Storytelling as a Pedagogical Tool in Computer Science Education: A Case Study on Software Systems Verification and Validation

Bogdan Iudean[a]

Babeş-Bolyai University, Faculty of Mathematics and Computer Science, Computer Science Department, Cluj-Napoca, Romania

Keywords: Computer Science Education, Digital Storytelling, Software Systems Verification and Validation.

Abstract: This position paper explores the potential of storytelling as a teaching method in Computer Science Education. It addresses the challenges faced in traditional teaching methods and advocates the effectiveness of storytelling in enhancing students' understanding and engagement. The paper presents a case study on using storytelling in teaching Software Systems Verification and Validation, evaluating its impact on students' learning outcomes and feedback. It also discusses potential objections and limitations of this method, providing counterarguments based on findings and previous studies. The paper concludes with reflections on the potential of storytelling in Computer Science Education and suggestions for future research and practice.

1 INTRODUCTION

Imagine a world where computer science is not just about coding and algorithms, but also about stories. Picture a classroom where students are not just listening to lectures, but are captivated by tales that bring concepts to life. This is the vision we are exploring, the integration of storytelling into computer science education.

Computer Science (CS) (Committee, 2016) is a rapidly evolving field that has a significant impact on various aspects of human life, such as communication, entertainment, health, education, and security. However, Computer Science Education (CSE) faces many challenges (Webb et al., 2017), (YeckehZaare et al., 2022), (Wang, 2021) in preparing students to become competent and creative problem solvers in the digital age. According to (Towey and Chen, 2015), some of these challenges, especially when teaching subjects like Software Systems Verification and Validation (SSVV), include students' lack of understanding of the big picture of the learned concepts, difficulty in linking the learned concepts to real-world applications, and low levels of engagement and motivation in the classroom. Moreover, traditional teaching methods, such as lectures, textbooks, and exams, may not be sufficient or effective in fostering students' computational thinking skills, creativity, and collaboration. Therefore, there is a need for new and innovative teaching methods that can address these challenges and enhance students' learning outcomes and experiences.

Storytelling, a promising teaching method, has gained attention in recent years (Barchas-Lichtenstein et al., ; Landrum et al., 2019). It conveys messages, emotions, and values through narratives, with its versatility facilitating already various educational fields, including fostering creativity in Architecture Design (Lee et al., 2023) and enhancing data literacy in Data Science (Li et al., 2023). As a pedagogical tool, storytelling practices have extended even to encouraging students to create digital stories themselves (Orhan Göksün and Gürsoy, 2022), which are multimedia products that combine text, images, sound, and animation.

This paper aims to advocate for the assertion that storytelling, whether facilitated by the teacher or guided to be crafted by students themselves, provides a meaningful additional educational tool for Computer Science students. We advocate for the idea that this approach mitigates consecrated CS problems by:

- aiding students in understanding the big picture of the learned concepts,
- bridging the gap between the theoretical concepts and real-world problems and solutions,
- increasing students enjoyment of computer science education.

[a] https://orcid.org/0000-0001-6233-3570

Iudean, B.
Storytelling as a Pedagogical Tool in Computer Science Education: A Case Study on Software Systems Verification and Validation.
DOI: 10.5220/0012687300003693

In *Proceedings of the 16th International Conference on Computer Supported Education (CSEDU 2024) - Volume 2*, pages 557-564
ISBN: 978-989-758-697-2; ISSN: 2184-5026

The paper is organized as follows: Section 2 outlines the current challenges in Computer Science Education. In Section 3 we present our position on the use of storytelling in teaching computer science concepts, along with a case study on teaching software testing concepts. Section 4 provides several counterarguments to our position and also explanations on why our position is still valid. Section 5 outlines the conclusions and offers some further future research actions.

2 CHALLENGES ON COMPUTER SCIENCE EDUCATION

In this section, we explore the fundamental challenges within computer science education, elucidating the gaps in comprehensive understanding, real-world applicability, and student engagement. Additionally, we underscore the pertinence of these challenges in specialized subjects like Software Systems Verification and Validation (SSVV), highlighting the critical need for innovative teaching approaches to enhance the educational experience in these domains.

