1651008169`
6. Bocconi, S., Chioccariello, A., Kamylis, P., Dagiene, V., Wastiau, P., Engelhardt, K., Earp, J., Horvath, M., Jasute, E., Malagoli, C., Masiulionyte-Dagiene, V., Stupuriene, G.: Reviewing computational thinking in compulsory education. Scientific analysis or review KJ-06-22-069-EN-N (online), Publications Office of the European Union, Luxembourg (2022). https://doi.org/10.2760/126955 (online)
7. Brinda, T., Brüggen, N., Diethelm, I., Knaus, T., Kommer, S., Kopf, C., Missomelius, P., Leschke, R., Tilemann, F., Weich, A.: Frankfurt-Dreieck zur Bildung in der digital vernetzten Welt – Ein interdisziplinäres Modell (2019)
8. EACEA: Structural indicators for monitoring education and training systems in Europe 2023 – Digital competence at school. Publications Office of the European Union (2023). https://doi.org/doi/10.2797/886074
9. European Commission, Directorate-General for Education, Youth, S., Culture, for Economic Co-operation, O., Development, Scheerens, J.: Teachers' professional development Europe in international comparison. Publications Office (2010). https://doi.org/doi/10.2766/63494
10. Hartmann, M., Schratz, M.: Schulleitung als Agentin des Wandels in der autonomen Schulentwicklung. TALIS 2008: Schule als Lernumfeld und Arbeitsplatz. Vertiefende Analysen aus österreichischer Perspektive (2010)
11. Hörmann, C., Schmidthaler, E., Sabitzer, B.: Introducing Digital Education as a Mandatory Subject – The Struggle of the Implementation of a new Curriculum in Austria. Proceedings of the 15th International Conference on Computer Supported Education (CSEDU 2023) (2023)
12. Informatikportal AHS Österreich: Kompetenzraster Digitale Grundbildung (2022), `https://www.ahs-informatik.com/app/download/10375968385/LP-VERORDNUNG-DGB-2022-07-06-KOMPETENZRASTER-ANWENDUNGEN-1-4-LETZTFASSUNG.pdf?t=1669384723`
13. Kuckartz, U., Rädiker, S.: Qualitative Inhaltsanalyse. Methoden, Praxis, Computerunterstützung. Beltz Juventa, 5 edn. (2022)
14. Limesurvey: Free university surveys & questionnaires, `https://www.limesurvey.org/solutions/universities`
15. Müller, F., Kemethofer, D., Andreitz, I., Nachbaur, G., Soukup-Altrichter, K.: Lehrerfort- und Lehrerweiterbildung. Bildungsbericht (2018). https://doi.org/10.17888/nbb2018-2-3, `http://doi.org/10.17888/nbb2018-2-3`
16. Pädagogische Hochschule NÖ: Curriculum Hochschullehrgang Digitale Grundbildung (2023), `https://www.ph-noe.ac.at/fileadmin/lehrgaenge/720_714_Digitale_Grundbildung.pdf`

Educational Strategies and Methods for Teaching Computational Thinking

Process of Creating and Evaluating Workbooks as a Resource for the Structured Teaching of Computational Thinking at Primary School

Taina Lehtimäki[0009−0009−9095−8848], Rosemary Monahan[0000−0003−3886−4675], Kevin Casey[0000−0001−7245−9572], Thomas J. Naughton[0000−0002−5330−5428], and Aidan Mooney[0000−0001−8945−0697]

Department of Computer Science, Maynooth University, Maynooth, Ireland
`{Taina.Lehtimaki,Rosemary.Monahan,Kevin.Casey,`
`Tom.Naughton,Aidan.Mooney}@mu.ie`

Abstract. We report on a process of creating and evaluating a set of computational thinking (CT) workbooks for use in primary school. This resource was requested by our teacher feedback group for the structured teaching of CT. Our workbooks extend tasks from the Bebras international computational thinking initiative. The process of creating the three workbook set for different age groups has evolved to use co-creation with teachers. As an initial step, we created a pilot workbook with ten lessons for 3rd through 6th class students (approx. 8 to 12 year olds) from which to gain teacher feedback. A subset of the teachers from our community of practice (12 participants) volunteered to complete a pilot questionnaire about what age/class we should focus on, quality of the content, and whether the workbook was useful for teachers and interesting for pupils. The pilot workbook/questionnaire informed our work on the three workbook set and the main feedback questionnaire. This paper details our full process and what we have learned from our experiences of creating and evaluating new CT resources for primary schools.

Keywords: Computational thinking · Teaching resources · Unplugged · Primary school · K-12 education · Bebras.

1 Introduction

In 2020, computer science (CS) was introduced as a Leaving Certificate (formal end of high school state exam) subject in Ireland. Additionally, a short course in computer programming which requires 100 hours of student engagement, has been available since 2017 for the Junior Cycle (12 to 15 year olds). However, despite these subjects in secondary school there is no formal CS curriculum in primary schools. The organisation responsible for curriculum decisions is the National Council for Curriculum and Assessment (NCCA). It recognises the need for computational thinking (CT) [27] and programming to be taught at primary school level, and although not recommending CS as a full primary level subject, it does recommend [17] that mathematics and science are the most appropriate locations for introducing CS into the classroom. In the most recent Mathematics primary level curriculum, four processes of how children learn, called elements, were identified. One element relates to *Applying and problem-solving*, which has a reference to students developing and sharing their CT skills [18]. This is in line with many other countries where CS and in particular CT are embedded in primary schools [2].

There is no formal CS/CT training for primary teachers, and no CT lessons in primary school textbooks. However, primary teachers appreciate the value in teaching CT as an approach to problem solving, either in the context of mathematics or science, or during teacher discretionary time. Many teachers believe that it is only a matter of time before CT appears formally on the curriculum, and they want to get a started early.

Workbooks: Our workbooks, co-created with teachers, extend tasks from the Bebras international computational thinking initiative [6]. Our project involves the creation of a set of three CT workbooks for different ages. Each lesson in the workbook consists of a Bebras-style task, and a "second page", which is an additional activity that allows students to practice the particular CT skills illustrated with that Bebras task. Each lesson includes comprehensive teacher notes comprising a lesson plan, sample solutions, differentiation suggestions, extension activities, curriculum links, and links to CT.

In advance of creating the three workbook set, to understand our audience, we created a single pilot CT workbook with ten lessons for a broad range from third to sixth class (approximately 8 to 12 year olds) to get initial teacher feedback. We collected teacher feedback about what age/class we should focus on, the quality of the content, was the workbook useful for teachers, and was it interesting for pupils. We also collected detailed feedback on each of the ten lessons.

Evaluation: CS and CT are not on the primary school curriculum in Ireland. Therefore, in order to elicit opinions from teachers with as much knowledge as possible about CT, we invited to take part in our study teachers who had previously used CT resources that we had created. A total of 264 teachers from our community of practice answered our call to participate in the study, and all received printed copies of the pilot workbook for their classrooms. A subset of the teachers (12 participants) volunteered to complete a pilot questionnaire for the purposes of determining the suitability of specific tasks for different age groups, and in order for us to gain experience for the final questionnaire that asks the main body of teachers to evaluate the workbook.

This pilot questionnaire revealed some interesting results, such as that there is no universally agreed task that is definitely too easy for any age. Results also suggest strongest interest from 1^{st} through 6^{th} class for such workbooks. As a result of feedback, we have adopted a principle of not titling the workbook with specific ages/levels, due to the wide range of ages where the workbook can used. The results of the pilot questionnaire informed our work to finalise of the main feedback questionnaire, for example to include free form questions to better capture the range of responses to how well our workbook answers/fulfils the needs/expectations of the teachers.

In this paper, we report on the process of creating the three workbook set, the results of the feedback (pilot questionnaire) from the pilot workbook, the results of the feedback from the associated teacher notes, and what we have learned from our experiences of creating and evaluating new CT resources.

2 Related work

Workbooks are a key feature in education at primary school level in Ireland. The central aim of a workbook is to help to create a natural period of thinking for the students in solving theoretical tasks through reinforcement, practice, and consolidation. Utami et al. [25] found that the use of a workbook gives beneficial impact on students' learning since it can be a source of learning in addition to the teacher's explanation. These workbooks allow students to understand material with simple context and various methods of practice [20]. Preliminary evaluations of a higher education workbook to teach core concepts of computer programming suggest that it has fostered an interest in practical hands-on activities and collaborative work among students [22].

In previous work [12], we documented the co-creation of the teacher notes (lesson plans). This involved third-level CS academics co-creating resources with in-service and pre-service teachers during workshops. These teachers tend to have no prior CS or CT knowledge and thus are newly exposed of the material. One perceived benefit of pre-service teachers being involved in the curriculum co-creation process is that they can begin to think and practice differently and a shift in their metacognitive understanding of learning is often experienced [5,15].

Co-creating motivates learners by increasing their sense of ownership and engagement in the teaching and learning process [5]. Co-creation and partnership share many common values, including shared respect, shared decision-making, negotiation, valuing all perspectives, and shared responsibility [4]. Co-creation allows learners to develop knowledge and skills through their engagement with new concepts and through their experiences with staff and their peers [24]. The co-creation method, rooted in the principles of constructivism [3], which has links to Piaget's theory of intellectual development [19], empowers students to gain knowledge through interactions with an expert who evaluates the differences between the learner's existing knowledge and their capacity for learning [26].

Saito-Stehberger et al. [21] suggest that a strong foundation in CT is required in early education to allow students to develop an instinctive CT perspective of the world. They note that CT instruction is needed in primary school but it is hampered by the shortage of teachers qualified in, and interested in CT. In their work, when modifying a CT curriculum for novice teachers and language learners, the use of students' workbooks was seen as critical.

Cognitive load theory [9] acknowledges that meaningful learning occurs when cognitive processing does not exceed the learner's available cognitive capacity. Saito-Stehberger [21] attempted to reduce the cognitive load on learners, when

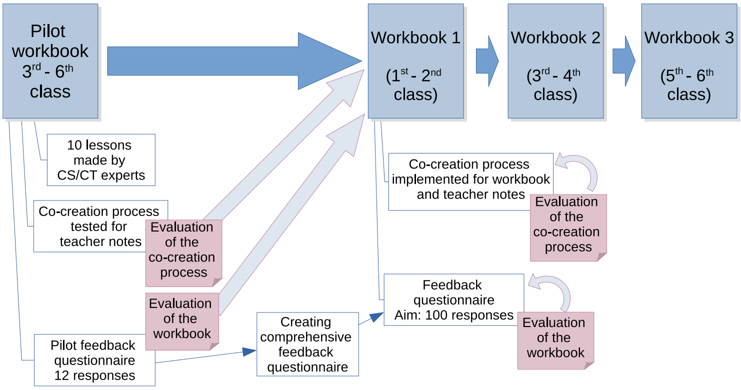


Fig. 1. An overview of the whole process of a pilot workbook followed by a three workbook set. The detail of the process for each workbook 1 through 3 is shown in Fig. 2.

developing their workbook by making changes around reducing ambiguity and wordiness, ensuring simpler sentence structure and eliminating unnecessary detail. They also ensured consistent activity sequences and headings were present throughout, allowing for familiarity to be developed by the learners. The starting point for our workbooks, namely Bebras tasks, ensure that the cognitive load is minimal for the child to understand the task, due to the well-established iterative process involved in creating these tasks [8]. Furthermore, the consistent usage in the workbook of a Bebras task followed by a second page of related activities allows children to quickly become familiar with the structure.

CS activities such as the Computer Science Unplugged project [1] have seen widespread interest from educators worldwide. Besides outreach, such activities are present in the primary school curriculum of many countries and are recommended in the ACM K-12 curriculum. Many are hands-on activities, but many are also suitable for direct inclusion in workbooks. Shimabuku at al. [23] produced a workbook that teaches programming concepts such as a control structure, a data structure, and an algorithm, using unplugged activities similar to Bebras tasks. They found that primary school pupils could understand such programming concepts using these activities. Learning by doing using workbooks allows students to learn in a more informal and supplementary fashion, and this learning by doing paradigm has been shown to be effective in numerous studies related to CT, including ones by Margaria [16] and Gossen et al. [10].

3 Process of creating and evaluating CT workbooks

Our process of workbook creation and evaluation is shown in Fig. 1. It shows the production and evaluation of a pilot workbook and a three workbook set. The pilot workbook was produced by our research team without direct teacher involvement (available in electronic form from https://pact.cs.nuim.ie/workbooks). Se-

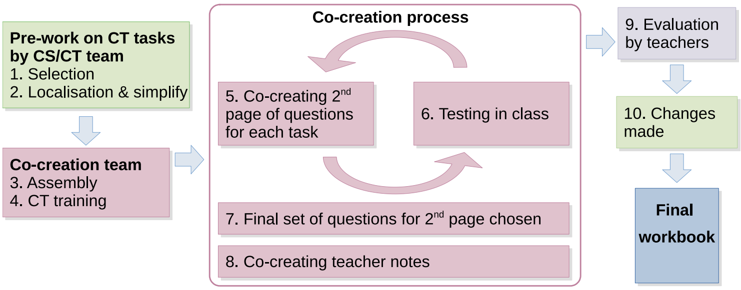


Fig. 2. The detail of the identical process used for each workbook in the three workbook set, where the numbering corresponds to the numbered explanations in Sect. 3.

lecting tasks used in the workbook was based on our experience using particular Bebras tasks during teacher training workshops [12], CT school visits [13], and feedback from teachers about seasonal CT resources provided online [14]. For selecting tasks to cover a variety of CT concepts (see Table 1 in the appendix), we used the set of CT concepts defined for our secondary school resource [11].

Co-creation of our resources was first introduced [12] for the pilot workbook's teacher notes (also available from https://pact.cs.nuim.ie/workbooks). Through involving experts from multiple domains we increased productivity, and have evolved our process to use it for each subsequent workbook and teacher notes. The pilot workbook was evaluated by 12 teachers via questionnaires investigating how they used the resources (results in Sect. 4). This evaluation informs our development of the three workbook set, and the choice of questions for our final questionnaire.

The creation and evaluation process for each of the three workbooks in the set is identical (see Fig. 2) and comprises the following steps:

1. A CS/CT expert pre-selected a superset of tasks based on
    - covering a range of CT topics/concepts,
    - indirect recommendations from the international Bebras community of teachers, education experts, and CS experts from the list of tasks selected for the Bebras challenge in multiple countries,
    - reasonable possibilities for an engaging second page in the workbook,
    - possibility of variants at different levels of difficulty (to fit tasks at different ends, or for the second page in the workbook), and
    - possibilities for a wide variety of extension activities in the teacher notes.
2. The narrative, language, and graphics used in tasks have been carefully prepared by the Bebras community to be easily localisable and translatable. However, we simplify the language and introduce local features (story, graphics, names of characters) to make it more engaging for local schoolchildren.
3. A diverse cohort of co-creators (primary and secondary pre-service teachers) is assembled, representative of the target groups for the workbook.

4. These teachers are trained in the topics/concepts of CT so they can distinguish CT from problem solving techniques such as lateral thinking, pure arithmetic, and common sense (and other non-mathematical approaches). The CT concepts covered in the training are documented in [11].
5. The development cycle for the superset of tasks and their second pages begins with off-line work individually or in pairs, for a group meeting of education domain experts and computer science subject experts each week.
6. Since co-creators have regular contact with classes during term time, they are encouraged to try out unpolished versions of tasks and give immediate feedback that is incorporated into the development cycle. In addition, we deliver workbook tasks in the course of our regular programme of free CT school visits throughout the country [13].
7. A subset of the tasks is chosen, based on how engaging they are for children, and ensuring a range of CT topics/concepts are covered. Then tasks are ordered, based on difficulty.
8. A comprehensive set of teacher notes is developed by the same co-creation team. Teacher notes include: lesson plan, sample solutions, extension activities, an explanation of links to CT topics/concepts (similar to the examples in [7]), and links to the primary mathematics and science curriculum.
9. A draft of the workbook with the teacher notes is given to a set of in-service teachers (with and without CT experience) for their feedback.
10. The results from a feedback questionnaire are analysed and documented for the next workbook. Recommended changes are made to the workbook.

4 Evaluating workbooks

In this section, we present a summary of the responses we obtained from surveying teachers in relation to their use of workbooks. The motivations for the survey were to uncover how the workbooks were being employed, importance and suitability of these type of CT lessons for range of classes, quality of the workbook, was the workbook useful and interesting, and the teacher preferences for the different lessons. Our aim was to get responses to at least to cover classes from 3rd through 6th (ages from 8 to 12 years). Twelve teachers responded. They had used the workbooks with total of 17 classes. The distribution of classes was from 2nd class of primary school to 1st year of secondary school (ages from 7 to 13 years). This distribution covered our target well. The majority of teachers had used the workbook with 6th class (for full distribution see Fig. 8 in Appendix).

Importance of computational thinking workbook by age (Fig. 3) The results for this question show that there was a very clear trend of increasing importance as students get older. This supports our plan to produce our three workbooks aimed from 1st class to 6th class. The results indicate significant interest among teachers, even at 1st and 2nd class. Perceived importance at a very early stage (infant classes) is not quite as marked, which is understandable. Nonetheless, the majority of respondents thought this type of workbook was important or

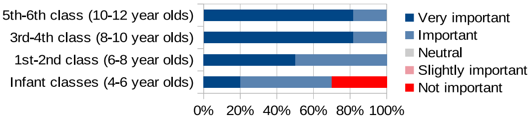


Fig. 3. How important is it to offer this type of computational thinking workbook to particular primary school classes, in your opinion?

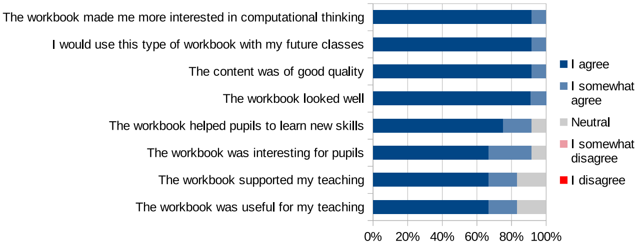


Fig. 4. We seek your opinions about the computational thinking workbook. Do you agree or disagree with the following statements?

very important - despite the challenges at that level (reading ability, teamwork/communication skills). This motivates us to create more tailored materials for these younger classes.

Quality/relevance of workbook (Fig. 4) This series of statements assessed the perceived quality of the workbook, its capacity to generate more interest in CT, whether it taught new skills, and how useful it was. Broadly speaking, the results were positive with over 90 percent of teachers agreeing that the workbook made them more interested in CT and that they would use similar resources in the future. Given that CT material is not yet on the curriculum, the responses to statements such as *The workbook supported my teaching* support the argument that CT is useful across many subject areas.

The difference between the responses to the first four statements and second four statements is interesting. In the main, the first four statements enquire about the teachers' opinions of the workbooks as academic resources, while the second four relate to the workbooks as teaching aids. While the difference is marginal, it appears there might be scope to improve the workbooks as a teaching resource.

Computational thinking lesson suitability for different classes (Fig. 5) This survey question was included to help us address the age appropriateness of individual lessons within the workbook. The results showed that some tasks were perceived as being appropriate for a wide range of ages (e.g. Pearl Bracelet, Passwords), while others were perceived as being suitable for a narrower range of ages (e.g. Footprints, Car Transportation). Again, it's worth noting that all tasks considered as suitable for younger age groups, seem to also be considered

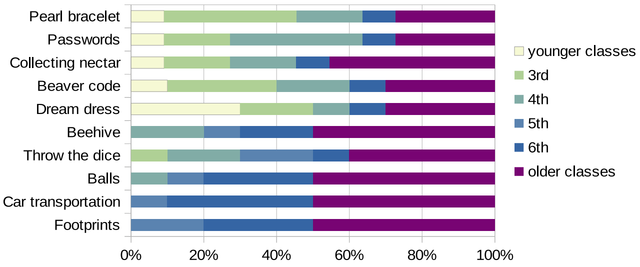


Fig. 5. Which class would each computational thinking lesson be most suitable for, in your opinion? (Each teacher selected for one class per lesson.)

as suitable for older students. For other tasks, we suspect that there is a minimum threshold age for students to understand the question posed in a lesson, after which it is useful for all levels. We plan to uncover this in future work with more refined feedback. In the meantime, this has implications for how we label the workbook and the associated guidance we provide teachers.

How workbooks were employed Teachers reported using the workbooks in different ways. In response to the open ended question *How did you fit these CT activities into your teaching?* teachers reported using the workbook...

- During maths (seven teachers): “problem solving in maths” (×3), “an introduction to maths lessons”, “unlimited maths possibilities”, “mental arithmetic”, and “Maths Week activity”,
- Outside maths (four teachers): “reading comprehension activity”, “STEM Club”, “orienteering activities”, “with the Active School notices”,
- Between subjects (six teachers): “morning activity” (×3), “transition activities between lessons”, “Busy Break”, and “homework”, and,
- As rewards (three teachers): “fun Friday activity”, “reward for good behaviour”, and “early finisher activity”.

This shows that although CT is not in the primary school curriculum, its inherent multidisciplinary nature and link with mathematics (possibly combined with the versatility of the Bebras task concept), meant that teachers were motivated, and succeeded, to find ways to fit it into their teaching.

5 Feedback on teacher notes

We sought feedback on the accompanying teacher notes from the teachers who used our workbooks. These teacher notes were co-created by computer science and educational experts, and provided common lesson guidelines, focusing on the pupils’ thought processes when interacting with tasks in the workbook: articulating the different ways they have solved a given workbook task and reflecting on their approaches. The notes provide a guide to how a teacher might run the 30 –

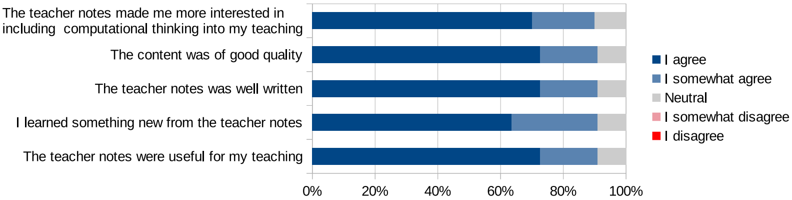


Fig. 6. We seek your opinions of the teacher notes for the CT workbook. Do you agree or disagree with the following statements?

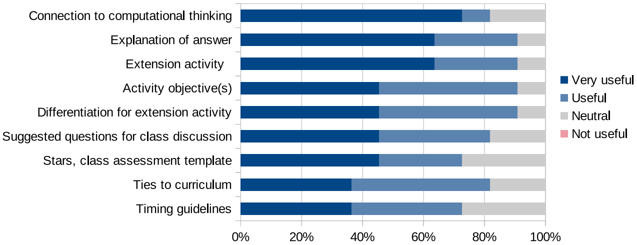


Fig. 7. How useful did you find the following sections of the teacher notes?

40 minute lessons (e.g. pupils working in pairs with the teacher as a facilitator) with associated recommended timings: on introducing the tasks; on the time spent solving the tasks (pair-work encouraged); on additional extension tasks (pair-work encouraged); on teacher-led class discussion where selected pairs can explain how they solved the task to the whole class; on comparison of answers and strategies and on guided questioning to lead a class discussion.

The 12 teachers that provided feedback on the workbooks also answered a questionnaire on the teacher notes. We discuss the responses below.

Opinions on CT workbook teacher notes (Fig. 6): In this figure, we show the teacher opinions of the accompanying teacher notes for the CT workbook. We see that at least 70 percent of teachers agreed that the teacher notes made them more interested in including computational thinking in their teaching, that the content was well written and of good quality, and that the teacher notes were useful for their teaching. Over 60 percent of teachers indicated that they learned something new from the teacher notes.

Teacher notes utility (Fig. 7): This figure shows the responses regarding the utility of each section of the teacher notes. All sections were deemed useful or very useful by over 70 percent of the teachers, with the section describing the connection of the task to computational thinking, the explanation of the answer, and the suggested extension activity reported as very useful in over 60 percent of responses. The weaker positive aspects were timing guidelines and ties to curriculum. Timing guidelines may need further refinement, given the wide range of ages the workbook is being used at. The issue of ties to the curriculum is a natural one, given the current status of CT in the national curriculum.

Extension activities (for details see Fig. 9 in Appendix): In this question, teachers reported on the extension activities that they did, or that they plan to do later. Responses indicate that teachers engaged with the extension activities in all tasks, with at least half engaging immediately with extension activities for four lessons and over 70 percent of teachers planning to engage with all ten lessons. This concurs with the extension activities being reported as being very useful in Fig. 7. Overall, from the responses, we conclude that the teacher notes (especially the answer explanations and extension activities) provided significant added value.

6 Conclusions and future work

Over the last four years, we have been refining a process which we feel would be useful to the wider community for generating and evaluating CT resources, such as workbooks. There are a number of central aspects to the process such as the key role that co-creation with teachers plays in the success of these resources, and the need to incorporate feedback into the process. Certainly, some aspects may be unique to our situation, such as the lack of CT in the primary school curriculum. Involving teachers in the co-creation process was invaluable, but not at every stage; we found that in the initial problem selection phase, working exclusively with CT experts was more valuable to get a good cross-sectional representation of CT tasks, as is documented in the appendix in Table 1.

Feedback and evaluation, as would be expected, are critical parts of our process. Some valuable nuggets of information have been uncovered. For example, it seems that no tasks are considered by teachers to be unsuitable for older age groups of students (i.e. no tasks are too simple). This has implications in terms of how we title and promote the set of workbooks. There is also quite a variation in which tasks are preferred by teachers, with some tasks eliciting a strong preference either positively or negatively, and others not. This deserves further exploration in the next iteration of the feedback questionnaire.

Our workbooks were used with 17 classes from 2^{nd} class of primary school to 1^{st} year of secondary school. It was interesting to see that teachers' opinions on the workbook (e.g. Fig. 4 and Fig. 6) were very similar despite the wide range of classes the workbook was used with.

Our co-creative process to developing suitable CT classroom and supporting materials has been successful. Buy-in from teachers is a natural result of the process and is evidenced by their wide range of uses for the materials in the classroom. Maintaining a feedback channel from teachers as they employ these materials in the classroom is a valuable component, allowing us to evolve our existing material and generate new material that teachers will find useful and have a sense of ownership over. This combination of co-creating materials and the establishment of long-term feedback channels alongside is an approach we have found extremely effective, and would recommend to practitioners with similar goals to ours.

Acknowledgments. We wish to acknowledge the assistance of Robert Cleary and Liam Jordon (both from Maynooth University) during workbook teacher notes/lesson plan co-creation, and all in-service and pre-service teachers that contributed to this study. We acknowledge financial support from the Science Foundation Ireland Discover Programme, under grant numbers 18/DP/5887, 20/DP/8067, and 22/DP/10525.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Bell, T., Alexander, J., Freeman, I., Grimley, M.: Computer science unplugged: school students doing real computing without computers. New Zealand Journal of Applied Computing and Information Technology **13**, 20–29 (2009)
2. Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., Engelhardt, K.: Developing computational thinking in compulsory education-implications for policy and practice. Tech. rep., Joint Research Centre (Seville site) (2016)
3. Bodner, G.M.: Constructivism: a theory of knowledge. Journal of Chemical Education **63**(10), 873–878 (1986). `https://doi.org/10.1021/ed063p873`
4. Bovill, C., Cook-Sather, A., Felten, P.: Students as co-creators of teaching approaches, course design, and curricula: implications for academic developers. International Journal for Academic Development **16**, 133–145 (2011). `https://doi.org/10.1080/1360144X.2011.568690`
5. Cook-Sather, A., Bovill, C., Felten, P.: Engaging Students as Partners in Learning and Teaching: A Guide for Faculty. Jossey-Bass, Hoboken , NJ (2014)
6. Dagienė, V., Futschek, G.: Bebras international contest on informatics and computer literacy: Criteria for good tasks. In: ISSEP 2008. LNCS, vol. 5090, pp. 19–30. Springer (2008). `https://doi.org/10.1007/978-3-540-69924-8_2`
7. Dagienė, V., Sentance, S.: It's computational thinking! Bebras tasks in the curriculum. In: ISSEP 2016. LNCS, vol. 9973, pp. 28–39. Springer (2016). `https://doi.org/10.1007/978-3-319-46747-4_3`
8. Dagienė, V., Stupurienė, G.: Bebras – a sustainable community building model for the concept based learning of informatics and computational thinking. Informatics in Education **15**(1), 25–44 (2016). `https://doi.org/10.15388/infedu.2016.02`
9. Duran, R., Zavgorodniaia, A., Sorva, J.: Cognitive load theory in computing education research: a review. ACM Trans. Comput. Educ. **22**(4) (2022). `https://doi.org/10.1145/3483843`
10. Gossen, F., Kühn, D., Margaria, T., Lamprecht, A.L.: Computational thinking: Learning by doing with the Cinco adventure game tool. In: COMPSAC 2018. pp. 990–999. IEEE (2018). `https://doi.org/10.1109/COMPSAC.2018.00175`
11. Lehtimäki, T., Hamm, J., Mooney, A., Casey, K., Monahan, R., Naughton, T.J.: A computational thinking module for secondary students and pre-service teachers using Bebras-style tasks. In: Proceedings of UKICER 2022. ACM (2022). `https://doi.org/10.1145/3555009.3555030`

12. Lehtimäki, T., Monahan, R., Mooney, A., Casey, K., Naughton, T.J.: Bebras-inspired computational thinking primary school resources co-created by computer science academics and teachers. In: Proceedings of ITiCSE 2022. p. 207–213. ACM (2022). `https://doi.org/10.1145/3502718.3524804`
13. Lehtimäki, T., Monahan, R., Mooney, A., Casey, K., Naughton, T.J.: A computational thinking obstacle course based on Bebras tasks for K-12 schools. In: Proceedings of ITiCSE 2023. p. 478–484. ACM (2023). `https://doi.org/10.1145/3587102.3588775`
14. Lehtimäki, T., Monahan, R., Mooney, A., Casey, K., Naughton, T.J.: Computational thinking resources inspired by Bebras. In: Proceedings of ITiCSE 2023. p. 663. ACM (2023). `https://doi.org/10.1145/3587103.3594203`
15. Lubicz-Nawrocka, T., Bovill, C.: Do students experience transformation through co-creating curriculum in higher education? Teaching in Higher Education **28**(7), 1744–1760 (2023). `https://doi.org/10.1080/13562517.2021.1928060`
16. Margaria, T.: From computational thinking to constructive design with simple models. In: Leveraging Applications of Formal Methods, Verification and Validation. Modeling. pp. 261–278. Springer International Publishing (2018)
17. National Council for Curriculum and Assessment: Final report on the coding in primary schools initiative (2019), `https://ncca.ie/media/4155/primary-coding_final-report-on-the-coding-in-primary-schools-initiative.pdf` [Accessed: (17/01/2024)]
18. National Council for Curriculum and Assessment: Primary mathematics curriculum for primary and special schools (2023), `https://www.curriculumonline.ie/primary/curriculum-areas/mathematics/` [Accessed: (17/01/2024)]
19. Piaget, J.: Piaget's theory. In: Mussen, P.H., Kessen, W. (eds.) Handbook of Child Psychology, vol. I History, Theory, and Methods, pp. 41–102. John Wiley and Sons, New York, 4th edn. (1983)
20. Ristevska, M., Kocoska, J., Gramatkovski, B., Sivakova-Neskovska, D.: The role of workbooks in the learning process in primary schools in the Republic of Macedonia. International Journal of Innovation and Applied Studies **11**(3), 691–698 (2015)
21. Saito-Stehberger, D., Garcia, L., Warschauer, M.: Modifying curriculum for novice computational thinking elementary teachers and English language learners. In: Proceedings of ITiCSE 2021. p. 136–142. ACM (2021). `https://doi.org/10.1145/3430665.3456355`
22. Servin, C.: Programming workbook: A collaborative coding fusion (print and online) for mastering programming fundamentals. In: Proceedings of ITiCSE 2024. p. 789–790. ACM (2024). `https://doi.org/10.1145/3649405.3659523`
23. Shimabuku, M., Nagataki, H., Kanemune, S.: Development and evaluation of programming workbooks designed for elementary school children. In: Proceedings of Innovate Learning Summit 2021. pp. 226–231 (2021), `https://www.learntechlib.org/p/220290`
24. Thiele, T., Homer, D.: 'Trying to open the doors': The co-creation of digital resources for disadvantaged primary school pupils. International Journal for Students as Partners **7**, 65–83 (2023). `https://doi.org/10.15173/ijsap.v7i2.5166`
25. Utami, A., Aminatun, D., Fatriana, N.: Student workbook use: does it still matter to the effectiveness of students' learning? Journal of English Language Teaching and Learning **1**, 7–12 (2020). `https://doi.org/10.33365/jeltl.v1i1.247`
26. Villamil, H.R.: Del constructivismo al construccionismo: implicaciones educativas. Rev. Educ. y Desarro. Soc **2**, 71–89 (2008)
27. Wing, J.M.: Computational thinking. Commun. ACM **49**(3), 33–35 (2006). `https://doi.org/10.1145/1118178.1118215`

A Appendix

The selection of tasks in our workbook hits a broad range of CT concepts, as shown in Table 1. The age distribution of the classes that used the workbook in this study is shown in Fig. 8. The reported usage of extension activities by teachers for different lessons is shown in Fig. 9. Finally, a sample lesson from the workbook, and associated pages for the lesson from the teacher notes, is shown in Fig. 10.

