Evolution of SofEMN: prototyping a module for teaching numerical methods

Evolução do SofEMN: prototipação de um módulo para o ensino de métodos numéricos

Aline Brum Loreto[I] , **Arthur Cordeiro Andrade**[I], **Ana Luisa Soubhia**[I],
Amanda Spolaor[II], **Guilherme de Lima de Menezes**[II],
Bruna de Abreu Dias[II] , **Alice Fonseca Finger**[II]

[I] Universidade Federal de Santa Maria, Cachoeira do Sul, RS, Brazil
[II] Universidade Federal do Pampa, Alegrete, RS, Brazil

ABSTRACT

Motivating teaching and learning strategies for students, particularly with mathematical content, are deemed essential. The SofEMN tool was developed to assist in learning Numerical Methods. In the first version, usability, ease of use, interface, and intended use were evaluated by students, and improvements in the software were proposed. The prototype for the third version of SofEMN is presented in this work, which was designed based on the functional requirements for creating a teaching module that contains explanations and examples of the contents and activities to assess learning. Therefore, content can be learned by the student, and the results can be checked using the same tool.

Keywords: Numerical methods; Teaching; Technology in education; Teaching support tool

RESUMO

Estratégias de ensino e aprendizagem que motivem os alunos são essenciais em sala de aula, principalmente com conteúdos matemáticos. O Software de Apoio ao Ensino de Métodos Numéricos, SofEMN, foi desenvolvido com o objetivo de auxiliar no aprendizado de Métodos Numéricos. Na primeira versão, a usabilidade, a facilidade de uso, a interface e a pretensão de uso foram avaliadas pelos alunos, e melhorias foram inseridas. Este trabalho apresenta o protótipo para a terceira versão do SofEMN, que foi projetado a partir de requisitos funcionais elencados para a criação de um módulo de ensino, contendo explicação e exemplos dos conteúdos, além de atividades para avaliar o aprendizado. Assim, é possível que o aluno aprenda o conteúdo e verifique os resultados na mesma ferramenta.

Palavras-chave: Métodos numéricos; Ensino; Tecnologia na educação; Ferramenta de apoio ao ensino

1 INTRODUCTION

Mathematics is one of the great villains in education, being considered, by some, as a subject with complex content at all levels of education (Tatto; Scapin, 2004). This difficulty tends to evolve according to the demand for mastering more abstract concepts that increase throughout the student's school development.

In higher education, some areas such as Science, Mathematics, and Computer Science have many mathematics concepts present in their curricula, and this, according to Hoed (2016), is one of the several factors responsible for the high dropout rates of the courses, as students bring knowledge insufficient due to learning difficulties faced in primary education (Raeder; Py; Rigo; Pinheiro, 2016). Therefore, it is necessary to search for alternatives to reduce these reflexes. One of the possible paths for this may be the use of Information and Communication Technologies (ICT) in teaching since technology and education can go hand in hand (Kenski, 2007).

In order to have a new tool, in 2020, the development of SofEMN began, a software developed in Python that allows the use of the Numerical Methods studied in the classroom, in which, from the selection of the method and data information input, the software returns the result. As a complement, a SofEMN teaching module was created, built from the usability assessment of the first version and the elicitation of pedagogical requirements. In the teaching module, the student can learn the content of numerical methods, allowing the visualization of videos and images and providing exercises for fixing the content.

In this work, we show the results of the usability evaluation of the second version of SofEMN, compared with the results of the first version. The development of a third version was motivated by the evaluation of the second version. In the requirements survey stage, the functional requirements for a teaching module, which still needed to be created in the tool, were described. Next, we the screen prototyping stage, allowing the software to be evaluated and validated before its implementation. The main contribution of the third version is the creation of the "Learning Content" module. In

addition, we adjustments to the existing module called "Verifying Learning."

The work is divided as follows: Section 2 describes the existing mathematical tools in the literature; Section 3 exposes the steps involved in the development of the work; in Section 4, we show quantitative and qualitative data from the evaluation of the second version of SofEMN; Section 5 specifies the listed requirements for the development of the new module; Section 6 explains the software developed for the third version of the software. Finally, Section 7 presents the final considerations.

2 RELATED WORKS

Through the work of Abreu and Gonçalves (2017), Silva (2013) and Henning *et al*. (2016) we note that teachers use different software to support the teaching of numerical methods, including Scilab[1], Maple[2], Matlab[3], Maxima[4], Mathematica[5], and GNU Octave[6]. Some of these software allows the use of different mathematical contents and are of more general use, requiring the implementation and adaptation of resources to apply numerical methods.

Ad hoc Google Scholar search on the topic 'tools for teaching Numerical Methods' resulted in the VCN-Visual Numerical Calculation software developed by professors at PUC-Minas, which features a direct numerical input interface. VCN-Visual Numerical Calculation has resources that allow students to use tools to find answers to mathematical problems more simply and quickly.

There is also SageMath[7], similar to the software mentioned above, used as a tool for calculating and solving exercises but differs due to its potential in teaching numerical methods. The software is accessible online and has ready-to-use features (Abreu; Gonçalves, 2017).

[1] https://www.scilab.org/
[2] https://www.maplesoft.com/products/Maple/
[3] https://www.mathworks.com/products/matlab.html
[4] https://maxima.sourceforge.io/pt/index.html
[5] https://www.wolfram.com/mathematica/
[6] https://www.gnu.org/software/octave/index
[7] https://www.sagemath.org/

According to Barbosa, Arimoto, Isotani and Maldonado (2015), there is still a strong dependence on the use of software and proprietary educational content in Computing and other areas of knowledge. Restrictions imposed by this software prevent software evolutions from constantly coinciding as educational content evolves, in addition to technical legal and price restrictions. Still, over time, software versions and proprietary content end up being discontinued, becoming obsolete and unsuitable for current use (Barbosa; Arimoto; Isotani; Maldonado, 2015).

Thus, knowing the importance of spreading the culture of free software more and more, Table 1 highlights each software mentioned above and the type of license each one has.

Unlike the previously mentioned mathematical software, the SofEMN software (Finger; Loreto; Soubhia; Andrade, 2021) appears to aid in teaching numerical methods, both as a tool for demonstrating exercise resolutions by teachers and verifying activities by students.

