Education Tech Research Dev
https://doi.org/10.1007/s11423-025-10500-2

DEVELOPMENT ARTICLE

Applying participatory design for developing an unplugged game to learning graph theory

Ana Paula da Silva[1] · Aluizio Haendchen Filho[1] · Edilson Ferneda[2] · Hércules Antonio do Prado[2] · André Luís Alice Raabe[1]

Accepted: 10 April 2025

Abstract

Teaching Computer Science concepts, such as graph theory, is often challenging. This study proposes an approach for teaching graph theory using an unplugged game (*Graph-Game*) developed through a participatory design process that includes usage observation, clarifying meaning, prototyping, and implementation. This process was carried out with a group of Brazilian middle school students and involved collecting observations, interviews, ideation activities, and iterative prototyping with quantitative and qualitative tests. The game offers an interactive way to explore graph algorithms, improving abstraction capacity, a skill related to Computational Thinking. The effectiveness of the game in facilitating the learning of fundamental graphs concepts among high school students was evaluated with an independent sample of students. The results pointed out the proposal as a promising alternative for teaching graph theory, a complex computing topic, in an engaging way. By enhancing playful learning, this work offers an alternative to make teaching Computer Science more enjoyable and effective.

Keywords Computational thinking · Educational games · Participatory design · Graphs theory

✉ Hércules Antonio do Prado
prado.hercules@gmail.com

Ana Paula da Silva
deanapaulasilva@gmail.com

Aluizio Haendchen Filho
aluizio.h.filho@gmail.com

Edilson Ferneda
eferneda@gmail.com

André Luís Alice Raabe
raabe@univali.br

[1] Graduate Program in Applied Computing, University of Vale Do Itajaí - Univali, Itajaí, Santa Catarina 88302-901, Brazil

[2] Graduate Program in Governance, Technology and Innovation, Catholic University of Brasília, Brasília 71966-700, Brazil

Published online: 21 April 2025

Introduction

The use of information technologies and electronic devices reaches audiences of all ages, spreading the idea that computing is strongly related to adopting these artifacts. In the opposite direction and in a complementary way, Wing (2006) suggests using computing as a way of thinking. The author believes computational knowledge is so relevant that, like reading and writing, Computational Thinking (CT) should also be included in people's lives regardless of the profession.

Building on this idea, knowing concepts such as abstraction, parallelism, error detection, or algorithms can stimulate people's interest in computing and incorporate CT into their lives, enabling them to solve everyday problems (Wing, 2006). According to Costa Júnior et al. (2022), logical reasoning and CT should be taught early on, as they increase the capacity for deduction and problem resolution. In line with this perspective, the Brazilian Computing Society (SBC, 2023) emphasizes that computer proficiency is essential for individuals of all ages. It should, therefore, be introduced in middle school along with other core subjects.

CT involves abstraction, generalization, decomposition of complex problems, pattern recognition, algorithm design, and evaluation of alternative solutions (Grover & Pea, 2013; Wing, 2006). Developing these skills helps solve problems in areas other than IT. However, despite the importance of CT, teaching these concepts is often limited by access to computer equipment and technology. Bell et al. (2009) argued about the benefits of exploring alternatives that allow teaching CT disconnectedly (without using computers or electronic devices).

According to Kazimoglu et al. (2012), playful activities, such as games and puzzles, effectively teach fundamental computational concepts and stimulate logical and algorithmic reasoning. These activities provide an interactive and enjoyable learning experience, engaging students in problem-solving and critical thinking. Sapounidis et al. (2019) emphasize the potential of using manipulative materials, such as logic blocks, to visually represent abstract concepts such as loops, conditionals, and variables. By providing practical, tangible experiences, these disconnected activities make learning more accessible and engaging, allowing students to understand complex ideas more tangibly and intuitively.

Brackmann et al. (2017) highlight how unplugged computing challenges the conventional reliance on machines or electronic equipment for computation. In this sense, Bell et al. (2015) explore diverse methods of teaching computer science concepts without the need for a computer. The authors suggest a collection of affordable and user-friendly unplugged activities that can be implemented in several settings, even by individuals without a specialized background in computing.

This work was focused on designing *GraphGame*, an unplugged game that can teach graph concepts engagingly. The investigation focuses on how an unplugged game, incorporating participatory design, can enhance the learning process. To demonstrate the effectiveness of this approach, the teaching of Graph Theory is used as an example. A participatory design process elicited the features relevant to this game. The game incorporates challenge cards with progressively increasing difficulty levels, facilitating interactive learning and a comprehensive understanding of core graph concepts. It covers a range of graph types, encouraging players to identify and manipulate graphs playfully and enjoyably. The game was evaluated with eighth-grade students to assess its effectiveness and produced promising results, demonstrating the successful acquisition of valuable graph concepts.

Background

This section discusses two works related to the present proposal and introduces the concepts of Computational Thinking, Unplugged Computing, and Participatory Design on which this proposal was developed. Considering that CT is a skill strongly connected to learning mathematic concepts, a short review of this matter is also presented.

Related works

Two studies that utilized games for educational purposes are discussed next. Despite addressing different themes and target audiences, both studies highlight the use of unplugged games as educational resources. Incorporating playful elements and the involvement of learners in the creative process were identified as positive characteristics, serving as a valuable reference for this project.

Grafápagos game

Grafápagos (Moraes, 2016) was developed to aid in teaching graph algorithms to computer science students. The game board represents an archipelago, with each island symbolizing a vertex and the maritime routes between islands representing arcs. Navigator families were created to inhabit the archipelago and must follow the algorithm step-by-step to achieve the game's result. The game establishes connections between the Grafápagos universe and the real world, providing context and explaining these algorithms'background, purpose, creators, and historical significance. Figure 1 presents the game board, with the crests representing the algorithm families in the upper right corner.