Table 1. Representation of CT concepts in the tasks used in the workbook.

Task	Decomposition	Pattern recognition	Representation	Abstraction	Algorithms	Evaluation	Logic	Generalisation
Pearl bracelet	x	x				x		
Passwords	x	x	x					
Collecting nectar	x				x	x		
Beaver code	x	x	x					
Dream dress	x			x	x		x	
Beehive	x	x				x	x	
Throw the dice	x				x		x	
Balls	x				x			x
Car transportation	x	x	x		x			
Footprints	x	x			x			

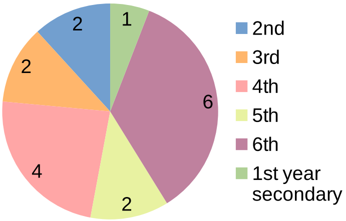


Fig. 8. What classes were the workbooks used with? The numbers in each segment specify how many classes of that age group used the workbook.

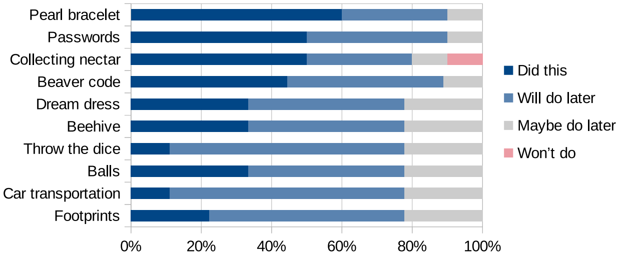
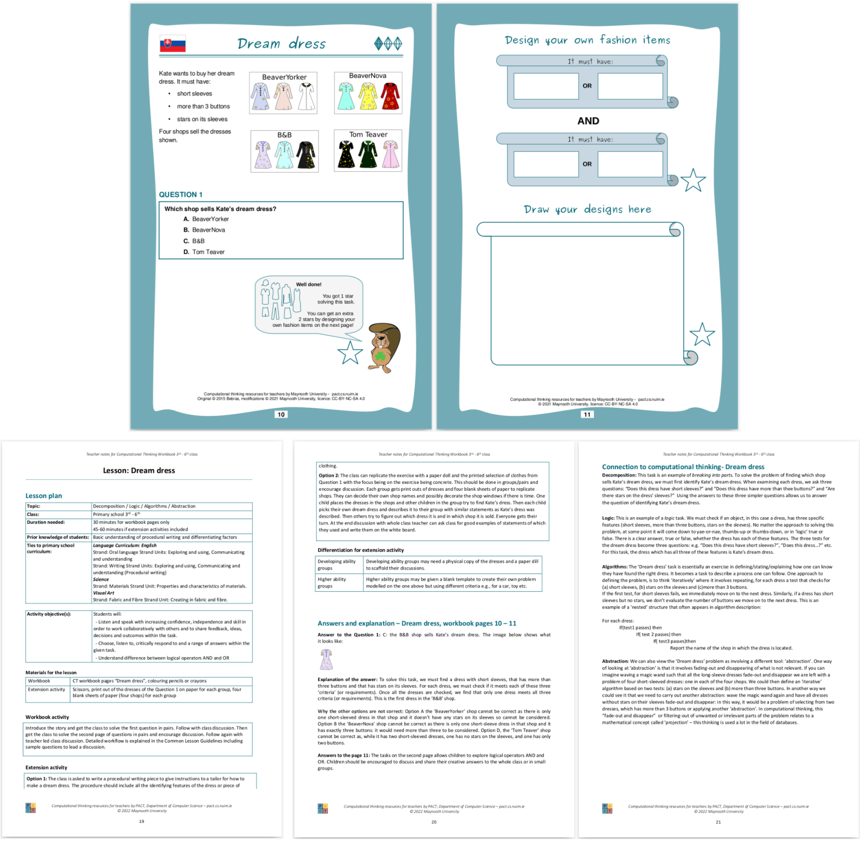


Fig. 9. Which extension activities did you do, or do you plan to do later?



Teacher notes for Computational Thinking Workbook 3rd - 6th class

Lesson: Dream dress

Lesson plan

Topic:	Decomposition / Logic / Algorithms / Abstraction
Class:	Primary school 3rd - 6th
Duration needed:	30 minutes for workbook pages only 45-60 minutes if extension activities included
Prior knowledge of students:	Basic understanding of procedural writing and differentiating factors
Ties to primary school curriculum:	***Language Curriculum: English*** Strand: Oral language Strand Units: Exploring and using, Communicating and understanding Strand: Writing Strand Units: Exploring and using, Communicating and understanding (Procedural writing) ***Science*** Strand: Materials Strand Unit: Properties and characteristics of materials. ***Visual Art*** Strand: Fabric and Fibre Strand Unit: Creating in fabric and fibre.

Activity objective(s):	Students will: - Listen and speak with increasing confidence, independence and skill in order to work collaboratively with others and to share feedback, ideas, decisions and outcomes within the task. - Choose, listen to, critically respond to and a range of answers within the given task. - Understand difference between logical operators AND and OR

Materials for the lesson

Workbook	CT workbook pages "Dream dress", colouring pencils or crayons
Extension activity	Scissors, print out of the dresses of the Question 1 on paper for each group, four blank sheets of paper (four shops) for each group

Workbook activity

Introduce the story and get the class to solve the first question in pairs. Follow with class discussion. Then get the class to solve the second page of questions in pairs and encourage discussion. Follow again with teacher led class discussion. Detailed workflow is explained in the Common Lesson Guidelines including sample questions to lead a discussion.

Extension activity

Option 1: The class is asked to write a procedural writing piece to give instructions to a tailor for how to make a dream dress. The procedure should include all the identifying features of the dress or piece of

Computational thinking resources for teachers by PACT, Department of Computer Science – pact.cs.nuim.ie
© 2022 Maynooth University

19

Teacher notes for Computational Thinking Workbook 3rd - 6th class

clothing.

Option 2: The class can replicate the exercise with a paper doll and the printed selection of clothes from Question 1 with the focus being on the exercise being concrete. This should be done in groups/pairs and encourage discussion. Each group gets print outs of dresses and four blank sheets of paper to replicate shops. They can decide their own shop names and possibly decorate the shop windows if there is time. One child places the dresses in the shops and other children in the group try to find Kate's dress. Then each child picks their own dream dress and describes it to their group with similar statements as Kate's dress was described. Then others try to figure out which dress it is and in which shop it is sold. Everyone gets their turn. At the end discussion with whole class teacher can ask class for good examples of statements of which they used and write them on the white board.

Differentiation for extension activity

Developing ability groups	Developing ability groups may need a physical copy of the dresses and a paper dill to scaffold their discussions.
Higher ability groups	Higher ability groups may be given a blank template to create their own problem modelled on the one above but using different criteria e.g., for a car, toy etc.

Answers and explanation – Dream dress, workbook pages 10 – 11

Answer to the Question 1: C: the B&B shop sells Kate's dream dress. The image below shows what it looks like:

Explanation of the answer: To solve this task, we must find a dress with short sleeves, that has more than three buttons and that has stars on its sleeves. For each dress, we must check if it meets each of these three 'criteria' (or requirements). Once all the dresses are checked, we find that only one dress meets all three criteria (or requirements). This is the first dress in the 'B&B' shop.

Why the other options are not correct: Option A the 'BeaverYorker' shop cannot be correct as there is only one short-sleeved dress in that shop and it doesn't have any stars on its sleeves so cannot be considered. Option B the 'BeaverNova' shop cannot be correct as there is only one short-sleeve dress in that shop and it has exactly three buttons: it would need more than three to be considered. Option D, the 'Tom Teaver' shop cannot be correct as, while it has two short-sleeved dresses, one has no stars on the sleeves, and one has only two buttons.

Answers to the page 11: The tasks on the second page allows children to explore logical operators AND and OR. Children should be encouraged to discuss and share their creative answers to the whole class or in small groups.

Computational thinking resources for teachers by PACT, Department of Computer Science – pact.cs.nuim.ie
© 2022 Maynooth University

20

Teacher notes for Computational Thinking Workbook 3rd - 6th class

Connection to computational thinking- Dream dress

Decomposition: This task is an example of *breaking into parts*. To solve the problem of finding which shop sells Kate's dream dress, we must first identify Kate's dream dress. When examining each dress, we ask three questions: "Does this dress have short sleeves?" and "Does this dress have more than thee buttons?" and "Are there stars on the dress' sleeves?" Using the answers to these three simpler questions allows us to answer the question of identifying Kate's dream dress.

Logic: This is an example of a *logic* task. We must check if an object, in this case a dress, has three specific features (short sleeves, more than three buttons, stars on the sleeves). No matter the approach to solving this problem, at some point it will come down to yae-or-nae, thumbs-up or thumbs-down, or in 'logic' true or false. There is a clear answer, true or false, whether the dress has each of these features. The three tests for the dream dress become three questions: e.g. "Does this dress have short sleeves?", "Does this dress…?" etc. For this task, the dress which has all three of these features is Kate's dream dress.

Algorithms: The 'Dream dress' task is essentially an exercise in defining/stating/explaining how one can know they have found the right dress. It becomes a task to describe a process one can follow. One approach to defining the problem, is to think 'iteratively' where it involves repeating, for each dress a test that checks for (a) short sleeves, (b) stars on the sleeves and (c)more than 3 buttons.
If the first test, for short sleeves fails, we immediately move on to the next dress. Similarly, if a dress has short sleeves but no stars, we don't evaluate the number of buttons we move on to the next dress. This is an example of a 'nested' structure that often appears in algorithm description:

```
For each dress:
    If(test1 passes) then
        If( test 2 passes) then
            If( test3 passes)then
                Report the name of the shop in which the dress is located.
```

Abstraction: We can also view the 'Dream dress' problem as involving a different tool: 'abstraction'. One way of looking at 'abstraction' is that it involves fading-out and disappearing of what is not relevant. If you can imagine waving a magic wand such that all the long-sleeve dresses fade-out and disappear we are left with a problem of four short-sleeved dresses: one in each of the four shops. We could then define an 'iterative' algorithm based on two tests: (a) stars on the sleeves and (b) more than three buttons. In another way we could see it that we need to carry out another abstraction: wave the magic wand again and have all dresses without stars on their sleeves fade-out and disappear: in this way, it would be a problem of selecting from two dresses, which has more than 3 buttons or applying another 'abstraction'. In computational thinking, this "fade-out and disappear" or filtering-out of unwanted or irrelevant parts of the problem relates to a mathematical concept called 'projection' – this thinking is used a lot in the field of databases.

Computational thinking resources for teachers by PACT, Department of Computer Science – pact.cs.nuim.ie
© 2022 Maynooth University

21

Fig. 10. Sample lesson "Dream Dress" with the associated teacher notes.

Place of "crafts" in the education of Algebraic and Computational Thinking

Sarmasági, Pál[0009-0002-7683-8515], Rumbus, Anikó[0009-0003-1300-9426],
Pluhár, Zsuzsa[0000-0003-2688-4652] and Margitay-Becht, András[0009-0005-9251-9591]

Eötvös Loránd University, Budapest, Hungary
psarmasagi@inf.elte.hu

Abstract. It is widely recognized in educational research that the acquisition of mathematical and computational skills goes beyond mere numeracy. Intelligence, observation, memory, motivation, effective pedagogical methods, and a quality teaching approach are key to the learning process. Our research has focused on the learning characteristics of 9-14 year olds, with particular attention to the importance of empirical experience at this age. We identified a number of pedagogical methods in the literature that promote the learning effectiveness of this age group, including the "CS Unplugged" for teaching computer science, which aims to provide instruction without digital tools. Furthermore, in our survey of six countries, we found that unplugged methods are widely used in lower grades (grades 3-4), but are less common in grades 5-8, which may limit students' understanding of the interrelationships between systems.

Keywords: Algebraic Thinking, Computational Thinking, STEM education, empirical experience, National Curriculum.

1 Introduction

Nowadays, the labor market is defined by digital transformation, industry 4.0, and AI, so it requires well-educated IT professionals. It is confirmed by several sources [21], for instance, a search on the internet for the most popular or in-demand IT positions [19]. The substantial evolution of Information Technology (IT) manifests a discernible impact on the realm of education. These specialists are trained in Computer Science programs at universities, and the common, essential part of their skills is Computational Thinking, which interacts with Algebraic Thinking. In a different approach, the STEM umbrella terms were used in the early 1990s for the science, technology, engineering, and mathematics subjects. This new term was reasoned by the importance of the subject behind STEM, which education specialists recognized. Computational Thinking (CT) and Algebraic Thinking (AT) are essential parts of STEM subjects, so the development of AT and CT helps in learning STEM subjects. The escalating demand for professions necessitating competence in scientific knowledge coupled with IT proficiency requests a paradigm shift in education, the cultivation of proficient professionals shouldn’t be confined to the university level; rather, the initiation of developmental processes at an early age is imperative. These reasoned the creation of CT&MATHable project, which investigates the ways of

CT and AT development at primary school, with the finance of Erasmus+ by the EU [6]. The project examined how the national curriculum develops CT and AT components in each country, what are the common parts and characteristics, and how teaching AT and CT components can be improved. Based on the curricula analyses, qualitative research, a workshop, and a small-size quantitative research this article points out, that experiential learning, empiric practice, and craft activities are important parts of a primary school's math and digital culture education.

2 Concepts

Computational Thinking (CT) is an educational expression by Seymour Papert [13], who developed the Logo educational language and it was popularized by Jeanette Wing [20]. Nowadays, several approaches and definitions of CT describe the concept. Based on our interpretation CT is a cognitive skill set essential for problem-solving and navigating the complexities of the digital age. Computational Thinking transcends coding proficiency, emphasizing logical reasoning, abstraction, and algorithmic problem-solving applicable across various disciplines.

In math education, Algebraic Thinking (AT) is as important as Computational Thinking is in computer science. AT is a specific part of mathematical thinking that focuses on understanding and manipulating variables, expressions, and equations, while Mathematical Thinking is an encompassing skill such as problem-solving, logical reasoning, abstraction, generalization, recognizing, and applying mathematical concepts, structures, and relationships in various contexts. AT is also an educational concept, and its origin is not so clear. The history of math and algebra is measurable in millennia, however, the education methodology was strengthened only in the XX. century, so the concept of Algebraic Thinking appeared in the middle of the last century in published articles. More relevant are the ones published after the 1990s, when AT can be compared with CT. Based on the works of Lins [12], Kieran [9], Kriegler [10], Stramel [15], Blanton, and Kaput [4], we understand the components of AT as the following: Relational thinking, which contains equality, and inequality; Pattern recognition, which is part of most learning processes; generalization, and its base, abstraction; numbers and operations; mathematical language, which includes symbols; and problem-solving.

Algebraic Thinking helps students with the skills to solve abstract problems and the development of mathematical intuition. Understanding symbolic representations, equations, and algebraic structures enhances the cultivation of analytical thinking and problem-solving skills. The synergy between Computational and Algebraic Thinking is particularly relevant in educational contexts.

The expression Learning path has several meanings that should be clarified. The most frequent usage of this expression is the flexible learning paths of Learning Management Systems (LMS). The role of the Learning path is important for instance in the case of flexible lifelong learning requires comparability and exchangeability of courses, programs, and other types of learning actions both in a national and international context. Another frequent usage of the learning path expression occurs the special needs students, in the case of gifted or dropped out, who need individual learning paths based on their special skills or the lack of the proper skills. Learning path expression can be used to evaluate the different schools within a national school system, by tracking the way of students. A third meaning of the learning path expression is the possible teaching ways and the topic order in the curriculum. Which topics are in the curriculum, what is their order, and how can teachers build the syllabus of a given class topic by topic for the best learning outcomes? In our context, we used Learning Path (LP) in this last meaning.

Experiential education is a philosophy of education that describes the process that occurs between a teacher and student and infuses direct experience with the learning environment and content. This concept is distinct from experiential learning, however experiential learning is a subfield and operates under the methodologies associated with experiential education [8]. Experience is an important part of everyday life nowadays, like user experience in application development, customer experience in marketing communication, and gamification in human resource development. And it is an important part of public education.

Since the 1990s, the concept of Computer Science Unplugged (CS Unplugged) has been actively pursued by professionals. The pioneers of this initiative were Tim Bell, Mike Fellows, and Ian Witten, who first explored the topic. It quickly caught on, resulting in the collection "Computer Science Unplugged: off-line activities and games for all ages", which has been translated into more than 20 languages [7]. The effectiveness of the method has been demonstrated in numerous studies, and one of its key advantages is that it can be used in teaching and learning with little or no prior knowledge [3]. While in the 90's the main benefit was practicing without computers as there were not enough computers, nowadays it helps students to learn to make decisions about whether they need a computer to solve a given problem or not [16].

3 Method and process

The researchers of 8 universities form the project team from the participant 6 countries and their investigation focuses on education, from grades 3 to 8. The second work process explored the common parts of current learning paths in 6 countries' curricula of math and informatics curriculum. The structure of national curricula is different, so for comparison, the curricula had to be consolidated during the processing. The result is that 31 math and 12 informatics topics were defined based on the six countries' curricula. The summarized curriculum list was analyzed statistically and by content, but these analyses are limited because only the curriculum topics were analyzed and the number of hours each topic is taught was not part of this analysis. The statistical analyses confirmed the national curricula are similar, the rate of common topics is over 80 percent. The content analyses explored the main learning paths which are characterized by the repetition of topics at an increasingly higher level.

In-depth interviews were made among math and informatics teachers regarding their daily practice of education. As an exploratory methodology, it drove the team's attention to the empiric experience. Using the results of qualitative research a workshop was organized, where the participants met several empirical, craft exercises, which develop algebraic and computational thinking. At the end of the workshop, teachers responded to a questionnaire, that was analyzed as quantitative research.

4 Results

The results of statistical analyses were presented at The 16th International Conference on Informatics in Schools (ISSEP 2023) [14].

The content analysis confirmed there is no relevant difference among the country curricula regarding the foundation of math. Math education starts with playing, when the students stack, sort, group, and classify simple but distinguishable objects. These playful experiences prepare the students to the language of mathematics and to learn the basic concepts. The categorization, comparison, problem solving, and sort topics are covered by this introductory and foundational process, as well as the relations and relationships that introduce student to relational thinking and problem-solving.

The process of education is similar in the case of other topics in lower grades. For the geometry and measurements introduction is also required the presentation objects, like the using of poligons, and polyhedrons or weights and measuring cups. Students take in their hands these objects to get the necessary experience and empirical knowledge during the craft activity.

There were involved five teachers into the in-depth interviews, two from lower-primary schools, two from higher-primary school and a teacher from secondary schools. The respondent teachers from lower-primary school emphasid that the arithmeti-cal skills require more time and more practice. When the students have an established arithmetical skill, then they can explore the relations, patterns and they can generalize. Further information is that subtraction is a difficult operation that requires more attention, multiplication should be approached from the point of view of number sequences, multiplication with 7 is hard for students, and the multiplication table must repeat in every autumn. The lack of experience and concept knowledge blocks the learning of units of measurement. Life has changed, several students arrive at school by car, and they don't walk to the corner shop to buy a half kilogram of bread with cash, so they don't have a concept of distance, weight, or coins. So, this topic also requires a lot of illustration tools, experience, and practice. Additionally, one respondent teacher highlighted the students avoid the thinking and calculation process, and they would like to fill out only multiple-choice tests. These results also drove attention to the importance and necessity of craft, and empirical practice in education.

The workshop aimed to showcase the ongoing project's partial results to the teachers, it promoted the existing system and gave an opportunity to the teachers to try it out and give adequate feedback regarding the state of it at the time of the test. The connected presentations aimed to demonstrate the structure of the project and showcase the possible usage of the developed program within the educational possibilities. A demo was prepared with a set of tasks in the ViLLE system [17], in which each task showed the types and possibilities of the tasks to be created, in addition to the possibilities and diversity of the system. Finally, the partners and the teachers could test the small beaver task-based competition with unplugged activities, a highlight of the event, aimed to engage participants to help develop their algorithmic thinking in a fun way. Participants navigated through several stations, each adorned with colorful and interactive beaver tasks, that were requires craft activities, challenging their creativity and problem-solving skills. Alongside these activities, engaging sessions with micro:bits added a modern twist, offering hands-on exploration of coding and electronics. The participating teachers were very motivated by the playful craft, and practical exercises presented at the workshop.

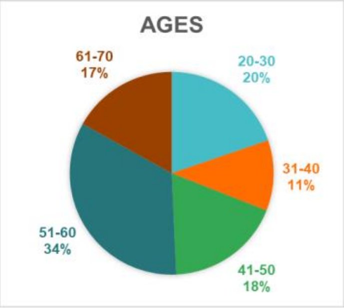


Fig. 1: Respondents' demography distribution by age (n=71) (Source: Own research.)

The questionnaire of quantitative research was filled out by 71 teachers. Most of the teachers were between 40-60 years old (52%) (Fig. 1) – as represented in the Hungarian national education system as well, they teach informatics and math (47%). Most of the teachers (42%, 48%) teach the oldest students (classes 9-10 and 11-12). 90% of the teachers use project-based methods in their education and approximately 84% work with their students on more complex tasks related to everyday life. The ViLLE system presented tasks with more and less interactivity. The teachers typically use drag&drop, pairing tasks in their education, however appr.50% use all kinds of the shown interactivities. The most commonly used are Learning apps, Canva, Kahoot, Redmenta and Mentimeter.

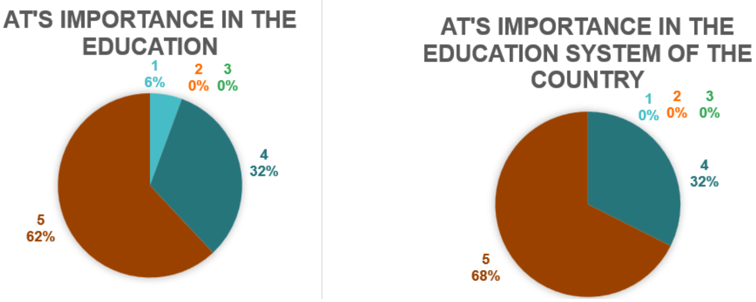


Fig. 2: Percentages of the answers of the questions: How important do you find AT. (n=71) (Source: Own research.)

We asked the teachers how important they find such a project in their everyday education and the education system. They agreed, totally agreed with the importance of such a project not only in their education but in the whole educational system (Fig. 2). Most of the teachers received some new information about Algebraic Thinking (44% totally agreed and 44% agreed with the statement). Only 4 teachers (6%) thought they could not use what they saw here in their teaching process. The others also saw opportunities in the IT unplugged activities and the playful ViLLE tasks

- to combine maths and IT,
- to make students thinks, and
- to approach everyday problems.

The further details of quantitative research focused on math topics and their relevancies, which explored the students' difficulties and mainly confirmed the results of qualitative research.

5 Craft Activities in Computer Science Education

As this project research is confirmed, the empirical, craft activities are general in math education at lower grades. Based on teachers' opinion it must be extended to higher grades at least at the introduction of new topics. Children like playful practices and they are motivated if the teacher sometimes interrupts the frontal teaching and starts a craft activity to get empirical experience in the classroom.

Similar to math teaching, the begining of IT education contains some introductory activities, that prepares students to the learning IT. There are several available CS unplugged tasks and activities, that don't require a computer and motivate students with their playful implementation. The CS unplugged websites [5] for primary school students, and a published book [18] for secondary school students. In addition, the tasks of Bebras Challenge [1] are also useful and widespread in public education.

This method effectively supports offline understanding of concepts, contexts, and processes related to computer science. It does not require digital tools, so it reduces costs, but also makes it more widely accessible and applicable. 9-10-year-olds respond particularly well to playful learning methods, which help them to understand concepts and contexts more easily and keep them motivated. For this age group, conceptualization is often done through the analysis of concrete factual material. Sensory experience is also key, as it allows children to experience, explore, and perceive processes. Although the factuality of sensory experience can sometimes be questioned, the tools can also be used to illustrate real-world problems, which also promotes deeper understanding. Using computer applications can also open up possibilities that would not exist in the real world or would contradict physical laws [11]. The advantages of the CS Unplugged method can be observed here as well. The learning is done through active participation and simulation of processes, from the creation of tools to their use. These are small, easy-to-produce, and cost-effective tools that are available to students after class and can be used at home [2]. Visualizing and re-enacting the processes gives students the opportunity to gain a deeper understanding. Tasks can also be combined with extra-curricular activities to increase students' motivation and promote deeper learning.

In addition, several CS unplugged tasks are suitable to support math education in higher grades. The introductory and practice of numeral systems, place value systems, especially the binary system, or mathematical logic are supported by CS unplugged tasks.

6 Sample exercises

The sample tasks present the new kinds of unplugged activities that were developped by project team, each exercise an age group. These tasks are applicable to develop both thinking skills, either Computational Thinking or Algebraic Thinking.

6.1 Task 1st for grades 3-4

The name of the game: Trueball
Topic: logic in math and informatics
Necessary device(s): ball
Teacher instructions:

- Tell a statement and throw the ball to one of the students.
- The student needs to define the truth value of the statement (telling "true" or "false"). Then the student tells a statement and throws the ball to another student.
- etc.

The statements need to be clearly defined and include information that all students know. Teachers can use a special topic (like geometry, numbers, or any other topic from other subjects). A further variation is when the statements need to include "all", "exists", "not", and "non of" expressions, based on the age group's need.

6.2 Task 2nd for grades 5-6

The name of the game: Binary Boats (based on the Bebras task 2013-JP-04)
Topic: Number system, digits, numbers in math, data representation in informatics
Necessary device(s): lego, or small paper boxes (Fig. 3).
Teacher instructions:

- Build small boats for 8, 4, 2, and 1 figure(s) ("person") from, for example, legos, or paper boxes (fig. 4.)
- Take 1-15 figures at a time. The students need to arrange the figures into the boats by following two rules:
  a) All figures need to be in a boat.
  b) If at least one figure is in one boat, all places in that boat need to be filled with figures i.e. the boat needs to be full.
  c) You can give some hints: "Start with the biggest boat that can be full."
- Continue the activity with several numbers of figures.

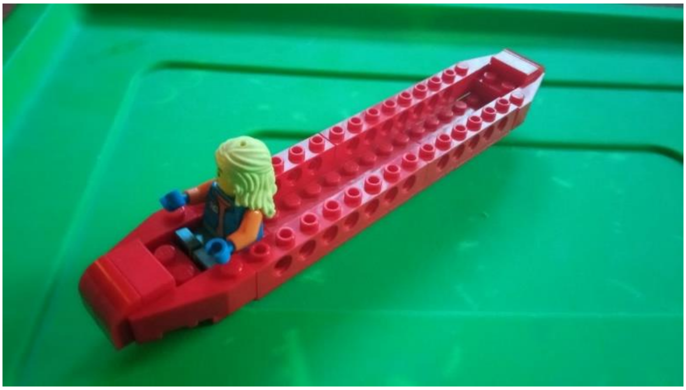


Fig. 3: Percentages of the answers of the questions: How important do you find AT?

Discuss the idea of the number systems and how they can improve the boats using another number system.

A variation of this task is applicable to practice other number systems.

Necessary device(s): Cards and pen

Teacher instructions:

- Prepare cards with dots: 1, 3, 9.
- Give two cards from each one (6 cards at the end) to each student and tell a number from 0 to 26.
- The students need to put the cards face up having as many dots as the number.
- Discuss the solution:
  a) Which strategy was used? For example, starting with the card with the most dots.
  b) How many cards were used?
  c) Why don't students have the 3rd card?
- Continue with a new number! The students can also tell the numbers.

6.3 Task 3rd for grades 7-8

The name of the game: String around nails (based on the Bebras task 2013-JP-04)

Topic: Measurements and units in math, graph, the shortest path in informatics

Necessary device(s): boards, nails and string

Teacher instructions:

- Give boards with 10-15 nails and a string for each group. Nails can be hammered into the board randomly.
- Mark the starting point or students can decide the starting point themselves.
- Hand out the boards and ask the students to find the shortest route possible. The route has to go around each nail and return to the starting point.
- After the students have tested a route, they place a mark on the string to indicate the total length of the route — this way they can recognize which route is the shortest.

6.4 Task 4th for grades 9-12

The name of the game: Book searching
Topic: Numbers theory, LCM, GCD, power, root in math, search in informatics
Necessary device(s): Blank covered book
Teacher instructions: The first time the books are in unsorted order, and students must find a given book, while the cover doesn't contain info. The second time the books are in sorted order and the task is the same. Students count the number of trials before find the given book. The teacher helps to find the proper strategies for both cases, and students make a statistical analysis to approach the number of steps, finally, they define the right expressions for the number of steps.

The tasks presented were carried out with students from grades 3-8, who took part in the Bebras competition results ceremony on 1 June 2023 in Budapest. The students understood the tasks easily and could solve them correctly with minimal help. In their opinion, the tool-supported solutions seemed simpler, more transparent and easier to understand than the classical school solutions. By this I mean that few demonstration tools are used in usual classroom work.

7 Conclusions

The project team compared the mathematics and IT curricula of the six countries during the second phase of work. The consolidation pointed out there is no relevant differences among national curricula, 31 mathematics, and 12 IT topics were identified and analyzed statistically and content-wise. The analyses confirmed that the curricula are more than 80% similar and that the learning pathways are characterized by an increasing repetition of topics. Through in-depth interviews and workshops, we collected empirical data on teachers' teaching practices and conducted a quantitative analysis of teachers' feedback in Hungary. The teaching methods are based on playful learning, which promotes a deeper understanding of concepts and contexts. Teachers stressed that developing arithmetic skills requires more practice and that students find it challenging to master subtraction and multiplication tables.

The importance of craft activities and empirical experiences was highlighted to reinforce conceptual knowledge. The workshop aimed to showcase the partial results of the project, giving teachers the opportunity to test the system and provide feedback. The unplugged exercises were a great success, with teachers unanimously saying that the hands-on, testable, and visual learning method promoted and facilitated understanding. It confirmed the aim of CT and AT development. Both are a cognitive function as a thinking method, to gain these abilities the playful activities and the everyday experience are important.

It is clear from our research that this is an issue that needs to be addressed seriously, as we are facing almost identical problems in the countries we are studying. The transfer of unplugged methods to higher grades could lead to results. In this research, we have not yet obtained concrete results on this, so it is a further task for this project.

