A Personalized-based Study Guides Tool for Software Engineering Flipped Classes

Necio Veras[1], Lincoln Rocha[2], Windson Viana[2]
necio.veras@ifce.edu.br,lincoln@dc.ufc.br,windson@virtual.ufc.br
[1]Federal Institute of Ceará
[2]Federal University of Ceará

ABSTRACT

Teaching Software Engineering (SE) poses challenges due to diverse levels of student knowledge. Active methods like Flipped Classrooms and Adaptive Learning can enhance SE learning. Flipped Classrooms involve pre-class study and in-class problem-solving, and Adaptive Learning tailors content to individual student needs. Our research mixes these approaches by introducing a personalized study guide tool. It's a web-based platform for creating customized study guides for SE Flipped Classes. The tool adapts the guide content based on student's performance and engagement. Within the tool, teachers set rules for content adaptation, focusing on class preparation and gaining insights into student progress. We evaluated the personalized study guides from three perspectives: perceptions, motivations, and achievements. The evaluation in a software requirements lesson with 22 students showed a significant improvement in scores and received positive feedback from students. Our tool has the potential to support the enhancement of SE education through personalized study guides in Flipped Classrooms.

KEYWORDS

Flipped classroom, Software engineering teaching, Tool, Personalized learning

1 INTRODUCTION

The software industry frequently grapples with challenges when seeking professionals possessing technical skills in Software Engineering (SE). As students embark on their professional journeys, they encounter hurdles concerning aligning the skills acquired during their academic training with the practical demands imposed by the professional environment. This contradiction often creates a gap between theoretical knowledge and its practical application, making it difficult the transition into the professional environment. In this context, Akdur (2022) advocates for improving efforts from both academics and the industry to bridge this gap. Active Learning (AL) methods and practical activities that simulate real problems and situations in the software ecosystem can help reduce this gap.

Ouhbi and Pombo report several AL methods implemented in SE education that aim to improve students' motivation and engagement [32] (e.g., problem-based learning, gamification, and flipped classroom). In many domains, including Computer Science Education, AL methods stand out as an approach to encourage student interaction, improving knowledge building instead of receiving content passively from instructors [7]. This approach holds out, mainly because passive students lose concentration between 10 and 15 minutes after starting a 50-minute class [31]. In this sense, the literature reports many strategies to promote active learning to minimize the risks of students' low learning [3, 18, 19, 25, 30, 43]. For instance, the risk of low learning is reduced by selecting short activities, properly structured [3]. Involving students in a series of challenges on current topics also allows teachers to conduct the discussion flow, which turns students into active participants [19].

Following this trend, our research integrates Adaptive Learning [1] into Flipped Classroom [2] methods by introducing a personalized study guide tool to assist SE teaching. Our web-based authoring tool creates personalized study guides dynamically by adapting them according to the student's performance and engagement levels.

This paper presents our tool for supporting software engineering flipped classes and describes how we evaluated it considering the following research question: "*What is the impact of using the personalized-based study guide tool in a software engineering lesson on students' perceptions, motivation, and academic achievements?*". We organized the research methodology into three phases: Problem, Solution, and Evaluation. The problem phase encompasses a literature discussion on the flipped process in software engineering teaching to determine the initial insights, desired features, and design principles for building a flipped classroom support tool. The solution phase represents the product (tool) development as a web-based tool. Finally, the evaluation phase evaluated the tool during the software requirements class.

The structure of the remaining paper is organized as follows. First, Section 2 provides a comprehensive theoretical foundation on the concepts of the Flipped Classroom and Adaptive Learning. Subsequently, Section 3 outlines our research methodology, including insights from a survey conducted with flipped classroom researchers and developing a personalized-based study guides tool. Section 4 outlines the study design and materials employed to evaluate the proposed tool, along with a detailed description of a flipped classroom session focused on software requirements. In Section 5, we present the results of our analysis concerning student perception, motivation, and academic performance. Section 6 is dedicated

[1]Adaptive learning refers to an educational approach that leverages technology and data to personalize the learning experience for each student.
[2]A Flipped Classroom is an active learning methodology focused on student engagement [36], which recommends that students come to the class after completing significant preparatory work.

The author(s) or third-parties are allowed to reproduce or distribute, in part or in whole, the material extracted from this work, in textual form, adapted or remixed, as well as the creation or production based on its content, for non-commercial purposes, since the proper credit is provided to the original creation, under CC BY-NC 4.0 License.
EduComp'24, Abril 22-27, 2024, São Paulo, São Paulo, Brasil (On-line)
© 2024 Copyright held by the owner/author(s). Publication rights licensed to Brazilian Computing Society (SBC).

to discussing our research questions, key insights, and the limitations of our study. The threats to validity are discussed in Section 7, and the concluding remarks are presented in Section 8.

2 BACKGROUND

2.1 Flipped Classroom

The Flipped Classroom (or Inverted Classroom) method advocates that most learning content delivery should occur before class, in contrast to traditional approaches [8]. Face-to-face classes are dedicated to more meaningful learning activities collaboratively. Practical activities in SE teaching, such as software design, programming in labs, and classroom debates, are examples of those activities that fit the in-class approach of FC [8].

FC allows the instructor to better deal with distinct levels of students, trying to homogenize their knowledge before in-class time [8]. With this model, the FC changes instruction to a learner-centered model in which teachers reserve in-class time to explore topics in greater depth. Lecture-style classes are no longer the unique way to take profit from classroom time. In an FC, the teacher engages students through other learning activities, such as discussion, problem-solving (proposed by students), practical activities, guidance, and game-based tasks [1].

Flipping a classroom is not a one-size-fits-all approach, but Bergmann and Sams method presents a standard model (2012). Teachers can prepare guidelines for content delivery and present this material to students before meeting face-to-face. A study guide can be used to curate digital learning materials like videos, podcasts, and content-related texts (such as book chapters, white papers, and scientific papers) alongside self-assessment tests (like Google Forms quizzes) [5]. Once presented with the study guide, learners can explore the content in-depth, building on their prior knowledge. This may involve conducting online research or participating in online discussions to help them prepare for the in-class portion of the course [13]. However, it's important to note that using a flipped classroom approach can be less effective if the instructional content is of low quality or too time-consuming, which may decrease student engagement [28].

In [44], we identified several challenges associated with implementing the flipped classroom in SE education, including the overload and time constraints faced by instructors, difficulties in class preparation, substandard quality of educational materials, significant effort and time investment required from students, issues related to learning assessments, absence of student feedback, and student heterogeneity in both in-class and out-of-class activities. The tool we propose in this paper addresses some of these challenges.

2.2 Adaptive Learning

A challenge concerning SE learning is students' different characteristics, paces, prior knowledge, and motivations to study [6]. Especially about the way they explore instructional materials. The same educational materials and methods are presented to students, ignoring their peculiarities, weaknesses, or potential. The one-size-fits-all content model may cause different results for each student. When preparing instructional materials, the teacher should use technologies to customize the content according to the student's profile or performance [21]. This content personalization saves extra effort for the student to organize the sequence of study and makes learning more efficient [16]. Adaptive learning systems consist of multiple components, which together allow instruction to be tailored to the needs of individual learners [42]. Adaptation can occur according to the student's prior knowledge, adapting the content to be delivered.