Table 1 – Software License

Software	Free	Owner
matlab		x
scilab	x	
maple		x
maximum	x	
Mathematica		x
GNU Octave	x	
VCN-Visual Numerical Calculation	x	
SageMath	x	

Source: Organization of the authors

In the first version, the software allows students to use it to check the answers to the exercises. This version was evaluated by users for usability, resulting in a high

agreement on the ease of learning to use the software and its usefulness in carrying out activities (Finger; Loreto; Soubhia; Andrade, 2021). This work presents the results of the usability evaluation of the second version compared with the results of the first. From the evaluation of the second version, the development of a third version was motivated, following software development steps. In the requirements gathering stage, functional requirements were described for a teaching module, which did not exist in the tool. Afterwards, the screen prototyping stage was carried out, allowing the software to be validated and evaluated before its implementation. One of the main contributions of the third prototype version is the division of the tool into two modules, one called "Verifying Learning", and the creation of a new module called "Learning Content". In this new module, it will be possible to learn the numerical methods, even before verifying the learning, which is the main differential of the software under development.

3 DEVELOPMENT

SofEMN[8] includes, on a single platform, the numerical methods studied in the Numerical and Computational Methods discipline at the Federal University of Santa Maria (UFSM), Cachoeira do Sul campus, with the aim of being a tool to help teachers and students in teaching numerical methods (Finger; Loreto; Soubhia; Andrade, 2021). The first version provides a module that allows the student to interact with a calculator, in which he chooses the numerical method to be calculated, informs the values of each variable, and the software displays the result on the screen. Additionally, there is the possibility to view your graph. The initial screen of the first version of SoFEMN can be seen in Figure 1.
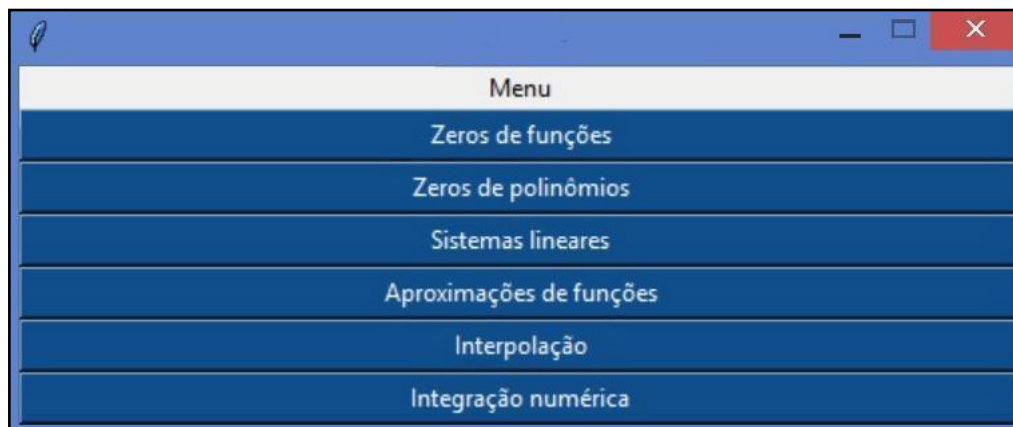
Then, in Figure 2, the data entries of the first version of the software are displayed. After applying a usability evaluation, focusing on evaluating the ease of use, the usefulness of the software, interface and teaching in the classroom, several

---

[8] https://github.com/ArthurCoAnd/MetodosNumericos

suggestions were collected and a new version was developed. The initial screen of the second version of SofEMN can be seen in Figure 3.

Figure 1 – Initial screen of the first version of SofEMN

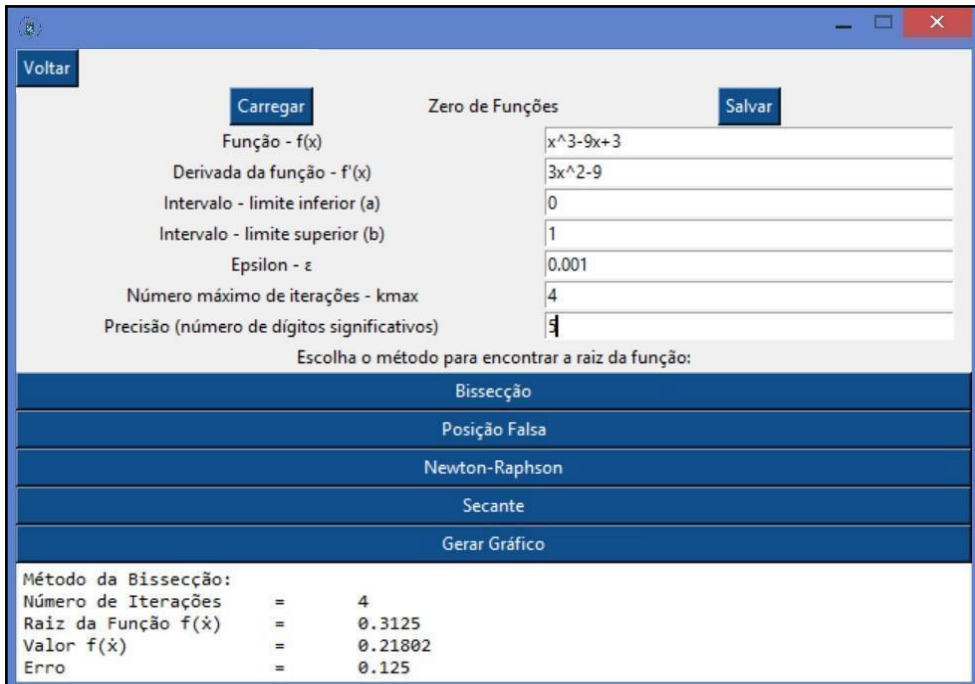Source: Authors (2022)

Caption: Screen of the first version

In the second version of SofEMN, modifications were made to the data entry screen, as shown in Figure 4.

In Figures 1 and 3, it is possible to notice a significant change in the layout of the main menu. Colors and buttons were modified to meet suggestions for improvements made by the participants of the first test (Finger; Loreto; Soubhia; Andrade, 2021). Figures 2 and 4 show that the screen for entering data regarding the selected method was better designed, with a more uniform distribution of each variable to be entered, as well as darker colors were chosen.

From the evaluation of the second version, some stages of software development, previously not performed, were elaborated so that it would be possible to develop a third version with the inclusion of a module for teaching.