8 References

1. Bebras Computing Challenge, https://www.bebraschallenge.org/, last accessed 2024/05/30.
2. Bende, I.: Algoritmusoktatás online oktatási rendszerben. Infodidact 2019, pp. 13-22. (2020).
3. Bell, T., Witten, I. H., Fellows, M.: Computer Science Unplugged... off-line activities and games for all ages. (1998), https://classic.csunplugged.org/documents/books/english/unplugged-book-v1.pdf, last accessed 2024/06/27.
4. Blanton, M.L., Kaput, J.J.: Functional Thinking as a Route To Algebraic Thinking. In: J. Cai, E. Knuth (ed.) Early Algebrization. Springer (2011).
5. CS Unplugged (2024), https://www.csunplugged.org/en/, last accessed 2024/06/01.
6. CT&MathAble Homepage (2024), https://bebras.ehu.eus/CT&Mathable/Index.html, last accessed 2023/05/19.
7. Duncan, C., Bell, T.: A Pilot Computer Science and Programming Course for Primary School Students. Association for Computing Machinery. WiPSCE '15, pp. 39-48. (2015).
8. Experiential Education. Wikipedia (2024), https://en.wikipedia.org/wiki/Experiential_education, last accessed 2024/06/03.
9. Kieran, C.: Algebraic thinking in the early grades: What is it. The Mathematics Educator, 8(1), (2004).
10. Kriegler, S.: Just what is algebraic thinking? (2008), https://currikicdn.s3-us-west-2.amazonaws.com/resourcedocs/54d281a1d8bbd.pdf, last accessed 2023/05/27.
11. Lénárd, A.: A digitális környezet következményei és lehetőségei kisgyermekkorban. Iskolakultúra, 29(4-5), pp. 99-114, (2019).
12. Lins, R.C.: A framework for understanding what algebraic thinking is. University of Nothingen (1992), https://eprints.nottingham.ac.uk/13227/1/316414.pdf, last accessed 2023/05/27.
13. Papert, S.: An exploration in the space of Mathematics Education, International Journal of Computers for Mathematics, 1(1), pp. 95-123. (1996).
14. Pears, A., Bilbao, J., Dagienė, V., Gulbahar, Y., Margithy-Brecht, A., Parviainen, M., Pluhár, Zs., Sarmasági, P.: Integrating Computational Thinking with Mathematical Problem Solving. In: Pellet, J.-P., Parriaux, G. (eds.): ISSEP 2023 Local Proceedings, pp. 189–192. (2023). https://doi.org/10.5281/zenodo.8431989
15. Stramel, J.: Mathematics Methods for Early Childhood. FHSU Digital Press (2021).
16. Sysło, M.: Computer Science Education with a Computer in the Background In: Pellet, J.-P., Parriaux, G. (eds.): ISSEP 2023 Local Proceedings, pp. 89–101. (2023). https://doi.org/10.5281/zenodo.8431926
17. Ville Learning Analytics, (2024), University of Turku https://en.learninganalytics.fi/ville last accessed 2024/06/01.
18. Vöcking, B., Alt, H., Dietzfelbinger, M., Reischuk, R., Scheideler, C., Vollmer, H., Wagner, D.: Algorithms Unplugged. (2011). https://doi.org/10.1007/978-3-642-15328-0
19. White, S.K.: The 15 most in-demand tech jobs for 2024 – and how to hire for them. CIO, 5. January, (2024), https://www.cio.com/article/230935/hiring-the-most-in-demand-tech-jobs-for-2021.html, last accessed 2024/05/30.
20. Wing, J.M.: Computational thinking, Communications of the ACM, 49(3), pp. 33-35. (2006).
21. World Economic Forum, Future of Jobs Report 2023 (2023), https://www.weforum.org/reports/the-future-of-jobs-report-2023, last accessed 2024/05/22.

Designing a Computational Thinking Intervention for Kindergarten Students

Tobias Bahr[0000−0002−0766−9992]

Institute of Educational Science, University of Stuttgart, Stuttgart 70174, Germany
`bahr@ife.uni-stuttgart.de`

Abstract. In recent years, the emphasis on Future Skills–such as Communication, Critical Thinking, Creativity, Cooperation, and Computational Thinking (CT)–has significantly increased in K-12 education, as these skills can empower students to understand new phenomena that shape society. Various countries are striving to provide students with learning opportunities to acquire these skills starting in early childhood education. However, the state of research on the implementation of CT in educational settings by teachers, the effectiveness of piloted educational materials, and the outcomes of teacher training programs remains limited, especially in early childhood education compared to other K-12 age groups. This work-in-progress paper[1] presents an overview of related work on CT in early childhood Computer Science education, describes an intervention, and concludes with an outlook on the implications of the findings and potential future directions.

Keywords: Computational Thinking · Computer Science Education · Kindergarten · Early Childhood Education

1 Introduction

In addition to the disciplinary computer science (CS) skills that students should acquire, as recommended by several expert groups [6], interdisciplinary skills such as future skills [10] have gained increasing importance in recent years [23, 10]. These skills include problem-solving, critical thinking, creativity, cooperation, and communication (the 4 C's) [23, 10], as well as Computational Thinking (CT) [39]. Several countries, including the USA and England, have implemented CT learning opportunities in CS curricula to foster these skills in early childhood education [17, 31]. Initial research indicates that CT interventions can provide students with the opportunity to acquire these aforementioned skills [31, 11, 38]. However, whether and how these learning opportunities are implemented in kindergartens remains an under-researched area [31, 15, 18]. Ultimately, it is up to kindergarten teachers to integrate these learning opportunities [18, 15].
Numerous literature reviews have explored Computational Thinking (CT) within

---

[1] This work was funded by the TÜV SÜD Stiftung (project number: S212/10079/2024).

the context of early childhood Computer Science Education (CSE), highlighting the challenges in selecting appropriate learning opportunities and tools [34, 1, 42]. Su and Yang [34] emphasized the necessity of age-appropriate CT interventions. Bati [1] suggested that unplugged CS activities (without a robot [18]) may be more effective in developing students' CT skills compared to plugged-in activities. Zeng et al. [42] indicated that research predominantly focuses on CT concepts. This work-in-progress paper aims to contribute to the literature by presenting potential CT learning opportunities and an intervention informed by related research and experts in early childhood education (ECE). The paper concludes with a discussion of future research directions for CT in early childhood CSE (EC-CSE).

2 Theory

According to Wing [37], CT is a fundamental skill that spans several disciplines, including computer science. Various definitions of CT exist and vary depending on the context [32, 8]. Eickelmann et al. [13] describe CT as the ability to identify and model a real-world problem, break it down into several parts, and develop a solution that can be understood and applied by a person or a computer. Based on this and other definitions, CT can be described by the following facets, as outlined in a literature review by Shute et al. [32]: decomposition, abstraction (data collection and analysis, pattern recognition, modeling), algorithms (algorithm design, parallelism, efficiency, automation), debugging, iteration, and generalization. Furthermore, Dagienė et al. [9] provided criteria for categorizing CT tasks: *CT1: Abstraction* (Removing unnecessary details; Spotting key elements in problem; Choosing a representation of a system), *CT2: Algorithmic thinking* (Thinking in terms of sequences and rules; Executing an algorithm; Creating an algorithm), *CT3: Decomposition* (Breaking down tasks; Thinking about problems in terms of component parts; Making decisions about dividing into sub-tasks with integration in mind, e.g. deduction), *CT4: Evaluation* (Finding best solution; Making decisions about good use of resources; Fitness for purpose), *CT5: Generalisation* (Identifying patterns as well as similarities and connections; Solving new problems based on already-solved problems; Utilizing the general solution, e.g. induction). Similar categorizations exist such as the one from Brennan and Resnick [5] and Su and Yang [34]. However, those did not involve expert feedback from the bebras community from several countries [9].

3 Related Work

A recent literature review on CT in EC-CSE [12] identified the Bee-bot [31], ScratchJr [24], and KIBO [22] as the main educational tools used in recent studies. Zeng et al. [42] found in their literature review in the same context that the CT framework of Brennan and Resnick [5] provides a good basis for CT curricula in EC-CSE. Another finding was that research on CT concepts dominates over

CT practices [42]. The literature review by Bati [1] found that unplugged activities might yield a favourable outcome regarding children's CT skills. Su and Yang [34] found that children can develop future skills such as communication, collaboration, and problem solving from CT learning opportunities. Furthermore, the results of their literature review indicate the lack of valid assessment instruments, and they identify the challenge of selecting appropriate learning tools [34]. This work-in-progress report tries to fill that gap with the explicit search for learning opportunities in the context of CT in EC-CSE, briefly discussing the operationalization of CT interventions and suggesting one. For this, the 17 papers included in the literature review [34] with empirical experiments in ECE were analysed.

3.1 Learning opportunities

All CT interventions had a duration between two and 13 sessions [31, 18, 34] with one exception (three weeks [26]). Another result analyzing the learning opportunities was that interventions with multiple sessions were around 30 to a maximum of 90 minutes long per session [18, 34]. Six studies incorporated initial unplugged sessions followed by sessions using an educational robot (e.g., the Bee-bot) [18, 7, 29]. Eleven studies used educational robots or SratchJr directly [34]. According to Kim et al. [18], early childhood CT learning opportunities should emphasize a holistic experience with more exploration and play than instruction and focus on fostering a growth mindset as well as communication. Several educational tools and platforms have been used to foster kindergarten students' CT skills, namely the Ozobot, the Bee-bot [31], LEGO® robotics kits [4], the Cubetto [19], Bebras tasks [40, 35], the KIBO kit [2], CHERP [4], and Kodable [25], according to the literature reviews [12, 42, 1, 34]. While ScratchJr and other programs allow kindergarten students to create their first programs on their own, the literature reviews and studies in the field of EC-CSE suggest starting with unplugged activities and robots [18, 12]. Educators and researchers alike consider the Bee-bot an adequate educational robot to foster students' CT skills at an early age [31, 5]. Furthermore, it can be used in versatile situations (drawing shapes and moving objects) and can be included in other STEM subjects such as mathematics as well [31].

3.2 Operationalization of CT skills

Several CT tests have been adapted for K-4 CSE and were piloted or validated, such as the Computational Thinking Test for Beginners (*BCTt*) (25 items, 45 min, 1st and 2nd grade students, Cronbach's $\alpha = .82$) [41], the adapted Computational Thinking Test (*CTt*) (21 items, 20 min, 1st to 4th grade students, Cronbach's $\alpha = .637$) [36], the competent Computational Thinking Test (*cCTt*) (25 items, 20 min, 3rd and 4th grade students, Cronbach's $\alpha = .85$) [14], and the *TechCheck-K* (15 items, 25 min, kindergarten students, Cronbach's $\alpha = .87$) [27, 21]. Only the latter was piloted and validated with students in kindergarten [27, 21] and, according to the authors, does not require prior programming knowledge. Hence, the *TechCheck-K* could be favoured for kindergarten CSE. As outlined in the limitations by the cited CT test publications, CT tests mainly measure CT skills and partly computational practices [36]. Therefore, video analyses and reported observations during a CT intervention in kindergarten could additionally be used to assess students' CT processes [18, 31, 3, 33].

4 The intervention

Based on the related work found in the literature and after discussing the existing learning opportunities with experts from the field of ECE (duration, feasibility [29], hardware limitations [4], teacher training [15], appropriate number of students in groups [31], age group [7]), the following intervention is proposed and visualized in Figure 1. The concept consists of nine 60-minute CT sessions with an active learning time around 30 minutes. The duration was deemed adequate as other interventions in the same context chose a similar length [40, 20, 34]. First, unplugged learning opportunities are provided to the students [7], beginning with a cake recipe written using symbols [30] created collaboratively in groups. Continuing with real-world contexts, the second activity allows students to create their route to the kindergarten with programming steps (with cards) and communicate it. Unlike the structured cake recipe task, this activity emphasizes open-ended solutions (divergent task) [28]. Physical education can be integrated into the third learning opportunity. After an introduction from teachers, one student plays the role of a robot while another gives instructions, ultimately navigating an obstacle course (C1-4, CT1-4) [30]. For students to successfully learn the precise formulation of algorithms, teachers should first play the role of the robot [30]. In the next session, students solve Bebras tasks [40], engaging with CT2 and CT4. Afterwards, students solve various Bee-bot tasks with increasing difficulty, sketches, and more according to the didactical principles from the literature review by Seckel et al. [31] (C1-4, CT1-4). In the final project, students are encouraged to create their own Bee-bot tasks and connect CS with mathematics (C1-4, CT1-5) [31].

5 Discussion and future work

All of the aforementioned studies conclude that the different learning opportunities were successful in fostering students' problem-solving, future skills, and/or CT skills [2, 4, 18, 3, 40, 16, 20]. Research gaps identified by different researchers include the lack of valid assessment instruments [34], the need for further analysis on gender differences to ensure educational equality [34, 25, 21], and the necessity of focusing on both pre-service and in-service kindergarten teachers [15]. Additionally, there is a need for a greater focus on students' CT (computational thinking) practices [3, 42]. Thus, a research gap is identified in the context of an intervention using a quasi-experimental design with a pre- and post-test to assess kindergarten students' CT skills using the *TechCheck-K* [27], alongside teacher

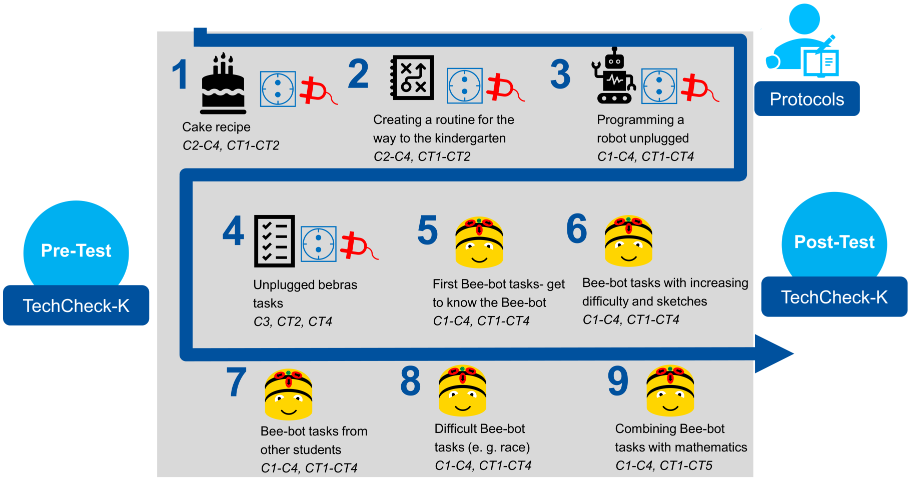


Fig. 1. Visualization of the intervention concept using existing and piloted CT learning opportunities. Including a quantitative assessment with the *TechCheck-K* [27] as a pre- and post-test as well as qualitative observation protocols from educators during the intervention based on a classification scheme [3]. The learning opportunities were categorized with the 4 C's (C1 Creativity, C2 Communication, C3 Critical Thinking, C4 Collaboration) as well as the CT categorization [9]

and researcher observations of students' CT practices [3] in a naturalistic setting [33] taught by kindergarten teachers.
Based on the findings of this paper, future work related to CT in EC-CSE could focus on CS curriculum development with piloted learning materials tested through quasi-experimental studies. Developing teacher training programs in the context of CT requires grounding these programs in empirical findings from existing studies and the practical experiences of kindergarten teachers. Previous research and the experiences of educators who have implemented CT in ECE (e.g., [31, 18, 19, 2]) provide valuable insights. These programs could greatly benefit school administrations and future kindergarten teachers. Additionally, they serve as a resource for educators involved in designing new instructional materials. As the number of CS curricula in ECE increases, comparisons of lessons learned from different implementations across countries could be of interest for the research community as well.

References

1. Bati, K.: A systematic literature review regarding computational thinking and programming in early childhood education. Education and Information Technologies **27**(2), 2059–2082 (2022). https://doi.org/10.1007/s10639-021-10700-2

2. Bers, M., González González, C., Torres, M.: Coding as a playground: Promoting positive learning experiences in childhood classrooms. Computers & Education **138** (2019). https://doi.org/10.1016/j.compedu.2019.04.013
3. Bers, M.U.: Coding as a Playground: Programming and Computational Thinking in the Early Childhood Classroom. Routledge Press (2020)
4. Bers, M.U., Flannery, L., Kazakoff, E.R., Sullivan, A.: Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. Computers & Education **72**, 145–157 (2014). https://doi.org/10.1016/j.compedu.2013.10.020
5. Brennan, K., Resnick, M.: New frameworks for studying and assessing the development of computational thinking. In: Proceedings of the 2012 annual meeting of the American educational research association, Vancouver, Canada. vol. 1, p. 25 (2012)
6. Caspersen, M.E., Diethelm, I., Gal-Ezer, J., McGettrick, A., Nardelli, E., Passey, D., Rovan, B., Webb, M.: Informatics Reference Framework for School (2022), https://www.informaticsforall.org/wp-content/uploads/2022/03/Informatics-Reference-Framework-for-School-release-February-2022.pdf
7. Critten, V., Hagon, H., Messer, D.: Can Pre-school Children Learn Programming and Coding Through Guided Play Activities? A Case Study in Computational Thinking. Early Childhood Education Journal **50**(6), 969–981 (2022). https://doi.org/10.1007/s10643-021-01236-8
8. Curzon, P., Bell, T., Waite, J., Dorling, M.: Computational Thinking, p. 513–546. Cambridge Handbooks in Psychology, Cambridge University Press (2019)
9. Dagienė, V., Sentance, S., Stupurienė, G.: Developing a two-dimensional categorization system for educational tasks in informatics. Informatica **28**(1), 23–44 (2017). https://doi.org/10.15388/Informatica.2017.119
10. Davies, A., Fidler, D., Gorbis, M.: Future work skills 2020. Institute for the Future for University of Phoenix research Institute **540** (2011), https://www.voced.edu.au/content/ngv:49812
11. Ebner, M., Schön, S., Narr, K., Grandl, M., Khoo, E., Hurajová, L.: Learning design for children and youth in makerspaces: Methodical-didactical variations of maker education activities concerning learner's interest, learning with others and task description (2021). https://doi.org/10.2991/assehr.k.211212.038
12. Edelberto Franco Silva, B.J.D., Semaan, G.S.: A literature review of computational thinking in early ages. International Journal of Early Years Education **31**(3), 753–772 (2023). https://doi.org/10.1080/09669760.2022.2107491
13. Eickelmann, B., Bos, W., Gerick, J., Goldhammer, F., Schaumburg, H., Schwippert, K., Senkbeil, M., Vahrenhold, J. (eds.): ICILS 2018 #Deutschland. Computer- und informationsbezogene Kompetenzen von Schülerinnen und Schülern im zweiten internationalen Vergleich und Kompetenzen im Bereich Computational Thinking. Waxmann, Münster and New York (2019), http://nbn-resolving.org/urn:nbn:de:0111-pedocs-181664
14. El-Hamamsy, L., Zapata-Cáceres, M., Barroso, E.M., Mondada, F., Zufferey, J.D., Bruno, B.: The competent computational thinking test: Development and validation of an unplugged computational thinking test for upper primary school. Journal of Educational Computing Research **60**(7), 1818–1866 (2022). https://doi.org/10.1177/07356331221081753
15. Espinal, A., Vieira, C., Magana, A.J.: Professional development in computational thinking: A systematic literature review. ACM Trans. Comput. Educ. **24**(2) (2024). https://doi.org/10.1145/3648477

16. Fu, C., Hao, X., Shi, D., Wang, L., Geng, F.: Effect of coding learning on the computational thinking of young Chinese children: based on the three-dimensional framework. Education and Information Technologies **28**(11), 14897–14914 (2023). https://doi.org/10.1007/s10639-023-11807-4
17. Grover, S., Pea, R.: Computational thinking in k–12: A review of the state of the field. Educational Researcher **42**(1), 38–43 (2013). https://doi.org/10.3102/0013189X12463051
18. Kim, J., Leftwich, A., Castner, D.: Beyond teaching computational thinking: Exploring kindergarten teachers' computational thinking and computer science curriculum design considerations. Education and Information Technologies (2024). https://doi.org/10.1007/s10639-023-12406-z
19. Marinus, E., Powell, Z., Thornton, R., McArthur, G., Crain, S.: Unravelling the cognition of coding in 3-to-6-year olds: The development of an assessment tool and the relation between coding ability and cognitive compiling of syntax in natural language. In: Proceedings of the 2018 ACM Conference on International Computing Education Research. p. 133–141. ICER '18, Association for Computing Machinery, New York, NY, USA (2018). https://doi.org/10.1145/3230977.3230984
20. Messer, D., Thomas, L., Holliman, A., Kucirkova, N.: Evaluating the effectiveness of an educational programming intervention on children's mathematics skills, spatial awareness and working memory. Education and Information Technologies **23**(6), 2879–2888 (2018). https://doi.org/10.1007/s10639-018-9747-x
21. Metin, Ş., Başaran, M., Seheryeli, M.Y., Relkin, E., Kalyenci, D.: Adaptation of the Computational Thinking Skills Assessment Tool (TechCheck-K) in Early Childhood. Journal of Science Education and Technology **33**(3), 365–382 (2024). https://doi.org/10.1007/s10956-023-10089-2
22. Mollie Elkin, A.S., Bers, M.U.: Programming with the kibo robotics kit in preschool classrooms. Computers in the Schools **33**(3), 169–186 (2016). https://doi.org/10.1080/07380569.2016.1216251
23. OECD: Oecd future of education and skills 2030: Oecd learning compass 2030. a series of concept notes (2019), https://www.oecd.org/education/2030-project/teaching-and-learning/learning/learning-compass-2030/OECD_Learning_Compass_2030_Concept_Note_Series.pdf
24. Papadakis, S., Kalogiannakis, M., Zaranis, N.: Developing fundamental programming concepts and computational thinking with scratchjr in preschool education: a case study. International Journal of Mobile Learning and Organisation **10**(3), 187–202 (2016)
25. Pila, S., Aladé, F., Sheehan, K.J., Lauricella, A.R., Wartella, E.A.: Learning to code via tablet applications: An evaluation of daisy the dinosaur and kodable as learning tools for young children. Computers & Education **128**, 52–62 (2019). https://doi.org/10.1016/j.compedu.2018.09.006
26. Pugnali, A., Sullivan, A., Bers, M.U.: The impact of user interface on young children's computational thinking. Journal of Information Technology Education.Innovations in Practice **16**, 171–193 (2017)
27. Relkin, E., Bers, M.: Techcheck-k: A measure of computational thinking for kindergarten children. In: 2021 IEEE Global Engineering Education Conference (EDUCON). pp. 1696–1702 (2021). https://doi.org/10.1109/EDUCON46332.2021.9453926
28. Romeike, R.: Creative students: what can we learn from them for teaching computer science? In: Proceedings of the 6th Baltic Sea Conference on Computing Education Research: Koli Calling 2006. p. 149–150. Baltic Sea

'06, Association for Computing Machinery, New York, NY, USA (2006). https://doi.org/10.1145/1315803.1315835

29. Saxena, A., Lo, C.K., Hew, K.F., Wong, G.K.W.: Designing Unplugged and Plugged Activities to Cultivate Computational Thinking: An Exploratory Study in Early Childhood Education. The Asia-Pacific Education Researcher **29**(1), 55–66 (2020). https://doi.org/10.1007/s40299-019-00478-w
30. Schmitz, D.: Medienpädagogik trotz knapper ressourcen: Vorstellung der „digiboxen“ als niederschwelliges angebot für das medienpädagogische arbeiten in kindergarten und schule. Medienimpulse **61**(3), 24–Seiten (2023)
31. Seckel, M.J., Salinas, C., Font, V., Sala-Sebastià, G.: Guidelines to develop computational thinking using the Bee-bot robot from the literature. Education and Information Technologies **28**(12), 16127–16151 (2023). https://doi.org/10.1007/s10639-023-11843-0
32. Shute, V.J., Sun, C., Asbell-Clarke, J.: Demystifying computational thinking. Educational Research Review **22**, 142–158 (2017). https://doi.org/10.1016/j.edurev.2017.09.003
33. Strawhacker, A., Lee, M., Bers, M.U.: Teaching tools, teachers' rules: exploring the impact of teaching styles on young children's programming knowledge in ScratchJr. International Journal of Technology and Design Education **28**(2), 347–376 (2018). https://doi.org/10.1007/s10798-017-9400-9
34. Su, J., Yang, W.: A systematic review of integrating computational thinking in early childhood education. Computers and Education Open **4**, 100122 (2023). https://doi.org/10.1016/j.caeo.2023.100122
35. Sung, J.: Assessing young Korean children's computational thinking: A validation study of two measurements. Education and Information Technologies **27**(9), 12969–12997 (2022). https://doi.org/10.1007/s10639-022-11137-x
36. Tsarava, K., Moeller, K., Román-González, M., Golle, J., Leifheit, L., Butz, M.V., Ninaus, M.: A cognitive definition of computational thinking in primary education. Computers & Education **179**, 104425 (2022). https://doi.org/10.1016/j.compedu.2021.104425
37. Wing, J.M.: Computational thinking. Commun. ACM **49**(3), 33–35 (2006). https://doi.org/10.1145/1118178.1118215
38. Xiang, S., Yang, W., Yeter, I.H.: Making a makerspace for children: A mixed-methods study in chinese kindergartens. International Journal of Child-Computer Interaction **36**, 100583 (2023). https://doi.org/10.1016/j.ijcci.2023.100583
39. Yadav, A., Good, J., Voogt, J., Fisser, P.: Computational Thinking as an Emerging Competence Domain, pp. 1051–1067. Springer International Publishing, Cham (2017). https://doi.org/10.1007/978-3-319-41713-4_49
40. Zapata-Cáceres, M., Marcelino, P., El-Hamamsy, L., Martín-Barroso, E.: A Bebras Computational Thinking (ABC-Thinking) program for primary school: Evaluation using the competent computational thinking test. Education and Information Technologies (2024). https://doi.org/10.1007/s10639-023-12441-w
41. Zapata-Cáceres, M., Martín-Barroso, E., Román-González, M.: Computational thinking test for beginners: Design and content validation. In: 2020 IEEE Global Engineering Education Conference (EDUCON). pp. 1905–1914 (2020). https://doi.org/10.1109/EDUCON45650.2020.9125368
42. Zeng, Y., Yang, W., Bautista, A.: Computational thinking in early childhood education: Reviewing the literature and redeveloping the three-dimensional framework. Educational Research Review **39**, 100520 (2023). https://doi.org/10.1016/j.edurev.2023.100520

AI and Robotics in Early Education: Engaging Young Learners with Emerging Technologies

Experience AI: Introducing AI/ML to Grade 6–8 students in the UK

Jane Waite[1][0000−0002−0270−2124], Ben Garside[1][0009−0003−0625−7341], Robert Whyte[2][0000−0002−1214−1957], Diana Kirby[2][0000−0003−3630−2548], and Sue Sentance[2][0000−0002−0259−7408]

[1] Raspberry Pi Foundation, Cambridge, UK
`{jane.waite,ben.garside,bobby.whyte}@raspberrypi.org`
[2] Raspberry Pi Computing Education Research Centre,
University of Cambridge, Cambridge, UK
`{rw724,dgk21,ss2600}@cam.ac.uk`

Abstract. The call for school students to learn about artificial intelligence and machine learning is increasing, yet what should be taught and how is still to be agreed upon. Despite this, resources are starting to be developed. Such material can be used to explore pedagogical content knowledge decisions and evaluate the impact of teaching activities. In this experience report, we detail the development and implementation of Experience AI, a free curriculum unit consisting of six one-hour lessons designed for classroom use by teachers of students aged 11 to 14 years old (Grade 6–8) in the United Kingdom. The lessons were designed with an underlying set of design principles developed in consultation with industry experts. The design principles we focus on in this report are (i) avoiding anthropomorphisation of language and images used in the resources, (ii) incorporating careers materials and activities, and (iii) increased teacher support for lesson delivery. The resources include teacher guides, classroom presentations, explanations of key terms, student activities, and assessment ideas. From an independent evaluation of the implementation of the lessons, initial survey results are reported (student conceptions of AI, student and teacher AI careers awareness, teacher self-efficacy when teaching about AI). Evidence from the evaluation has provided early yet encouraging evidence that teachers who used the curriculum unit may have improved their AI careers awareness and efficacy when teaching about AI. We suggest the design principles, lesson materials and evaluation instruments may be useful to other researchers working in this field.

Keywords: Artificial intelligence · machine learning · curriculum

1 Introduction

Even though artificial intelligence (AI) systems are becoming ubiquitous across society [9], AI technology is not widely understood by those affected [14]. Creating an AI-ready workforce is a significant focus for many governments [4]. In the United Kingdom (UK), AI policy has included the development of recommendations [18], policy options [21], and roadmaps for research and change [25]. AI education has also been called to be incorporated into the school and teacher preparation curricula [21, 25]. In the computing education research community, there are a growing number of initiatives to do this [24, 22]. However, there is limited empirical research to understand what and how AI should be taught, and what the impact of early initiatives might be [23, 17]. This presents a challenge for K–12 computing education researchers, education resource developers, and teachers alike to consider how AI and machine learning (ML) can be introduced.

In this experience report, we describe the design of a set of 6 lessons introducing AI/ML to students aged 11 to 14 (Grades 6 to 8) in the UK. The resources were developed with support from Google DeepMind through a volunteer industry expert working group. We also detail results from an independent evaluation of the school implementation of the resources.

2 Background and related work

What might be taught about AI has been defined in various but conflicting ways. For example, the AI4K12 working group suggested 'Five Big Ideas' for K–12 AI: perception, representation and reasoning, learning, natural interaction, and societal impact [24]. From an analysis of 30 K–12 instructional units on ML, Marques et al. [8] identified 12 ML topics (e.g. neural networks), 13 ML applications (e.g. sentiment analysis), and 7 ML processes (e.g. model evaluation). A computational thinking (CT) view of learning about AI has been suggested [23], whereby the difference between rule-driven and data-driven system development paradigms is emphasised, and a new CT2.0 is defined. Differences in the problem-solving workflow of CT2.0 (data-driven) to CT1.0 (rule-driven) are compared, including describing the job and collecting the data rather than formalising the problem; filtering, cleaning and labelling the data rather than designing an algorithmic solution; training a model rather than implementing the algorithm; and evaluating and using the model rather than compiling and executing the program [23]. Clearly, a consensus about what should be taught has yet to be reached. Olari and Romeike [12] argued that most AI literacy frameworks fail to capture data science (or "data literacy") concepts and skills, and Druga et al. [3] noted that a common language for AI and ML teaching resources had not been agreed upon.

As well as considering what should be taught, an open question is how AI and ML should be taught and which pedagogy should be used. A set of 15 design considerations have also been defined [7], including contextualising data, opportunities to program, and leveraging learners' interests. From a synthesis of AI teaching studies in K–12 and a supposition of what might work for teaching this age group and topic, a taxonomy of pedagogies for AI has been suggested, including active learning, personalised learning, participatory, problem-based, interactive, project, inquiry, and design-oriented learning [20]. Finally, a simple AI and ML learning framework, called SEAME has been proposed for use in reviewing AI teaching resources [26] and research activities [17]. The framework comprises four levels: Social and Ethical, Application, Model and Engine. In SEAME, the levels do not dictate the order in which learning must occur, and some activities will span more than one level. The levels provide an intuitive way for educators, resource developers, and researchers to frame the main aims of a learning experience. As students make progress, it is expected students can move between levels, and, at times, the boundaries between levels will blur.

3 Design principles

For the lesson design, a set of design principles were devised to guide the development of AI resources. These principles extended existing general resource design principles for computing resource development [16]. In this paper, we focus on three of the AI design principles. These principles we report on here were selected as we can comment on their enactment through evaluation data. The three design principles we focus on are:

- avoiding anthropomorphisation
- promoting awareness of AI/AI-related careers
- increased teacher support

Other design principles, but not reported on in detail here, include (i) using the SEAME framework [26] to help develop the learning objectives and learning progression [16], (ii) developing a set of working explanations for learning objective concepts and sub-concepts for the research team and educators [16], and (iii) using semantic profiling to align everyday contexts to abstract technical language [10] for explanations and lesson design.

