Sánchez, J. (2024) Editor. Nuevas Ideas en Informática Educativa, Volumen 18, p. 53 - 61. Santiago de Chile.

Teaching Programming: Integrating Active Methodologies to Enhance Learning

Lia Alessandra da S. Martins
lia.martins@ifam.edu.br
Federal Institute of Education, Science, and Technology of Amazonas - Manaus Zona Leste *Campus*
Manaus, Amazonas, Brazil

David B. Ramos
david.ramos@ifam.edu.br
Federal Institute of Education, Science, and Technology of Amazonas - Manaus Zona Leste *Campus*
Manaus, Amazonas, Brazil

Julieuza de S. Natividade
julieuza.natividade@ifam.edu.br
Federal Institute of Education, Science, and Technology of Amazonas - Manaus Distrito Industrial *Campus*
Manaus, Amazonas, Brazil

José E. B. de S. Linhares
breno.linhares@ifam.edu.br
Federal Institute of Education, Science, and Technology of Amazonas - Manaus Zona Leste *Campus*
Manaus, Amazonas, Brazil

Lohan Silva
lohan.silva@lge.com
LG Eletronics do Brasil Ltda
Manaus, Amazonas, Brazil

Nelson Gouvea
nelson.gouvea@lge.com
LG Eletronics do Brasil Ltda
Manaus, Amazonas, Brazil

Luís Arantes
luis.filho@lge.com
LG Eletronics do Brasil Ltda
Manaus, Amazonas, Brazil

Abstract

This article presents the results obtained in the Python Leveling discipline of the project called "Advanced Training Course in Hyperautomation", whose objective is to provide college students, from Engineering or Technology courses, the opportunity to develop skills to work in digital transformation, requirements necessary for employment in the job market. To achieve this objective, active methodologies were used through problem-based learning, with instructors and mentors participation to encourage challenges resolution, combining theory and practice. The results obtained suggest that using active methodologies can contribute to students technical development and socio-emotional skills, in the curricular components.

Keywords: Python Programming Language, Problem-Based Learning, Digital Transformation

1 Introduction

Teaching programming has become an essential skill in a world increasingly driven by technology. The ability to master a programming language is not just restricted to technical development, but involves the ability to solve complex problems in a logical and structured way. In this context, active methodologies are emerging as an effective response to contemporary challenges in education. By integrating these methodologies into programming teaching, not only technical training is sought, but also the promotion of socio-emotional and critical thinking skills, known as soft skills, which are essential for meeting the demands of a constantly changing job market [10].

Although technical education is indispensable, one of the biggest challenges lies in balancing the development of hard skills (technical skills) and building socio-emotional competencies. Simply imparting technical knowledge is no longer enough to prepare students for the complex challenges of the modern job market. In this scenario, the integration of active methodologies into the teaching-learning process, such as Project-Based Learning (PBL) [2], Design Thinking (DT) [9] and Challenge-Based Learning (CBL) [6] stand out as promising solutions. These methodologies engage students more deeply by encouraging them to apply knowledge in a practical and collaborative way.

PBL and DT are widely recognized for promoting the resolution of real problems, creating opportunities for students to apply the content learned in the classroom to concrete situations. These methodologies also stimulate creativity, critical thinking and teamwork, which are crucial aspects for students' all-round development. In addition, CBL goes further by inserting students into authentic challenges that simulate complex problems in industry and society, requiring a combination of technical knowledge and socio-emotional skills to overcome them.

In light of this scenario, partnerships between the industrial sector and educational institutions emerge as an effective strategy for the application of these methodologies in programming education. The collaboration between a global technology company and a federal educational institution, as in the case of the Advanced Training in Hyperautomation project, offers a unique opportunity to train students in technology and automation areas. This type of partnership ensures not only the quality of the pedagogical content but also its practical relevance, bringing students closer to real market demands and preparing them for their professional future.

The Python Leveling course, part of the Advanced Training in Hyperautomation project, is a clear example of how the application of active methodologies can enhance learning. Throughout the weeks of the course, students are exposed to a series of challenges and practical activities that require mastery of the Python language, as well as the ability to collaborate in teams, solve problems creatively, and think critically. The course structure, divided into instruction and mentoring phases, ensures a balance between theoretical content and practice, creating a dynamic learning environment that closely resembles professional reality.