2.1 Lack of Big Picture

In the context of CSE, the "big picture" refers to the holistic understanding of how individual concepts and topics interrelate and contribute to the broader field. However, students often learn these topics in isolation, making it difficult for them to see how these pieces fit together (Tim, 2018) to form a coherent whole. This disjointed learning experience can lead to a fragmented understanding of the subject (Tim, 2018), where students see multiple topics spread over several years as a disjoint set of independent academic ideas and practical skills that do not necessarily have an obvious purpose.

SSVV is a discipline that sits at the intersection of several complex domains (Clear et al., 2019). It involves ensuring that a system meets specified requirements (verification) and that it fulfills its intended purpose (validation). This requires a deep understanding of both the system under test and the principles of SSVV.

The importance of imparting Software Testing (ST) concepts in education is increasingly recognized. The Computing Curricula 2020 (Clear et al., 2019) advocates for a competency-based approach to computing. In the field of Software Engineering (SE), Verification and Validation are assigned a competency level ranging from 3 to 5 for undergraduate students (in terms of computing knowledge). Essential ST skills highlighted in the SE competencies include the ability to perform integrative testing and analysis of software components using the black-box method, conduct regression testing of software components, utilize suitable testing tools for testing, and devise and implement strategies for test case design.

Moreover, the systems that are subject to SSVV are often complex in their own right. They can involve multiple interacting components, each with its own logic and behavior. Understanding these systems requires a solid grasp of various Computer Science concepts (Cheah, 2020), from algorithms and data structures to software engineering and artificial intelligence.

Addressing this challenge requires a shift in the way Computer Science traditionally is taught. Instead of teaching topics in isolation, educators need to focus on showing students *how individual concepts fit into the larger context of the field*. This approach can help students see the relevance of what they are learning, thus enhancing their understanding and engagement with the subject (Webb et al., 2017).

2.2 Missing Link to Real-World Scenarios

Another significant challenge in CSE and SSVV is the difficulty students face in linking the learned concepts to real-world applications. This issue stems from the abstract nature of many Computer Science concepts, which can make it difficult for students to see their practical applications (Sezer and Namukasa, 2021).

This challenge is particularly pronounced in SSVV, where the complexity of the systems under test and the intricacy of the verification and validation processes can make it difficult for students to see the real-world relevance of what they are learning. For instance, a student may understand the concept of a black box testing boundary value analysis (Sholeh et al., 2021) in theory, but may not be able to see how it is used to verify the correctness of a real-world software system.

Moreover, the rapid pace of technological advancement means that the real-world applications of Computer Science and SSVV are constantly evolving (Yazdanian et al., 2021). This can make it even more challenging for educators to provide up-to-date, relevant examples of how concepts taught in class are used in practice.

Addressing this challenge requires a pedagogical approach that *emphasizes real-world applications* and provides students with ample means to understand *how the concepts they learn are applied in practical scenarios*.

2.3 Lack of Engagement

In the context of Computer Science education, lack of student engagement can be a significant challenge (Sinclair et al., 2015). This issue can manifest itself in various forms, such as low participation in class discussions, minimal interaction with course materials, and lack of enthusiasm for practical assignments.

One of the primary reasons for this lack of engagement is the abstract nature of many Computer Science concepts. Students often find it difficult to relate these abstract concepts to real-world applications (Mirolo et al., 2022). This disconnect can lead to a lack of interest and engagement in the subject matter. Another contributing factor is the rapid evolution of technology. The fast-paced changes in the field can make some of the curriculum content seem outdated, leading to a perception that the material is not relevant. This perception can significantly reduce student engagement.

Moreover, the traditional lecture-based teaching methods used in many Computer Science classes do not cater to the diverse learning styles of students. These methods often fail to provide an interactive learning environment, which is crucial for engaging students.

Addressing this challenge requires a shift in pedagogical approach. *Incorporating interactive teaching methods*, such as digital storytelling, can cater to diverse learning styles. *Utilizing technology for virtual labs* and online discussions can also foster an engaging learning environment. By doing so, we can ensure that Computer Science education is inclusive, effective, and stimulating for all students.