Avoiding anthropomorphisation: A key consideration was the need to avoid anthropomorphisation in student- and teacher-facing materials. Anthropmorphisation is “the action or fact of attributing human characteristics, form, or personality to something non-human (in later use esp. an animal)”.[3]

The rationale for this choice was that attributing human characteristics to computers has led to programming misconceptions [13], and more specifically for AI, may lead to system developers (including novice programmers) developing incorrect mental models of how AI works, as the technology is humanised, black-boxed, and oversimplified [23]. Additionally, when using technology, young children have been found to view robots as peers rather than devices, seeing them as less smart ‘people’ or overestimating technology capabilities [2], or developing relationships with the devices [23], leading to high risks of either unintended influence, purposeful manipulation, or policing [28]. Compounding the issue of delegating the responsibility of the human system developer and human user to an imagined responsible AI agent [19], anthropomorphised AI agents have been predominately portrayed as white in colour and as such exacerbating racism in technology and society at large [1]. However, by engaging students to learn

---

[3] $https://www.oed.com/dictionary/anthropomorphization_n$

about AI and making the ‘black box’ transparent, students become more sceptical about the technology and recognise responsibility for AI design as associated with the humans who are ‘in the loop’ rather than the AI technology [2, 23]. For Experience AI, in practice, this principle was enacted by there being no illustrations that depicted devices with human-like faces and the replacement of vocabulary that was associated with the behaviour of people (see, look, recognise, create, make) with system-type words (detect, input, pattern match, generate, produce).

In line with suggestions by Druga et al. [3] to create a shared language among curriculum developers, a set of working explanations for learning objective concepts/sub-concepts was developed in partnership with an industry working group. For example, a working definition of AI literacy was defined as *“AI literacy is a set of competencies that enables people to use AI applications in everyday life creatively and ethically, to identify and evaluate AI technologies critically, and to have a basic knowledge and understanding of the key concepts and processes associated with AI applications, models, and engines.”* [16].

Promoting awareness of AI/AI-related careers: A second design choice was to provide career examples in each lesson that shared social and ethical issues through relatable real-world examples of applications of ML models. The aims were to 1) engage students and 2) help them understand the relevance and impact AI has in their lives. The rationale for this choice was the need for students to understand the career and societal implications of AI developments. Prior work in computing education has highlighted the challenge in promoting the aspirations of young people, particularly female students, in pursuing careers in the field [5]. Studies have indicated multiple factors that impact young people’s career aspirations in computing, including a lack of exposure to CS, the need for role models, and the influence of parents and self-efficacy beliefs. As such, researchers have underscored the importance of promoting awareness of AI career opportunities and the broader impact of AI across disciplines [29]. In our lesson materials, the careers principle was enacted by demonstrating the breadth of careers in AI/AI-related fields through real-world examples. Video interviews with researchers and scientists working at Google DeepMind were featured throughout the lessons to enrich classroom discussions on career goals.

Increased teacher support: Though teaching and learning of AI and ML in schools is an increasingly important topic being suggested for classroom teachers to consider [4], it is unlikely that teachers will have prior experience of AI/ML or appropriate pedagogical knowledge when working with school-aged learners [23, 30]. Therefore, in Experience AI, to increase teacher confidence and self-efficacy, teacher support was provided through the following means:

- *Student-facing concept videos* embedded in the lesson slides, including industry experts explaining key AI/ML concepts.
- *A free asynchronous online teacher professional development course* on introducing AI/ML, how to deliver the lessons, and understanding concepts.
- *Teacher support videos* to introduce lesson activities, including screen-casts demonstrating the steps students will need to follow for practical tasks.

- *Webinars* for teachers including in-depth discussions on AI/ML concepts with industry experts and guidance on how to deliver lesson activities.
- *Lesson plans* which include detailed guidance on how to deliver the lesson activities, including additional information about the concepts and suggestions on formative assessment questions to ask students.

Prior surveys of AI/ML teaching and learning resources indicate a lack of teacher documentation in most currently-available resources [26]. Our intention was to provide support materials to anticipate teacher needs and provide guidance on difficulties that could be faced in the classroom.

4 Lesson resources

The free unit of work consists of six lessons available for educators to download [16]. An overview of the lessons and links to the full resource set is provided in Appendix A. The lessons are intended to cover a six-week period. Each lesson is designed to be taught in a one-hour classroom-based lesson. The unit of work is aimed at school educators in the UK who teach students aged 11 to 14 years old, although they are available for use by any educator in other countries. There are no specific hardware requirements for the lessons, except for equipment to access the internet. A web browser is needed to download and view lesson material; no software needs to be installed locally. Machine Learning For Kids[4] is used as a web app and is accessible without the need for an account.

The ambition for the six lessons is to provide students with a foundational knowledge of AI concepts, contexts, and the careers involved in developing AI applications. Building on the four strands of SEAME [26], key concepts covered include: (i) rule-based vs data-driven approaches to programming; (ii) applications of AI; (iii) ML models; (iv) bias and ethics; (v) decision trees; (vi) the AI data life cycle; and (vii) careers in AI. A description of concepts covered in the lessons is provided in Appendix A.

For each lesson, there are lesson plans, teaching slides, student activity worksheets, teacher support videos, projects for students to select from depending on their personal interests, and student assessment activities (formative and summative). In addition, there are three overarching documents available to teachers: a unit overview, learning graphs (to demonstrate progression), and a set of explanations of key terms (e.g. concepts).

The learning objectives were reviewed to draw out candidate sub-concepts (key terms); for each sub-concept, an explanation was developed [16]. Example sub-concepts incorporated in the lessons include (this is not exhaustive) prediction, supervised learning, unsupervised learning, reinforcement learning, ML classification, ML class, ML label, ML decision trees node, data types, ML training, ML training data, ML test data, data bias, societal bias, ML explainability, ML accuracy, ML confidence, data cleaning, ML model card, computer vision, and generative AI. The degree to which knowledge is built for each of these

---

[4] https://machinelearningforkids.co.uk/

sub-concepts varies; some are introduced in passing to provide context for the exercise. Some sub-concepts demonstrate the breadth of ML to avoid introducing alternate conceptions, such as thinking that supervised learning is the only method of solving problems using ML. Other concepts are returned to multiple times, such as ML training.

5 Method

An independent evaluation was conducted for both student and teacher participants. The evaluation consisted of an online pre/post survey distributed to students and teachers via Qualtrics. The evaluation study was approved by the University of Cambridge Department of Computer Science and Technology Ethics Committee (#2023) and informed consent was obtained from all participants.

Instrument design: The student survey gathered quantitative data via five-point Likert-type questions relating to multiple constructs, including students' interest in AI and their awareness of AI/AI-related careers, and was adapted from existing validated instruments [29]. For interest in AI, example statements included *'I am interested in learning about AI'* and *'I want to learn more about AI outside of school'*. For AI careers awareness, example statements included *'I know about jobs that use AI'* and *'I am interested in jobs that use AI'*. Using data from 474 students, both scales obtained a Cronbach's alpha value of $\alpha = .939$ and $\alpha = .837$ respectively, indicating excellent internal consistency. Students were also asked to define AI via a free-text question (*'In the box below, write down what you think AI is'*).

The teacher survey focused on teachers' AI careers awareness and self-efficacy when teaching about AI via two 5-point Likert-type scales [27]. For AI careers awareness, example statements included *'I know about AI careers'* and *'I know where to find resources for teaching students about AI careers'*. For the *Personal AI Teaching Efficacy and Beliefs* scale, example statements included *'I am confident that I can explain to students how AI works'* and *'I understand AI concepts well enough to be effective in teaching about AI'*. Using data from 41 teachers, both scales obtained a Cronbach's alpha value of $\alpha = .921$ and $\alpha = .918$ respectively, indicating excellent internal consistency.

Data collection: Pre-lesson delivery data from 474 students and 41 teachers were collected through the survey. Correspondingly, post-lesson data was collected from 112 students and 6 teachers. Due to the disparities in sample sizes, any conclusions drawn should be treated with caution. Pre/post responses were matched where possible and appropriate statistical tests were employed. Further details are noted in the *Limitations* (Section 7.1). For qualitative data—such as student free-text questions—we decided to analyse all data currently collected as we intend to use a range of student responses to inform future work.

Data analysis: Due to the imbalance in sample sizes of the current data collected, we employed appropriate statistical techniques for analysing data. Student data that were normally distributed (assessed by Shapiro-Wilk's test, $p < .05$) were analysed using paired-samples t-tests, while non-normally distributed data were accordingly analysed using a series of Wilcoxon signed-rank tests. Similar tests were undertaken for teacher survey responses.

Principal component analysis (PCA) was used to assess inherent latent variables within the survey Likert-type items. The suitability of PCA was assessed prior to analysis (i.e. the assumptions of independent sampling, normality, linear relationships between pairs of variables, and the variables being correlated at a moderate level). For both surveys, PCA revealed components that were consistent with the underlying sub-scales, namely that there were strong loadings.

For the qualitative data of student definitions of AI (from the open-text question), responses were analysed by two researchers until agreement was reached. Using both pre- and post-data, we analysed responses based on whether their response ascribed anthropomorphic features when describing AI. We aimed to find evidence that students would attribute fewer human characteristics (e.g. emotions, consciousness, morality) after the lessons. Examples of anthropomorphic responses are shown in Table 1.

6 Results

Teacher survey: The teacher survey measured teachers' AI teaching self-efficacy and their AI careers awareness. Teachers' post-test *Personal AI Teaching Efficacy and Beliefs* scores were .70 higher than pre-test (95% CI, .22 to 1.18), a non-significant median increase, $z(3) = 1.46$, $p = .14$. Likewise, teachers' post-test *AI careers awareness* scores were .34 higher (95% CI, -.60 to 1.29) than pre-test scores, a non-significant mean increase, $t(3) = 1.15$, $p = .167$. For teachers' self-efficacy scores, the greatest mean differences were observed for multiple items relating to teacher confidence when teaching about AI such as *'I know the steps necessary to teach about AI effectively'* (+.86) and *'I am confident that I can answer students' questions about AI'* (+.73). Likewise, the greatest mean differences for AI careers awareness were observed across multiple items including *'I know about AI careers'* (+.80) and *'I know where to go to learn more about AI careers'* (+.80). These results suggest that some teachers who took part in this programme may have felt more aware of AI careers, and some may have felt more confident in their AI teaching skills after completing these programmes.

Student survey: The student survey measured students' attitudes and awareness of AI/AI-related careers. Only a small amount of paired data was available. Post-test *interest in AI* scores were .09 higher (95% CI, -.16 to .34) than pre-test scores, a non-significant increase ($t(30) = 1.659$, $p = .24$). However, post-test *AI careers awareness* scores were .73 (95% CI, .42 to 1.03) higher than pre-test, a statistically significant increase ($t(30) = 4.919$, $p = .05$). In particular, group differences in certain items (e.g. *'I know someone like me who works in an AI-related field'* (+0.55) and *'I plan to study AI after secondary school'* (+0.40)) suggest that student awareness and interest in AI careers may have been impacted by their participation in the lessons. Additional items that had large mean differences centred on student awareness of AI-related jobs (+0.42) and discussion of jobs with families and friends (+0.46).

Table 1. Anthropomorphism in student responses to "write what you think AI is"

AI description	Examples	Pre #	%	Post #	%
Anthropomorphic	*"AI is a robot which is starting to become more human like with feelings" / "A computer that has a mind of its own and can think its own thoughts."*	78	16.46%	12	10.71%
Non-anthropomorphic	*"A simulation of intelligence made by humans [...] It has no actual intelligence" / "The development of a concept or tool created to mimic the intelligence of humans"*	396	83.54%	100	89.29%

Student conceptions of AI: Students were asked to define AI before and after taking part in the lessons. We found evidence that most students demonstrated non-anthropomorphic descriptions (e.g. *"AI simulates human behaviour in machines which helps [in] tasks such as problem solving"*) in both pre- and post-responses (83.54% and 89.29%, respectively). Students gave proportionally fewer anthropomorphic answers (e.g. *"[AI is] someone who helps [you] online"*) in post-test data (16.46% vs 10.71%) (see Table 1).

7 Discussion

Previous work has detailed the extent of resources to support teaching/learning about AI/ML [26]. However, most resources were found to not include specific learning objectives, recommended age groups, or assessment materials. Likewise, a lack of common vocabulary underpinning learning material was noted [3]. Our curriculum, Experience AI, represents a significant attempt to provide educators (and researchers) with a research-informed set of teaching materials, including teacher guides, classroom presentations, explanations of key terms, student activities and assessment ideas. Evidence from the independent evaluation provided initial evidence that this approach could support teachers' AI career awareness and efficacy when teaching about AI. We discuss these results in relation to our original design principles and relate this to prior literature.

Avoiding anthropomorphisation: Design decisions were taken to avoid anthropomorphisation in language use, however, students were not explicitly taught about the topic. Results from the independent evaluation found proportionally fewer uses of anthropomorphic language, though this was not significant given the unbalanced sample. This result highlights a challenge in understanding student awareness from written responses and also that, without explicit teaching, the change in language and imagery and motivation for this was not transparent to students. Previous work in computing education has suggested that the use of natural language (e.g. metaphors) may lead to naive preconceptions [15]. It is possible that the use of anthropomorphic language (e.g. analogies to the human brain) may serve as a scaffolding measure to aid understanding. Follow-up work could focus on qualitative analysis of student perspectives of typical AI characteristics. This could include scenario-based tasks to discover whether more robust mental models [23] or fewer problematic associations of gender, race, over-reliance or an increased view of human responsibility for AI design [2, 1, 19, 28] are developed through avoiding—or discussing the limitations of—anthropomorphisation. While differences in student use of anthropomorphic language could not be identified, we were instead provided with important insights into students' emerging understanding of AI technologies.

Promoting AI/AI-related career awareness: The curriculum unit design featured a strong emphasis on the social and ethical dimensions of AI. As in the case of the DAILy curriculum [6], we sought to raise student awareness of careers and the extent to which AI features in their everyday lives. For instance, the role AI could play in students' careers both directly—working within the field—and indirectly—impacted by AI—was explored through real-world examples. Findings from the survey data collected were limited though suggested that both students and teachers improved their awareness of AI and AI-related careers. We were encouraged to observe mean differences for student responses of appropriate Likert items *'I know about jobs that use AI'* and *'I plan to study AI after secondary school'*, as these suggest that our AI career activities both illustrated a broad range of careers (research scientists, robotics engineers, ethics researchers employed at Google DeepMind) and promoted interest among participants. Students also compared different career pathways against multiple dimensions of AI (e.g. social and ethical, creating applications and tools, training models) to demonstrate the implications of AI in a broad range of fields. Embedding career pathways in lesson materials was also intended to support teachers. As gatekeepers to facilitate student awareness of AI careers [27], it was encouraging to see large mean differences in teachers' awareness of careers and career resources. However, further work is required to better understand which resources were impactful and how teachers' and students' perceptions of AI/AI-related careers were influenced.

Teacher support: Finally, teacher support was a focus for Experience AI. Following prior work that suggests a lack of suitable documentation (e.g. learning outcomes, differentiation, and assessment activities) in most currently-available resources [26], we focused on supporting teachers through teaching materials, online courses, webinars, and student-facing materials. Survey data from the independent evaluation found early evidence that this approach could support their efficacy when teaching about AI. In particular, teachers felt most confident knowing the steps to teach about AI and receiving and answering student questions. In future work, we could seek to better understand what specific measures could support teachers implementing the curriculum in schools.

7.1 Limitations

Our findings are limited to the extent that the independent evaluation was conducted separately to the lesson design and development. As such, only the impact of some of our design principles could be investigated in light of limited data collected (e.g. teacher self-efficacy). Nonetheless, findings from the independent evaluation were encouraging and provided some early indication of the impact of teachers' and students' participation in Experience AI. Survey data were collected anonymously meaning that no additional demographic data were collected beyond age, year group (if applicable), gender and ethnicity. This meant that follow-up work could not be organised, such as focusing on more in-depth qualitative analysis of the student and teacher experience when taking part in Experience AI. This would have provided extra insights into student reasoning around AI where survey data is limited. For example, one issue with the open-text question was that students might have interpreted it to mean what future capabilities *could* be achieved using AI technologies as opposed to its current functionalities and limitations. In-depth follow-up work with students is required to differentiate between these two perspectives. Finally, non-significant increases in student and teacher attitudes merely suggest effects and necessitate further data collection and analysis to rule out chance effects and provide confirmatory evidence.

8 Conclusion

This paper has described some of the key design principles of six lessons that can be used to introduce AI/ML to Grade 6–8 students in the UK. The design principles foregrounded in our lesson unit include avoiding anthropomorphisation, embedding careers and increased teacher support. Evidence from the independent evaluation indicates that student and teacher AI careers awareness may have been positively impacted. Evidence has also provided early yet encouraging evidence that teachers who taught the curriculum unit may have had higher self-efficacy when teaching about AI. By providing the full resource set (see Appendix A), we propose that the materials may be useful to educators new to AI/ML as well as other researchers.

Further work is needed to investigate whether the content covered and design decisions, such as teaching about the difference between rule-based and data-driven systems, are the most effective way to develop useful mental models. Similarly, whether starting with decision trees, rather than neural networks or other machine learning engines, is the most effective way to establish an effective progression of knowledge building. Also, whether the instructional approaches used sufficiently scaffold learning for all, or if they kept the nuances of ML too hidden to help students overcome either too much or too little trust in the predictions of ML models requires further research. We look forward to exploring these issues and hope that our resources will be useful to others to investigate.

Acknowledgments. We are grateful to Google DeepMind for funding the development and evaluation of this programme.

References

1. Cave, S., Dihal, K.: The Whiteness of AI. Philosophy & Technology **33**(4), 685–703 (Dec 2020). https://doi.org/10.1007/s13347-020-00415-6, https://link.springer.com/10.1007/s13347-020-00415-6
2. Druga, S., Ko, A.J.: How do children's perceptions of machine intelligence change when training and coding smart programs? In: Interaction Design and Children. pp. 49–61. ACM, Athens Greece (Jun 2021). https://doi.org/10.1145/3459990.3460712
3. Druga, S., Otero, N., Ko, A.J.: The landscape of teaching resources for AI education. In: Proceedings of the 27th ACM Conference on on Innovation and Technology in Computer Science Education Vol. 1. pp. 96–102. ITiCSE '22, Association for Computing Machinery, New York, NY, USA (Jul 2022)
4. Galindo, L., Perset, K., Sheeka, F.: An overview of national AI strategies and policies. OECD Going Digital Toolkit Notes **14** (2021). https://doi.org/https://doi.org/https://doi.org/10.1787/c05140d9-en
5. Hur, J.W., Andrzejewski, C.E., Marghitu, D.: Girls and computer science: experiences, perceptions, and career aspirations. Computer Science Education **3408**(October), 100–120 (2017)
6. Lee, I., Ali, S., Zhang, H., DiPaola, D., Breazeal, C.: Developing middle school students' AI literacy. In: Proceedings of the 52nd ACM Technical Symposium on Computer Science Education. pp. 191–197. SIGCSE '21, Association for Computing Machinery, New York, NY, USA (Mar 2021)
7. Long, D., Magerko, B.: What is AI Literacy? Competencies and Design Considerations. In: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. pp. 1–16. ACM, Honolulu HI USA (Apr 2020). https://doi.org/10.1145/3313831.3376727
8. Marques, L.S., Gresse Von Wangenheim, C., Hauck, J.C.R.: Teaching machine learning in school: A systematic mapping of the state of the art. Informatics in Education pp. 283–321 (Jun 2020)
9. Maslej, N., Fattorini, L., Brynjolfsson, E., Etchemendy, J., Ligett, K., Lyons, T., Manyika, J., Ngo, H., Niebles, J.C., Parli, V., Shoham, Y., Wald, R., Clark, J., Perrault, R.: The AI Index 2023 Annual Report (2023)
10. Maton, K.: Making semantic waves: A key to cumulative knowledge-building. Linguistics and Education **24**(1), 8–22 (Apr 2013)
11. Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., Spitzer, E., Raji, I.D., Gebru, T.: Model Cards for Model Reporting. In: Proceedings of the Conference on Fairness, Accountability, and Transparency. pp. 220–229 (Jan 2019). https://doi.org/10.1145/3287560.3287596, arXiv:1810.03993 [cs]
12. Olari, V., Romeike, R.: Addressing AI and Data Literacy in Teacher Education: A Review of Existing Educational Frameworks. In: The 16th Workshop in Primary and Secondary Computing Education. WiPSCE '21, ACM, New York, NY, USA (2021). https://doi.org/10.1145/3481312.3481351
13. Pea, R.D.: Language-independent conceptual "bugs? in novice programming. Journal of educational computing research **2**(1), 25–36 (1986)
14. Pedro, F., Subosa, M., Rivas, A., Valverde, P.: Artificial intelligence in education: Challenges and opportunities for sustainable development. Tech. rep., UNESCO (2019)
15. Qian, Y., Lehman, J.: Students' Misconceptions and Other Difficulties in Introductory Programming: A Literature Review. ACM Trans. Comput. Educ. **18**(1), 1:1–1:24 (2017)

16. Raspberry Pi Foundation: Experience AI design principles study (2024), rpf.io/ai-design-principles-study
17. Rizvi, S., Waite, J., Sentance, S.: Artificial Intelligence teaching and learning in K-12 from 2019 to 2022: A systematic literature review. Computers and Education: Artificial Intelligence **4**, 100145 (2023). https://doi.org/https://doi.org/10.1016/j.caeai.2023.100145
18. Royal Society: Machine learning: the power and promise of computers that learn by example. Royal Society (2017), https://royalsociety.org/-/media/policy/projects/machine-learning/publications/machine-learning-report.pdf
19. Salles, A., Evers, K., Farisco, M.: Anthropomorphism in AI. AJOB Neuroscience **11**(2), 88–95 (Apr 2020). https://doi.org/10.1080/21507740.2020.1740350, https://www.tandfonline.com/doi/full/10.1080/21507740.2020.1740350
20. Sanusi, I.T., Oyelere, S.S., Agbo, F.J., Suhonen, J.: Survey of Resources for Introducing Machine Learning in K-12 Context. In: 2021 IEEE Frontiers in Education Conference (FIE). pp. 1–9. IEEE, Lincoln, NE, USA (Oct 2021). https://doi.org/10.1109/FIE49875.2021.9637393, https://ieeexplore.ieee.org/document/9637393/
21. Select Committee on AI: AI in the UK: ready, willing and able? House of Lords (2018), https://publications.parliament.uk/pa/ld201719/ldselect/ldai/100/100.pdf
22. Tedre, M., Denning, P., Toivonen, T.: CT 2.0. In: 21st Koli Calling International Conference on Computing Education Research. Koli Calling '21, Association for Computing Machinery, New York, NY, USA (2021). https://doi.org/10.1145/3488042.3488053
23. Tedre, M., Toivonen, T., Kahila, J., Vartiainen, H., Valtonen, T., Jormanainen, I., Pears, A.: Teaching Machine Learning in K-12 Computing Education: Potential and Pitfalls (2021). https://doi.org/10.48550/ARXIV.2106.11034, https://arxiv.org/abs/2106.11034
24. Touretzky, D., Gardner-McCune, C., Martin, F., Seehorn, D.: Envisioning AI for K-12: What Should Every Child Know about AI? Proceedings of the AAAI Conference on Artificial Intelligence **33**(01), 9795–9799 (Jul 2019). https://doi.org/10.1609/aaai.v33i01.33019795
25. UK AI Council: AI Roadmap. Nuffield Foundation (2019), https://assets.publishing.service.gov.uk/media/5ff3bc6e8fa8f53b76ccee23/AI_Council_AI_Roadmap.pdf
26. Waite, J., Tshukudu, E., Cucuiat, V., Whyte, R., Sentance, S.: Towards a Framework for Learning Content Analysis in K-12 AI/ML Education. In: 2023 IEEE Frontiers in Education Conference (FIE). pp. 1–5. IEEE Computer Society, Los Alamitos, CA, USA (Oct 2023). https://doi.org/10.1109/FIE58773.2023.10343368
27. Whyte, R., Kirby, D., Sentance, S.: Measuring Teacher Self-Efficacy and Careers Awareness in K-12 AI Education. In: Proceedings of the 2024 ITiCSE, July 8–10, 2024, Milan, Italy. ITiCSE '24, ACM, New York, NY, USA (Oct 2024). https://doi.org/10.1145/3649405.3659489
28. Williams, R., Ali, S., Devasia, N., DiPaola, D., Hong, J., Kaputsos, S.P., Jordan, B., Breazeal, C.: AI + ethics curricula for middle school youth: Lessons learned from three Project-Based curricula. Int J Artif Intell Educ pp. 1–59 (Aug 2022)
29. Zhang, H., Lee, I., Ali, S., DiPaola, D., Cheng, Y., Breazeal, C.: Integrating ethics and career futures with technical learning to promote AI literacy for middle school students: An exploratory study. Int J Artif Intell Educ pp. 1–35 (May 2022)
30. Zhou, X., Brummelen, J.V., Lin, P.: Designing AI Learning Experiences for K-12: Emerging Works, Future Opportunities and a Design Framework (2020)

A Appendix 1

All resources are free to use by anyone now and in perpetuity under a Creative Commons license (CC BY-NC-ND 4.0). The full resource set is available via the study website [16].

A.1 Lesson 1 - What is AI?

Students explore the current state of artificial intelligence (AI), and how it is used in the world. They explore the difference between rule-based systems and data-driven models (Figure 1) and consider the benefits and drawbacks of AI systems. The learning objectives covered are: (i) Describe the difference between 'data-driven' and 'rule-based' approaches to application development; (ii) Name examples of AI applications; (iii) Outline some benefits and issues of using AI applications.

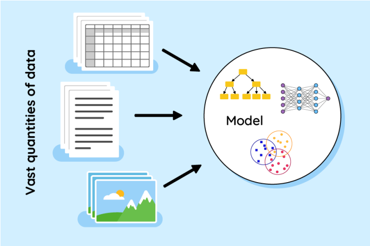


Fig. 1. Figure used to illustrate data-driven models.

A.2 Lesson 2 - How computers learn from data

The activities in this lesson help students think critically about which parts of a system use AI components and the role of ML models. Through a video, students hear from experts about the different types of ML and example problems solved. Students are introduced to a specific example of ML: classification. The learning objectives are: (iv) Define machine learning's relationship to artificial intelligence; (v) Name the three common approaches to machine learning; (vi) Describe how classification can be solved using supervised learning.

A.3 Lesson 3 - Bias in, bias out

Students create an ML model using Machine Learning for Kids[5]. The model classifies images of apples and tomatoes (Figure 2), but students discover that

---

[5] https://machinelearningforkids.co.uk/

their model is flawed due to the limited training data set. Next, students explore training data bias and biased predictions. Learning objectives include: (vii) Describe the impact of data on the accuracy of a machine learning (ML) model; (viii) Explain the need for both training and test data; (ix) Explain how bias can influence the predictions generated by an ML model.

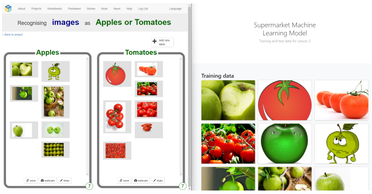


Fig. 2. Slides showing adding training data.

A.4 Lesson 4 - Decision trees

Students take their first in-depth look at a type of engine: decision trees. The activities build on students' learning from Lessons 1–3 about classification, training and test data, and the data-driven nature of models. The aim is for students to gain an understanding of the processes used to create ML models. Learning objectives: (x) Describe how decision trees are used to build a classification ML model; (xi) Describe how training data changes an ML model; (xii) Explain why ML is used to create decision trees.

A.5 Lesson 5 - Solving problems with ML models

Students are introduced to the AI project lifecycle. They follow the stages to create an ML model to solve a problem of their choice from example projects. They train the model and test it to determine its accuracy. (xiii) Describe the stages of the AI project lifecycle; (xiv) Use a machine learning tool to import data and train a model; (xv) Test and examine the accuracy of an ML model.

A.6 Lesson 6 - Model cards and careers

In this lesson, students complete the final stages of the AI project lifecycle: evaluating and explaining a model. To help them explain their model, students are introduced to model cards [11]. (xvi) Evaluate an ML model; (xvii) Produce a model card to explain an ML model; (xviii) Recognise the range of opportunities that exist in AI-related careers. The main instructional approaches used are active learning through discussion, project-based learning, and to provide student choice. Students conclude by exploring careers both in AI and fields in which it is used.

Children and Robotic Education

Children's Persistent Belief in Humanoid Robots, or, Why is your robot crying?

Corinna Mößlacher[2], Markus Wieser[1], and Andreas Bollin[1]

[1] Department of Informatics Didactics at University of Klagenfurt
[2] University College of Teacher Education Carinthia
`{markus.wieser|andreas.bollin}@aau.at`,
`corinna.moesslacher@ph-kaernten.ac.at`

Abstract. In society, the belief in the properties of robots is strongly influenced by their humanoid appearance and behavior as portrayed in the media – robots are regarded as "perfect" replicas of humans. This work describes children's understanding of robots, including their appearance, properties, and behaviors. This may influence their interaction with and their understanding of systems. When working with educational, non-humanoid robots in workshops with pupils, we observed that they tend not to see BeeBots or Ozobots, despite their name, as robots but tend to describe robots as human-like. After one of our robotics workshops with lower secondary school students who had no prior knowledge of robots, they were tasked with drawing a robot and doing a survey to describe this robot and robots in general. Unsurprisingly, most children drew robots that we would characterize as humanoids, but in the survey, they did not categorize them like this. On the one hand, they gave their robots names, human characteristics, properties, senses, and emotions; on the other hand, many pupils said their robots could not refuse or fail their commands and tasks.

Keywords: robotic education · secondary education · humanoid robots

1 Introduction

Robotics are an important topic in the educational field. They are not only a motivational factor when learning about computer science, but educational robots also help to understand different concepts and develop various skills that are relevant in STEM [7].

Our lab Informatik-Werkstatt[3] offers workshops on various topics for pupils of different ages. In recent years, there has been an increasing demand for robotics workshops to introduce pupils to the subject of programming in a playful way. Many teachers take the opportunity to work with material in our lab that is unavailable in schools. One of our offers is a workshop with different stations where

---

[3] https://www.rfdz-informatik.at/informatik-werkstatt/

you can learn about different educational robots and try their first programming steps. We select different robots to suit the age group. Our main target group for this topic is the 5th grade. According to their curriculum, they should work on the following content: the principle of input-processing-output and simple algorithms (sequences and loops). We work with BeeBots, ProBots, OzoBots, and Cubelets for this group. BeeBots and ProBots are used to design and try out the first simple algorithms (also with loops). When building simple robots with the Cubelets, the principle of input-processing-output is illustrated. OzoBots can be programmed using color codes to illustrate the cause-effect concept.

As an extra activity in the workshop, the students had to draw a robot. We observed that even though they are in the middle of a robot workshop that does not contain humanoid robots, the students are drawing humanoid robots and giving them human characteristics and behavior.

We ran additional similar workshops the following year to focus on this surprising result. Pupils attended the workshop and were then given the task of drawing a robot (without further instructions on what to draw). After finishing the drawing, they had to complete a questionnaire that included questions about their robot and robots in general. Our study aimed to answer the following questions about the students in our workshops:

- RQ 1: What ideas do children associate with the term ‘robot’? What do they look like in their opinion?
- RQ 2: How do the children define a robot?
- RQ 3: What characteristics do the children associate with robots (senses, abilities, ...)? Are there any connections between these characteristics, their perception of robots, and their interests?

This paper is structured as follows: We begin by presenting related work and our methodology. Next, we describe our findings and results, starting with survey results about the pupils’ drawings and then about the pupil’s view of robots in general. Finally, we conclude with a discussion of selected results.

2 Related Work

Children often perceive these machines as having human-like characteristics when interacting with robots. For instance, they believe that robots can see, hear, and even consider them potential friends, though they also recognize that robots do not possess complete knowledge [3]. This perception is shaped by their direct interactions and the responses they receive from the robots.

However, as children become more informed about robots’ inner workings and capabilities, their initial impression of robots as human-like beings diminishes. Their trust in robots declines once they understand their limitations and functions more. Despite this decrease in perceived human likeness and trust, children’s social relationships with robots remain largely unaffected [6]. Additionally, when robots disclose information about themselves during interactions, children's perception of the robot's emotional understanding decreases, but their perception of the robots' social role does not change [5].

Anthropomorphism and behavior significantly influence children's attitudes toward robots. While robots must exhibit humanoid traits, they should not be indistinguishable from humans [9]. There is no substantial difference in attitudes among children aged 8 to 14, but gender differences are notable: girls prefer humanoid robots more than boys do [8].