To ensure a smooth shift from traditional teaching to the FC active method, it is crucial to have technologies providing reliability, interactivity, and collaboration in a student-centered scenario. Performing real-time, adaptive learning monitoring are examples of features that enable teachers to advise their students efficiently [17]. In this sense, an adaptive system can deliver online instructional materials, where the student's interaction with the previous content determines the nature of the materials provided later.

Blending the FC method with adaptive learning to personalize out-of-class study materials can increase the method's effectiveness for teaching, as individuals face different possibilities and difficulties during the learning process [9]. For instance, Cevikbas and Kaiser propose an approach to personalized content according to students' personal needs, interests, expectations, skills, strengths, and weaknesses. The adaptation is based on analytical data reports and provides immediate feedback and real-time assessment. The technology helped instructors monitor their students' learning progress and adapt their teaching. However, challenges for students and instructors are described, for example, technology glitches, time-consuming activities, and the need for familiarity with new learning environments and tasks.

2.3 Related works

In 2008, Gannod reported the use of flipped classroom method in SE teaching [13]. Subsequently, numerous studies have employed this method for teaching various SE topics, as evidenced by the works of [10, 14, 15, 22, 33, 34, 41].

Prior research has yet to be found on implementing the flipped classroom in software engineering by integrating an adaptive learning approach. Our study stands apart from the works discussed in this paper by incorporating personalized learning into the flipped classroom model. Table 1 presents some differences between the related works and our study.

3 RESEARCH METHOD

Our research methodology was structured into three distinct phases: Problem, Solution, and Evaluation, with each phase designed to address specific research questions, actions, and research types. In Phase 1 (Problem), which involved an exploratory and descriptive study, the primary objective was to answer our first research question (RQ1): *What are the characteristics an authoring tool should have to help SE professors prepare Flipped Classrooms?* We describe this phase in 3.1. Subsequently, in Phase 2 (Solution), we aimed to answer our second research question (RQ2): *Is creating a Personalized Study Guide-based Flipped Classroom Tool to teach Software Engineering feasible?* The proposed tool is presented in 3.2. Finally, in the third phase (Evaluation), we address our third research question (RQ3): *What is the impact of using the personalized-based study guide tool in a software engineering lesson on students' perceptions,*

Table 1: Comparing related works

Paper	Area of SE	Evaluations	Instructional materials (at home)	Adaptive strategy
[13]	General	Engagement, performance	Podcast, audio and video lesson	None
[22]	Management	Performance	Video	None
[33]	Construction	Acceptance, engagement, performance	Animation, video, web resource	None
[34]	General	Engagement, performance	Video, text	None
[14]	Professional practice	Engagement, performance	Web resource	None
[15]	General	Performance	Video, quiz	None
[11]	General	Engagement, performance	Video	None
[41]	Software design	Acceptance, performance	Video, web resource	None
This	Software requirements	Acceptance, motivation, engagement, performance	Video, text, web resource, online form	Personalized learning

motivation, and academic achievements? Section 4 presents this third phase.

3.1 Survey With FC researchers

To develop the tool, we conducted a descriptive survey to gather insights and data about specific features that would underpin the design principles of the tool to support Flipped Classrooms in Software Engineering (SE). Initially, we identified tool features based on the literature review described in [44] After, we incorporated additional features gleaned from a survey with FC researchers in SE.

We designed an electronic questionnaire[3] to serve as a data collection instrument. The questionnaire items were categorized into six contexts: challenges, difficulties, advantages, drawbacks, resources, and evaluation. We identified ninety-six researchers from 38 papers listed in [44] and sent eighty-two emails to SE researchers. Fourteen email addresses needed to be updated, and one researcher could not be contacted. Twelve researchers (14.63%) answered the survey. Each author was associated with a distinct paper in the review. Consequently, we collected responses from 12 of the 38 papers (31.58%).

The responding authors represent ten countries. The USA was the most significant country regarding the number of researchers (three). The other countries had one answer each (India, China, UK, Austria, Spain, Sweden, Argentina, Ukraine, and Serbia). The emails and answers were sent and received between November and December 2021. As a result, the insights gathered regarding the challenges and difficulties of implementing Flipped Classrooms contributed to characterizing the tool features. Table 2 outlines the main features mapped by this study.

3.2 A Personalized-based Study Guides Tool

The tool is a web-based authoring tool with graphical resources used to build study guides (out-of-class studies) and collect feedback on students' data (performance and engagement). We implemented the fourteen features described in Table 2. The tool allows the teacher to insert activities individually, organizing them in a visual study guide like a learning path (outside the classroom). The main goal was to build a platform where learners could increase their knowledge through study guides in contextualized software engineering activities. At the same time, simultaneously, the teachers can generate high-quality learning resources for their students.

[3]Accessible at URL https://forms.gle/S5qRDoX9wtN2uvnD6

Table 2: Tool features

ID	Feature
F1	Teachers and students must be authenticated by the tool
F2	Teachers can add or create virtual classrooms
F3	Teachers can add students to a virtual classroom
F4	Students can access a virtual room
F5	Teachers can manage (create, list, change, archive, publish) study guides for flipped lessons
F6	The study guide should be made with a learning path and built using a visual interface (Usability)
F7	The study guide can contain content by video, text (PDF), forms, or links
F8	The teacher can create a study guide using personalized points according to student performance or engagement
F9	The teacher can reuse a study guide in other flipped classes
F10	The teacher can monitor student activities and performance and engagement from a flipped lesson
F11	Students can view the instructional material for their flipped lessons
F12	The student can view his flipped classroom activities' progress
F13	Students can do activities from a study guide and add individual feedback
F14	The students can use the tool from a mobile device (Usability/Responsiveness)

Inspired by our prior experiences using flipped classes (showed in [3–6]), our FC approach to teaching Software Engineering uses three steps: (1) the first step happens before class; (2) the second step occurs during the practical class; and (3) the third step occurs post-class. Before class, professors propose study guides (e.g., a list of videos, a research paper, and read suggestions) to students studying at home. During the practical class, professors start with brief explanations about class content. Afterward, he dedicates some minutes to answering FAQs brought by students. After that, professors conduct practical activities (like quizzes, software games, project planning, requisite meets, programming tasks, and brainstorms). Finally, after the practical class, a review activity is worked on by the professor.

Before the practical class, the process of the flipped class begins with the production of the study guide to be sent to the students. Using the proposed tool, the teacher build a study guide using a graphical web environment with drag-and-drop resources to elaborate the learning path to be fulfilled by students. The course instructor structures the content and defines students' learning paths in each flipped class. Subsequently, students received a link for knowledge assessment through the academic system and submitted their responses.

In the next phase, accessing the web tool, students can follow the guidelines, for instance, to study the topic contents and resolve evaluation quizzes. Students are involved in practices related to the content under study. After the practical class, a review lesson is prepared and sent using the tool to review the content. Our tool allows the teacher to add personalization points to adapt the learning content.

In the tool, the available points are *if/else*, *conjunction*, *disjunction*, *optional* and *student decision*. In the *if/else* point, the teacher sets a condition to be evaluated by the tool during student studies. An example rule: *if* the student gets a grade higher than 7 in a form *then* will follow the upper path, *else* will follow the lower path. The point *conjunction* is similar. However, the teacher programs have two conditions that need to be met. An example rule for the *conjunction*: *if* the student viewed 80% of the lesson video *AND* the student got a grade higher than seven *then* the path will be released. The rules for the point *disjunction* resemble those of the point *conjunction*, but only one of the conditions must be satisfied to release the path. The point *optional* has a single condition to release the path. The teacher can use this point to, for example, challenge the students. Finally, the point *Student Decision* allows the student to choose one of two possible paths. The teacher should explain/comment on the paths.

