FEDERAL UNIVERSITY OF CEARÁ
CENTER OF SCIENCE
DEPARTMENT OF COMPUTER
POST-GRADUATION PROGRAM IN COMPUTER SCIENCE
DOCTORAL DEGREE IN COMPUTER SCIENCE

NÉCIO DE LIMA VERAS

MODELING, DEVELOPMENT, ASSESSMENT, AND EMPIRICAL EVALUATION OF PERSONALIZED STUDY GUIDES IN SOFTWARE ENGINEERING FLIPPED CLASSES

FORTALEZA
2024

NÉCIO DE LIMA VERAS

MODELING, DEVELOPMENT, ASSESSMENT, AND EMPIRICAL EVALUATION OF PERSONALIZED STUDY GUIDES IN SOFTWARE ENGINEERING FLIPPED CLASSES

Thesis submitted to the Post-Graduation Program in Computer Science of the Center of Science of the Federal University of Ceará, as a partial requirement for obtaining the title of Doctor in Computer Science. Concentration Area: Software Engineering

Advisor: Prof. Dr. Windson Viana de Carvalho

Co-advisor: Prof. Dr. Lincoln Souza Rocha

FORTALEZA

2024

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

V584m Veras, Nécio de Lima.
Modeling, development, assessment, and empirical evaluation of personalized study guides in software engineering flipped classes / Nécio de Lima Veras. – 2024.
182 f. : il. color.

Tese (doutorado) – Universidade Federal do Ceará, Centro de Ciências, Programa de Pós-Graduação em Ciência da Computação , Fortaleza, 2024.
Orientação: Prof. Dr. Windson Viana de Carvalho.
Coorientação: Prof. Dr. Lincoln Souza Rocha.

1. Ensino de engenharia de software. 2. Sala de aula invertida. 3. Aprendizagem personalizada. I. Título.
CDD 005

NÉCIO DE LIMA VERAS

MODELING, DEVELOPMENT, ASSESSMENT, AND EMPIRICAL EVALUATION OF PERSONALIZED STUDY GUIDES IN SOFTWARE ENGINEERING FLIPPED CLASSES

Thesis submitted to the Post-Graduation Program in Computer Science of the Center of Science of the Federal University of Ceará, as a partial requirement for obtaining the title of Doctor in Computer Science. Concentration Area: Software Engineering

Approved on: 23th February 2024

EXAMINATION BOARD

Prof. Dr. Windson Viana de Carvalho (Advisor)
Federal University of Ceará (UFC)

Prof. Dr. Lincoln Souza Rocha (Co-advisor)
Federal University of Ceará (UFC)

Prof. Dr. Flávio Rubens de Carvalho Sousa
Federal University of Ceará (UFC)

Prof. Dr. Marco Túlio Valente
Federal University of Minas Gerais (UFMG)

Prof. Dra. Fabiane Barreto Vavassori Benitti
Federal University of Santa Catarina (UFSC)

Para toda minha família com muito amor e dedicação.

For my entire family, with all my love and dedication.

ACKNOWLEDGEMENTS

I ask for permission from the readers to express all my gratitude in Portuguese. Inicialmente, gostaria de agradecer a Deus por permitir que guias espirituais iluminassem toda a minha jornada até o presente momento. Tenho convicção de que todas as pessoas que cruzaram meu caminho no processo de doutoramento foram anjos, cuidadosamente escolhidos e que abençoaram de alguma forma as minhas decisões. Neste sentido, só tenho que agradecer amplamente aos anjos que sempre me apoiaram durante minhas idas à Fortaleza-CE.

De maneira especial, agradeço por toda a compreensão e apoio à minha esposa, Elaine Cristina, e ao meu filho amado, Mateus. Vocês sempre foram minha fortaleza para continuar firme e em frente. Agradeço amplamente à minha mãe, Maria das Graças, por todo o apoio e amor incondicional que, com sua sabedoria, sempre me motivou, incentivou e jamais duvidou dessa tão sonhada conquista. Não posso deixar de expressar minha gratidão ao meu pai, Antônio Domingues (*in memoriam*), que durante sua passagem aqui conosco sempre foi um incentivador, vibrando com cada conquista de nossa família. Tenho certeza de que estaria muito feliz e que contaria a todos os seus amigos que seu filho se tornou um doutor em computação. Pai, esse título é seu também. Complemento agradecendo meus irmãos, Cévio e Gésio, pelo companheirismo, parceria, amizade e suporte incondicional em toda e qualquer circunstância necessária. Sem vocês, tudo seria mais difícil e sem graça.

Agradeço aos colegas do IFCE, campus Tianguá pelo suporte com as aulas, especialmente por não ter tido dedicação exclusiva à pesquisa e ao curso de doutorado. Agradeço à Professora Cynthia, que gentilmente cedeu sua turma de Engenharia de Software para que o experimento controlado fosse realizado. Agradeço ao Douglas por todo o apoio técnico e dedicação para que a ferramenta ficasse online e disponível nos servidores do campus. Agradeço imensamente aos alunos e professores que aceitaram participar das avaliações e do experimento com a ferramenta. Vocês foram essencialmente importantes para essa conquista. Sou muito grato por toda seriedade e honestidade que vocês tiveram enquanto participantes da pesquisa. Agradeço também ao programa MDCC da UFC pelo apoio financeiro durante a participação em eventos ao longo do curso. Deixo minha gratidão à secretaria do MDCC, a todos que compõem o GREat, aos colegas/irmãos de orientador, vulgos **Netos do Vento**.

Finalmente, mas não menos importante, agradeço aos professores do MDCC que tive a honra de ser aluno, especialmente aos professores Lincoln Rocha e Windson Viana, meus orientadores. Ao Lincoln, toda minha gratidão pelos ensinamentos, dicas, conversas e orientações.

Ao Windson, não há palavras para expressar toda a gratidão que tenho. Agradeço toda paciência, competência, maestria, humanidade e serenidade com que conduziu minha jornada até aqui. Neste momento, após a conclusão do curso, perco um orientador, mas ganho um amigo para o resto da vida e um parceiro para atuar em muitas pesquisas à frente.

ABSTRACT

Software Engineering (SE) teaching aims to provide a theoretical basis for students to develop a deep understanding of fundamental concepts and principles. However, such a theoretical basis must be combined with hands-on activities to improve practical skills and competencies to solve real-world problems, resulting in software products. Furthermore, both hard and soft skills are mandatory in SE education. While hard skills cover the technical content, soft skills go beyond communicating, cooperating, connecting pieces of knowledge to discover solutions, and thinking critically. In Computer Science Education, Active Learning methods stand out as a student interactive approach, stimulating the building of knowledge instead of passively receiving it from instructors. In this sense, the literature reports many strategies to promote active learning to minimize the risks of students' low learning. Flipped Classroom (FC) is an active learning methodology focused on student engagement, recommending that students come to class after completing significant preparatory work. Unlike traditional approaches and face-to-face classes, FC learning content delivery occurs before class. This pre-study can help students to dedicate more meaningful learning activities collaboratively in class. Students should read or watch videos and analyze the lesson material before class time. Afterward, they will apply these concepts in classroom activities conducted by their teachers. The FC method could mitigate some challenges of SE teaching since it encourages professors to develop distinct learning experiences fitting for any student, respecting their context. New technologies are needed to support transitioning from traditional passive teaching to the FC active method to ensure reliability, interactivity, and collaboration in a student-centered system. Performing real-time, adaptive learning monitoring is crucial for teachers to advise students efficiently. We modeled a personalized study guide-based flipped class to address this. We materialized the model on a platform capable of delivering online instructional materials, where the student's interaction with the previous content determines the nature of the materials delivered later. It is a web-based platform for creating personalized study guides for SE Flipped Classes. The tool adapts the guide content based on the student's performance and engagement. Within the tool, teachers set rules for content adaptation, focusing on class preparation and gaining insights into student progress. The process allowed the professor to create adaptive learning paths in a personalized learning experience. We evaluated the model in three stages. In the first stage, we interviewed software engineering professors with and without experience in flipped classrooms and introduced our tool to gather data on technology acceptance. In the second stage, we assessed the impact of the model in a software engineering class comprising twenty-two students. In the third stage, we conducted a randomized controlled experiment designing personalized flipped classes using the tool. We planned the experiment based on software engineering flipped classes, both with and without personalized learning. The evaluation with professors showed an acceptance rate above 90% according to the technology acceptance model. The second stage encompassed an evaluation of a software requirements lesson with 22 students, showing a significant improvement in scores and receiving positive student feedback. In the third stage, we evaluated our learning model using a randomized controlled experiment with 26 students divided into two groups (experimental and control). The results show positive student perception of the applied model, a significant reduction in anxiety, and increased engagement during the preparation of students in the experimental group. Our personalized study guide-based flipped classes model can support the enhancement of SE education. In this scenario, using adaptive learning technology and the flipped classroom method supported by an authoring tool positively contributed to student motivation, acceptance, engagement, and learning gain in software engineering teaching. Thus, this thesis investigated the effectiveness of using the flipped classroom approach blended with adaptive learning in teaching Software Engineering.

Keywords: Software engineering teaching; Flipped classroom; Personalized learning.

RESUMO

O ensino de Engenharia de Software (ES) tem como objetivo fornecer uma base teórica para que os alunos desenvolvam uma compreensão profunda dos conceitos e princípios fundamentais. No entanto, essa base teórica deve ser combinada com atividades práticas para melhorar as habilidades e competências para resolver problemas do mundo real, resultando em produtos de software. Além disso, tanto habilidades técnicas quanto habilidades interpessoais são importantes na educação em ES. Enquanto as habilidades técnicas abrangem o conteúdo específico de ES, as habilidades interpessoais vão além da capacidade técnica, envolendo a comunicação, cooperação, junção de conhecimentos para descobrir soluções e o pensamento crítico. Na Educação em Ciência da Computação, os métodos de Aprendizagem Ativa destacam-se como uma abordagem interativa do aluno, estimulando a construção do conhecimento em vez de recebê-lo passivamente dos instrutores. Nesse sentido, a literatura relata muitas estratégias para promover a aprendizagem ativa e minimizar os riscos de baixo aprendizado dos alunos. A Sala de Aula Invertida (SAI) é uma metodologia de aprendizagem ativa focada no engajamento do aluno, recomendando que os alunos cheguem à aula após completar um trabalho preparatório significativo. Ao contrário das abordagens tradicionais e das aulas presenciais, a entrega de conteúdo de aprendizagem da SAI ocorre antes da aula. Esse estudo prévio pode ajudar os alunos a engajar-se em atividades de aprendizagem mais significativas e de forma colaborativa em sala de aula. Os alunos devem ler ou assistir a vídeos e analisar o material da lição antes do horário da aula. Em seguida, eles aplicarão esses conceitos em atividades em sala de aula conduzidas por seus professores. O método da SAI poderia mitigar alguns desafios do ensino de ES, uma vez que incentiva os professores a desenvolverem experiências de aprendizagem distintas adequadas para qualquer aluno, respeitando seu contexto. Novas tecnologias são necessárias para apoiar a transição do ensino passivo tradicional para o método ativo, garantindo confiabilidade, interatividade e colaboração em uma abordagem centrado no aluno. Realizar o monitoramento da aprendizagem adaptativa em tempo real é crucial para que os professores aconselhem os alunos de forma eficiente. Para isso, modelamos um roteiro de estudo personalizado baseado em aulas invertidas e materializamos o modelo em uma plataforma capaz de fornecer materiais instrucionais online, onde a interação do aluno com o conteúdo anterior determina a natureza dos materiais entregues posteriormente. Trata-se de uma plataforma baseada na web para apoiar a criação de roteiros de estudo personalizados para aulas invertidas de ES. A ferramenta adapta o conteúdo do roteiro com base no desempenho ou engajamento do aluno. Dentro da ferramenta, os professores estabelecem regras para adaptação de conteúdo, focando na preparação para a aula e obtendo insights sobre o progresso do aluno. O processo permitiu ao professor criar caminhos de aprendizagem adaptativos em uma experiência de aprendizagem personalizada. Avaliamos o modelo em três etapas. Na primeira etapa, entrevistamos professores de engenharia de software com e sem experiência em salas de aula invertidas e introduzimos nossa ferramenta para coletar dados sobre a aceitação de tecnologia. Na segunda etapa, avaliamos o impacto do modelo em uma aula de engenharia de software composta por vinte e dois alunos. Na terceira etapa, realizamos um experimento controlado randomizado projetando aulas invertidas personalizadas usando a ferramenta. Planejamos o experimento com base em aulas invertidas de engenharia de software, tanto com quanto sem o aprendizado personalizado. A avaliação com os professores mostrou uma taxa de aceitação acima de 90% de acordo com o modelo de aceitação de tecnologia. A segunda etapa envolveu uma avaliação de uma aula sobre requisitos de software com 22 alunos, mostrando uma melhoria significativa nas notas e recebendo feedback positivo dos alunos. Na terceira etapa, avaliamos nosso modelo de aprendizagem usando um experimento controlado randomizado com 26 alunos divididos em dois grupos (experimental e controle). Os resultados mostram uma percepção positiva dos alunos sobre o modelo aplicado, uma redução significativa na ansiedade e o aumento do engajamento durante a preparação dos alunos no grupo experimental. Nosso modelo de aulas invertidas baseado em roteiros de estudo personalizados pode apoiar o aprimoramento da educação em ES. Nesse cenário, o uso de tecnologia de aprendizagem adaptativa e o método de sala de aula invertida, apoiados por uma ferramenta de autoria, contribuíram positivamente para a motivação, aceitação, engajamento e ganho de aprendizado dos alunos no ensino de engenharia de software. Assim, esta tese investigou a eficácia do uso da abordagem de sala de aula invertida combinada com aprendizagem adaptativa no ensino de Engenharia de Software.

Palavras-chave: Ensino de engenharia de software; Sala de aula invertida; aprendizagem personalizada.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

AL	Active Learning
BPMN	Business Model and Notation
CDP	Conditional Decision Point
CP	Conjuntion Point
CS	Computer Science
CSEE	Conference on Software Engineering Education
CSEE&T	Conference on Software Engineering Education and Training
DP	Disjunction Point
EC	Exclusion criteria
FC	Flipped Classroom
FCTool	Flipped Classroom Tool
GBL	Game-based Design
GCR	Google Classroom
IC	Inclusion criteria
ICSE	International Conference on Software Engineering
IT	Information Technology
ITU	Intention to use
JiTT	Just-in-Time Teaching
LMS	Learning Management System
MVP	Minimum Viable Product
OP	Optional Point
PBL	Project-based Learning
PEOU	Perceived ease of use
PICO	Population, Intervention, Comparasion, and Outcomes
PSG-model	Personalized Study Guide Model
PU	Perceived usefulness
SBES	Brazilian Symposium on Software Engineering
SDP	Student Decision Point
SE	Software Engineering
SEET	Software Engineering Education and Training, track of International Conference on Software Engineering

SG-model	Non-Personalized Study Guide Model
TAM	Technology Acceptance Model
TE-ALEs	Technology-Enabled Active Learning Environments
UBD	Understanding by Design
VR	Virtual Reality

CONTENTS

1 INTRODUCTION

This thesis investigates the effectiveness of combining FC with adaptive learning techniques for teaching SE. We also propose a model for personalizing FC study guides, and we materialize it on a web-based authoring tool.

This chapter introduces the main aspects of this PhD in Section 1.1, followed by the research problem in Section 1.2. Next, section 1.3 introduces the research hypothesis and the purpose of this thesis. Next, Section 1.4 presents the research questions that guide this entire work, and Section 1.5 presents the research methodology that conducts this PhD thesis. Finally, Section 1.6 offers a roadmap of the thesis.

1.1 Contextualization

Software Engineering teaching aims to provide a theoretical basis for the students to develop a deep understanding of fundamental concepts and principles. However, such a theoretical basis must be combined with hands-on activities that improve practical skills and competencies to solve real-world problems, resulting in software products. The global software market and its dynamic and segmented ecosystem pose several challenges to SE education. For example, how to blend theory and practice in learning activities to teach SE contents (e.g., software architecture, fault-tolerance, and reliability engineering) on a small scale when the cost-benefit is only worth it on a large scale. Furthermore, both hard and soft skills are mandatory in SE education. While hard skills cover the technical content (e.g., requirements engineering, design, software construction, and software testing), soft skills go beyond, including communicating, cooperating, connecting pieces of knowledge to discover solutions, and thinking critically. Teaching (and learning) these skills is challenging, requiring educators' and learners' time and engagement.

A study reports on several pedagogical approaches implemented in SE education that aim to improve students' motivation and engagement (OUHBI; POMBO, 2020) (e.g., problem-based learning, gamification, and flipped classroom). In many domains, including Computer Science Education, Active Learning (AL) methods stand out as an approach to encourage student interaction, improving knowledge building instead of receiving content passively from instructors (BARBOSA; MOURA, 2013). This approach holds out, especially because passive students lose concentration between 10 and 15 minutes after starting a 50-minute class (MCCONNELL, 1996). In this sense, the literature reports many strategies to promote active learning to minimize the risks of students' low learning (HWANG *et al.*, 2020; MAXIM *et al.*, 2021; VASILACHE, 2021; HSIA *et al.*, 2021; ARAÚJO *et al.*, 2020; LIMA *et al.*, 2021). For instance, the risk of low learning is reduced by selecting short activities, properly structured (ARAÚJO *et al.*, 2020). Involving students in a series of challenges on current topics also allows teachers to conduct the discussion flow, which turns students into active participants (HWANG *et al.*, 2020). In addition, grouping students into small teams and encouraging them to solve a problem improves activity success (VASILACHE, 2021). In short, AL can be understood as changing students' behavior toward more active participation through examples, demonstrations, or practical activities in learning environments designed to stimulate students (RAJ; RENUMOL, 2022; XIE *et al.*, 2019; MUÑOZ *et al.*, 2022).

A Flipped Classroom (or Inverted Classroom) is an active learning methodology focused on student engagement (ROEHL *et al.*, 2013), which recommends that students come to the class after completing significant preparatory work. It means learning content delivery takes place before class, in contrast to traditional approaches (BERGMANN; SAMS, 2012). 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 (BERGMANN; SAMS, 2012). There is no unique way to flip a class, but the model presented in (BERGMANN; SAMS, 2012) frequently represents a standard model for flipped classrooms. Content delivery starts with guidelines prepared by the teachers and may use many formats. It is presented to students before the face-to-face classes and can be designed in a study guide to guide the students throughout the content exploration, helping to make a path for their learning. One is the study guide, which consists of digital curation of learning materials, such as videos, podcasts, content-related texts (i.e., book chapters, white papers, scientific papers), and self-assessment tests (e.g., Google Forms quiz) (ARAUJO *et al.*, 2019). After receiving the study guide, the learner should read and watch the lesson content, going deep according to their previous knowledge. Students can also carry out online research or participate in online discussions (e.g., at home or in the library) to be better prepared for the in-class moment (GANNOD *et al.*, 2008). However, the advantages of the FC approach can be threatened when the instructional content quality is low, or it is a timing-consuming task, leading to a decrease in student engagement (MAHER *et al.*, 2015).

FC allows the instructor to better deal with distinct levels of students, trying to homogenize their knowledge before in-class time (BERGMANN; SAMS, 2012). 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 more the unique way to take profit of 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 (AKÇAYIR; AKÇAYIR, 2018). Teachers flipped their classes in many domains (e.g., mathematics, health, social sciences, humanities) in schools and universities around the world (AKÇAYIR; AKÇAYIR, 2018; ZHAO *et al.*, 2021; LIN *et al.*, 2019; MAHASNEH, 2020; LIN *et al.*, 2021; HAN; RØKENES, 2020). FC moves activities into the classroom, including those that may have traditionally been considered homework. For instance, in Fassbinder *et al.* (2015), the first moment in the classroom is devoted to questions and answers about the study guide content. Besides, the professor gives a brief explanation of the topic. After that, students do practices related to the lesson under study. Students worked collaboratively performing problem-solving, simulations, debates, game-based learning activities, media creation, and programming challenges.

One of the main challenges in FC is that students have different characteristics, paces, prior knowledge, and motivations to study (ARAÚJO, 2019). 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 (KERR, 2016). This content adaptation saves extra effort for the student to organize the sequence of study and makes learning more efficient (HAIDER *et al.*, 2010). Adaptive learning systems consist of multiple components, which together allow instruction to be tailored to the needs of individual learners (SETERS *et al.*, 2012). Adaptation can occur according to the student's prior knowledge, adapting the content to be delivered.

1.2 Motivation

Adaptive learning with the FC method enhances its efficacy in personalizing study guides to suit individual learners. By accounting for the diverse challenges and opportunities presented during the learning paths, this approach considers each student's unique requirements, interests, aptitudes, and areas for improvement. Furthermore, it leverages analytical data reports to deliver personalized content and offers real-time feedback and assessment, fostering an optimal learning experience (CEVIKBAS; KAISER, 2022). 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. Cevikbas and Kaiser (2022) confirmed that emerging technologies (e.g., Virtual Reality (VR), cloud technology, machine learning, big data, learning analytics, mobile learning, and gamification) are promising for improving the coupling between FC and content personalization.

FC method could help mitigate some challenges of SE teaching since it encourages professors to develop distinct learning experiences fitting for any student, respecting their context (FASSBINDER *et al.*, 2015). Furthermore, findings reported in the specialized literature showed that the FC model positively impacted student learning (SCHEFER-WENZL; MILADINOVIC, 2018a; PUGSEE, 2017; SIMPSON; STORER, 2017; PÉRAIRE, 2019; TEINIKER; SEUCHTER, 2020). Although using FC methodology will imply dealing with additional obstacles (STRELAN *et al.*, 2020), such as (1) professors' difficulties during digital curation – they need to be motivated and confident with adequate levels of time, skills, and resources; (2) lack of specific tools for FC since, in general, professors use a set of combined tools (e.g., Youtube, Edpuzzle, Moodle/Google Classroom, Github) to create and make available their study guide; and (3) to chose engaging activities to produce meaningful learning experiences in-class time.

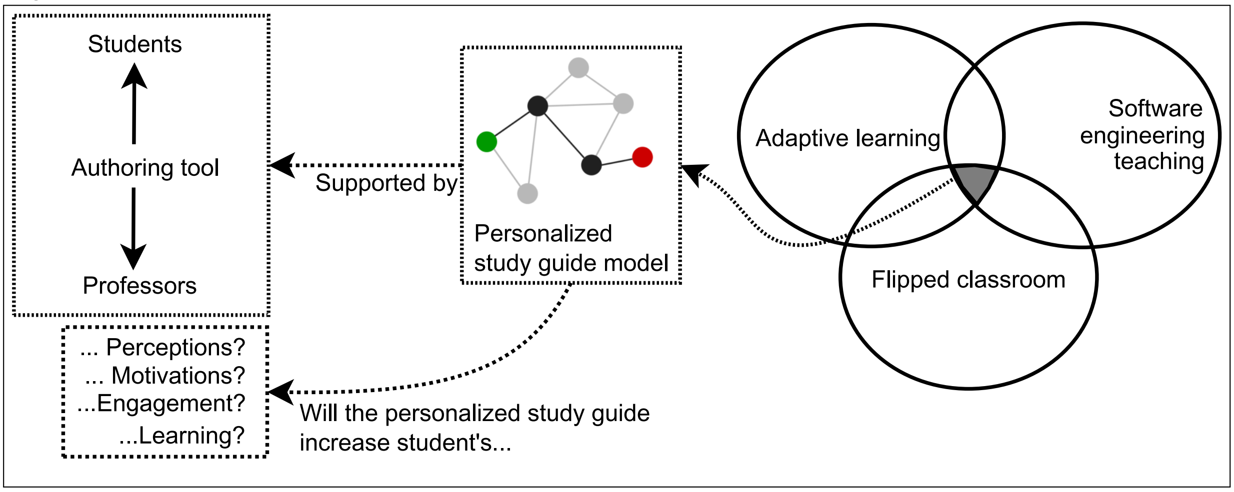
New technologies are needed to support the transition from traditional teaching to the FC active method to ensure reliability, interactivity, and collaboration in a student-centered system. It is crucial to perform real-time, adaptive learning monitoring to enable teachers to advise their students efficiently (HAMMAMI; MATHKOUR, 2015). 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 delivered later. It is purpose is to generate a personalized learning experience.

In SE flipped classes, using a personalized study guide model, supported by an authoring tool (adaptive learning technology), may positively impact student motivation, acceptance, engagement, and learning gain. For this reason, this thesis addresses the following research problem, also illustrated in Figure 1:

Research Problem

How do the Personalized Study Guides in Software Engineering Flipped Classes impact students' perception, motivation, engagement, and learning gain during software engineering teaching?

Figure 1 – Overview of the Research Problem



Source: Author

1.3 Research Hypothesis and Goal

In this way, the actual thesis investigates the effectiveness of using personalized study guides in software engineering flipped classes. Therefore, the following hypothesis is established and will be investigated in this thesis:

Research Hypothesis

Software Engineering Flipped Classes using Personalized Study Guides positively impact students' perception, motivation, engagement, and learning.

During the literature problem investigation, we found no FC approach with personalized study guides for SE teaching. Helping SE teachers to apply a personalized FC model, the actual thesis has the following objective:

Research Goal

Investigate the impacts on students' perception, motivation, engagement, and learning using Personalized Study Guides in Software Engineering flipped classes.

1.4 Research Questions

This thesis aims to answer the following research questions:

Research Questions

- **RQ1 - What is the state of the art of using the Flipped Classroom method in Software Engineering teaching?**
  An in-depth investigation regarding the use of FC in teaching SE is needed to understand the challenges, benefits, and opportunities.
- **RQ2 - What are the characteristics that an authoring tool should have to help Software Engineering professors during the preparation of flipped classes?**
  We surveyed Software Engineering professors to gain a deeper understanding of the flipped process and determine the initial insights, desired features, and design principles for a Flipped Classroom support tool.
- **RQ3 - Is feasible to design, formalize, and implement a Personalized Study Guide Model to represent out-of-class material for SE flipped classes?**
  This question is related to formalizing the Personalized Study Guide Model and making it feasible by creating instances of real case lessons as a proof-of-concept.
- **RQ4 - What are the SE professors' perceived usefulness and ease of use of the Flipped Classroom Tool?**
  This question concerns how to know the levels of acceptance and use of the Flipped Classroom Tool for personalized study guides by Software Engineering professors according to the Technology Acceptance Model.
- **RQ5 - What is the impact of Personalized Study Guides on students' perception, motivation, engagement, and learning gain during software engineering flipped classes?**
  This question is concerned with the evaluation of the thesis. This model, blended with

> the content personalization technique, is expected to positively impact SE students' motivation, acceptance, engagement, and learning.

1.5 Research Methodology

The research methodology followed in this PhD thesis was inspired by Carvalho (2019), who organized her research stages based on Ghazi and Glinz (2018) and structured them following the proposal of Wieringa *et al.* (2006).

Our research is organized into three phases: Problem, Solution, and Evaluation (shown by Figure 2). The Problem phase encompasses a systematic in-depth study to define the hypothesis and the thesis objective. It also deepens 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 model formalization and the product (tool) development in this PhD. A BPMN-based model inspired our formalization, and this allowed create real case a lesson instance as proof of concept for our Personalized Study Guide Model. Furthermore, a Minimum Viable Product (MVP) of the web-based tool materialized the model, and Software Engineering teachers and students evaluated their acceptance. Finally, the Evaluation phase will evaluate the proposed approach during Software Engineering classes using the model.

The Problem phase is classified as Exploratory and Descriptive research. Explanatory is a kind of research that aims to deepen the subject that is the target of study and allows the problem refinement and goals definition (CARVALHO, 2019). Descriptive research can be conducted to find assertions about some populations and determine specific characteristics. The instruments for data collection for exploratory and descriptive research are commonly semi-structured, mixing open and closed questions (WOHLIN *et al.*, 2012). This Ph.D. investigates how researchers used the Flipped Classroom in Software Engineering teaching, identifying recurrent practices, trends, opportunities, and gaps. To answer the first research question (RQ1), we performed a secondary study using two research methods: systematic mapping and snowballing. Systematic literature review studies use a basic methodology guiding research questions, search strategy, criteria for inclusion or exclusion of primary papers, and data extraction (KITCHENHAM, 2004).

To answer the second research question (RQ2), we carried out a descriptive survey, collecting insights and data about specific features from SE researchers (authors identified by

Figure 2 – Ph.D. Research Methodology

Research Phase	Problem		Solution		Evaluation	
Chapter Title	Literature Review	Deepening the Flipped Process in Software Engineering Teaching	A Personalized Study Guide Model	Flipped Classroom Tool	Evaluation	
Research Question	RQ1 What is the state of the art of using the Flipped Classroom method in Software Engineering teaching?	RQ2 What are the characteristics that an authoring tool should have to help SE professors during the preparation of Flipped Classrooms?	RQ3 Is feasible to design, formalize, and implement a Personalized Study Guide Model to represent out-of-class material for SE flipped classes?		RQ4 What are the SE professors' perceived usefulness and ease of use of the Flipped Classroom Tool?	RQ5 What is the impact of Personalized Study Guides on students' perception, motivation, engagement, and learning gain during software engineering flipped classes?
Research Type	Exploratory	Descriptive	Mixed-method			Empirical
Research Method	Systematic mapping and snowballing	Survey	Formalization	Demonstration	Interview	Controlled experiment
Content Elements	• Systematic mapping study • Snowballing	• Survey with SE researchers from the literature review • Initial insights and tool design principles	• Model description and formalization • Lesson simulation as proof-of-concept	• FCTool features and Minimal Viable Product deployment • Tool applicability demonstration based on the model	• Interview with SE professors	• Practical experiences on personalized FC using the tool with SE students

Source: Author

literature review) with knowledge and experience in Flipped Classrooms. The main outcome was to identify design principles to support the development of tools for the flipped classroom method.

The Solution phase is classified as mixed-method research and is associated with question 3 (RQ3). The mixed-method research type employs data collection and analysis techniques related to quantitative and qualitative data (EASTERBROOK *et al.*, 2008). To answer RQ3, we describe and formalize our Personalized Study Guide Model. Then, to demonstrate the model's feasibility, we created a real case lesson instance as proof of concept and performed a computer-based simulation. Also, we developed a Minimum Viable Product (tool) from the design principles identified during the Problem phase (RQ2) to demonstrate the tool's applicability based on the model.

Finally, the Evaluation phase seeks to answer research questions 4 (RQ4) and 5 (RQ5). First, we conduct interviews with SE professors and students to understand the SE professor's perceived usefulness and ease of use of the tool according to the Technology Acceptance Model. After the tool development was completed (Solution phase result) and the SE professor's perceived were known, we performed an empirical evaluation through a controlled experiment with SE students. The research included assessing the tool and investigating the impacts on students' perception, motivation, engagement, and learning gain from the Personalized Study Guide Model. Students participated in flipped classes using the tool (with an experimental group). The results helped to compare the impacts of the model in Software Engineering teaching and the use of the tool—also benefits and difficulties from students' points of view.

1.6 Document Structure

This thesis is organized into six more chapters. Figure 3 shows an overview of them, a summary, and the research question related to each chapter.

Figure 3 – Ph.D. Roadmap

Chapter 2	Literature Review	• Systematic mapping study • Snowballing	RQ1 - What is the state of the art of using the Flipped Classroom method in Software Engineering teaching?
Chapter 3	Deepening the Flipped Process in Software Engineering Teaching	• Survey with SE researchers from the literature review • Initial insights and tool design principles	RQ2 - What are the characteristics that an authoring tool should have to help SE professors during the preparation of Flipped Classrooms?
Chapter 4	A Personalized Study Guide Model	• Model description and formalization as a graph-based problem • Instances of real case lesson as proof of concept	RQ3 - Is feasible to design, formalize, and implement a Personalized Study Guide Model to represent out-of-class material for SE flipped classes?
Chapter 5	Flipped Classroom Tool	• FCTool features and Minimal Viable Product deployment • Interview with SE professors	
Chapter 6	Evaluation	• Practical experiences on personalized FC using the tool with SE students	RQ4 - What are the SE professors' perceived usefulness and ease of use of the Flipped Classroom Tool? RQ5 - What is the impact of Personalized Study Guides on students' perception, motivation, engagement, and learning gain during software engineering flipped classes?
Chapter 7	Conclusions	• Outcomes • Limitations of study • Threats to validity • Future works	

Source: Author

Chapter 2 provides the theoretical background regards areas that compose this PhD. The chapter reports the advantages, challenges, tools, methods, impacts, evaluations, and active learning methodologies concerning the FC method used in SE teaching.

Chapter 3 starts by presenting the survey result with SE researchers identified by the literature review reported in Chapter 2. Then, initial insights and tool design principles are detailed. Finally, a questionnaire was used to deepen the flipped process from SE teachers with knowledge and experience in Flipped Classrooms.

Chapter 4 formally describes the Personalized Study Guide Model. Instances of real case lessons were created as proof of concept of the model.

Chapter 5 implements our model described in Chapter 4 into a web-based tool in a Minimum Viable Product. We demonstrated the tool applicability through examples based on the proposed model.