In our study, we aim to describe the profile of robots from the pupils' perspectives. We include various characteristics that can be used to define an individual's identity (because the pupils often describe robots as human-like) and seek to observe the connections between different personality traits to better understand the pupil's point of view.

3 Methodology

We held five workshops, each with one class of school children aged 10 to 12, in the summer semester 2024. In each workshop, the students completed the same activities: BeeBot, ProBot, Cubelet, and Ozobot. Each activity contained a short introduction, a few tasks to get to know the robot, and some time to experiment. They had about 25-30 minutes for each activity. At the end of the workshops, 117 (male: 33, female: 82, no response to this question: 2) participating students had about 20 minutes to draw a robot and complete the online survey. The children had little or no knowledge about robots (according to their teachers) and only a little experience with block-based programming.

With the survey, we want to confirm our impression that students have humanoid robots in mind when we talk about robots. Some students draw detailed illustrations, but others leave much room for interpretation. This makes it difficult to categorize the drawings for further evaluation. Therefore, we added a questionnaire that helps the students describe their robot. We only use the drawings to confirm the results of the survey. The drawings are intended to help the students convey their ideas before they answer the questions.

The survey consisted of a questionnaire, most of which were Likert-scale or single/multiple choice. Our standard questionnaire in our workshops includes six questions for the general interests following Holland (RIASEC: Realistic, Investigative, Artistic, Social, Enterprising, Conventional) [4, 1] (we have adapted the questions to an age-appropriate form including a visualization). The robot-specific questions were divided into (a) about their robot and (b) about robots in general.

The dimensions of the questions on the robot are based on the model of the five pillars of identity, according to Hilarion Gottfried Petzold. This comprises the following aspects: Body and health (here in the sense of appearance, senses, feelings, and abilities), social relationships, work and performance, financial security (not relevant here), values and meaning [2]. The following questions (translated from German for the sake of readability) were used. We divided the question into two parts. The first part contains questions about the pupil's drawings:

- Performance: Where is your robot on the scale? (7-Likert, Toaster - human - superhero)
- Appearance: My robot looks like ... (a) a person, (b) an animal, (c) a technical device, (d) a work tool, (e) something else (single choice)
- Senses: What senses does your robot have? (a) see, (b) hear, (c) feel, (d) smell, (e) taste, (f) balance (multiple choice)
- Abilities: When I give my robot a task, it is easy because it ... (a) can recognize and see for himself what he should do, (b) can listen to me and understand what I say to him, (c) can be programmed by me via buttons (multiple choice)
- Values and meaning: My robot occasionally fails to perform a task because ... (a) he is sometimes stubborn, (b) he doesn't always understand what I want from him, (c) That can't happen (single choice)
- Feelings: If a task does not work out ... (a) my robot can cry because it is sad (b) it doesn't matter because a robot is just a machine (single choice)

The second part is about robots in general:

- What is a robot? (open question)
- Appearance: Which of these pictures represents a robot? (multiple choice, given 8 examples)
- Abilities: A robot can ... (a) solve a very specific task for which it was programmed (e.g., water the flowers on the windowsill every day at 8 a.m.), (b) work on a task all by himself (e.g., find all the flowers in the house and decide whether the robot has to water them or not), (c) independently solve all tasks that a human could also fulfill, (d) can solve tasks that a human could not fulfill (single choice)
- Social relationships: Robots are ... (a) friends, (b) strangers, (c) part of the family, (d) Robots can't have friends (single choice)

The closed questions of the questionnaire were analyzed using descriptive statistics. For the open questions, we categorized the answers using keywords. In general, all open questions were answered briefly so that categorization was straightforward.

Due to the relatively small number of drawings categorized as humanoid (contrary to our assumption), we repeated this categorization ourselves. Two people analyzed the drawings independently of each other and assigned the drawings to categories according to certain criteria. For example, the drawings were categorized as human-like if features such as the head, face, and arms were recognizable. Animals were identifiable. The remaining drawings were classified as either "other life form" (e.g., eyes but no human or animal shape) or "no life form."

4 Findings and Results

4.1 Children's Robot classification

To gather insight into the pupil's perception of their robot, we let them draw and name it and asked questions about it. The questions were about a general classification, the appearance of their robot, the senses, perception, self-will, and ability to feel.

When we asked the children to classify their robots on a seven-point Likert scale from toaster to human to superhero, we found that they don't see their robots generally as toasters or superheroes. It was a relatively even distribution (Fig. 7a), but only 10 out of 117, which is 9%, classified their robot explicitly as human. If we reduce the scale to three instead of 7, combining The lower two, the three in the middle, and the upper two, there are 24 of 117 (21%) with a tendency to toaster, 53 (45%) rather human, and 40 (34%) superheroes.

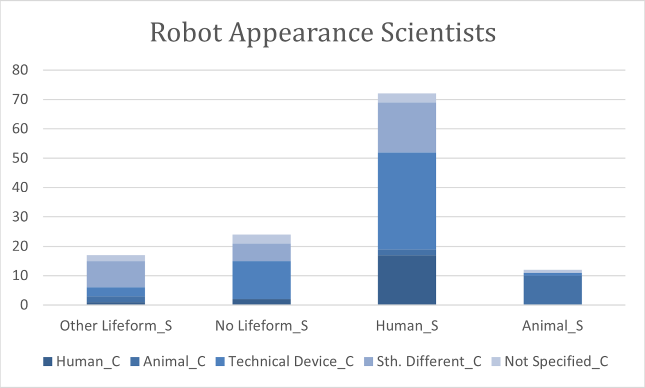


Fig. 1: Robot appearance classified by scientists (n=125). The categories marked with a C are the children's categories, the categories marked with S are the scientist's.

We then asked what their robot looked like. There were five options: Human, Animal, technical Device, Tool, or something completely different. 20 of 117 (17%) stated their robot has a human appearance, 14 (12%) said it looks like an animal, 50 of 117 (43%) said it was a technical device or tool, and 33 of 117 (28%) classified it as something completely different. So 29% of the students see their robot as a life-form, 43% as a tool, and 28% as something else (Fig. 7b), while more than 50% of the robots have human names. We also classified the drawings into four classes: Human, Animal, other lifeforms, and no lifeforms. Out of the 125 drawings, we found 72 Human-like robots (58%), 12 Animal robots (10%), 17 other lifeforms (14%), and 24 were no lifeforms at all (19%)(Fig. 1). So we found that a majority of the children's robots seem to have a human appearance and name but were not classified by them as such. In addition, we classified 33 of the 49 robots (67%), categorized as something else by the children, as humanoid and only 13 (27%) as no lifeforms.

When asked about their robots' senses, the students had six options. See, hear, feel, smell, taste, and keep balance. 94 of 117 (80%) said their robot could see, 86 Robots could hear (74%), 68 were able to feel (58%), 41 could smell (35%), 36 taste (31%), and 78 could keep their balance (67%) (Fig. 8a). The three most important senses for the robots are, therefore, to see, to hear, and to keep the balance, while it seems not to be necessary for a robot to smell or taste.

Another question was about the robot's perception. There were three possibilities: be able to recognize their task themselves, listen and understand, or be programmed via keys. 67 of 117 (57%) could recognize their task by themselves, and 84 of 117 (72%) could listen and understand the user's commands. This is unsurprising, as many students may be used to cleaning robots and voice assistants. Only 19 of 117 (16%) robots were programmable via keys. (Fig. 8b) On the other hand, this does not seem necessary for the students to command a robot.

We also asked the children about the robot's self-will. The first two options are the robot does not do a task because he is either self-willing or does not understand what the user wants. The third option is that it can't happen. We found 12 of 116 (10%) self-willed robots, 28 of 116 (24%) that don't always understand, and a majority of 76 of 116 (66%) where something like that could not happen. (Fig.9a)

Another item was about the robot's ability to feel. If the robot cannot do a task, he is either crying because he is sad, or it doesn't matter because he is a machine anyway. Interestingly, 68 of 116 (59%) stated their robot has feelings, while 48 of 116 (41%) said it can't because it's only a machine. (Fig.9b)

4.2 Children's Robots - Combined Items

We then combined the items regarding the children's robots. First, we found that 70% (14 of 20) of the robots classified with humanoid appearance by the children were in the middle of the toaster to superhero scale and therefore tending towards humanoid. This shows that our children do not see humanoid robots as super-powered, contrary to animal robots. Although the sample of robots classified as animals is pretty small, 50% (7 of 14) of the children say that their robot with an animal appearance is a superhero. In the children's perception, the 50 technical devices could be like toasters (32%), have human abilities (42%), or have superpowers (26%). The distribution there is nearly even, with a slight tendency to human abilities. Out of the 33 robots classified as something different, interestingly, 13 (39%) were classified as human, while 16 (41%) were defined as super-powered. So, 74% of these robots have at least human abilities.

When combining appearance and senses, we found, overall, independent from the robot's appearance, the most important senses are seeing, followed by hearing and holding the balance, while smelling and tasting are not that important. This is not true for those robots, defined as technical devices by the children, where 60% can feel, and 58% can hold their balance, and the robots defined as humanoid by our group, where feel and balance are equal at 61%. (Fig. 2) Interestingly, for Animal Robots, it is slightly more important to hear than to see (93% to 86%) and to smell than to feel (64% to 57%). The children mention the sense of smell more often than humanoid robots (64% to 40%). Also, according to them, holding their balance is more important for animals (79%) than for humanoid robots (75%).

When we combined the robot's appearance and perception, one result was that listening and understanding are always the most important. (Fig. 3) Even more important than recognizing tasks autonomously. The difference is greatest with Animal robots (93% to 50%). Even for humanoid robots, listening is more important than recognizing tasks. (75% to 65%) The option to be programmable via keys is unimportant for the children. With a maximum of 20%, technical devices are most programmable. When using our classification, the picture is more or less the same, but of the robots we classified as other lifeforms, 40% are programmable via keys.

Matching the Perception and Senses of the robots reveals that it doesn't matter if the robot can recognize tasks autonomously or is just listening and understanding. The sense distribution is more or less the same, but in comparison, for listening robots, to hear (80% to 76%) and interestingly hold their balance (70% to 64%) is slightly more important than for autonomous robots, whereas to see is more important for the autonomous ones (88% to 82%). If a robot is programmable, the percentage of every sense is higher than for the other two possibilities. (Fig. 4)

Another interesting combination is about the robot's feelings and their appearance. When considering the children's classification, 9 of 20 humanoid robots cry when they cannot fulfill a task, and for 11 of 20, it doesn't matter because they are machines anyway. So the children classified them as humanoid, and 55% cannot feel. On the other hand, there were 35 of 50, which are 70%, technical devices which are nevertheless able to feel. The categories of Animals and something different are more or less balanced regarding feelings. 8 of 14 animals and 16 of 32 different things can cry. (Fig. 5, upper table) The results differ slightly when combining our classification and the children's answers about their robot's feelings. Whereas the animals are more or less balanced (55% able to cry and 45% don't) here too, in our appearance classification, contrary to the children's version, 59% of the humanoid robots and 60% of the other lifeforms can cry, and 57% of the robots we classified as no lifeforms have feelings too. (Fig. 5, lower table)

4.3 Robots in general

To get some insight into the pupil's perceptions of robots in general, they were also asked to define a robot ("What is a robot?"). Their answers were categorized to get an impression of what is characteristic of a robot from the pupils' point of

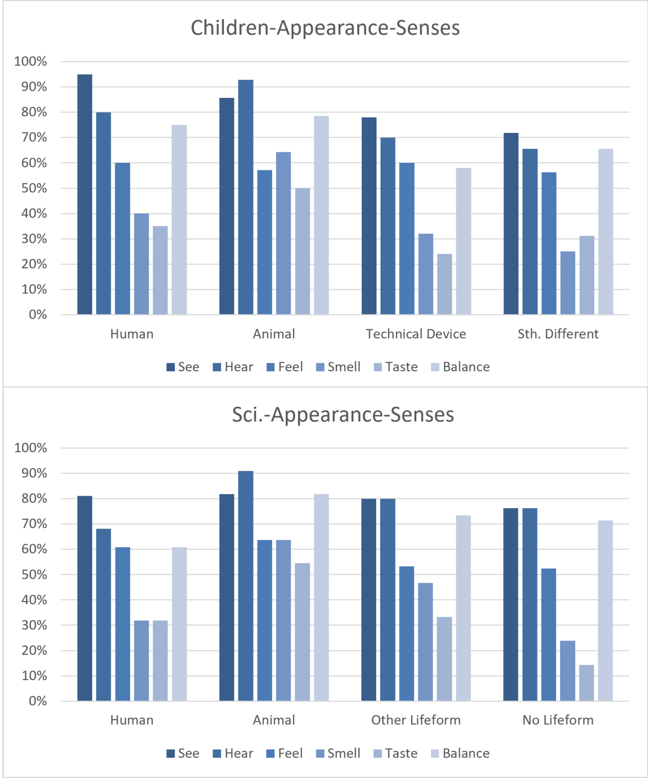


Fig. 2: Robot's appearance matched with the robot's senses. The upper diagram displays the senses and the classification of children's appearance; the lower is the scientist's version.

view. 100 pupils answered the question and described a robot in 1-2 sentences. In 58 answers, robots being machines or devices were mentioned, and 2 compared them to computers. 30 answers contained a description of "technical," and 12 mentioned them being "electric/electronic" (one of them contained both - due to the categorization of the mentioned aspects, multiple categories per answer are possible). 10 pupils mentioned that robots are programmable, and 6 that we can give them instructions. 3 pupils said robots are intelligent or have knowledge; 7 students even mentioned artificial intelligence. 30 students mentioned that robots are there to help us and do things for us. 7 students said they are human-like/humanoid, and many said they are friends. So, not many descriptions focus on the humanoid traits but more on technical or task-related.

We also presented the pupils with eight pictures. They had to decide whether the picture showed a robot or not. 104 of 116 pupils (90%) labeled the humanoid robot and 100 (86%) Robby, our lab logo, robots. More than half of the pupils also chose the BeeBot (73 of 116, 63%), ProBot (68 of 116, 59%), and robot

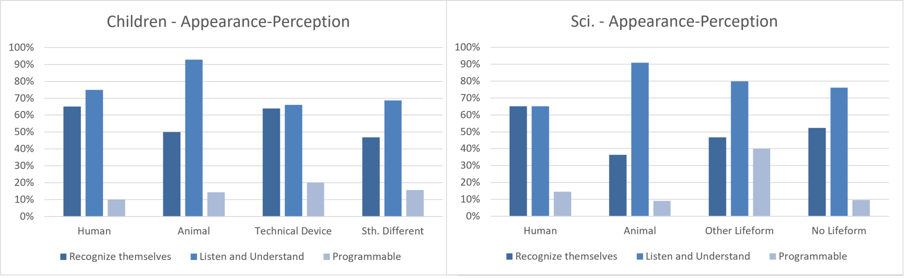


Fig. 3: Robot's appearance matched with the robot's perception. Left diagram is based on the children's appearance classification, the right one is the scientist's version. The bars in the diagram stand for "Recognize themselves", "Listen and Understand", and "Programmable". (n=116)

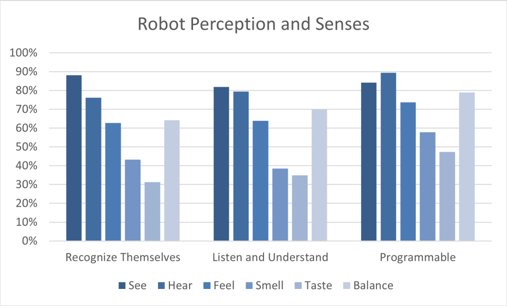


Fig. 4: Robot's perception matched with the robot's senses. It shows the distribution of the robot's senses, such as if they can recognize tasks themselves, listen and understand, or are programmable. The bars in the diagram stand for "See", "Hear","Feel", "Smell", "Taste", and "Balance". (n=116)

arm (61 of 116, 53%). Only a few chose the washing machine (29 of 116, 25%), the gantry system robot (24 of 116, 21%), and the toaster (31 of 116, 27%). 13 pupils (11%) labeled all of the pictures as robots.

The pupils were also asked to decide how robots know how to solve their tasks. They are provided the options "perform a specific task for which it has been programmed" (40%), "work on a task all by himself" (7%), "independently solve all tasks that a human could also fulfil" (22%) or "can solve tasks that a human could not fulfil" (31%).

For the pupils, robots are friends (61%). Only 5 (4%) decided that they are strangers, part of the family (16%), or can not have any friends (19%). Regarding whether robots are friends, there are hardly any differences between girls (64%) and boys (58%), but 33%

Children	Human	Animal	Tool	sth. Different
Cry when task fails	45%	57%	70%	50%
is only a machine	55%	43%	30%	50%

Scientists	Human	Animal	other lifeform	no lifeform
Cry when task fails	59%	55%	60%	57%
is only a machine	41%	45%	40%	43%

Fig. 5: Robot's appearance matched with the ability to feel. The upper table displays the feelings combined with the children's appearance classification; the lower table is the scientist's version. (n=116)

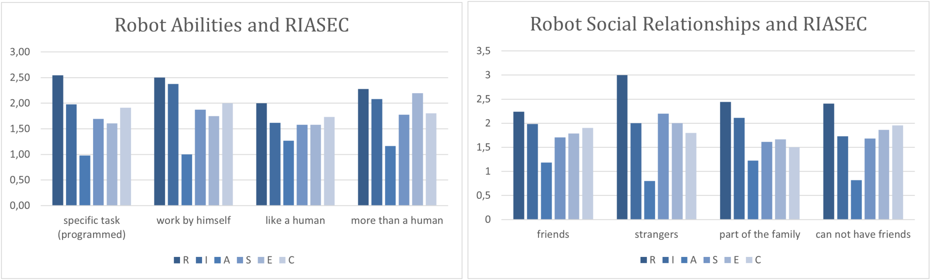
4.4 Robots in general - Combined Items

If we compare the results of the question about robot examples, the example of the BeeBot and the ProBot can be considered as one category. 57 % gave both as examples of robots. 35 % did not see either as a robot. This leaves only 8 % of responses that identified only one of the two as a robot. The situation is similar for the examples of the humanoid robot and the robot logo (78 % chose both, and 2 % chose neither). In comparison, there are different results for the examples of humanoid robot and robot arm: 39 % chose both, 8 % chose neither. 51 % identified the humanoid robot as the robot, not the robot arm. Only 3 % did the opposite (rounding errors). A toaster and washing machine were rarely selected and deliver comparable results.

The ascribed abilities differ slightly depending on the robot examples selected. Of the 104 children who recognized the humanoid robot, 40 % stated that they could perform certain tasks for which they were programmed. In the case of the BeeBot, 33 out of 73 children (45%) stated this. Similar results were achieved regarding relationships: Of the 104 children who recognized the humanoid robot, 57% stated that robots are friends. In the case of the BeeBot, 45 out of 73 children (61%) stated this. Major differences can be seen when we compare the examples differentiated by gender: 86% of the girls (70 out of 81) and nearly all the boys (32 out of 33) chose the humanoid robot. For the example of the BeeBot, the results of boys and girls are nearly the same (girls: 62%, boys: 64%).

When we categorized the definition of a robot, there can be seen some gender differences: 6 out of 33 boys (18%) mentioned "technical", but 30 of the 81 girls (37%). Similar results for "electric/electronic" (each 1%) can be observed. The term "machine or device" was used similarly (61%, 62%). Slight differences can be seen in the category "to help and do things": 7 of 33 boys (21%) and 24 of 81 girls (30%).

The pupils were asked about their General Interests following the RIASEC model at the beginning of the questionnaire. The mean values in the interests are as follows: Realistic - 2.34, Investigative - 1.96, Artistic - 1.10, Social - 1.71, Enterprising - 1.79, Conventional - 1.84. Differentiated by the chosen abilities, there can be seen some differences in the RIASEC profile of the pupils (Fig. 6a), e.g., pupils that have the opinion that robots can work by themselves on any task



(a) (b)

Fig. 6: Robot Abilities and Social Relationships combined with Children General Interests (n=116)

have higher results in the Investigative dimension (mean: 2.38). Even higher differences can be seen when the RIASEC dimensions are compared with the results of the questions about the social relation to robots (Fig. 6b). The highest results in the Realistic dimension are achieved by pupils who see robots as strangers (mean: 3).

5 Discussion

When evaluating the first part of the questionnaire, it became evident that the students have very specific ideas about robots in certain areas (e.g., the ability to see, hear, and maintain balance). Surprisingly, our concept of humanoid robots does not align with the students' categorizations. Our assumptions (face, eyes, arms, etc.) are insufficient to classify a robot as a humanoid. Technical details (e.g., edges and mechanical joints) may be enough for the children not to associate the robots with a human appearance. Robots can still have human senses, feelings, will, and friends. Regardless of appearance, however, robots are attributed to human senses and qualities (e.g., emotions). In their general definition of robots, the children frequently mention technical details rather than human characteristics. When confronted with examples, children mainly classify humanoid robots as robots (but not, for example, robot arms).

6 Conclusio

The categorization of the appearance of the robots (RQ1) differs between the pupil's own categorization and ours. The pupils described fewer robots as humanoid than we did. However, when we examine the data in detail, we find that they associate robots more with a humanoid appearance, but they would not necessarily describe them as humanoid.

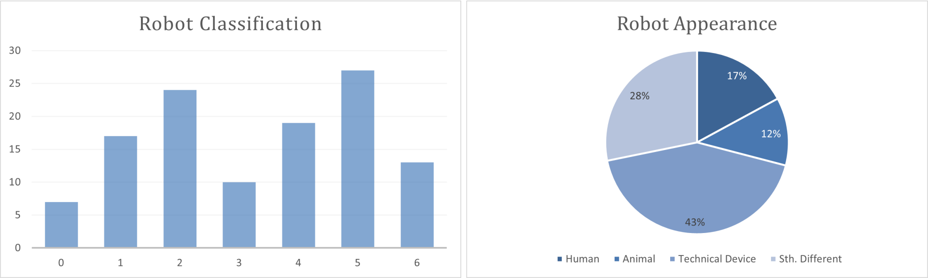
The pupil's general definition of robots (RQ2) focuses on technical details without mentioning their appearance or humanoid traits. In general, pupils view robots as individuals capable of forming friendships and experiencing emotions. When the data is stratified based on the pupils' self-reported interest levels (RQ3), some differences emerge regarding the robots' abilities. For instance, pupils who perceive the robots' abilities in doing specific tasks or being programmed tend to exhibit higher levels of interest in the Realistic domain and lower levels of interest in the Artistic domain. Furthermore, the questions about the relationships and the robot examples show that the results vary according to the participants' general interests. The profiles can be differentiated according to the pupils' interests.

The study results offer valuable insights that can improve workshops beyond our own. It's essential to refine materials to better meet the target audience's needs. Differences in terminology, such as "robot" and "humanoid robot," between students and instructors require a more focused approach. Targeted tasks are crucial for conveying concepts and correcting misunderstandings. Workshops focusing on non-humanoid robots may not fully achieve educational goals, highlighting the need for a more varied approach in robot-based learning activities.

References

1. Bergmann, C., Eder, F.: AIST-R. Allgemeiner Interessen-Struktur-Test mit Umwelt-Struktur-Test (UST-R) – Revision. Manual. Hogrefe, 1 edn. (2005)
2. Eremit, B., Weber, K.F., Eremit, B., Weber, K.F.: Die fünf säulen der identität. Individuelle Persönlichkeitsentwicklung: Growing by Transformation: Quick Finder—Die wichtigsten Tools im Business Coaching pp. 47–52 (2016)
3. Escobar-Planas, M., Charisi, V., Gomez, E.: "that robot played with us!" children's perceptions of a robot after a child-robot group interaction. Proceedings of the ACM on Human-Computer Interaction **6**(CSCW2), 1–23 (2022)
4. Holland, J.: Exploring careers with a typology - what we have learned and some new directions; am. psychol. American Psych., Vol.51(4), pp.397-406 (1996)
5. van Straten, C.L., Peter, J., Kühne, R., Barco, A.: On sharing and caring: Investigating the effects of a robot's self-disclosure and question-asking on children's robot perceptions and child-robot relationship formation. Computers in Human Behavior **129**, 107135 (2022)
6. Straten, C.L.v., Peter, J., Kühne, R., Barco, A.: Transparency about a robot's lack of human psychological capacities: effects on child-robot perception and relationship formation. ACM Transactions on Human-Robot Interaction (THRI) **9**(2), 1–22 (2020)
7. Toh, L.P.E., Causo, A., Tzuo, P.W., Chen, I.M., Yeo, S.H.: A review on the use of robots in education and young children. Journal of Educational Technology & Society **19**(2), 148–163 (2016)
8. Tung, F.W.: Influence of gender and age on the attitudes of children towards humanoid robots. In: Human-Computer Interaction. Users and Applications: 14th International Conference, HCI International 2011, Orlando, FL, USA, July 9-14, 2011, Proceedings, Part IV 14. pp. 637–646. Springer (2011)
9. Tung, F.W.: Child perception of humanoid robot appearance and behavior. International Journal of Human-Computer Interaction **32**(6), 493–502 (2016)

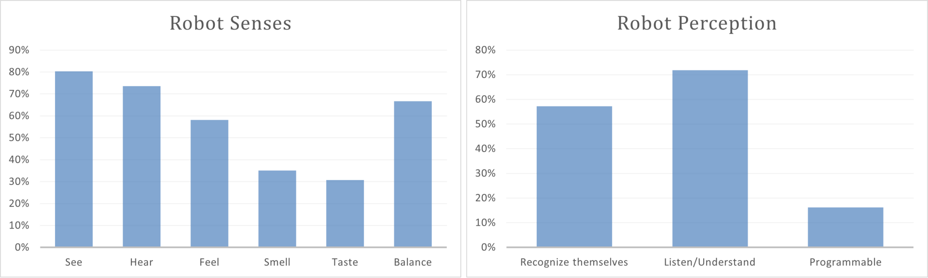
A Additional Diagrams



(a) Classification: The scale runs from 0 (toaster) via 3 (human) to 6 (superhero)

(b) Appearance: 17% human, 12% animal, 43% tool, 28% something different

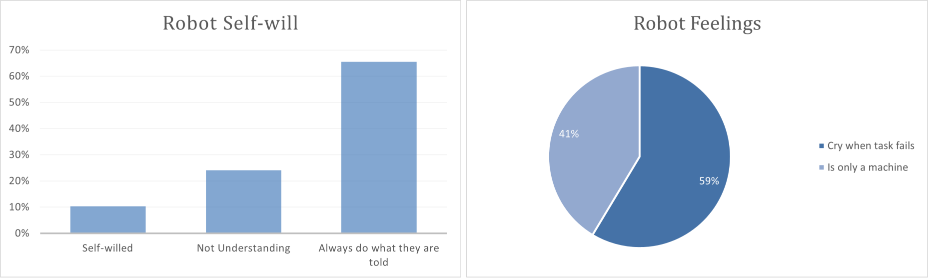
Fig. 7: Classification and appearance of the student's own robots (n=117)



(a) The percentage of robots able to see, hear, feel, smell, taste and hold the balance (multiple answers possible)

(b) Perception: recognize themselves, listen and understand, programmable (multiple answers possible)

Fig. 8: Senses and perception of the children's robots (n=117)



(a) Self-will: The grade of self-will the children's robots have, from self-willed to always do what they are told

(b) Feelings: The robot is crying if a task is not working out or it doesn't matter, because it is just a machine

Fig. 9: Self-will and feelings of the children's robots (n=116)

Poster Descriptions

“Are the Insides of a Smartphone Different from a Desktop?”: Study about Preconceptions of 6th and 8th Graders Comparing Hardware Components

Anna Yaghobová[1][0000−0002−2102−7515], Anna Drobná[1,2][0000−0003−2714−9529], Marek Urban[3][0000−0003−2772−1388], and Cyril Brom[1][0000−0001−5945−0514]

[1] Faculty of Mathematics and Physics, Charles University, Prague, Czechia
[2] Faculty of Education, Charles University, Prague, Czechia
[3] Institute of Psychology, The Czech Academy of Sciences

Abstract. This study investigates sixth and eighth graders’ understanding of the differences and similarities between smartphone and desktop hardware components. Through semi-structured online interviews with 49 students, the research reveals that children most often compared the number or size of components in the devices, or noted that both types of devices have some components (e.g., *memory*) but not others (e.g., *fan*). Additionally, eighth graders, compared to sixth graders, tended to possess more scientifically accurate views, particularly in recognising key components like processors in both smartphones and desktops and were more likely to be aware of invisible components (e.g., *processor*). These findings underscore the importance of tailoring educational materials to address these preconceptions. Future work will focus on developing educational resources that deepen students’ understanding of desktop architecture and components, guiding them from intuitive notions to scientifically grounded knowledge.

Keywords: Computer Science · Education · Lower secondary school · Conceptions · Hardware · Hardware components

1 Introduction

Teaching about hardware, computer components, and the internals of computers remains important, as emphasised in current curriculum documents (e.g., [3]). This knowledge is also practical for everyday life—most of us have had to purchase new electronics (e.g., a smartphone) or troubleshoot a malfunctioning device. To effectively teach this topic, quality educational materials are essential. However, current materials are often not evidence-based. It is well understood that such materials must build on children’s preconceptions to help them develop concepts that align with current scientific knowledge [4]. Research shows that children’s understanding of digital technologies is typically intuitive and fragmented, shaped by everyday experiences (e.g., [5]). Furthermore, a deep, conceptual grasp of underlying principles is rare, even among adolescents (e.g., [2]).

A key idea in computer hardware education is recognising that similar components with similar purposes exist across different digital devices. While some research has explored children's preconceptions about desktop computer components (e.g., [6], [5]), the comparative aspect regarding other devices is less frequently studied (e.g., [1], [6]). This study investigates sixth and eighth graders' understanding of the similarities and differences between smartphone and desktop computer internal components. We are not concerned with normative correctness, as every preconception can be valuable for developing effective teaching materials [4].

We chose this age group because adolescents are generally familiar with both desktop computers and mobile devices [2] and should recognise some differences and similarities.

2 Method

This study was part of a larger mixed-methods project. As part of this project, children participated in 45-minute semi-structured online interviews conducted via Zoom, during which they answered the following question on comparing desktop computers and cell phones: "How does the inside of a computer differ from the inside of a smartphone?". This question was followed by additional follow-up questions, when relevant. This poster presents findings specifically from this segment of the interviews.

The study involved 25 sixth-graders (around 12 years old, 13 girls) and 24 eighth-graders (around 14 years old, 12 girls), all of whom had minimal prior exposure to computer science topics in school. Participants were recruited from various regions across Czechia through Facebook and a network of teachers to match the characteristics of the general school population in the Czech Republic (except for neglected audiences), and they were compensated with table games or LEGO sets valued at approximately 20 EUR.

We conducted an inductive thematic analysis on the transcribed interviews using Atlas.ti 24 for Mac, with two coders reaching a consensus on the coding. Later, we also did a frequency analysis for each preconception. It is important to mention that the occurrence of a preconception in a child means that the child spontaneously mentioned it. Therefore, if a preconception has a low frequency, it means that not many children spontaneously mentioned it, not that many children do not have it. It could be that they didn't talk about it (due to various reasons) or they simply didn't know. It could happen that one child could have multiple preconceptions.

3 Results

We identified 15 unique preconceptions, all listed in Table 3. The most common preconception was that *the inside of a smartphone is identical to a desktop* *computer, only everything is smaller* ($n = 35$; 71%). Some children believed that certain components are found in both smartphones and computers, such as *memory* ($n = 4$; 8%), *battery* ($n = 6$; 12%), and *processor or some centre* ($n = 4$; 8%). We did not distinguish between types of memory.

Fifteen children (31%) reported that *a smartphone has far fewer components than a computer.* Some components were identified as absent in smartphones, such as a fan ($n = 9$; 18%) and *processor or some centre* ($n = 1$; 2%), or *as being fewer in number*, such as *cables* ($n = 4$; 8%). Interestingly, while 11 children (22%) believed that *a computer has more memory than a smartphone*, one child (2%) expressed the opposite view, stating that *the smartphone has more memory than the computer.*

Eighth-graders descriptively tended to have more preconceptions related to “invisible” components (e.g., *both smartphone and desktop have a processor or centre*), though these results were not statistically significant. Overall, the results suggest a low level of awareness about this topic among children in both age groups.