Next, the tool enables the teacher to monitor the activities performed by the students in a flipped class. Teachers can view the performance of each student and the individual records (logs), displaying logs of a student's interactions with an instructional content (video, form, text-based, or link).

Figure 1 presents a snapshot of student data across three instructional contents within a study guide. The depicted example showcases the student's complete video viewership, engagement with an external content link, and a 2.0 score attained in the electronic form (exercise).

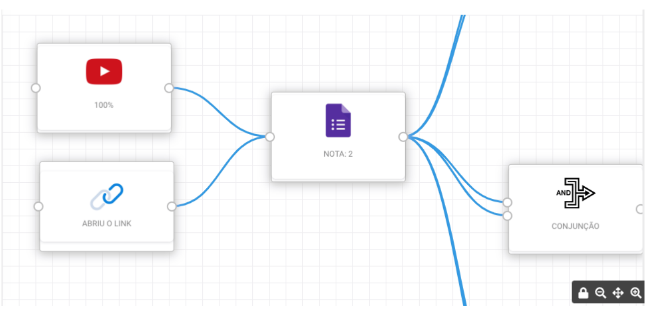

Figure 1: Monitoring of a student's performance by the teacher (in Portuguese)

A video showcasing the tool's usage from the students' perspective is available at https://youtu.be/54tsr8d1yY8. Additionally, a video illustrating student monitoring from the teacher's viewpoint is accessible at https://youtu.be/vR9Zx3tESB8.

4 TOOL EVALUATION

4.1 Study Design

This study aims to measure the effects of personalized flipped classes from the student's point of view. In the literature, many approaches have attempted to evaluate the impact of the FC method. For example, authors often compare content coverage, engagement, student grade performance, perceptions of teaching, learning impact, FC format, and self-efficacy [20, 26, 27, 29, 40]. We designed our evaluation inspired by [40], which sought to investigate the impact of the flipped classroom model on student performance, and by [38], which reported a study about the impact of the flipped classroom on student performance and retention.

Our study examined the impacts of the personalized flipped classroom in SE teaching, we followed the recommendations described in [46]. Impacts were measured through students' perceptions [23] (cognitive presence, learner presence, and technology use), motivations [35] (control of learning beliefs, self-efficacy for learning and performance, test anxiety), and achievements (engagement and score tests) during lessons.

Data collection for this study happened in September 2023. The author of this thesis also served as the Software Engineering (SE) course instructor. Participants were technical degree students in informatics at the Federal Institute of Ceará, Brazil. The evaluation involved 22 informatics students (technical degree). It is essential to note that all participants were beginners with no prior exposure to Software Engineering concepts.

Three dimensions were simultaneously observed during proposed activities for students: students' (1) perception, (2) motivations, and (3) achievements.

- **Students' Perception:** refers to the cognitive presence, learner presence, and technology use concerning the flipped class tool used. In other words, the students' perceptions during the method application and their feelings regarding critical and creative thinking, self-regulation of learning, interaction among students and instructors, and acceptance of technology use [23].
- **Students' Motivations:** refers to three components of motivation, such as (1) control of learning beliefs, (2) self-efficacy for learning and performance, and (3) test anxiety [35]. In other words, (1) if students believe their efforts make a difference in their learning, they may study more strategically and effectively. (2) Self-efficacy refers to judgments about their ability and confidence to perform an activity. (3) Students' anxiety and negative thoughts may disrupt performance.
- **Students' Achievements:** refer to flipped class tool impacts on student engagement and learning [20]. The measure will characterize the students' performance during flipped class activities.

In terms of ethical considerations, we carefully thought about our evaluation with users. We found that these activities did not pose any health risks, confidentiality issues, data security problems, or other vulnerabilities to the participants. After analyses, we anonymized all collected data during the study. Our study involved only participants who were above 18 years old. All investigations were conducted in a standard university environment, specifically the classroom. All participants were given an informed consent form and could decide whether to participate in the study. Furthermore, the research was conducted without any funding.

4.2 Materials

We utilized the tool to create one study guide and implemented it during a software requirement lesson (detailed in Section 4.3). The study guide employed adaptive learning techniques through a personalized flipped classroom. We develop a series of activities in the software requirements lesson context, encompassing out-of-class tasks, in-class activities, and final exercises. These activities were strategically implemented before and after the in-class sessions to gather data on students' academic performance and achievements. These activities' overarching objective was to convey theoretical software engineering content (outside of the traditional classroom setting) and practical application (during in-class sessions).

Quantitative and qualitative data were collected for the evaluation. First, data from students' perceptions (cognitive presence, learner presence, and technology use) were obtained. Then, their motivations (control of learning beliefs, self-efficacy for learning, and performance) and achievements (engagement and score tests) were collected. A questionnaire was created using the Google Suite platform to collect students' perceptions and motivations. The evaluation form elaborated questions using the Likert scale [24]. In addition, the studies [23] and [35] inspired the questions for perceptions and motivations, respectively.

We gathered self-reported feedback by administering surveys after the flipped classroom sessions to assess students' perceptions and motivation. We evaluated students' perceptions using a set of twenty-two questions and their motivation with seventeen questions listed in the next section. The methods employed to collect information about student achievements encompass active participation, the outcomes generated, and the grades earned by the students.

The evaluation of student engagement involved the assessment of their performance through both formative and summative activities. Additionally, a consent form was developed to clarify the roles of the students in the evaluation process. Prior to the commencement of the pre-test, the consent form was presented to the students, offering a detailed explanation of the evaluation format. Students were given the choice to participate or decline involvement in the study. All the instruments developed for the evaluation can be accessed through the following link: https://nupreds.ifce.edu.br/necio/educomp24/educomp24-materials.pdf.

Furthermore, we designed two tests (pre- and post-tests). We used them to gauge the learning impacts of the evaluation. Both tests contained identical questions, each comprising multiple-choice items with five options, of which only one option was correct. The pre-test evaluated the student's prior understanding of the subject matter before the experiment commenced.

In contrast, the post-test assessed the students' knowledge after the evaluation. After completing the practical activities, the post-test was administered to students and addressed any questions during the exercises. The test results served as a baseline measure, enabling the quantification of how much knowledge students had acquired and retained throughout the evaluation.

4.3 Software Requirements Flipped Lesson

The evaluation process of this paper was inspired by [19]. Before starting the flipped class, the teacher informed the students about the teaching model used for SE education. Then, the students must accomplish the activities, although their scores would not impact their course grades. Our software requirements flipped lesson was based on three steps (illustrated in Figure 2).

The instructor explained the questionnaires to students before the **Step 1**. After that, the students were invited to participate in the pre-test and self-efficacy questionnaire. The instructor distributed the pre-test and asked them to answer the questions individually, without querying any additional knowledge source (e.g., the Internet and books). The instructor prepared a study guide with lesson content, including videos, reading materials, online quizzes, and self-assessment exercises (with automatic feedback).