3 STORYTELLING APPROACH IN TEACHING COMPUTER SCIENCE CONCEPTS

In this section, we assert our position on the use of storytelling in teaching computer science concepts, emphasizing its transformative potential and impact in CSE. Additionally, we provide insights from our personal case study, employing storytelling as a teaching method for SSVV.

Throughout this paper, the term *"storytelling"* will be used as an umbrella term covering all variations of educational acts involving the exposure of students to a narrative involving a scenario.

3.1 Impact and Benefits of Storytelling in Teaching Computer Science Concepts

This section explores how storytelling techniques and their integration are harmonizing with established pedagogical methods, offering a multifaceted approach in CSE. Through contextual narratives that bridge theoretical concepts to real-world scenarios, we explore their benefits not only in fostering a comprehensive understanding but also in significantly enhancing student engagement.

3.1.1 Storytelling Fosters Comprehensive Understanding

The integration of storytelling in Computer Science Education, particularly in subjects like SSVV, provides a unique opportunity to contextualize learning, making abstract concepts more relatable and understandable.

Storytelling allows the presentation of complex computer science concepts in a more digestible format. The (Saritepeci, 2021) study found that Digital Storytelling Activities (DST) in Science class had a positive effect on instructional effectiveness and learners' satisfaction. By presenting computer science concepts within the framework of a story, students may find it easier to understand and remember these concepts. The study (Saritepeci, 2021) supports this by showing that DST implementation made learners have a high level of satisfaction during the learning process.

Storytelling can help students see the relevance of computer science in their everyday lives. Another research project aimed to create a constructivist learning environment with digital storytelling. The (Smeda et al., 2014) research investigated the pedagogical aspects of digital storytelling and the impact of digital storytelling on student learning when teachers and students use digital stories. Through storytelling, students can see how computer science principles apply in real-world scenarios, enhancing their understanding and retention of the material. The research (Smeda et al., 2014) supports this by suggesting that digital storytelling is a powerful tool to integrate instructional messages with learning activities to create more engaging and exciting learning environments.

Storytelling can foster a more inclusive learning environment in computer science education. Incorporating diverse narratives into the curriculum can help to engage a broader range of students (Vivitsou, 2018). Storytelling can help to break down barriers and make computer science more accessible for

students of diverse backgrounds. The (Smeda et al., 2014) research supports this by suggesting that digital storytelling is a meaningful approach to creating a constructivist learning environment based on novel principles of teaching and learning.

3.1.2 Bridging Theory and Practice Through Storytelling

Storytelling in Computer Science Education, particularly in subjects like SSVV, serves as a powerful tool for linking theoretical concepts to real-world applications, thereby enhancing the practical understanding of students.

Storytelling provides a platform for demonstrating the real-world applications of computer science concepts. The (Rainer, 2021) study found that storytelling in human-centric software engineering research can contribute to data collection, data analyses, ways of knowing, research outputs, interventions in practice, and advocacy, and can integrate with evidence and arguments. By incorporating real-world scenarios into stories, students can see the practical applications of the concepts they are learning, which can enhance their understanding and interest in the subject. The study (Rainer, 2021) supports this by showing that storytelling implementation made learners have a high level of satisfaction during the learning process.

Storytelling can help students understand the impact of computer science on society. Another research project, (Trichopoulos et al., 2023), aimed to create a constructivist learning environment with digital storytelling. The research investigated the pedagogical aspects of digital storytelling and the impact of digital storytelling on student learning when teachers and students use digital stories. Through storytelling, students can gain insights into how computer science concepts and technologies impact society, which can motivate them to learn more and contribute positively to the field.

3.1.3 Enhancing Engagement with Storytelling

Incorporating storytelling as a pedagogical tool can also be an effective strategy to enhance student engagement. Storytelling can make the learning process more heuristic, interactive, and relatable, thereby increasing student interest and participation.