The mentoring model adopted was a key element in the success of the course, reinforcing the importance of collaborative learning. During the practical sessions, students are organized into teams, which promotes the exchange of knowledge and experiences. The creation of a blog to record activities, for example, encourages reflection on the learning process and allows students to share their challenges and progress.This dynamic creates a more collaborative and interactive learning environment, which is essential for developing skills that go beyond technical knowledge.

The challenges proposed during the course also play a crucial role in developing technical and socio-emotional skills. By simulating real problems and giving teams the task of solving them, the course encourages critical thinking and creativity in problem-solving. The format adopted, which includes both the conception of problems and the coding of solutions, mirrors the dynamics of software development in the market, allowing students to experience, in a controlled environment, the challenges they will face in the job market.

Another essential aspect of the methodology is the emphasis on continuous and collaborative assessment. The dynamics of the conversation circle and the presentation of the results in a "jury court" format provide a unique opportunity for students to reflect on their approaches and receive feedbacks from their peers and mentors. This type of assessment fosters self-criticism and a sense of responsibility, skills that are essential for personal and professional development. In addition, exchanging feedbacks between teams encourages the development of communication and teamwork skills, which are increasingly valued in the market.

Finally, the thematic workshops offered during the course complement the students' training, providing an even more complete educational experience. Workshops such as Google Collaboratory and OS Library in Python allow students to come into contact with tools that are widely used in the market, broadening their perspective on the use of technologies in everyday professional life. These practical activities, aligned with the proposed challenges and mentoring, guarantee a robust education, integrating technical knowledge and a critical and strategic view of the use of technology.

In summary, this article seeks to present the skills and competences developed during the Python Leveling course, demonstrating how the application of active methodologies, especially Challenge-Based Learning, contributed to the students' all-round development. The analysis of this experience provides valuable insights into the best practices for implementing active methodologies in programming teaching, promoting a more complete education in line with the contemporary demands of the technology market.

Thus, the structure of this article is organized as follows: in Section 2, five relevant studies will be presented that address active methodologies in programming teaching. Section 3 details the methodologies applied in the subject of Python Leveling, focusing on the integration of PBL, DT and CBL approaches. In Section 4, the format of the classes will be explained, highlighting how these methodologies were used in a practical way to enhance student learning. Finally, in Section 5, the results obtained will be analyzed based on the students' grades and the data collected through forms, allowing a critical reflection on the effectiveness of the methodologies applied in the course.

2 Related Works

In this section, we present some related works regarding innovative learning researches. These works, which also focus on learning, adopted different strategies such as problem-based learning, flipped classroom, computational thinking, project-based learning, design thinking as well as involving practical activities promoting more engaging and dynamic learning practices.

"ROBÔ-EDU" [11]

This paper presents the results of an extension project called Robô-Edu. The project sought to apply the use of educational robotics to encourage public school students to develop their computational thinking skills. The project gave the students participating in the project access to LEGO Mindstorms EV3 robot kits, equipment that is generally unavailable to public schools due to the high cost of acquisition. The project was divided into training the instructors and then classes with the students. The classes used the Problem-Based Learning (PBL) methodology, in which specific content was presented, followed by practical activities, in which computers were used to program the robots. At the end of each lesson, there was a time to reflect on the behavior of the robots and the concepts presented in the lesson. The work was carried out in partnership with a federal institution responsible for the project and schools in four municipalities in the state of Amazonas, Brazil. The authors concluded that they had a high rate of success in terms of student involvement in the activities, showing that the use of educational robotics stimulated their participation. Another result was the qualification of teacher multipliers in the municipalities involved in the project.

"Personalização em experiências de aprendizagem" [4]

The article presents how the application of learning personalization is perceived from the teaching point of view, especially with adult learning. The authors report that an introductory extension course on practical personalization processes in learning experiences was offered at the Federal University of Rio Grande do Sul. The flipped classroom method was applied online. Thus, part of the studies were carried out asynchronously and part synchronously online for practical and collaborative activities. The virtual learning environment used a variety of resources such as text, videos, animations, presentations, forums, among others. The participants were 29 educators, including teachers, instructors and tutors, with a focus on learning for people over the age of 18. The authors presented the main factors that impact on the personalization process: context, cognition and motivation for students; support and methodology for teachers; and environment and technologies for both. The main advantages were: greater focus on learning through experience, better performance, greater engagement. The main obstacles were: adapting an existing curriculum, time taken to prepare learning experiences and lack of initiative on the part of some students.