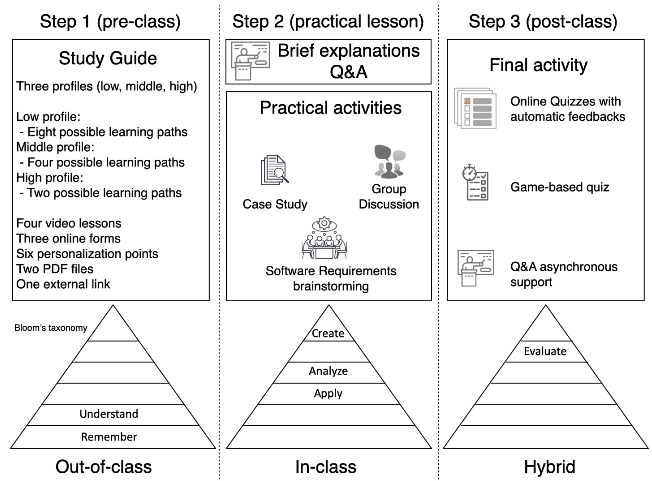

Figure 2: Software requirements flipped class.

4.3.1 Step 1: pre-class (one week). The flipped class begins with home studies through a personalized study guide. We have crafted an adaptive study guide tailored to three distinct student profiles. These profiles are automatically discerned through performance assessment in a pre-test phase. The study guide encompasses a spectrum of up to eight potential learning paths, contingent upon the specific profile of each student. Notably, students with greater challenges with the subject matter are furnished with more comprehensive instructional materials. The study guide incorporates sixteen educational components: four video lessons, three online forms, six personalization points, two PDF files, and one hyperlink to an external content resource. In this first step, students should understand the essential theme of the lesson. In line with the "Understanding" and "Remember" dimensions of Bloom's taxonomy [12], the study guide lesson goals are to:

- **Identify** the initial aspects of software requirements,
- **Define** the requirements discovery process, and
- **Elaborate** a case study-based guide for interview.

The study guide described in step 1 starts with the student watching a video lesson and answering an exercise (form). Then, the learning path personalization happens according to the student's **performance**. There are three profiles for personalization:

1. **Low grade** (grade ≤ 4): students who did not identify the initial aspects of requirements engineering,
2. **Middle grade**(grade > 4 AND > 8): students who did not define the requirements discovery process and
3. **High grade** (grade ≥ 8): the student who performed well in the previous two items.

Upon identification of their profile, the student exercises the prerogative to opt for the learning material they deem suitable according to the profile identified. When students fail to meet the requisites of the "high grade" profile, they must navigate through the preceding pathway iteratively until they attain a grade that satisfies the advancement criteria. It is incumbent upon all students to ascend to the "high grade" profile and culminate their independent study endeavors by completing a comprehensive case study form.

This strategic approach is underpinned by the overarching objective of fostering educational equity among students. By providing additional content to those students with comparatively lesser proficiency, the model endeavors to level the academic landscape. The strategy thus harmonizes instructional distribution based on individual knowledge disparities. Figure 3 illustrates the study guide, modeled using the Business Process Model and Notation (BPMN) semantics [45].This example of a personalized study guide has eleven instructional contents and six personalization points. Students in the "low grade" profile will have eight possible learning paths, while those in the "middle grade" profile will have four paths. The "high grade" profile has two possible paths. Full details of each instructional content are available at https://nupreds.ifce.edu.br/necio/educomp24/educomp24-materials.pdf.

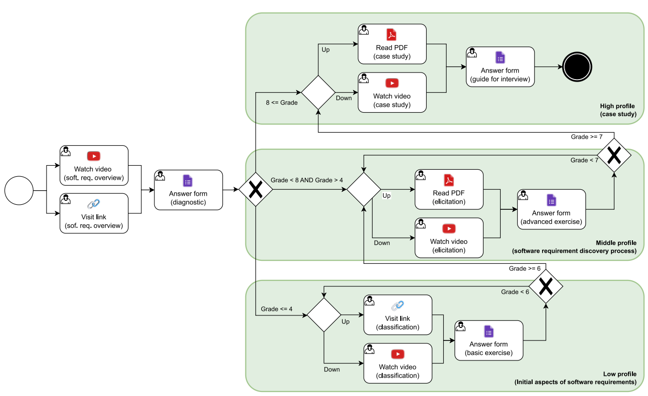

Figure 3: Software requirements study guide modeled from BPMN-model

4.3.2 Step 2: Practical class (six hours). After studying at home, the flipped classroom's second step is a practical lesson in the classroom. This phase is orchestrated as a two-step practical lesson spanning two distinct sessions [4]. The first class starts with a brief explanation, and then the teacher instigates Q&A on the initial aspects of software requirements. This discourse addresses queries that may have arisen during the home study phase. Subsequently, students are grouped to analyze a case study grounded in real-world scenarios. This moment entails the formulation of an interview guide designed to elicit software requirements through an investigative approach.

In the second practical class, the students should, in groups, classify software requirements from real-world-based case studies. The practical lesson goal is to apply the concepts studied at home from the analysis of the cases to discover (create) software requirements. In alignment with the dimensions of Bloom's Taxonomy - "Create", "Analyze", and "Apply" - the goals of these practical lessons are articulated as follows:

a) **Analyze** software scenarios rooted in real-world case studies.
b) **Discover** latent software requirements inherent within the context.
c) **Classify** identified software requirements based on functional or non-functional.

This practical phase serves as a conduit for students to materialize their theoretical insights, thereby fostering a practical comprehension of software requirements within a tangible, real-world-oriented domain.

4.3.3 Step 3: Post-class (two hours). After the in-class practical lesson, the third phase of the flipped classroom model happens as an evaluation process centered around a final activity. This moment can be done during the next meeting in the classroom or using an asynchronous way (out-of-classroom). In the context of the software requirements flipped classroom example, the third phase unfolds across two stages, collectively aimed at gauging and assessing students' acquired skills.

The initial stage occurs between the end of step 2 and the next in-class lesson. During this period, students are prompted to engage with an online form, giving them the autonomy to respond from their home environments. The inherent advantage of this approach lies in the provision of automated feedback, fostering an asynchronous learning experience (the instructor offers asynchronous support). In the subsequent stage, situated within the in-class session, students participate in a quiz-based activity integrated with a game-based format. This interactive gamification strategy adds an element of engagement and dynamism to the evaluation process. The objective of this third phase is to systematically **evaluate** the degree to which students have internalized and retained the acquired knowledge.