In assembling the game map, an archipelago of islands is constructed, with each island representing a vertex and each maritime route between islands representing an arch in the graph. The map aims to demonstrate graph concepts and algorithms like breadth-first

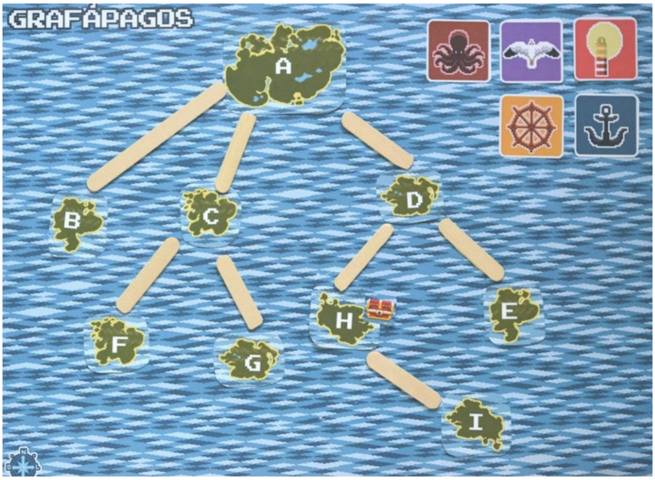

Fig. 1 Game board, with the crests representing the algorithm families

search, depth-first search, Kruskal's algorithm, Prim's algorithm, and network flow algorithms.

Specifically, five characters'families connect the islands according to their specialized algorithm: (i) the Gargoyle family uses breadth-first search to traverse the graph outwards in layers from the starting vertex, (ii) the Profundus family applies a depth-first search to fully explore one branch before backtracking to other branches, (iii) the Kraken family follows Kruskal's algorithm of adding arcs in ascending order of weights, avoiding cycles, (iv) the Prism family uses Prim's algorithm to build a minimum spanning tree by repeatedly adding the lowest-cost arch that connects to a visited vertex, and (v) the Asteroid family constructs a flow network that maximizes throughput from sources to sinks subject to capacity constraints.

As players actively build connections between islands using these algorithms, they gain an intuitive, hands-on understanding of how graph traversal and connectivity algorithms operate, which complements the step-by-step instructions provided in the game. The open-ended nature of map assembly allows the player to exercise logical reasoning in applying algorithms to build a fully connected graph.

The visually appealing design and captivating narrative of Grafápagos make it suitable for the most education levels. The author observed that including a story and visually friendly elements significantly enhanced the game's appeal, expanding its application beyond specific grade levels to include middle school. Consequently, the developed game effectively conveys the concept of graph algorithms (Moraes, 2016).

RoPE (educational programmable robot)

Viana et al. (2021) propose a programmable toy designed to introduce young children to the basics of algorithms and to promote the growth of Computational Thinking skills. The interfaces of programmable toys and details the creation of three programming interfaces for RoPE, an Educational Programmable Robot, are investigated. The first interface is intended for children between the ages of 3 and 6 and features brightly colored physical buttons. The second interface is a smartphone application that allows programming by piecing together blocks, which remain synchronized with the toy's physical buttons. The third interface consists of tangible blocks that facilitate intuitive interaction and foster teamwork among users. A quantitative study indicated that using physical buttons is more effective than assembling blocks on a smartphone screen. The tangible interface was assessed qualitatively through video and content analysis, revealing that children frequently interacted with the tangible blocks, which promoted algorithm debugging and collaboration among peers.

Computational thinking

Computational Thinking is a problem-solving approach that applies computing concepts to analyze and solve real-world problems. Wing (2006) recognizes CT as a valuable skill and argues that it should be integrated into the education of young people.

Developing competence in CT requires acquiring essential skills such as (Csizmadia et al., 2015; Wing, 2006): (i) abstraction, (ii) generalization, (iii) decomposition, (iv) solving problems algorithmically, and *(v)* evaluating and analyzing data. Abstraction allows individuals to filter important information and discard irrelevant details. Generalization develops solutions applicable across multiple situations beyond individual problems.

Decomposition enables breaking down complex problems into smaller, more manageable parts. Solving problems as an algorithm refers to the ability to follow step-by-step instructions, while data evaluation and analysis involve interpreting information for decision-making.

Integrating CT into the school curriculum is increasingly recognized to prepare students to face the challenges of the digital age (Wing, 2006). Papert (1980) has advocated the use of computers to help students understand what they are learning, arguing that computers are not devices that you tell them what to do, but rather objects to think with. Resnick (2017) argued that CT should be introduced to young people early and proposed Scratch, a tool designed to support the development of CT in children.

The growing adoption of CT in Brazilian secondary education highlights its relevance in this country. According to a recent report by the Brazilian Center for Innovation in Education (CIEB, 2019), almost 35% of Brazilian schools have already incorporated some form of CP practice. This demonstrates the growing recognition of the benefits and relevance of CT in the educational system.

These skills are valuable for future technology careers (Shein & Tsai, 2015) and practical everyday applications like route planning and personal budgeting. Furthermore, the unplugged computing approach can teach CT without computers. This approach involves hands-on activities, games, and physical materials to teach computational concepts. By engaging students in interactive and tangible experiences, unplugged computing provides a playful and accessible way to develop CT skills.

A study by Calao et al. (2015) with middle school students demonstrated that training in CT improved math performance, even without a specific focus on the subject. This emphasizes the value of CT as a fundamental skill in the formative years. In summary, integrating CT into the school curriculum brings cognitive and practical benefits to students, providing them with valuable skills for the digital age. It is a growing trend in Brazil and worldwide, reflecting the recognition of the importance of CT for the new generation.

Unplugged computing

Unplugged computing is an educational approach that aims to disseminate knowledge about Computer Science without using computers or digital media (Santos et al., 2016). This approach provides a playful and accessible way of teaching computational concepts through practical activities, games, and physical materials and has been successfully implemented in several countries. Bell et al. (2009) observed significant increases in student engagement and understanding of concepts in New Zealand classrooms using unplugged computing. In Brazil, it has become one of the most adopted tools for developing CT, with approximately 28% of schools incorporating it as of 2016 (Santos et al., 2016).