Chapter 6 explains the evaluation methodology regarding the impacts on students' perception, motivation, engagement, and learning gain from the Personalized Study Guide Model. The effects on perception and motivation were evaluated with SE professors and students through interviews. Next, practical experiences on personalized FC using the tool with SE students through a controlled experiment were performed.

Chapter 7 presents the conclusions of this thesis showing the results achieved, limitations, threats to validity, and future work.

2 LITERATURE REVIEW

This chapter aims to systematically examine the literature using the FC method in SE teaching to answer our first research question (RQ1): "*What is the state of the art of using the Flipped Classroom method in Software Engineering teaching?*". The chapter presents a systematic literature review conducted during the first semester 2020. The study started with 769 studies and finished with 26 primaries, satisfying the selection criteria. The search for primary studies considered the first half of 2020 and before. As a result, the review found papers from 2008 to 2020 published in SE conferences or journals. Then, the chapter shows an update of the mapping study through an extension (using a Snowballing technique) produced in 2021. As a result, the extension could identify more SE researchers with knowledge and experience from outcomes, adding 12 papers to the systematic literature review (after starting with 1378 papers - 602 backward snowballing and 776 forward snowballing).

The chapter is structured as follows: Section 2.1 presents the background and Section 2.2 other systematic reviews. Section 2.3 shows the systematic study protocol and the review process used. Section 2.4 describes the main results of the secondary study. Section 2.5, discusses each research question proposed in the review regarding the findings, the threats to validity, and some opportunities for future research. Finally, Section 2.6 contains the chapter summary.

2.1 Background

Flipped Classroom advocates that students must first contact the lesson content at home. They should read or watch videos and analyze the lesson material before class time. Afterward, they will apply these concepts in classroom activities conducted by their teachers (BERGMANN; SAMS, 2012). Thus, content presentation occurs individually at home (contrasting with traditional approaches), and practical exercises (e.g., simulations and programming micro-projects) are held collaboratively during class time. This type of class changes the role of professors from the leading player in learning to the mediator of students' learning (KING, 1993). FC moves activities into the classroom, including those that may have traditionally been considered homework. For instance, in Fassbinder *et al.* (2015), the first moment in the classroom is devoted to questions and answers about the study guide content. Besides, the professor gives a brief explanation of the topic. After that, students do practices related to the lesson under study. Problem-solving, simulations, debates, game-based learning activities, media creation, and programming challenges collaboratively among students. A set of best practices for FC execution is provided in Harris *et al.* (2016) and divided into two perspectives. From the learner's perspective, Harris *et al.* (2016) summarize three main aspects:

a) **Engagement** (students who do not prepare for class). Suggests for teachers: develop online quizzes corresponding to lectures and other class preparation materials. Instructors should incorporate the quizzes into the students' course grades to encourage engagement.
b) **Preparation** (Students must complete the out-of-class requirements and have a basic understanding of the content). Suggests for teachers: each class must begin with a preparation summary. Teachers should engage students in a guided discussion or require students to create their questions related to the educational material and share these questions with the group.
c) **Resistance** (Students are adjusting to a new teaching model and learning "how to learn" in the FC). Suggests for teachers: Teachers in the Flipped Classroom should help students to change their habits. To help with student buy-in, Teachers must be prepared to answer student questions and explain how the FC method works. In addition, it is necessary to communicate expectations and responsibilities and how the class time will be used for engaging activities.

From the teacher's perspective, in contrast, the authors demonstrate six issues and their best practices to minimize them:

a) **Select and master the Flipped Classroom paradigm before trying it on students**. Best practice: Teachers need to master FC techniques such as Project-based Learning (PBL), Understanding by Design (UBD), and Game-based Design (GBL) and explain to students what they will learn from these activities.
b) **Assess student learning**. Teachers must plan courses with learning outcomes and objectives based on the end in mind. They will build goals and results on each assignment and activity at the beginning of the course.
c) **Be willing to give up some classroom control**. It is a paradigm shift. Students must "unlearn" behaviors and habits they have spent years developing. Teachers must relinquish some "classroom power" to engage students in knowledge sharing and be responsible for their learning.
d) **Start small**. An FC model starts into units (a single content/lesson). The classes

must be planned thoughtfully, incorporating teacher and student expectations, course objectives, outcomes, and practical activities.

e) **Plan activities well**. Students need to connect content to learning objectives. They must practice active learning approaches, such as PBL projects, game-based laboratory sessions, hackathons, or hands-on activities.

f) **Engage students in planning the flip**. Recruit savvy students to curate online videos to support the course material. The students should understand what they are looking for in content and course objectives.

2.2 Other Systematic Reviews

Giannakos *et al.* (2014) presented a review of the FC approach in the area of Computer Science (CS) education. Thirty-two peer-reviewed articles were collected and analyzed in that study. The results show the benefits and challenges of adopting a flipped approach in the classroom. The review covers some subject areas related to computer science, and although it predominates in most works (15), only two papers are directly linked to SE. The authors' perception is that students' learning performance is often perceived to increase, and others remain at the same level (when compared to traditional instructional approaches). In addition, student involvement, group work, and critical thinking benefited from inverting the classroom (especially from practical activities using active learning moments).

Furthermore, in another paper, Giannakos *et al.* (2018) has reviewed the FC approach with 37 peer-reviewed papers. The study aimed to identify the most common technologies, subject domains, evaluation methods, and pedagogical designs applied in FC research. The authors offered a systematic analysis of both benefits and drawbacks, identifying opportunities to improve future research. The most prevalent areas in the reviewed studies were CS and Information Technology (IT), with 16 papers. Among the sixteen studies in the review, only two were about SE (duplicating the number of papers found previously by Giannakos *et al.* (2014). Overall, the reviewed studies tend to six common benefits regarding the use of the FC: (1) increased learning performance; (2) positive attitudes by students; (3) high engagement; (4) students have more discussions (qualitatively measured); (5) cooperative learning; and (6) improves students' learning habits. Besides, the studies show three challenges regarding the use of the FC: (1) high initial cost and very time-consuming for the instructor; (2) students' initial unreceptive behavior with the new approach; and (3) decreased students' attendance. The review shows that the number of papers on FC substantially increased after 2012, and there is a wide demand for the CS/IT domain.

Chanin *et al.* (2018) aimed to describe the state-of-the-art of SE education through the education of software startups, analyzing and identifying the best practices, opportunities, and gaps in this field. Therefore, the authors conducted a systematic mapping study. Although the study did not discuss SE teaching using flipped classes, it found that using active learning methods improves students' knowledge. In addition, the study found that offering a real-world experience to students remains a challenge. However, it is possible to minimize this gap with a connection between the educational setting and the university ecosystem (e.g., technology parks and incubators). Shi *et al.* (2019) found high-quality empirical research on cognitive learning among university students using a meta-analysis to determine the overall effectiveness of Technology-Enabled Active Learning Environments (TE-ALEs). TE-ALEs can be implemented through FC models. Their systematic review found 31 papers from high-quality peer-reviewed journals and only one in SE.

Finally, Marques *et al.* (2014) performed a mapping study on practical approaches to teaching SE. The study reported some instructional approaches to addressing practical experiences. The learning by doing approach is the most used, with 93 studies, followed by the problem-based learning approach with 21 studies, case studies with 15, games with 12, and FC with only one. It is important to highlight that this systematic literature review has not found a specific systematic study that deals with SE teaching using FC.

2.3 Methodology

The goal of this secondary study (mapping study and snowballing) is to determine how FC has been used in SE teaching. The purpose is to investigate the scientific literature and identify the current techniques for content delivery and in-class activities. Besides, it aims to determine how researchers evaluate their experiences of using FC in ES courses. This chapter expects to identify recurrent practices, trends, opportunities, and gaps using flipped classes in ES courses. The systematic study follows the methodologies and stages proposed by Kitchenham (2004), Wright *et al.* (2007) and Wohlin (2014). The main steps of the methodology are: (a) definition of specific research questions and search strategy ; (b) criteria elaboration for papers selection; (c) suitability criteria for included studies; (d) data extraction strategy elaboration; (e) data analysis, and (f) interpretation of results. The study stages were structured in three pivotal phases: (1) planning, (2) execution, and (3) reporting the results.

2.3.1 Specific Research Questions and Search Strategy

Systematic study goals lead to six specific research questions (SRQs) (Table 1). They attempt to find the most used FC models and their relationships with other teaching methodologies. SRQs also seek to map the tools professors use to implement FC, the evaluation methodologies of the studies, and evidence reports on the impact of the research in FC on SE courses.

Table 1 – Specific Research Questions

Research Goal	
What is the state of the art of using the flipped classroom method in Software Engineering Teaching?	
ID	**Specific Research Question**
SRQ1	Which were the active learning methodologies used for designing the in-class activities?
SRQ2	What are the other teaching/learning methods used in collaboration with the flipped classroom method?
SRQ3	Which tools were used to support the flipped classroom method in Software Engineering classes?
SRQ4	How were the studies evaluated?
SRQ5	What were the main impacts on student learning?
SRQ6	What are the main benefits and challenges regarding the use of Flipped Classroom in Software Engineering teaching?

Source: Author (2020).

The study adopted the Population, Intervention, Comparasion, and Outcomes (PICO) approach (KITCHENHAM *et al.*, 2006) to guide the keywords identification and create the search string from the six SRQs (without applying the Comparison). Table 2 shows the final search string performed on the databases of ACM[1], IEEE Xplore[2], Scopus[3], Web of Sciences[4], and Springer[5]. The databases were selected based on the experience reported in Marques and Robledo (2018). The mapping study collected data during the first semester of 2020, and the search for primary studies considered the first half of 2020 and earlier. Subsequently, the mapping updated the data during the first semester of 2021 through an extension, partially utilizing a snowballing technique. The outputs of the mapping study constituted the initial set of papers for snowballing. Two iterations were conducted on the entire set. Initially, one iteration involved backward snowballing using the reference lists to identify new candidate papers. Subsequently,

1 http://www.dl.acm.org
2 https://ieeexplore.ieee.org/Xplore
3 https://www.scopus.com/
4 http:// www.webofknowledge.com
5 https://link.springer.com/

another iteration was carried out for forward snowballing, utilizing Google Scholar citations to identify a new set of candidate papers based on the initial set's citations. We conducted a few iterations because our primary objective was to update the review seeking novelties moving forward (second half of 2020 and beyond), considering that the mapping covered the past.

Table 2 – Search String

Scope	String Keywords
Population: Software engineering teaching	*"software engineering"* ***OR*** *"software modelling"* ***OR*** *"software design"* ***OR*** *"software requirements"* ***OR*** *"software concept"* ***OR*** *"software test"*
AND	
Intervention: Use of the flipped classroom method	*"flipped classroom"* ***OR*** *"inverted classroom"* ***OR*** *"flip classroom"* ***OR*** *"reversed classroom"* ***OR*** *"flip* class*"*
AND	
Outcomes: Learning impacts from primary studies	*"cognitive impact"* ***OR*** *"learning evaluation"* ***OR*** *"learning impact"* ***OR*** *"student motivation"* ***OR*** *"student assessment"* ***OR*** *"gained knowledge"* ***OR*** *"learning assessment"* ***OR*** *"exploratory study"* ***OR*** *"student surveys"* ***OR*** *"student feedback"* ***OR*** *"experimental study"* ***OR*** *"case study"*

Source: Author (2020).

2.3.2 Paper Selection Criteria

The study excluded searched papers based on their titles, abstracts, and full text. Therefore, it is necessary to create exclusion and inclusion criteria to select papers according to goals. These criteria created were applied sequentially in an online form. Studies found in the search that matched exclusion criteria were discarded. On the other hand, studies that matched inclusion criteria were selected for the next phase, the whole reading. The Exclusion criteria (EC) are listed in Table 3.

Table 3 – Exclusion criterias

ID	Exclusion criteria
EC1	The paper describes a secondary study (review or systematic mapping)
EC2	The paper is written in a language other than English
EC3	The paper deals with flipped classes, but in domains different from software engineering
EC4	The paper is related to SE teaching but does not use the FC method
EC5	The paper does not use the FC method
EC6	Can not access the paper, or the paper refers only to an abstract
EC7	The paper does not report any evaluation

Source: Author (2020).

Two Inclusion criteria (IC) were used: (IC1) the paper deals with SE teaching using the FC method, and (IC2) the paper provides empirical evidence regarding the learning impacts of SE teaching.

2.3.3 *Data Extraction Strategy*

Table 4 shows the specific research questions and the desired extracted data related to them. In addition to those data, bibliographic data such as year, author(s), publication venue, and total citations received by the study were also extracted. Furthermore, the Mendeley [6] was used to support the data extraction process. Used it to catalog, organize, and manage notes about papers. Besides, Google Forms was used too to assist the data extraction process.

Table 4 – Data extraction regarding selected papers

Question	Data to be extracted	Description
SRQ1	Active learning methodologies	A brief description of the active learning methodologies used during practical classes
SRQ2	Other teaching/learning methods	A list of other methods used in combination with the flipped classroom method (e.g., PBL)
SRQ3	Content delivery	A summary of how content is delivered to students (e.g., e-mail, LMS, study guides, intelligent systems)
	Authoring tools	Tools used to support content production (e.g., Google Suite, video editors)
	Consumption tools	Tools used to support content consumption during the flipped classroom (e.g., LMS, Youtube)
SRQ4	Evaluation methods	Identify the evaluation methods adopted in the studies (e.g., survey, pre and posttests)
	Attributes assessed	Identify which aspects were assessed in the studies (e.g., students' performance, motivation)
SRQ5	Learning impacts	A list of impacts on student learning
SRQ6	Advantages	A summary of the advantages related to using FC in SE teaching
	Challenges	A summary of the challenges related to using FC in SE teaching

Source: Author (2020).

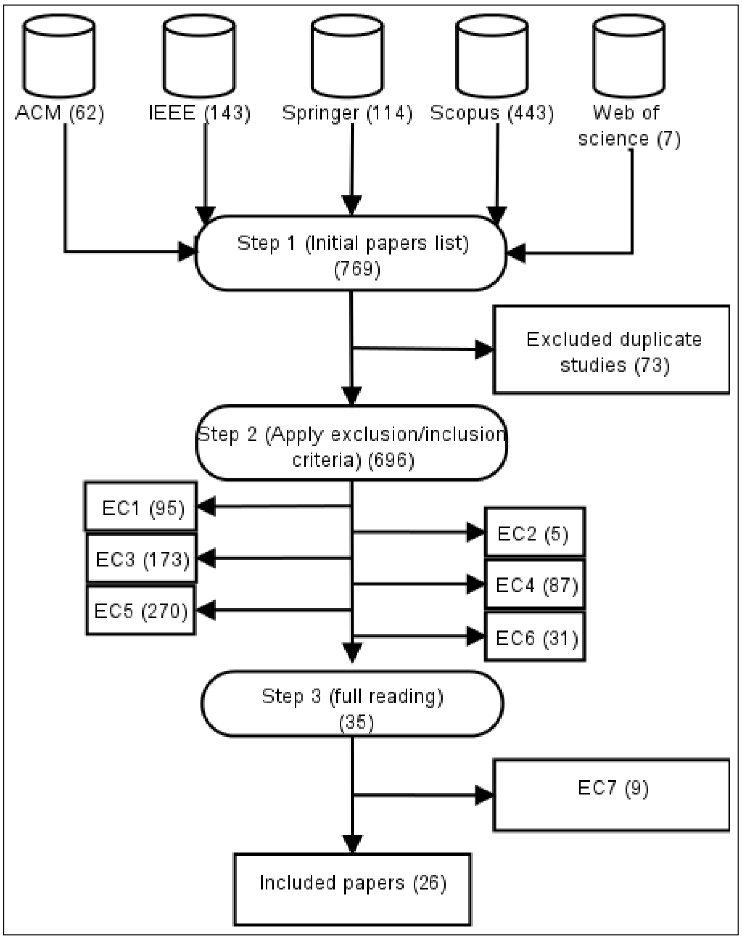
2.4 Results

Figure 4 summarizes the studies selection process during the systematic mapping study. First, the research string was performed on the five databases. Seven hundred sixty-nine papers (step 1) returned from the databases: ACM (62), IEEE-Xplore (143), Springer Link (114), Scopus (443), and Web of Science (7). After that, the duplicated study entries were excluded using Mendeley (73 duplicate studies). Then, the titles and abstracts were read by applying the exclusion criteria. In the case of doubts, the supervisors make a final decision. The result was a set of 35 (step 2) potentially relevant studies. Finally, after full reading and applying EC7 (step 3), the result was 26 papers included in the systematic mapping. Besides, the study looked at the

6 Available at <https://www.mendeley.com/>

papers related to search string published in the Brazilian Symposium on Software Engineering (SBES) in the last ten years (2010-2020). However, the search could not find any study (even in Portuguese). All data extracted can be looked at in a spreadsheet [7].

Figure 4 – Systematic Mapping Process

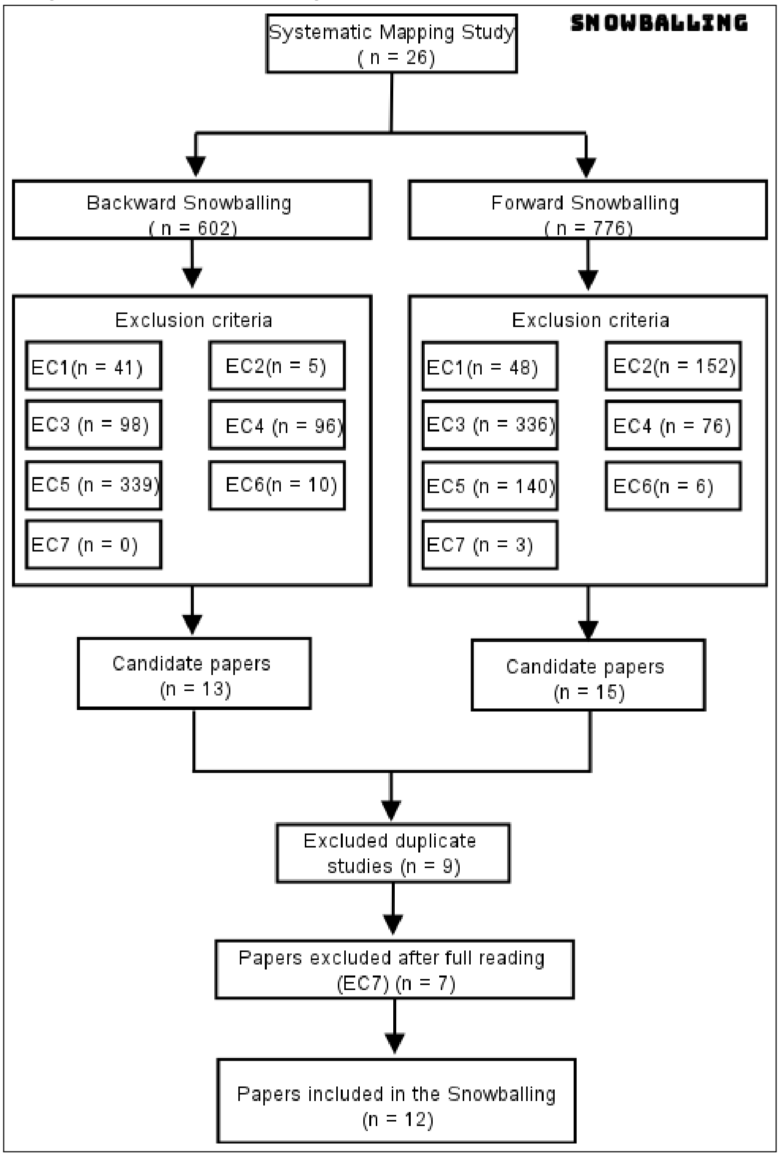


Source: (VERAS *et al.*, 2020)

Then, the literature review was completed with a snowballing procedure following guidelines by Wohlin (2014). First, backward snowballing was performed considering all the references from included studies found in the systematic mapping study (previous step). After forward snowballing, considered all citations to the papers included in the mapping study. As a result, it was discovered and added 12 relevant papers on the snowballing procedure. Accordingly, the entire process ended with 38 papers in the literature review. Figure 5 summarizes the study's selection process during snowballing. Finally, all snowballing data for each paper was organized in a spreadsheet [8].

7 Available at <https://nupreds.ifce.edu.br/necio/mapping-allpapers.pdf>
8 available at <http://nupreds.ifce.edu.br/necio/fc-se-snowballing-data.zip>

Figure 5 – Snowballing Process



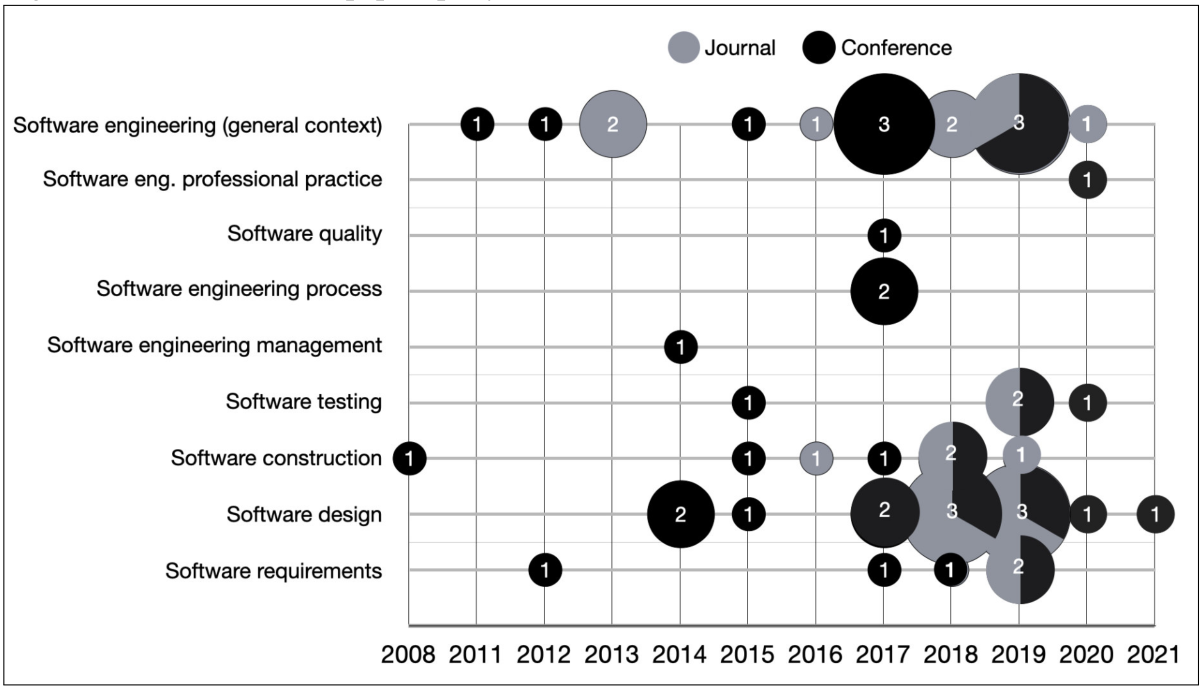
Source: Author (2021)

2.4.1 Included Papers Profile

Before discussing the answer to each specific research question, the section presents a profile of the included papers. The goal is to allow data visualization, including topics, tools, and active methodologies used in the SE education context. Figure 6 shows the paper's distribution by year, classifying them according to the software engineering knowledge areas defined by Bourque *et al.* (2014).

Fifteen studies (39.5%) used FC in the general context of ES courses. The other twenty-three (60.5%) explored specific topics (e.g., software construction, software testing, software design, software engineering management). Besides, Figure 6 shows publishing venues over the years. The literature review found thirteen journal papers (34.2%), for instance,

Figure 6 – Distribution of papers per year



Source: Author (2021)

published in IEEE Transactions on Education and ACM Transaction on Computing Education. Twenty-five (65.8%)of the documents were published in Software Engineering conferences, such as International Conference on Software Engineering (ICSE), Conference on Software Engineering Education (CSEE), Conference on Software Engineering Education and Training (CSEE&T), and Software Engineering Education and Training, track of International Conference on Software Engineering (SEET). Table 5 shows the list of papers, origin, author(s), year, and the number of citations from Google Scholar.

There is no predominance of authors or research groups, but exists small groups with two publications on the subject, such as Cecile Peraire and Hakan Erdogmus - Carnegie Mellon University; Leo Natan Paschoal and Simone R. S. Souza - University of São Paulo, Brazil; Philip Johnson - University of Hawaii, USA; Pakawan Pugsee - Department of Mathematics and Computer Science, Faculty of Science, Chulalongkorn University; and Rajiv Ramnath, Michael J. Herold, Thomas Bihari, and Jayashree Ramanathan - Department of Computer Science and Engineering, The Ohio State University; and with three publications, Schefer-Wenzl and Sigrid Miladinovic - University of Applied Science Campus in Vienna. Besides, Figure 7 shows, organized by year, a cross-citations graph with relationships between the included papers in the mapping study. The paper (GANNOD *et al.*, 2008) is the oldest (2008) and, as expected, the most cited (five papers cite it). Papers (ERDOGMUS; PÉRAIRE, 2017; KIAT; KWONG,

Table 5 – List of papers

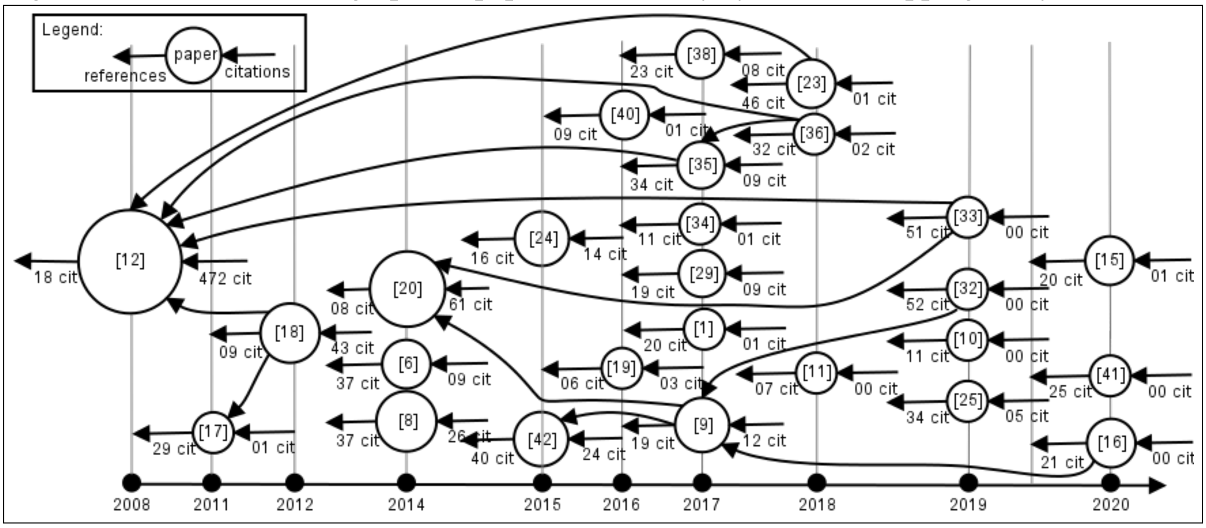
Identifier	Origin	Authors	Year	Citations
[12]	Mapping	GANNOD *et al.*	2008	472
[20]	Mapping	LIN	2019	71
[2]	Snowballing	CHOI	2013	67
[3]	Snowballing	KIAT; KWONG	2014	61
[18]	Mapping	HEROLD *et al.*	2012	43
[8]	Mapping	ELLIOTT	2014	26
[42]	Mapping	TOWEY	2015	24
[4]	Snowballing	PAEZ	2017	23
[5]	Snowballing	SUREKA *et al.*	2013	22
[7]	Snowballing	ÖZYURT; ÖZYURT	2018	16
[24]	Mapping	LEE *et al.*	2015	14
[9]	Mapping	ERDOGMUS; PÉRAIRE	2017	12
[6]	Mapping	CHETLUR *et al.*	2014	9
[29]	Mapping	MATALONGA *et al.*	2017	9
[35]	Mapping	SCHEFER-WENZL; MILADINOVIC	2017	9
[38]	Mapping	SIMPSON; STORER	2017	8
[25]	Mapping	LUBURIĆ *et al.*	2019	5
[13]	Snowballing	ERDOGMUS *et al.*	2019	3
[19]	Mapping	JOHNSON *et al.*	2016	3
[14]	Snowballing	TROYA *et al.*	2021	2
[21]	Snowballing	JOHNSON	2019	2
[36]	Mapping	SCHEFER-WENZL; MILADINOVIC	2018a	2
[22]	Snowballing	SCHEFER-WENZL; MILADINOVIC	2018b	2
[26]	Snowballing	PASCHOAL *et al.*	2020	1
[17]	Mapping	HEROLD *et al.*	2011	1
[40]	Mapping	SU *et al.*	2016	1
[1]	Mapping	ACHARYA *et al.*	2017	1
[34]	Mapping	PUGSEE	2017	1
[23]	Mapping	KOCHUMARANGOLIL; RENUMOL	2018	1
[15]	Mapping	GLAZUNOVA *et al.*	2020	1
[27]	Snowballing	PUGSEE	2018	0
[11]	Mapping	FORTUNA *et al.*	2018	0
[28]	Snowballing	MARTÍN *et al.*	2017	0
[10]	Mapping	FETAJI *et al.*	2019	0
[32]	Mapping	PASCHOAL *et al.*	2019	0
[33]	Mapping	PÉRAIRE	2019	0
[16]	Mapping	GREN	2020	0
[41]	Mapping	TEINIKER; SEUCHTER	2020	0

Source: Author (2021).

2014) received two citations, and the papers (HEROLD *et al.*, 2011; TOWEY, 2015; SCHEFER-WENZL; MILADINOVIC, 2017) only one. The other twenty (76.9%) did not receive any cross-citations.

Table 4 lists some data extracted for each of the thirty-four papers included in this literature review. For instance, eighteen papers described the active methodology adopted for in-class activities and explained details about study guides. 60.5% (twenty-three) of the papers reported at least one tool used to support the FC method. Regarding the evaluation aspects, eighteen papers presented at least one evaluation. Twelve studies (PUGSEE, 2017; JOHNSON *et*

Figure 7 – Cross-citations graph for papers included by systematic mapping study



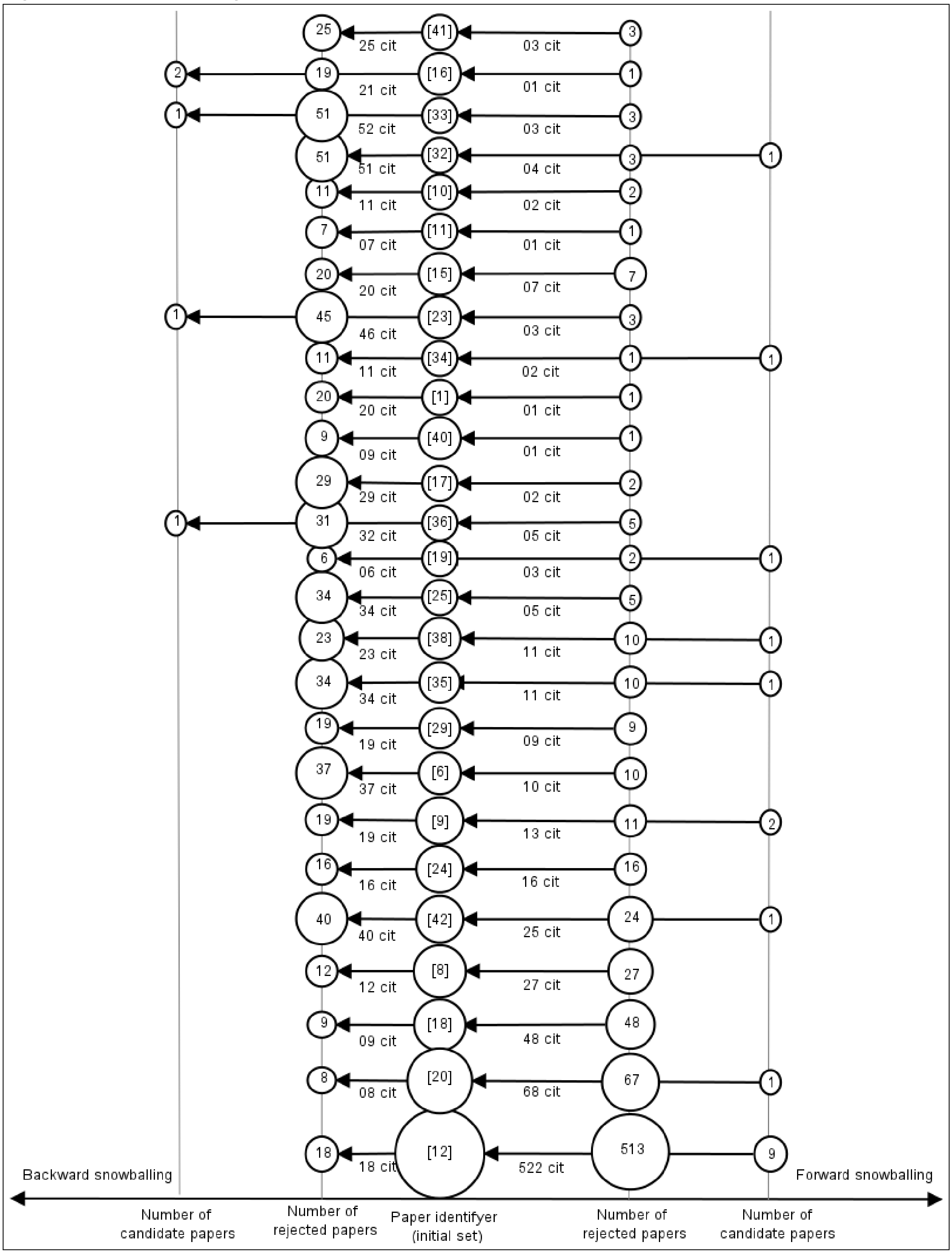
Source: (VERAS *et al.*, 2020)

al., 2016; HEROLD *et al.*, 2011; HEROLD *et al.*, 2012; KOCHUMARANGOLIL; RENUMOL, 2018; GLAZUNOVA *et al.*, 2020; ÖZYURT; ÖZYURT, 2018; PUGSEE, 2018; JOHNSON, 2019; TROYA *et al.*, 2021; LIN, 2019; SCHEFER-WENZL; MILADINOVIC, 2018b) perform more than two types of evaluations (e.g., survey, interviews). Thirty-one studies (81.6%) defined their evaluation goals clearly. However, twenty (52.6%) discussed the learning impacts of using FC in SE teaching. Finally, only eight papers (21.1%) describe the statistical tests with explicit hypotheses (null and alternative) declaration. Next, Figure 8 graphically shows the snowballing results (without excluding duplicate papers) for each paper from mapping.