5 RESULTS

5.1 Student's perception analysis

The students' perceptions were analyzed through twenty-two questions presented to the students using the Likert scale. The questions were categorized into eight items regarding cognitive presence, eight concerning learner presence, and six regarding technology acceptance. Each question was associated with five Likert-scale-based responses: Strongly Agree, Agree, Neutral, Disagree, and Strongly Disagree. These responses were correspondingly mapped

[4]Session plan 1: https://www.dropbox.com/scl/fi/3b8ar2s5wvvft0keia0m7/Atividade-pr-tica-1-Roteiro-de-Entrevista.pdf?rlkey=643c60agr6ce0oew8q73cqgpc&dl=0
Session plan 2: https://www.dropbox.com/scl/fi/loz85an35rj7fm2o4ti6a/Atividade-pr-tica-2-Descoberta-e-classificacao.pdf?rlkey=t2rxar07cxqjkbs2birfjwl59&dl=0

to 5 points, 4 points, 3 points, 2 points, and 1 point, respectively. The questionnaire items were the following:

Cognitive presence

Q1 : Problems posed increased my interest in SE issues
Q2 : I felt motivated to explore SE content related questions
Q3 : I utilized a variety of information sources to explore problems posed in this flipped lesson
Q4 : The activities proposed helped me resolve SE content related questions
Q5 : Combining new information helped me answer questions raised in practical activities
Q6 : Learning activities helped me construct explanations/solutions
Q7 : Reflection on SE content and discussions helped me understand fundamental concepts in this flipped class
Q8 : I can apply the knowledge created in this flipped classroom to my work or other non-class related activities

Learner presence

Q9 : When I studied for the SE activities, I set goals for myself in order to direct my activities in each study period
Q10 : I tried to change the way I studied to fit the activity requirements and the flipped classroom style
Q11 : I worked hard to get a good grade even when I was not interested in some topics
Q12 : I tried to think through a topic and decide what I am supposed to learn from it rather than just reading materials or following directions
Q13 : Before I began studying, I thought about the things I will need to do to learn
Q14 : When studying for the activities I tried to determine which concepts I did not understand well
Q15 : When I was working on learning activities I stopped once in a while and went over what I have done
Q16 : In general, I felt confident using the technologies associated with the out-of-class activities

Technology acceptance

Q17 : It was easy for me to find and access the out-of-class materials associated with flipped classroom activities in the tool
Q18 : In general, technologies associated with the out-of-class activities were easy to use
Q19 : The technologies used for the out-of-class activities positively interfered with my learning ability
Q20 : Using the tool technology can increase my study productivity
Q21 : Using tool technology in flipped classrooms is a good idea
Q22 : I would like to use the tool in other flipped classes whenever possible

The results about students' perception are presented graphically in Figure 4.

We obtained an average acceptance rate of 85.25% for cognitive presence. Among the items, item Q3 had the lowest acceptance rate (73%). This rate may suggest that students perceived the available information sources as sufficient or suitable for exploring the proposed problems in the flipped classroom and, therefore, did not feel the need to explore other varied sources beyond those available in

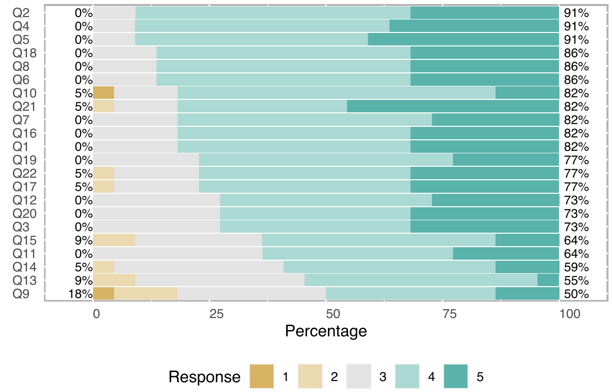

Figure 4: Perception's data

the study guide. Regarding learner presence, we observed an average acceptance rate of 66.12%. Within this dimension, we analyzed the level of self-management and organization among students during their study activities, attitudes, and self-confidence. The lowest acceptance rate was in item Q9. This rate may suggest the low ability of some students to set goals for themselves, plan their study activities, and exercise self-discipline during the preparation period for the practical class. Regarding the technology acceptance rate, we had an average acceptance rate of 78.66%. The lowest-rated item in this dimension was Q20 with a 73% acceptance rate. The high overall acceptance indicates that students recognized the value and utility of the tool in optimizing their study processes.

5.2 Student's motivation analysis

Similarly to the analysis of perceptions, we assessed students' motivation through seventeen Likert-scale items comprising five levels. We categorized the items into three motivational aspects, with four items relating to the control of learning beliefs, eight items focusing on self-efficacy for learning and performance, and five items concerning test anxiety. The seventeen evaluated items were as follows:

Control of learning beliefs

Q23 If I study in appropriate ways, then I will be able to learn the material about SE
Q24 It is my own fault if I don't learn the material about SE
Q25 If I try hard enough, then I will understand the SE material
Q26 If I do not understand the SE material, it is because I did not try hard enough

Self-efficacy for learning and performance

Q27 I believe I will receive an excellent grade in this flipped class
Q28 I'm certain I can understand the most difficult material presented in the SE readings
Q29 I'm confident I can understand the basic concepts taught in this SE course
Q30 I'm confident I can understand the most complex material presented by the instructor in this SE course
Q31 I'm confident I can do an excellent job on the assignments and tests in this SE course
Q32 I expect to do well in this flipped class

Q33 I'm certain I can master the skills being taught in this flipped class
Q34 Considering the difficulty of this SE course, the teacher, and my skills, I think I will do well in this flipped class
Test anxiety
Q35 When I take a test I think about how poorly I am doing compared with other students
Q36 When I take a test I think about items on other parts of the test I can't answer
Q37 When I take tests I think of the consequences of failing
Q38 I have an uneasy, upset feeling when I take an exam
Q39 I feel my heart beating fast when I take an exam

We gathered responses from the students before and after the flipped classroom to assess the tool's impact on their motivation. The compiled student motivation responses in Table 3 were analyzed to identify variations. The objective was to ascertain the significance of differences using the results of the Mann-Whitney test (confidence level at 95%), given that the normality test indicated that the data did not follow a normal distribution. The bolded P-values offer substantial evidence to support the conclusion that the data exhibit statistical dissimilarity.

Table 3: Motivation's data

	Pre-class			**Post**-class			p-value
	Mean	SD[1]	Rate(%)	Mean	SD[1]	Rate(%)	
Q23	4.27	0.55	95.5	3.95	0.72	72.7	0.0631
Q24	3.09	1.19	36.4	3.41	0.85	45.5	0.1729
Q25	4.09	0.81	81.8	3.77	0.61	68.2	**0.0497**
Q26	3.41	1.05	45.5	3.28	1.02	45.5	0.3703
	Average(%)		64.8	Average(%)		58.0	0.1118
Q27	3.68	0.57	63.6	3.63	0.79	54.5	0.3926
Q28	3.45	0.86	40.9	3.68	0.84	54.5	0.1825
Q29	4.27	0.70	86.4	4.09	0.81	72.7	0.2361
Q30	3.64	0.95	50.0	3.59	0.79	50.0	0.4503
Q31	3.77	0.75	68.2	3.41	0.66	40.9	**0.0394**
Q32	4.36	0.58	95.5	3.68	0.64	68.2	**0.0005**
Q33	3.91	0.68	72.7	3.68	0.78	59.1	0.1683
Q34	3.68	0.72	54.5	3.91	0.75	77.3	0.1094
	Average(%)		66.5	Average(%)		59.7	0.0515
Q35	3.14	1.24	40.9	2.95	1.09	31.8	0.3000
Q36	3.68	0.99	63.6	3.41	1.00	54.5	0.1809
Q37	3.68	1.21	72.7	2.81	1.18	36.4	**0.0084**
Q38	3.23	1.31	45.5	2.32	0.78	4.50	**0.0045**
Q39	3.09	1.23	45.5	2.36	0.90	9.10	**0.0170**
	Average(%)		53.6	Average(%)		27.3	**4.228e-05**

[1]Standard deviation.