Unplugged activities can help students learn computing concepts by employing interactive games and physical representations. For example, students can act out by following instructions from an algorithm or create graphs using strings and cardboard. This hands-on approach creates a playful learning environment that enhances enjoyment and comprehension. Studies have shown that unplugged computing improves student engagement, knowledge retention, and interest in computer science (Bell et al., 2009).

Furthermore, the unplugged format eliminates the need for technology access, making it suitable for diverse school contexts and socioeconomic backgrounds. This research highlights how unplugged computing can make early computer science education more equitable, effective, and enjoyable.

Bordini et al. (2016) carried out a systematic literature review involving 60 articles published between 2010 and 2015, in significant events related to Computing Teaching in Brazil. The study revealed that most articles addressed algorithms, programming, and unplugged computing, with 18 works explicitly dedicated to the latter. These studies aimed to disseminate computer science knowledge in elementary and secondary schools through engaging and concrete technology-free activities. Among the unplugged computing-related works, the authors mention: (i) the traditional game Monopoly applied to simulate economic and financial concepts through a playful dynamic of buying and selling properties, (ii) strategy games like Battleship and Rest One employed to enhance logical reasoning and algorithmic thinking, (iii) dedicated options like Circuit Maze and Code Master explicitly designed to introduce programming and CT concepts in a concrete and enjoyable manner.

Bell et al. (2015) propose a comprehensive collection of practical introductory computer science teaching activities. These activities involve games, puzzles, plays, and real-world analogies to illustrate concepts such as algorithms, bits, and Boolean logic. For example, algorithms can be represented by people moving, network concepts can be modeled using strings, and map activities can be employed to teach graph and tree problems.

The wide range of possibilities demonstrates how unplugged activities can make learning computer science more natural and intuitive. The positive outcomes of the analyzed studies further validate the potential of unplugged computing to provide a tangible, engaging, and accessible approach to computer science education for children and youth across different education levels. This teaching method has proven effective in fostering the dissemination of CT.

Participatory design

Participatory Design (PD) is a method to co-create, co-operate, and co-design (Carroll, 1995; Jeffries et al., 2011; Wroblewski, 2003). It is an approach where all stakeholders, including employees, customers, end-users, partners, designers, and researchers, are actively involved throughout the design process. PD exercises are used across several fields, such as software, product design, architecture, and graphic design. The aim is to create products and services that genuinely meet the needs and expectations of users by applying their knowledge and experiences. PD is also known as cooperative, co-, or community design (Schaper et al., 2023). While some methodologies are restricted to specialized professionals, PD methodology seeks to collect and analyze user ideas and needs to develop solutions in collaboration with them (Aytekin & Rızvanoğlu, 2019; Camargo & Fazani, 2014).

Melo et al. (2008) discuss the design of technology aimed at children, emphasizing a process model centered on children's participation. It highlights the importance of involving children in the design process, not just as end users, but as active participants who contribute to the conception and development of systems and virtual environments. The approach combines concepts from PD and Organizational Semiotics to create interfaces and systems that meet children's needs and expectations.

Camargo and Fazani (2014) state that "participatory practices are intended to be applied throughout the entire life cycle of the product, considering the participants involved in this activity". Collaboration with users during development improves product quality. Furthermore, it allows the combination of different knowledge throughout the project. PD brings several benefits to participants: it provides an opportunity for discussion among them, resulting in a high degree of responsibility and commitment to the growth of the product. It also contributes to understanding the business domain (Cherry & Macredie, 1999).

PD actively engages end users through collaboration, co-creation, and iterative testing. This allows designers to gain valuable user insights, feedback, and ownership in developing truly user-centric solutions that are relevant and meet user needs. The approach emphasizes discussion among participants, shared responsibility, and application of user knowledge to improve product quality.

Several design methodologies can apply this collaborative approach, such as:

- Cooperative Design (Ehn & Kyng, 1987) focuses on collaboration between designers and users throughout the design process. Users are seen as experts in their work, contributing domain knowledge. Standard techniques involve storyboarding, rapid prototyping, and participatory workshops to co-create human-centered solutions.
- Contextual Design (Beyer & Holtzblatt, 1998) relies on ethnographic observations of users in their actual work context. Designers spend time immersed in that environment to understand existing needs and practices, guiding the development of solutions that fit within workflow and organizational culture.
- Scenario-based Design (Carroll, 2000) generates concrete representations of future usage contexts using personas and storyboards. This helps users engage in the design process by exploring possibilities and providing contextualized feedback.
- Design Thinking (Brown, 2008) stresses multidisciplinary collaboration, rapid prototyping, and user testing in a non-linear process focused on human-centered solutions that address real user needs.

These methodologies can benefit from the adoption of PD.

Methodology

A variety of methods for PD can be found in the literature, such as Stegner et al. (2023) for designing interactions between older adults and robots or Anderton-Coto et al. (2023) for developing a mobile application for e-democracy involving low-income Latine communities to promote family learning. In this article, we chose to adapt the one applied by Melo et al. (2008) due to the similarity in the application domains, both focused on educational issues. The methodology encompasses four stages (see Fig. 2): (i) *Usage Observation* in the classroom to identify learning problems; (ii) *Clarification of Meanings* involving interviews and PD activities to gather demands for the game; (iii) *Prototyping* with students developing paper prototypes as suggestions for the game; and (iv) *Implementation*, that involves creating the initial prototype and refining it based on student feedback. The internal cycle can be repeated as often as necessary during development, signaling that the designer and participants may need to revisit previously addressed points (Melo et al., 2008).