"Construa uma casa" [1]

The authors present the use of computational thinking as well as the use of algorithms in the context of math teaching and in a workshop for primary and secondary school students. The experiment consisted of an unplugged activity. The aim was to teach programming in which the student had to create an algorithm for building a house, using concepts from analytical geometry. The algorithms were represented using flowcharts. By following the algorithms, the students painted specific squares on a sheet of paper. In the case of the high school students integrated with computer science, as they had prior knowledge, they chose to represent the algorithms in portugol. When answering the survey questions, they also showed that they had no difficulty in carrying out the tasks because they already had prior knowledge. For the workshop, the representation was made using directional arrows and an image representing the action of painting a square, in which the participants were also able to carry out the proposed challenges. In this way, the authors also highlighted the need to adapt the level of difficulty of the proposed activities, taking into account the participants' prior knowledge.

Chatbot Cerebrum IBM Watson Assistant [5]

This paper presents the use of a chatbot to assist in the teaching and learning process of mathematics, specifically first-degree algebraic equations with one variable. The chatbot was developed on IBM's Watson platform. Fifty-one seventh-grade students from a municipal public school in the Paranhana Valley took part in the study in 2022. They were divided into two classes, A and B, where class A, with 25 students, used the chatbot, and class B, with 26 students, used the textbook for study. Both the textbook and the chatbot presented the same strategic principles for solving first-degree equations. The authors point out that the students in both classes had no contact with learning algebra until they took part in the experiments and that there was no intervention from the teacher who conducted the experiments. In a post-test, all the students were given the same set of questions. For all the questions, the average number of correct answers in the class was higher than in class B. Thus, the authors showed that the overall percentage of correct answers for students who used Cerebrum was 82.69%, and 65% for those who did not. This showed that the chatbot was able to help in the teaching-learning process in the subject proposed in the research.

"Aprendizagem Baseada em Projetos e Simulação Computacional para o Ensino de Engenharia" [7]

This paper presents an experience report on the use of simulator software, rubric-based assessment and project-based learning applied to the subject of Simulation in a degree course in Production Engineering. Thirty-four students from three different semesters between 2021 and 2022 took part and developed 14 projects. Using the rubrics, the authors assessed the following competences: formulating the problem, planning the project, characterizing the system under study, formulating the conceptual model, collecting data, processing data, formulating the computational model, validating and verifying models, designing experiments, identifying the solution, documenting and presenting the results. The classes alternated between dialogued lectures to present the content and practical projects to carry out the practical activities. The projects developed were aimed at sizing service teams, analyzing productivity and identifying bottlenecks, and the students visited real systems to support their models. On a scale of 0 to 10, the average score was 9.1. Regarding the students' perception, it was stated that the understanding of concepts became more tangible in practice and that the use of modeling and simulation facilitated problem solving. The students also said that the use of projects allowed them to practice the knowledge they had acquired and to get closer to the professional reality of the area. As the main contribution of the study, the authors point to the presentation of a proposal for the development of competences in the area of engineering.

“Metodologias ativas e trabalho colaborativo: uma integração entre diferentes abordagens” [8]

This study described the implementation of an intervention with a fourth-grade class in a public school. The objective was to integrate approaches such as Design Thinking, Maker Culture, and the STEAM approach to promote creative and collaborative learning. The methodology included a multidisciplinary team from the areas of education and technology. The STEAM approach and the integration of different disciplines contributed to more meaningful and contextualized learning. The research involved working together on the disciplines of Science, Arts, and Mathematics. The students were encouraged to think about what an ideal school would be like. They then carried out a practical activity of building a model of the school they had designed. Finally, the groups shared their experiences with the activity. According to the authors, the results indicated the development of critical, creative, and scientific thinking skills, as well as socioemotional skills.

As with related work, this research is based on teaching through active methodologies. Most of these works develop both hard and soft skills to enhance learning. Robô-Edu [11] in context of Educational Robotic used PBL as active methodologies what involves lessons and application of some challenges. Students worked in groups training soft skills. Paper [4] uses flipped classroom as learning strategies e learning by experiences and engagement. In [1] was used computational think and unplugged activity. Work [5] used chatbot to assist the learning of first-degree algebraic equations. Work [7] applied simulation software, project-based learning and rubric-based assessment for the development of competences. Work [8] integrated active methodologies with collaborative activity.