The steps involved in this work are Evaluation, Requirements Gathering, and Prototyping. In the Evaluation stage, we conducted a usability test (Valentim *et al*., 2014) to validate the functionalities already available and improved in the second version of SofEMN.
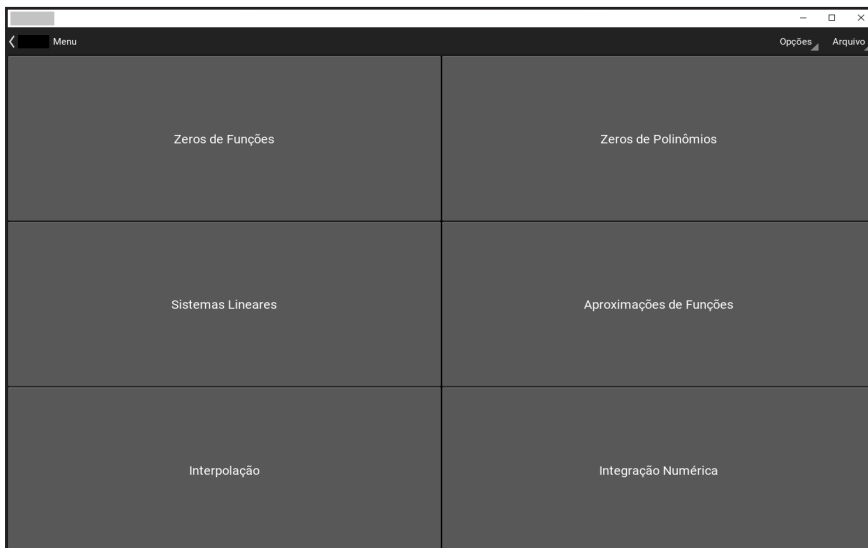
Figure 2 – Entry screen of the first version of SofEMN

Source: Authors (2022)
Caption: Entry screen of the first developed version

Figure 3 – Initial screen of the second version of SofEMN

Source: Authors (2022)

Figure 4 – Second version data entry screen

Source: Authors (2022)

Caption: Data entry screen of the second version

From the evaluation of the second version, some stages of software development, previously not performed, were elaborated so that it would be possible to develop a third version with the inclusion of a module for teaching.

The steps involved in this work are Evaluation, Requirements Gathering, and Prototyping. In the Evaluation stage, we conducted a usability test (Valentim *et al.*, 2014) to validate the functionalities already available and improved in the second version of SofEMN.

The goal of the test is described according to the GQM (Goal-Question-Metric) (Basili; Rombach, 1988) and consists of analyzing the usability of the software to evaluate it, in the context of use by higher education students, regarding the perception of ease of use, perceived usefulness, interface and teaching in the classroom.

To guide the test, a scenario[9] was created with steps to be followed, which include the use of features that had not yet been used in the first version of the software. We developed the test with the same methodology in the previous version, that is, the same questions were applied. The questions present in the evaluation form are presented in Table 2.

---

[9] https://bit.ly/cenario2SofEMN

The test took place in August 2021, remotely through the Google Meet platform, during classes in the Numerical and Computational Methods discipline at the Federal University of Santa Maria/UFSM, Cachoeira do Sul Campus. It had the participation of the same students who evaluated the first version of the tool, with the aim to collect their perceptions on the improvements made. The subject's professors instructed users to follow the activities described in the scenario and respond to the proposed questionnaire.

Table 2 – Questions to assess the perception of SofEMN software users

	A - Ease of use
1	It was easy to learn to use the software
2	I could understand what happened while using the software
3	It was easy to gain usage skills while performing activities in the software
4	It's easy to remember how to use the software
5	I find the software easy to use
	B - Utility of the software
1	I find the software useful to improve my learning
2	I believe that the software would improve my productivity in carrying out activities and learning
3	I believe that the software would facilitate the performance of my activities
	C - Software interface
1	I find the software's colors and buttons pleasing
2	I can clearly see all the buttons and information within the software
3	I easily understand the words, nomenclature and icons of the software
4	Images and icons in the software are easy to recognize
5	I can see all the features of the software
6	I can navigate well through all the software screens
	D - Teaching in the classroom
1	After getting used to the software I think the software facilitates teaching and learning
2	With the software I think the classes would be more interesting
3	With the software I could explore the available materials well
4	With the software I would have greater control over the activities in the classroom
5	With the software the level of learning would increase

Source: Authors (2022)

After analyzing the evaluation results, in the second stage of the work, **Requirements Gathering,** the functional requirements considered essential for teaching software were identified. Finally, in the last stage, called **Prototyping**, a medium-fidelity prototype was developed for the development of the third version of the software.

4 SOFEMN ACCEPTANCE ASSESSMENT

Usability is essential for developing a system that aims to verify information about acceptance and interactive applications. Therefore, we should perform usability tests to collect information about possible usability problems of the tool (Valentim *et al.*, 2014). The design process must carry out the evaluation, and project improvements must use its results (Rocha; Baranauskas, 2003).

We propose a usability evaluation for the second version that aims to observe users' perception of ease of use, perceived usefulness, interface, and teaching in the classroom. This evaluation seeks to involve users in the development of a high-quality product that will be both usable and valuable to interested parties (Baranauskas; Martins; Valente, 2013; Gomes; Padovani, 2005).

We collected 33 responses during the Numerical and Computational Methods discipline at the Federal University of Santa Maria/UFSM, Cachoeira do Sul Campus. The test involved students from the Electrical, Mechanical, Agricultural, and Transport and Logistics Engineering courses, divided into three groups. However, six participants had problems and could not perform the tests as expected, either due to technical problems or other factors. Therefore, we exclude these responses from the data analysis, with 27 valid responses.

Figures 5 and 6 show the quantitative result of the evaluation of the first and second versions, respectively, with ease of use (A1 - A5), software usefulness (B1 - B3), interface (C1 - C6), and the intended use (D1 - D5) of SofEMN.

Regarding the software's ease of use, the evaluation of the second version obtained more significant agreement in all questions. In the second version, most users (96%) found it easy to learn to use the software. In the previous version, 89.7% of users totally or partially agreed with this statement (A1).

Regarding understanding how the software works (A2), 96% of users agreed totally or partially, a value close to the previous evaluation that had an agreement

of 92%. Users' perceptions regarding usefulness show that, in the second version, all users consider the software useful to improve learning (B1), while in the first version, this percentage was 92.3%. In addition, everyone considers that it would improve their productivity in carrying out activities (B2). In the first evaluation, this value was 94.9%. Finally, everyone considers that the software would facilitate the performance of activities (B3), while in the first version, this agreement was 94.9%.