2.4.2 SRQ1 - Active Learning Methodologies

Active learning methodologies are an FC central component to promote meaningful learning. Identifying those methodologies used in face-to-face meets is the focus of SRQ1 (*Which were the active learning methodologies used for designing the in-class activities*?). The active learning methods found were grouped according to the categories presented by Vujović *et al.* (2014). Table 6 shows the results of SRQ1. Eighteen papers (47.4%) employed the project-based learning method. For problem-based and self-direct learning, we listed 44.7% (17) of the papers. Finally, only 21.1% (8) of the papers used the team-based learning method. For example, some active methods found: Hackathons (JOHNSON *et al.*, 2016; SCHEFER-WENZL; MILADINOVIC, 2018a; SCHEFER-WENZL; MILADINOVIC, 2017; SCHEFER-WENZL; MILADINOVIC, 2018b), Lego game or gamified activities (HEROLD *et al.*, 2011; TROYA *et al.*, 2021; CHOI, 2013), programming challenges (PÉRAIRE, 2019; JOHNSON *et al.*, 2016)

Figure 8 – Snowballing results



Source: Author (2021)

and case studies (software projects) (LUBURIĆ *et al.*, 2019; ACHARYA *et al.*, 2017; PUGSEE, 2017; SUREKA *et al.*, 2013; LIN, 2019).

Table 6 – Active methodologies

Ref.	Activities	Active learning strategy
(SIMPSON; STORER, 2017)	Laboratory working on project activities (software project)	Project-based learning
(TEINIKER; SEUCHTER, 2020)	Laboratory Examinations and classroom instructions	Self-direct learning
(LUBURIĆ *et al.*, 2019)	Case study and laboratory exercise	Project-based learning
(ERDOGMUS; PÉRAIRE, 2017)	Live sessions, tech talks	Project-based learning
(ACHARYA *et al.*, 2017)	Case studies, class exercises, and case study videos	Problem-based learning
(JOHNSON *et al.*, 2016)	Timed programming task, “Workout of the day”, Hackathons	Self-direct learning
(PASCHOAL *et al.*, 2019)	Activity in pairs	Problem-based learning
(FORTUNA *et al.*, 2018)	Tests, quizzes, group work, presentations, and discussions	Self-direct learning
(SU *et al.*, 2016)	Hands-on (coding, writing, and presenting)	Self-direct learning
(SCHEFER-WENZL; MILADINOVIC, 2017)	Assignments, laboratories, and discussions	Project-based learning
(ELLIOTT, 2014)	Hands-on activities	Problem-based learning
(LEE *et al.*, 2015)	Practical hands-on, SE Studio	Project-based learning
(PUGSEE, 2017)	Discussions and group works	Project-based learning
(GREN, 2020)	Interactive group learning activities	Self-direct learning
(GLAZUNOVA *et al.*, 2020)	Interactive lectures and laboratory work	Problem-based learning
(TOWEY, 2015)	Laboratory sessions (pairs)	Team-based learning
(GANNOD *et al.*, 2008)	Programming assignments	Self-direct learning
(KIAT; KWONG, 2014)	Case study	Project-based learning
(KOCHUMARANGOLIL; RENUMOL, 2018)	Hands-on activities	Problem-based learning
(HEROLD *et al.*, 2011)	Quizzes, discussion, project work	Project-based learning
(FETAJI *et al.*, 2019)	Cognitive work	Problem-based learning
(HEROLD *et al.*, 2012)	Hands-on activities	Project-based learning
(SCHEFER-WENZL; MILADINOVIC, 2018a)	Hackathon, simulate software project	Self-direct learning
(PÉRAIRE, 2019)	Challenges, task project	Project-based learning
(MATALONGA *et al.*, 2017)	Exercises, short case studies, group discussions	Team-based learning
(SUREKA *et al.*, 2013)	Large-group and peer evaluation, real-client based project and SE Studio	Project and team-based learning
(PAEZ, 2017)	Coding Dojos and Coding Katas	Team-based learning
(CHOI, 2013)	Design studio and Gamestorming	Project, team and problem-based learning
(ÖZYURT; ÖZYURT, 2018)	Laboratory sessions (pairs)	Problem and Team-based learning
(PUGSEE, 2017)	Exercises, case studies and workshops	Project-based learning
(LIN, 2019)	Hands-on activities and case studies	Team-based learning
(SCHEFER-WENZL; MILADINOVIC, 2018b)	Hands-on (programming) activities, Hackathon	Project-based learning
(JOHNSON, 2019)	Timed programming task, “Workout of the day”, Tests	Project-based and Self-direct learning
(TROYA *et al.*, 2021)	Gamification quizzes and Hands-on activities	Project and team-based learning
(ERDOGMUS; PÉRAIRE, 2017)	Quizzes, Q&A, and supervised activities	Project-based learning
(PASCHOAL *et al.*, 2020)	Exercises/questions and supervised activities	Problem-based learning
(MARTÍN *et al.*, 2017)	Laboratory sessions	Project-based learning

Source: Author (2021).

2.4.3 SRQ2 - Other Pedagogical Strategies

For SRQ2 (*What other teaching/learning methods were used in collaboration with the flipped classroom method*?), seventeen papers (44.7%) did not combine any other method with flipped classroom (ERDOGMUS; PÉRAIRE, 2017; PASCHOAL *et al.*, 2019; CHETLUR *et al.*, 2014; PUGSEE, 2017; GREN, 2020; TOWEY, 2015; GANNOD *et al.*, 2008; KIAT; KWONG, 2014; FETAJI *et al.*, 2019; LUBURIĆ *et al.*, 2019; ACHARYA *et al.*, 2017; FORTUNA *et al.*, 2018; SU *et al.*, 2016; ELLIOTT, 2014; GLAZUNOVA *et al.*, 2020; LIN, 2019; ÖZYURT; ÖZYURT, 2018).

Six papers (15.8%) combined flipped classroom with gamification. For instance, in Herold *et al.* (2011), game-based instructional content was used in the agile software development context with LEGO. Johnson *et al.* (2016), Johnson (2019) propose an "athletic" method, a kind of gamification, to motivate students during in-class exercises. Herold *et al.* (2012) designed an analogic game about agile development principles to be played in the classroom. In this game, the students playfully form small groups and learn about the agile method. Troya *et al.* (2021) presents an experience report on FC to the laboratory sessions of an undergraduate computer science course. The in-class lessons started with a 6-question quiz using the game-based platform Kahoot to check students' understanding of the videos. Finally, in Matalonga *et al.* (2017), a serious game was introduced in the classes with team-based learning principles to keep the students engaged during in-class activities.

Four papers (SCHEFER-WENZL; MILADINOVIC, 2017; SCHEFER-WENZL; MILADINOVIC, 2018a; ERDOGMUS *et al.*, 2019; SCHEFER-WENZL; MILADINOVIC, 2018b) applied FC combining with Just-in-Time Teaching (JiTT) elements and mobile learning. JiTT is a teaching method that combines web-based assignments with active in-class hours. Teiniker and Seuchter (2020) reported improvements in the FC model by combining it with inductive learning aspects. Inductive learning is based on examples that provide a natural approach for part-time students (theoretical concepts can be taught more practically). In Simpson and Storer (2017), professors used mentoring as a complementary pedagogical method for improving project experiences in flipped classes. Paez (2017) describes the experience of adopting a flipped classroom approach combined with some other non-traditional teaching techniques, such as *Coding Dojo* and *Coding Kata*. *Coding Dojo* is a programming challenge where students are encouraged to participate and share their programming skills with the audience while solving the problem. *Coding Kata* is a programming practice based on repetition and practice to improve skills.

The FC method is slightly a complement to an approach named Software Design Studio in Lee *et al.* (2015), Sureka *et al.* (2013) and Choi (2013). This method provided an opportunity to practice SE principles and concepts in broad complex project contexts. In Péraire (2019), the course delivery format is based on a blended approach (named dual-track agile), combining flipped and traditional classes in a project-based learning context. Dual-track agile approach combines the traditional agility strengths and user experience approaches. Finally, FC is one of the methods used in an activity-oriented teaching strategy in Kochumarangolil and Renumol (2018).

2.4.4 SRQ3 - Tools

For SRQ3 (*Which tools were used to support the flipped classroom method in Software Engineering classes*?), 57.9% of the studies (22) provides information about content production tools and 27 (71.1%) about content consumption tools, both described in the following.

Content Production Tools. Twenty-two papers (57.9%) did not refer to how the educational content was produced. Looking into the other part of the reviewed papers, there is a list of tools used for content creation: Microsoft PowerPoint and Apple Keynote for producing slide presentations (GANNOD *et al.*; HEROLD *et al.*; HEROLD *et al.*; GLAZUNOVA *et al.*; ACHARYA *et al.*); ProfCast to produce the podcasts (GANNOD *et al.*, 2008); Google Docs to create and share text documents (ERDOGMUS; PÉRAIRE, 2017); Google Sheets to create and share spreadsheet documents (ELLIOTT, 2014; CHETLUR *et al.*, 2014) ; iMovie for capturing full-motion talking head lectures (GANNOD *et al.*, 2008); iWeb for deploying podcasts on a standard web server (PASCHOAL *et al.*, 2019; GANNOD *et al.*, 2008); Blackboard to save and deploy PowerPoint and PDF files, as well as for grade-book and assignment management (HEROLD *et al.*, 2011; HEROLD *et al.*, 2012; KOCHUMARANGOLIL; RENUMOL, 2018; FORTUNA *et al.*, 2018; GANNOD *et al.*, 2008); Lucidchart to collaboratively diagramming system (ELLIOTT, 2014); Socrative and Kahoot for quizzes (MARTÍN *et al.*, 2017) ; Panopto to record videos (ERDOGMUS; PÉRAIRE, 2017); Screenflow to screencast and video editing software (ERDOGMUS; PÉRAIRE, 2017; JOHNSON *et al.*, 2016); Moodle (or similar LMS) to compose and delivery content (GLAZUNOVA *et al.*, 2020; TEINIKER; SEUCHTER, 2020; CHETLUR *et al.*, 2014; PASCHOAL *et al.*, 2019; PASCHOAL *et al.*, 2020; ERDOGMUS *et* *al.*, 2019); Microsoft Sway (GLAZUNOVA *et al.*, 2020); and Miro and Camtasia Studio for recording videos (GLAZUNOVA *et al.*, 2020; PAEZ, 2017; MARTÍN *et al.*, 2017).

Content Consumption Tools. Eighteen papers (47,4%) used a browser for students to consume all educational material. Thirteen papers (34.2%) did not specify any content consumption tool. Looking into the other part of the reviewed papers, there is a list of tools used for content consumption: Adobe Connect Pro for in-class assessment (the assessment is performed through tool polls that cover the assigned materials) (ERDOGMUS; PÉRAIRE, 2017); iTunes music software system as a podcasting client (or a non-podcasting client) (GANNOD *et al.*, 2008); Online Git service for programming activities (ERDOGMUS; PÉRAIRE, 2017; GLAZUNOVA *et al.*, 2020; JOHNSON *et al.*, 2016; TEINIKER; SEUCHTER, 2020); A proprietary tool was described by (PÉRAIRE, 2019; LIN, 2019; PUGSEE, 2018); Facebook group (ÖZYURT; ÖZYURT, 2018); YouTube (browser or app) for video consumption (ACHARYA *et al.*, 2017; TEINIKER; SEUCHTER, 2020; LEE *et al.*, 2015; HEROLD *et al.*, 2012; PUGSEE, 2017; KIAT; KWONG, 2014; HEROLD *et al.*, 2011; JOHNSON *et al.*, 2016; FETAJI *et al.*, 2019; GREN, 2020; CHETLUR *et al.*, 2014; MARTÍN *et al.*, 2017; TROYA *et al.*, 2021; SUREKA *et al.*, 2013; PAEZ, 2017); DVD (CHOI, 2013); and Google Classroom, Moodle or Canvas LMS as a consumer environment (KOCHUMARANGOLIL; RENUMOL, 2018; FETAJI *et al.*, 2019; ERDOGMUS *et al.*, 2019; PASCHOAL *et al.*, 2020). In Herold *et al.* (2012), the students freely chose their method for consuming the lecture material.

Finally, it is possible to observe that neither paper made adaptations (or personalizations) during content delivery to learners (i.e., the same content was delivered to all students independently from their learning profiles or knowledge background). However, two studies (of the same authors' group) mentioned a manual adaptation (SCHEFER-WENZL; MILADINOVIC, 2018a; SCHEFER-WENZL; MILADINOVIC, 2017). In this adaptation, professors could give feedback to each student before the next in-class lesson and manually adjust the lesson contents.

2.4.5 SRQ4 - Evaluations methods

In the fifth specific research question (*What evaluation methods were performed in the studies?*), the goal was to discover which evaluation methods were performed in the studies, so it was evaluated if the studies selected were somehow validated. The seventh exclusion criterion (EC7) eliminated all articles without any evaluation. Paper evaluation methods were organized according to the methods presented below.

- **Questionnaire/survey**: (HEROLD *et al.*, 2012; GANNOD *et al.*, 2008; ERDOGMUS; PÉRAIRE, 2017; KOCHUMARANGOLIL; RENUMOL, 2018; FORTUNA *et al.*, 2018; TEINIKER; SEUCHTER, 2020; GLAZUNOVA *et al.*, 2020; ELLIOTT, 2014; FETAJI *et al.*, 2019; JOHNSON *et al.*, 2016; MATALONGA *et al.*, 2017; PUGSEE, 2017; SUREKA *et al.*, 2013; PAEZ, 2017; PUGSEE, 2018; LIN, 2019; JOHNSON, 2019; TROYA *et al.*, 2021; MARTÍN *et al.*, 2017)
- **Interview**: (HEROLD *et al.*, 2012; SCHEFER-WENZL; MILADINOVIC, 2017; SIMPSON; STORER, 2017; LEE *et al.*, 2015; SCHEFER-WENZL; MILADINOVIC, 2018b)
- **Exams/Tests/Quizzes**: (HEROLD *et al.*, 2012; TEINIKER; SEUCHTER, 2020; FETAJI *et al.*, 2019; GREN, 2020; TOWEY, 2015; CHOI, 2013; ÖZYURT; ÖZYURT, 2018; PUGSEE, 2018; LIN, 2019; TROYA *et al.*, 2021; ERDOGMUS *et al.*, 2019; MARTÍN *et al.*, 2017)
- **Activities Evaluation (e.g., projects, pair programming exercises, laboratories)**: (GANNOD *et al.*, 2008; KIAT; KWONG, 2014; SCHEFER-WENZL; MILADINOVIC, 2017; ERDOGMUS; PÉRAIRE, 2017; KOCHUMARANGOLIL; RENUMOL, 2018; TEINIKER; SEUCHTER, 2020; GLAZUNOVA *et al.*, 2020; ACHARYA *et al.*, 2017; CHETLUR *et al.*, 2014; ELLIOTT, 2014; HEROLD *et al.*, 2011; JOHNSON *et al.*, 2016; LEE *et al.*, 2015; LUBURIĆ *et al.*, 2019; MATALONGA *et al.*, 2017; PASCHOAL *et al.*, 2019; PÉRAIRE, 2019; PUGSEE, 2017; SCHEFER-WENZL; MILADINOVIC, 2018a; SIMPSON; STORER, 2017; SU *et al.*, 2016; PAEZ, 2017; CHOI, 2013; ÖZYURT; ÖZYURT, 2018; PUGSEE, 2018; SCHEFER-WENZL; MILADINOVIC, 2018b; JOHNSON, 2019; PASCHOAL *et al.*, 2020)

One paper can contain one or more evaluation methods. This literature review is interesting to know the paper evaluation methods and what was assessed. Below is what each paper assessed.

- **Motivation**: (HEROLD *et al.*, 2012; TOWEY, 2015; SIMPSON; STORER, 2017; SCHEFER-WENZL; MILADINOVIC, 2017; FETAJI *et al.*, 2019; JOHNSON *et al.*, 2016; SCHEFER-WENZL; MILADINOVIC, 2018a; SUREKA *et al.*, 2013; LIN, 2019; SCHEFER-WENZL; MILADINOVIC, 2018b; JOHNSON, 2019; MARTÍN *et al.*, 2017)
- **Attitude or engagement**: (GANNOD *et al.*, 2008; TOWEY, 2015; SIMPSON; STORER, 2017; ERDOGMUS; PÉRAIRE, 2017; GLAZUNOVA *et al.*, 2020; PAEZ, 2017; ÖZYURT; ÖZYURT, 2018; PUGSEE, 2018; LIN, 2019; JOHNSON, 2019; TROYA *et al.*, 2021)

- **Students' acceptance and satisfaction**: (GANNOD *et al.*, 2008; PUGSEE, 2017; JOHNSON *et al.*, 2016; ELLIOTT, 2014; SCHEFER-WENZL; MILADINOVIC, 2017; KOCHUMARANGOLIL; RENUMOL, 2018; FORTUNA *et al.*, 2018; TEINIKER; SEUCHTER, 2020; SIMPSON; STORER, 2017; LEE *et al.*, 2015; MATALONGA *et al.*, 2017; CHOI, 2013; TROYA *et al.*, 2021; MARTÍN *et al.*, 2017)
- **Performance or Learning impact**: (HEROLD *et al.*, 2012; GANNOD *et al.*, 2008; KIAT; KWONG, 2014; SU *et al.*, 2016; ACHARYA *et al.*, 2017; SCHEFER-WENZL; MILADINOVIC, 2018a; PASCHOAL *et al.*, 2019; FETAJI *et al.*, 2019; HEROLD *et al.*, 2011; LUBURIĆ *et al.*, 2019; CHETLUR *et al.*, 2014; ELLIOTT, 2014; SIMPSON; STORER, 2017; SCHEFER-WENZL; MILADINOVIC, 2017; ERDOGMUS; PÉRAIRE, 2017; KOCHUMARANGOLIL; RENUMOL, 2018; TEINIKER; SEUCHTER, 2020; GLAZUNOVA *et al.*, 2020; GREN, 2020; JOHNSON *et al.*, 2016; LEE *et al.*, 2015; MATALONGA *et al.*, 2017; PÉRAIRE, 2019; PUGSEE, 2017; PAEZ, 2017; CHOI, 2013; ÖZYURT; ÖZYURT, 2018; PUGSEE, 2018; LIN, 2019; SCHEFER-WENZL; MILADINOVIC, 2018b; JOHNSON, 2019; TROYA *et al.*, 2021; ERDOGMUS *et al.*, 2019; PASCHOAL *et al.*, 2020; MARTÍN *et al.*, 2017)

A large number of papers (30) did not present in their evaluation methods the use of a randomized experiment with a control group (78.9%). Five papers used a non-randomized control group: (1) quantitative methods divided into two groups (control and experiment) were used in a research study (FETAJI *et al.*, 2019). They used questionnaires with 9 questions for cross-checking students in different flipped classroom aspect; (2) the paper performed an analysis of the data from 27 students (13 from the control group and 14 from the experimental group) to verify its impact as a teaching model, concerning learning, knowledge retention, and study effort, from the researcher's point of view, in the context of software engineering students learning software testing (PASCHOAL *et al.*, 2019); (3) In Luburić *et al.* (2019) a comparative analysis for Security Design Analysis teaching using a flipped classroom was performed between students group using the flipped method and another group using traditional laboratories; (4) Paschoal *et al.* (2020) used a traditional in-class period for control group (42 students) and the flipped classroom model were described as experimental group (51 students); (5) In Lin (2019), the experimental group students learned with the flipped-classroom learning approach, while the students in the control group learned with the traditional-classroom learning approach. The experimental results show a significantly improved student learning achievement, motivation, attitude, and problem-solving ability compared to the traditional classroom learning approach.

Three papers used a randomized control group: (1) for a course given in Sweden, grades and course evaluation questionnaires were collected for all four years the course was given. In the first year of this article (2014), the entire course was given using classical lecturing for 50 students, who served as a control group. The second, third, and fourth years served as the experimental group, with some variation in the fourth year (169 students) (GREN, 2020); (2) Erdogmus *et al.* (2019) presented a new intervention in a flipped software engineering course over two semesters. They compared the average scores of students who completed pre- and post-test quizzes (treatment group T) to those who did not receive the prep quizzes (control group C). The paper results demonstrate an improved flipped software engineering course; (3) controlled experiments were designed to measure the effectiveness of inverted classrooms in software engineering education in Choi (2013). The population of the control group (inverted classroom) and the experimental group (traditional lecture) was 38 and 35, respectively. The authors concluded that the concept of mentoring in an inverted classroom affects educating several software engineering skills and the ability to project progress through face-to-face education during activities.

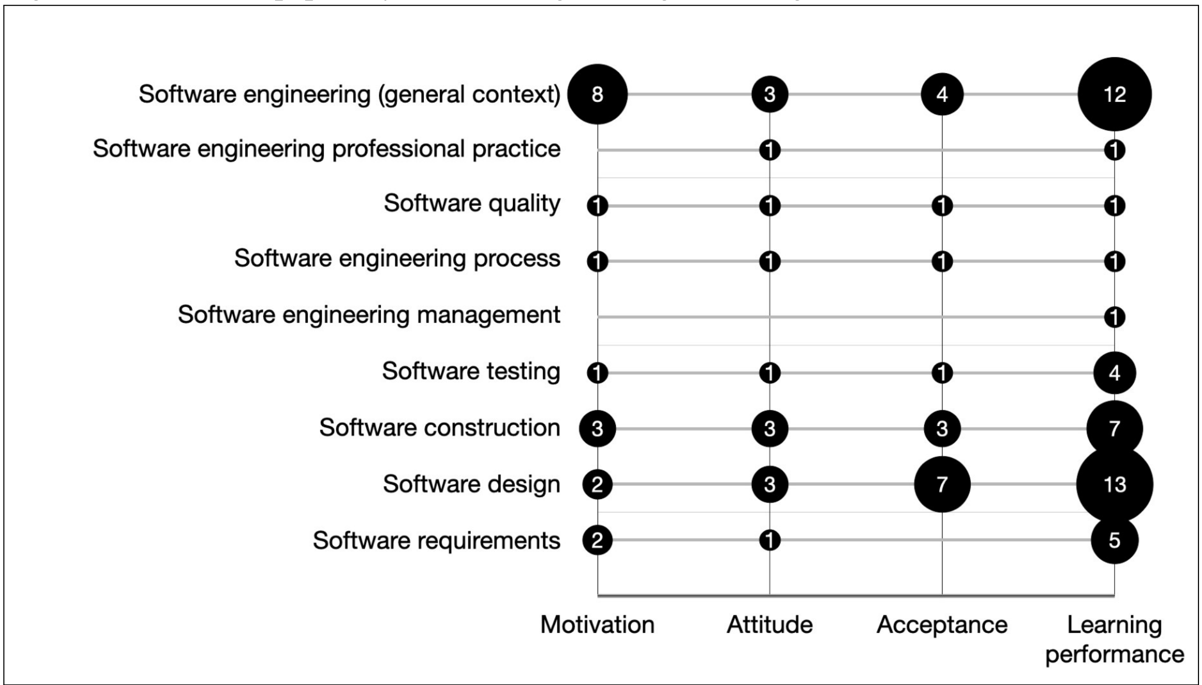
The horizontal axis in Figure 9 shows the distribution of papers by evaluated methods. Most studies (92.1%) evaluated the impact on learning, while 36.8% of the students accepted and satisfaction with the method used. Another 31.6% assessed motivation and, finally, 26.3% attitude. The figure also shows the classification of papers identified in the SE's knowledge area and evaluation approach. The results show two different trends: most of the works cover the specific area of software design; or software engineering in a general context.

2.4.6 SRQ5 - Student learning impact

To answer SRQ5 (*What were the main impacts on student learning?*), papers were classified into two categories of learning impact. The first impact is on improved understanding and knowledge of subjects studied, while the second is on increased student performance. We found a positive effect on student learning in thirty-five papers (92.1%).

Seventeen of thirty-five papers (48.6%) reported improvements in the understanding or knowledge acquired of the studied topics. A cycle of exams showed that the students were able to keep the fundamental knowledge achieved during the course module (SCHEFER-WENZL; MILADINOVIC, 2018a; HEROLD *et al.*, 2011). In Erdogmus *et al.* (2019), the assessments

Figure 9 – Assessed papers by software engineering knowledge areas



Source: Author (2021)

improved students' preparedness for in-class live sessions. Two comparative studies showed that most students developed better critical thinking, analysis, and skill design than in traditional SE classes (PUGSEE, 2017; SIMPSON; STORER, 2017; PUGSEE, 2018). The studies of Matalonga *et al.* (2017), Teiniker and Seuchter (2020), Péraire (2019), Troya *et al.* (2021) reported an increase in student effort and engagement in watching the video lectures and reading the class-related documents when SE professors used the Flipped Classroom method. Luburić *et al.* (2019) present results that the student teams of instance 2016/2017 have achieved better overall correctness and completeness on the threat identification task compared to the student teams of instance 2015/2016. Johnson (2019) evaluated an "athletic" pedagogy over four years and six semesters with 286 students and found pieces of evidence to increase the students' understanding. The study of Lee *et al.* (2015) captured, during the students' evaluation, positive results concerning the use of Software Design Studio - a method proposed by the authors- combined with the Flipped Classroom approach. Comparative studies showed that students in the flipped classroom group have a more in-depth understanding of the course content than the control group (PASCHOAL *et al.*, 2019; ELLIOTT, 2014). In (CHOI, 2013), students in the inverted classroom group had a deeper understanding of the course content than in the traditional classroom. The flipped classroom methods helped improve the understanding of complex concepts compared to traditional classes (SCHEFER-WENZL; MILADINOVIC, 2017) and enabled SE learners to have a better learning experience. Among those papers, the only one using a non-randomized experiment with a control group was (PASCHOAL *et al.*, 2019), and two used a randomized control group (CHOI, 2013; ERDOGMUS *et al.*, 2019).

The other eighteen papers (51.4%) reported an increase in student performance in SE courses (e.g., grades, scores). In Herold *et al.* (2012), Paez (2017), Özyurt and Özyurt (2018), Pugsee (2018), Lin (2019), for instance, the experiment results showed a significant increase in student grades, meaning that the flipped classroom method was successful. In Kiat and Kwong (2014), Glazunova *et al.* (2020), students learned in their own time and out-of-class, allowing professors to improve the interaction, collaboration, and individual learning monitoring concerning the task performance (hands-on activities). Martín *et al.* (2017) presents good results regarding flipped classrooms during lessons of the bachelor's degree in Software Engineering. Students have increased involvement in the class and have valued the resources elaborated as applicable.

Chetlur *et al.* (2014) found better student performance using a tool (called EduPaL) to support the flipped classes. EduPaL is a plug-and-learn solution that utilizes a low-cost USB flash drive as the main infrastructure for learners to watch videos and answer quizzes. In the study of Gren (2020), FC improved students' grades when the professor had pedagogical training and experience in flipping classes. Johnson *et al.* (2016) collected empirical data on student performance, self-reported assessments (by students), and classroom surveys. Much of this data generated positive impacts but with some outstanding challenges (e.g., fear of failure and activity performance anxiety at the beginning of the course).

Kochumarangolil and Renumol (2018) interviewed thirty-four students and eight teachers after conducting interdisciplinary approaches for solving problems on the flipped learning method. The course used a cloud-based environment to support the learning process. Students said that classroom activities were useful in understanding the concepts and clearing up doubts and that the method helped them learn more about SE's practical side. However, they reported spending more time completing all activities. Acharya *et al.* (2017) reported that the student's grades with the flipped classroom method were better than students' scores from previous years (traditional classroom). Paschoal *et al.* (2020) presents an experimental study and analyzes the students' effectiveness concerning test requirements. The study compared test requirements produced by students who studied with the flipped classroom model and students who studied with the traditional teaching model.

Su *et al.* (2016) partially assessed the flipped classroom effectiveness by collecting student performance in 2013 and 2014. In 2014, most students achieved good grades, while in 2013, it was the other way around. Luburić *et al.* (2019) compared the laboratory exercises implemented by the students. They reported that those using the flipped classroom method produced better results than those in traditional laboratory classes. Finally, Fetaji *et al.* (2019) and Erdogmus and Péraire (2017) attested students' performance improvements with varied improvement rates. Most students did reasonably well in both studies. Among these eighteen papers, three used a non-randomized experiment with a control group (FETAJI *et al.*, 2019; LIN, 2019; PASCHOAL *et al.*, 2020) and another used a randomized experiment with a group control (GREN, 2020).

2.4.7 SRQ6 - Advantages and challenges

The SRQ6 question aims to identify the advantages and challenges reported in the selected papers (*What are the main benefits and challenges regarding using Flipped Classroom in Software Engineering teaching?*). Below is a summary of the main benefits reported.

- **An improvement in student's satisfaction; Students' motivation, dedication, and confidence** (MATALONGA *et al.*, 2017; JOHNSON *et al.*, 2016; SCHEFER-WENZL; MILADINOVIC, 2017; LEE *et al.*, 2015; FETAJI *et al.*, 2019; ELLIOTT, 2014; SIMPSON; STORER, 2017; CHOI, 2013; LIN, 2019; SCHEFER-WENZL; MILADINOVIC, 2018b; JOHNSON, 2019; TROYA *et al.*, 2021)
- **Improvements in Software Engineering learning; in the monitoring of student progress regarding out-of-class studies** (PASCHOAL *et al.*, 2019; CHETLUR *et al.*, 2014; PUGSEE, 2017; GREN, 2020; SCHEFER-WENZL; MILADINOVIC, 2018a; MATALONGA *et al.*, 2017; TEINIKER; SEUCHTER, 2020; KIAT; KWONG, 2014; CHOI, 2013; PUGSEE, 2018; ERDOGMUS *et al.*, 2019; PASCHOAL *et al.*, 2020; MARTÍN *et al.*, 2017)
- **An increase in students' engagement** (ERDOGMUS; PÉRAIRE, 2017; GREN, 2020; TOWEY, 2015; HEROLD *et al.*, 2011; PÉRAIRE, 2019; KOCHUMARANGOLIL; RENUMOL, 2018; SU *et al.*, 2016; ACHARYA *et al.*, 2017; SIMPSON; STORER, 2017; PAEZ, 2017; PASCHOAL *et al.*, 2020; MARTÍN *et al.*, 2017)
- **Optimization of class time** (ERDOGMUS; PÉRAIRE, 2017; PASCHOAL *et al.*, 2019; HEROLD *et al.*, 2012; MATALONGA *et al.*, 2017; SU *et al.*, 2016; ACHARYA *et al.*,

2017; PUGSEE, 2018; TROYA *et al.*, 2021)

- **Encourage students to form study groups; Work integrated learning; Student's performance; Collaborative learning** (SU *et al.*, 2016; GLAZUNOVA *et al.*, 2020; LUBURIĆ *et al.*, 2019; ÖZYURT; ÖZYURT, 2018)

Another advantages were mentioned less frequently, such as, enhancement of instructor-student and student-to-student relationships (ERDOGMUS; PÉRAIRE, 2017; SIMPSON; STORER, 2017; SUREKA *et al.*, 2013); class ubiquity (CHETLUR *et al.*, 2014) and content delivered asynchronously (GANNOD *et al.*, 2008).