The first stage, *Usage Observation*, was based on the technique of Muller and Druin (2012). It involves observing students during regular classes to gain insights into their interactions and identify challenges and opportunities to improve the learning process. It involved presenting students with basic concepts about graph theory. The unsystematic observation method (Oliveira, 2009), which does not include prior planning, was applied. Notes, recordings, and photographs were used to analyze this activity. The information

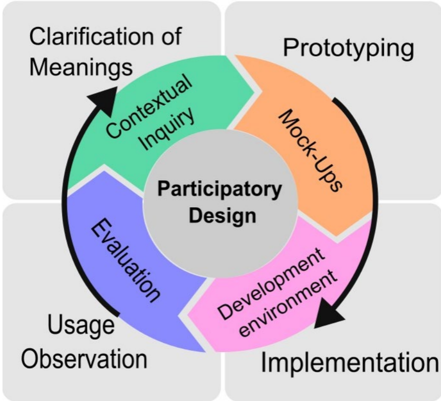

Fig. 2 Participatory design activities

collected at this stage was essential for developing an educational game that effectively supported the didactic content. After completing the first cycle, an evaluation is carried out based on user feedback on the implemented product itself.

The second stage, *Clarification of Meanings*, involves individual and group conversations with students to understand better how collaborative learning occurs in the classroom. It employs interview techniques within the school environment, applying Druin's (1999) and Holtzblatt and Jones's (2007) suggestions. The Contextual Inquiry (CI) technique was used in the Clarification of Meanings stage. CI is a research methodology that allows designers to deeply understand user tasks and work processes (Druin, 1999). CI provides rich insights into user goals, pain points, and needs by directly observing and interacting with users in their natural work context. The interviews aimed to openly gather students'needs, preferences, and suggestions regarding the educational game being developed. Additionally, hands-on PD activities (Muller & Druin, 2012), such as brainstorming and quick prototyping using simple materials, were conducted to engage students in generating creative ideas and ensure that their specific demands were incorporated into the game's development.

The third stage, *Prototyping*, was conducted using the mock-up technique. High-fidelity prototyping can be time-consuming and expensive, so utilizing mock-ups aided in the visualization process of the final prototype, supporting both designers and participants in decision-making. Mock-ups act as experimental models that effectively allow for the visualization and testing of design ideas (Bødker et al., 2000; Brandt et al., 2006). These prototypes represent the students'suggestions regarding the game's dynamics, components, and visual aspects. During participatory sessions, students constructed tangible models to simulate their ideas, performing iterative test refinements, engaging in discussions, and voting on alternatives. This practical approach facilitated the validation of the student's suggestions and the integration of their preferences into the cardboard prototypes, which served as a foundation for guiding the implementation phase of the game.

The fourth stage, *Implementation*, involves developing a tangible game that delivers the board game based on user requests identified in the previous steps. Mock-up models supported product creation and proved highly valuable during implementation. The implementation process was divided into two phases: *(i)* the development of a functional prototype of the educational game using resources such as 3D printing (additive technology) and *(ii)* improvements based on student feedback after testing the initial prototype. The main objective was to guarantee the concrete development of the game, maintaining continuous alignment with the needs and interests expressed by the students and fulfilling the defined educational objectives. Several testing cycles and refinements were conducted to improve understanding of the rules, balance challenge and learning, and engage students.

Development of *GraphGame*

This section presents the development of a board game with 8 th-year students at the "Escola de Aplicação"[1] Univali, Santa Catarina state, Brazil. PD methodology was applied, and students were involved in creating the game. Eight (four girls and four boys) of the 25 students (12 girls and 13 boys) voluntarily participated in extracurricular activities held half a day a week. Participating students were selected due to their strong command of the teaching material and enthusiasm for the project.

Usage observation

The main objective of this step is to obtain a broad understanding of the context of the game. Therefore, meetings were held with students to introduce them to the problem-based approach and collaboratively develop an educational game focused on graphs. Problems were presented as examples to promote a deeper understanding of the subject. To this end, two 4-h meetings were held with the students. In the first, an introductory lecture was given on graph concepts; in the second, board game models were presented for students to use as a basis for creating their games.

The first meeting began with an expository and interactive lecture covering simple and thoughtful graphs, arcs, and vertices. In addition, practical challenges were presented using Eulerian, Hamiltonian, shortest-paths, and cycles concepts. In the case of Eulerian graphs, it was proposed to draw a house without lifting the pencil. For Hamiltonian graphs, the traveling salesman problem and mail delivery were discussed, which consisted of traveling a circular route passing through each vertex only once. The shortest path problem, which seeks the lowest cost path between two vertices, was illustrated by Dijkstra's algorithm. The Muddy City exercise (Bell et al., 2015) was applied as a practical activity, where students were asked to suggest paving the streets so that it would be possible to go from any house to any other using only paved roads.

After the first two 4-h meetings, weekly 2-h meetings were held. At these meetings, additional details about the graph were presented, and the active participation of students in the game development process was encouraged. Creativity and collaboration among students were stimulated, and at the end of each meeting, the groups' progress was evaluated, and feedback was provided. Additionally, students were encouraged to continue working on the game outside of class. The weekly meetings aimed to introduce students to the problematizing approach, present graph concepts, and encourage students' active participation in the collaborative creation of the educational board game.

[1] The expression"Escola de Aplicação"in Portuguese can be translated as "School of Application" and in Brazil it refers to a school generally linked to a higher education institution, where future teachers carry out their practical training.

Figure 3 shows students in two classroom activities: on the left, they are practicing a graph algorithm, and on the second, they are using the Circuit Maze game algorithm. The current market offers various board games that can be played individually or with family and friends. Prominent examples include Monopoly, Battleship, Uno, Circuit Maze, and Code Master. These games were made available to students and teachers for use in the classroom, and each of them was presented and explored during the meetings held in this phase. By incorporating board games into the learning process, researchers sought to harness their potential to improve CT skills and promote a collaborative and engaging educational experience.