Figure 5 – Result of the quantitative evaluation of the first version
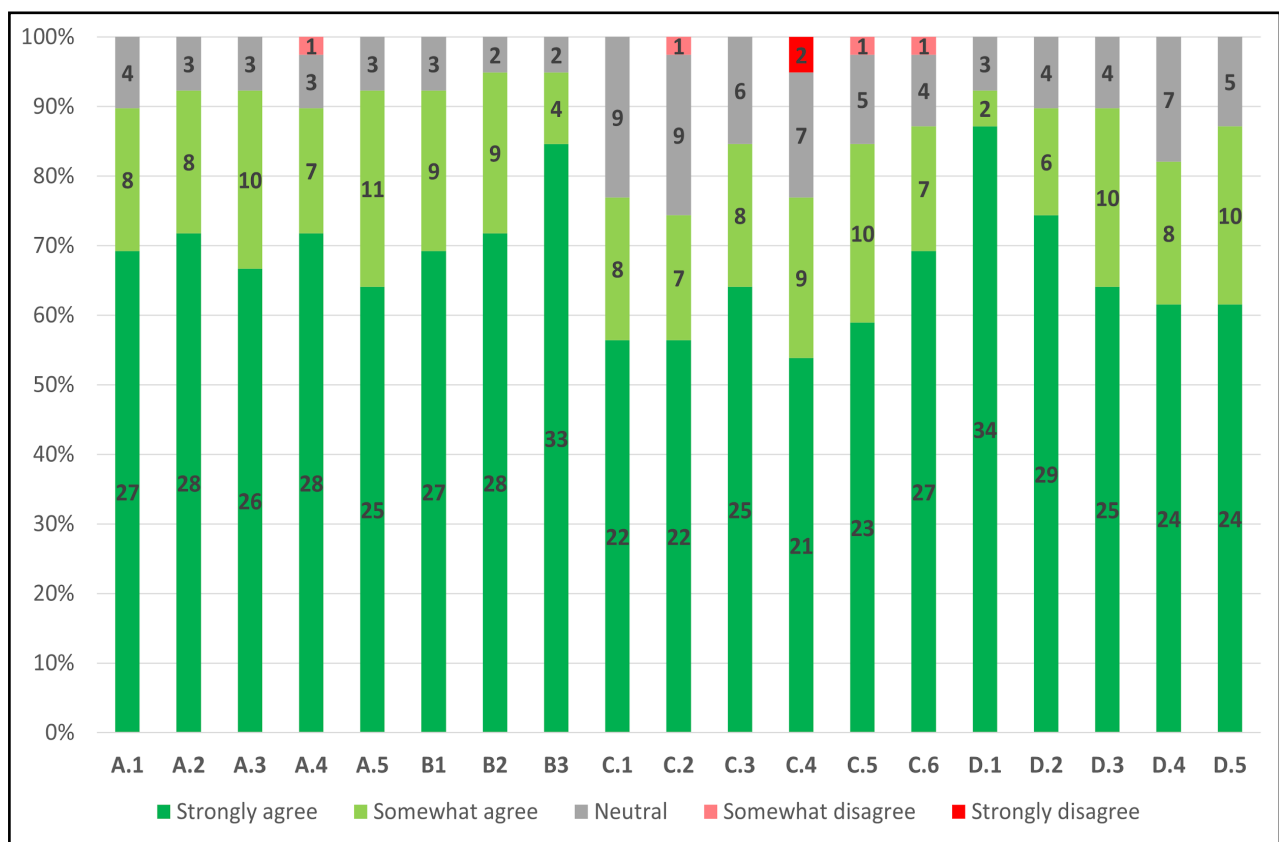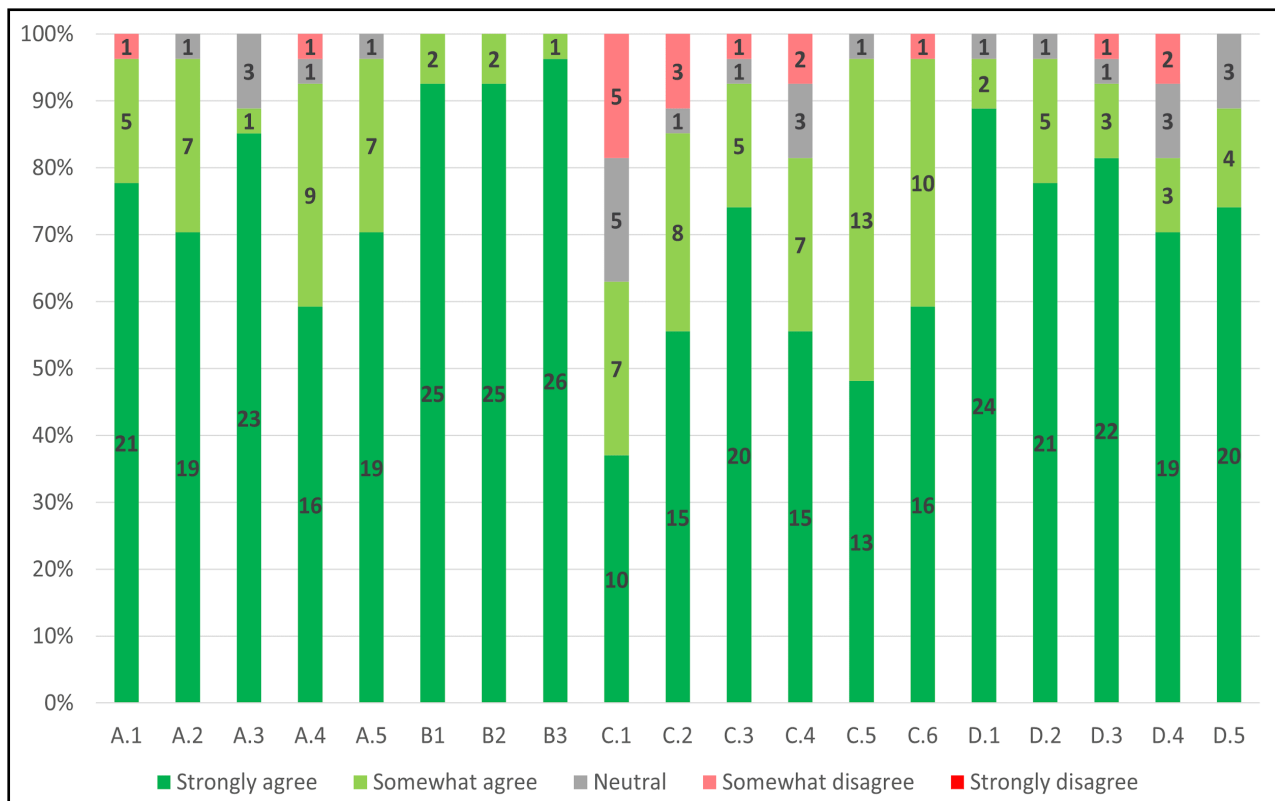

Source: Authors (2022)
Caption: Result obtained from usability tests of the first version of the software

When questioned about the ease of gaining use skills during the execution of activities (A3), in the previous version, 92.3% agreed totally or partially. In version 2, this agreement decreased to 88.9%.

Regarding the ease of learning to use the software (A4), in the first version, 89.7% of users totally or partially agreed, and in the second version, 92.6%. Finally, the agreement in considering the easy-to-use software (A5) increased from 92.3% to 96.3%.