Table 1. Preconceptions about the differences and similarities between smartphone and desktop components. Percentages are expressed from the number in the given group.

Preconception	**Total**	**Grade 6**	**Grade 8**
inside of a smartphone is identical to a desktop computer, only everything is smaller	35 (71%)	16 (64%)	19 (79%)
smartphone has fewer components than desktop	15 (31%)	9 (36%)	6 (25%)
desktop has bigger memory (of unspecified type) than smartphone	11 (22%)	7 (28%)	4 (17%)
desktop is more efficient than smartphone	10 (20%)	4 (16%)	6 (25%)
smartphone does not have a fan	9 (18%)	4 (16%)	5 (21%)
both smartphone and desktop have a battery	6 (12%)	1 (4%)	5 (21%)
both smartphone and desktop have a centre/processor	4 (8%)	0 (0%)	4 (17%)
both smartphone and desktop have a memory (unspecified)	4 (8%)	0 (0%)	4 (17%)
smartphone has fewer cables than a desktop	4 (8%)	2 (8%)	2 (8%)
both smartphone and desktop have unspecified discs	2 (4%)	1 (4%)	1 (4%)
components are stored differently in a smartphone than in a desktop	2 (4%)	0 (0%)	2 (8%)
components are wired differently in the smartphone and in the desktop	2 (4%)	1 (4%)	1 (4%)
smartphone has metal parts	1 (2%)	0 (0%)	1 (4%)
smartphone does not have a centre/processor	1 (2%)	1 (4%)	0 (0%)
smartphone has bigger memory than a processor	1 (2%)	1 (4%)	0 (0%)

4 Discussion and Conclusion

Our findings strengthen and extend the abovementioned existing body of literature in this area by identifying 15 preconceptions. Some of these preconceptions are partially described (albeit in different words) in the literature (e.g., [1], [6]), and we confirm their existence in a new, younger, and larger sample. Others are novel (e.g., *smartphone does not have a fan*). At the same time, the results offer practical implications, as they can assist teachers in focusing their lessons on highlighting the similarities and differences between smartphones and desktop computers, emphasising that elementary components such as the processor can be found in all such electronic devices, but that there may also be differences in material, size or wiring.

When comparing the two age groups, we observed that older students were generally more likely to even identify specific components. The underlying reasons for this were not revealed in the interviews. We speculate that eighth graders may be more familiar with the internals of a computer than a smartphone, leading them to make assumptions about its components. Generally, eighth graders tended to have more scientifically accurate preconceptions (e.g., *both smartphone and desktop have a centre/processor*).

However, this research has certain limitations. The sample size and diversity could be improved for better generalisability; especially, we were unable to recruit participants from disadvantaged communities. Additionally, the online format reduced our control over the children's activities during interviews.

Our next step will be to develop educational materials tailored to different levels of children's knowledge about hardware components and architecture, based on their preconceptions. The goal is to guide students from these preconceptions to a more accurate, scientifically grounded understanding.

Acknowledgments. This study was funded by GAUK 360322 and GAČR 22-20771S.

References

1. Brinda, T., Braun, F.: Which computing-related conceptions do learners have about the design and operation of smartphones? results of an interview study. In: Proceedings of the 12th Workshop on Primary and Secondary Computing Education. p. 73–81. WiPSCE '17, ACM, NY, USA (2017)
2. Brom, C., Yaghobová, A., Drobná, A., Urban, M.: 'the internet is in the satellites!': A systematic review of 3–15-year-olds'conceptions about the internet. Education and Information Technologies **28**(11), 14639–14668 (2023)
3. CSTA: Csta k–12 computer science standards, revised 2017. (Mar 2017), https://csteachers.org/k12standards/
4. Diethelm, I., Hubwieser, P., Klaus, R.: Students, teachers and phenomena: educational reconstruction for computer science education. In: Proceedings of the 12th Koli Calling International Conference on Computing Education Research. p. 164–173. Koli Calling '12, ACM, NY, USA (2012)
5. Mertala, P.: Young children's perceptions of ubiquitous computing and the internet of things. British Journal of Educational Technology **51**(1), 84–102 (2020)
6. Pancratz, N., Diethelm, I.: "draw us how smartphones, video gaming consoles, and robotic vacuum cleaners look like from the inside": students' conceptions of computing system architecture. In: Proceedings of the 15th Workshop on Primary and Secondary Computing Education. WiPSCE '20, ACM, NY, USA (2020)

Card-Based Activity to Raise People's Awareness about how the Digital World Works

Sébastien Combéfis[1,2][0000−0002−8987−9589]

[1] Computer Science and IT in Education ASBL, 1348 Louvain-la-Neuve, Belgium
[2] AEI Consulting, 1348 Louvain-la-Neuve, Belgium
`sebastien@combefis.be`
https://sebastien.combefis.be

Abstract. Digital technologies are omnipresent in our daily lives. Since people have to interact with them, it is crucial that they understand how the digital world works. This contribution proposes an activity based on cards to raise people's awareness on computer science concepts. During a 60 to 90 minutes workshop, participants working in groups of 2 to 4 people have the opportunity to analyse a real-world situation involving an interaction with a computer system. The cards and a board both structure their analysis and guide them through their thinking. The activity has been tested once and improvements are planned as future work.

Keywords: Informatics · Digital world concepts · Card game.

1 Introduction

Digital technologies are spreading, reaching mostly every aspects of our daily lives and this will be even more true in the future. To be able to interact genuinely with them, humans need to understand a bit how they are working. This contribution proposes a card-based activity that can be conducted in a short time to make their participants aware of what is the digital world. The activity makes the participants think, starting from a real-world situation they have to analyse regarding the components the analysed computer system is made of.

1.1 Related work

Using games to teach computer science concepts is a very common approach that has been used in many projects [7, 3, 4]. In particular, card-based games or activities are being developed, should it be for pupils and targeted to broad concepts [5] or for older students and on more specific subjects [6, 1].

1.2 Motivation

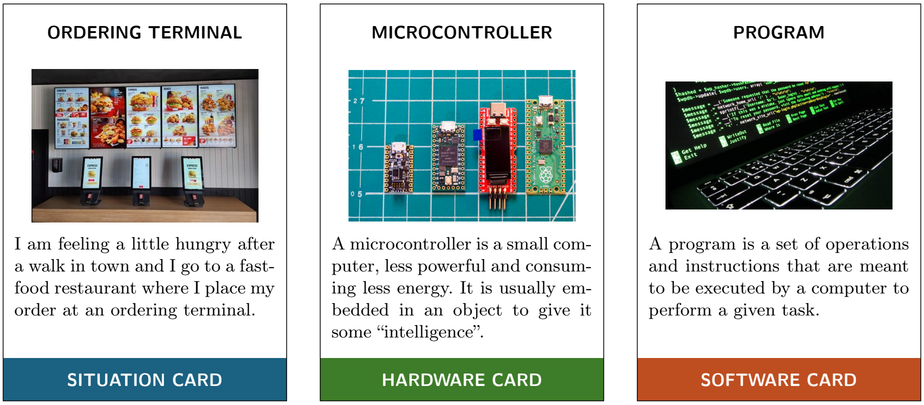
Designing an activity based on cards and that can be used with a limited amount of time makes it suitable for many kinds of events. Also, having a pen-and-paper unplugged activity makes it more engaging and motivating [2]. The proposed activity can also be easily adapted to different age groups and learning goals, simply by selecting the cards to use or by designing new cards.

2 Card Game Design

The proposed activity to raise awareness among people about how the digital world works is based on cards. These latter are organised in different categories depending on their role in the activity. The proposed activity lasts between 60 and 90 minutes and can be organised in groups of 2 to 4 people.

2.1 Cards

The spirit of the proposed activity being to be connected to the real world, the first category of cards is the “situation cards”. These cards describe a situation in which there is an interaction with a computer system. As shown on Figure 1 (left card), these cards have a title, a picture and a description text. During the activity, participants have to use cards from other categories to analyse the situation they chose to work on. Figure 1 shows one “hardware card” and one “software card” (on the right). These two categories are at the heart of the proposed activity as any computer system always consists of at least one hardware and one software element. Participants will have to identify which hardware and software components may be involved in the computer system they are analysing, starting from the situation they are working on. Other categories of cards have been defined, like “input cards” and “output cards”.



I am feeling a little hungry after a walk in town and I go to a fast-food restaurant where I place my order at an ordering terminal.

A microcontroller is a small computer, less powerful and consuming less energy. It is usually embedded in an object to give it some “intelligence”.

A program is a set of operations and instructions that are meant to be executed by a computer to perform a given task.

Fig. 1. Situation cards are the starting point of the activity, proposing a real-world situation to be analysed. Hardware and software cards are use to make participants aware of invisible parts of computer systems.

2.2 Activity board

To organise the activity, each group of participants receives a sheet of paper which is the activity board, used to guide the activity. Figure 2 shows the board which consists of four areas. When analysing the real-world situation in the activity, the participants first have to go through all the hardware and software cards, identifying those which they think are involved in the computer system under analysis. The group then writes for which purpose each chosen hardware and software component is used in the computer system. They then think about the input provided to the system and the output that it produces, to refine their analysis. Each group then presents the results of their work in front of everyone, to get a direct feedback from the other groups and from the animator.

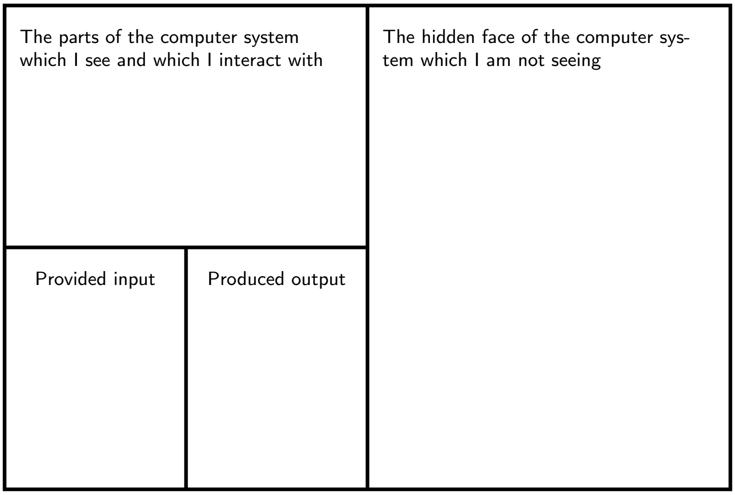


Fig. 2. The activity board consists of four areas that are guiding the analysis produced by the participants based on the cards.

3 Conclusion and Future Work

To conclude, this contribution presents an activity based on cards that can be used to help raise people's awareness about how the digital world works. The proposed activity has been tested with three groups of adults who never got any courses related to computer science in their education. The feedback collected right after the workshop was positive as the participants felt that they learned things they were not aware of. The activity also sparked the curiosity of the participants to know more about informatics. Future work includes adding new cards in each category and also attach a level on each card, to ease the design of several activities adapting to different audiences.

Acknowledgments. The author thanks his work colleagues Didier, Ludivine, Marie, Omar, Olivier and Sophie who tried out a first version of the card game and provided relevant feedback to improve the activity.

References

1. Baker, A., Navarro, E.O., van der Hoe, A.: An experimental card game for teaching software engineering processes. Journal of Systems and Software **75**, 3–16 (2005)
2. Bell, T., Alexander, J., Freeman, I., Grimley, M.: Computer science unplugged: School students doing real computing without computers. The New Zealand Journal of Applied Computing and Information Technology **13**(1), 20–29 (2009)
3. Combéfis, S., Beresnevičius, G., Dagienė, V.: Learning programming through games and contests: Overview, characterisation and discussion. Olympiads in Informatics **10**, 39–60 (2016)
4. Combéfis, S., Stupurienė, G.: Bebras based activities for computer science education: Review and perspectives. In: Proceedings of the 13th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives (ISSEP 2020). pp. 15–29 (2020)
5. Dagienė, V., Futschek, G., Koivisto, J., Stupurienė, G.: The card game of bebras-like tasks for introducing informatics concepts. In: ISSEP 2017 Online Proceedings (2017)
6. Denning, T., Lerner, A., Shostack, A., Kohno, T.: Control-alt-hack: the design and evaluation of a card game for computer security awareness and education. In: Proceedings of the 2013 ACM SIGSAC conference on Computer & Communications Security. pp. 915–928 (2013)
7. Videnovik, M., Vold, T., Kiønig, L., Bogdanova, A.M., Trajkovik, V.: Game-based learning in computer science education: a scoping literature review. International Journal of STEM Education pp. 10–54 (2023)

Development of Interactive Database Courses Using aDBenture, H5P and Storytelling

Nina Lobnig, Claudia Steinberger, Andreas Bollin

University of Klagenfurt
`{firstname}.{lastname}@aau.at`

Abstract. The ability to work with data is increasingly vital in today's digital age, and databases are still an essential part of computer science education. The key concepts are part of beginner university courses and secondary school classes. However, many current courses and textbooks contain dull examples, and the motivation to learn about databases doesn't seem particularly strong. In this submission, we want to present our approach in developing two massive open online database courses. We pursue a playful storytelling approach and work with H5P elements and aDBenture (an online tool for SQL queries) to make learning more interactive and motivating and utilize some of the advantages of game-based learning.

Keywords: database education · database teaching · game based learning · massive open online courses

1 Motivation and Background

Databases are omnipresent in the age of digitalization - they are needed whenever data is stored, retrieved, and analyzed. In this respect, relational databases are still the most widely used and form the foundation [4]. Understanding and managing the storage of large amounts of data is still (increasingly) important, whether as part of general education, computer science studies, or in emerging professional fields (like data science). Learning about databases should, therefore, be well-designed and organized in a way that motivates learners.

In 2020, we got the opportunity to develop a two-part open online course for databases as part of a project[1] named *eInformatics@Austria (publicly called 1ng0)* where five Austrian universities worked together in designing, creating, and evaluating massive open online courses (MOOCs) for introductory computer science topics. The target audience was freshmen students in higher education, interested students, and secondary school teachers. The goal was to make the learning material interactive and motivating so that it differs from the often uninspiring examples in textbooks and courses. Additionally, we wanted them to be easily reusable so that it is possible to use parts of the course(s) and use the course in different settings. In the following, we describe the concept behind our database courses and how we tackled all these ambitions.

---

[1] This project was funded under the 2019 Call 'Digital and social Transformation in University Education' by the Federal Ministry of Education, Science and Research.

2 The Concept behind our MOOCs

Competencies and Course Content. First, we designed the MOOCs in a way that requires no special prior knowledge. To adapt the course content to the relevant target groups, we based our selection of content on the competencies of the ACM/IEEE Computer Science Curriculum 2013 [1] and the core concepts from the database courses at our university, which largely also align with the school curricula. From this, we narrowed our target competencies and defined our underlying competence model with three resp. four levels (based on Bloom's taxonomy [8]). We have learning materials that provide theoretical knowledge using small examples in which the learner takes on the role of a mere listener (level 1). Then, we have those in which the learner participates in a story and can demonstrate their understanding of the content through small interactions and have solutions explained to them (level 2). there are also exercises for solving tasks independently and applying or combining learning content (levels 3, 4).

Regarding the content: The first course[2] is about the design of a database - from modeling with UML diagrams to relational schemas as well as the quality of those schemas and corresponding anomalies. The second course[3] is mainly focused on SQL databases - from a brief introduction to relational databases and mathematical basics (Relational Algebra) to the creation and querying of SQL databases. Learning how to write multiple SQL queries (including aggregate functions and subqueries) forms the central part of this course.

Structure of the Course Units. We focused on a coherent content structure and free access and navigation between units when organizing the courses. The first unit is an overview and introductory unit. Each unit contains an introductory short text and a list of keywords and questions that can be answered after completing the unit (as recommended by Handke [7]). This provides a good overview, creates the possibility of differentiation for participants, and makes it easier to find content again. Each unit starts with a task (in the context of a story) defining a goal we want to achieve. This should provide an impulse to engage with the learning materials (as their content is needed to solve the task) and also help to activate and motivate learners. It also contributes to differentiation, as learners can try to solve this task directly and thus have a good indication of whether the unit's content is relevant to them or whether they have already mastered it. The task solution is given for comparison and reference. Each unit also ends with a multiple-choice quiz on its main content, marking the unit as completed.

Activating Learners through Storytelling, aDBenture, and H5P. Our main focus is activating and motivating learners. That's why we don't have a professor explaining content in a classroom setting. Instead, our animated (female) expert, Sam, appears in different settings, like a police station or outdoors.

---

[2] The courses are in German, but description available in English as well: https://imoox.at/course/Datenbankentwurf

[3] https://imoox.at/course/SQL-Datenbanken

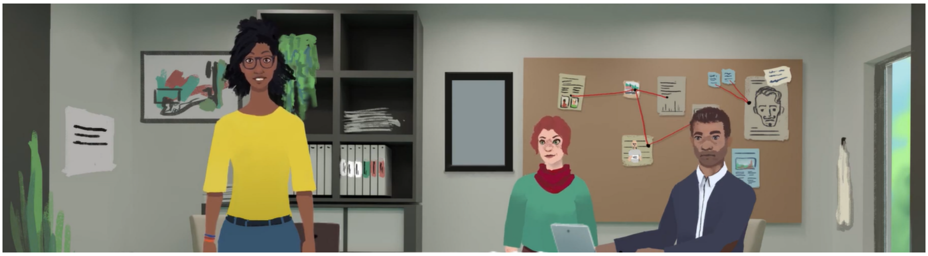


Fig. 1. Screenshot from a video with our expert Sam and the police duo

She explains content while walking through a park or chatting with friends in a bakery, usually embedded in a storyline. Storytelling is a central part of our learning materials, and learners are often involved in helping the police. E. g., they have to solve several criminal cases at the police station using SQL queries, such as finding a dog kidnapper based on evidence from the threatening letter. This storytelling approach is also part of the online tool we used for our querying exercises in SQL (although it was also possible to work with PGAdmin). It is a free online web tool called aDBenture[4] that is made for SQL exercises using storytelling and uses parts of the advantages of game-based learning [3][11].

Our videos focus on the person speaking, including eye contact, moving away from continuously visible slides, and the focus on displayed code. This follows recommendations for videos regarding decreased concentration in slide presentations without face-to-face contact [6]). For the same reason, we kept the duration of our videos relatively short. To further engage learners, we take advantage of H5P elements. All our level 1 and 2 contents are either interactive H5P videos or interactive H5P books with multiple choice questions, gap fills, drag-and-drop, and other task types. The interactions encourage thinking along and deeper learning, breaking up the content designed to keep concentration high. Research has shown that it drops significantly after about 10 minutes and can thus be improved [2][6].

(Re)Use of the Course. We also paid particular attention to possible uses and reuses of the course and its learning materials. MOOCs can be used in various settings: As a standalone, as an additional resource, or as a main driver in a course at a university or school (e.g., in a flipped or inverted classroom setting) [5][9]. In addition to that, we also considered modular use to facilitate the individualization and reusability of the course. For example, by allowing teachers to use only parts of the course in their teaching without compromising the usefulness and comprehensibility of the content. To achieve this, we designed and built our course in a particular, competency-based, modular way (according to the competence-oriented MOOC life cycle [10]), and our stories only cover one learning object at a time. Additionally, we licensed everything as open educational resources (under CC-BY). Overall, this makes the courses easy to maintain and extend, as content can easily be adapted or changed.

---

[4] Note: The Webtool is available in three languages, we used it in German

Other Aspects. All our videos have available subtitles and are supplemented with an alternative text document (transcript with the most important images from the video) to be more accessible and enable participation for different learners. The gender and diversity aspect was important to us as well. That's why we deliberately created a female, non-white role model in our expert Sam, and the mixed police duo Lara Fischer and Amar Kovać play a central role in our stories. We take great care to avoid recurring stereotypes and distribute the parts of speech evenly - or sometimes intentionally female-dominated (with the idea of reversing the roles, following the deconstructivist approach in gender research).

3 Conclusion

Our MOOC has been carefully thought through, considering various aspects. We rely on a mixture of storytelling, interactive H5P elements, and diverse learning materials and use an online SQL query tool that supports our approach. We have already tested the courses as an added value in university courses and evaluated and improved them. The feedback received from the students and the MOOC participants was overall positive. Nevertheless, the courses are designed for a broader target group and different usage scenarios to be suitable for our students as well as other universities interested school students, and teachers.

References

1. ACM and IEEE CS: Computer Science Curricula 2013. ACM, NY, USA (2013)
2. Bollin, A.: Videos als Lernobjekte in MOOCs und POOCs. In: Reichel, E., Orasche, B. (eds.) Lernen mit Erklärvideos. IMST and IUS, University of Klagenfurt (2021)
3. Burgstaller, N., Morak, M., van der Rijst, C., Steinberger, C.: aDBenture - Learning SQL in a Game-based Way. In: Proceedings of EdMedia 2023. AACE, Vienna, AT
4. DZone: Trend Report: Database Systems - Architecture Advancements and Data Storage Reimagined (september 2023)
5. Ebner, M., Schön, S., Braun, C.: More Than a MOOC - Seven Learning and Teaching Scenarios to Use MOOCs in Higher Education and Beyond. Springer Singapore (2020)
6. Guo, P.J., Kim, J., Rubin, R.: How video production affects student engagement: an empirical study of MOOC videos. In: Proceedings of the First ACM Conference on Learning @ Scale Conference. p. 41–50. L@S '14, ACM, NY, USA (2014)
7. Handke, J.: Handbuch Hochschullehre Digital - Leitfaden für eine moderne und mediengerechte Lehre. Tectum, Baden-Baden, 3 edn. (2020)
8. Johnson, C.G., Fuller, U.: Is Bloom's taxonomy appropriate for computer science? In: Proceedings Koli Calling 2006. p. 120–123. Baltic Sea '06, ACM, NY, USA
9. Pérez-Sanagustín, M., Hilliger, I., Alario-Hoyos, C., Kloos, C.D., Rayyan, S.: H-MOOC framework: reusing MOOCs for hybrid education. Journal of Computing in Higher Education **29**(1), 47–64 (2017)
10. Steinberger, C., Lobnig, N., Morak, M.: Increasing the reusability of moocs via competence-oriented development. In: Proceedings of IEEE LWMOOCS 2023
11. Tang, K.Y., Hsiao, C.H.: Game-Based Research in Education: A Key-Route Main Path Analysis of Literature (2008-2017). ICIEI '18, ACM, NY, USA (2018)

DigiFit4All - Advantages and Challenges of Competency-Based and Personalized Open Online Courses (POOCs)

Tatjana Angermann, Stefan Pasterk, and Andreas Bollin

University of Klagenfurt, Informatic Didactics, Klagenfurt, Austria
`{tatjana.angermann, stefan.pasterk, andreas.bollin}@aau.at`
https://www.aau.at

Abstract. Over the past decades, Massive Open Online Courses or short MOOCs have become increasingly popular. Although they offer many benefits, some challenges have emerged over time. Mass courses are needed to address increased demands, diversity, and heterogeneity among learners. Therefore, the DigiFit4All project was initiated to develop Personalized Open Online Courses (POOCs) based on competency models. It also includes creating course materials related to digital skills and informatics. This poster provides an overview of the DigiFit4All project, highlights relevant points, and illustrates some advantages and challenges associated with POOCs.

Keywords: Personalized Open Online Courses · Digital Skills · Competency based

1 Introduction

Since their introduction in 2008 by George Siemens and Stephen Downes, Massive Open Online Courses (MOOCs) have significantly impacted the educational landscape. [1] They rapidly gained popularity as an effective alternative to traditional classroom-based learning. The New York Times called 2012 the 'Year of the MOOC' because of all the hype around MOOC platforms. [2] However, challenges related to learner diversity and individual needs began to surface as their use grew. This led to the development of Competency-based and Personalized Open Online Courses (POOCs). This paper will explore the benefits and challenges of competency-based and individualized open online courses, focusing on the DigiFit4All[1] project, which uses the *GECKO*[2] and *KAUA*[3] platforms developed at the University of Klagenfurt to personalize the courses.

---

[1] https://www.digifit4all.at/

[2] https://gecko.aau.at

[3] https://kaua.aau.at

1.1 The DigiFit4All Project

The DigiFit4All project started in 2020 as a collaboration between the University of Klagenfurt, Vienna University of Technology[4], Johannes Kepler University Linz[5], and Danube University Krems[6]. The project aims to provide competency-based and personalized open online courses for Austrian universities' pupils, students, teachers, and administrative staff. The focus also includes providing materials related to digital skills and information technology tailored to the individual needs of each participant. [3, 4]

1.2 Platforms Used in the Project

In DigiFit4All, several platforms are used to enable the creation of POOCs. The GECKO (Graph-based Environment for Competency and Knowledge-Item Organization) platform was developed to collect and analyze curricula, educational standards, and competency models [5]. It has been adapted for the DigiFit4All project and extended to include several functions like connecting learning objects and competencies or creating courses based on selected competencies. GECKO enables the collection, management, and creation of competency models and the calculation of learning paths within these models. These learning paths serve as the foundation for generating personalized online courses. KAUA (Košice and Alpen-Adria University Assessment) is an online platform for anonymized long-term surveys [6]. In DigiFit4All it is used for pre- and post-tests for participants to enable individualization and assessment. Finally, POOCs created in GECKO are exported to the LMS Moodle, where learners can complete the course.

1.3 Workflows for Teachers and Learners

Teachers log in to set up or modify a course in the GECKO platform. They pick the skills they want pupils or students to learn, choose any needed prerequisites, and select teaching materials. They can also create exams and add their assessments. Finally, they import everything into the course. The workflow for learners is to log in through their institution's learning management system (in case of the project Moodle), and start with a pre-test to create a personalized learning profile. This profile customizes the course content to focus on areas needing more help. After studying the materials, students take a post-test to check their progress, which does not impact their final grade. [3]

2 Advantages

Competency-based and personalized open online courses offer many benefits. Personalized Open Online Courses (POOCs) are an extension of traditional

---

[4] https://www.tuwien.at/

[5] https://www.jku.at/

[6] https://www.donau-uni.ac.at/de.html

MOOCs, offering flexible, globally accessible learning tailored to individual students' unique needs, paces, and styles. [7] These courses use adaptive learning paths to help students focus on their strengths and address weaknesses, enhancing motivation and engagement by making learning relevant to their interests and goals. POOCs break down geographical and time barriers, providing flexible access to education from anywhere. They can serve many students simultaneously without compromising quality, incorporating instant feedback to improve learning effectiveness. By reducing the need for physical infrastructure, POOCs lower education costs and are especially beneficial for non-traditional students, such as working professionals, parents, and those with learning disabilities. [8] For teachers POOCs and primarily DigiFit4All provide additional advantages. They can use the materials for their students and take on the role of coaches. This gives them the time to support students with problems and to accompany them more intensively in the learning process. Interviews with participating teachers show that selecting competencies to create a course is very useful.

3 Challenges

Competency-based and personalized open online courses face several challenges. Customizing learning paths requires complex and expensive technology. Accessibility can be an issue, particularly for those with poor internet access. Students need technical proficiency and reliable internet access, which can disadvantage those without these resources. Additionally, the limited interaction and support in POOCs may make learning more difficult, as these courses often lack the face-to-face engagement and networking opportunities found in traditional programs. The self-paced nature of POOCs requires high levels of self-motivation and discipline from students, as the lack of structure can reduce engagement. [8] Two significant challenges were examined and dealt with through master's theses.

3.1 Security in a Web-based System

Reliable and scalable technology is essential for a good user experience, and data privacy is crucial given the handling of sensitive information. The main actions to ensure the security of the DigiFit4All project are based on given standards for secure software development. In the first step, 34 actions were identified and prioritized, including password confirmation and reset via e-mail, automatic account logout, or password strength display. [7]

3.2 Quality Assurance of Course Material

Quality assurance is needed to ensure learning goals are clearly defined and effectively taught. By selecting learning goals from the collection in GECKO, users or course developers don't need to determine their own goals. The material created during the project is evaluated according to a specially defined quality model, including 17 prioritized criteria grouped into content, didactic design, accessibility, and usability. Experts used these criteria in several iterations to show which aspects of the created material had to be improved or revised. It showed that the DigiFit4All material has its strength in content and accessibility, whereas the didactic design needed some adjustment. [9]

4 Conclusion

The DigiFit4All project aims to deliver competency-based and personalized open online courses (POOCs) tailored to the needs of learners. The GECKO platform, central to this project, helps create and manage personalized learning paths by allowing educators to design and adjust courses based on specific competencies. POOCs extend the benefits of traditional MOOCs by offering flexible, globally accessible learning that adapts to individual needs, breaks down geographical barriers, and lowers costs. However, challenges include the complexity and cost of technology, ensuring quality, maintaining student motivation, and addressing accessibility and technical issues.

Acknowledgments. This project was funded by the *Bundesministerium für Bildung, Wissenschaft und Forschung Austria (BMBWF, Federal Ministry of Education, Science and Research)*, for which we would like to take this opportunity to thank them.

References

1. Downes, S.: Connectivism and Connective Knowledge. EdTech Books (2012)
2. New York Times. The Year of the MOOC. [Accessed: 2024-08-07], https://www.nytimes.com/2012/11/04/education/edlife/massive-open-online-courses-are-multiplying-at-a-rapid-pace.html.
3. Pasterk, S., Pagitz, L., Weiß, A., Bollin, A.: DigiFit4All – Conceptualisation of a Platform to Generate Personalised Open Online Courses (POOCs). In: IFIP Advances in Information and Communication Technology, Springer International Publishing (2022), 247–258.
4. Pasterk S., Lobnig N. A., Kuka L., Rottenhofer M., Oppl S., Futschek G., Hörmanseder M., Tiefenthaller F., Bollin A.: DigiFit4All – Kompetenzen als Basis für Digitalisierung in der Lehre. In: Zeitschrift für Hochschulentwicklung.: Sonderheft Hochschullehre (April 2023) Graz: Verein Forum Neue Medien in der Lehre Austria, 2023, S. 137 – 15
5. Pasterk, S.: Competency-Based Informatics Education in Primary and Lower Secondary Schools. Dissertation at University of Klagenfurt, 2020.
6. Bollin, A. and Kesselbacher, M. and Mößlacher, C.: Ready for Computing Science? A Closer Look at Personality, Interests and Self-concept of Girls and Boys at Secondary Level. In: Informatics in Schools. Engaging Learners in Computational Thinking, Springer International Publishing, 2020.
7. Mödritscher, P.: Systemsicherheits-Optimierungen in Web-basierten Systemen am Beispielprojekt DigiFit4All. Masterthesis, University of Klagenfurt, 2021.
8. Mehal, S.: How online competency-based education can enable greater access to higher education. In: The Journal of Competency-Based Education, Wiley (2019), doi: 10.1002/cbe2.1201
9. Brugger, K.: ALOHA - Ein Qualitätsmodell zur Evaluation von Lernobjekten im Projekt DigiFit4All. Masterthesis, University of Klagenfurt, 2022.

DigiTeaMap: A Digital Map to Represent Addressed Competencies in Digital and Computer Science Education in Austria

Stefan Pasterk[1], Corinna Hörmann[2], and Andreas Bollin[1]

[1] University of Klagenfurt, 9020 Klagenfurt, Austria
`{stefan.pasterk, andreas.bollin}@aau.at`
https://www.aau.at/
[2] Johannes Kepler University Linz, 4040 Linz, Austria
`corinna.hörmann@jku.at`
https://www.jku.at/

Abstract. The relevance of digital and IT education in today's working world is increasing noticeably, and numerous associated competencies are required. For this reason, subjects, courses, or training programs that teach digital and IT skills are currently of great interest to everyone involved. As topics are dealt with differently due to various disciplines and locations, Austria has no uniform or coordinated formats. This makes it challenging to obtain an overview of the content taught or competencies achieved. A real-time indicator is urgently needed for political reasons, among others. The objectives of the project *Digital Teaching Map (DigiTeaMap)* are the collection of data on taught content about digital and IT competencies via an interface developed for this purpose and a clear presentation of this data, which enables the comparison at the level of individual universities and regions, as well as at the national level. This poster describes the idea and background of the project and first experiences with the platform.