3 Methodological Procedures

This section describes the methodology applied during the execution of Python Leveling, the subject of the “Advanced Training Course in Hyperautomation” project. which is the subject of this article. The research methodology is based on active learning methodologies, with a focus on problem-based learning. This methodology seeks to place the student at the center of the learning process, encouraging the development of technical and socio-emotional skills, by a group of teachers and with the facilitation of a tutor. [3]

In terms of the teaching-learning process, the course was divided into two classes, separated into two main phases: instruction and mentoring. The sessions were held weekly, per shift, divided into 2 (two) days for instruction and 3 (three) days for mentoring. Each class was made up of 20 students, totaling 40 participants, with ages ranging from 18 to 39 years old and with previous programming experience.

Initially, on Mondays and Tuesdays, the class was led by the subject instructors, who were responsible for teaching the theory and proposing exercises in the form of challenges. The mentors were then responsible for conducting the hands-on methodology, helping the class to solve the proposed exercises.

The mentoring sessions took place on Wednesdays, Thursdays and Fridays, in person. During the course, each session was dedicated to different aspects of the project’s development, including introducing the mentors, analyzing skills and difficulties, dividing the class into teams, creating and monitoring the blog for the log, evaluating the previous week, solving the list of exercises and themed workshops.

4 Development

This section describes the activities carried out with the classes during the mentoring period. The activities developed were: i) analysis of skills and difficulties; ii) division of the class into teams; iii) creation and monitoring of the blog for the logbook; iv) evaluation of the previous week; v) resolution of the list of exercises and challenges and vi) thematic workshops.

Initially, the students organized themselves into teams, divided into 4 (four) members, making a total of 5 (five) teams per class. The nicknames of the teams were chosen, using arbitrary contexts for class 1 and the context of computer personalities for class 2. For class 2, the team names were: Hopper, Gates, Turing, Jobs and Linus.

The process of choosing team members was based on an analysis of skills and difficulties collected orally by the mentors. Each team was formed with students who had previous experience in areas of interest to the course and with the participation of females, balancing them out. To solve the list of exercises provided by the instructor of the Python Leveling course, a dynamic conversation circle was held, in which each team was positioned on their benches for better interaction. Questions were selected on the beecrowd platform about the content covered in class and, using an intra-team relationship, the students chose the questions to solve and socialized the resolution process, impediments identified and alternative routes to completion. Figure 1 below shows an illustration of the interaction between team members in the process of solving exercises:

In addition, time was set aside during the mentoring session to create a blog log, using the Google application called Blogger. The aim of this teaching tool was to record the activities carried out by the teams during the week, including success stories and obstacles, if any. At these times, usually during the last session of the week, it was intended to produce reports in the blog for the log.

Each week during which the mentors interacted with the class, the weekly agenda was presented. This included general information and the itinerary of activities to be carried out by the students.

Figure 1. Students from the Gates team of class 2 in the process of solving exercises, divided into teams by benches.

Figure 2. Student from the Linus team in class 2 in the process of analyzing compliance with requirements to resolve the weekly challenge.

From the second week onwards, there was a dynamic evaluation of the previous week. Basically, each team was assigned to go to the front of the classroom, in which it was agreed that they should comment on the facilities and difficulties encountered in solving the proposed exercises, the interaction developed with the team and the content presented in the classroom with the Python Leveling instructor.

As well as exercises to fix the content presented in class, the teams were given challenges that simulated real everyday problems. One of them was a challenge, which consisted of two stages: problem design by one team (stakeholders) and solution coding by the adjacent team (developers). For class 2, for example, the strategy adopted to carry out the challenge was as follows: team Hopper solves the problem proposed by team Gates; team Gates solves the problem proposed by team Turing; team Turing solves the problem proposed by team Jobs; team Jobs solves the problem proposed by team Linus; and team Linus solves the problem proposed by team Hopper.

According to Figure 2, there were times when the teams consulted the stakeholders (adjacent team) to analyze the conformity of the requirements for solving the problem, rehearsing an important stage of the software development process: elucidation of the requirements. The presentation of the results obtained by the teams took place in a "jury court" format: the team responsible for coding at the front of the classroom and the stakeholders at the back for evaluation. The order of the teams' presentation was drawn by lot, in the following sequence: Linus, Jobs, Gates, Hopper, Turing.

In the last week of the course, the mentors promoted two thematic workshops: Google Collaboratory (see Figure 3) and OS Library in Python (see Figure 4). With a format divided between theory and practice, the students had the opportunity to learn about and improve the technical skills (hard skills) and socio-emotional skills (soft skills) needed to succeed in the "Advanced Training Course in Hyperautomation".