More than half (63%) of participants positively evaluated the interface of SofEMN (C1) in its second version regarding its colors and buttons. However, in the previous version, 76.9% of the participants positively evaluated the interface.

Figure 6 – Result of the quantitative evaluation of the second version

Source: Authors (2022)
Caption: Result obtained from usability tests of the second version of the software

85% of users consider the buttons and information visible within the software (C2), while in the first evaluation, this percentage was 74.7%. 93% of users claimed to understand the software's words, nomenclatures, and icons (C3) in its second version, while in the first evaluation, this total was 84.6%. About images and icons (C4), 81% of participants consider them easy to recognize, a value greater than 76.9% of the first evaluation. We verified that 96% of the users approved the visualization of the functionalities and the navigation between the software screens (C5 and C6). In the first evaluation, C5 received a percentage of 84.6%, and C6 received 87.2%. Therefore,

although most of the evaluated items received much approval, there is still room for improvement in some aspects.

The last item evaluated concerns the use of SofEMN in the classroom, and it can be concluded that it would facilitate teaching and learning (D1), in addition to making classes more interesting for 96% of the participants (D2), representing a significant improvement compared to the percentage of 89.7% in the previous assessment. For 93% of the respondents, using the software would allow a better exploration of the available materials (D3), while in the previous version, the percentage was 89.7%.

Finally, according to 81% of the participants, the software allows for greater control over classroom activities and improves the level of learning (D4 and D5). The previous version agreed on 82% (D4) and 87.1% (D5).

Based on the analysis of the quantitative data obtained through the applied questionnaire, it was possible to verify that the users consider the software easy to use, valuable, and essential to aid learning. However, the software interface could have shown more satisfactory results, indicating the need to improve this aspect. As a result, it was decided to prototype a third version, which is currently under development.

In addition to the quantitative questions that assessed specific characteristics of the software, a non-compulsory question with an essay answer was created: "In this field, give your opinion about the SofEMN software". Including an essay question allowed users to express their opinions about the SofEMN software in a more accessible and comprehensive way.

Participant P1 comments that *"for me, I just need to detail the step by step"*. Participant P2 comments that the software is complete and suggests *"that it would also be valid to add some steps in the middle of the methods, and not only the final result, it would not need to be all detailed, but at least some value in between to facilitate understanding and to help recognize mistakes"*. Participant P3 considers the software simple and easy to use and confesses that" *it would be interesting to present the development of the calculation so that the student could find his mistake if he makes a mistake in an exercise "*.

Students also show interest in learning from step-by-step or error checking, as seen from these opinions. That is, they are not only interested in the software's final answer.

Participant P4 demonstrated his opinion, going according to the quantitative data, suggesting that although the software is superior to other software and easy to use, he considers "the interface is a bit 'raw' concerning the color scheme it uses, making some functionalities out of the intuitive field of view".

The participants' opinions demonstrate the importance of taking into account the functionalities and usability of the software and how the information is presented in the interface. Adding intermediate steps in the calculations and allowing the visualization of the calculation development can contribute to a better understanding of the concepts by the users. In addition, criticism of the interface reinforces the importance of investing in design and usability to make the software more intuitive and easy to use. These feedbacks are valuable for developing a third version of the software, which seeks to meet the needs and expectations of users.

5 REQUIREMENT SURVEY

Based on the quantitative and qualitative analyses, it was seen that the need to construct a module called "Learning the Content" so that the student could learn the content and not just check the exercises' answers.

The development team designed the third version of the software with consideration for the requirements gathering phase, ensuring that it aligns with the expectations of the end-users of the software. In addition to the requirements added to the software, based on the evaluation of the students, other requirements were considered for better tool utilization. We listed functional requirements to add educational software features to the first version of the software. We listed the functional requirements in Table 3.

For software to be considered educational, educational aspects must be considered in its development (Gladcheff; Sanches; Silva, 2002). In addition, educational

software is an instructional program that promotes self-teaching and aims to reinforce, complete, or serve as pedagogical material in disciplines (Meneguci, 2022).

Thus, it is important to list functional requirements classified as pedagogical. Therefore, some Pedagogical Requirements (RP) (Henrique, 2016) were followed for the creation of the new module "Learning Content" (Table 4).

Table 3 – Functional Requirements for the module "Learning Content"

Id	Description
RF01	The system should allow the user to access the material (texts and video classes) available in each method.
RF02	The system should allow the user to carry out activities related to each method studied.
RF03	The activities should be divided into two levels, Level 1: Theoretical and Level 2: Calculate.
RF04	The system should allow the user to check their progress in each level of activity.
RF05	The system must have a reward scheme for each correct activity.
RF06	The system should allow the user to check the response of activities related to each method studied.
RF07	The system should allow the user to ask for help/assistance when answering the questions contained in the activities.
RF08	The system should allow the user to generate a report with the activities that were carried out.
RF09	The system must have a top menu to facilitate navigation.

Source: Authors (2022)

The RF01 is related to the RP01 pedagogical requirement, which states that the contents must be planned from general to the most specific so that the student's prior knowledge is considered. In addition, RF02 is related to RP02: the content and resources made available must have pedagogical quality; they must meet the teacher's goals and suit the target audience. Moreover, with the RP18, the exercises and/or challenges follow an order; the user will only move on to the following exercises or challenges when he finishes the previous one. Another existing relationship is between RF04 and RF06 requirements with RP13: the system must display individualized *feedback*, that is, the student has feedback on their performance, and with RP61: The system must provide

immediate *feedback* to the user. Another current relationship is between the RF08 and the RP21: the system must allow the teacher to evaluate the student's performance through the generated artifacts and/or answers to the exercises (challenges). Finally, RF03 is related to RP63, which says that the contents must be divided into several levels of knowledge, always starting from the lowest level of complexity.

We will present the module created from the listed requirements in the following section, pointing out each requirement on the screen and referring to its implementation.

Table 4 – Pedagogical Requirements for the module "Learning Content"

Id	Description
RP01	Content should be planned from general to more specific, so that the student's prior knowledge is taken into account.
RP02	The content and resources made available must have pedagogical quality, that is, they must meet the teacher's objectives and suit the target audience.
RP13	The system should display individualized feedback, so that the student has feedback on their performance.
RP18	The exercises and/or challenges follow an order, this means that the user will only move on to the next exercises and/or challenges when he finishes the previous one.
RP21	The system should allow the user (teacher) to evaluate the performance of the user (student), through the generated artifacts and/or answers to the exercises (challenges).
RP61	The system must provide immediate feedback to the user.
RP63	The contents must be divided into several levels of knowledge, always starting from the lowest level of complexity.