The adoption of Story Programming in computer science has gained prominence for its pronounced impact on student engagement and learning outcomes. Studies in the domain of engineering education have shown that the incorporation of storytelling as an instructional aid fosters critical thinking (Osgood and Bressan, 2021). This robust body of research supports the claim that Story Programming serves as a compelling tool not only for engaging students but also for enriching their overall learning experience.

Storytelling in the form of Story Programming brings a strong factor of engagement that can motivate students to attend the lab out of sheer curiosity for the story outcome even when they were supposed to be absent. While this is a bold claim, it is grounded in the first storytelling experiments that we conducted at our University, *Babes-Bolyai University of Cluj-Napoca*, in 2021 (Lőrincz et al., 2021). Through a carefully designed experiment, we introduced various gamification and alternative teaching methods in the SSVV classes, among which storytelling showed promising results.

3.2 Case Study: Using Storytelling in Teaching SSVV Concepts

Drawing from our own experiences in the implementation of storytelling techniques for the SSVV laboratories, we continued to experiment with the design of the narrative method.

We will briefly describe the methodology for teaching SSVV concepts with our latest narrative approach and a short overview of open-ended feedback collected to enforce our claims.

3.2.1 Our Design on Using Storytelling for SSVV Laboratory Work

The SSV labs spanned the entire semester (12 weeks) for third-year students, with each lab lasting 2 hours. Students collaborated in pairs on tasks assigned for in-class implementation. The labs covered a diverse range of subjects, including Black Box Testing, White Box Testing, Integration Testing, and Web Testing.

Incorporating storytelling into the lab sessions has been seamless, acting as a supplementary tool without disrupting the existing structure. This integration enhanced the learning experience by providing an additional layer of engagement while still relying on the traditional activities of problem solving during the rest of the lab. The stories introduced were carefully tailored to align with the subject matter covered in each session.

The storytelling teaching method centers around a series of interconnected stories featuring Tudor the Tester, a fictional Stone Age character responsible for validating the development of the first-ever stone-age car. Through these narratives, we impart crucial testing concepts, employing primitive-themed yet imaginative representations of modern testing tools and

strategies. At the beginning of each lab, the teacher presented short stories using live narration, accompanied by animated presentations. The visual presentations featured custom-made, illustration-rich slides depicting a cartoon-like scenery. In these slides, the characters moved and interacted with each other and with technologies, as illustrated in Figure 1. The narratives serve as an engaging introduction, seamlessly blending humor and educational content. Despite Tudor the Tester's frequent humorous failures, the stories set the stage for the day's learning objectives.

Figure 1: Sample from the Black Box Testing laboratory slides.

The laboratories themselves involve hands-on exercises, often completed in teams. This team-based approach has proven to be a key factor in the positive experiences reported by students. The interactive and immersive presentations delivered by the professor at the beginning of each lab have been consistently highlighted as a source of enjoyment and understanding. The use of team-based exercises and take-home components has also contributed to a relaxed yet productive learning atmosphere.

3.2.2 Impact Analysis of Storytelling on Student Engagement and Concept Understanding

In response to inquiries regarding the utilization of storytelling and their grasp of concepts, the students provided valuable insight. This comprehensive feedback forms a crucial part of our exploration into the effectiveness of storytelling as a pedagogical tool in computer science education.

The students have been provided an open-ended feedback question: *Mention some aspects that gave you a pleasant experience in the learning process based on the Laboratory Activities.*. A short summary of samples from the positive responses is listed:

- *"Probably the most memorable aspect was the laboratory teacher. I thought that his presentation and teaching style was very enjoyable and efficient, and I loved the presentations he made. I thought they were both entertaining and educational. Similarly to the seminar, I liked that the lab activities were team-based, and also the fact that they had a take-home component."*
- *"The teacher made the labs with cool presentations and stories, which i liked alot."*
- *"I loved the story telling and enthusiasm of the teacher."*
- *"The presentations of the teacher were the best."*
- *"The story that was presented, with the cave man and so on, it was a cool story honestly."*
- *"The immersive presentations done by the lab teacher."*
- *"Teachers' highly interactive method of teaching."*

Reflecting on the responses, a high level of satisfaction and engagement can be drawn from the experimental storytelling method. The comments consistently praise the teachers' presentations, describing them as both entertaining and educational. The use of storytelling is frequently cited as a memorable and enjoyable aspect, contributing to an overall positive and relaxed learning experience. The approachability and enthusiasm of the teacher, coupled with the team-based activities, have garnered acclaim, creating an environment where students feel encouraged to ask questions and seek help. The integration of Stone Age-themed stories collectively contributes to a multifaceted learning experience that resonates with students.