Keywords: Digital Education · National Curricula · Data Representation

1 Introduction

Universities and other educational institutions currently find themselves in a situation where the overview of courses on offer and the boundaries between subject areas are unclear. For example, similar content in digital and computer science education is addressed independently and not in a coordinated manner in various curricula. On the one hand, this is due to the confusion of terms such as 'digital skills,' 'media skills', or 'IT skills' [1, 2]. There is a clear overlap, meaning that skills can be located in several subject areas. On the other hand, various curricula and study plans also follow different focal points and, for these reasons, sometimes cover different competencies [3]. While this is undoubtedly in the interests of education and the education system, it tends to hinder a clear view of the range of courses offered locally, like for example in Austria, or even internationally. The points mentioned concern the higher education sector and are also an issue in the school environment. In computer science alone, the example of Germany becomes complex regarding an overview of this subject in the various federal states. For this reason, the German Informatics Society (GI) has published the *Informatics Monitor*[3]. This clearly shows, in the form of a map and using color coding, which federal state is introducing computer science and when and in what form. However, the data is based on static data sets. The objectives of the project *Digital Teaching Map (DigiTeaMap)* are (a) to ensure the collection of data on digital skills in education via a low-threshold interface, (b) to create a clear presentation of the content taught, and (c) to enable the possibility of a comparison - also graphically - at the level of individual universities, regions, but also national level. To achieve these goals, the *DigiTeaMap* platform is being developed, which is based on an existing system for recording competencies (GECKO) and offers an interface for recording, displaying, and comparing taught competencies suitable for each user group.

2 Background and Details of the Project

2.1 Goals and Benefits

The project is relevant from several dimensions arising from the different use perspectives. On the one hand, it can serve as a tool to clearly communicate the current state of IT and digital education to education policy-makers and thus accelerate decision-making. For teachers at universities and in the general education sector, the project supports the planning of content and offers. Furthermore, the search for synergies, coordination between teachers, and cooperation between them will be promoted. From a scientific perspective, the collected data provides a basis for further research into digitalization's socially relevant components and sustainability. In addition, the project results will provide learners and companies with a basis for decision-making. Students can understand which universities focus on their areas of interest and thus receive help in their selection.

2.2 Related Work

The project was inspired by the *Informatics Monitor* but differs from existing work in several respects. For example, no national skills database currently records the digital and IT services offered by educational institutions, initiatives, and associations. This would provide a basis for further research and additional information gathering. The very diverse picture of education in Austria is broken down into a straightforward form of presentation (zoomable map). Although such a representation has already proven worth in similar contexts (e.g., Informatics Monitor), it is not currently used in Austria. The representation as a map and comparison platform is not static but offers a 'real-time status' similar to Google Maps. In comparison, the Informatics monitor is based on static data.

---

[3] https://informatik-monitor.de/

2.3 Competency-based Education

Another current topic in the educational landscape is competency orientation. The teaching of competencies is assumed, which means, among other things, it is stated what learners should be able to do after individual units or entire courses. This is becoming the standard in Austria, primarily in the school context but also increasingly in higher education [4]. The GECKO (Graph-based Environment for Competency and Knowledge-Item Organization) environment from the Institute for Informatics Didactics at the University of Klagenfurt enables the collection, management, and analysis of competencies and their dependencies. The DigiTeaMap uses GECKO as a basis for the data collection.

3 First Results

3.1 Collecting the Data

An interface was developed and integrated into the GECKO platform to record and integrate the teaching sequences. It uses a split screen to display the sequence on the one hand and the pool of selectable competencies on the other hand. Teachers can search for suitable competencies and drag and drop them into their sequence, arranging them in the order in which they teach them. If a competency can not be found, a new one can be generated and stored. Competencies can be grouped into topics, and the time used to teach these topics can be added.

3.2 Displaying Collected Data

The collected data is displayed in a filterable map of Austria, which can be viewed at the state and district levels. In the GECKO environment, competencies are stored with attributes, which are used as options to filter the interactive map. A color code similar to a heat map is used to reflect the different frequencies of occurrence of the individual attributes. For example, one main filter option is the category. Each competence is assigned to one of the following five areas: *algorithms and programming*, *data representation*, *digital applications*, *digital device and infrastructure*, and *humans and computers*. Users can now filter on the map how often competencies belonging to one of the five categories occur in learning sequences in the Austrian area: a red state or district indicates a high occurrence. In contrast, a yellow state indicates rare references to competencies in a particular area. This can be seen in Fig. 1 and in Fig. 2.

3.3 User Experience

The data was collected using existing lesson plans from teachers in different states in Austria. Since lesson plans in Austria must include competency descriptions, these could easily be converted into learning sequences and incorporated into the platform. When writing this article, 33 learning sequences have already been collected and are shown on the map. Participating teachers liked

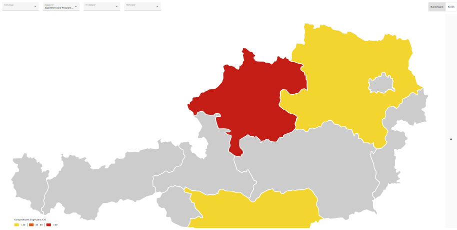


Fig. 1. Interactive map at state level

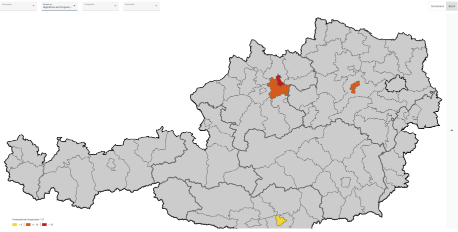


Fig. 2. Interactive map at district level

comparing their plans with others, and the representation form was also mentioned as enjoyable. The feedback from testing the platform will be considered for subsequent releases.

4 Conclusion and Future Work

In recent years, many different offers concerning digital education have come up. It is becoming increasingly difficult to maintain an overview and compare content. The project DigiTeaMap aims to develop a fast and easy way to collect necessary data and to represent the results in a valuable and representative way. Based on competencies included in learning sequences, different data can be shown on an interactive map of Austria. First tests and data collections indicate that the format is well-chosen and valuable for users. In the project's next steps, further data will be collected to fill the map with information. Further steps will include an expansion to an international level.

Acknowledgments. This project was funded by the *Forum Neue Medien Austria (FNMA, Forum New Media Austria)*, for which we would like to take this opportunity to thank them.

References

1. Kesselbacher, M.: Die Vermittlung Informatischen Denkens in der Allgemeinbildung am Beispiel Österreich. In: Klagenfurter Interdisziplinäres Kolleg 2020, Herausgeber: Andreas Bollin, Alexander Brauneis, Werner Drobesch, Horst Peter Groß, Andreas Hudelist. Editor: Werner Drobesch, Horst Peter Groß, Klagenfurt.
2. Kesselbacher, M., and Oberdorfer, G.: Brennpunkte in der Vermittlung von Medienkompetenz und Informatischem Denken. In: Klagenfurter Interdisziplinäres Kolleg 2020, Herausgeber: Andreas Bollin, Alexander Brauneis, Werner Drobesch, Horst Peter Groß, Andreas Hudelist. Editor: Werner Drobesch, Horst Peter Groß, Klagenfurt.
3. Pasterk, S., and Bollin, A.: Digital literacy or computer science: Where do information technology related primary education models focus on? 15th International Conference on Emerging eLearning Technologies and Applications (ICETA), 2017
4. Kern, A.: Weiterentwicklung der Lehrpläne der Primar- und Sekundarstufe in Österreich. Medienimpulse, Jg. 58 / 1, 2020

Exploring Physical Computing In Schools: Designing a Longitudinal Study

Sue Sentance[0000−0002−0259−7408] and Jessie Durk[0000−0002−4725−8215]

Raspberry Pi Computing Education Research Centre,
University of Cambridge, Cambridge, UK
`ss2600,jd871@cam.ac.uk`

Abstract. Physical computing, using programmable devices to learn computing concepts and skills, is becoming increasingly widespread in schools, with studies reporting that children find it engaging and creative. However, there is still a need for research to investigate the impact of physical computing longitudinally. The study described in this short paper uses a longitudinal concurrent mixed methods design to answer questions about young people's creativity, self-efficacy, and agency over a period of five years, while also drawing on parents' and teachers' contributions. Here we describe the motivation for this study, the overall design, and the progress to date. We anticipate that this study will make a significant contribution to approaches to teaching computing in school.

Keywords: physical computing · computing education · K-12 education · mixed-methods research

1 Introduction and Background

Physical computing involves combining software and hardware to build concrete and tangible physical systems that sense and respond to the real world [6]; it is becoming more prevalent in schools with the development of new low-cost devices [3]. Previous research in physical computing has shown that young people find it motivating [6]; more research is needed to examine whether these early experiences impact confidence and creativity in future years. This short paper describes the design of a five-year longitudinal project to investigate the experience of young people who have engaged with physical computing over time. The project will examine changes in attitudes, creativity and agency at key points as the young people progress from primary through to secondary education in the UK, and will also explore the perceptions and attitudes of both parents and educators involved with the young people over a similar timeframe.

Recent studies have investigated the impact of physical computing on young people, providing some evidence that physical computing develops creativity [7, 8], generates interest in computing [4], develops intrinsic motivation [6], and that its tangibility supports learning [8]. Other research in physical computing suggests that it makes computer science concepts and skills accessible to a broader population [3], and some evidence has started to indicate that physical computing may help to address gender imbalance in the subject [7, 9]. To address the need to ascertain the longer-term impact of working with programmable devices, this new project, Exploring Physical Computing in School (EPICS), focuses on providing a mix of qualitative and quantitative evidence over a number of years, giving robust empirical evidence as to whether physical computing provides a valuable context in which to learn the wide-ranging aspects of computing and technology and can prepare young people for an increasingly technologically complex world. Three research questions motivate the EPICS research design:

RQ1 How does engagement with physical computing support the development over time of young people's creativity, technological self-efficacy and socio-technological agency?

RQ2 Are there gender-related differences in the way that children engage with physical computing, and if so, what are the reasons for these?

RQ3 What is the role of teachers and parents with respect to the development of children's digital capital through physical computing experiences?

2 The Study

2.1 The BBC micro:bit

An example of a physical computing device is the BBC micro:bit [1]. Over 6.5 million micro:bits have been manufactured and sold since 2015 [2]. In a recent initiative, all primary schools in the UK who requested them have received a class set of 30 micro:bits [2]. The fact that thousands of primary schools are now able to access physical computing has enabled us to focus on the BBC micro:bit as the means through which the pupils in our study engage with physical computing. However, the study is not intended to be device-dependent, and the research design will focus on aspects that relate to physical computing in general.

2.2 EPICS Research design

Within the project, there are two streams of activity:

- a qualitative study following a cohort of pupils, alongside their parents and teachers, at schools across the UK, and
- a quantitative study with non-matched cohorts of pupils and teachers.

For the qualitative aspect of the study, the schools within the project cohort have all received a class set of micro:bits and committed to delivering a unit of work around physical computing in each of the three years, until the pupils leave primary school. For the final two years of the project, we will continue to follow the pupils at the secondary schools they move on to (which is as yet unknown), and monitor any future engagement with physical computing. Interviews with teachers and parents will also form part of the study.

The purpose of the quantitative strand of the mixed-methods design is to gather data from a larger number of pupils, and relate this to engagement with physical computing. The project will be completed in the early months of 2029.

Figure 1 provides a diagrammatic view of the fully longitudinal concurrent mixed methods design [5] adopted. This research design approach involves both quantitative and qualitative data being collected at each (or almost each) time point, and then merged to create a set of data that is analysed together to answer the research questions [5]. The design encompasses four time points in project years 1,2,3 and 5, with qualitative data collected at each time point, and quantitative data in alternate years 1,3 and 5.

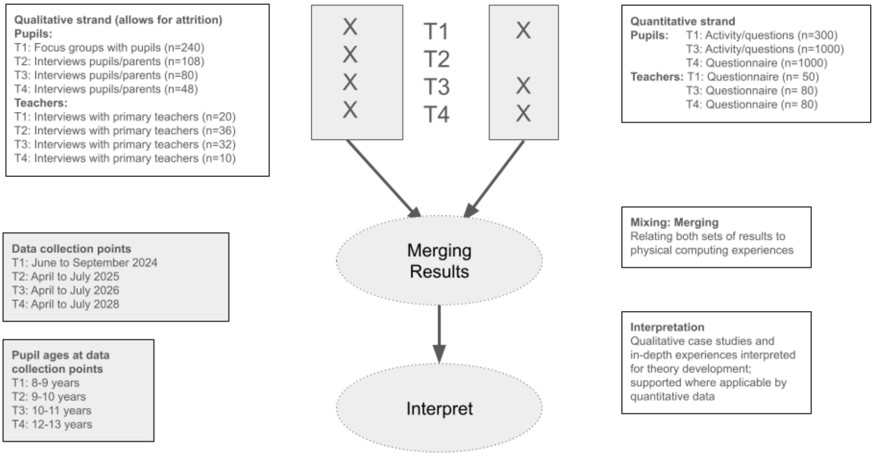


Fig. 1. Longitudinal concurrent mixed-methods research design (Diagrammatic approach is drawn from [5])

2.3 Initial data collection

In the first stage of data collection (T1 Qualitative), 19 schools have been recruited to the study. In developing the research instruments, we used the foci within the research questions – self-efficacy, creativity, socio-technological agency and digital capital – to create interview and focus group schedules. At the time of writing, 43 focus groups (involving 232 pupils) and 22 teacher interviews have been held across the cohort of schools. Appropriate ethical procedures were followed, and consent received from parents, teachers and pupils involved. The next stage of data collection is the quantitative data from the first year for which the target is 300 pupils; this is lower than for subsequent years as due to the children's young age an activity will be used in preference to a questionnaire.

3 Summary and next steps

We have outlined the design of a longitudinal mixed-methods research study focusing on the impact of physical computing on primary school children over a period of five years, involving pupils, parents and teachers. The next steps in the project are to complete the data collection and analysis for the first year, and to report baseline findings from pupils, now aged 8-9 years old, and their teachers. We look forward to sharing more details of this project as it unfolds.

Acknowledgments. We are grateful to the Micro:bit Educational Foundation, BBC and Nominet for their support for this project.

References

1. Austin, J., Baker, H., Ball, T., Devine, J., Finney, J., De Halleux, P., Hodges, S., Moskal, M., Stockdale, G.: The BBC micro:bit: from the UK to the world. Communications of the ACM **63**(3), 62–69 (2020)
2. BBC: Everything you need to know about the BBC micro:bit. News item available at https://www.bbc.co.uk/teach/microbit/articles/zfjg8p3 (2023)
3. Hodges, S., Sentance, S., Finney, J., Ball, T.: Physical Computing: A Key Element of Modern Computer Science Education. Computer **53**(4), 20–30 (April 2020). https://doi.org/10.1109/MC.2019.2935058
4. Jayathirtha, G., Kafai, Y.B.: Interactive Stitch Sampler: A Synthesis of a Decade of Research on Using Electronic Textiles to Answer the Who, Where, How, and What for K–12 Computer Science Education. ACM Trans. Comput. Educ. **20**(4) (Oct 2020). https://doi.org/10.1145/3418299
5. Plano Clark, V.L., Anderson, N., Wertz, J.A., Zhou, Y., Schumacher, K., Miaskowski, C.: Conceptualizing longitudinal mixed methods designs: A methodological review of health sciences research. Journal of Mixed Methods Research **9**(4), 297–319 (2015)
6. Przybylla, M., Romeike, R.: Impact of Physical Computing on Learner Motivation. In: Proceedings of the 18th Koli Calling International Conference on Computing Education Research. pp. 1–10. Koli Calling '18, Association for Computing Machinery (2022). https://doi.org/10.1145/3279720.3279730
7. Sentance, S., Schwiderski-Grosche, S.: Challenge and creativity: using .NET gadgeteer in schools. In: Proceedings of the 7th Workshop in Primary and Secondary Computing Education. pp. 90–100. ACM (2012), http://dl.acm.org/citation.cfm?id=2481473
8. Sentance, S., Waite, J., Hodges, S., MacLeod, E., Yeomans, L.: "Creating Cool Stuff": Pupils' Experience of the BBC Micro:Bit. In: Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education. pp. 531–536. SIGCSE '17, ACM, New York, NY, USA (2017). https://doi.org/10.1145/3017680.3017749
9. Stupurienė, G., Juškevičienė, A., Jevsikova, T., Dagienė, V., Meškauskienė, A.: Girls' Summer School for Physical Computing: Methodology and Acceptance Issues. In: Informatics in Schools. Rethinking Computing Education: 14th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, ISSEP 2021, Virtual Event, November 3–5, 2021, Proceedings 14. pp. 95–108. Springer (2021). https://doi.org/10.1007/978-3-030-90228-5_8

Knowledge of Cookies and Personalized Ads among Lower Secondary Students: Effects of a Simple Treatment

Cyril Brom[1][0000−0001−5945−0514], Anna Yaghobová[1][0000−0002−2102−7515], Anna Drobná[1,2][0000−0003−2714−9529] , Daniel Šťastný[1][0009-0006-5085-2975] and Marek Urban[3][0000−0003−2772−1388]

[1] Faculty of Mathematics and Physics, Charles University, Prague, Czechia
[2] Faculty of Education, Charles University, Prague, Czechia
[3] Institute of Psychology, The Czech Academy of Sciences

Abstract. Little is known about lower secondary school students' understanding of cookies and their awareness of personalized advertising, and even less is known about how much this knowledge can be improved. This experimental study addresses this gap with Grade 6 ($N = 44$) and Grade 8 ($N = 35$) students. We assessed their knowledge through interviews and provided a brief, one-on-one intervention to half the participants. A follow-up interview five months later evaluated the long-term impact. Initial understanding of students was mixed, with some inaccurately believing cookies protect against online threats. Eighth-graders had better knowledge than sixth graders. The experimental group exhibited strong gain in understanding cookies ($\eta^2_p = 0.16$), but not personalized ads. This study suggests that children often have an experience-based initial understanding of digital concepts, but a simple intervention can enhance it.

Keywords: Computer science, Education, Lower secondary school, Cookies, Personalized advertisement, Conceptions, Learning.

1 Introduction

Children develop initial conceptions, also called pre-conceptions, about the world around them, including digital technologies. Understanding these pre-conceptions is crucial before developing new educational materials on respective educational targets [2]. Recent studies have mapped various child pre-conceptions about digital technologies (e.g., [1–6]), indicating an intuitive and fragmented understanding often rooted in everyday experience. Conceptual understanding of "invisible" principles behind observable aspects of these technologies is rare, even among adolescents (e.g., [1]). Additionally, evidence-based materials promoting such understanding are lacking.

Cookies are a cornerstone concept in the networking and cybersecurity strands of new K-9 computing curricula (e.g., [7]), but children's understanding of cookies and how to promote it remains underexplored. Hug [3] only briefly mentioned that schoolchildren may have incomplete knowledge about them, a concern that also applies to adolescents aged 16–18 [9].

Hence, this study examines understanding of cookies and personalized ads (a vivid example of cookie usage) among Grades 6 and 8 students. We chose this age group because most adolescents already have long experience with digital devices [8], likely having encountered cookies and ads while browsing the web and, as constructivists frameworks imply (e.g., [2]), they should be aware of, at least, cookies' existence. However, awareness does not necessarily imply understanding. Hence, we explore whether and how much we can boost children's comprehension of these concepts.

2 Method

This study was part of a larger mixed-methods project with children involving interviews about internet principles, including cookies and personalized ads. Half received an online one-on-one tutoring session on these topics, followed by another interview five months later. Here, we present only findings about cookies and personalized ads.

Participants included 44 sixth-graders (age ~12y, 19 girls) and 35 eighth-graders (age ~14y, 15 girls) with limited prior exposure to computer science topics in school. They were recruited from various regions of Czechia via Facebook and a teacher network, and compensated with a table games worth ~50 EUR. They were randomly assigned to experimental ($n_{Gr.6/8}$ = 26/17) and control groups ($n_{Gr.6/8}$ = 24/20).

Each child participated in three 45–60-minute online sessions via Zoom: an initial interview (April–June 2022), a week-later tutoring session (experimental) or unrelated, no-instruction activities (control), and a final interview (autumn 2022). Pre- and post-interviews were semi-structured, with set questions and follow-ups based on responses. Key interview questions relevant in this paper focused on recognition and understanding of cookies and personalized ads (e.g., *Have you even heard the word 'cookie'? What does it mean to you?* etc.).

The teaching session (experimental group only) covered internet principles, including a 5–10-minute segment on personalized ads and cookies, using graphics, explanations, discussions, and activities. Key points from this segment included information we deemed relevant for and understandable by adolescents that: a) ads do not appear randomly on devices, but their content depends on our internet activities; b) what we do on the internet is stored on servers (the concept of servers was also explained); c) cookies are files allowing servers to collect information about our online behavior; d) cookies are associated with both benefits and risks (including examples of each).

Pre- and post-interviews were transcribed and analyzed using inductive (bottom-up) thematic analysis in Atlas.ti 22 by four coders and an auditor.

3 Results

Codes. The thorough inductive analysis yielded not more than three codes on cookies: 1) *awareness* without deeper understanding; 2) *partial or correct* understanding; and 3) a misconception that *cookies serve as protection*. Examples include:

1. Awareness without understanding: “You have to confirm it on the internet, that you agree with it.” (Grade 6)
2. Partial or correct understanding: “[Cookies] are small documents that save information eeh about you basically and help … show relevant ads.” (Grade 8)
3. Misconception about protection: “[Cookies] are protection against hackers [and viruses]. … it appears on a webpage when opening it. And there is an option to agree with all the protections by cookies…” (Grade 6)

Regarding personalized ads, participants either were or were not aware of them; more nuanced pre-conceptions were not found.

Pre-conceptions. In the initial interview, 22 (50%) sixth-graders were aware of cookies, 14 (32%) had a partially correct or correct understanding, and 8 (18%) never noticed cookies. Almost all eighth-graders were either aware of cookies ($n = 15$, 43%) or had a partially correct or correct understanding ($n = 19$, 54%). Better knowledge of cookies was strongly associated with awareness of personalized ads ($r_s = .49$, $p < .001$). The misconception that cookies serve as protection was noted seven times among sixth-graders and only once among eighth-graders. Additionally, 18 (41%) sixth-graders were aware of personalized ads, compared to 30 (86%) eighth-graders.

Learning. For pre-post analysis on cookies, 2 points were assigned for correct understanding, 1 point for awareness, and 0 points for neither. The intervention significantly improved knowledge about cookies ($F(1, 75) = 14.73$, $p < .001$, $\eta^2_p = .16$) for both age cohorts ($F(1, 75) = 0.11$, $p = .739$, $\eta^2_p = .00$) (Figure 1). Knowledge of personalized ads improved between pre-test and post-test ($F(1, 83) = 11.74$, $p < .001$, $\eta^2_p = .12$), but the change was similar in both experimental and control groups ($F(1, 75) = 0.04$, $p = 0.835$, $\eta^2_p = .00$). Post-learning, 17 (65%) sixth-graders from the experimental and 14 (58%) from the control condition were aware of personalized ads, whereas 17 (100%) eighth-graders from the experimental and 18 (90%) from the control condition were aware.

4 Discussion and Conclusion

We demonstrated that some sixth-graders were unaware of cookies or misunderstood their purpose, despite encountering frequent prompts to accept cookies on their personal devices. Eighth-graders generally had better, but still limited, knowledge (despite years of exposure). Older students also showed greater awareness of personalized ads. Altogether, exposure increases awareness, but not necessarily understanding. This corroborates previous pre-conception studies (e.g., [1–6]).

A brief educational intervention significantly improved students’ understanding of cookies, with effects lasting nearly six months. This suggests that enhancing conceptual knowledge of digital world elements can be achieved in schools with relatively small effort, particularly for understanding cookies. However, the intervention did not improve knowledge of personalized ads compared to the control group.

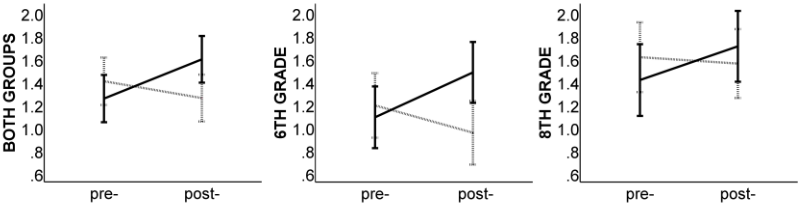


Fig. 1. Points for comprehension of cookies by pre- and post-test and experimental (full line) vs. control group (dashed line). Left: both age cohorts. Middle: Grade 6. Right: Grade 8. Scale 0 – 2.

Despite a small sample size and the tutoring nature of the intervention, the study's experimental design is a key strength. Future research should explore effects of teaching sessions about other non-programming computer science concepts, as they complement computational thinking strands of new computing curricula.

Acknowledgments. Main funding: GAČR 22-20771S. AD: Student grant GAUK 484722. AY: Student grant GAUK 360322. MU: RVO 68081740.

References

1. Brom, C., Yaghobová, A., Drobná, A., Urban, M.: 'The internet is in the satellites!': A systematic review of 3–15-year-olds' conceptions about the internet. Education and Information Technologies **28**(11), 1–30 (2023)
2. Diethelm, I., Hubwieser, P., Klaus, R.: Students, teachers and phenomena: Educational reconstruction for computer science education. In: *Proceedings of the 12th Koli Calling International Conference on Computing Education Research*. ACM, 164-17. (2012)
3. Hug, A.: "I've got nothing to hide!" survey on data privacy competence with German schoolchildren. In: *Proceedings of the 13th Workshop in Primary and Secondary Computing Education*. ACM (2018)
4. Livingstone, S., Stoilova, M., Nandagiri, R.: Children's data and privacy online: growing up in a digital age: an evidence review. London School of Economics and Political Science, Department of Media and Communications, London, UK. (2019)
5. Mertala, P., Fagerlund, J., Calderon, O.: Finnish 5th and 6th grade students' pre-instructional conceptions of artificial intelligence (AI) and their implications for AI literacy education. Computers and Education: Artificial Intelligence **3**, 100095 (2022)
6. Rücker, M. T., Pinkwart, N.: Review and discussion of children's conceptions of computers. Journal of Science Education and Technology **25**, 274-283 (2016)
7. Computer Science Teachers Association: CSTA K-12 Computer Science Standards, Revised 2017. Retrieved from https://csteachers.org/k12standards/ (acc. 2024-06-13) (2017)
8. Šmahel, D., Macháčková, H., Mascheroni, G., et al.: EU Kids Online 2020: Survey results from 19 countries. EU Kids Online. http://hdl.handle.net/20.500.12162/529 (2020)
9. Zarouali, B., Ponnet, K., Walrave, M., Poels, K.: "Do you like cookies?" Adolescents' skeptical processing of retargeted Facebook-ads and the moderating role of privacy concern and a textual debriefing. Computers in Human Behavior **69**, 157-165 (2017)

Novice Primary School Teachers’ Conceptions of Internet Structure: A Qualitative Analysis

Anna Drobná[1,2][0000-0003-2714-9529], Anna Yaghobová[1][0000-0002-2102-7515], David Šosvald[1][0009-0005-5848-6028], Daniel Šťastný[3][0009-0006-5085-2975], Marek Urban[4][0000-0003-2772-1388] and Cyril Brom[1][0000-0001-5945-0514]

[1] Faculty of Mathematics and Physics, Charles University, Prague, Czechia
[2] Faculty of Education, Charles University, Prague, Czechia
[3] Faculty of Arts, Charles University, Prague, Czechia
[4] Institute of Psychology, The Czech Academy of Sciences

Abstract. This study investigates primary school teachers' conceptions of internet architecture, a largely underexplored area in early computer science education. We analyzed the conceptions of 59 pre-service and novice in-service primary school teachers from the Czech Republic, who are expected to teach the functioning of the internet in their future practice. Participants drew and described their understanding of internet architecture during semi-structured online interviews. Thematic analysis revealed conceptions similar to the conceptions of children, or ones that could hinder the teachers' later understanding of how the internet functions. These findings contribute to the development of educational materials and the preparation of novice primary school teachers.

Keywords: Computer science, Education, Primary school, Pre-service Teachers, Conceptions.

1 Introduction

Computer science education is being introduced or revised at the primary level in many countries, including the Czech Republic [e.g., 4]. Recent trends emphasize the importance of teaching the underlying principles of internet as early as primary school (ISCED 1). For teachers, this presents a new and challenging topic, highlighting the need for effective preparation and the development of appropriate teaching materials. A crucial first step in this preparation is for instructional designers to understand teachers’ conceptions of the subject matter [3].

At the primary school level, understanding the basic principles of internet functionality involves grasping simplified concepts such as data storage on servers, data transmission, and awareness of digital footprints and the decentralized nature of the internet. Teaching these fundamental concepts can significantly enhance primary school students’ understanding of internet safety.

Conceptions of the structure of the Internet have been explored, particularly in studies with children [e.g., 2] but have been less extensively examined among teachers. This project aims to investigate the conceptions of novice primary school teachers regarding the structure and functioning of the internet, with a focus on their graphical representations of internet architecture.

2 Method

Data were collected through semi-structured interviews, a common method in preconception research (e.g., [2]). Participants, 59 pre-service and novice in-service primary school teachers (Table 1) from 10 Czech universities (average age ~24 years, 56 female), were asked to draw their concept of internet architecture and describe their drawings. These drawings are the focus of this analysis.

Recruitment was conducted via Facebook groups for teachers and snowball sampling, with participants receiving approximately 20 EUR. Data collection took place online via Zoom, each session lasting about 60 minutes, following a structured interview protocol. The drawing task was introduced with: *“If we could see the whole internet from a bird’s eye view, how do you think it would look? What would it look like from a technical perspective?”* Follow-up questions were adapted to participants’ drawings (e.g., *“Does it have any parts?”* or “*What did you just draw?”).*

The interviews were transcribed and analysed using inductive thematic analysis [1] with Atlas.ti 22 by two independent coders. This paper focuses on the drawings and associated responses; full interview analysis will be part of a larger study. The research was approved by the university’s ethics committee.

Pre-service or in-service status and teaching experience	*Number of participants*
Pre-service teacher	**38**
No teaching experience (except pre-service training during university studies)	27
Some teaching experience	11
In-service teacher after graduation	**21**
Teaching for less than one year	5
Teaching for one to three years	16

Tab. 1. Characteristics of the participants according to their teacher training and length of teaching experience.

3 Results

The analysis revealed 12 codes representing either the overall structure (e.g., the internet as a spider web) or specific elements within the images (e.g., the presence of a satellite). All identified codes are listed in Table 2, along with the number of occurrences. Typically, more than one code was identified in each participant’s image.

Most participants depicted the internet as a network or connection of two or more devices with an unclear structure and ambiguous components (e.g., Figures 1–3). Some of these representations were described as “spider webs” or “webs of information”.

Another significant group of images depicted the internet as an invisible force or signal “radiating” from a source (Figures 4–6). Participants in this group described the internet as ubiquitous and invisible, originating from a source, which sometimes included multiple sources. In several cases, this source was identified as a satellite (Figure 5).

Codes	*Number of code appearances*
Connecting one user device to another user device	17
“Radiating” a non-specific signal all around with no internal structure	11
Including satellite	10
Including internet center (one to one hundred centers)	7
Including transmitting tower	6
Network with unexplained parts (often spider web)	6
“Earth Globe” covered by some kind of signal	4
The participant refused to draw a picture (only explained verbally).	4
Network with explained parts and their functions (e.g. router, server, user device)	3
Including servers	3
Other extraordinary conceptions	2
Devices without connection	1

Tab. 2. Codes related to participant images depicting the structure of the internet, along with the frequency of each element.

Three participants demonstrated a clear understanding of the internet’s structure, closely aligning with how the internet actually operates (Figures 7 and 8). Six participants provided detailed explanations using technical terms such as “server” and “network router”.

Four participants depicted Earth covered in an unexplained structure, with one attributing this idea to a movie. A total of 10 participants incorporated satellites into their conceptions (e.g., Figure 5, 10), either as a source of internet connection, a transmission medium, or data storage.