Figure 3. Completion of the workshop on Google Colaboratory.

5 Presentation and Result Analysis

This section will present the results achieved during the execution of the Python Leveling. The course was taught over 4 weeks, with each week following a five-day cycle: 2 days of theoretical classes led by the instructor and 3 days dedicated to mentoring, during which the practical activities illustrated in Figure 5.

Figure 4. Workshop on Library OS.

Figure 5. Activities carried out in mentoring.

In **Week 1**, the instructor began the Python Level, explaining the definition and importance of programming logic. Basic language concepts were also covered, such as variables, identifiers, data types, input, output, assignment operators, arithmetic operators, simple and compound conditional structures, relational and logical operators. Based on this content, during the week's mentoring, the students solved questions on the beecrowd platform, as shown in Figure 6 and took part in a round table discussion, which was very productive, as through this dynamic the students were able to share the process they adopted to solve the questions, the challenges they encountered and how they solved these problems. Also this week, each team created a blog, as shown in Figure 7, to use as a logbook during the lessons.

At the end of Week 1, the students reported that they found the sessions highly productive and reported no difficulties in carrying out the dynamic "conversation circle" to solve the list of exercises. It was also observed that some students had a low level of familiarity with project management tools, particularly evidenced by their use of the GitHub

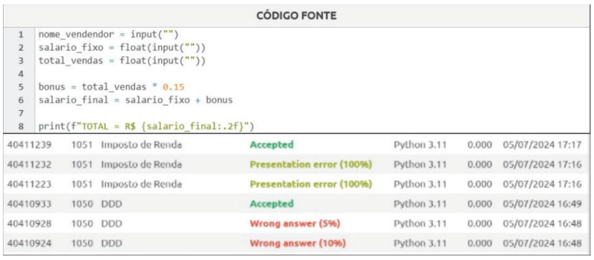

Figure 6. Example of issues resolved on the platform beecrowd.

Projects tool. The analysis revealed that limited understanding of how to effectively use GitHub Projects can result in difficulties in coordinating activities, setting clear goals and communicating effectively within teams.

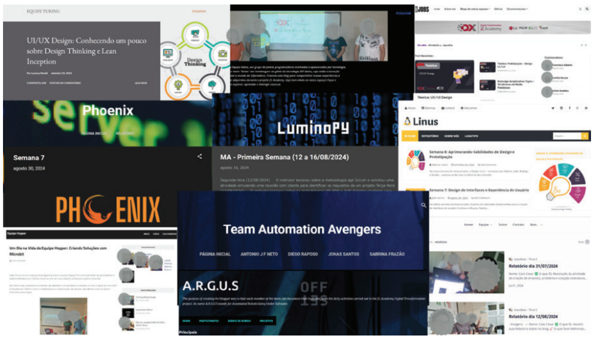

Figure 7. Blogs pages created by teams.

During the development of the projects, significant obstacles were identified relating to communication between team members, which directly impacted on the effectiveness of resolving issues. It was observed that difficulties in exchanging information and coordinating efforts contributed to the slow resolution of problems and the emergence of unresolved impediments. In order to address these difficulties and improve the flow of communication, it became necessary to dedicate a specific period within the class so that the teams could meet and discuss the obstacles they were facing. This additional time was crucial to allow team members to share their experiences, identify and discuss the sticking points and collaborate in finding joint solutions. This approach aimed not only to promote greater clarity on the issues at hand, but also to strengthen teamwork and collaborative problem-solving.

In **Week 2**, the instructor moved on with the content, moving on to chained conditional structures, repetition structures, vectors and matrices. Based on these topics, the teams carried out collaborative activities, where each team improved the designer of their blog, attaching images and posts related to the project. In addition, they also wrote their autobiography, in which each student wrote a bit of their story, recounting facts about their life and perspectives after completing the course.

The students also successfully completed Challenge 1 and the list of exercises 2 provided by the Python Leveling instructor, in which they showed no difficulty in completing them. At the end of each week, the students, using the Miro tool, presented what they had most enjoyed doing and what they had least enjoyed during the week, as shown in Figure 8. As a result of this week, it was observed that the students showed confidence in their abilities and no difficulties in this second week, and a good command of the programming language Python, which is a good indication of a positive learning environment. The feedback received from the teams points to teamwork and communication between members, allowing for the exchange of ideas and solutions.

Figure 8. Assessment of activities carried out during the week using the Miro tool.