Regarding control of learning beliefs, we noted a significant reduction in item Q25. This decrease suggests a possible lack of effort in some students regarding their understanding of the material or a gap in the clarity of the produced study material. The reductions in items Q31 and Q32 support the Q25 result, showing students' self-awareness of their shortcomings in efforts during out-of-class preparation. However, the most significant reduction was related to test anxiety. Three out of the five analyzed items substantially reduced student anxiety.

5.3 Student's achievements analysis

We analyzed students' achievements from two perspectives: (1) engagement and performance in the activities included in the study guide (pre-class) and (2) comparative performance between the pre and post-tests.

5.3.1 Students' engagement and exercise performance. We measured students' engagement by the percentage of content viewed in each video and the access (reading) to each PDF/Link. We computed students' performance in the study guide by the average grades obtained in the exercises. Table 4 presents textually the students' data regarding the study guide presented by Figure 3.

Table 4: Students' engagement data

ID	$video_1$ score	$link_1$ engag	**Diagn. grade**	**Profile**	**Learn. path engage. (%)**	**Grade avg**
S1	100	Not	3.0	Low	100	8.6
S2	95	Visited	3.0	Low	98.6	8.0
S3	96	Visited	2.0	Low	33.3	5.6
S4	100	Visited	2.0	Low	100	9.3
S5	100	Visited	2.0	Low	100	7.7
S6	100	Visited	3.0	Low	33.3	1.7
S7	100	Visited	3.0	Low	100	8.7
S8	100	Visited	3.0	Low	100	9.0
S9	100	Visited	2.0	Low	80	8.0
S10	100	Visited	3.0	Low	100	9.3
S11	100	Visited	3.0	Low	83.3	7.0
S12	100	Visited	2.0	Low	100	8.8
S13	94	Visited	1.0	Low	100	8.5
S14	91	Visited	3.0	Low	66.7	6.0
S15	100	Visited	1.0	Low	100	7.2
S16	100	Not	1.0	Low	66.7	7.1
S17	100	Visited	2.0	Low	100	7.7
S18	97	Visited	4.0	Low	98	8.0
S19	100	Visited	1.0	Low	33.3	0.7
S20	100	Visited	3.0	Low	100	8.4
S21	4	Visited	2.0	Low	10.3	7.3
S22	87	Visited	3.0	Low	89.4	8.3

Figure 5 shows the learning path undertaken by the students. All twenty-two students were initially classified with the "Low grade" profile. Among these, nineteen completed the study guide. Within the "Low grade" profile, ten students needed to revisit and study additional materials, repeating the basic exercise until they achieved the minimum grade required to progress to the next level, denoted as "Middle grade". Meanwhile, within the "Middle grade" profile, four students had to retrace their steps and repeat the advanced exercise.

5.3.2 Students' performance (pre and post-tests). We used a pre-test to assess students' pre-existing knowledge of software requirements engineering before the flipped classroom started. Approximately two weeks after the beginning of the class, we administered the same test to evaluate students' knowledge after the completion of the flipped classroom (post-test). The graph in Figure 6 depicts the scores in both tests.

After collecting the scores for the two tests, we used the independent samples t-test as a method of analysis. The results of the

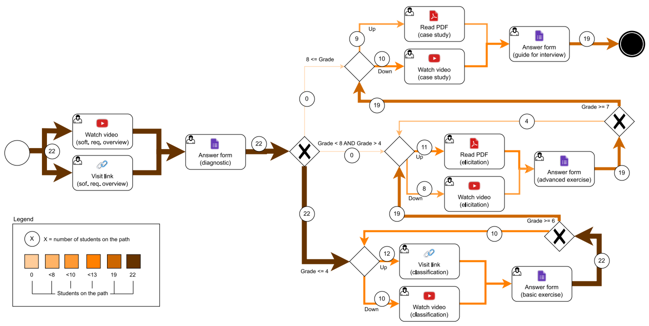

Figure 5: Student's data regarding study guide

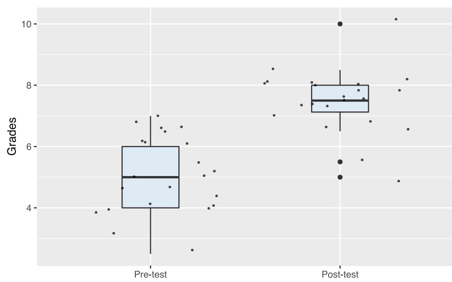

Figure 6: Students' grades in the pre- and post-tests

analysis are shown in Table 5. The results showed that the students obtained significant differences in learning (t = -10.58, p < 0.001). Based on the results of the analysis, it is suggested that the use of personalized teaching scripts during a flipped classroom improves students' cognitive load.

Table 5: t-test of learning performance

Test	N	Mean	SD	*t*	*p*
Pre-test	22	5.04	1.27	-10.258	$6.195e - 10 < 0.001$
Post-test		7.50	1.02		

6 DISCUSSION

The research questions raised in this study and their corresponding results are discussed below:

6.1 Tool characteristics - RQ1

Summary of RQ1: (i) Survey with 12 SE researches; (ii) Insights about challenges and difficulties; (iii) Fourteen basic tool features.

To address our first research question (*What are the characteristics an authoring tool should have to help SE professors prepare Flipped Classrooms?*), we conducted a survey with SE researchers experienced in FC. We perceived that the personalized study guide could be essential for leveling students during pre-class studies. Ensuring that students have a relatively similar knowledge base when they enter the collaborative learning environment is vital to creating a more cohesive and focused classroom dynamic, as students can participate in debates and activities without significant disparities in their basic understanding. This leveling process ultimately lays the foundations for a more productive hands-on learning experience in which students can collectively explore, apply, and master the subjects of software requirements engineering more efficiently and effectively.

6.2 Feasibility of creating a personalized study guide - RQ2

Summary of RQ2: (i) Creation of a personalized-based study guide tool; (ii) Demonstration of the feasibility of making a personalized study guide through a lesson on software requirements.

To address our second research question (*Is creating a Personalized Study Guide-based Flipped Classroom Tool to teach Software Engineering feasible?*), we initially utilized the outcomes from RQ1 to guide the construction of a tool that supports the flipped classroom method (Section 3.2). Subsequently, we assessed the feasibility of constructing a flipped classroom using a personalized study guide for teaching software engineering (Section 4.3). We demonstrated that it was feasible to plan, model, and build a lesson on software requirements using the constructed tool. The lesson was developed using a personalized-based study guide so that students could prepare for the classroom (pre-class study).

The tool facilitated the construction of a personalized-based study guide on students' performance, engagement, and individual choices. Personalization can relieve one of the challenges found in the literature (Section 2.1) concerning student heterogeneity during in-class and out-of-class activities, as it promotes equity in student preparation. Adapting the study material to meet each student's specific learning needs can promote deeper understanding and better retention of fundamental software engineering concepts. Additionally, this adaptation can help address knowledge gaps and strengthen students' weaknesses, providing them with feedback for continuous learning and self-awareness from automatic feedback. Discussing the effectiveness of this personalization and its impact on outcomes can underscore the significance of the personalized approach in software engineering education.

Despite the discussed benefits, challenges persist, as described in the literature, for implementing the flipped classroom approach in software engineering education. The use of the tool and the potential for creating personalized-based study guides do not alleviate time constraints for curation or student resistance to changes in teaching methods. However, it may contribute to reducing technical complexity for content adaptation. To achieve this, the availability of technology and adequate training for teachers is necessary.