Source: Authors (2022)

6 MODULE LEARNING CONTENT

According to Theis, Souza, Fialho and Pereira (2021), prototyping has applications in five major groups: (i) generator of interest and motivation, (ii) way to save time and resources, (iii) tool for testing, (iv) generator of design alternatives and (v) analysis tool accessible to all those involved in the project.

Therefore, we used prototyping to validate the listed requirements (iii) and provide better visualization for users (teachers and students) (v).

The great advantage of using prototypes is allowing a preview of the final solution, which can be validated quickly and allows users to request some change being carried out in the prototype, not during software development. They used the Figma[10] tool for the prototyping stage, a vector graphics editor, and prototyping design projects. Below, in Figure 7, some screens resulting from the prototyping are shown, which exemplify the flow of use of the "Learning Content" module.

Figure 7 shows that the user can choose on the initial screen of the system (01) between the module "Learning Content", which presents functionalities aimed at teaching numerical methods, and the module "Verifying Content". "Learning Content" works like a calculator since, for each method, the user enters the values of the variables, and the software returns the final result. The top menu of the screen (02) allows you to choose which method you want to study. After choosing, the system releases the content prepared for this selected method. This functionality refers to the RF01 requirement contained in Table 3.

In Figure 7, screen 03, the levels of activities are presented, divided into theoretical and practical parts. The theoretical part consists of solving questions about the content, and the practice directly involves solving calculations. The screen in question represents the RF03 and RF04 requirements.

On-screen 04, we can see that the software presents an objective question, and we have four alternatives to answer, and we can request help if necessary. On-screen 05, the user can use the light bulb icon to ask for a simple or more specific tip. The resource works as a reward because after the user gets a question right, he receives extra tips automatically. The screen in question represents the RF05 and RF07 requirements in Table 3.

---

[10] https://www.figma.com/

Figure 7 – Learning the Content module usage flow
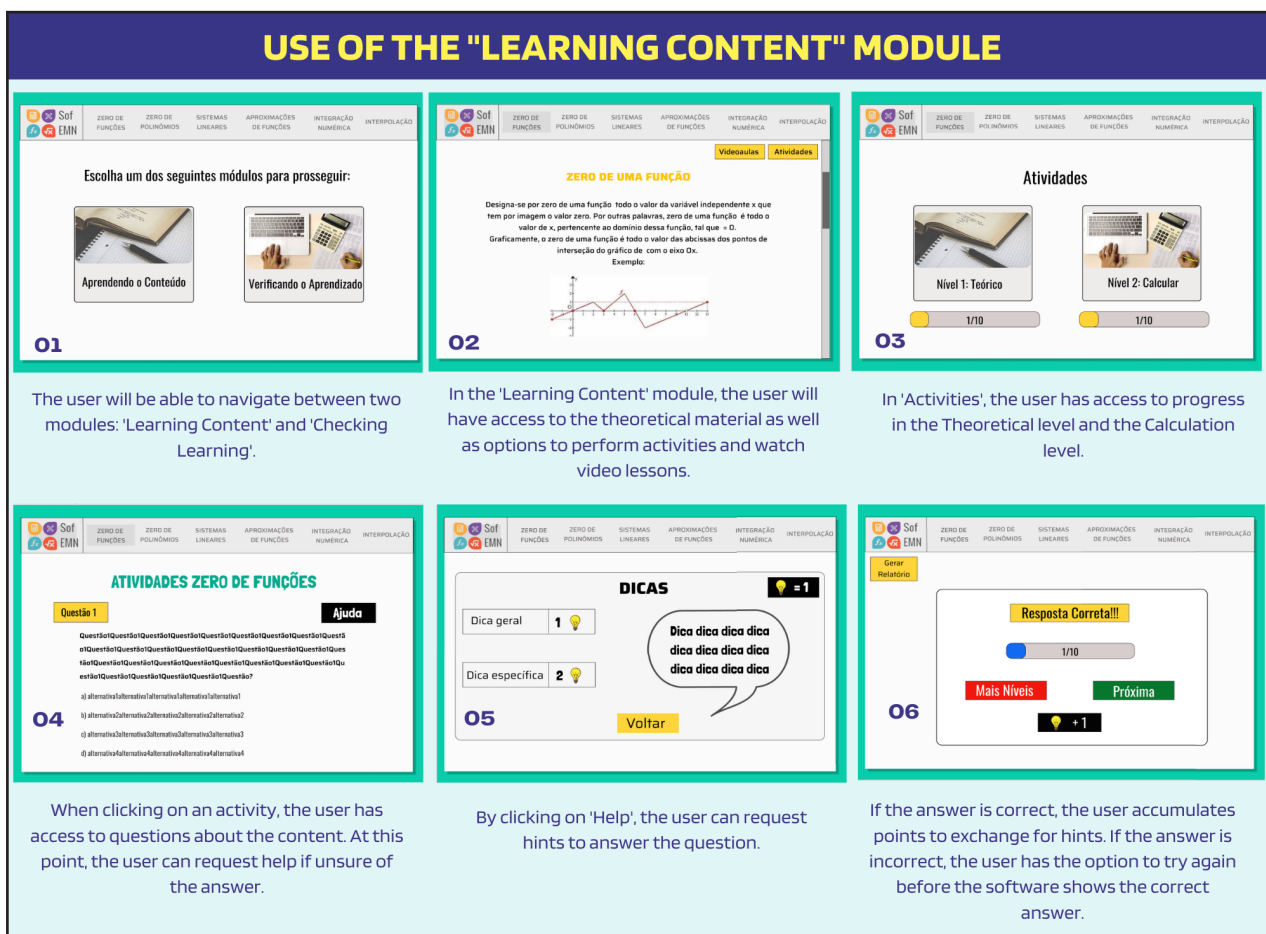
Source: Authors (2022)
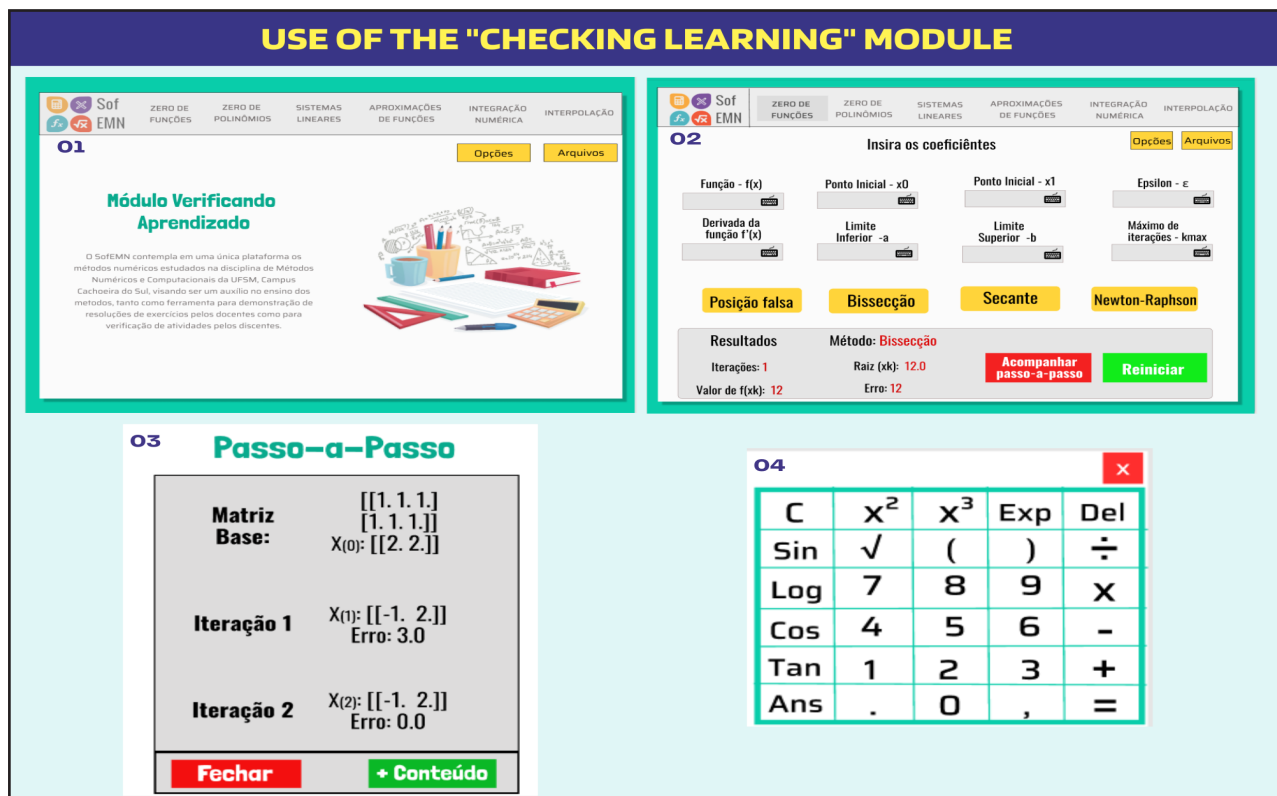Caption: Screens resulting from the prototyping of the flow of use of the "Learning Content" module