Some challenges cited by the studies are highlighted below:

- **Overworked and time-constrained professors; difficulties in class preparation; low materials quality; huge effort and time investment by students** (ERDOGMUS; PÉRAIRE, 2017; PASCHOAL *et al.*, 2019; LEE *et al.*, 2015; CHETLUR *et al.*, 2014; PUGSEE, 2017; GREN, 2020; GANNOD *et al.*, 2008; KIAT; KWONG, 2014; HEROLD *et al.*, 2011; HEROLD *et al.*, 2012; MATALONGA *et al.*, 2017; PAEZ, 2017; PUGSEE, 2018; PASCHOAL *et al.*, 2020; MARTÍN *et al.*, 2017)
- **Difficulty in sustaining student motivation** (ERDOGMUS; PÉRAIRE, 2017; LEE *et al.*, 2015; KIAT; KWONG, 2014; SCHEFER-WENZL; MILADINOVIC, 2018a; PÉRAIRE, 2019; MATALONGA *et al.*, 2017; TEINIKER; SEUCHTER, 2020; PUGSEE, 2018)
- **Scalability (e.g., big classes); Using the approach in distance learning courses** (ERDOGMUS; PÉRAIRE, 2017; PASCHOAL *et al.*, 2019; LEE *et al.*, 2015; GANNOD *et al.*, 2008; ACHARYA *et al.*, 2017; SIMPSON; STORER, 2017)
- **Learning assessments; lack of feedback from students; absence of the professor to assist students in content understanding out-of-classroom** (CHETLUR *et al.*, 2014; GANNOD *et al.*, 2008; KOCHUMARANGOLIL; RENUMOL, 2018; SUREKA *et al.*, 2013; PAEZ, 2017; CHOI, 2013)
- **Heterogeneity of the student dedication in both in-class and out-of-class activities; Sustain engagement in practical activities (after getting solid knowledge from theoretical activities)** (PASCHOAL *et al.*, 2019)(SCHEFER-WENZL; MILADINOVIC, 2018a; JOHNSON *et al.*, 2016; SCHEFER-WENZL; MILADINOVIC, 2017; GANNOD *et al.*, 2008; FORTUNA *et al.*, 2018)

Reviewed studies reported other challenges. For instance, achieving high performance can cause anxiety disorder in students (JOHNSON *et al.*, 2016; HEROLD *et al.*, 2012;

JOHNSON, 2019). Also, the model requires an Internet connection with sufficient quality to watch videos and, sometimes, demands a personal computer to perform the tasks, which may not be the norm for some students, especially in developing countries (CHETLUR *et al.*, 2014; FETAJI *et al.*, 2019). Authors also described cultural (TOWEY, 2015; TEINIKER; SEUCHTER, 2020) and technological challenges (GLAZUNOVA *et al.*, 2020). Another challenge is to propose activities related to real-world industrial projects (LEE *et al.*, 2015; PÉRAIRE, 2019; SIMPSON; STORER, 2017).

Papers (ELLIOTT, 2014; SU *et al.*, 2016; LUBURIĆ *et al.*, 2019; ÖZYURT; ÖZYURT, 2018; LIN, 2019; SCHEFER-WENZL; MILADINOVIC, 2018b; TROYA *et al.*, 2021; ERDOGMUS *et al.*, 2019) do not mention any challenges about the use of flipped classroom in SE.

2.5 Discussions

2.5.1 Specific Research Questions

This literature review identified 38 relevant papers to answer six specific research questions related to the flipped classroom method usage in Software Engineering teaching. Initially, the study observed a tendency to explore general SE context (Section 2.4.1) with fifteen papers, and following the most explored was software design (thirteen papers). Based on the data extracted from these papers, the chapter gathered elements related to (1) active methods used in-class activity, (2) pedagogical strategies that were blended with flipped classroom method, (3) tools to support the flipped classroom method, (4) assessment methods, (5) student learning impact, and (6) advantages and challenges about flipped classroom usage. Although every flipped class has an active method during in-class activities, eighteen papers described the technique. In all documents, several active learning activities were identified and grouped into four learning strategies (project-based, problem-based, self-direct, and team-based). Furthermore, eleven papers describe in-depth the study guide used in flipped classes. A study guide without detail impedes a study's replication. About this, again, eleven papers describe any restrictions for study replicability. Regarding Emergency Remote Teaching and Distance Learning Settings, the flipped classroom is a viable solution when mixing asynchronous and synchronous classes. Fortuna *et al.* (2018) provides an overview of the motivation, challenges, and opportunities related to converting an existing Software Engineering Technology curriculum to a fully online format. The study highlights student feedback in rejecting the notion of a flipped classroom in favor of a more traditional delivery model (converted to an online format). From that chapter, two critical aspects that require more attention from teachers when using flipped classrooms are emphasized: (1) instructional material quality (e.g., videos, study guides); and (2) virtual classroom practical activities (with any active methodology). For (1) high-quality materials increase student motivation and satisfaction. However, developing them is one of the flipped classroom method challenges. While (2) virtual practical activities improve students' understanding when dealing with experience in real projects/problems, in addition to working on trust, teamwork, engagement, and experience.

Also, when combined with flipped classes, positive results on learning impact from the project- and problem-based methods usages were highlighted. For instance, authors reported students' active participation in learning, critical thinking promotion, soft skills development, and a taste of real-world projects. Thirty-five studies have shown positive effects on student learning. Eight used statistical methods to show significant conclusions ($p \leq 0.05$). For example, in (GREN, 2020), the Kruskal-Wallis statistical test was used. The authors noted that flipping the classroom increases students' exam scores, although there was no clear effect on students' perception of the course. Sixteen experimentally validated studies (e.g., experiments and case studies) for the flipped classroom method use in the software engineering teaching context, but only three of them (GREN, 2020; CHOI, 2013; ERDOGMUS *et al.*, 2019) used a randomized controlled experiment.

2.5.2 What did we learn?

This literature study on using the flipped classroom method in software engineering teaching has provided insights and lessons that can guide us toward new approaches to applying the method.

The first lesson is about the diversification in Software Engineering teaching, with an initial trend of exploring the overall context of SE and software design. This diversification suggests that the flipped classroom method can be adapted to various aspects of SE, allowing flexibility in its application. We identified gaps in some areas and opportunities for new studies, such as in Software Quality and Software Testing.

Another lesson learned is that using active learning strategies combined with the flipped classroom method can improve learning, especially during in-class practical sessions.

Regarding out-of-class studies, we observed the importance of organizing study guides and curating materials. We noted the influence of material quality on student motivation and satisfaction, emphasizing the importance of investing time and effort in developing adequate resources.

The third lesson learned pertains to the challenges and opportunities associated with using the flipped classroom method in SE education. We found that some students may resist the flipped classroom in favor of traditional models, signaling the need to analyze the quality of instructional material and in-class activities carefully. On the other hand, we observed positive outcomes in the impact on learning associated with flipped classrooms, including active student participation, promotion of critical thinking, development of social skills, and exposure to real-world projects, providing empirical evidence of positive results regarding the benefits of the flipped classroom method.

2.5.3 Research Opportunities

We didn't find any paper proposing either content adaptation or personalization during the student guide delivery to learners (i.e., the same content was delivered to all students independently from their learning profiles or knowledge background). In light of this, a new research opportunity emerged, and neither of the 38 papers did not explore it: adaptive learning combined with the Flipped Classroom.

2.5.4 Threats to Validity of the SLR

Systematic literature review methods are used to avoid bias and enable a more objective analysis of results. However, the technique is not free from bias. For example, in this literature review, the author applied inclusion and exclusion criteria to titles and abstracts. Consequently, each paper was only reviewed by a single author. This approach poses a threat to the reliability of the study. However, when the author had a doubt, the other authors (supervisors) consulted to make a collective final decision.

Nevertheless, some prominent studies may have been left out of this review. This threat can happen for reasons: i) the study paper is not indexed by the databases used; ii) the search string did not encompass text within the document, or it's possible that the search string did not cover specific areas of software engineering; and iii) the study is in a language other than English. To mitigate this threat, relevant electronic databases in Computer Science were used, which are often consulted for research and systematic reviews. Besides, several attempts were made to construct the final version of the search string.

2.6 Chapter summary

This chapter aimed to systematically identify the state of the art of using the FC method in SE teaching. The chapter demonstrated that FC is a promising active methodology that has positively impacted student learning and motivation in higher education, including in Computer Science programs. We started with 769 documents in the first phase (mapping study) and included 26 studies utilizing FC in SE teaching. Subsequently, in the second phase (snowballing), 1378 papers were found, and 12 new papers were included, resulting in 38 papers.

The chapter overviews the flipped classroom (FC) teaching approach in software engineering (SE) education. The chapter discussed the advantages and challenges of using the FC method in SE teaching, emphasizing how it can help students learn more actively and engage more deeply with the course material. The chapter also highlighted various tools that can be used for producing, consuming, and delivering out-of-class content, such as video lectures, online quizzes, and interactive simulations. Furthermore, other teaching methods can be combined with the flipped classroom approach to enhance effectiveness. Among these, gamification was the most frequently employed method. Overall, the chapter provided a detailed and insightful analysis of the flipped classroom approach in SE education and its potential benefits for students and instructors.

The chapter used the data from the reviewed articles to extract information for mapped aims. In order to gain deeper insights into the flipped classroom process, Chapter 3 surveyed the authors selected by literature review. The survey results extracted preliminary insights and shaped the design principles.

3 DEEPENING THE FLIPPED PROCESS IN SOFTWARE ENGINEERING TEACHING

This chapter deeps the flipped process in software engineering education to answer our study's second research question (RQ2): "*What are the characteristics that an authoring tool should have to help SE professors during the preparation of Flipped Classrooms?*".

In this chapter, we begin by presenting the results of a survey conducted among Software Engineering researchers who are knowledgeable about the Flipped Classroom method. These researchers were identified through a literature review that was reported in Chapter 2. The survey was conducted to gain a deeper understanding of the flipped process and collect insights and data about specific features that can be used to identify tool design principles supporting SE flipped classrooms. The chapter is structured as follows: Section 3.1 outlines the entire survey process and provides a detailed account of our results. Section 3.2 introduces design principles for Flipped Classroom tools, and Section 3.3 describes the design principles for Personalized Study Guide creation. Finally, in Section 3.4, we present a chapter summary.

3.1 Survey

3.1.1 Material and Methods

A survey is not simply an instrument for collecting information. It is a research method that collects data to describe, compare, or explain knowledge, attitudes, and behaviors (FINK, 1995). The survey's purpose was to deep-in the review results found and to collect information about (RQ3) which characteristics an authoring tool should have to help SE professors during the FC preparation and (RQ4) which features a tool should have to support content personalization.

We created an electronic questionnaire[1] as the data collection instrument for this research. The questionnaire begins with a Confidentiality Statement, and researchers must accept the terms presented to participate in the survey. The questionnaire was designed with three item types: closed questions, open questions, and Likert scale-based statements (LIKERT, 1932). The items were grouped into six contexts (challenge, difficulty, advantage, drawback, resource, and evaluation). A question is open when respondents are asked to frame their answers. On the other hand, a question is closed when respondents select an answer from a list of predefined choices

[1] Available at <https://forms.gle/pnfMNKpQNyQSfrRC7>

(KITCHENHAM; PFLEEGER, 2008). Table 7 presents the closed questions, while Table 8 the open ones, and finally, Table 9 lists the Likert scale items. The answer options for closed questions are shown in Subsection 3.1.4 with their results.

Table 7 – Survey closed questions

ID	Question	Key Context
Q1	DURING the planning of Flipped Classes, what were the main CHALLENGES of adopting the method for Software Engineering teaching?	Challenge
Q2	Starting in your prior experiences regarding Flipped Classes, what are the main DIFFICULTIES of using the method during Software Engineering teaching?	Difficulty
Q3	AFTER carried out Flipped Classes, what were the main ADVANTAGES of using the method during Software Engineering teaching?	Advantage
Q4	AFTER carried out Flipped Classes, what were the main DRAWBACKS of using the method during Software Engineering teaching?	Drawback

Source: Author

Table 8 – Survey open questions

ID	Question	Key Context
Q5	What do you miss most from the tools that you used to prepare your Flipped Classes?	Resource
Q6	Can you write about how do you make the students' monitoring (e.g., data about performance, engagement, motivation) while in-class studies?	Difficulty
Q7	Are there any features you would like a tool to support students (or professors) in your Flipped Classes in Software Engineering to offer while in-class studies?	Resource
Q8	Are there any features you would like a tool to support professors in your Flipped Classes in Software Engineering during the evaluation process?	Evaluation

Source: Author

Table 9 – Survey Likert-scale-based questions

ID	Question	Key Context
Q9	Choose resources that you use most often when preparing study guides (for out-of-class studies)	Resource
Q10	Choose tools that you use most often when preparing study guides (for out-of-class studies)	Resource
Q11	What tools do you use to deliver educational content for students during out-of-class studies?	Resource
Q12	Choose practical activities that you use most often while in-class studies	Resource
Q13	What were the main evaluation aspects of Flipped Classes in Software Engineering teaching?	Evaluation
Q14	What were the main evaluation instruments for Flipped Classes in Software Engineering teaching?	Evaluation
Q15	What were the main learning strategies for Flipped Classes in Software Engineering teaching?	Evaluation
Q16	In which moment do you miss more a specific tool to support the Flipped Class Method in Software Engineering teaching?	Difficulty
Q17	How can a specific tool that supports the Flipped Classroom Method in Software Engineering help students during out-of-class studies?	Advantage

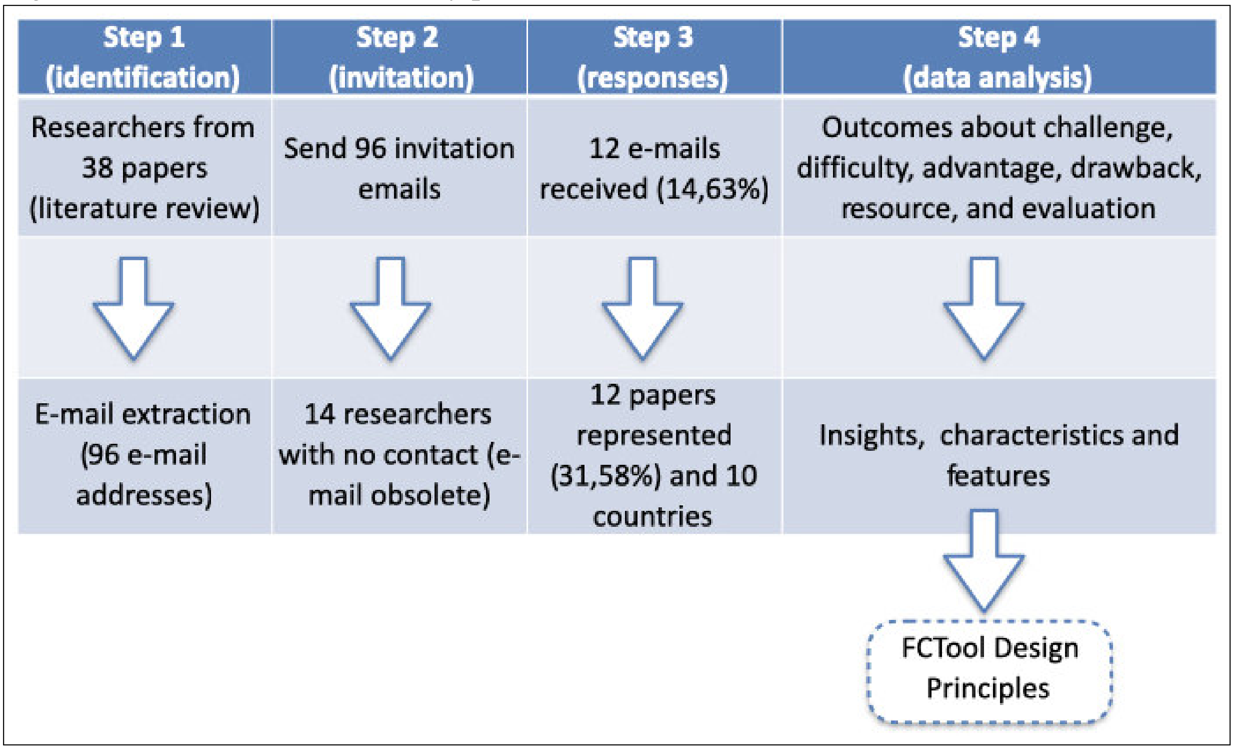
Source: Author

Nine items with Likert scale were asked of the researchers. It is generally best to use an ordinal scale for attitudes and preferences. Some scales exist: agreement, frequency, importance, quality, and evaluation scales (KITCHENHAM; PFLEEGER, 2008; BROWN, 2010). In this research, we opted for importance and frequency scales. Each statement has seven Likert-scale-based answers: Low relevant to high relevant (importance scale) and never to always (frequency scale).

3.1.2 Sample and Procedure

Ninety-six researchers were identified from 38 papers from the literature review (Chapter 2), and eighty-two email invitations were sent out to SE researchers. Figure 10 shows an overview survey procedure.

Figure 10 – Overview of the survey procedure



Source: Author

Fourteen email addresses were obsolete, and the 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 articles included (31.58%).

3.1.3 Participants

The data for this survey was collected over two months, from November to December 2021. The survey involved the participation of twelve experienced software engineering researchers who have taught in flipped classrooms across ten different 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). Out of the twelve participants, eleven researchers (91.7% of the total number) have more than five years of experience. In contrast, the remaining researchers (8.3%) have less than three years of experience.

In terms of their teaching experience, six researchers (50% of the total number) have taught more than ten flipped classes. Two researchers (16.7%) have taught fewer than ten flipped classes. Additionally, two researchers have taught fewer than five flipped classes, while the remaining two have taught fewer than three. This information provides an overview of the participants' experience and background, which can be helpful for future research and analysis.

3.1.4 Results

The first question (Q1) asked in the survey was about the main challenges of planning Flipped Classrooms in SE teaching. The results were:

a) To assure students are prepared for in-class activities **(10 researchers - 83.3%)**
b) Planning the learning assessments **(6 researchers - 50%)**
c) Professors were overworked and time-constrained for planning the Flipped Classes activities **(4 researchers - 33.3%)**
d) Difficulties in scale for big classes (Scalability) **(3 researchers - 25%)**
e) To adopt Flipped Classes in distance learning courses (Or emergency remote teaching) **(3 researchers - 25%)**
f) Sustain engagement in practical activities (after getting solid knowledge from theoretical activities) **(3 researchers - 25%)**
g) Avoid the decrease of students' attendance during in-class activities **(3 researchers - 25%)**
h) To take into account the diversity of technological barriers that the students face **(3 researchers - 25%)**

i) No particular problems **(1 researcher - 8.3%)**

The second question (Q2) referred to the difficulties of using the FC method during SE teaching. The main goal was to quantify the difficulties found. Below is the entire result list.

a) Heterogeneity of the student dedication in both in-class and out-of-class activities **(5 researchers - 41.7%)**
b) Difficulties in-class preparation materials quality **(4 researchers - 33.3%)**
c) Students' initial unreceptive behavior with the new approach **(4 researchers - 33.3%)**
d) Gather feedback from students during out-of-class studies **(4 researchers - 33.3%)**
e) Difficulty in sustaining student's motivation **(3 researchers - 25%)**
f) Quality of materials curated by the professor (Third-party materials) **(2 researchers - 16.7%)**
g) Internet connection with sufficient quality to watch videos and, sometimes, demand a personal computer to perform the tasks, which may not be the norm for part of the students **(2 researchers - 16.7%)**
h) Carry out activities related to real-world industrial projects **(2 researchers - 16.7%)**
i) Absence of the professor to assist students in content understanding during out-of-class activities **(1 researcher - 8.3%)**
j) No particular problem **(1 researcher - 8.3%)**

To deepen the benefits perceived by the researchers, the survey asked (Q3) about the main advantages of using the method in SE teaching. The main advantages pointed out were increased student engagement and collaborative learning. Below are the results for the third question.

a) An increase in students' engagement (**9 researchers - 75%**)
b) Collaborative learning (**9 researchers - 75%**)
c) Improvements in Software Engineering Learning (**8 researchers - 66.7%**)
d) Class time optimization (**8 researchers - 66.7%**)
e) Students' motivation, dedication, and confidence (**6 researchers - 50%**)

f) Positive attitudes by students (**6 researchers - 50%**)

g) Improves students' learning habits (**6 researchers - 50%**)

h) Encourage students to form study groups (**6 researchers - 50%**)

i) Enhancement of instructor-student and student-to-student relationships (**6 researchers - 50%**)

j) Improvements in student's satisfaction (**5 researchers - 41.7%**)

k) Work integrated learning (**5 researchers - 41.7%**)

l) Content delivered asynchronously (**3 researchers - 25%**)

m) Monitoring of student progress regarding out-of-class studies (**2 researchers - 16.7%**)

n) Student's performance (**2 researchers - 16.7%**)

o) Reusable materials (**1 researcher - 8.3%**)

In the fourth question (Q4), the survey asked about the drawbacks of using the FC method in SE teaching. The researchers noted that the most significant disadvantage was the effort and time invested during the instructional material preparation. Below is the complete result.

a) Initial huge effort and time investment by professors to prepare instructional/educational materials (**10 researchers - 83.3%**)

b) Students do not perform studies before practical class (**7 researchers - 58.3%**)

c) Difficulties in preparation materials (curation) from professors (**3 researchers - 25%**)

d) Lack of student feedback during out-of-class studies (**3 researchers - 25%**)

e) Huge effort and time investment by students during flipped studies (**2 researchers - 16.7%**)

f) Low engagement from students (**1 researcher - 8.3%**)

g) No particular drawbacks (**1 researcher - 8.3%**)

The fifth question (Q5) openly asked about using tools during the flipped lesson preparation. Specifically, it asked what the researcher missed most about the tool used. Three researchers listed two resources they missed: integration and lack of standardization. Two researchers mentioned integration with other tools like GitLab. The other aspect described was the lack of standardization of e-learning. It means a lack of focus on developing learning content following some standard (e.g., LOM/CC/QTI).

In question six (Q6), the survey asked the researchers to write about how they made the monitoring of students during in-class activities (e.g., performance, engagement, and motivation). The researchers answered that they monitor students in class using the following strategies: (1) empirical observations (with a preplanned spreadsheet to monitor different dimensions); (2) quizzes and final individual exam; (3) survey at the end of each class; (4) rate against expected objective; and (5) weekly attendance monitoring and formative assessment by coaches.

Question seven (Q7) asked the researchers which resources they would like in a tool to support students or teachers during SE in-class activities using FC. The form received four responses:

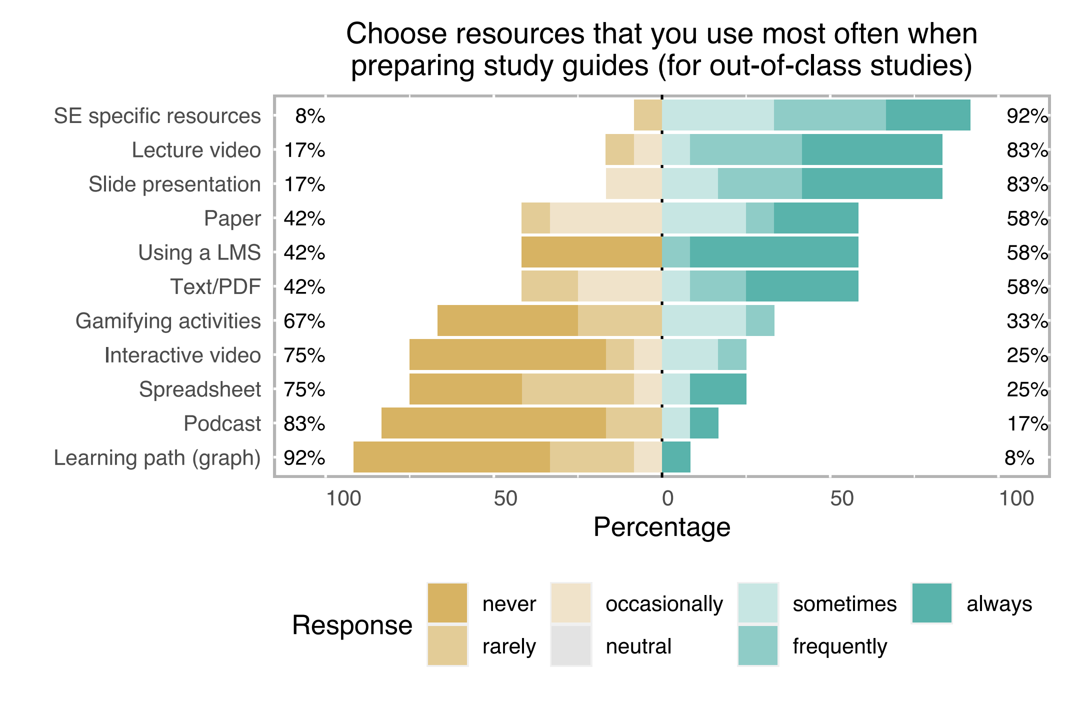
a) Integration of different online and local resources (e.g., for the use of mind maps)
b) Engagement monitoring, with details of times, duration, and specific focus of attention (e.g., which activity)
c) Visualization of multiple software project states simultaneously
d) Integrated way to track each student during each activity

The eighth question (Q8) asked the researchers about the features they would like to have in the tool to support teachers. The respondents provided a range of answers, which included (a) the capability of making automatic corrections of exercises or quizzes and (b) the continuous monitoring and tracking of student behavior during both in-class and out-of-class activities. The researchers emphasized the importance of a tool that could provide teachers with valuable insights into students' learning behavior to help them make informed decisions and improve their learning experience.

Questions Q9 through Q17 were presented to the researchers with options using a Likert (1932) scale. In question nine (Q9), the researcher initially noted the use frequency for each resource listed concerning study guide preparation. Figure 11 presents the results of Q9. The resources most used by the researchers during study guide preparation were (a) SE-specific resources, (b) video lectures, (c) slide presentations, (d) scientific papers or materials, (e) content preparation using an LMS, and (f) texts or PDFs. On the other hand, the least used resources were (a) graphs to represent learning paths, (b) podcasts, (c) spreadsheets, (d) interactive video lessons, and (e) gamified activities.

The tenth question (Q10) aims to identify the frequently used tools by teachers

Figure 11 – Resources used during study guide preparation



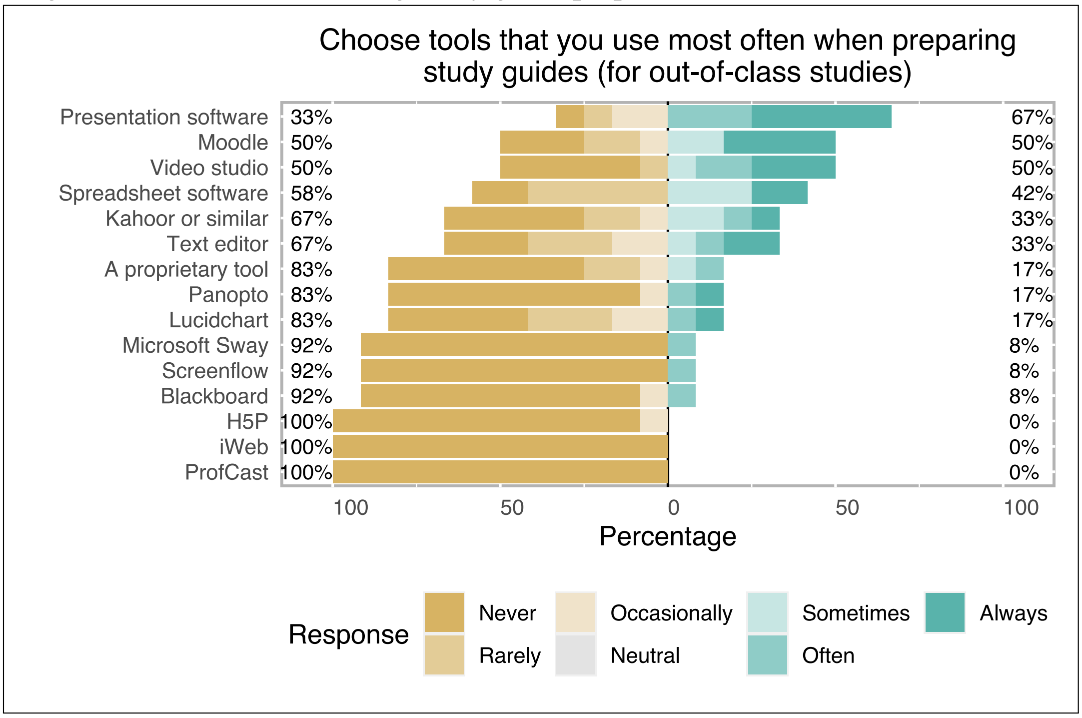
Source: Author

when preparing the study guide for flipped classrooms. The survey respondents reported that they use a variety of tools for this purpose. However, the most commonly used tools were slide presentation software such as Microsoft PowerPoint, Apple Keynote, LibreOffice Impress, and Google Presentations. Moodle software, a popular learning management system, was also frequently used by teachers. Another tool that teachers frequently used was video studio software such as iMovie, Miro, and Camtasia Studio, which is used to create engaging educational videos.

Moreover, spreadsheet software such as Microsoft Excel, Apple Numbers, LibreOffice Calc, and Google Sheets were frequently used by teachers to organize data and perform calculations. Lastly, text/PDF editors like Microsoft Word, Apple Pages, LibreOffice Writer, and Google Docs were commonly used to create and edit content. It is worth noting that the tools iWeb and ProfCast were not used by any researchers who responded to the survey. Figure 12 refers to a visual representation of these results.

Question eleven (Q11) collected answers about the tools used (frequency) to deliver educational content to students to study at home before practical class. The results show that the researchers pointed out that YouTube is the primary tool for delivering educational material to students. Website, Moodle, Online Git Service, and proprietary tools are also cited as tools used. On the other hand, iTunes (or similar for podcasting clients), Observable, Google Classroom, and Adobe Connect Pro are rarely used by researchers to deliver material to students.

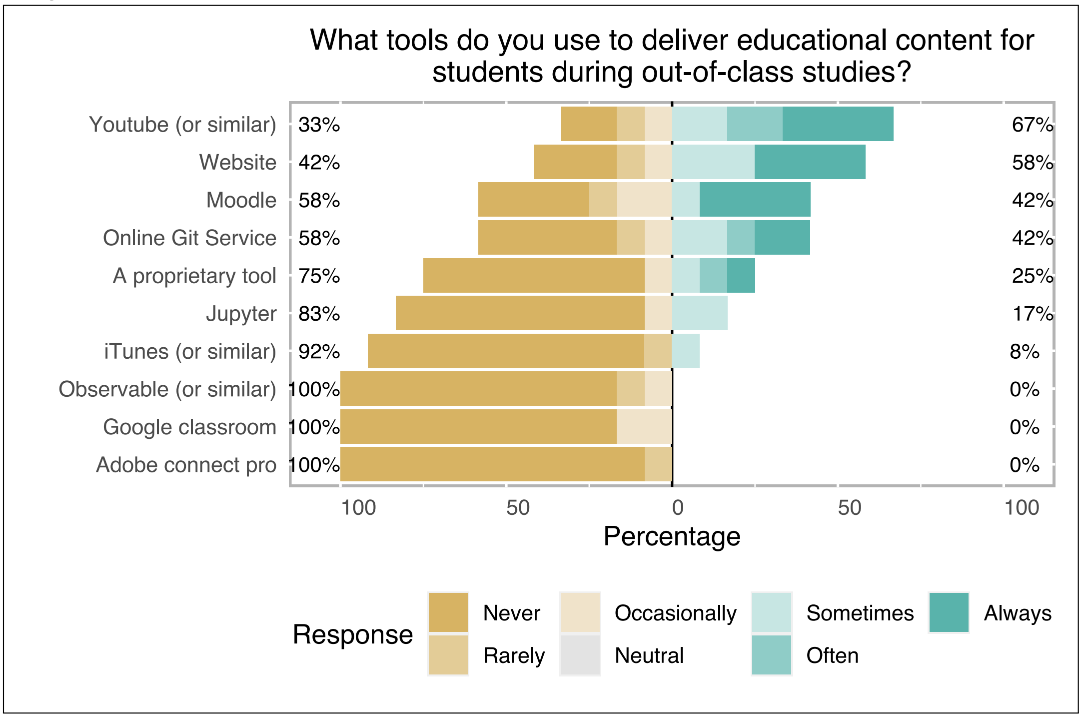
Figure 12 – Tools used during study guide preparation.



Source: Author

Figure 13 presents survey results for question eleven. The result corroborates with questions ten and eleven about the resources and tools used. For example, slides and videos as primary resources (Q9) and presentation software and video studio as tools (Q10) are aligned with Q11 results. Youtube, Websites, and Moodle were the primary tools to deliver content.