6.3 Personalized-based study guide impact - RQ3

Summary of RQ3: (i) Average acceptance rate of 76.7%; (ii) Average reduction of 34.6% in test anxiety rate; (iii) Average engagement of 81.5 in the study guide and average performance of 7.3 in the exercises; (iv) Average gain of 2.46 in post-test grades.

The results presented in Section 5 concerning the tool evaluation described in Section 4 help answer our third research question (*What is the impact of using the personalized-based study guide tool in a software engineering lesson on students' perceptions, motivation, and academic achievements?*).

Initially, we analyzed students' perceptions across three dimensions: cognitive presence, learner presence, and technology acceptance. The findings about cognitive presence highlight the importance of providing comprehensive and well-rounded information sources within the study guide to cater to students' needs and encourage them to engage more deeply with the content. In addition, it is essential to consider whether the lack of exploration of varied sources affected the more profound understanding of the concepts discussed in the flipped classroom. The learner presence outcome underscores and highlights the importance of goal setting for learning. Lacking this practice could affect the efficiency of realizing the study plan for some students. The technology acceptance outcomes suggest that the tool can offer tangible benefits and enhance students' study efficiency and effectiveness.

Next, we analyzed student motivation across three dimensions: control of learning beliefs, self-efficacy for learning and performance, and test anxiety. Our analysis investigated whether the tool's use impacted any of the examined motivation dimensions. We observed significant differences in at least one item from each dimension. Regarding control of learning beliefs outcomes, the rate decline in Q25 item indicates the need for additional support for some students and a self-awareness of their shortcomings in efforts during out-of-class preparation. General motivation outcomes suggests that the tool's use positively impacts test anxiety reduction in students, likely because they feel more prepared and have higher self-esteem after out-of-class preparation.

Finally, we analyzed the students' achievements in two dimensions: (1) engagement and performance in the exercises and (2) test performance (pre- and post-test). The tool enabled us to monitor student engagement and performance data during out-of-class study activities, allowing us to align practical activities (in-class) according to student performance. We identified individual student strengths and weaknesses, efficiently aligning the practical class to meet the proposed lesson objectives. In our prior experiences teaching SE using the FC method, we needed the tool to have adequate data on student engagement during pre-class home study before the practical class. Perceptions were based on the delivery of pre-class activities and student observation during the practical class, making it challenging to plan practical class strategies accurately concerning student difficulties. Using the tool, we learned to design adaptive materials, analyze real-time engagement and performance data, and tailor practical activities based on each student's strengths and weaknesses. The data on student grade improvement (pre- and post-tests) demonstrate a significant increase in student learning gains at the end of the flipped class. To understand the relationship between home study, practical class, and test performance, we calculated the Pearson correlation [39]. Figure 7 depicts a strong correlation between student performance on study guide exercises and their engagement (0.73). We also noted a weak correlation between study guide engagement and post-test performance (0.48). We expected this weak correlation because two practical classes occurred before the post-test, which weakens the cause-and-effect relationship between home study and post-test performance.

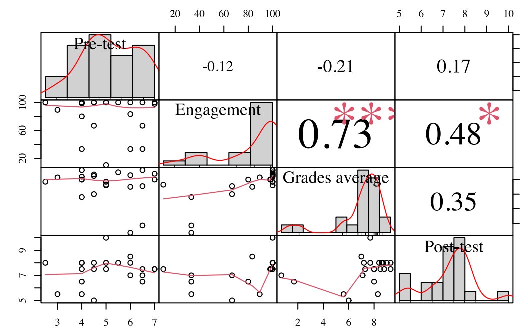

Figure 7: Pearson's correlation coefficient

The research findings indicate that the personalized learning in our approach can improve how we teach software engineering. The approach facilitates self-paced learning, allowing students to progress at their own pace, which enhances learning outcomes.

7 THREATS TO VALIDITY

The discussion of threats to validity encompasses the categories of construct validity, internal validity, and reliability, as described by [37]. In terms of construct validity in our study, the issue is associated with the subjectivity involved in the analysis of the scalar data gathered through the questionnaires, which might not accurately reflect the intended constructs. Concerning internal validity and the potential for researcher interpretation bias, it is noteworthy that a single researcher was responsible for teaching the content in the classroom, which could have influenced the learning outcomes. Lastly, for ensuring reliability, we developed a replication kit available in the repository[5], containing the dataset and scripts for data analysis, as well as scripts for generating results presented in figures and tables.

8 FINAL REMARKS

While this study aims for rigor, it faces certain limitations that restrict its generalization. Firstly, the sample size of 22 participants was obtained from a non-random selection within a single group. Additionally, the generalization of the study's findings to other domains of software engineering education remains unverified. Although the outcomes demonstrate positive impacts of flipped classroom teaching using personalized study guides on students' perception, motivation, and achievement, the applicability of these findings to other educational contexts remains a topic for future exploration.

ACKNOWLEDGMENT

This research was partially funded by CNPQ, under grant number 314425/2021-7.

[5]https://nupreds.ifce.edu.br/necio/educomp24/educomp24-materials.pdf

REFERENCES

[1] Gökçe Akçayır and Murat Akçayır. 2018. The flipped classroom: A review of its advantages and challenges. *Computers & Education* 126, 334–345.

[2] Deniz Akdur. 2022. Analysis of software engineering skills gap in the industry. *ACM Transactions on Computing Education* 23, 1, 1–28.

[3] Pedro Araújo, Cecília Costa, Windson Viana, Nécio de Lima Veras, and Eder JP Farias. 2020. Automatic Personalisation of Study Guides in Flipped Classroom: A Case Study in a Distributed Systems Course. In *2020 IEEE Frontiers in Education Conference (FIE)*. IEEE, 1–9.

[4] Pedro Araujo, Windson Viana, Necio Veras, and José Aires de Castro Filho. 2018. Aprendizagem Adaptativa em Aulas Invertidas de Sistemas Distribuídos: um estudo de caso apoiado no G Suite. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, Vol. 29. 328.

[5] Pedro Araujo, Windson Viana, Necio Veras, Eder Jacques Farias, and José Aires de Castro Filho. 2019. Exploring students perceptions and performance in flipped classroom designed with adaptive learning techniques: A study in distributed systems courses. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, Vol. 30. 219.

[6] Pedro Teixeira de Araújo. 2019. FCTOOL: uma ferramenta para a geração de roteiros adaptativos de aulas invertidas de sistemas distribuídos.

[7] Eduardo Fernandes Barbosa and Dácio Guimarães de Moura. 2013. Metodologias ativas de aprendizagem na educação profissional e tecnológica. *Boletim Técnico do Senac* 39, 2, 48–67.

[8] Jonathan Bergmann and Aaron Sams. 2012. *Flip your classroom: Reach every student in every class every day.* International society for technology in education. http://books.google.com/books?id=nBi2pwAACAAJ

[9] Mustafa Cevikbas and Gabriele Kaiser. 2022. Promoting Personalized Learning in Flipped Classrooms: A Systematic Review Study. *Sustainability* 14, 18, 11393.

[10] Hakan Erdogmus, Soniya Gadgil, and Cecile Peraire. 2019. Introducing low-stakes just-in-time assessments to a flipped software engineering course. In *Proceedings of the 52nd Hawaii International Conference on System Sciences*.