As the user hits or misses the question, the software behaves differently. On-screen 06, the possible interactions can be seen when the user answers the question correctly. The user can move on to the next question if the answer is correct. Otherwise, if the answer is incorrect, he can try again. After the user misses the answer twice consecutively, the system immediately presents the correct answer. It later releases access to the theoretical materials on the subject addressed in the question through the buttons in the left-side menu. The screens represent the RF06 requirement.

Consequently, after the user answers the question, he can generate a report and send it to the teachers. The report must contain all performance in the activities, both correct and incorrect questions, according to the RF08 requirement.

The focus of this work was to present the new module inserted in the software,

which aims to teach the content to the students. Additionally, the software underwent improvements in terms of interface issues that users highly commented on during evaluations. Figure 8 shows the layout of the new software version.

Figure 8 – Checking Learning module screens

Source: Authors (2022)
Caption: Screens resulting from the prototyping of the use flow of the "Verifying Learning" module

Figure 8 shows a significant change in the main screens of the software. On-screen 01, there is the initial screen of the module, and at the top, the menu of options of methods available to be used. Also, the "Options" and "Files" icons can be accessed at any time, allowing the student to browse the contents of the previous module. On-screen 02, there are the entries for the chosen method. At this stage, the change occurred in the format and colors of each button and the interface as a whole. A keyboard was also added to the software, screen 04, which appears when the icon in each input field of screen 02 is selected, facilitating the use of variables and other characters as input for each method. Finally, screen 03 shows the step-by-step process that was performed to reach the final answer of the method, providing the user with a verification of the calculations and a better understanding.

7 FINAL CONSIDERATIONS

To assist students in learning Numerical Methods and serve as a support in solving exercises was the purpose of creating SofEMN. After developing a second version, an evaluation of the software was carried out, through which it was possible to perceive that some improvements still needed to be incorporated. The evaluation responses noted that a new layout was needed, mainly new features that would meet the students' learning demands.

Based on the evaluation responses, we decided to plan the development stages of the third version, starting with requirements gathering and developing a prototype. The functional requirements could be validated in this way. The construction of a teaching module, in which the contents and exercises are presented so that the student can review and appropriate the content seen in the classroom, thus providing opportunities for self-teaching, was the main contribution of the third version of the software. In this way, the tool under development gained an educational bias. Therefore, the evolution of SofEMN will improve teaching and assist students and teachers in studying and learning numerical methods.

In future works, the goal is to improve the third version of the software. In addition, apply two assessments, one considering usability and the other to verify the students' learning when using SofEMN during the Numerical and Computational Methods course.

REFERENCES

ABREU, T; GONÇALVES, R. Uma experiência com o uso do software SageMath no ensino de métodos numéricos. *In*: CONGRESSO NACIONAL DE PRÁTICAS PEDAGÓGICAS NO ENSINO SUPERIOR, 4., Setúbal, Portugal. **Anais** [...] Setúbal: IPL, 2017. p. 15-20.

BARANAUSKAS, M. C. C. O modelo semioparticipativo de design. In: BARANAUSKAS, M. C. C.; MARTINS, M. C.; VALENTE, J. A. (eds.). **Codesign de Redes Digitais**: tecnologia a serviço da inclusao social. Porto Alegre: Editora Penso, 2013. p. 38–66.

BARBOSA, E. F.; ARIMOTO, M. M.; ISOTANI, S.; MALDONADO, J. C. Software livre e conteúdos educacionais abertos no ensino de computação. **Computação Brasil**, Porto Alegre, v. 27, n. 2, p. 64-70, 2015. Disponível em: https://www.sbc.org.br/component/flippingbook/book/21. Acesso em: 24 mar. 2023.

BASILI, V. R.; ROMBACH, H. D. The TAME project: towards improvement-oriented software environments. **IEEE Transactions on Software Engineering**, Washington, DC, v. 14, n. 6, p. 758–773, jun. 1988. DOI 10.1109/32.6156. Disponível em: https://ieeexplore.ieee.org/abstract/document/6156. Acesso em: 24 mar. 2023.

GLADCHEFF, A. P.; SANCHES, R.; SILVA, D. M. da. Um instrumento de avaliação de qualidade de software educacional: como elaborá-lo. **Pensamento e Realidade**, São Paulo, v. 5, n. 11, p. 3-20, 2002. Disponível em: https://revistas.pucsp.br/pensamentorealidade/article/view/8484. Acesso em: 24 mar. 2023.