Fig. 1–3. Drawings of participants: Unclear structure or ambiguous components.

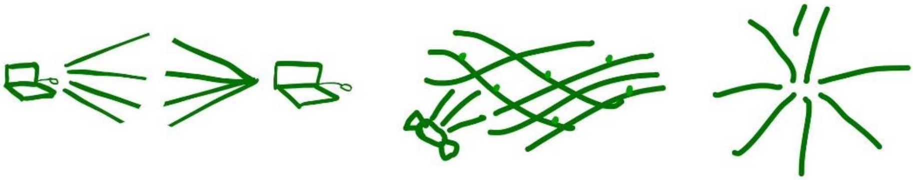
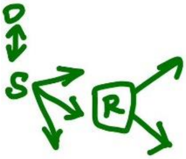
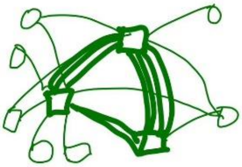


Fig. 4–6. Drawings of participants: Non-specific wave or signal.





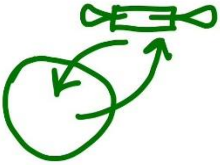


Fig. 7, 8. Drawings of participants: Advanced network. **Fig. 9, 10.** Drawings of participants: "Earth globe".

4 Discussion and Conclusion

This study employed semi-structured interviews and drawing tasks to explore the conceptions of novice in-service and pre-service primary school teachers about internet architecture. Many of the teachers' conceptions resembled preconceptions typically found among children (e.g., the presence of a satellite) [2], though the adult versions tended to be more developed.

For research of this kind, it is important not to rely solely on drawings and brief comments from participants, but to ask additional in-depth questions, which is what we did. These findings hold potential significance for teacher education related to internet functionality.

However, the study has limitations. For example, participants were primarily recruited through social media, and while they were unaware of the specific interview topic, it is likely that they were more engaged teachers. As a next step, it is recommended to develop educational materials to better acquaint teachers with the structure and functioning of the internet.

Acknowledgments. This study was funded by GAUK 484722 and GAČR 22-20771S.

References

1. Braun, V., Clarke, V.: Using thematic analysis in psychology. Qualitative research in psychology 3, 2 (2006), 77–101.https://doi.org/10.1191/1478088706qp063oa. (2006)
2. Brom, C., Yaghobová, A., Drobná, A., Urban, M.: 'The internet is in the satellites!': A systematic review of 3–15-year-olds' conceptions about the internet. Education and Information Technologies 28(11), 1–30. 10.1007/s10639-023-11775-9. (2023)
3. Diethelm, I., Hubwieser, P., Klaus, R.: Students, teachers and phenomena: Educational reconstruction for computer science education. In: Proceedings of the 12th Koli Calling International Conference on Computing Education Research. ACM, 164-17 (2012)
4. Oda, M., Noborimoto, Y., Horita, T.: International Trends in K-12 Computer Science Curricula through Comparative Analysis: Implications for the Primary Curricula. International Journal of Computer Science Education in Schools, 4(4), (2021)

PRIMMDebug: A Tool and Process for Teaching Text-Based Debugging to Beginner Programmers

Laurie Gale[1][0009−0004−4299−6704] and Sue Sentance[1][0000−0002−0259−7408]

Raspberry Pi Computing Education Research Centre, University of Cambridge,
Cambridge, United Kingdom
`{lpg28,ss2600}@cam.ac.uk`

Abstract. Debugging is known to be a difficult process for beginner programmers, especially for those learning their first text-based programming language. This is often reflected in the ineffective debugging strategies employed and the negative emotional reactions that ensue in students. While much debugging-specific research has taken place, educators still report the same student difficulties that were present decades ago. In this poster, we present PRIMMDebug, a work-in-progress process and tool designed to encourage effective debugging strategies.

Keywords: Debugging · Programming education · Teaching tools

1 Introduction and Background

The process of finding and fixing errors in programs is a notably steep learning curve for many beginner programmers. Its challenges have been well-documented over several decades; ultimately, debugging requires sufficient prior knowledge and experience that beginners do not possess. Several positive interventions have been explored to ease the challenges of beginner debugging, such as explicit debugging teaching processes [1], improved programming error messages [2], and debugging-specific tooling [4]. Despite these efforts, numerous recent studies highlight the difficulty of teaching debugging in the classroom [5] and the ineffective methods that some students use to resolve their errors [3].

This poster introduces PRIMMDebug, a work-in-progress approach for explicitly teaching debugging to school students learning text-based programming, with the aim of encouraging effective debugging strategies. We seek to do this through the combination of a process and a tool.

2 PRIMMDebug

PRIMMDebug is designed for school students learning text-based programming. The process is based on PRIMM [6], a model for teaching text-based programming in schools. The core idea of PRIMM is to scaffold the process of independently writing code by gradually transitioning from a foreign program to a program written by the student.

PRIMMDebug also builds on systematic debugging teaching processes, that is, sets of steps used to explicitly model the debugging process to students. The first of these was introduced by Carver and Risinger [1], with more developed in recent years. Although initial results show promise, there is a lack of tooling that concretely models the scaffolding of these processes in their entirety.

2.1 The Process

The PRIMMDebug process is shown in Figure 1 (the italicised text represents the prompts students are given), with its steps described below. It begins with an erroneous program, a description of what that program should do, and potentially some test cases. We initially envision programs to contain just one error due to the focus on the initial experiences of debugging.

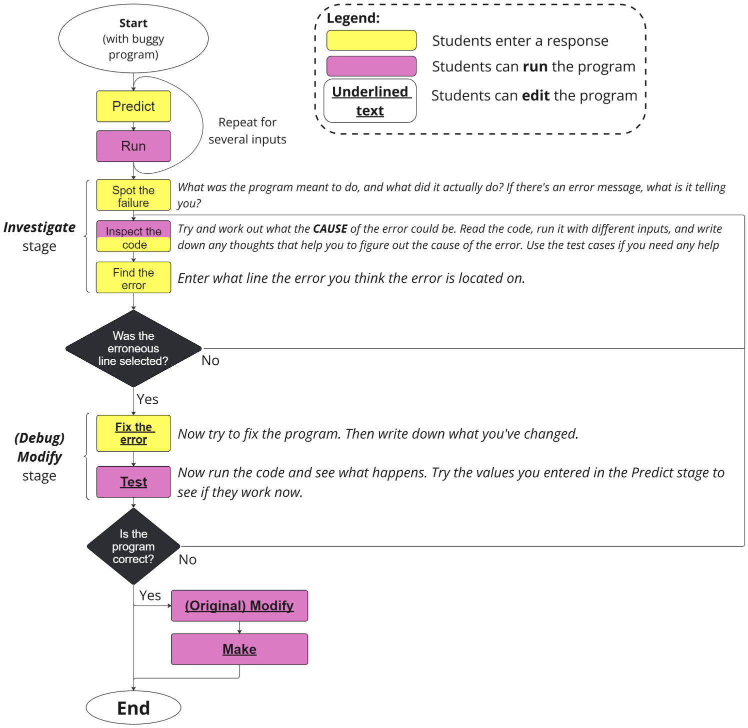


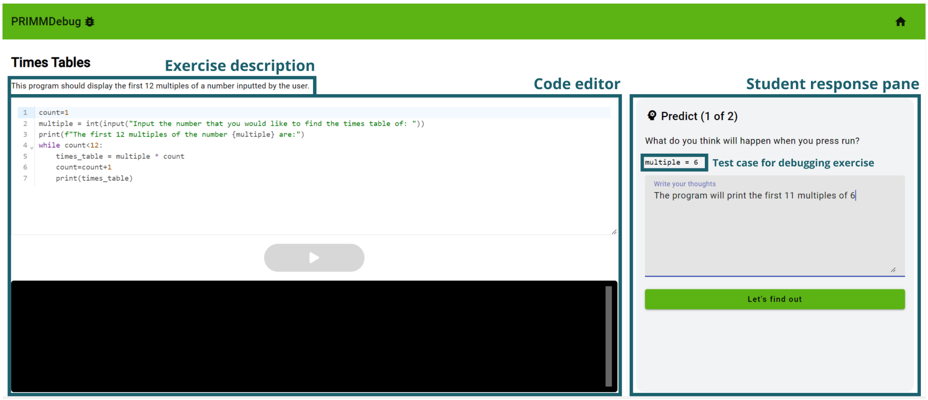
Fig. 1: The PRIMMDebug Process as a Flowchart

1. **Predict**: Students predict the output of the program. If applicable, the prediction is given for a test case (set of variable values).
2. **Run:** Students run the program, potentially for a given test case. In cases where multiple runs are required to ascertain information about the error, multiple predict-run cycles are completed per test case.

3. **Investigate:** We decompose the original investigate stage of PRIMM into the initial stages of debugging.
    (a) **Spot the failure:** Students consider the difference between the intended and actual output of the program.
    (b) **Inspect the code:** Students then formulate hypotheses about the location of the error.
    (c) **Find the error:** Students enter the line that they think contains the error, moving on to the next stage if they are correct.
4. **(Debug) Modify:** Students now attempt to resolve the error, in turn modifying the buggy code. This is a different modify stage to the original PRIMM as there is a definitive correct change that students can make.
    (a) **Fix the error:** Students edit the program to attempt to fix the error. Changes to the program should not need to be too significant.
    (b) **Test:** Students run their program with the previously used test cases and identify whether it is working as intended. If the program still contains errors, they move to stage 3(b) to generate an alternative or more detailed hypothesis. If students' changes are corrective, they may end the exercise or continue with the final two stages of PRIMM.
5. **(Original) Modify/Make**: Students optionally modify the working program to extend its functionality and create a similar instance of the program.

2.2 The Tool

A lightweight tool, currently in the prototype phase, is being developed to enact the PRIMMDebug process. Figure 2 shows an annotated view of the tool in its current state.



Times Tables

This program should display the first 12 multiples of a number inputted by the user.

```
count=1
multiple = int(input("Input the number that you would like to find the times table of: "))
print(f"The first 12 multiples of the number {multiple} are:")
while count<12:
    times_table = multiple * count
    count=count+1
    print(times_table)
```

Predict (1 of 2)

What do you think will happen when you press run?

`multiple = 6`

Write your thoughts: The program will print the first 11 multiples of 6

Fig. 2: The PRIMMDebug User Interface

Ultimately, the tool implements the PRIMMDebug process, inputting students' responses and preventing the running and editing of code at certain stages.

This tool-process combination aims to instil a more self-regulated debugging behaviour than is commonly seen in beginner programmers.

3 Conclusion and Future Work

The need for beginner programmers to develop effective and robust debugging strategies is paramount. Previous approaches have promise but only limited evidence of their efficacy. We present PRIMMDebug, a combination of a scaffolded debugging process and a tool that enforces this process. By introducing the relevant PRIMMDebug steps to students' own programs, we aim to provide a strategy that will improve students' success with debugging.

There is still much work to be done to improve and investigate PRIMMDebug. We first wish to refine the process and tool through feedback from teachers, students, and researchers. We also plan to conduct a study of students' debugging behaviour within PRIMMDebug, provisionally in 2025. Such a process-tool combination study may be of wider use in determining the lasting effect that a process-enforcing tool has on students' problem-solving behaviour.

References

1. Carver, S.M., Risinger, S.C.: Improving children's debugging skills. In: Empirical studies of programmers: second workshop. pp. 147–171. Ablex Publishing Corp., Norwood, New Jersey (12 1987), https://www.researchgate.net/publication/262311647
2. Denny, P., Prather, J., Becker, B.A., Mooney, C., Homer, J., Albrecht, Z., Powell, G.: On designing programming error messages for novices: Readability and its constituent factors. In: Conference on Human Factors in Computing Systems - Proceedings. Association for Computing Machinery (5 2021). https://doi.org/10.1145/3411764.3445696, https://dl.acm.org/doi/10.1145/3411764.3445696
3. Liu, Z., Zhi, R., Hicks, A., Barnes, T.: Understanding problem solving behavior of 6–8 graders in a debugging game. Computer Science Education **27**(1), 1–29 (1 2017). https://doi.org/10.1080/08993408.2017.1308651, https://www.tandfonline.com/action/journalInformation?journalCode=ncse20
4. Luxton-Reilly, A., McMillan, E., Stevenson, E., Tempero, E., Denny, P.: Ladebug: An online tool to help novice programmers improve their debugging skills. In: Annual Conference on Innovation and Technology in Computer Science Education, ITiCSE. pp. 159–164. Association for Computing Machinery, New York, NY, USA (7 2018). https://doi.org/10.1145/3197091.3197098, https://dl.acm.org/doi/10.1145/3197091.3197098
5. Michaeli, T., Romeike, R.: Current Status and Perspectives of Debugging in the K12 Classroom: A Qualitative Study. In: 2019 IEEE Global Engineering Education Conference (EDUCON). pp. 1030–1038. IEEE Computer Society, Washington, DC (5 2019), 10.1109/EDUCON.2019.8725282
6. Sentance, S., Waite, J., Kallia, M.: Teaching computer programming with PRIMM: a sociocultural perspective. Computer Science Education **29**(2-3), 136–176 (7 2019). https://doi.org/10.1080/08993408.2019.1608781, https://www.tandfonline.com/action/journalInformation?journalCode=ncse20

Towards Generating Bebras Tasks Using Large Language Models: What Criteria to Use for Evaluation?

Mohsen Asgari[1] [0009−0003−9558−8790], Filip Strömbäck[0000−0002−0556−6893], and Linda Mannila[0000−0001−6020−2737]

Linköping University, Linköping, Sweden
[1] `mohsen.asgari@liu.se`

Abstract. The use of Large Language Models (LLMs) for creating educational content presents a promising avenue for supporting computational thinking education. The goal of this study is to explore the use of LLMs in evaluating the quality of Bebras tasks based on criteria established in previous research. We propose an automated evaluation pipeline using the open-source library LangChain, where LLMs assess AI generated tasks against pre-defined criteria. The outcome is an evaluation card for each generated task, providing insights into its quality. The proposed methodology can later be used to evaluate AI-generated Bebras tasks.

Keywords: Educational Content Generation · Computational Thinking · Large Language Models · Bebras

1 Problem Statement

Bebras is an international challenge that promotes computing and computational thinking (CT) in K-12 education. Every year, educators collaboratively design and select tasks for the challenge. After years of successful Bebras contests, large language models (LLMs) present new opportunities to apply previous tasks in generative AI for CT education. A key challenge in using LLMs for education is evaluating AI-generated content.

In experiments with OpenAI's GPT-4o, we found that its generated tasks are often unsuitable for direct assessment. While capable of producing tasks with clear educational goals, it struggles to solve them and to generate quality multiple-choice options. This may stem from the model's inability to revise outputs and a tendency to hallucinate [5] when uncertain.

AI-generated Bebras tasks can be used for classroom practice with teacher oversight. Quality measures are essential for handling large volumes of generated tasks that require human refinement. Such evaluations assist in selecting appropriate tasks for local Bebras challenges.

To evaluate AI-generated Bebras tasks, a methodology is needed. This poster outlines a design using criteria for good Bebras tasks proposed in ISSEP 2008 [1], and also [3].

2 Methodology

The aim is to design an automated methodology to evaluate AI-generated Bebras tasks. We propose a pipeline using LangChain which is an open-source framework designed for building applications with large language models, enabling integration with various data sources and interactions with other applications. It provides components (modular abstractions) and chains (customizable pipelines) for use cases like data extraction, document-based question answering, summarization, and structured data analysis [2]. A 2024 study showed that LLMs like GPT-4, when used as judges, can closely align with human preferences, achieving over 80% agreement—comparable to human-to-human agreement [6].

After the generation of a task it gets fed into the LangChain pipeline, and in each stage an LLM will analyse the task based on each item in the criteria set and provide a result for each item. Figure 1 illustrates this process. Finally, each generated task receives an evaluation card created by all LLMs, assessing various criteria items and measuring the quality of the task. The proposed criteria are shown in the first column of Table 1, derived from Dagienė's criteria [1] (normal text) and Vaníček's updates [3] (*italic text*). Additionally, GPT-4o was provided with Bebras tasks, using few-shot learning to complement the criteria, which are shown in **bold text**.

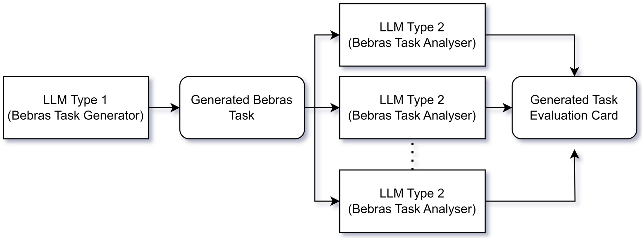


Fig. 1. Overview of the evaluation process for generated Bebras tasks.

3 Preliminary Results

We generated some examples of Bebras tasks using few-shot learning. The generated tasks show educational value though still requiring human modifications. For example on a generated task named "Castle Defense Strategy" [1] despite generating an interesting question, the model was unable to reason and solve it correctly by itself. The same situation occured for the "Mirror Maze"[2] task,

---

[1] https://chatgpt.com/share/c160b6f1-42da-4a96-9fd3-e73fa58d6c51

[2] https://chatgpt.com/share/5b22781e-07ae-414f-8de9-f43af06eec0d

which also had poor quality in the incorrect multiple choice options. "Its Informatics" and keywords sections are generally correct and do not need much human modification.

Moreover, when asking the model to generate new tasks, the new generations still look very similar to the previously generated ones. This hints at the importance of the context and the need to update it after each generation to avoid repetition. This is a known challenge for LLMs [4].

We have also applied our pipeline to generated tasks, and the results for "Mirror Maze" are provided in the fourth column of Table 1.

4 Future Work

We are planning to get ISSEP 2024 participants' input about the proposed process and then use the received comments for improvements. The goal is to develop a robust version of the process for evaluating AI-generated Bebras tasks, which is also an important step in facilitating the use of LLMs to generate tasks for CT learning.

References

1. Dagienė, V., Futschek, G.: Bebras international contest on informatics and computer literacy: Criteria for good tasks. In: Informatics Education-Supporting Computational Thinking: Third International Conference on Informatics in Secondary Schools-Evolution and Perspectives, ISSEP 2008 Torun Poland, July 1-4, 2008 Proceedings 3. pp. 19–30. Springer (2008)
2. Topsakal, O., Akinci, T.C.: Creating large language model applications utilizing langchain: A primer on developing llm apps fast. International Conference on Applied Engineering and Natural Sciences **1**(1), 1050–1056 (Jul 2023). https://doi.org/10.59287/icaens.1127, https://as-proceeding.com/index.php/icaens/article/view/1127
3. Vaníček, J.: Bebras informatics contest: criteria for good tasks revised. In: Informatics in Schools. Teaching and Learning Perspectives: 7th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, ISSEP 2014, Istanbul, Turkey, September 22-25, 2014. Proceedings 7. pp. 17–28. Springer (2014)
4. Xu, J., Liu, X., Yan, J., Cai, D., Li, H., Li, J.: Learning to break the loop: Analyzing and mitigating repetitions for neural text generation. Advances in Neural Information Processing Systems **35**, 3082–3095 (2022)
5. Zhang, Y., Li, Y., Cui, L., Cai, D., Liu, L., Fu, T., Huang, X., Zhao, E., Zhang, Y., Chen, Y., et al.: Siren's song in the ai ocean: a survey on hallucination in large language models. arXiv preprint arXiv:2309.01219 (2023)
6. Zheng, L., Chiang, W.L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E., et al.: Judging llm-as-a-judge with mt-bench and chatbot arena. Advances in Neural Information Processing Systems **36** (2024)

Table 1. Different items in the criteria set, their description, example LLM prompt and pipeline evaluation summary for the "Mirror Maze" task (First Three Rows).

Criteria	**Description**	**Example LLM Prompt**	**Answer Summary**
Relevance to Informatics	Task relates to core informatics concepts such as algorithms, data representation, or logical thinking.	Which informatics concepts are covered by this task?	Algorithms, programming, data structures, and representations.
Clarity and Understandability	Problem statement is clear, concise, and easy to understand.	You are a [intended age] years old student, write what do you understand from the question in few sentences.	The task involves understanding how light reflects off mirrors in a grid and determining the path it takes to exit the maze.
Learning Experience	Task provides a meaningful learning experience, teaching new concepts or reinforcing existing knowledge.	What will a student learn from solving this task?	Students will learn about ray tracing, algorithmic thinking, and the use of grids and conditionals in solving problems.
Time Appropriateness	Task can be solved within an average time of 3 minutes.	What is estimated time needed to solve this task for [its intended audience]?	Around 10-15 minutes
Has a difficulty Level	Simple, all pupils of the target group should be able to solve. Intermediate, challenging tasks that need some thinking to solve and Hard which only the best can solve.	What is the difficulty level of this task (easy, intermediate, hard) for [the intended audience]?	12-14yo: Hard, 14-16yo: Medium, 16-19yo: Easy.
Age Appropriateness	Task is suitable for the specific age group it is intended for.	What age is the appropriate age for being able to solve this task?	Around 14-16
Curriculum Independence	Task is independent of any specific curriculum or educational system.	What specific curriculum or educational system is needed to solve this task?	General informatics or computer science knowledge is sufficient
System Independence	Task is independent of specific IT systems, software, or programming languages.	What specific IT systems, software, or programming languages are needed to solve this question?	None.
Single Screen Presentation	Task can be presented on a single screen without requiring scrolling.	Does this task fit on a single screen without requiring scrolling?	Yes
No Additional Tools Required	Task can be solved without the need for additional hardware, software, or materials like paper and pencil.	Does solving this question require hardware, software, or materials like paper and pencil?	No
Political Correctness	Task avoids gender, racial, or religious stereotypes and biases.	What are gender, racial or religious stereotypes and biases in this task?	None
Engagement and Fun	Task is engaging, interesting, and potentially fun for the participants.	How engaging and fun is this task for students?	Moderately engaging; the problem-solving aspect is intriguing.
Quality of Wrong Choices	In multiple-choice tasks, wrong choices are plausible and designed to avoid easy guessing.	Are the wrong choices plausible and designed to avoid easy guessing?	The wrong choices are plausible, designed to prevent easy guessing and encourage careful consideration.
Cognitive Load	Task's cognitive load is appropriate for the intended age group, avoiding overwhelming students.	Does the task's cognitive load match the [intended audience]?	The cognitive load is appropriate for the intended age groups, challenging but achievable.
Text-Based Visual Elements	Task effectively uses text-based elements to create visual representations or explanations.	Does the task effectively use text-based elements to create visual representations or explanations?	Yes, it effectively uses text-based elements to explain the problem.
Discovery Learning	Task allows students to discover principles or solutions through exploration rather than merely providing instructions.	Does the task allow students to discover principles or solutions through exploration rather than merely providing instructions?	The task allows for exploration as students must mentally or physically trace the light's path, discovering the solution through analysis rather than direct instruction.

Fostering AI Literacy in Primary Education

Gabrielė Stupurienė[1] [0000-0001-5577-1054] and Gintarė Perednytė[2]

[1] Institute of Data Science and Digital Technologies, Vilnius University, Lithuania

[2] Institute of Educational Sciences, Vilnius University, Lithuania

gabriele.stupuriene@mif.vu.lt, gintare.perednyte@gmail.com

Abstract. This study explores the development of artificial intelligence literacy in primary education, emphasizing the importance of early introduction to AI concepts. AI literacy encompasses not only understanding AI technologies but also developing skills necessary for effective use in the 21st century. The research involved three educational activities conducted with fourth-grade students to introduce them to AI and its underlying principles. A mixed-methods approach was used to evaluate the effectiveness of these activities, with data collected through questionnaires. Findings indicate that students are highly motivated and capable of developing AI literacy through engaging and interactive activities. The study concludes that integrating AI literacy into the primary school curriculum is essential.

Keywords: AI literacy, primary education, educational activities, student engagement.

1 Introduction

Artificial intelligence literacy (AI literacy) is a multifaceted concept that includes not only the understanding of artificial intelligence technologies but also the skills and competencies that every 21st-century citizen needs to effectively use AI technologies [1]; [2]. Incorporating AI literacy into the primary education curriculum earlier enables students' ability to effectively navigate a complex technological sea [3]; [4]. There is a need to develop AI literacy in primary school by integrating AI concepts into core subjects, developing competency frameworks, and co-creating curricula with teachers to enhance the learning experience [5]. By understanding how artificial intelligence works and how it can be used in different fields, children are empowered to understand and analyze complex problems better. AI literacy can help students make informed decisions and contribute responsibly to society's technological progress.

All children must be able to recognize examples of AI in their environment, understand the operation of common artificial intelligence algorithms, apply these algorithms to solve personally significant problems, and critically assess the impact of AI systems on society [6]. To effectively develop artificial intelligence literacy, it is necessary [4]: introduce children to the basic concepts of AI and informatics at an early age; encourage them to explore the connections between AI programs and the underlying concepts.

2 Educational Activities

Based on the literature and examples, three educational activities were prepared and tested with fourth-grade students. Each session lasted up to 45 minutes. These activities were designed to introduce students to AI and five AI literacy topics [7]; [8]; [9].

The first educational activity, ‘How does AI perceive the world?’ was intended to reveal what AI is, where it can be found, and how AI gets data from the world. Students were given the following learning tasks:

1. Using the knowledge gained during the discussion, students could name at least 3 devices or apps that use AI.

2. Using the information provided and after testing the AI app, they could name at least two sensors used by AI and give examples of their use.

Interaction with AI apps and devices develops AI literacy [3], so in the first session, it was chosen to allow students to test and explore AI technology independently. After that, students' theoretical knowledge acquired at the beginning of the session and practical experience in AI technology were compared to avoid the gap in linking AI concepts to practical skills [4].

The second educational activity was to discover what kind of thought processes AI can perform and how it does it. The task posed in this activity: using the information provided and the completed tasks, students could be able to describe how AI ‘reasons’. The activity covered the principle of AI operation, decision tree structures, and errors made by AI, which helped to see the bias of AI. The game activity ‘good monkey, bad monkey’ was selected from the methodological material for students to get to know the AI ‘thinking’ process more deeply [10]. Worksheets were also prepared in which students looked for repetitions, similarities, and differences using drawings of monsters and drew them themselves. Students looked for differences and repetitions, tried to identify objects that fit the group, and questioned the mistakes made by AI. By participating in activities and completing tasks, students could see the choices and moves made by AI and discover repetitions and similarities, which are an integral part of AI. Also, during the session, the mistakes made by AI were discussed, and with the help of the discussion, the topic of ethics, which is extremely important for the education of AI literacy, was touched upon. The consequences of using AI indiscriminately and without verifying information were considered during the discussion, and the threats were discussed. This topic highlighted the importance of students' critical thinking, one of the AI literacy skills [6]; [3].

The third learning activity focused on what machine learning is, how it works, and potential biases in AI. Students were given two tasks:

1. Through collaboration, students could build and test a machine-learning model.

2. Based on the information provided, students could describe possible biases in machine learning.

These skills were developed by reviewing the principles of learning AI technologies using visual aids. Following [11], a machine learning model development activity was conducted using the Teachable machine program. According to [12], after exploring the basic concepts, children can begin to design and construct their own machine-learning programs with the help of more experienced peers or adults. Through iterative processes of design, experimentation, and play, students can gradually gain a better understanding of various machine learning methodologies, data collection, concepts such as under- and over-shooting, and the iterative process of testing and improving their systems. When testing their models, students encountered rudimentary examples of bias in the lack of diversity in the data and discussed these in a discussion, which builds an understanding of how AI technology works.

3 Methodology

The study was conducted during classes for fourth-grade students (ages 10-11) to evaluate the possibilities of artificial intelligence literacy development in primary school. 17 students from the public school participated in the study, 11 of them in all three activities, so only their data were included in the data analysis. The students had no previous AI literacy lessons. Each activity was followed by a discussion and summarization of information, after which the students were given a questionnaire to which they answered the questions in writing. A written survey was given to students after each activity as a reflection. Students evaluated the activity and their improvement 5 minutes before the end of the lesson. The questionnaire consisted of 6 questions, of which 3 were quantitative statements that had to be evaluated on a 5-point Likert scale, and three were open-ended qualitative questions.

A mixed strategy was chosen for the empirical research of the study. Quantitative data obtained during the study were analyzed using descriptive statistics. The questionnaire's responses to the open-ended questions were analyzed using thematic data analysis. Thematic analysis is a method that helps in coding qualitative data to identify typical patterns and formulate themes related to the research problem [13]. Specifically, structured tabular thematic analysis was applied in this case, which provides an adaptable technique for relatively structured work with short qualitative data [14].

4 Conclusion

The theoretical research analysis revealed that playful, involving digital technologies, creative activities and cooperation motivate students and arouse their curiosity to learn about AI. The use of various activities in the education of AI literacy is an important aspect of the multifaceted education of students.

Analysis of the research data revealed that students are sufficiently motivated and capable of developing AI literacy in primary school, and additional support provided can further encourage student engagement. Also, it became clear that the opportunities for the development of AI literacy in primary school are revealed through activities that interest and educate students. Examining AI concepts and operating principles, using AI technologies, and ethical discussions are interesting and engaging activities for students.

The results of the empirical study show that AI literacy educational activities that use digital tools, collaboration, creativity, and reflect the five big ideas of AI literacy are suitable for developing AI literacy in primary school. To ensure that all students have the opportunity to learn and develop skills in the field of AI, it is necessary to promote the inclusion of AI literacy in the learning process. It is important that the development of AI literacy starts already in primary school. Future research could examine effective teacher education strategies, curriculum development, assessment methods, and the long-term effects of AI literacy on students' academic achievement.

References

1. Ng, D. T., Leung, J. K., Su, J., Ng, R. C., & Chu, S. K.: Teachers’ AI digital competencies and twenty-first century skills in the post-pandemic world. Educational technology research and development 71(1), 137–161 (2023)
2. Su, J., Ng, D. T., & Chu, S. K.: Artificial intelligence (AI) literacy in early childhood education: The challenges and opportunities. Computers and Education: Artificial Intelligence 4, 100124 (2023)
3. Yang, W.: Artificial intelligence education for young children: Why, what, and how in curriculum design and implementation. Computers and Education: Artificial Intelligence 3, 100061 (2022)
4. Kandlhofer, M., Steinbauer, G., Hirschmugl-Gaisch, S., & Huber, P.: Artificial intelligence and computer science in education: From kindergarten to university. 2016 IEEE Frontiers in Education Conference (FIE) (2016)
5. Casal-Otero, L., Catala, A., Fernández-Morante, C. et al.: AI literacy in K-12: a systematic literature review. IJ STEM Ed 10, 29 (2023)
6. Ali, S., Payne, B. H., Williams, R., Park, H. W., Breazeal, C.: Constructionism, ethics, and creativity: Developing primary and middle school artificial intelligence education. In International workshop on education in artificial intelligence K-12 (Vol. 2, pp. 1-4). MIT MEDIA LAB Palo Alto, California (2019)
7. AI4K12: The Artificial Intelligence (AI) for K-12 initiative (2022) https://ai4k12.org/
8. Su, J., Yang, W.: Artificial Intelligence and Robotics for Young Children: Redeveloping the Five Big Ideas Framework. ECNU Review of Education (2023)
9. Touretzky, D., Gardner-McCune, C. & Seehorn, D: Machine Learning and the Five Big Ideas in AI. Int J Artif Intell Educ 33, (pp. 233–266) (2023)
10. Dagienė, V., Masiulionytė-Dagienė, V.: Metodinė medžiaga. Dirbtinis intelektas. Žaidybinės veiklos pradiniam ugdymui (2022)
11. Sayginer, Ş., Tekdal, M.: The potential of using the teachable machine application for machine learning in education. 16th International computer and instructional technologies symposium (pp. 14-22) (2023)
12. Vartiainen, H., Tedre, M., Valtonen, T.: Learning machine learning with very young children: Who is teaching whom? International Journal of Child-Computer Interaction 25, 100182 (2020)
13. Braun, V., & Clarke, V.: Thematic analysis: A practical guide. SAGE (2021)
14. Robinson, O. C.: Conducting thematic analysis on brief texts: The structured tabular approach. Qualitative Psychology, 9(2) 194–208 (2022)