Clarification of meanings

In the *Clarification Meaning* stage, concepts, terms, and elements for the project are defined and clarified (Spinuzzi, 2005). The clarification provided a solid foundation for creating the game, making it challenging and fun. It also made identifying students'difficulties with graphs and games possible, enabling more effective guidance (Muller & Druin, 2012). Resources such as verbal communication, notes on a board, and a simple physical graphic model with popsicle sticks and Styrofoam balls were used to facilitate visualization and understanding of the abstract concept (Beyer & Holtzblatt, 1998). The CI technique was applied, observing and interviewing students in their natural school study environment to understand their needs and expectations about the board game. This approach made it possible to identify the students'main difficulties and develop a graphic board game that met the needs and interests of the participants.

The students wanted to create a game with cards of varying difficulty levels so the player could perform the proposed challenge with some guidance described on the card. Each challenge could cover a specific graph algorithm, such as finding the minimum path and identifying if the graph is Hamiltonian or Eulerian. Inspired by the Circuit Maze game, they requested that upon completing each challenge in the game, the player should press a physical button to validate whether the solution drawn on the board would be correct, recording the time spent. Thus, the opponent could make the same move, and participants would identify who found the solution fastest, determining the round winner. Figure 4 shows students creating simple physical models during one of the classroom sessions.

The following concrete elements compose the game: (i) Board: has dimensions of 35 ×35 cm, with a grid of 15 lines and 15 columns, totaling 225 circular holes; (ii) Vertices: represented as colored plastic pins that fit firmly into the circular holes of the board; (iii) Arches: to represent the arcs, a thin retractable card was inserted into each pin/vertex; (iv)

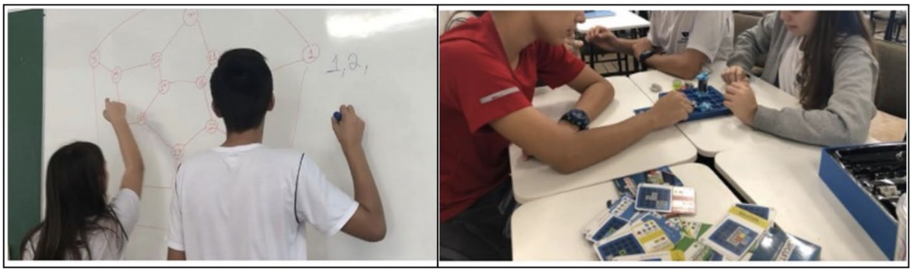

Fig. 3 Students learning about graphs and applying algorithms in the Circuit Maze game

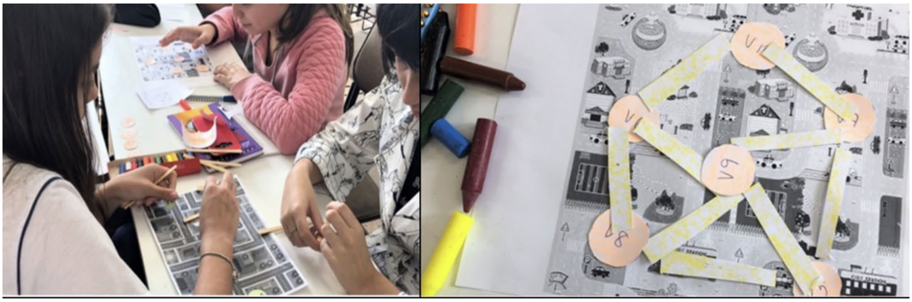

Fig. 4 Students creating the game with the physical model on the right

Scenarios: sheets of paper with various illustrations to be placed on the board; (v) Challenge cards: present the specific problem and the elements necessary to design a solution on the board; indicate which scenario should be used and define the maximum number of vertices allowed in the solution.

Prototyping

The prototype has a paper scenario base attached to a translucent perforated board. The board design and vertex representation are shown in Fig. 5.

During the prototyping phase, in addition to building the board, two challenge cards were created to present specific problems for players to solve (Fig. 6). Each card is associated with a scenario, specifying the maximum number of vertices that can be used in the solution. In some problems, the scenario also indicates the location of the arcs. Scenarios illustrate the involved context, like a city map, and specify elements that must be considered, such as mandatory crossing points.

In this phase specific tests were driven with the same set of students involved in PD for the sake of game usability, the resistance of the material used (wooden/acrylic). During testing, several issues arose, including players needing help understanding the game's rules and instructions more clearly. Based on this finding, the rules were simplified, and a more concise and objective manual was created. Additionally, improvements were made to the game model, arcs, and board layout, resolving some identified issues.

The card on the left presents a Hamiltonian graph challenge, which involves finding a circular path through all the vertices while visiting each vertex only once—modeling