[11] Hakan Erdogmus and Cécile Péraire. 2017. Flipping a graduate-level software engineering foundations course. In *2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering Education and Training Track (ICSE-SEET)*. IEEE, 23–32.

[12] Mary Forehand. 2010. Bloom's taxonomy. *Emerging perspectives on learning, teaching, and technology* 41, 4, 47–56.

[13] Gerald Gannod, Janet Burge, and Michael Helmick. 2008. Using the inverted classroom to teach software engineering. In *2008 ACM/IEEE 30th International Conference on Software Engineering*. IEEE, 777–786.

[14] Olena Glazunova, Tetiana Voloshyna, Valentyna Korolchuk, and Oleksandra Parhomenko. 2020. Cloud-oriented environment for flipped learning of the future IT specialists. *E3SWC* 166, 10014.

[15] Lucas Gren. 2020. A Flipped Classroom Approach to Teaching Empirical Software Engineering. *IEEE Transactions on Education*.

[16] M Haider, A Sinha, and B Chaudhary. 2010. An Investigation of relationship between learning styles and performance of learners. *International Journal of Engineering Science and Technology* 2, 7, 2813–2819.

[17] Salah Hammami and Hassen Mathkour. 2015. Adaptive e-learning system based on agents and object petri nets (AELS-A/OPN). *Computer applications in engineering education* 23, 2, 170–190.

[18] Lu-Ho Hsia, Yen-Nan Lin, and Gwo-Jen Hwang. 2021. A creative problem solving-based flipped learning strategy for promoting students' performing creativity, skills and tendencies of creative thinking and collaboration. *British Journal of Educational Technology* 52, 4, 1771–1787.

[19] Gwo-Jen Hwang, Shao-Chen Chang, Yanjie Song, and Min-Chuan Hsieh. 2020. Powering up flipped learning: An online learning environment with a concept map-guided problem-posing strategy. *Journal of Computer Assisted Learning*.

[20] Gwo-Jen Hwang, Chih-Hung Chen, and Wen-Hui Chen. 2022. A concept mapping-based prediction-observation-explanation approach to promoting students' flipped learning achievements and perceptions. *Educational technology research and development*, 1–20.

[21] Philip Kerr. 2016. Adaptive learning. *Elt Journal* 70, 1, 88–93.

[22] Pang Nai Kiat and Yap Tat Kwong. 2014. The flipped classroom experience. In *2014 IEEE 27th Conference on Software Engineering Education and Training (CSEE&T)*. IEEE, 39–43.

[23] Min Kyu Kim, So Mi Kim, Otto Khera, and Joan Getman. 2014. The experience of three flipped classrooms in an urban university: An exploration of design principles. *The Internet and Higher Education* 22, 37–50.

[24] Rensis Likert. 1932. A technique for the measurement of attitudes. *Archives of psychology*.

[25] José Lima, Fernanda Alencar, and Wylliams Santos. 2021. A Preliminary Guide for Assertive Selection of Active Methodologies in Software Engineering Education. In *Brazilian Symposium on Software Engineering*. 170–179.

[26] Hui-Chen Lin, Gwo-Jen Hwang, and Yaw-Don Hsu. 2019. Effects of ASQ-based flipped learning on nurse practitioner learners' nursing skills, learning achievement and learning perceptions. *Computers & Education* 139, 207–221.

[27] Omar M Mahasneh. 2020. The effectiveness of flipped learning strategy in the development of scientific research skills in procedural research course among higher education diploma students. *Research in Learning Technology* 28.

[28] Mary Lou Maher, Celine Latulipe, Heather Lipford, and Audrey Rorrer. 2015. Flipped classroom strategies for CS education. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*. ACM, 218–223.

[29] Gregory S Mason, Teodora Rutar Shuman, and Kathleen E Cook. 2013. Comparing the effectiveness of an inverted classroom to a traditional classroom in an upper-division engineering course. *IEEE transactions on education* 56, 4, 430–435.

[30] Bruce R Maxim, Thomas Limbaugh, and Jeffrey J Yackley. 2021. Student Engagement in an Online Software Engineering Course. In *2021 IEEE Frontiers in Education Conference (FIE)*. IEEE, 1–9.

[31] Jeffrey J McConnell. 1996. Active learning and its use in computer science. *ACM SIGCSE Bulletin* 28, SI, 52–54.

[32] Sofia Ouhbi and Nuno Pombo. 2020. Software Engineering Education: Challenges and Perspectives. In *2020 IEEE Global Engineering Education Conference (EDUCON)*. IEEE, 202–209.

[33] Hacer Özyurt and Özcan Özyurt. 2018. Analyzing the effects of adapted flipped classroom approach on computer programming success, attitude toward programming, and programming self-efficacy. *Computer Applications in Engineering Education* 26, 6, 2036–2046.

[34] Nicolás Martín Paez. 2017. A flipped classroom experience teaching software engineering. In *2017 IEEE/ACM 1st International Workshop on Software Engineering Curricula for Millennials (SECM)*. IEEE, 16–20.

[35] Paul R Pintrich et al. 1991. A manual for the use of the Motivated Strategies for Learning Questionnaire (MSLQ).

[36] Amy Roehl, Shweta Linga Reddy, and Gayla Jett Shannon. 2013. The flipped classroom: An opportunity to engage millennial students through active learning strategies. *Journal of Family & Consumer Sciences* 105, 2, 44–49.

[37] Per Runeson, Martin Host, Austen Rainer, and Bjorn Regnell. 2012. *Case study research in software engineering: Guidelines and examples*. John Wiley & Sons.

[38] Michael D Ryan and Scott A Reid. 2016. Impact of the flipped classroom on student performance and retention: A parallel controlled study in general chemistry. *Journal of Chemical Education* 93, 1, 13–23.

[39] Patrick Schober, Christa Boer, and Lothar A Schwarte. 2018. Correlation coefficients: appropriate use and interpretation. *Anesthesia & analgesia* 126, 5, 1763–1768.

[40] David Schultz, Stacy Duffield, Seth C Rasmussen, and Justin Wageman. 2014. Effects of the flipped classroom model on student performance for advanced placement high school chemistry students. *Journal of chemical education* 91, 9, 1334–1339.

[41] Egon Teiniker and Gerhard Seuchter. 2020. Improving the Flipped Classroom Model by the Use of Inductive Learning. In *2020 IEEE Global Engineering Education Conference (EDUCON)*. IEEE, 512–520.

[42] JR Van Seters, MA Ossevoort, J Tramper, and Martin J Goedhart. 2012. The influence of student characteristics on the use of adaptive e-learning material. *Computers & Education* 58, 3, 942–952.

[43] Simona Vasilache. 2021. Suddenly Online: Active Learning Implementation Strategies During Remote Teaching of a Software Engineering Course. In *International Conference on Interactive Collaborative Learning*. Springer, 395–402.

[44] Nécio L Veras, Lincoln S Rocha, and Windson Viana. 2020. Flipped Classroom in Software Engineering: A Systematic Mapping Study. In *Proceedings of the 34th Brazilian Symposium on Software Engineering*. 720–729.

[45] Mark von Rosing, Stephen White, Fred Cummins, and Henk de Man. 2015. Business Process Model and Notation-BPMN.

[46] Claes Wohlin, Per Runeson, Martin Höst, Magnus C Ohlsson, Björn Regnell, and Anders Wesslén. 2012. *Experimentation in software engineering*. Springer Science & Business Media.