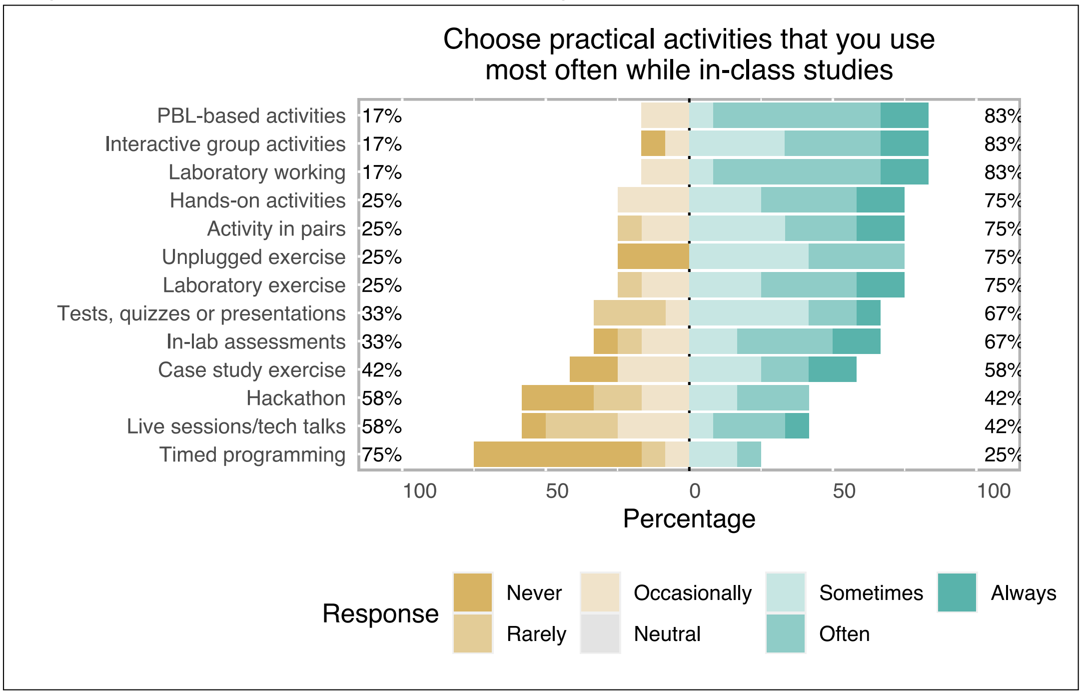
Figure 13 – Tools used to deliver education contents



Source: Author

Question twelve (Q12) identifies the in-class practical activities used most frequently among the research respondents. The most frequently performed practical activities among the researchers during in-class lessons were concentrated on group and laboratory activities with an emphasis on problem-based activities (PBL). Some respondents also cited using exercises without computers (unplugged). Figure 14 presents the percentage for each practical activity. In contrast, the least used activities by researchers were (a) hackathons, (b) live sessions and tech talks, and (c) timed programming activities.

Figure 14 – Practical activities used during in-class studies

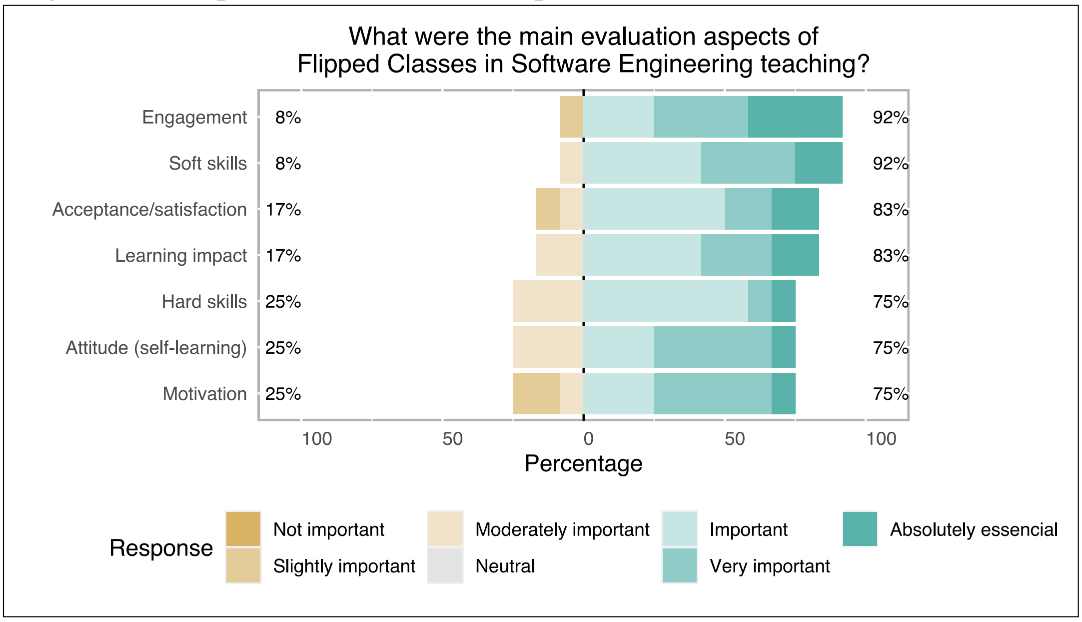


Source: Author

Question (Q13) aims to determine the significance of evaluation aspects in flipped classes when teaching software engineering (SE). The respondents' answers revealed that all the evaluation aspects hold significant importance, but most considered engagement, soft skills, acceptance/satisfaction, and learning impact slightly more crucial. The proportion of respondents who selected each evaluation aspect is depicted in Figure 15.

Next, question fourteen (Q14) asks about the importance of evaluation instruments for flipped classrooms in software engineering teaching. All respondents stated that hands-on activities are "absolutely essential" for flipped classrooms during software engineering teaching. Exams, quizzes, and tests were also pointed out as important by 92% of the respondents. Questionnaires or surveys were considered important by 67% of the researchers, while 87% answered that interview has low importance for FC in SE teaching. Figure 16 presents the

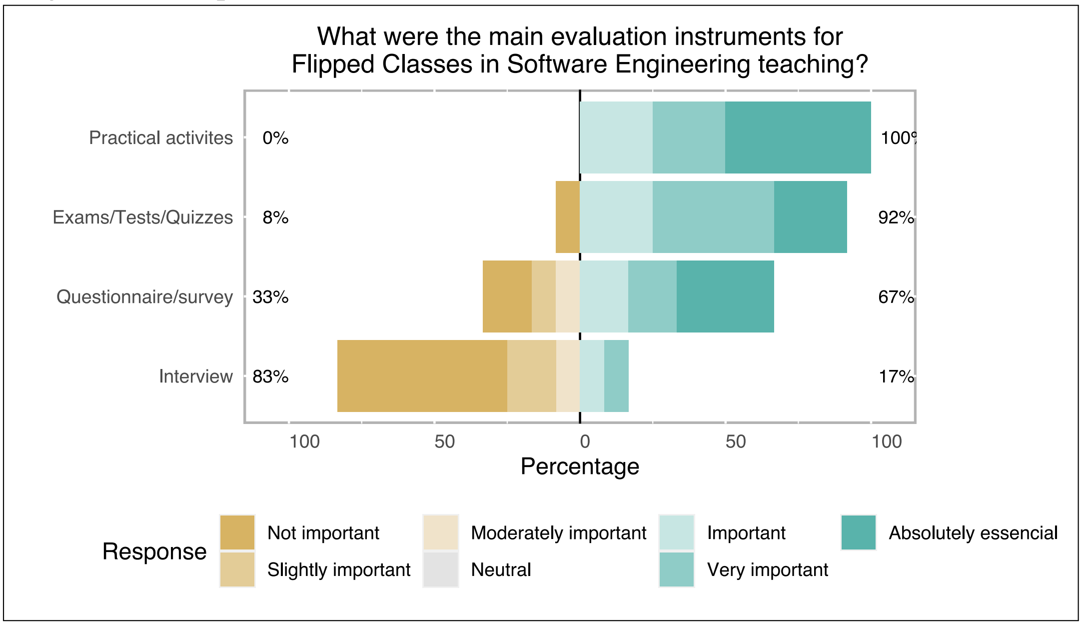
Figure 15 – Importance of evaluation aspects



Source: Author

chart with each item's positive and negative responses. This result is consistent with the most frequently used practical activities shown in Figure 14. All the researchers in the survey used some practical activity or performed some tests with their students. PBL-based, interactive groups and laboratory practices were the main hands-on activities. Researchers also used activities like hackathons, lego game activities, programming challenges, and case studies (software projects).

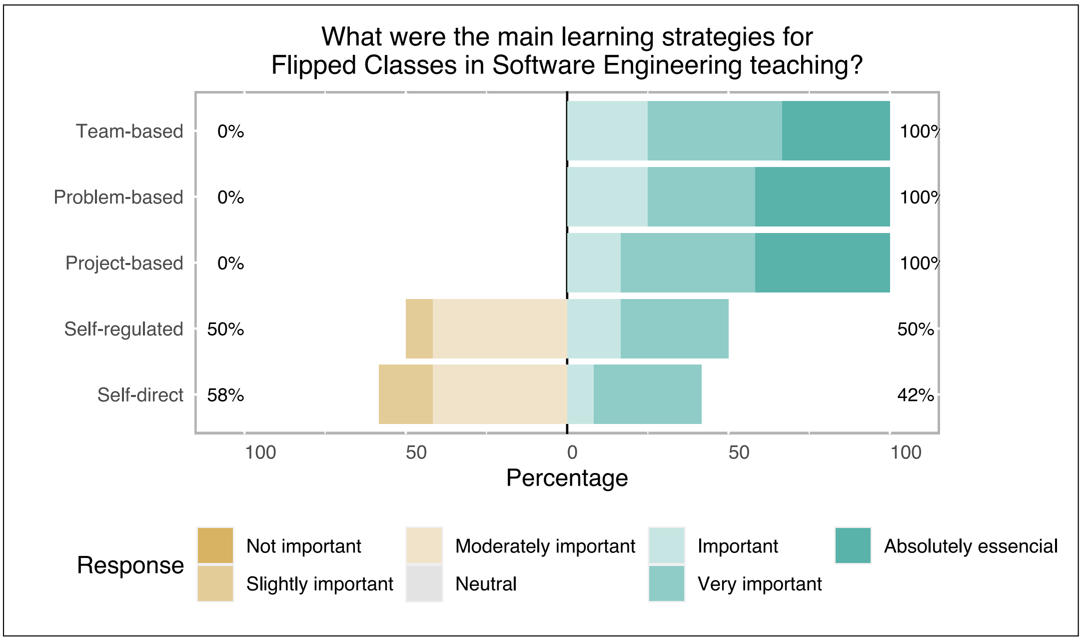
Figure 16 – Importance of evaluation instruments



Source: Author

Question fifteen (Q15) complemented the two previous questions (aspects and evaluation instruments) and asked about the importance of learning strategies for flipped classes in software engineering education. All respondents highlighted the high importance of team, problem, and project-based strategies for SE flipped classes. Figure 17 presents the percentage of the responses obtained.

Figure 17 – Importance of learning strategies

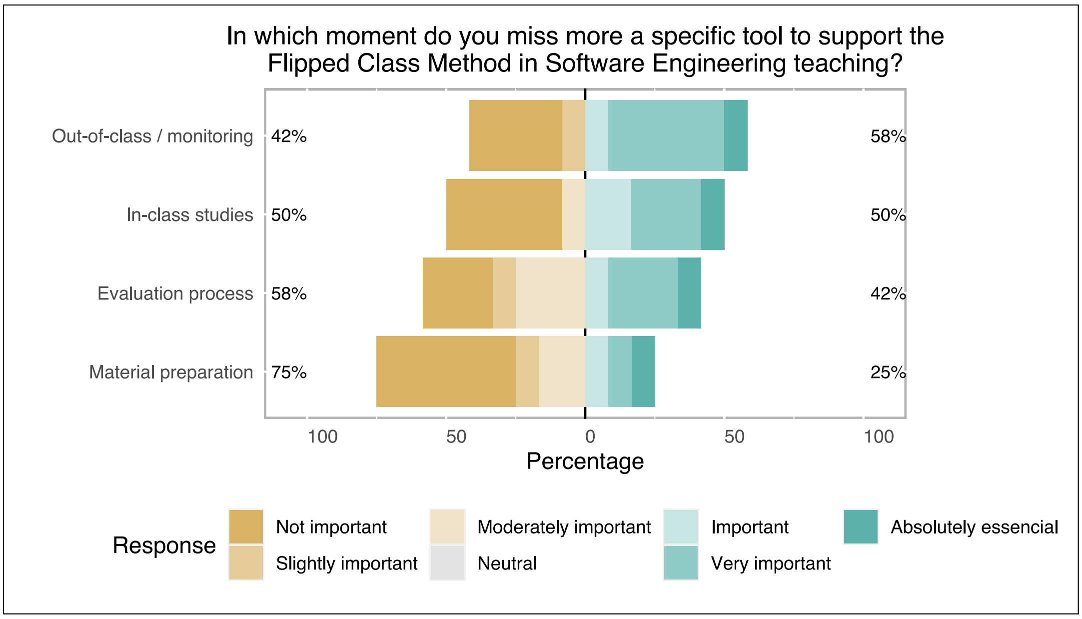


Source: Author

Question sixteen (Q16) aims to identify the importance of a new tool that supports the flipped classroom. The new tool aims to help teachers monitor student performance and provide feedback during out-of-class studies. The question asked the respondents to rate the importance of using the tool in different moments of a flipped classroom, such as during out-of-class studies, in-class studies, preparation of instructional material, and evaluation process. According to the survey responses, 58% of the respondents considered receiving student feedback during out-of-class studies the most critical moment to use the flipped classroom support tool.

Interestingly, using the tool during practical lessons, they received equal perceptions about the importance level, with 50% of the respondents rating it as essential and the other 50% rating it as unimportant. On the other hand, 75% of the respondents did not rate the use of the tool during the preparation of instructional material as necessary. Similarly, 58% of the respondents also rated it unimportant for the evaluation process. These findings suggest that the flipped classroom support tool can be most beneficial during out-of-class studies, where students work independently and may need guidance or feedback. However, its use during in-class studies, preparation of instructional material, and evaluation process may be less crucial. Figure 18 presents a graphical representation of the collected answers.

Figure 18 – Importance of using a specific tool in a flipped classroom moment



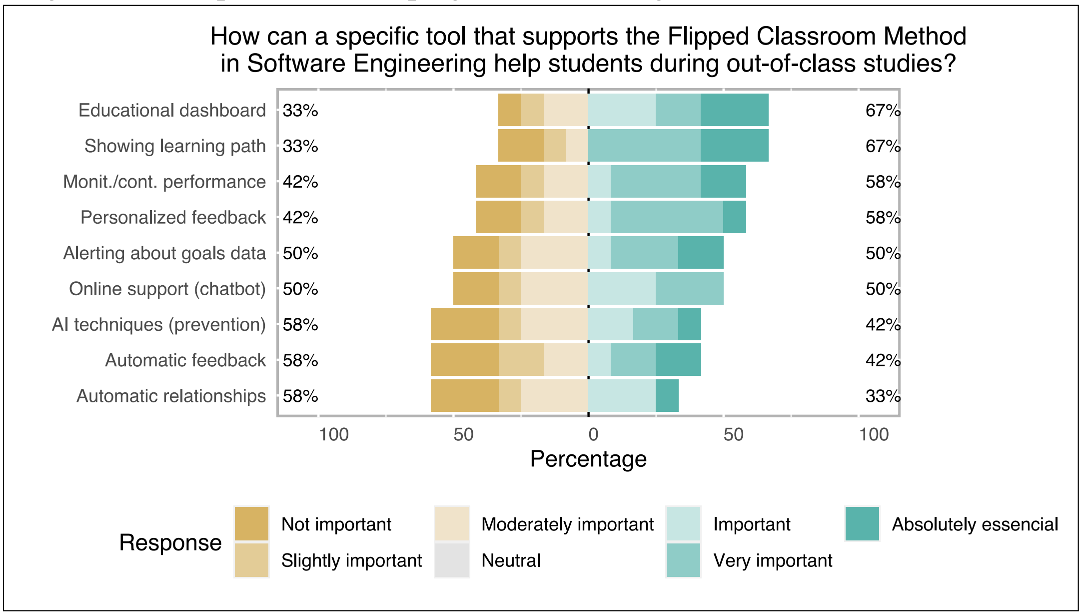
Source: Author

The researchers behind a study on flipped classes in software engineering teaching were asked in question seventeen (Q17) how a specific tool that supports this approach can help students during their studies outside the classroom. According to the majority of the researchers (67%), the most important benefits that the tool can offer to students are showing an educational dashboard and providing learning paths. These features were seen as aiding students in their independent studies.

Additionally, 58% of the respondents highlighted the importance of monitoring and controlling student performance data and providing personalized feedback to students. Half of the researchers answered that using a chatbot to provide online support to students is important. However, it is worth noting that 58% of the researchers do not believe that it is crucial to use A.I. techniques for underperformance prevention, provide automatic student feedback, or make automatic pedagogical relationships among students.

These findings provide insight into the opinions of experts in software engineering teaching on the use of flipped classes and the potential benefits of specific tools in supporting students' learning outside the classroom. The results of this inquiry are graphically presented in Figure 19.

Figure 19 – Importance of helping students during their out-of-class studies



Source: Author

3.2 Design Principles for Flipped Classroom Tools

Kim *et al.* (2014) proposed nine design principles for the flipped classroom. The first three were based on Brame (2022): (1) Provide an opportunity for students to gain first exposure prior to class; (2) Provide an incentive for students to prepare for class; and (3) Provide a mechanism to assess student understanding. The other six principles were suggested to create flipped events to better foster student-centered learning. Figure 20 shows the nine principles described in the paper.

Figure 20 – Nine flipped classroom design principles

Student-centered Learning	**Teaching Presence**	• Provide an incentive for students to prepare for class • Provide a mechanism to assess student understanding • Provide prompt/adaptive feedback on individual or group works
	Learner Presence	• Provide enough time for students to carry out the assignments
	Social Presence	• Provide facilitation for building a learning community • Provide technologies familiar and easy to access
	Cognitive Presence	• Provide an opportunity for students to gain first exposure prior to class • Provide clear connections between in-class and out-of-class activities • Provide clearly defined and well-structured guidance

Source: Adapted from Kim *et al.* (2014)

The design principles of flipped classrooms above guide this thesis for building a new tool to help FC preparation (Flipped Classroom Tool (FCTool)). Mainly following principles:

(a) provide a mechanism to assess student understanding, (b) provide adaptive feedback to students, (c) provide learning community, (d) provide familiar and easy-to-access technology, (e) provide clearly defined and well-structured guidance. We combined the results from the reviewed studies in Chapter 2 with deepening individual responses collected from the survey to align the mentioned principles with new design principles for Flipped Classroom Tools.

Initially, when analyzing the survey results, it was observed that the respondents faced **challenges** and **difficulties** while using FC. These observations helped formulate eight design principles explicitly derived from the results of SRQ6 of the systematic review (Chapter 2) and questions Q1, Q2, Q6, and Q16 of the survey (this Chapter). The eight design principles have been carefully crafted to address users' challenges while using FC (teachers and students). The following is the list of principles that have been identified:

Design principles for challenges and difficulties in FC preparing

P1: The tool must be used during activities inside and outside the classroom

P2: The teacher should reuse a study guide in another flipped class

P3: The tool must facilitate the flipped classroom preparation using a visual interface

P4: The tool should motivate the students

P5: The tool must allow the professor to monitor and evaluate the activities performed by the students

P6: The tool must collect feedback from students

P7: A tool should allow scale for big classes

P8: A tool must have a responsive interface

In the systematic review conducted in Chapter 2, we analyzed the results of SRQ6 and the survey responses to questions Q3, Q4, and Q17. This allowed us to understand the **advantages** and **drawbacks** of employing FC during SE teaching. Building on these insights, we have formulated eight additional design principles. These principles have been curated and are listed as follows:

Design principles for advantages and drawbacks in FC preparing

P9: The tool must help in monitoring the student's progress concerning their out-of-class studies

P10: The tool should help students' engagement and confidence

P11: The tool should support collaborative learning and group studies

P12: The tool should allow instructor-student and student-student interactions

P13: The tool must deliver content asynchronously

P14: The tool should support the students so that they finish their studies before the practical lesson

P15: The tool should show students a dashboard and their learning path

P16: The tool should remind students of their learning objectives

The insights about the tool **resources** and **evaluation** in preparing for the flipped classroom were drawn from survey questions Q5, Q7, Q8, Q9, Q10, Q11, Q12, Q13, Q14 and Q15. Some items were already included in the previous principles and were therefore not repeated (e.g., the tool should help in student engagement). The result created the following design principles:

Resources and evaluations design principles for FC preparation

P17: The tool should be integrated with other tools (e.g., GitLab) or with other local or online resources (e.g., use of mind maps)

P18: The tool must work with some standard during the educational content development (e.g. LOM[a]/CC[b]/QTI[c])

P19: The tool should allow viewing multiple project states simultaneously and tracking each student for each activity during in-class and out-of-class activities

P20: The tool must support specific SE educational resources, lecture videos (from Youtube), slides, and Texts/PDFs

[a] Learning Object Metadata
[b] The Open Educational Program at Creative Commons
[c] The Question & Test Interoperability Standard

3.3 Design Principles for Personalized Study Guides

Personalized learning means tailoring learning to each learner's interests, strengths, and needs. This approach encourages flexibility to support mastery and allows learners to influence how, what, when, and where they learn (PATRICK *et al.*, 2013).

Personalization of learning refers to an educational institution's effort to tailor its program (or course) to their students. Personalization can be accomplished differently, limited only by human, institutional, and instructional resources. There is no single or most appropriate way to personalize learning. It is an attempt to balance student characteristics with those of the learning environment. It is a learning process in which students follow a path to achieve their own goals, work cooperatively on assignments, keep track of their progress, and demonstrate their learning, all with the close support of instructors. Personalization requires an interactive learning environment, physical or virtual, designed to promote collaboration and reflection among students (KEEFE, 2007).

One way to materialize personalized learning is through study guides. Study guides are a powerful tool to help students manage their learning, especially when the information volume is overloaded. In a student-centered view, the teacher acts as a facilitator (and curator) and the student with independence. A study guide can be a helpful tool for supporting this approach (KHOGALI *et al.*, 2006). Using a study guide, students can access a varied online learning resources and flexibly choose their learning paths. Students can follow different learning paths depending on their understanding of specific content. The choice of learning path can be (a) free to a student (depending on your feeling), (b) manually suggested by the instructor according to the student, or (c) even automatic (made by a system). An innovative differential for the learning path construction is its adaptation according to the student's performance (or profile) (CARCHIOLO *et al.*, 2002).

However, without a technological tool is unlikely imagine a personalized learning by adaptive learning path using a customized study guide. Tools in personalized, blended, and online learning must support flexible pacing, differentiated instruction, immediate interventions, and anywhere, anytime learning (PATRICK *et al.*, 2013). In this sense, educational technologies can make personalized learning practical, reduce the time required to set up the learning path (and a customized study guide), provide online access to learning content and alternative instructional materials, and automatically assess learning progress.

Creating a technology to promote personalized learning requires the adoption of some previously defined design principles. In addition, several universal principles can be extracted from relevant models and theories to guide a new personalized learning design. In Watson and Watson (2016), five universal principles for personalized instruction are presented. The five principles are:

Design principles for Personalized Study Guides

P21: Personalized instructional goals: Students should set and periodically review their learning goals with appropriate amounts of guidance through interaction with their instructor. The instructor should help students identify strengths and interests they were unaware. Detailed student progress logs on personal learning paths should be kept

P22: Personalized task environment: The instructional material (or learning path) selection should be personalized according to the learner's interests, goals, and prior knowledge. The tool should offer subsets of potential alternatives for the learner to choose from. The collaboration level on tasks should be personalized, negotiating which tasks are best done individually and which in groups of learners

P23: Personalized scaffolding of instruction: The quantity and quality of instructional materials should be customized according to the student's self-regulation abilities and built needs through a combination of built-in scaffolding and teacher and peer scaffolding

P24: Personalized assessment of performance and learning: Performance indicators and task achievements index should be personalized, chosen by the instructor, external experts, peers, or automatically employing computer systems. In addition, the way for performance evaluation and task achievement should be personalized to align with the student's goals and interests through the performed activities

P25: Personalized reflection: When and how a student should reflect on their learning process should be personalized. Personalization must occur concerning when and how the student reflects on how their product or performance did not meet the expected results

3.4 Chapter summary

This chapter aimed to deepen the flipped classroom process in software engineering education. The goal was to discover the characteristics an authoring tool should have to help SE teachers prepare flipped classrooms. In order to achieve this objective, we surveyed software engineering professors with experience in implementing flipped classrooms. A total of 96 emails were sent out, and 12 responses were received, accounting for a response rate of 14.63%. These responses provided data regarding the challenges, difficulties, advantages, drawbacks, resources, and evaluation associated with the flipped classroom approach.

The chapter then discussed the insights and design principles developed for constructing tools to support software engineering flipped classes. One of the key tool innovations that were identified is the ability to generate personalized study guides, which can be positively effective in enhancing student engagement and learning outcomes. The survey was carried out in four distinct stages, developing twenty-five design principles for creating such tools.

The design principles were classified into three categories that are closely related to the preparation of flipped classrooms: (1) challenges and difficulties, (2) advantages and drawbacks, and (3) resources and evaluation. The design principles identified and discussed in this chapter were instrumental in conceptualizing a model for the personalized study guide-based flipped classroom, which will be described in detail in the subsequent chapter.

4 A PERSONALIZED STUDY GUIDE MODEL

This chapter introduces our approach for integration adaptive learning in SE flipped classes. Our goal is to answer the third research question of our study (RQ3): *Is feasible to design, formalize, and implement a Personalized Study Guide Model to represent out-of-class material for SE flipped classes?*

Initially, Section 4.1 provides a broad overview of our perspective on implementing a flipped classroom in SE, and Section 4.2 defines personalized study guides. Following that, Section 4.3 presents our Personalized Study Guide model, and it is formally defined using Business Model and Notation (BPMN), offering a visual representation of its essential elements (instructional contents and personalization points). To illustrate the practical application of the model, Section 4.4 presents a detailed example of a flipped classroom on Software Requirements. This example describes the learning objectives, study guide, practical activities, and materials. Additionally, we performed a computer-based simulation to generate simulated data demonstrating how students explore the personalized learning paths the model offers. These simulated data provided insights into the functioning of the model and its potential benefits for both teachers and students. Finally, in Section 4.5, the chapter is concluded, summarizing the key findings and highlighting the importance of the model in the context of the flipped classroom and personalized education.

4.1 Flipped Classroom and Adaptive Learning

The flipped classroom approach is a teaching method that emphasizes student-centered learning, where students are expected to prepare for a class in advance by accessing online learning resources, such as videos and reading materials, to understand the concepts beforehand. During the in-class time, students are given opportunities to engage in problem-solving and other collaborative learning activities, which require them to apply the concepts they learned online (BERGMANN; SAMS, 2012).

On the other hand, adaptive learning is a teaching approach that seeks to personalize the learning experience for each student. This adaptability means that the content, pace, and preferences (kind of material) are tailored to meet the specific needs of each student based on their prior knowledge, performance, and individual learning goals. Adaptive learning aims to create an individualized and optimized learning experience that promotes effective learning outcomes and student engagement (ARAÚJO *et al.*, 2020; CEVIKBAS; KAISER, 2022).

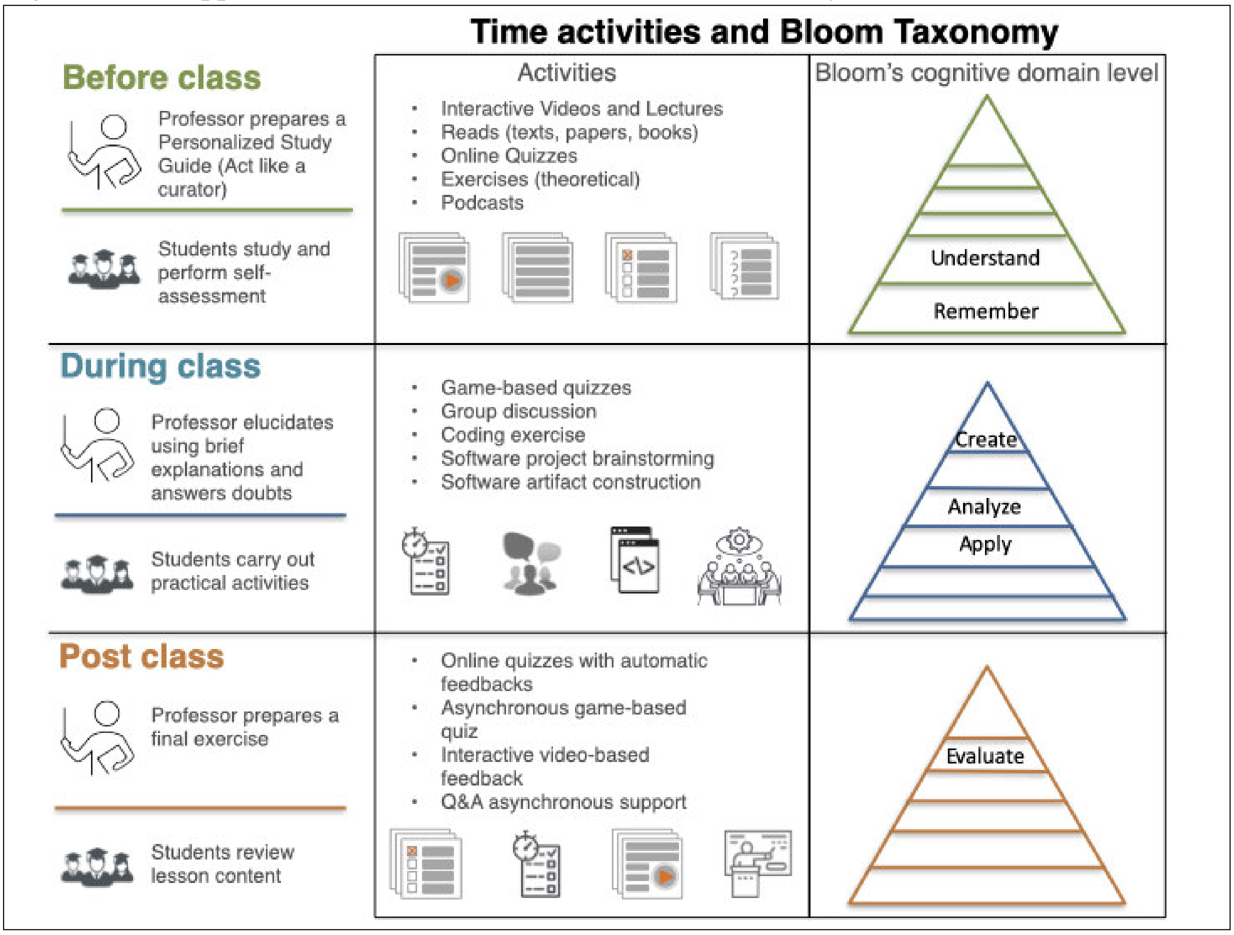
Our approach is an extension of the flipped classroom method. Our proposal integrates adaptive learning techniques by adapting study guides, i.e., out-of-class materials, according to the student's performance, profile, or prior knowledge. We designed a personalized study guide model as the core concept for flipped classes in Software Engineering teaching. The study guide is a set of contents with rules and directed flows that allow the creation of multiple student learning paths.

We integrated adaptive learning into SE flipped classes in three steps: (1) the first step happens before class using personalized study guides, (2) the second step occurs in class, and (3) the third step occurs post-class. Before class, the instructor prepares a personalized study guide (e.g., a list of videos, research papers, book chapters, and gray literature suggestions) for students studying at home. During the in-class moment, the instructor briefly explained the class content and could dedicate time to answering FAQs brought by students. In the remaining class time, the instructor conducts practical activities (like quizzes, software games, project planning, requisite meets, programming tasks, and brainstorms). Finally, students receive work to help them review content after class.

Figure 21 shows our flipped classroom steps and relate them to the Bloom's taxonomy levels. This taxonomy is a well-known framework for educational objectives. It consists of six levels that classify and prioritize learning objectives from lower to higher-order thinking skills. These levels are knowledge, comprehension, application, analysis, synthesis, and evaluation. In a flipped classroom, each step of the flipped class is associated with some learning objectives of the pyramid (KRATHWOHL, 2002). The relationship between the elements of the flipped classroom and each level of the cognitive domain is presented in Figure 21. This relationship helps ensure that the learning objectives align with the cognitive skills the students need to develop.

In our approach, the flipped class begins with the preparation (curation) of the personalized study guide made by the professor/instructor. Study guides sent to students consisted of many materials, such as videos, content-related texts, and assessment quizzes (at the end of the study guide). The study guide should offer relevant content to help students to memorize basic knowledge in their long-term memory (Bloom's cognitive process of remembering). Furthermore, the study guide will assist students in finding the meaning of the instructional material, including interpreting concepts, summarizing contents, and explaining processes (Understanding level). The structure of our study guides draws inspiration from the study proposed by Maher *et al.*

Figure 21 – Flipped Classroom activities and Bloom's Taxonomy



Source: Author

(2015).