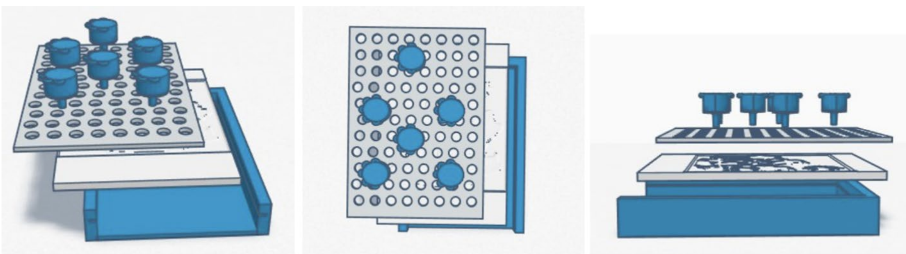

Fig. 5 Board design and vertex representation

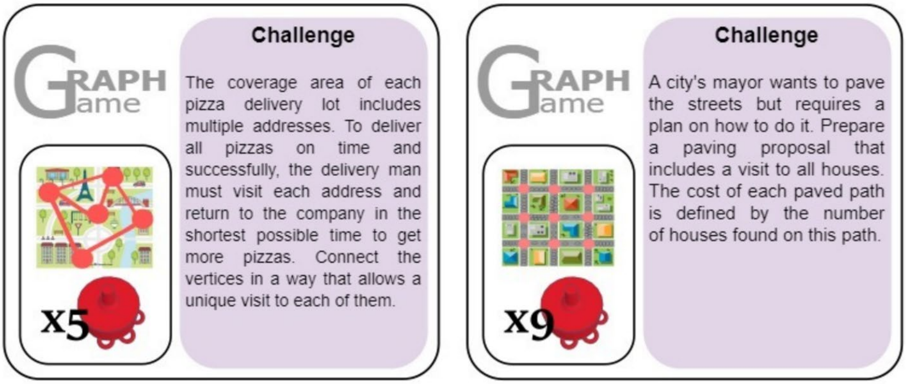

Fig. 6 Example of game challenge cards created in the prototyping phase

the traveling salesman problem. The card on the right presents the shortest path challenge, solved using Dijkstra's algorithm to identify the shortest route between two vertices. These cards guide players in selecting and planning solutions on the assembled game board. Players can participate in the game flow and problem-solving process with the board and a challenge card.

Implementation

This is the most time-consuming stage of the development process and involves constructing the real board game and components based on the prototype. It comprises two phases: (i) the first phase was focused on creating the game board, cards, pieces, and packaging based directly on the paper prototype, and (ii) the second one was dedicated to refining the game's visual design, component quality, and overall play experience. After each phase, an evaluation was carried out with an independent group of students.

Phase 1

During this phase, extensive hours of work were dedicated to 3D modeling and printing, laser cutting, and meticulous finishing cards, markers, and structural components. The game is a board that presents challenge cards and scenarios with different difficulty levels, aiming to teach different graph concepts through problems, puzzles, and riddles. These cards cover directed and weighted Eulerian graphs.

The manufacturing process involved multiple attempts, errors, and adaptations in the chosen materials, stressing the iterative nature of PD. A cardboard was used to display the challenge, and wood was used for the first board version (Fig. 7). In this phase, acrylic was disregarded due to the difficulty of machining.

Two wooden boards were constructed for evaluation purposes. The two initial challenge cards and their respective boards were presented during the fourth meeting for a preliminary assessment of the students. Participants enthusiastically accepted their first interaction with the game, understood the challenge cards, and carried out the proposed activities.

The tests performed in the prototyping phase revealed the need to improve the rules, manual, arch design, and board layout. Further rounds of testing were carried out to

Fig. 7 Students creating challenge proposals for the game

understand student needs better and identify improvements that would improve the overall quality of *GraphGame*. Despite the challenges encountered during the process, the result was an educational game aligned with the intended objectives.

Phase 2

This phase incorporated the enhancements identified during the preliminary assessment with students. With their participation, two new challenges were designed, each with corresponding scenarios and cards. To denote difficulty levels, the challenge cards and scenarios were color-coded. The board was upgraded with an acrylic addition to improve gameplay.

Considering the cards created in Phase 1, the game has a set of 4 challenges in this phase. The vertices were fixed on the board, and the thin retractable cards and the challenge card appeared on the right side of the board. Each challenge consists of a card and its respective scenario on the board. The card features a description of the challenge and a concise explanation of the solution. In addition, explanatory plaques were created for each model, positioned behind the corresponding scenarios on the board. This way, players can consult the boards if they have any questions without needing a manual. Figure 8 shows the acrylic plate superimposed on the scenarios and the points designated for placing the vertices.

A second evaluation occurred at the fifth and final meeting, involving eighth-grade students from two classes (A and B) at the"Escola de Aplicação"the same institution as the students who developed the game. The students in class A needed help understanding the challenges and assistance from the students involved in the development process. On the other hand, students in class B demonstrated greater ease in understanding the challenges and could carry out all the tasks on the board more efficiently. This difference is probably because these classes study in different shifts, morning and evening, or due to differences in age groups. Due to the time constraints of two 45-min CT classes, students played the game in groups, forming one group per class.

Evaluation and results

To evaluate its effectiveness in providing participants with a basic understanding of graph concepts, *GraphGame* was subjected to a practical two-hour trial with students who did not participate in the PD (Fig. 9). Twelve students played the game and

Fig. 8 Acrylic boards with scenarios and challenge cards

Fig. 9 Students evaluating the latest version of the game

answered five questions about the game's challenges. Students were enabled to use the challenge card, set the board, and perform the play. The first question asked students to draw a graph to assess their ability to represent the structure. The second question required identifying the elements of a given graph, such as vertices and arcs. Questions 3 and 4 focused on algorithms: (i) the third question asked students to determine the shortest path in a graph, and (ii) the fourth required them to identify whether a presented graph was Hamiltonian. Finally, the fifth question was discursive, asking students to explain the concept of a graph in their own words.

Eight students (66.3%) answered all the questions correctly, while 4 (33.3%) gave partially correct answers. The answers to question 5 further support these conclusions. Table 1 shows a selection of answers given by some of the students, representing the sample of 12 students.

The answers to the discursive question highlight that most students incorporated fundamental concepts related to graphs, such as vertices and arcs (mentioned by students 1 and 2). Furthermore, some students correctly referred to the structure of a graph (students 2, 4, and 5). Although not all students used the specific terminology introduced in the game, their explanations demonstrated a clear understanding of the purpose and function of a graph (students 3, 4, and 5). In summary, the evaluation results provide evidence that *GraphGame* effectively achieved its goal of conveying essential introductory concepts of graph theory. Most students demonstrated an understanding of the basic principles in objective and subjective questions, confirming that the game is an effective and engaging tool for learning.

Conclusions and future works

This study aimed to develop an unplugged game to assist in learning graph concepts. The PD process was carried out with the students, allowing them to contribute ideas and suggestions for the game. Before the development stage, students completed introductory activities about graphs and gameplay, which prepared them for the creative process.