In **Week 3**, the instructor continued advancing the topics, moving on to handling strings, lists, tuples, functions and handling txt files. With the foundation provided by this content, the teams took part in Challenge 2, which consisted of two stages: problem design by one team (stakeholders) and solution coding by the adjacent team (developers). As a result of the first stage of this challenge, Table 1 shows the challenges proposed by some of the participating teams. During the development of challenge 2, in the second stage (coding the solution), there were times when the teams consulted the stakeholders (adjacent team) to analyze the conformity of the requirements for solving the problem demanded, thus elucidating the requirements.

The completion of Challenge 2 by the teams culminated in the presentation of their results in a "jury court" format: the coding team at the front of the classroom and stakeholders positioned behind for evaluation. All teams carried out their activities without difficulties, demonstrating a good performance according to the demands presented throughout the week. The agility and ease with which the practical activity and the proposed challenge were delivered represent an increase in the students' hard skills. According to the

Team	Proposed Problem
Hopper	Hopper Snack Bar faces a significant problem with long wait times for customers, both in the queue to place orders and in the preparation and delivery of meals. This issue is especially critical during peak hours, when high demand overwhelms the kitchen and staff capacity. Customer dissatisfaction due to the wait time is resulting in lost business, increased complaints, and damage to the company's reputation.
Gates	The company Lans, a small organization, is facing significant challenges with its current passenger record system. Currently, the company maintains these records in manually filled spreadsheets, using pen and paper. This outdated method is not only inefficient but also prone to various issues that can compromise the quality and reliability of the recorded information. Among the identified problems, we emphasize: risk of data loss, susceptibility to damage, tendency for human errors, and operatio nal inefficiency.
Turing	The company AgroTech Ltd. operates in three different products: rice, corn, and wheat flour. The company desires a system that meets its needs, which are: total production in stock for a 30-day period; total production in stock by product; the system should ensure that each product's stock has at least 30% of the total available quantity to ensure sufficient inventory for future sales; generate a detailed monthly report on sales made by product.
Jobs	TecnoSupplies Ltd. is a medium-sized company specializing in the supply and distribution of materials for businesses across various sectors, including construction, manufacturing, and services. Problem to be solved: difficulties in identifying and locating materials; errors in inventory updates; lack of visibility and reporting; inefficiency in managing requisitions and returns.
Linus	The Little Market God is Love, a small local market, faces difficulties in managing its Produce section. Currently, inventory control and purchase processing are done manually, resulting in frequent errors and time-consuming processes. Specific problems include: inventory control, difficulty in updating data, lack of visibility, purchase processing, inefficient shopping cart, and invoice issuance.

Table 1. Examples of some problems that were proposed by the teams in Challenge 2.

teams' feedback, it can be perceived that the development of soft skills such as communication, empathy, teamwork, and leadership are being evidenced and enhanced. The students demonstrated commitment and mutual collaboration.

In **Week 4**, the instructor worked with students on programming in Python using the micro:bit platform. Students were encouraged to program tasks to be performed on this board, as shown in Figure 9. Although the students had never worked with hardware, all students successfully completed all the activities proposed by the instructor, which left them very motivated.

Also this week, two workshops were held, one on Google Colaboratory and the other on the OS library in Python, in which students collaboratively improved their hard skills, being able to put into practice what was taught in the workshops, with the development of a python notebook with the logic of a game within the document. Each team implemented a game with iteration of one or more players, such as: minesweeper, quizzes, Wheel-wheel and Blackjack. Figure 10 presents the million game, implemented by the LuminoPy team.

At the end of the last week, students responded to an evaluation form containing the questions described in Table 2. This form was used so that students could express their

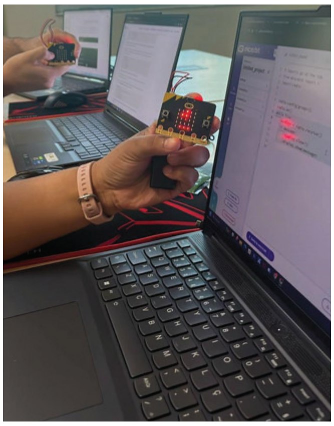

Figure 9. Student programming board micro:bit.