On the other hand, the negative feedback included the following statements:

- *"nothing was particularly pleasant. More like it was a pleasant feeling when I finally managed to understand and finish the assignments."*
- *"Eh.. I was not too found of laboratories :D"*
- *"I didn't particularly enjoy the lab assignments, but they taught me a lot."*

It is essential to acknowledge that storytelling, as a teaching method, may not cater comprehensively to all learning styles. While these negative responses were anticipated, the overall positive ratio of feedback encourages us to further refine and diversify our approach. This constructive criticism motivates us to explore ways to make storytelling in education even more inclusive, ensuring a richer learning experience for all students.

One student provided a detailed feedback on the laboratory activities, stating: "Looking back at the laboratory activities, several aspects provided a pleasant learning experience:

1. Hands-on Learning. The labs offered a practical learning environment, which I found engaging and exciting. There's something truly satisfying about seeing theories come to life and directly applying what I've learned in lectures.
2. Collaborative Environment. Labs often required teamwork, which added an interactive element to the learning experience. It was both enjoyable and rewarding to solve problems together and learn from my peers' perspectives and approaches.
3. Real-World Relevance. The presentations, problems and tasks in the labs often mirrored real-world scenarios. This connection made the learning more meaningful and engaging as it showed the practical relevance of the concepts we were learning.
4. Experimentation. The labs provided an opportunity for experimentation, which was both fun and educational. Making mistakes, troubleshooting, and eventually finding solutions was a fulfilling process that deepened my understanding of the subject.
5. Guided Learning. Although labs provided room for independent work, the availability of guided instructions and help from lab instructors when needed created a supportive and conducive learning environment. This balance boosted my confidence and made the learning process enjoyable.
6. Progressive Difficulty. The labs were designed with increasing levels of complexity, which made them challenging but also very rewarding. Overcoming each stage gave a sense of accomplishment that added to the overall pleasant experience."

The student finalized his/her feedback with the statement "Overall, these aspects of the lab activities greatly enhanced my learning experience, making it both enjoyable and beneficial."

In conclusion, our initiative into experimental storytelling in SSVV laboratories has not only enriched the teaching of testing concepts, but has also established a positive and engaging environment for both students and educators. Embracing storytelling as part of our educational toolkit while keeping the traditional teaching approaches has brought one extra step towards a more modern and efficient teaching and learning experience. The evidence collected through open-ended feedback reinforces the success of this approach, emphasizing its effectiveness in fostering a memorable and enjoyable learning journey.

4 COUNTERARGUMENTS AND EXPANSIONS ON OUR POSITION/STANDPOINT

In this section, we delve into counterarguments surrounding the integration of storytelling in education, acknowledging potential challenges. As an alternative teaching method not fully formalized, understanding both its potentials and limitations is important for a comprehensive assessment.

While storytelling can indeed make complex computer science concepts more digestible, a potential counterargument could be that it may oversimplify these concepts. This oversimplification could potentially lead to a lack of depth in understanding. For example, while a story might make a concept easier to grasp initially, it might not fully capture the complexities of the concept. On a similar note, storytelling might not cater to all learning styles. Although some students may find narrative-based learning engaging, others may prefer more traditional, direct methods of instruction. *Despite these arguments, we advocate for integrating storytelling's oversimplification* alongside traditional teaching methods' rigor to enhance the educational experience. Storytelling could potentially simplify and reduce the cognitive load (Sweller, 2020) in initial subject interactions, setting the stage for addressing more complex aspects through established educational methods, creating a heuristic learning journey.