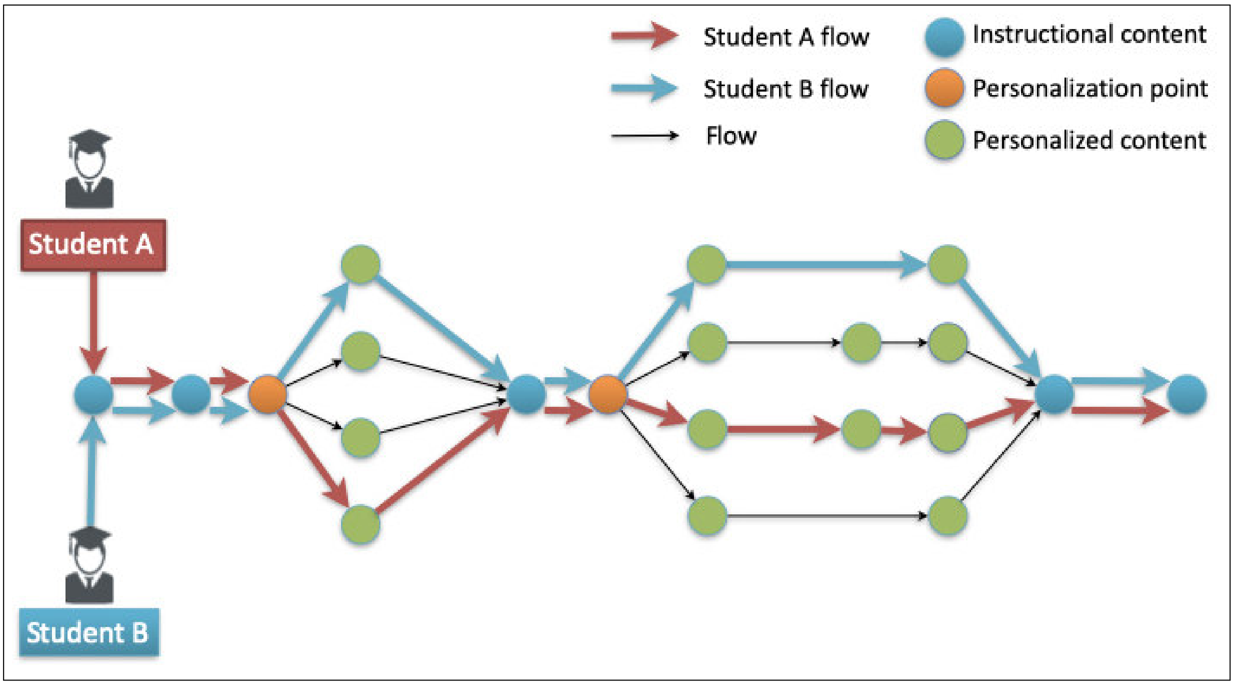
Subsequently, upon receiving the study guides, students are expected to adhere to the guidelines for studying the contents and completing the assessment quizzes outlined in the study guide. During class, the instructor primarily elucidates using brief explanations and focuses on answering students' doubts about the studied content. Finally, students are involved in practical activities related to the content under study. The activities happen quickly, allowing students to complete and deliver their answers in the same class. Students could engage in game-based quizzes, project planning, requirements discussions, software project brainstorms, and programming activities. The students carry out practical activities (Applying level from Bloom's Taxonomy), organize (with groups) different software artifacts to relate the links between the parts (Analyzing level), and then make software artifacts construction (Creating level). After the practical class, a review lesson is prepared and sent to students to make judgments based on criteria and standards (Evaluating level).

4.2 Personalized study guides

Our concept of a personalized study guide is to structure the study guide in multiple ways and paths so that a student can follow the material. Instructional contents may or may not be included in the student's learning path, as well as in the order in which the materials are presented. For that, instructors thoughtfully craft each study guide, mapping the content flow and creating various potential student learning paths. The professor is responsible for designing all possible paths of a study guide, ensuring that each student's journey is unique and optimized for success.

In that way, personalized study guides offer multiple structures and navigation paths through the instructional materials tailored to each student's specific needs and preferences. As students progress through their personalized path, their experience could be tailored to their performance, preferences, and engagement. The approach considers the student's pace, level of understanding, and preferences about kind of material to guide the next steps. The method may recommend repeating a section or moving on to the next one based on their performance. We aim to enable students to learn independently and achieve better outcomes. Figure 22 illustrates an example of two students' paths in the same study guide. Each student has a unique path that reflects their learning journey. Adaptive learning paths offer a flexible and personalized approach to learning that can help students achieve their academic goals.

Figure 22 – Personalized study guide example



Source: Author

Each blue circle in Figure 22 represents the instructional content of the study guide, and each orange circle represents a personalization point - a programmable rule (conditional decision, conjunction, disjunction, free student choice condition). Each green circle represents personalized content after the action of a personalization point. The personalization point allows the teacher to add and program elements for adaptive learning. In the conditional decision element (*if/else*), the teacher sets a condition to evaluate the student's performance/engagement during studies. An example rule: *if* the student gets a score *Greater than* 7 in a form *then* will follow the *Upper Path*, *else* will follow the *Lower Path*. The *conjunction* element is similar. However, the teacher programs have two conditions. An example rule for the *conjunction*: *if* the student viewed *Equal to* 80% of the lesson video *AND* the student got a score *Greater than* 7 *then* the path will be released. The rules for the *disjunction* element resemble those of the *conjunction* element, but only one of the conditions must be satisfied to release the path. The *optional* element has a single condition to release the path. The teacher can use this element to, for example, challenge the students. Finally, the *Student Decision* element allows the student to choose one of two possible paths. The teacher should explain/comment on the paths.

According to our approach, professors will create instructional materials, including videos, research papers, exercises, and external resources. These materials will be designed to conform to guidelines and regulations implemented throughout the personalized study guide. Using adaptive rules, teachers can tailor learning paths based on a student's prior knowledge, performance in exercises and tests, or self-directed choices. This adaptation means each student will receive a personalized learning experience tailored to their needs and abilities.

Furthermore, students will have the freedom to chart their path based on their learning objectives. This autonomy will empower students to take control of their education and guide their learning path. Students can choose their topics, learning pace, and resources.

4.3 A BPMN-based model

We formalized our study guide model **adapting** the Business Process Model and Notation (BPMN model) semantics (ROSING *et al.*, 2015). Our personalized study guide model contains elements to represent student model, instructional content, personalization points, and the content flow. We modeled our study guide as a BPMN diagram, and for this, we mapped the elements of our model with existing structures and concepts in the BPMN model.

4.3.1 Model elements

A **student model** combines characteristics with many attributes, such as learners' experience, education level, knowledge, skill, abilities, grades, interests, and preferences (FIQRI; NURJANAH, 2017). The attributes used in the student model can be varied, depending on the types of adaptation that will perform. An ideal student model would include all the data about behavior and knowledge that affect learning and performance (KAVCIC, 2000). Our student model incorporates dynamic data about performance (scores), engagement, content preferences, and learning habits. Dynamic data are information obtained from the student's interaction with a system.

Our model contains four types of **instructional contents** (Reading, Multimedia, Score-based, and Text-based) and supports three levels of interaction (low, medium, and high). At the low level, the student has few options for interaction, e.g., pausing a video or clicking on links. Students can also participate in form-based and code-programming activities with medium interactivity. Additionally, the model supports high-interactivity content in which students can watch, react, and interact with other students (e.g., interactive video lessons, quizzes, and forums). The instructional content interacts with the student model by updating the database (data input). The four instructional contents of our model are as follows:

a) **Reading content**: In this type of instructional content, students must study by accessing reading materials, such as a PDF file or a link to a webpage. A **PDF** is flexible enough to materialize as a handout, book, paper, or slide presentation. The most important is to cover the main concepts, theories, and practical applications, giving learners a resource for study and reference. With its portable format, learners can easily access the PDF on multiple devices, allowing flexibility in learning environments. In addition, learners can take personal notes and annotations as they engage with the content.

   By **link**, students can access a web page or a source of additional and complementary information to the study material. A web page can contain papers, videos, tutorials, practical examples, and other relevant resources related to the topic under study. This resource allows students to access multimedia materials from sources curated by the instructor.

b) **Multimedia content**: Multimedia content refers to the combination of different forms of media to present information, such as video lessons, podcasts, or music. In a **video lesson**, for example, students can watch an instructional video with relevant concepts and topics about the flipped classroom theme. In addition to providing theoretical knowledge, the video lesson can encourage active student participation by incorporating interactive moments, e.g., through questions, quizzes, and challenges incorporated into the video lesson, students can test their understanding and reinforce the concepts presented.

Another example is an **HTML5** resource. It allows students to explore interactive and dynamic elements, such as games, simulations, animations, and exercises. Students can interact directly with these resources through buttons, drag-and-drop, form-filling, and other custom interactions. Students can also receive immediate feedback during their interactions, allowing them to assess their progress and identify areas that need more attention.

c) **Score-based content**: Refers to instructional content that can be measured through a score, i.e., the student's interaction with the content will be assessed, allowing a value (importance) to be assigned to the student. Examples of score-based instructional content include: (1) An electronic **form** can promote active learner participation through questions designed by the instructor. We encourage students to provide their answers and solutions directly on the form. This approach stimulates reflection and self-assessment, as the form can automatically provide a performance score as feedback. Additionally, the form can offer immediate subjective feedback, enabling learners to assess their understanding and identify areas that require further study; (2) A **code programming** task allows students to develop practical programming skills, enabling them to create and manipulate algorithms and computational solutions. Through hands-on (or fundamental) exercises, projects, and challenges, students can apply their knowledge to solve real-world problems and build functional programs.

d) **Text-based content**: In this type of content, students will provide responses (or comments) based on texts they produce. For example, they participated in forums or wrote a short answer to a direct question. In a **short answer** production, instructors ask students to provide concise and direct answers to specific questions. This activity aims to develop synthesis skills, clarity in communication, and the ability to convey information objectively. Students are encouraged to demonstrate their understanding of the topic by responding accurately and coherently.

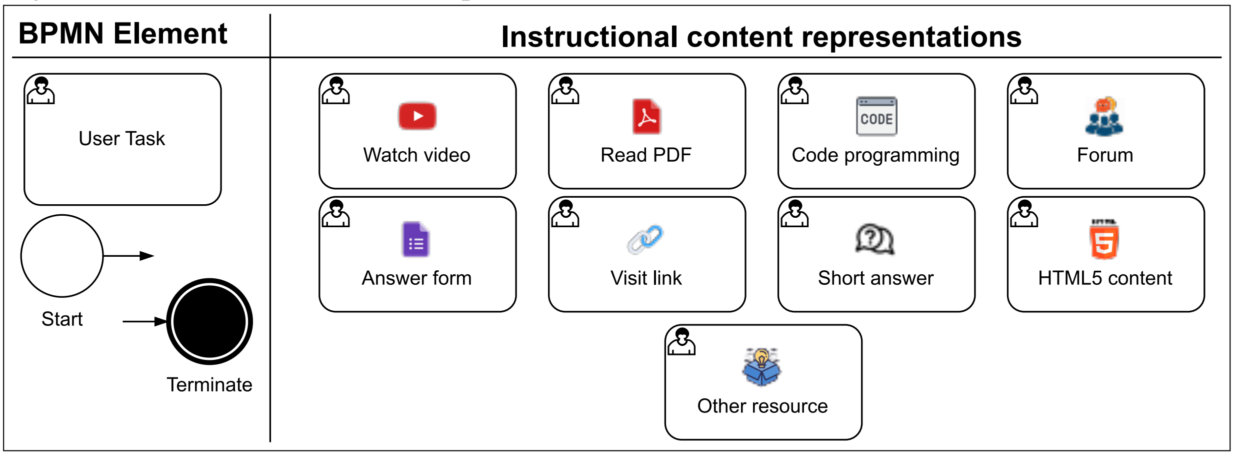
In a **forum** participation, students are encouraged to share ideas, exchange opinions, and discuss relevant topics related to the flipped classroom theme. Students can interact with each other and the instructor, contributing individual perspectives to enrich collective learning. The main idea is that students can collaborate with other classmates' responses, ask questions, provide insights, or share additional resources related to the topic under study. In addition, students can receive feedback and guidance from the instructor during forum interactions, enhancing their understanding of the subject matter and developing written communication skills.

We have also included an **other resource** into our model. This instructional resource serves as an indication of the potential for expanding the existing model. The model considers the possibility of incorporating a new instructional resource not initially included in the content model for study guides.

Content flow indicates the dependency and sequencing between two contents (or personalization points). In other words, the flow from content A to B shows that B can only be visited after A is visited. By leveraging the power of the BPMN model and its semantics, we can reuse the flow and relationships between different types of content, allowing for a comprehensive overview of the study guide structure. Our model can have multiple input or output flows without needing a gateway.

We used User Task representation as the base element of the BPMN model to represent our instructional content. A User Task in the BPMN model is a workflow task in which a person (user) performs a task with software support/assistance. In our model, we represent the instructional contents as a specific User Task with an assigned icon to differentiate the contents and their unique characteristics. We have mapped examples from each of the four types of instructional content to BPMN model elements to create visual representations, as graphically shown in Figure 23. Table 10 summarizes each instructional content, organizes the interaction levels, and presents their attributes.

Figure 23 – Instructional content representations



Source: Author

Table 10 – Instructional content details

Content	Interaction	Description	Attributes
PDF file	Low	Link to a PDF file configured by the teacher/instructor	{Title, Link, Status}
Link	Low	Link to any web resource curated by the teacher/instructor	{Title, Link, Status}
Video lesson	Low	Link to (Youtube) video lesson chosen by the teacher/instructor	{Title, Link, Percentage watched}
HTML5 content	High	Rich, interactive web content	{Title, Description, HTML5 File, Score}
Form	Medium	Link to an electronic form made by the teacher/instructor	{Title, Link, Score}
Code programming	Medium	Link to a code repository with automated testing	{Title, Link, Score}
Short answer	Medium	The student provides a short, subjective answer to a question asked by the instructor	{Title, Ask, Answer, Status}
Forum	High	Students interact on a subject/theme proposed by the teacher	{Title, Subject, Description, Answers, Status}

Source: Author

The *Percentage watched* attribute in video lesson indicates the extent to which a student has watched the lesson included in the study guide. The *Status* attribute signifies whether a student has confirmed reading a PDF file or has visited the web resource associated with the provided link. The *Status* also indicates whether the student has interacted in a forum or has sent a short answer. The *Score* is a value assigned to a student based on their responses to the electronic form in the Form content. In code programming indicates the student's performance on the test cases, and in HTML5 content, represents the student's performance.

Figure 24 exemplifies the sequential flows between the instructional contents of a study guide. The aim is to illustrate a simple study guide highlighting the interaction between content without personalization points. A student's interaction and use of instructional content create dynamic data that must update the student's database. The contents "Video lesson", "Form", "Code programming", "Short answer", "Forum", and "HTML5" provide data on performance, while "PDF file" and "Link" provide data on engagement. All content should provide data on learning habits via logs, i.e., a log should be recorded in "Video lesson" when a student pauses a video.

Figure 24 – Simple BPMN-based study guide example

Source: Author

Our model also has five **personalization** points, summarized in Table 11.

Table 11 – Personalization point details

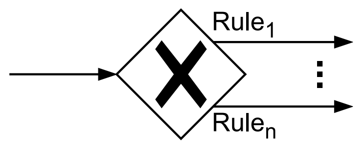
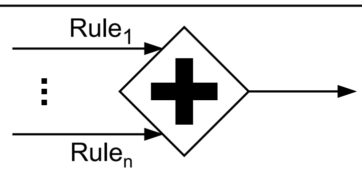
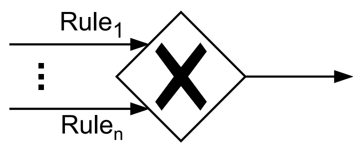
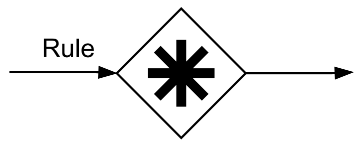
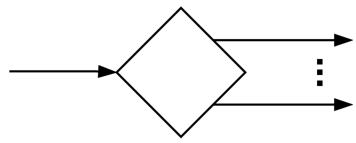
Personalization point	**Description**	**Attributes**
Conditional Decision	A set of rules programmed by the teacher in this point allows the student's learning path to be adjusted based on his performance/engagement	$\{Rule_1, Rule_2, ..., Rule_n\}$
Conjunction	If the rules are evaluated positively, this point enables the merging of multiple paths	$\{Rule_1, Rule_2, ..., Rule_n\}$
Disjunction	If at least one rule is evaluated positively, this point enables the merging of multiple paths	$\{Rule_1, Rule_2, ..., Rule_n\}$
Optional	This point allows a new path when the rule is evaluated positively	$\{Rule\}$
Student Decision	The student decides which path to follow	$\{Decision\}$

Source: Author

We used the Gateway representation as the fundamental element of the BPMN model to represent our personalization points. Gateways serve the purpose of controlling the convergence and divergence of flows within a process (ROSING *et al.*, 2015). In the context of our model, we employed gateways to illustrate how flows can converge and diverge within the learning path of a study guide.

Overall, our BPMN-based model with gateways empowers instructors to design and implement personalized learning experiences while providing learners the autonomy and flexibility to navigate their unique educational journeys. Each of the five types of personalization points was assigned a specific representation. Figure 25 provides a comprehensive visualization of the symbols used in our model, along with the mapped flows derived from the BPMN representation.

Figure 25 – Personalization points representation

BPMN Element	Symbol	Personalization point
Exclusive Gateway (Splitting)		Conditional Decision (n outputs)
Parallel Gateway (Merging)		Conjunction (only one output)
Exclusive Gateway (Merging)		Disjunction (only one output)
Complex Gateway (only one input)		Optional (only one output)
Gateway (Splitting)		Student Decision (n outputs)

Source: Author

Rules are strategically modeled at personalization points, encompassing Conditional Decisions, Conjunctions, Disjunctions, and Optionals, to facilitate the progression of students along their designated learning paths. The rule formulation process entails the creation of two distinct categories: quantitative and qualitative. The selection between these categories hinges on the nature of the input associated with the personalization point. A quantitative rule is characterized by its integration of quantitative attributes such as “Percentage Watched” or “Score”. Conversely, a qualitative rule operates through the utilization of the “Status” attribute. This rule-based framework not only orchestrates the release of students’ learning pathways but also offers a versatile mechanism for tailoring the educational experience to the specific needs and performance levels of individual learners. The quantitative rule structure is as follows:

Quantitative rule structure

IF (**Quantitative attribute*** <*logical condition*> [**AND/OR** <*logical condition*>]**) *Then* Follow Path, where the <*logical condition*> can be:

a) Greater than <value>
b) Less than <value>
c) Equal to <value>
d) Greater than or equal to <value>
e) Less than or equal to <value>

* "Percentage Watched" or "Score" attribute

** [] is an optional statement

The qualitative rule structure is as follows:

Qualitative rule structure

IF (**Qualitative attribute*** is <*status value*>) [**AND/OR** <*status value*>]**) *Then* Follow Path, where the <*status value*> can be:

a) Content marked as read (for PDF or link),
b) Content marked as skipped (for PDF or link),
c) Short answer sent,
d) Short answer not sent,
e) Forum participation,
f) No forum participation

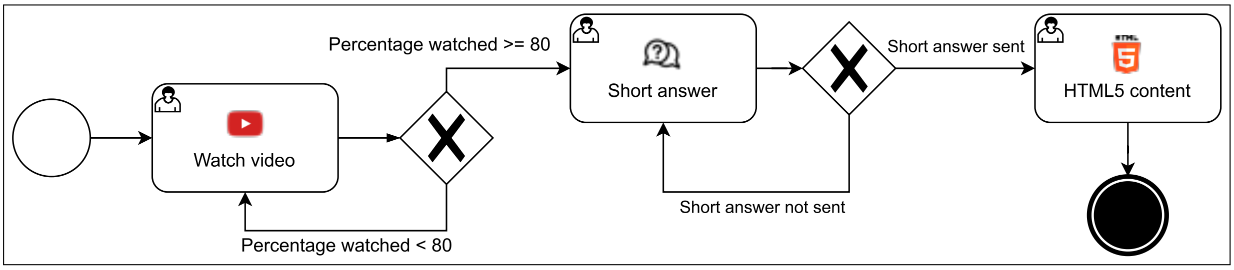
* Status attribute

** [] is an optional statement

The student's path is built dynamically by evaluating the rules defined by the teacher/instructor. Each of the five personalization points has its particular way of working, and it is up to the instructor to design the appropriate personalization rules according to the objectives of the study guide.

Conditional Decision Point (CDP). The system, in accordance with the defined rules, generates personalized outputs, affording the instructor the flexibility to modify the learning trajectory based on the individual student's data. A case in point is depicted in Figure 26, which showcases the CDP utilization involving distinct input categories - quantitative and qualitative. In this exemplification, access to the "Short Answer" content is granted solely when the percentage of video watched surpasses or equals 80, as stipulated by the quantitative rule. Specifically, this entails that the student is eligible to proceed to the "Short Answer" section solely after having consumed a minimum of 80% of the Video Lesson, thereby reflecting a quantitative threshold for content engagement.

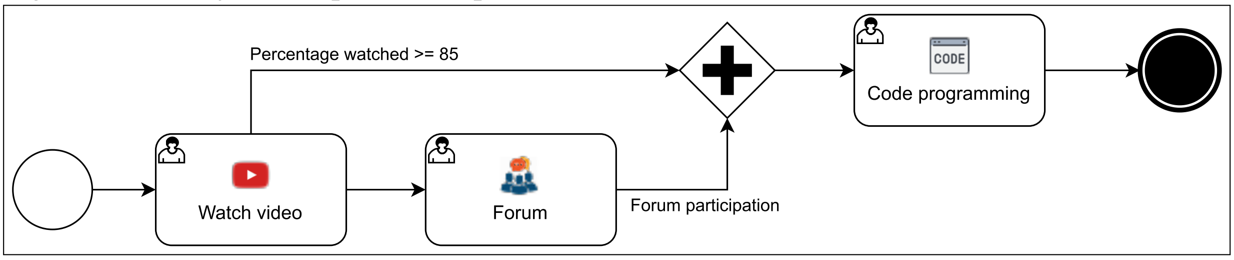
Figure 26 – Conditional decision point example



Source: Author

The **Conjuntion Point (CP)** serves as an evaluative AND clause, appraising all inputs prior to authorizing the pathway for student progression. Within this point, the regulatory constructs established by the instructor center their focus on inputs contiguous to the CP. This is exemplified in Figure 27, which provides an illustrative scenario featuring a Conjunction Point. This example encapsulates a pairing of distinct rules - a rule of quantitative nature alongside its qualitative counterpart. Specifically, the illustration underscores that a student is mandated to undertake the "Code Programming" task exclusively upon the fulfillment of two conditional prerequisites: first, the attainment of a minimum threshold of 85% in video lesson viewership; second, active involvement in the forum through substantive contributions, connoting the imperative to submit responses therein. The Conjunction Point thus orchestrates a synchronized alignment of multiple criteria, effectuating a nuanced nexus of conditions requisite for the unlocking of designated learning pathways.

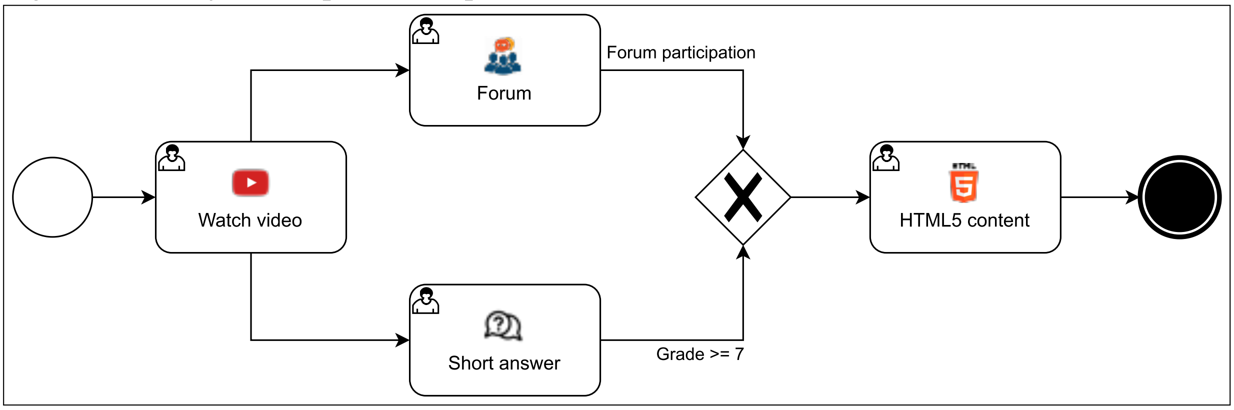
Figure 27 – Conjunction point example



Source: Author

Operating as an evaluative OR clause, the **Disjunction Point (DP)** orchestrates an assessment of inputs, sanctioning the pathway for student advancement. The trajectory gains its release upon the verification of a minimum of one positively affirmed logical condition. Figure 28 shows a Disjunction Point example featuring a dynamic interplay of two inputs. This illustration offers insight into the nuanced mechanics: a student's engagement with the "HTML5 Content" activity ensues solely upon the satisfaction of one of two distinct criteria. Either active participation in the forum, underpinned by a qualitative rule, or the attainment of a short answer score equating to or exceeding 7, anchored in a quantitative rule, stands as the prerequisite for accessing the specified learning content.

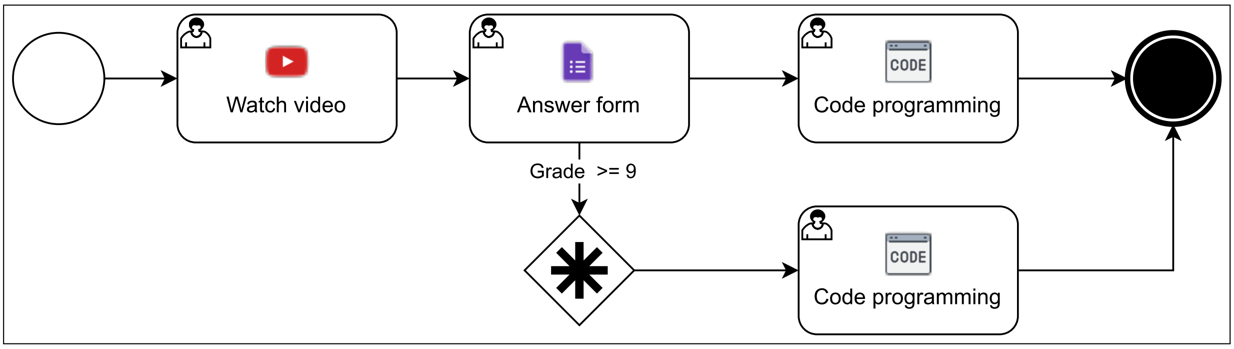
Figure 28 – Disjunction point example



Source: Author

Optional Point (OP) allows the teacher/instructor to create an alternative learning path. The instructor can propose challenges to the students or offer new learning paths. OP blocks the path forward until the programmed rule is positively evaluated. Figure 29 shows the optional point example. In the example, the student with a score greater than or equal to 9 on the form will be challenged to do a second "Code programming".

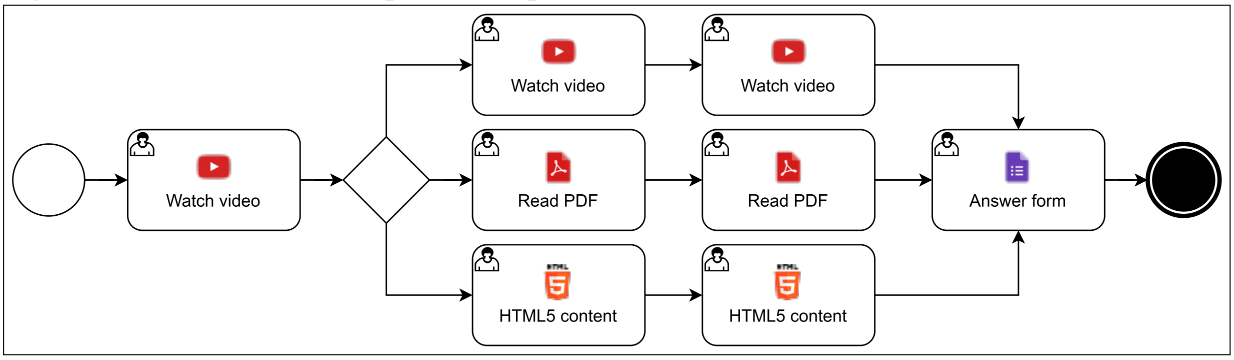
Figure 29 – Optional point example



Source: Author

Student Decision Point (SDP) blocks the path ahead until the student decides which path to follow. No rules are necessary for SDP. First, the instructor creates alternative learning paths and explains each path (no programming is required). Next, the student chooses their preferred path based on the instructor's explanation; the chosen path is unlocked after the choice is made. Figure 30 shows the SDP example. In the figure example, students can study using video lessons, PDF files, or HTML5 content. Ultimately, everyone will answer the same exercise in the form, regardless of the path chosen.

Figure 30 – Student decision point example



Source: Author

4.3.2 *Measuring students' engagement and performance*

Using our model, we can track how engaged students were with the study guide and how well they perform on the topics included in the document. Engagement in this context is about how actively students participate in the study guide materials, such as reading, watching, and responding in forums. Quantifying engagement helps measure student involvement during previous studies in a flipped classroom (out-of-class studies), assisting the teacher in motivating students to progress in their studies outside the classroom. Engagement does not guarantee that learning will happen, but it can provide an opportunity to reflect on learning activities (MAXIM *et al.*, 2022). Performance is a metric for student summative assessment on a topic covered by the study guide. It provides a quantitative measure of the student's current knowledge on the topic studied, assessed by a ***score*** obtained through a theoretical examination (or exercise) on the subject (AMRESH *et al.*, 2013).

We measure student engagement by *Percentage watched* and *Status* attributes from instructional contents. The *Percentage watched* represents, in percentage (0 to 100), how much engagement the student has achieved in the video lesson content, while the *Status* registers by the student's self-declaration about having or not accomplished the activity requested in the content. In order to leave them on the same scale of values, the *Status* is equal to 100 if a student confirms accomplishing the content's task and 0 otherwise. Thus, the student's engagement in instructional content is an attribute value. So, to compute the student engagement (S_e) in a learning path *LP*, we sum the engagement values at each content belonging to the path to calculate the mean. Let t_e the total number of instructional contents with engagement attributes in *LP*, then we calculate S_e using the following equation:

$$S_e = \frac{\sum_{i=1}^{t_e} content_i}{t_e} \text{, where: } content_i \in LP \tag{4.1}$$

A study guide (*SG*) can consist of multiple learning paths with n contents. Thus, the student's engagement regarding the complete study guide (SG_e) is the average of all the engagement values of the instructional contents. Thus, let t_{SG} the total number of instructional contents with engagement value in the study guide, then the equation calculates SG_e:

$$SG_e = \frac{\sum_{i=1}^{t_{SG}} content_i}{t_{SG}} \text{, where: } content_i \in SG \text{ and } t_{SG} \leq n \tag{4.2}$$

Complementary, the average engagement of y students ($\overline{S}_e$) in a study guide (*SG*) is measured as follows:

$$\begin{gathered} \overline{S}_e = \frac{\sum_{i=1}^{y} S_e^i}{y} \text{, where:} \\ S_e^i \text{ is the i-th student engagement value and} \\ y \text{ is total number of students} \end{gathered} \tag{4.3}$$

Similarly, to calculate the student's average performance (S_p) in a learning path *LP*, we divide the sum of scores achievement by the number of activities/contents taken. Let t_p the total number of activities/contents taken, then we calculate the average performance of a student using the following equation:

$$S_p = \frac{\sum_{i=1}^{t_p} content_{i\{Score\}}}{t_p} \text{, where: } content_i \in LP \tag{4.4}$$

Finally, the average performance for a set with y students ($\overline{S}_p$) in a study guide (*SG*) is given by equation:

$$\begin{gathered} \overline{S}_p = \frac{\sum_{i=1}^{y} S_p^i}{y} \text{, where:} \\ S_p^i \text{ is the i-th student performance and} \\ y \text{ is total number of students} \end{gathered} \tag{4.5}$$

4.4 Proof of Concept

This section presents two examples of study guides as proof of concepts from our model. The first is an example without content personalization. The second is an example of a flipped lesson on Software Requirements.

4.4.1 Toy example

First, we show a toy (minimalist and linear - no personalization of the learning path) study guide with three instructional content elements. Figure 31 presents the study guide.

Figure 31 – Toy study guide example

Source: Author

The Table 12 provides an overview of a study guide evaluation from a student's perspective. It outlines the characteristics of four hypothetical students using the study guide in their home studies. The aim is to offer an understanding of the students' experiences, views, and feedback regarding the guide's effectiveness in enhancing their learning outcomes.