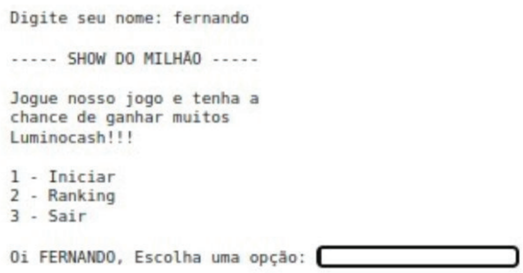

Figure 10. Example of a game prepared by teams.

opinions regarding the format in which the discipline Python Leveling was taught.

Ref.	Questions
P1	The objectives of the course were clear from the beginning:
P2	The course content was relevant and up-to-date:
P3	There was an appropriate balance between theory and practice:
P4	This division of the course contributed to your learning:
P5	The activities conducted by the mentors were helpful and enriching:
P6	The instructor was clear and organized in presenting the content:
P7	The instructor was approachable and available to answer questions:
P8	The instructor was able to engage and motivate the class:
P9	The mentors were clear and organized in conducting the activities:
P10	The mentors were approachable and available to answer questions:
P11	The mentors were able to engage and motivate the class:
P12	There was encouragement for critical thinking and student participation:

Table 2. Form questions.

The graph presented in Figure 11 shows that the majority of students agree that the methodologies adopted in the discipline Python Leveling were extremely effective, contributing significantly to the teaching-learning process in teaching programming.

The graph in Figure 12 highlights the essential role of instructors and mentors in conducting activities throughout the discipline Python Leveling, promoting motivation and better student performance, especially in the development of hard and soft skills.

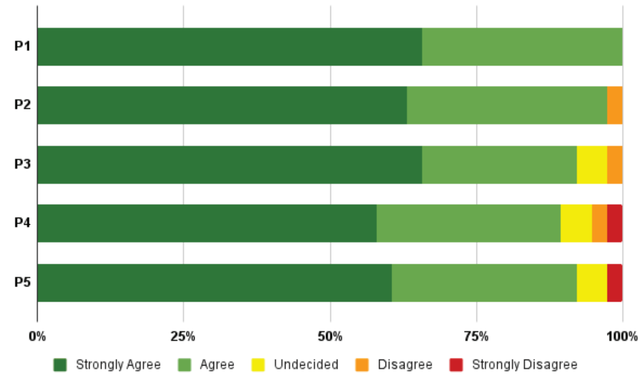

Figure 11. Answer to the questions presented in Table 2 from P1 to P5.

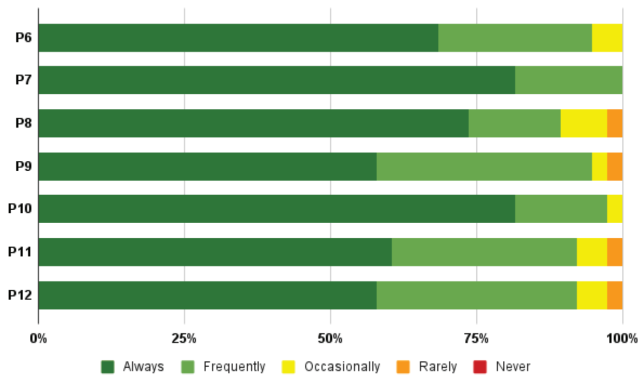

Figure 12. Answer to the questions presented in Table 2 from P6 to P12.

As the weeks went by, the work between the student teams improved and continuous collaboration became increasingly efficient. A soft skill that was highlighted last week was self-knowledge, in which students shared their difficulties in relation to interpersonal interaction. They also reported that activities such as updating the logbook contributed to making them more sociable and communicative. Therefore, it is evident that every week the soft skills were flourishing within the classroom environment and teams.

6 Final Considerations

This article explored the implementation of active methodologies in the teaching of programming, specifically in the subject of Python Leveling, part of the "Advanced Training Course in Hyperautomation" project. The application of approaches such as Problem-Based Learning (PBL) and Challenge-Based Learning (CBL) provided a dynamic, student-centered environment, stimulating the development of technical skills (hard skills) and socio-emotional skills (soft skills).

The model adopted, which combined phases of theoretical instruction and practical mentoring, proved to be effective in promoting collaborative learning and the development of essential skills for today's job market. Group dynamics, the use of tools such as blogs to record progress and the use of real challenges brought the students closer to professional reality and increased their engagement in the learning process.

The results obtained over the four-week course indicate that the methodology employed, combined with constant mentoring, provided a robust educational experience. The teams showed progress both in mastering the Python language and in improving their communication, teamwork and problem-solving skills. Although some difficulty was identified in familiarizing themselves with project management tools, such as GitHub Projects, the continuous use of these tools enabled gradual and significant learning.