Creating and implementing narrative scenarios in a curriculum can be time-consuming and resource-intensive (Falcon, 2012). Teachers may lack the technical knowledge of creating animated presentations, or these may take a significant amount of time to implement. *Despite these arguments, it is crucial to recognize the flexibility* inherent in storytelling methods. While formalization and standardization of teaching resources for storytelling can be demanding, it is important to emphasize that storytelling in education manifests itself in various lightweight forms. From oral presentations to computer-assisted, film-like methods (Hagedorn et al., 2023) and gamified approaches (Karram, 2021) to interactive cooperative games (Bell and Vahrenhold, 2018), storytelling offers a spectrum of options with differing levels of implementation effort.

Moreover, storytelling is highly dependent on the narrative skills of the educator. If the educator is not adept at crafting and delivering engaging and accurate stories, the effectiveness of this method could be compromised. *Despite these arguments, in line with our thesis, we promote storytelling as a teaching approach* that does not require the teacher to deliver the

story personally. This allows for engaging activities, such as designating students to craft a subject-related story for bonus points, and presenting it in the following lesson (Gürsoy, 2021).

5 CONCLUSION AND FUTURE WORK

In this position paper, our objective is to advocate for the assertion that storytelling, whether facilitated by the teacher or guided to be crafted by students themselves, provides a meaningful additional educational tool for computer science students. This approach helps to understand the big picture of the concepts learned, linking them to real-world problems and solutions, and ultimately increasing their enjoyment of computer science education.

The main benefits of incorporating storytelling into computer science education include improved comprehension, real-world application of theoretical knowledge, and increased student engagement.

Looking ahead, there are specific actions to be taken to further investigate storytelling in the context of computer science education. Expanding its application to a wider range of subjects and creating a common theme for stories across various fields could make the concepts more interdisciplinary and universally applicable.

One avenue for improvement is to enhance the effectiveness of our animated live narrations, potentially addressing more learning modes and preferences to cater to a diverse student audience.

Another important area for future research is to delve deeper into the long-term impact of storytelling on the retention and application of computer science concepts beyond the immediate learning context.

In addition, it is important to explore ways to make storytelling more accessible and inclusive, considering different learning styles and backgrounds among students.

We hope that this research encourages researchers to explore other avenues in the integration of storytelling within Computer Science Education, fostering continuous innovation and improvement in pedagogical approaches.

REFERENCES

Barchas-Lichtenstein, J., Sherman, M., Voiklis, J., and Clapman, L. Science through storytelling or storytelling about science? identifying cognitive task demands and expert strategies in cross-curricular stem education. In Frontiers in Education, volume 8, page 1279861. Frontiers.

Bell, T. and Vahrenhold, J. (2018). Cs unplugged—how is it used, and does it work? Adventures between lower bounds and higher altitudes: essays dedicated to Juraj Hromkovič on the occasion of his 60th birthday, pages 497–521.

Cheah, C. S. (2020). Factors contributing to the difficulties in teaching and learning of computer programming: A literature review. Contemporary Educational Technology, 12(2):ep272.

Clear, A., Parrish, A. S., Impagliazzo, J., and Zhang, M. (2019). Computing curricula 2020: Introduction and community engagement. SIGCSE '19, page 653–654, New York, NY, USA. Association for Computing Machinery.

Committee, K.-. C. S. F. S. (2016). K-12 computer science framework. Technical report, New York, NY, USA.

Falcon, R. (2012). A counterargument to ineffective technology in classrooms: Characteristics of high achieving schools. In Society for Information Technology & Teacher Education International Conference, pages 2774–2780. Association for the Advancement of Computing in Education (AACE).

Gürsoy, G. (2021). Digital storytelling: Developing 21st century skills in science education. European Journal of Educational Research, 10(1):97–113.

Hagedorn, C., Serth, S., and Meinel, C. (2023). The mysterious adventures of detective duke: How storified programming moocs support learners in achieving their learning goals. In Frontiers in Education, volume 7, page 1016401. Frontiers Media SA.

Karram, O. (2021). The role of computer games in teaching object-oriented programming in high schools-code combat as a game approach. WSEAS Transactions on Advances in Engineering Education, 18:37–46.