The game requirements defined during the PD process were: (i) to allow individual play, (ii) presentation in table format, (iii) allowing the representation of real-world scenarios, and (iv) challenge cards can be used with difficulty levels covering different problems. The practical and engaged participation of students involved in all phases of development applying PD was demonstrated in"Computational thinking"section. It clarifies the correct choice of PD methodology, which contributed significantly to creating and customizing the game. The evaluation described in"Unplugged computing"section demonstrated that *GraphGame* successfully achieved its goal of helping students learn graph concepts.

Existing research in the literature on the unplugged game for teaching graph-related algorithms found the game Grafápagos (Moraes, 2016) as the only resource. However, Grafápagos only allows different possibilities within a fixed scenario since real archipelagos have fixed locations. In contrast, *GraphGame* introduces challenge cards into arbitrary scenarios, providing unlimited opportunities for use and application. This allows you to resolve a broader range of cases compared to Grafápagos.

Table 1 Samples of student answers to the discursive question asking them to explain graphs in their own words

Student	Answer: explain graphs in their own words
1	It is a set of vertices, arcs, and cycles
2	It is a set of vertices and arcs. Each arc connects a pair of vertices, but a vertex can have more than one connection with different arcs
3	It is a drawing or structure that has the option of one or more different paths to the destination
4	They are a set of places being interconnected between some points and numbers
5	Graphs are a set of schematized vertices, interconnected by arcs, which trace different paths

GraphGame can still receive improvements, such as (i) the creation of new scenarios and challenges, enabling the learning of more types of algorithms and expanding the learning experience; (ii) refinement of the representation of the arcs, which was customized with retractable cards, and (iii) adjustment of the upper and lower parts of the board instead of screws.

This study opens possibilities for new research, such as investigating the effects of the game on students'engagement and motivation in learning graphs. Continuous expansion and improvement can maximize the project's educational potential.

Funding The work was supported by Fundação de Apoio à Pesquisa do Distrito Federal, 543/2022, Edilson Ferneda.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

Anderson-Coto, M. J., Salazar, J., Sedas, R. M., Campos, F., Bustamante, A. S., & Ahn, J. (2023). Toward culturally sustaining design: Centering community's voices for learning through participatory design. *International Journal of Child-Computer Interaction*. https://doi.org/10.1016/j.ijcci.2023.100621

Aytekin, B. A., & Rızvanoğlu, K. (2019). Creating learning bridges through participatory design and technology to achieve sustainability in local crafts: A participatory model to enable the transfer of tacit knowledge and experience between the traditional craftsmanship and academic education. *International Journal of Technology and Design Education, 29*, 603–632. https://doi.org/10.1007/s10798-018-9454-3

Bell, T., Alexander, J., Freeman, I., & Grimley, M. (2009). Computer science unplugged: School students doing natural computing without computers. *The New Zealand Journal of Applied Computing and Information Technology, 13*(1), 20–29.

Bell, T., Witten, I. H., Fellows, M. (2015). *Computer science unplugged: an enrichment and extension programme for primary-aged students.* Retrieved from https://classic.csunplugged.org/documents/books/english/CSUnplugged_OS_2015_v3.1.pdf

Beyer, H., & Holtzblatt, K. (1998). *Contextual design: Defining customer-centered systems*. Morgan Kaufmann Publishers.

Bødker, S., Ehn, P., Kammersgaard, J., Kyng, M., Sundblad, Y. (2000). Co-operative design—perspectives on 20 years with 'the Scandinavian IT design model. *Proceedings of the 1st Nordic Conference on Human-Computer Interaction* (NordiCHI '00). Stockholm, pp. 1–10.

Bordini, A., Avila, C. M. O., Weisshahn, Y., Cunha, M. M., Cavalheiro, S. A. C., Foss, L., Aguiar, M. S., & Reiser, R. H. S. (2016). Computing in basic education in Brazil: The state of the art. *Revista De Informática Teórica e Aplicada, 23*(2), 210–238. https://doi.org/10.22456/2175-2745.64431

Brackmann, C., Román-Gonzáles, M., Robles, G., Moreno-Leon, J., Casali, A., Barone, D. A. C. (2017). Development of computational thinking skills through unplugged activities in primary school. *Proceedings of the XII Workshop in Primary and Secondary Computing Education* (WiPSCE). ACM Press. p. 7.

Brandt, E., Messeter, J., Binder, T. (2006). Participatory design in an era of participation. *Proceedings of the 9th Participatory Design Conference* (PDC '06). ACM, pp. 81–84. https://doi.org/10.1145/1147261.1147276

Brown, T. (2008). Design thinking. *Harvard Business Review, 86*(6), 84–92.

Calao, J., Carrillo, J., & Ruiz, J. (2015). Computational thinking in primary education: A study on the development of computational thinking skills and its impact on mathematical creativity. *Computers & Education, 82*, 263–273.

Camargo, L. S. A., & Fazani, A. J. (2014). Exploring participatory design as an information systems development practice. *Revista De Ciência Da Informação e Documentação, 5*(1), 138–150. https://doi.org/10.11606/issn.2178-2075.v5i1p138-150

Carroll, J. M. (1995). Participatory design: Principles and practices. *ACM Transactions on Office Information Systems, 1*(4), 320–328. https://doi.org/10.1145/192162.192168

Carroll, J. M. (2000). *Making use: Scenario-based design of human-computer interactions*. MIT Press.

Cherry, C., & Macredie, R. D. (1999). The importance of context in information system design: An assessment of participatory design. *Requirements Engineering, 4*, 103–114. https://doi.org/10.1007/s007660050017

CIEB—Innovation Center for Brazilian Education. (2019). *Reference curriculum in technology and computing*. CIEB.

Costa Júnior, A., Honda, F., Fernandes, L., & Vieira, N. (2022). Computational thinking in computing teaching in schools: An experience report of internship in degree in computing in public schools. *Proceedings of the Computer Workshop at School*. pp. 47–58. https://doi.org/10.5753/wie.2022.224748

Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., Ng, T., Selby, C., & Woollard, J. (2015). *Computational thinking—a guide for teachers*. Computing at School.