GOMES, A. S.; PADOVANI, S. Usabilidade no ciclo de desenvolvimento de software educativo. **Simpósio Brasileiro de Informática na Educação - SBIE**, 16, 2005, Minas Gerais. **Anais** [...]. Porto Alegre: SBC, 2005. Tema: Educação a Distância.

HENNING, E. *et al*. Para além da computação estatística: o uso do ambiente R para o ensino de métodos numéricos. **RENOTE**, Porto Alegre, v. 14, n. 1, 2016. DOI 10.22456/1679-1916.67348. Disponível em: http://seer.ufrgs.br/index.php/renote/article/view/67348. Acesso em: 24 mar. 2023.

HENRIQUE, M. S. **EDUCATALOG4RE**: um catálogo de requisitos para auxiliar o desenvolvimento softwares educacionais. 2016. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Pernambuco, Recife, Pernambuco, 2016. Disponível em: https://repositorio.ufpe.br/handle/123456789/18029. Acesso em: 24 mar. 2023.

HOED, R. M. **Análise da evasão em cursos superiores**: o caso da evasão em cursos superiores da área de computação. 2016. Dissertação (Mestrado Profissional em Computação Aplicada) – Universidade de Brasília, Brasília, DF, 2016. DOI: http://dx.doi.org/10.26512/2016.12.D.22575. Disponível em: https://repositorio.unb.br/handle/10482/22575. Acesso em: 24 mar. 2023.

KENSKI, V. M. **Educação e tecnologias**: o novo ritmo da informação. 2. ed. Campinas, SP: Papirus, 2007.

MENEGUCI, M. do N. **Uma proposta de metodologia de ensino de logaritmos através da tecnologia**. Dissertação (Mestrado Profissional em Matemática em Rede Nacional) – Universidade Federal Rural do Rio de Janeiro, Rio de Janeiro, RJ, 2022. Disponível em: https://tede.ufrrj.br/handle/jspui/5736. Acesso em: 26 mar. 2023.

FINGER, A. F.; LORETO, A. B.; SOUBHIA, A. L.; ANDRADE, A. C. Avaliação de usabilidade do SofEMN: Software de apoio ao Ensino de Métodos Numéricos. *In*: XXXII Simpósio Brasileiro de Informática na Educação, 2021. **Anais** [...]. Porto Alegre: SBC, 2021. p. 103–112. Disponível em: https://sol.sbc.org.br/index.php/sbie/article/view/18047. Acesso em: 26 mar. 2023.

RAEDER, M.; PY, M.; RIGO, S.; PINHEIRO, J. L2PM: relato de uma experiência sobre o ensino integrado de Lógica, Programação e Matemática para Computação. *In*: XXIV Workshop sobre Educação em Computação, 2016, Porto Alegre. **Anais** […]. Porto Alegre: SBC, 2016. p. 2186–2195. Disponível em: https://sol.sbc.org.br/index.php/wei/article/view/9662. Acesso em: 24 mar. 2023.

ROCHA, H. V. da; BARANAUSKAS, M. C. C. **Design e Avaliação de Interfaces Humano-Computador**. Campinas, SP: Unicamp, 2003.

SILVA, S. R. X. da. O uso do *Scilab* como ferramenta para o ensino de cálculo numérico. *In*: XLI Congresso Brasileiro de Educação em Engenharia, 2013, Gramado. **Anais** […]. Brasília: ABENGE, 2013. Tema: Educação na Era do Conhecimento. Disponível em: https://turing.pro.br/anais/COBENGE-2013/pdf/117997_1.pdf. Acesso em: 24 mar. 2023.

TATTO, F.; SCAPIN, I. J. Matemática: por que o nível elevado de rejeição? **Revista de Ciências Humanas**, v. 5, n. 5, p. 57–70, 2004. Disponível em: http://revistas.fw.uri.br/index.php/revistadech/article/view/245. Acesso em: 24 mar. 2023.

THEIS, M. R.; SOUZA, G. G. A. de; FIALHO, F. A. P.; PEREIRA, R. A importância da prototipagem no processo de design e suas relações como mídia do conhecimento. *In*: XI Congresso Internacional de Conhecimento e Inovação, 2021, Maringá. **Anais** […], v. 1, n. 1. Maringá: Unicesumar, 2021. DOI: 10.48090/ciki.v1i1.1058. Disponível em: https://proceeding.ciki.ufsc.br/index.php/ciki/article/view/1058. Acesso em: 26 mar. 2023.

VALENTIM, N. M. C. Avaliando a qualidade de um aplicativo web móvel através de um teste de usabilidade: um relato de experiência. In: XIII Simpósio Brasileiro de Qualidade de Software, 2014, Blumenau. Anais […]. Porto Alegre: SBC, 2014. p. 256–263. DOI: https://doi.org/10.5753/sbqs.2014.15258. Disponível em: https://sol.sbc.org.br/index.php/sbqs/article/view/15258. Acesso em: 24 mar. 2023.

Authorship contribution

1 – Aline Brum Loreto

PhD in Computing
https://orcid.org/0000-0002-5001-4836 • aline.loreto@ufsm.br
Contribution: Orientation of the student who created the first versions of the software, writing and revision of the text.

2 – Arthur Cordeiro Andrade

Graduating in Electrical Engineering
https://orcid.org/0009-0008-1212-9713 • arthurcoand@gmail.com
Contribution: Software development.

3 – Ana Luisa Soubhia

PhD in Mechanical Engineering
https://orcid.org/0000-0002-0861-0255 • ana.soubhia@ufsm.br
Contribution: Orientation of the student who created the first versions of the software, writing and revision of the text.

4 - Amanda Spolaor

Graduating in Software Engineering
https://orcid.org/0009-0000-3847-8200 • amandaspolaor.aluno@unipampa.edu.br
Contribution: Prototype development, writing and revision of the text.

5 - Guilherme de Lima de Menezes

Degree in Mathematics, Specialist in Interdisciplinary Teaching Practices
Graduating in Software Engineering
https://orcid.org/0000-0002-6659-6551 • guilhermemenezes.aluno@unipampa.edu.br
Contribution: Prototype development, writing and revision of the text.

6 - Bruna de Abreu Dias

Bachelor of Software Engineering
https://orcid.org/0009-0006-0892-340X • brunadias.aluno@unipampa.edu.br
Contribution: Writing and revision of the text.

7 - Alice Fonseca Finger

PhD in Computing
https://orcid.org/0000-0003-4168-2872 • alicefinger@unipampa.edu.br
Contribution: Orientation of the students who elaborated the prototype, writing and revision of the text.

How to cite this article