Table 12 – Hypothetical students for the toy study guide

Student	$content_1$ **watched**	$content_2$ **status**	$content_3$ **score**	S_e	SG_e	$\overline{S}_e$	S_p	$\overline{S}_p$
HS1	82%	skipped	4.6	41%	41%	59%	4.6	6.5
HS2	100%	read	9.2	100%	100%		9.2	
HS3	15%	read	6.6	57.5%	57.5%		6.6	
HS4	75%	skipped	5.6	37.5%	37.5%		5.6	

Source: Author

In the given example, hypothetical student 3 engaged with only 15% of the video lesson content, "marked as read" of the PDF content, and achieved a score of 6.6 in the form assessment. These data inputs allowed the computation of the student's engagement, calculated at 57.5%, and their performance, evaluated at 6.6 (form's score). Considering the cumulative dataset of all four hypothetical students, the aggregate engagement across the study guide activities culminated at 59%, coupled with an average performance of 6.5. These outcomes provide a simulated insight into the student cohort's collective participation levels and performance metrics throughout the study guide's learning activities.

4.4.2 Example of software requirements flipped class

The software industry frequently grapples with challenges when seeking Software Engineering professionals possessing technical skills in software (requirements) engineering. As students embark on their professional journeys, they encounter hurdles concerning the alignment of the skills they've acquired during their academic training with the practical demands imposed by the professional environment. This incongruity often creates a gap between theoretical knowledge and its practical application, subsequently impeding a transition into the professional environment. In this context, Akdur (2022) advocates for concerted efforts from both academics and the industry to bridge this gap. The university should endeavor to offer students practical experiences that mirror real-world scenarios. To substantiate this notion, we developed a software engineering topic lesson as an exemplar that intertwines theory and practice. This lesson specifically pertains to the domain of software requirements, made from our personalized study guide model. The lesson content is tailored for students enrolled in a technical course in middle-level informatics. Our software requirements flipped classroom example is based on three steps (illustrated in Figure 32).

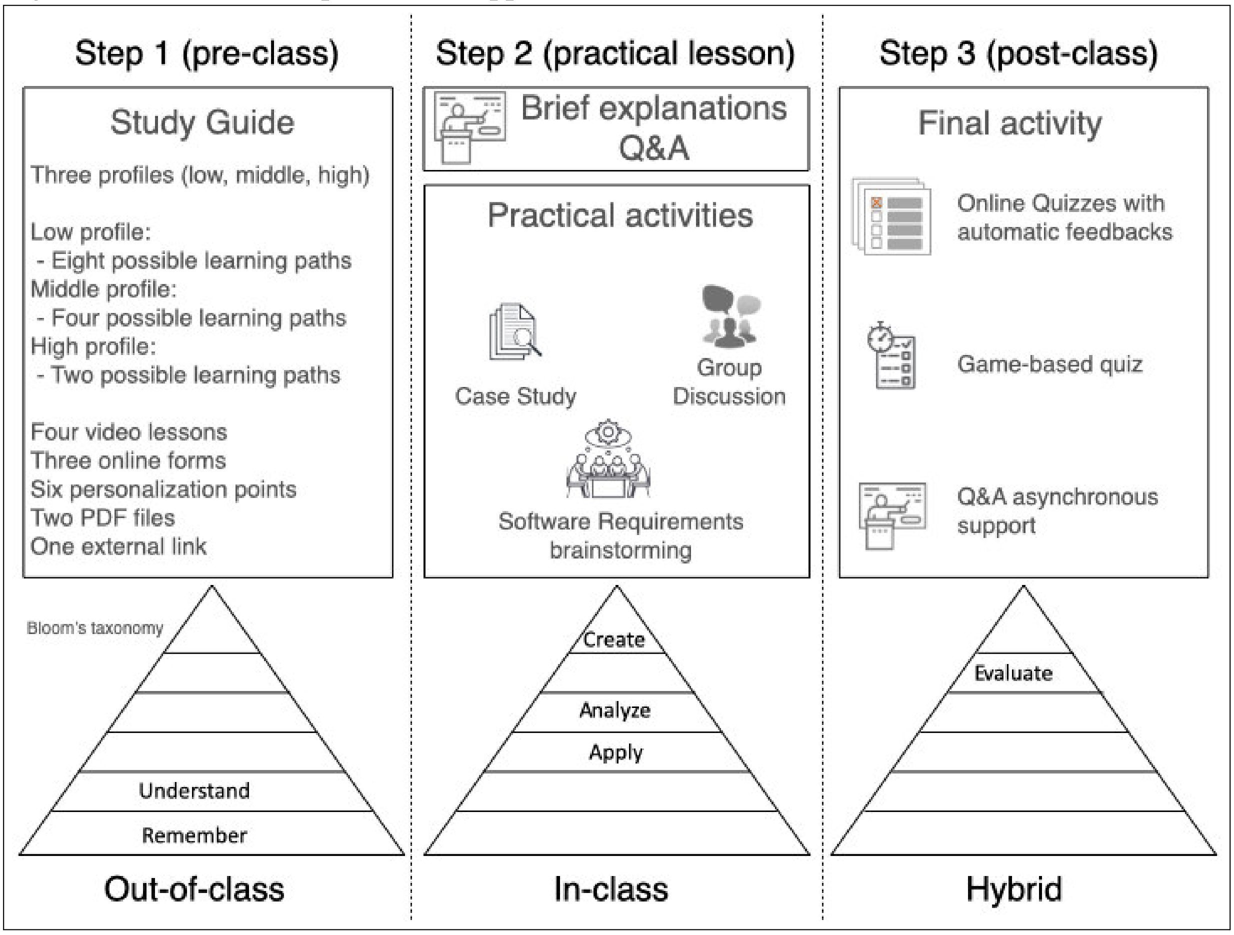
4.4.2.1 Step 1: Pre-class

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, the study guide lesson goals are to:

a) **identify** the initial aspects of software requirements,
b) **define** the requirements discovery process, and
c) **elaborate** a case study-based guide for interview.

The study guide described in step 1 starts with the student watching a video lesson

Figure 32 – Software requirements flipped class



Source: Author

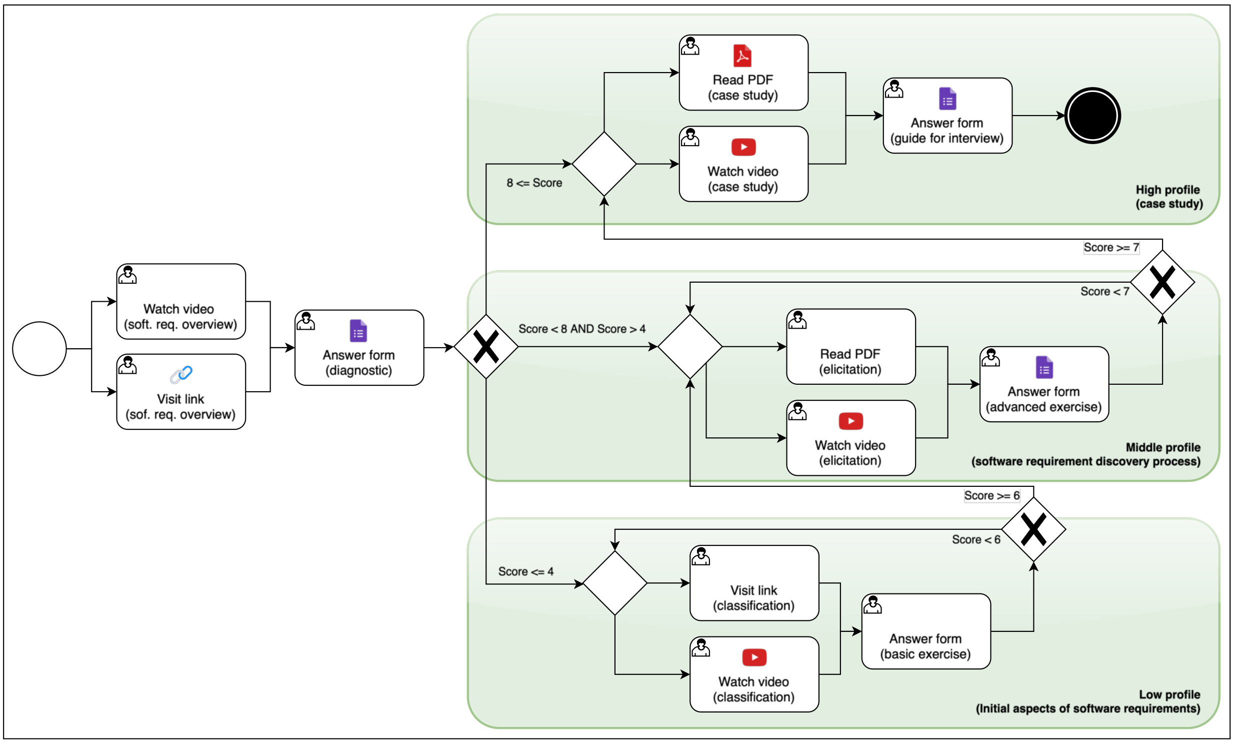
and answering an exercise (form). Then, the learning path personalization happens according to the student's **performance**. There are three profiles for personalization:

1. **Low score** (score ≤ 4): students who did not identify the initial aspects of requirements engineering,
2. **Middle score** (score > 4 AND > 8): students who did not define the requirements discovery process, and
3. **High score** (score ≥ 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 score" profile, they must navigate through the preceding pathway iteratively until they attain a score that satisfies the advancement criteria. It is incumbent upon all students to ascend to the "high score" 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 33 offers a graphical representation of the study guide with personalization points. This depiction is grounded in our BPMN-based model, as expounded upon in Section 4.3.

Figure 33 – Software requirements study guide



Source: Author

The personalized study guide can be an important tool for leveling students during pre-class studies. Ensuring that students have a relatively similar knowledge base when they enter the collaborative learning environment is key 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. This example of a personalized study guide has eleven instructional contents and six personalization points. Students in the "low score" profile will have eight possible learning paths, while those in the "middle score" profile will have four paths. The "high score" profile has two possible paths. Table 13 gives full details of each piece of instructional content.

Table 13 – Content of study guide elements

Instructional content	Profile	Link
Overview (video)	General	<https://www.youtube.com/watch?v=jajQyzOpLaE>
Overview (link)		<https://www.devmedia.com.br/introducao-a-requisitos-de-software/29580>
Diagnostic form		<https://forms.gle/B5Tujyx4pQLNQF5u6>
Classification (video)	Low	<https://youtu.be/WnO2r-HnvBk>
Classification (link)		<https://www.devmedia.com.br/artigo-engenharia-de-software-3-requisitos-nao-funcionais/9525>
Basic exercise		<https://forms.gle/jbXS8GsoBCxoTg1P6>
Elicitation (video)	Middle	<https://youtu.be/RFG_Do-PS0A>
Elicitation (PDF)		<https://www.dropbox.com/scl/fi/tqf9gzw7po92flztv0es9/Apostila-ES-cap2-Entrevista.pdf?rlkey=odtg3f7cmhhbd2icmlof222c8&dl=0>
Advanced exercise		<https://forms.gle/HEGBk6DMafCXqtji7>
Case study (video)	High	<https://youtu.be/nZRaPNF4QyI>
Case study (PDF)		<https://www.dropbox.com/scl/fi/u4w934olbi066v87e0ikw/Atividade-Elabora-o-de-Roteiro-para-Entrevista.pdf?rlkey=cjwiu6cmw14jh9e86ff972ryc&dl=0>
Guide for interview		<https://forms.gle/WUGff7udwHyBMZMd9>

Source: Author

4.4.2.2 Step 2: Practical class

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 [1]. 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 will, 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.

The practical phase is an integral part of the overall learning process for students.

[1] 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>

It allows them to apply the theoretical concepts and insights gained in a tangible, real-world-oriented domain. This phase enables students to develop a practical understanding of software requirements and how they can be implemented in real-life scenarios. Through hands-on experience, students can learn how to identify and analyze the specific needs of different software applications and develop practical solutions to meet those needs. The practical phase also facilitates students' interaction with real-world problems and helps them develop problem-solving skills, critical thinking, and decision-making abilities. Overall, this phase serves as a conduit for students to bridge the gap between theory and practice, enabling them to understand the software development process.

4.4.2.3 Step 3: Post-class

The third phase of the personalized study guide model occurs after the in-class practical lesson and is focused on evaluating the student's progress and acquired proficiencies. This evaluation process is centered around a final activity that can be carried out either during the next meeting in the classroom or through an asynchronous method, such as out-of-classroom assignments.

In the software requirements flipped classroom example context, the third phase unfolds across two stages to ensure a thorough assessment of students' abilities. The first stage involves gauging the student's comprehension of the material covered in the previous phases. The second stage in this phase aims to assess the student's ability to apply the acquired knowledge and skills in practical scenarios. This stage usually involves project-based assignments that require the students to demonstrate their proficiency in solving real-world problems using the concepts they have learned. Overall, the third phase of the personalized study guide model is critical in ensuring that students have acquired the necessary knowledge and skills to succeed in their learning journey and in providing feedback to educators on the effectiveness of their teaching methods.

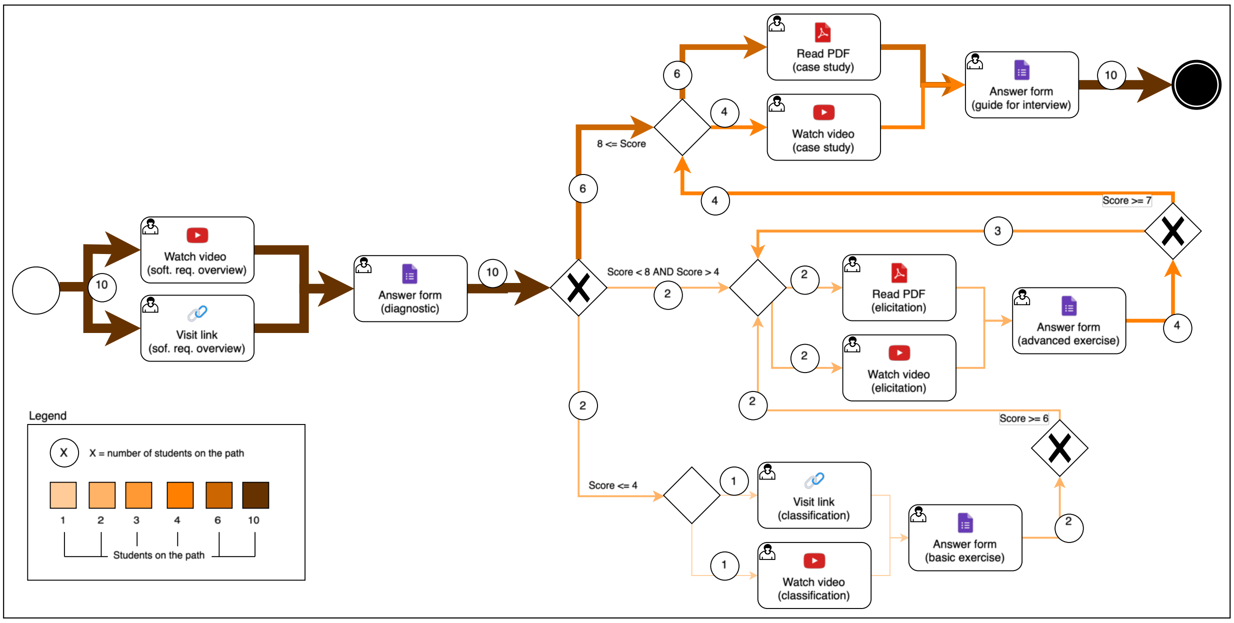
The first stage occurs between the end of step 2 and the next in-class lesson. During this period, students are prompted to engage with an online form, giving them the autonomy to respond from their home environments. This approach's inherent advantage lies in providing automated feedback, fostering 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.

4.4.2.4 Computer-based simulation

In order to observe the impact of various learning paths and evaluations on student performance, we conducted a computer-based simulation with a group of ten fictitious students. The simulation generated a distribution of six students in the "High score" profile, two in the "Middle score" profile, and two in the "Low score" profile. Each student was assigned a unique trajectory based on their identified profile. To achieve the "High score" profile, students in the "Low score" profile had to cross both the "low" and "middle" paths. It was observed that some students, namely HS7, HS9, and HS10, entered a looping process until they achieved a minimum score requirement to transition to a different profile.

The simulation demonstrated the potential of calculating individuals' and groups' engagement and performance. The simulation results were encouraging, but it is crucial to note that this was merely an illustrative example and not a definitive study. The data used in the simulation was randomly generated, meaning the results should be viewed as indicative rather than conclusive. A table showing the randomized simulation results can be found in Table 14, while Figure 34 graphically depicts the learning path data on simulation.

Figure 34 – Software requirements learning path data simulation



Source: Author

Table 14 – Computer-based simulation outcomes for software requirements study guide

Student	$video_1$ **score**	$form_1$ **score**	**Profile**	**Learning path**	S_e	SG_e	$\overline{S}_e$	S_p	$\overline{S}_p$
HS1	36%	10.0	High	PDF: read ; Form: 7.0	68%	19.4%	50.6%	8.5	7.2
HS2	67%	9.0	High	Video: 8% ; Form: 4.0	37.5%	10.7%		6.5	
HS3	59%	4.0	Low	Link: read ; Form: 10.0	64.7%	37%		7.0	
			Middle	PDF: skipped ; Form: 7.0					
			High	PDF: read ; Form: 7.0					
HS4	96%	9.0	High	Video: 31% ; Form: 6.0	63.5%	18.1%		7.5	
HS5	92%	8.0	High	PDF: read ; Form: 3.0	96.0%	27.4%		5.5	
HS6	6%	10.0	High	Video: 3% ; Form: 9.0	4.5%	0.5%		9.5	
HS7	8%	5.0	Middle	Try 1: Video:19% Form:1.0 Try 2: Video:57% Form:1.0 Try 3: Video:55% Form:0.0 Try 4: Video:22% Form:5.0 Try 5: Video:58% Form:9.0	22%	9.4%		7.0	
			High	PDF: skipped ; Form: 7.0					
HS8	0%	10.0	High	PDF: read ; Form: 4.0	50.0%	14.2%		7.0	
HS9	8%	4.0	Low	Video: 96% ; Form: 9.0	26%	15.7%		6.0	
			Middle	Try 1: Video:6% Form:6.0 Try 2: PDF:skipped Form:8.0					
			High	PDF: skipped ; Form: 3.0					
HS10	87%	6.0	Middle	Try 1: Video:8% Form:0.0 Try 2: Video:90% Form:0.0 Try 3: PDF:skipped Form:5.0 Try 4: PDF:skipped Form:1.0 Try 5: PDF:read Form:8.0	74%	44.6%		7.3	
			High	Video: 35% ; Form: 8.0					

Source: Author

The simulation results emphasize the model's benefits in supporting the teacher during the flipped classroom. From the perspective of the instructor, in brief, the model allows:

(1) The design of an adaptive study guide plays a pivotal role in leveling the students in an attempt to prepare them for the practical class session.

(2) Effective monitoring of students as they engage in their home studies. This monitoring enables teachers to identify any challenges or difficulties students may encounter during their preparation, enabling targeted interventions and personalized support when needed.

(3) Quantitative assessment of student engagement and performance during preparation. This data-driven approach provides insights into each student's progress, enabling understanding of their strengths and areas that require improvement.

From the student's perspective, the model guides the student progressively during their home studies, providing a personalized learning path that adapts to their knowledge and skills. Furthermore, when configured to act with equity, the model ensures that students with less prior knowledge of the subject receive additional support through more instructional content.

4.5 Chapter summary

This chapter introduced our personalized study guide-based flipped classroom. Initially, the model was informally described and designed following three steps before, during, and after practical classes. For each of the steps, some possible activities were related to the level of mastery in Bloom's Taxonomy. The model provides the teacher a pathway to conduct flipped classes based on a personalized study guide. Students who follow the study guide while preparing for the practical class receive personalized instructional content based on their prior knowledge and performance/engagement during their studies.

Next, the chapter formalizes the model using an adapted BPMN notation, containing elements representing student data, instructional content, personalization points, and content flow. The student model incorporated dynamic data about performance, engagement, and students' preferences regarding the kind of instructional material. Nine instructional contents were formally defined in terms of attributes, interaction levels, and visual representations. Five personalization points were formalized in terms of personalization rules and visual representation. Two types of rules were structured: quantitative and qualitative rules. The chapter exemplifies each visual element (instructional contents and personalization points).

The chapter continues the formalization of the model by presenting two pathways to evaluate the personalized study guide from the student's perspective: engagement and performance assessment. Afterward, a proof of concept of the model is presented through a minimalist example (toy), and a most comprehensive example of software requirements flipped class. In the software requirements flipped class example, all steps are detailed, describing learning objectives, the study guide, practical activities, and materials. Finally, the chapter presents results from a computer-based simulation with ten fictitious students studying from the adaptive study guide presented. The generated data emphasized the benefits of the model in supporting the teacher in developing the study guide as well as the advantages of the model from the student's perspective.

In the upcoming chapter, Chapter 5, we discuss an implementation example of our model. This implementation is a web-based tool designed to act as a minimally viable product. We also conduct a real-data evaluation of the model in Chapter 6. This evaluation provide a more comprehensive assessment of the model's performance.

5 A FLIPPED CLASSROOM TOOL

In this chapter, we present the Flipped Classroom Tool. A software web-based tool that implements the model previously introduced in Chapter 4. The tool was developed as a MVP of the proposed model to complement the answer to our third research question (RQ3): *Is feasible to design, formalize, and implement a Personalized Study Guide Model to represent out-of-class material for SE flipped classes?*

The tool is described in Section 5.1, providing insights into its conception. Contrarily, Section 5.2 analyzes the tool's features and the model, relating their respective elements. Also, the section pinpoints the limitations of the tool. Moving forward, Section 5.3 enumerates the implemented features and presents the developed prototype. Finally, Section 5.4 summarizes the chapter's key findings, emphasizing the tool's strengths in the context of flipped classrooms and personalized education.

5.1 Flipped classroom tool

The model presented in Chapter 4 is flexible to be implemented in many ways, such as in a mobile application, a web-based system, or a plugin for a specific learning management environment (e.g., Moodle). Each implementation method has its own merits and disadvantages. Among the multiple options available for implementing the model, we chose a responsive web-based approach. This choice was deliberate and grounded in factors strengthening the students' learning experience. Nowadays, a Web version provides ubiquitous access, allowing students to engage in proposed activities from anywhere and anytime. The distributed nature of the web eliminates geographical barriers and offers a learning experience for students in different locations.

Additionally, web implementation facilitates the addition of interactive features, multimedia resources, and collaborative environments, further enriching the learning process. The interactivity provided by the web creates opportunities for the immediate application of theoretical concepts, making learning more engaging and practical. We acknowledge, however, that circumstances and preferences may vary, and we encourage other researchers to adapt the model as needed to meet the specific needs of their students and teaching environments.

In this sense, our Flipped Classroom Tool (FCTool) is a responsive web-based authoring tool with graphical resources to help teachers build personalized study guides and collect valuable feedback on student performance. The tool was designed to be integrated with a virtual learning environment like Google Classroom. It allows the teacher to insert activities individually or in groups, organizing them in a visual study guide like a learning path that students can follow outside the classroom. The FCTool was developed to provide a platform where learners can increase their knowledge in activities through study guides. Simultaneously, the tool empowers teachers to generate personalized learning resources for their students. Further, the interface of our web-based implementation is responsive, meaning it adapts to the devices used by both students and teachers.

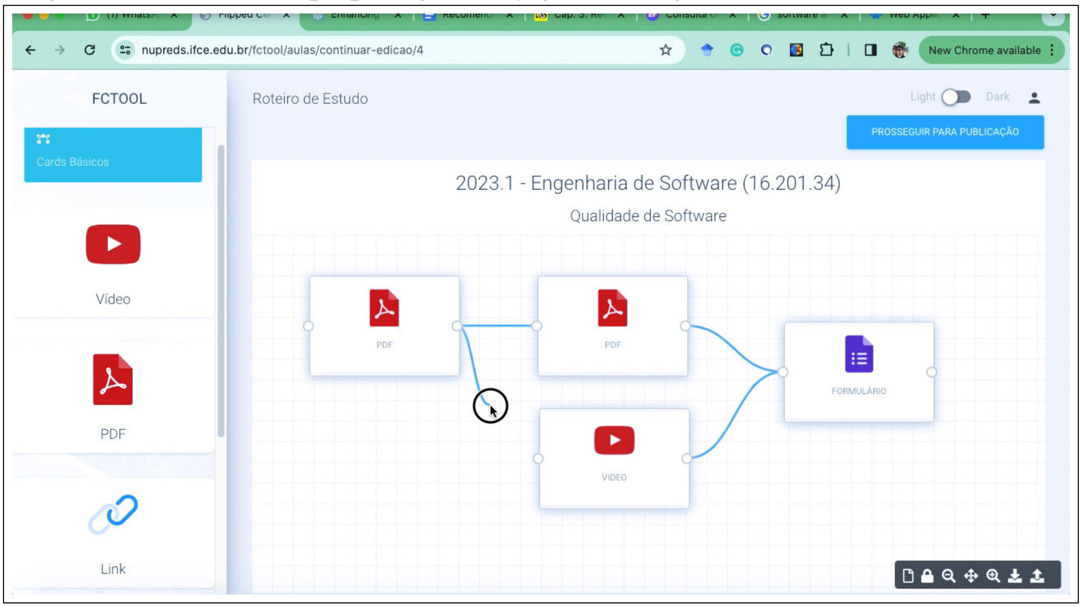
The tool has features that help teachers create engaging and interactive learning experiences for their students. By leveraging the FCTool, teachers can create study guides tailored to their students' needs, increasing their chances of achieving academic success. Additionally, students can benefit from the tool's feedback mechanism, which provides insights into their learning progress, allowing them to make informed decisions about their studies. The FCTool is an asset for teachers looking to enhance their student's learning experience using our personalized study guide-based model.

Inspired by our prior experiences using flipped classes (showed in Araujo *et al.* (2018), Araujo *et al.* (2019), Araújo (2019), Araújo *et al.* (2020)), 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 befalls 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. After, he dedicates some minutes to answering FAQs brought by students. In the remaining class, professors conduct practical activities (like quizzes, software games, project planning, requisite meets, programming tasks, and brainstorms). Finally, after the practical class, the professor works on a review activity.

Before the practical class, the process of the flipped class begins with the production of the study guide to be sent to the students (I - Professor, Prepare). With the use of FCTool, the teacher can build a study guide using a graphical web environment with drag-and-drop resources to elaborate the learning path followed by students (illustrated by Figure 35).

In the next phase, after receiving the study materials inside FCTool, the students followed the guidelines for studying the contents and resolution of evaluation quizzes in the study guide (II - Students, Study, and Self-Assessment). Students are involved in practices related to

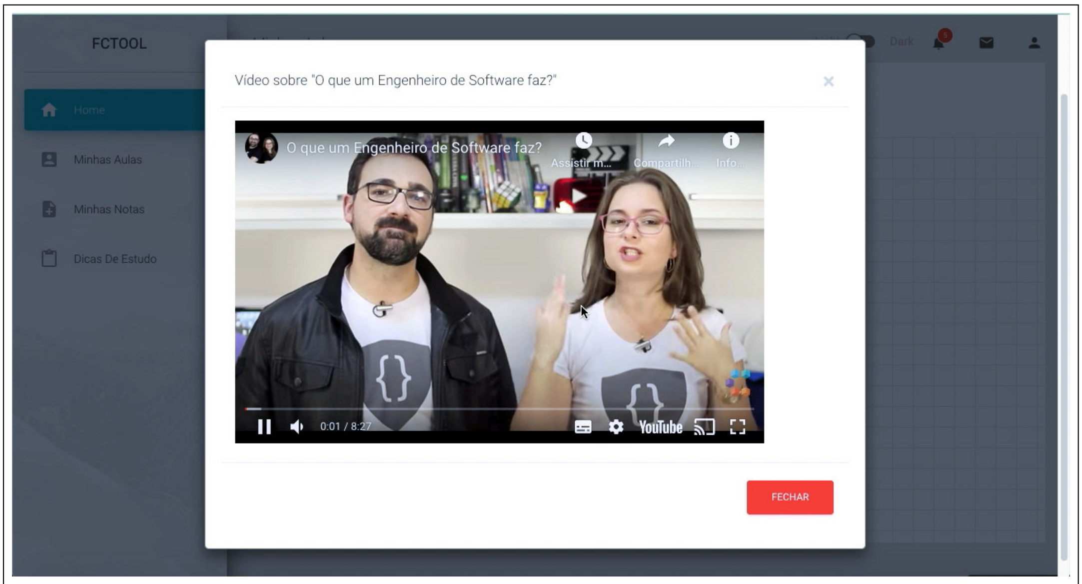
Figure 35 – Professor preparing a study guide using FCTool



Source: Author

the content under study (III - Students, Practice). After the practical class, a review lesson is prepared (IV - Professor, Prepare) and sent using FCTool to review the content (V - Student, Review). Figure 36 exemplifies a student studying through a video lesson within the FCTool.

Figure 36 – Student watching a video lesson into FCTool

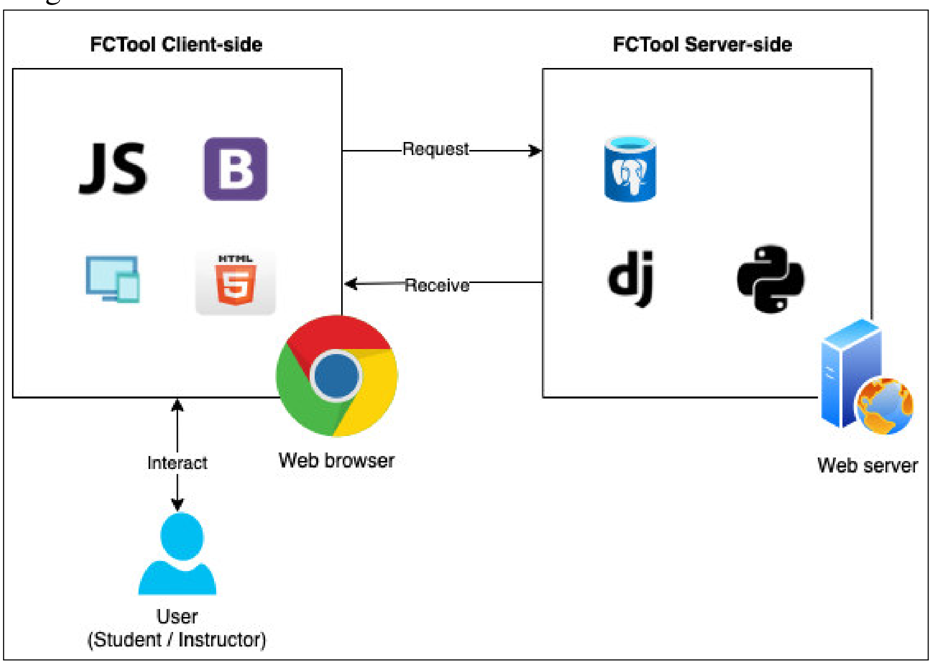


Source: Author

The approach aims to provide a professor with an innovative way to conduct a Flipped Classroom, starting with a personalized study guide. In each class, the professor responsible for the course models the content and possible learning paths the students would follow, ensuring that the students are actively engaged in the learning process. The students access the tool using a link via the academic system to provide their answers. The FCTool allows the teacher to create instructional materials for moments before and after the practical class, making the learning experience more interactive and engaging. The teacher can create various materials, including interactive videos, quizzes, and other multimedia resources, to help students learn more effectively. The FCTool has been designed to help teachers provide a personalized learning experience to each student, allowing them to learn at their own pace and level of understanding. The approach was designed to create a more dynamic and interactive learning environment that encourages students to be more actively engaged in the learning process. It also provides teachers with the necessary tools to create instructional materials that cater to the unique needs of each student.

Figure 37 provides an overview of FCTool through the interactions among its components. Using a web browser, users (students and instructors) directly interact with the FCTool client side. Since the interface is responsive, users can utilize a browser on any device (desktop, laptop, mobile, or TV). The client side has been implemented using JavaScript, HTML5, and Bootstrap (version 4.5.3) to interact with FCTool server-side and return the results to users. The server side receives requests from the browser, retrieves and processes data, and then sends a response to the request. The technologies used in the implementation include the Django framework (version 3.0), the Python language (version 3), and the Postgres SQL database manager (version 10).

Figure 37 – FCTool overview

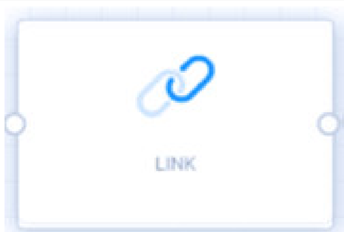


Source: Author

5.2 Personalized Study Guides in FCTool

In the current version, the tool offers four instructional content types implemented based on the contents presented in the model (Reading, Multimedia, Score-based, and Text-based). The elements present in the tool are “video lesson”, “PDF file”, “link”, and “form”, detailed by Table 15.