Landrum, R. E., Brakke, K., and McCarthy, M. A. (2019). The pedagogical power of storytelling. Scholarship of Teaching and Learning in Psychology, 5(3):247.

Lee, K., Kang, E., and Park, E. J. (2023). Storytelling as a learning tool in creative education: A case study in an architecture design studio. Thinking Skills and Creativity, 48:101274.

Li, Y., Wang, Y., Lee, Y., Chen, H., Petri, A. N., and Cha, T. (2023). Teaching data science through storytelling: Improving undergraduate data literacy. Thinking Skills and Creativity, 48:101311.

Lőrincz, B., Iudean, B., and Vescan, A. (2021). Experience report on teaching testing through gamification. In Proceedings of the 3rd International Workshop on Education through Advanced Software Engineering and Artificial Intelligence, pages 15–22.

Mirolo, C., Izu, C., Lonati, V., and Scapin, E. (2022). Abstraction in computer science education: An overview. Informatics in Education, 20(4):615–639.

Orhan Göksün, D. and Gürsoy, G. (2022). Digital storytelling in science teacher education: Evaluation of digital stories. Science Education International, 33(2):251–263.

Osgood, L. and Bressan, N. (2021). Learning electronics through storytelling. Proceedings of the Canadian Engineering Education Association (CEEA).

Rainer, A. (2021). Storytelling in human–centric software engineering research. In Evaluation and Assessment in Software Engineering, pages 241–246.

Saritepeci, M. (2021). Students' and parents' opinions on the use of digital storytelling in science education. Technology, Knowledge and Learning, 26(1):193–213.

Sezer, H. B. and Namukasa, I. K. (2021). Real-world problems through computational thinking tools and concepts: the case of coronavirus disease (covid-19). Journal of Research in Innovative Teaching & Learning, 14(1):46–64.

Sholeh, M., Gisfas, I., Fauzi, M. A., et al. (2021). Black box testing on ukmbantul. com page with boundary value analysis and equivalence partitioning methods. In Journal of Physics: Conference Series, volume 1823, page 012029. IOP Publishing.

Sinclair, J., Butler, M., Morgan, M., and Kalvala, S. (2015). Measures of student engagement in computer science. In Proceedings of the 2015 ACM conference on innovation and technology in computer science education, pages 242–247.

Smeda, N., Dakich, E., and Sharda, N. (2014). The effectiveness of digital storytelling in the classrooms: a comprehensive study. Smart Learning Environments, 1:1–21.

Sweller, J. (2020). Cognitive load theory and educational technology. Educational Technology Research and Development, 68(1):1–16.

Tim, B. (2018). Computer science in k-12 education: The big picture.

Towey, D. and Chen, T. Y. (2015). Teaching software testing skills: Metamorphic testing as vehicle for creativity and effectiveness in software testing. In 2015 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE), pages 161–162.

Trichopoulos, G., Alexandridis, G., and Caridakis, G. (2023). A survey on computational and emergent digital storytelling. Heritage, 6(2):1227–1263.

Vivitsou, M. (2018). Digital storytelling in teaching and research. Encyclopedia of Education and Information Technologies, Springer.

Wang, G. (2021). Problems and countermeasures in education of computer science and technology in the internet era. In 2021 4th International Conference on Information Systems and Computer Aided Education, ICISCAE 2021, page 1294–1297, New York, NY, USA. Association for Computing Machinery.

Webb, M., Davis, N., Bell, T., Katz, Y. J., Reynolds, N., Chambers, D. P., and Sysło, M. M. (2017). Computer science in k-12 school curricula of the 2lst century: Why, what and when? Education and Information Technologies, 22:445–468.

Yazdanian, R., West, R., and Dillenbourg, P. (2021). Keeping up with the trends: Analyzing the dynamics of online learning and hiring platforms in the software programming domain. International Journal of Artificial Intelligence in Education, 31:896–939.

YeckehZaare, I., Grot, G., Dimovski, I., Pollock, K., and Fox, E. (2022). Another victim of covid-19: Computer science education. In Proceedings of the 53rd ACM Technical Symposium on Computer Science Education-Volume 1, pages 913–919.