Druin, A. (1999). The role of children in the design of new technology. *Behavior & Information Technology, 18*(4), 231–240. https://doi.org/10.1080/014492901101086659

Ehn, P., & Kyng, M. (1987). The collective resource approach to systems design. In G. Bjerknes, P. Ehn, & M. Kyng (Eds.), *Computers and democracy—a scandinavian challenge.* Gower Pub Co.

Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. *Educational Researcher, 42*(1), 38–43.

Holtzblatt, K., & Jones, S. (2007). *Contextual inquiry: A participatory design process*. Morgan Kaufmann.

Jeffries, R., Dix, A., Cockton, G., & Davies, C. (2011). *Participatory design: Theory and practice*. Cambridge University Press.

Kazimoglu, C., Kiernan, M., Bacon, L., & MacKinnon, L. (2012). Learning programming at the computational thinking level via digital gameplay. *Procedia Computer Science, 9*, 522–531.

Melo, A. M., Baranauskas, M. C. C., & Soares, S. C. M. (2008). Design with children: From practice to a process model. *Brazilian Journal of Informatics in Education, 16*(1), 43–55.

Moraes, B. N. (2016). *Grafápagos: An unplugged tool to help teach graphs and graph algorithms in basic education*. Federal Technological University of Paraná.

Muller, M. J., & Druin, A. (2012). Participatory design: The third space in human-computer interaction. *Human-Computer Interaction, 27*(4), 123–156.

Oliveira, G. A. (2009) . Métodos e Técnicas - Ferramentas de Pesquisa. Ergonomia Informacional Na Travessia De Pedestre. Rio de Janeiro: Maxwell 0510349/ca, 2009. p. 1–52.

Papert, S. A. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books.

Resnick, M. (2017). *Lifelong kindergarten: Cultivating creativity through projects, passions, peers, and play*. MIT Press.

Santos, E. R., Soares, G., Bianco, G. D., Rocha Filho, J. B., & Lahm, R. A. (2016). Stimulating computational thinking through unplugged computing: A proposal for early childhood education. *Latin American Journal of Educational Technology, 15*(3), 1–14.

Sapounidis, T., Demetriadis, S., & Stamelos, L. (2019). Evaluating children's performance with graphical and tangible robot programming tools. *Personal and Ubiquitous Computing, 23*(1), 87–101.

SBC—the Brazilian computing society. (2023). *Biennium action plan 2023–2025*. SBC. Retrieved from https://www.sbc.org.br/institucional-3/eleicoes/plano-de-acao

Schaper, M.-M., Smith, R. C., van Mechelen, M., Tamashiro, M. A., & Iversen, O. S. (2023). Co-designing sustainable practices for emerging technologies education. *International Journal of Technology and Design Education*. https://doi.org/10.1007/s10798-023-09857-3

Shein, P., & Tsai, T.-H. (2015). Computational thinking in elementary and secondary teacher education. *Journal of Educational Technology & Society, 18*(1), 352–365.

Spinuzzi, C. (2005). The meaning of participation in participatory design. *Journal of the Association for Information Systems, 6*(1), 1–26.

Stegner, L., Senft, E. B., & Mutlu, Y. (2023). Situated participatory design: a method for in situ design of robotic interaction with older adults. *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. https://doi.org/10.1145/3544548.3580893

Viana, C. P., Raabe, A., & Viana, C. P. (2021). Designing programmable Toy's interfaces for small children. *Estudos Em Design, 29*(1), 82–97. https://doi.org/10.35522/eed.v29i1.1150

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

Wroblewski, D. M. (2003). *Democratizing design: Principles and practices of participatory design*. Addison-Wesley.

Ana Paula da Silva is a Computer Scientist from the University of Itajaí Valley and is currently pursuing a Master's in Applied Computing at the same institution. Her professional experience includes roles as Team Development Leader at Senior Sistemas (2022-2024) and System Analyst at both Senior Sistemas and Portobello Group (2016-2021). She has expertise in agile methodologies, PL/SQL, and Oracle EBS. Ana Paula is also certified in Agile Development and Remote Leadership.

Prof. Aluizio Haendchen Filho holds a PhD. in Computer Science from the Pontifical Catholic University of Rio de Janeiro and a Master's degree in Computer Science from the Federal University of Rio Grande do Sul. He is an independent consultant and researcher. His expertise includes software engineering, artificial intelligence, focusing on natural language processing, machine learning, and intelligent software agents.

Prof. Hércules Prado develops knowledge-based systems for organizations and conducts research in knowledge management, organizational learning, educational technology, competitive intelligence, and futures studies. He holds a D.Sc. in Computer Science (Federal University of Rio Grande do Sul), a Master's in Systems and Computing Engineering (Federal University of Rio de Janeiro), and a degree in Data Processing (Federal University of São Carlos). He has been a faculty member at the Catholic University of Brasília since 1992.

Prof. Edilson Ferneda holds a degree in Computing Technology from the Aeronautics Institute of Technology (Instituto Tecnológico de Aeronáutica, Brazil), a Master's in Systems and Computing from the Federal University of Paraíba, and a Ph.D. in Computer Science from the Laboratory of Informatics, Robotics, and Microelectronics of Montpellier (France). A former professor at the Federal University of Campina Grande, he has been a full professor at the Catholic University of Brasília since 2001. His interests include artificial intelligence, knowledge management, and consulting on government projects.

Prof. André Luís Alice Raabe holds a Ph.D. in Informatics in Education from the Federal University of Rio Grande do Sul (Brazil) with postdoctoral research at Stanford University. He is a productivity fellow in Technological Development and Innovative Extension and holds a Master's in Computer Science from the Pontifical Catholic University of Rio Grande do Sul and a degree in Informatics. He is a professor and researcher at the University of the Itajaí Valley, coordinating the Laboratory for Technological Innovation in Education.