In addition, the challenges that simulated real problems were crucial in promoting critical thinking and creativity among the students, allowing the teams not only to apply the concepts learned in class, but also to experience the process of designing problems and coding solutions, as occurs in the development of software.

However, despite the progress made, some limitations were also identified, especially in relation to communication between team members, which had an impact on performance in certain activities. The introduction of moments dedicated to resolving internal problems, such as communication blockages, proved to be an effective strategy for mitigating these challenges and improving collaboration between students.

In conclusion, the results obtained in the Python Leveling subject demonstrate the importance and effectiveness of active methodologies in teaching programming. The balance between theory and practice, coupled with the continuous support of mentors and the application of real challenges, was essential in promoting deep learning and preparing students for future challenges in the field of technology and automation. For future work, we suggest investigating the long-term impact of these methodologies on students' performance in the job market, as well as exploring additional strategies to improve communication and project management within teams.

Acknowledgment

This article is the result of the research and development project ZL Academy Digital Transformation, Agreement No. 01/2024 - Innovation Hub of IFAM, funded by LG Electronics do Brasil Ltda under the terms of Federal Law No. 8.387/1991, in accordance with Article 21 of Decree No. 10.521/2020. We also extend our gratitude to the Manaus Zona Leste Campus of the Federal Institute of Amazonas (IFAM) for their support and encouragement in carrying out this work.

References

[1] L. P. Alves, N. B. Nunes, and A. S. de Bona. 2022. "Construa uma casa" relato de experiência de atividade que trabalha conceitos de programaação através do pensamento computacional. In *Nuevas Ideas en Informática Educativa*, Vol. 16. 364–370.

[2] W. N. Bender. 2015. Aprendizagem baseada em projetos: educação diferenciada para o século XXI. Penso Editora.

[3] M. de C. Borges, S. G. F. Chachá, S. M. Quintana, L. C. C. de Freitas, and M. de L. V. Rodrigues. 2014. Aprendizado baseado em problemas. In *Medicina (Ribeirão Preto)*, Vol. 47. 301–307.

[4] A. de Campos, S. C. Cazella, and E. B. Reategui. 2022. Personalização em experiências de aprendizagem: as potencialidades e os desafios na visão de docentes. In *Nuevas Ideas en Informática Educativa*, Vol. 16. 163–172.

[5] L. Descovi, M. Santos, and F. Herpich. 2022. Chatbot Cerebrum IBM Watson Assistant: Agente conversacional de Equações Algébricas de Primeiro Grau com uma Variável. In *Nuevas Ideas en Informática Educativa*, Vol. 16. 417–422.

[6] S. E. Gallagher and T. Savage. 2023. Challenge-based learning in higher education: an exploratory literature review. In *Teaching in Higher Education*, Vol. 28. Taylor & Francis, 1135–1157.

[7] F. G. de D. Garbin and A. J. C Kampff. 2022. Aprendizagem Baseada em Projetos e Simulação Computacional para o Ensino de Engenharia. In *Nuevas Ideas en Informática Educativa*, Vol. 16. 264–268.

[8] Gabriel V. Barreto Francisco P. Junior Lidiane C. Moura Isabel D. Nunes, Charles Madeira and Midiã L. Paz. 2023. Metodologias ativas e trabalho colaborativo: uma integração entre diferentes abordagens. In *Nuevas Ideas en Informática Educativa*, Vol. 17. 531–534.

[9] A. Melo and R. Abelheira. 2015. Design Thinking & Thinking Design: Metodologia, ferramentas e uma reflexão sobre o tema. Novatec Editora.

[10] V. Pieterse and M. Van Eekelen. 2016. Which are harder? Soft skills or hard skills?. In *ICT Education: 45th Annual Conference of the Southern African Computer Lecturers' Association, SACLA 2016, Cullinan, South Africa, July 5-6, 2016, Revised Selected Papers 45*. Springer, 160–167.

[11] I. M. M. Ramos, D. B. Ramos, J. V. T. Tavares, G. G. Freitas, E. S. Goes Junior, G. R. A. Monteiro, and J. M. de Souza. 2022. ROBÔ-EDU: Robótica Educacional para o desenvolvimento e incentivo do Pensamento Computacional aos alunos do ensino fundamental no município de Parintins-Amazonas. In *Nuevas Ideas en Informática Educativa*, Vol. 16. 193–200.