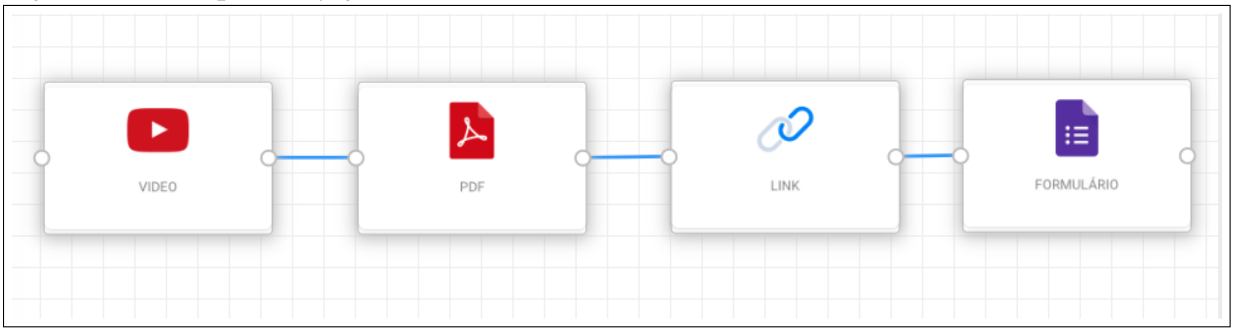
Table 15 – Instructional content elements into FCTool

Element	Description	Visual representation
Video lesson	Professor configures the title and link for a YouTube video. The tool utilizes the YouTube API (<https://developers.google.com/youtube/v3/docs>) to manage video controls for the student’s view, enabling the student to perform the following controls: play, pause, backward, and repeat video.	
PDF file	Professor provides a link to a PDF file, and the tool presents the file’s content to the student. After reading, the student can mark the content as read or, if preferred, mark it as skipped.	
Link	The “link” content works similarly to PDF; the student can decide whether to access the external resource or skip the content.	
Form	Professor configures a Google Form for the student. The tool presents the form content to the student and utilizes the Google Forms API (<https://developers.google.com/forms/api/reference/rest>) to retrieve and store the student’s performance. The tool retrieves and stores student actions in each instructional content, providing data for the instructor to conduct personalized monitoring.	

Source: Author

Figure 38 illustrates the sequential flow of a study guide created in the tool using the four instructional contents. In this example, when students receive the study guide, they can view and explore all instructional contents in the preferred order.

Figure 38 – Simple study guide made in FCTool

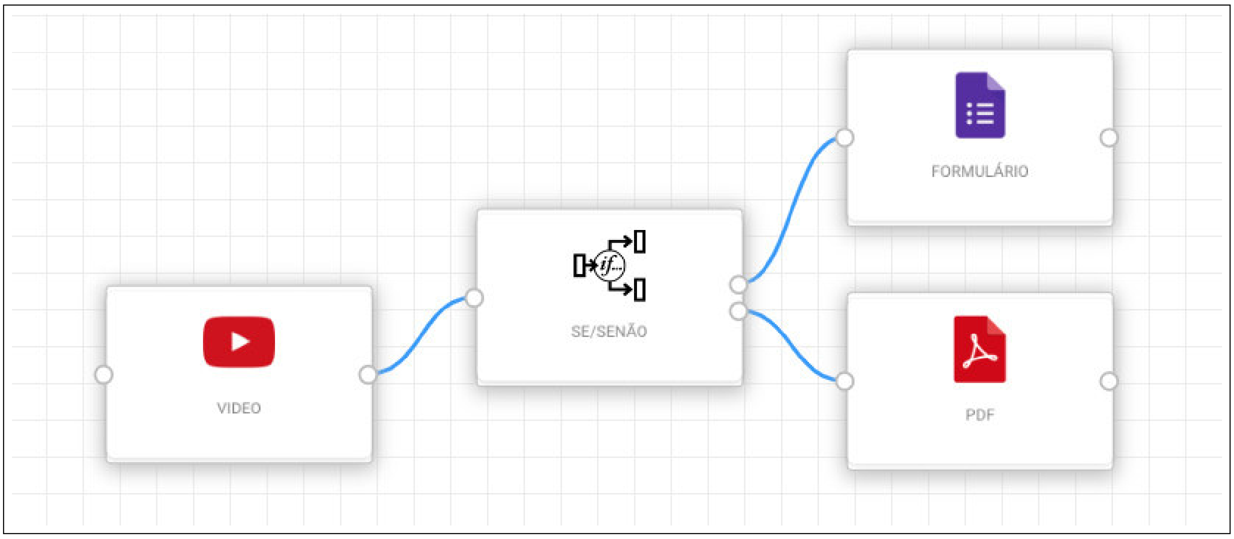


Source: Author

5.2.1 *Personalization Points*

The tool also provides the personalization points outlined in our model. We described five personalization points in the model and included them in the tool. The first personalization point presented in the model was "Conditional Decision", referred to as "If/Else" in the tool. In the tool, this point has a single rule programmed by the teacher related to content linked to the point (input connection). Two possible paths should be traced from the programmed rule (upper and lower). The set of rules and possible paths in the model are undefined, but in the tool, they are limited (according to the personalization point). Figure 39 illustrates an example of using "If/Else" in the tool.

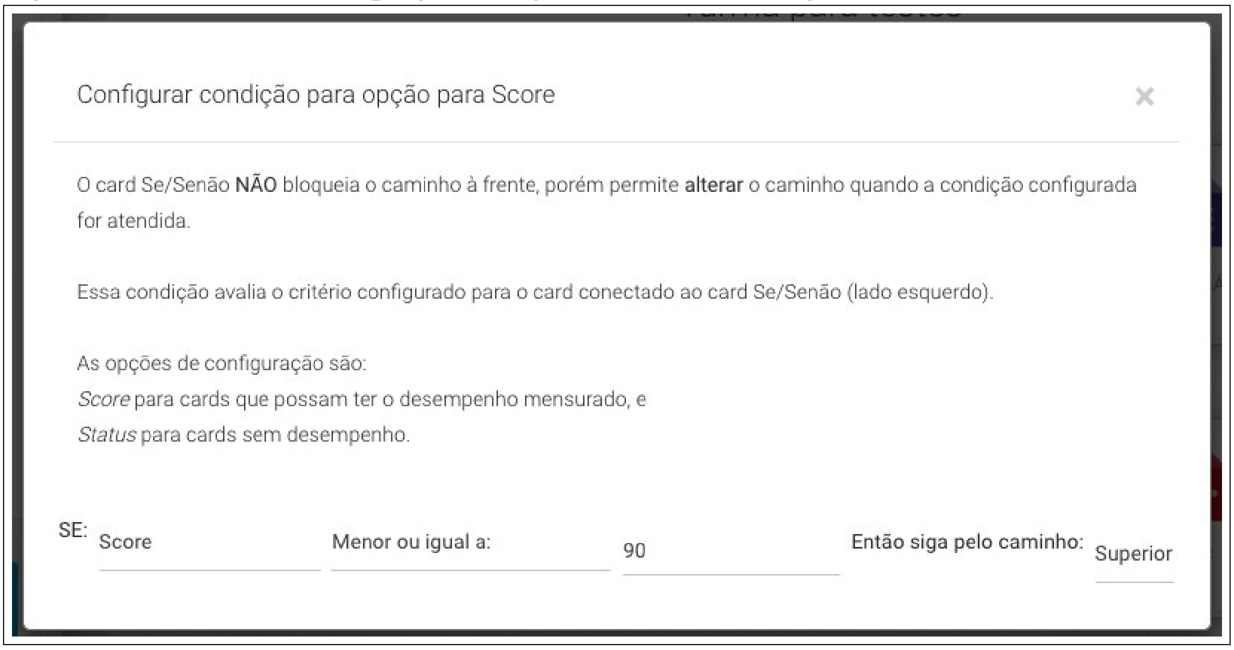
Figure 39 – Conditional decision point example made in FCTool



Source: Author

The tool's rule formulation process adheres to the same approach proposed in the model, retaining the structures presented for quantitative and qualitative rules (as demonstrated in Section 4.3). The tool allows the teacher to choose between a quantitative rule, using a score attribute, or a qualitative rule, using a status attribute. Figure 40 illustrates the configuration of a quantitative rule for the example shown in Figure 39. We configured the exemplified rule as "Greater than or equal to" 90, meaning that if true, the upper path will unlocked for the student. In other words, the student must watch at least 90% of the video for the upper path to be unlocked.

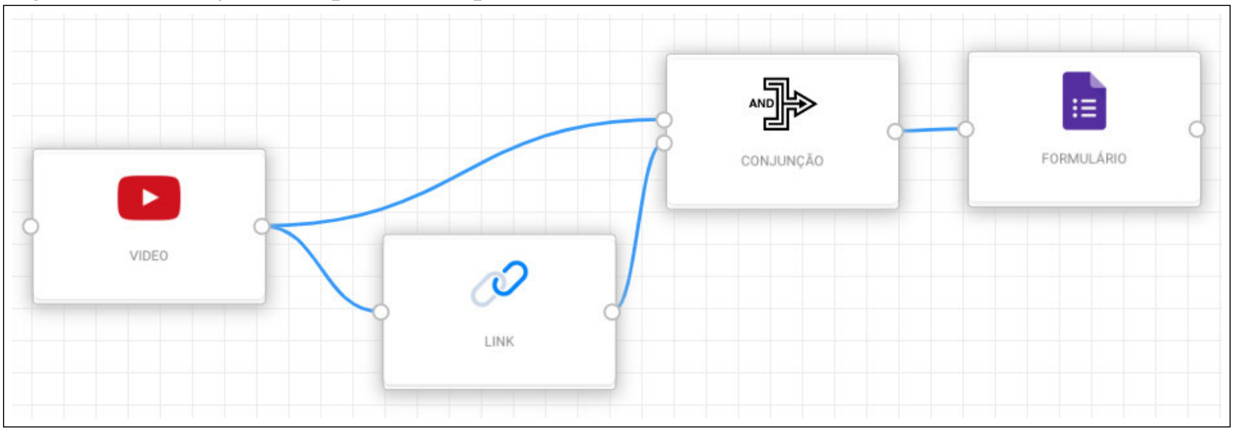
Figure 40 – Qualitative rule programming in FCTool (in Portuguese)



Source: Author

Next, the Conjunction point evaluates two separate rules connected to the CP as its input. The evaluation function acts as an AND clause, meaning the path forward is only unlocked when both rules are positively assessed. Figure 41 demonstrates the application of CP in a study guide using FCTool.

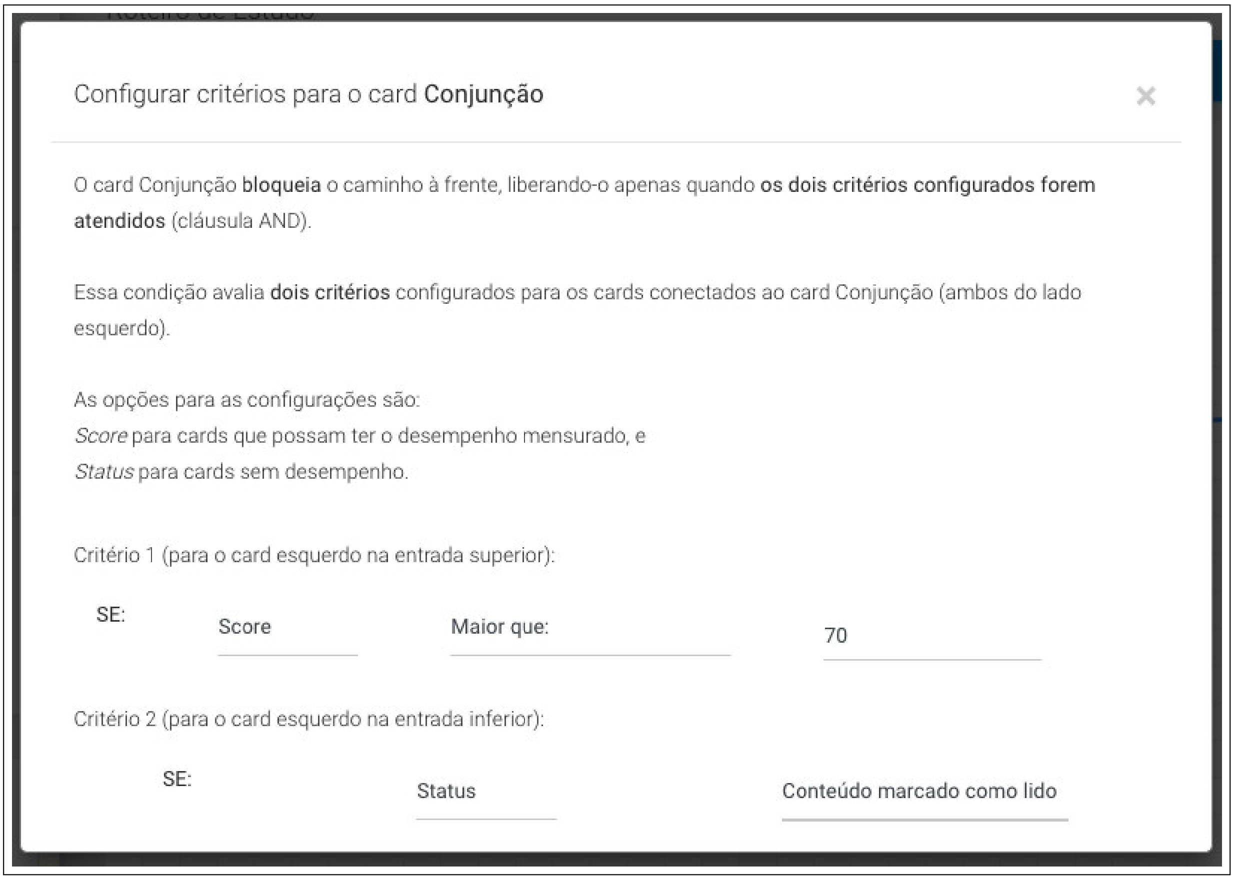
Figure 41 – Conjunction point example made in FCTool



Source: Author

The Conjunction point in the scenario mentioned above has two inputs: video and link content, and one output point, form content. When the student starts the study guide, the form will only appear if both input rules defined for the conjunction are true. To give an idea, Figure 42 demonstrates the configuration of two input rules: (1) "If score greater than 70" AND (2) "If status content marked as read". This clause implies that to access the form, the student must complete the following requirements: mark the link's content as read and watch at least 70% of the video lesson. Only when both rules are fulfilled will the student be able to access the form.

Figure 42 – Conjunction point rule configuration in FCTool (in Portuguese)

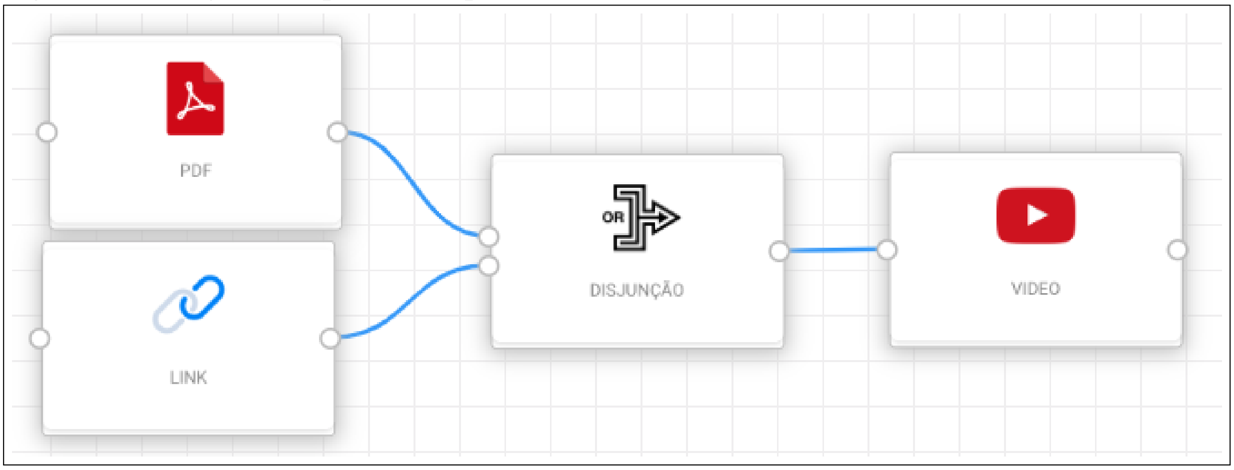


Source: Author

The Disjunction Point works based on an OR clause and shares a similar structure with the Conditional Point. It evaluates two input rules and activates one output when one of the two rules is evaluated positively. The example in Figure 43 illustrates this application. In this instance, a study guide was created with a "PDF File" and a "link" assigned to the inputs of the DP, while a "video lesson" was connected to the output. The Disjunction Point implies that once the "PDF File" or "link" input is evaluated positively, the "video lesson" output will be activated.

This operation is because the DP only requires one of the two inputs to be true for the output to be activated.

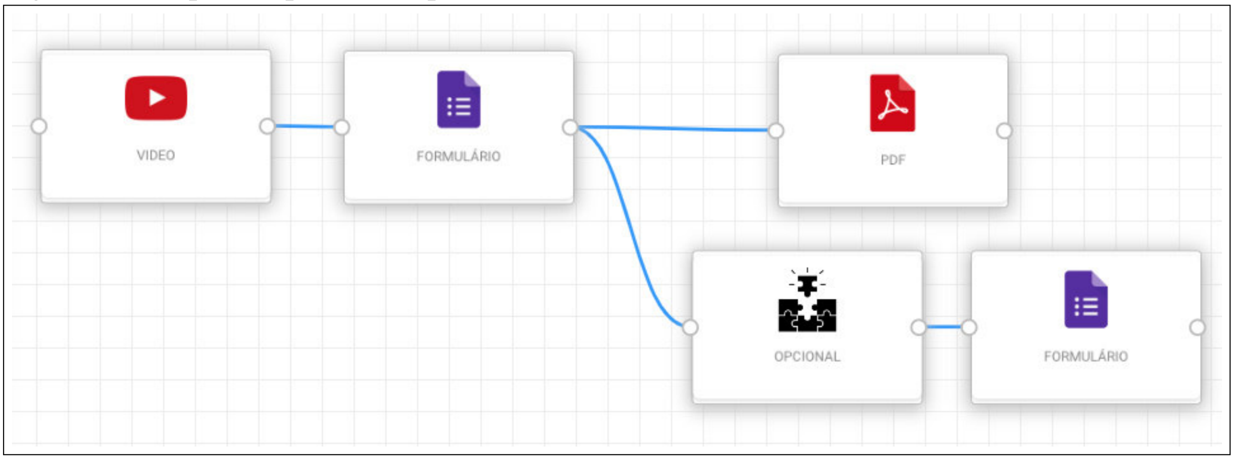
Figure 43 – Disjunction point example made in FCTool



Source: Author

The Optional Point concept is a part of the model that provides an alternative path that can be unlocked based on pre-programmed rules established by the teacher. This concept is implemented similarly in the tool as described in the model. However, it is limited to only one rule associated with a single input and output path. Figure 44 shows an example of this personalization point. In this example, when students begin their study guide, they are presented with a video lesson followed by a form. The optional point then evaluates the input from the form. Depending on the pre-programmed rule, the optional point may unlock access to another form, providing the student with a more personalized learning experience. With this feature, the tool can create a unique learning path for each student to optimize learning outcomes.

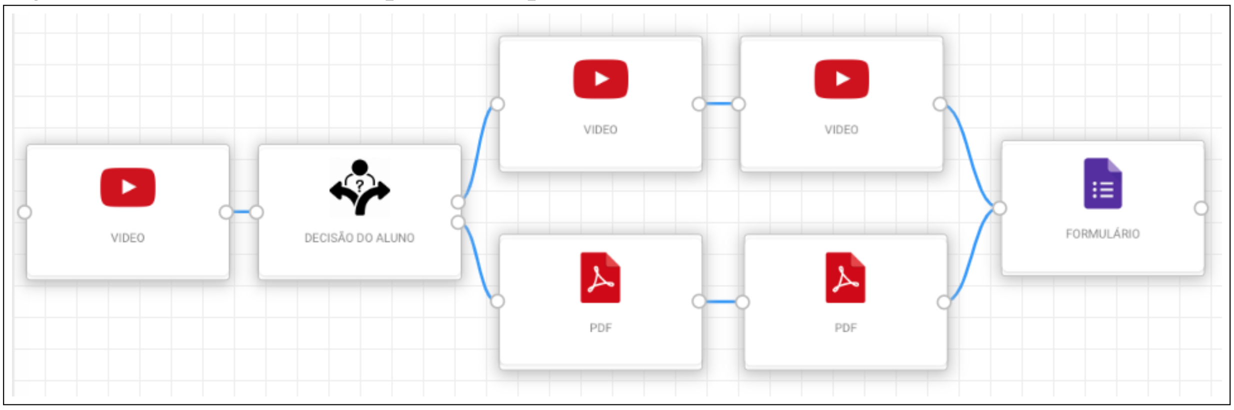
Figure 44 – Optional point example made in FCTool



Source: Author

The fifth personalization point described in the model is the Student Decision Point. In this personalization point, there is no rule programming; instead, there are explanations about the possible paths that a student can choose. The tool restricts the SDP to one input connection and two output connections. In the input connection, it is possible to associate it with any instructional content since there is no rule evaluation. After reading the information about each forward path, the student selects one of the two possibilities, and the path is unlocked. The tool allows students to change their choices anytime during their study. Figure 45 illustrates an example of SDP usage in the tool. Figure 46 presents a potential explanation of a student's paths regarding the "Student Decision" personalization point.

Figure 45 – Student decision point example made in FCTool



Source: Author

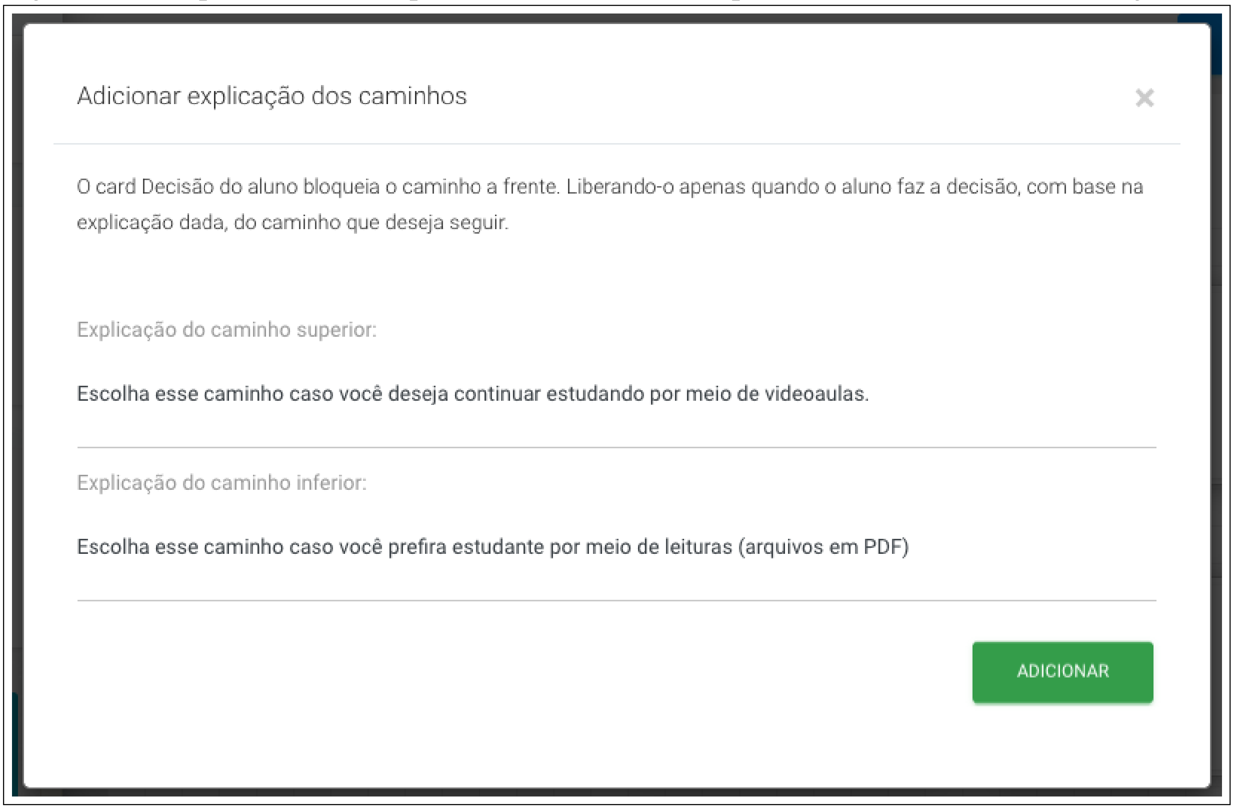
In this example, after watching a video lesson, the student can choose whether to continue studying using video lessons or reading PDF files. From the example above about SDP, Figure 46 provides a more detailed description by the professor, clarifying the two paths a student may choose. This additional information helps students decide which path to take based on their personal preferences and academic goals.

5.2.2 Running Example

In Section 4.4.2, we give a detailed example of how we use our flipped classroom model. We created a personalized study guide (see Figure 33) to help students grasp the ideas before a lesson on software requirements. We used FCtool to recreate this lesson. Figure 47 presents the same study guide modeled in our tool.

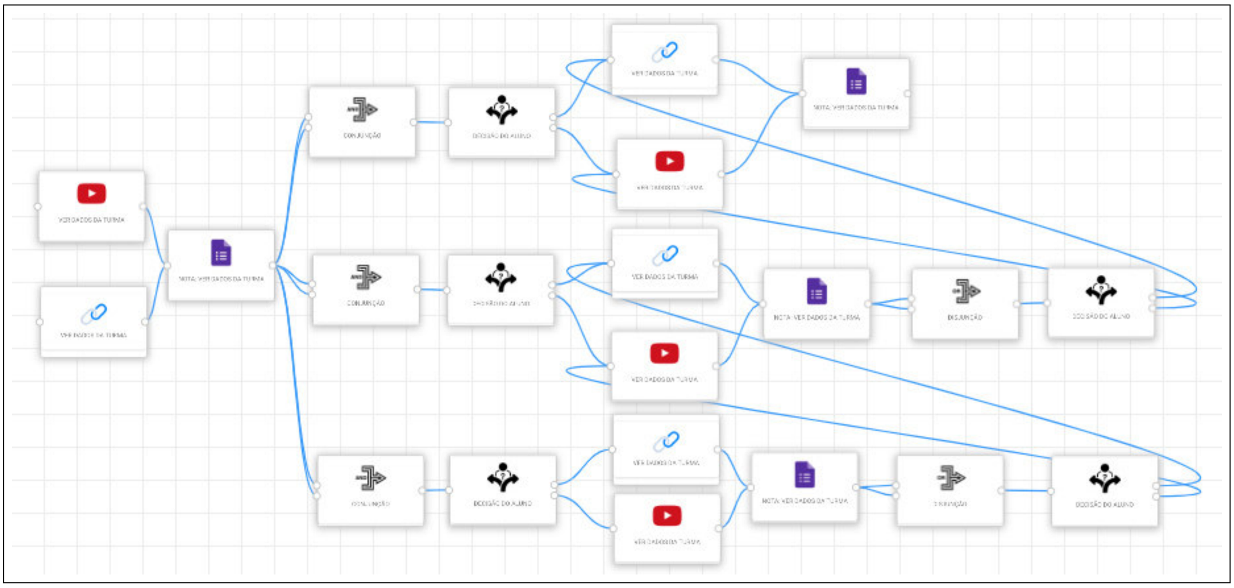
The study guide was designed to provide learners with an interactive and personalized learning experience and help them acquire knowledge of software requirements more effectively.

Figure 46 – Explanations about paths for student decision points made in FCTool (In Portuguese)



Source: Author

Figure 47 – Software requirements study guide made in FCTool



Source: Author

A student begins their learning journey by watching a video lesson and accessing reading material via a provided link. Afterward, they complete a diagnostic form determining their performance profile (low, middle, or high score). Based on their profile, the tool unlocks a specific learning path. Students can choose between reading (using a link or PDF file) or watching a video lesson before completing an exercise form. They progress through different profiles until they reach the final one (high score) and complete the last form (case study).

5.3 Features and prototype

This section serves as an introduction to the tool's available features and the developed prototype. We followed the design principles outlined in Sections 3.2 and 3.3 to create a personalized study guide tool. This process identified seventeen key features essential for the tool's effectiveness. These features were utilized to create a Minimum Viable Product that catered to our model's unique needs. We emphasize that only some design principles presented in this thesis (Sections 3.2 and 3.3) were implemented in our MVP. Table 16 lists the seventeen implemented features and their corresponding design principles (described in Chapter 3).

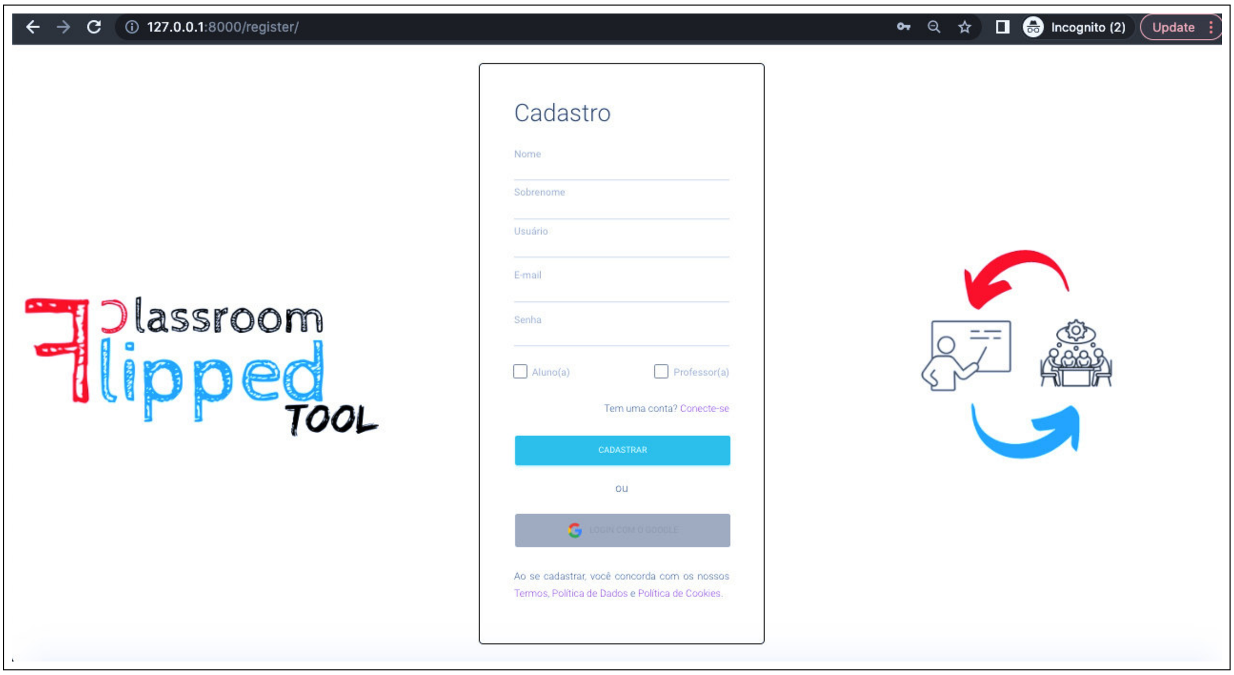
Table 16 – FCTool features

ID	Feature	Design principle
F1	Teachers and students must be authenticated by the tool	Basic
F2	Teachers can add or create virtual classrooms	P1
F3	Teachers can add students to a virtual classroom	Basic
F4	Students can access a virtual classroom	Basic
F5	Teachers and students can sign in using a Google account	P17
F6	Teachers can integrate the tool with their Google Classroom account	P17
F7	Teachers can import students from a virtual Google Classroom	P17
F8	Teachers can manage (create, list, modify, end, archive, publish) study guides for flipped lessons	P20
F9	The study guide should be made with a learning path and built using a visual interface	P3
F10	The study guide can contain content by video, text (PDF), forms, or links	P20
F11	The teacher can create a study guide using points of personalization according to student performance, engagement, or individual decisions	P21,P23,P24
F12	The teacher can reuse a study guide in others flipped classes	P2
F13	The teacher can monitor student activities and performance from a flipped lesson	P5,P9
F14	Students can view the instructional material for their flipped lessons	P13
F15	The student can view his flipped classroom activities' progress	P15
F16	Students can do activities from a study guide	P13,P14
F17	The students can use the web tool from mobile device	P8

Source: Author

For the first feature (F1), the FCTool enables the self-registration of both teachers and students. The registration form includes basic information such as first name, family name, username, email, and password. Additionally, users must specify whether the registration is for a student or a teacher. Validation of registration requests is conducted through a confirmation email, preventing registration errors and reducing attempts at anonymous access—authenticated users, whether students or teachers, can access the tool through the user management subsystem. Figure 48 illustrates the interface for self-registration. Another authentication method FCTool utilizes is the Google authentication mechanism, allowing users to link institutional accounts or use emails registered in Google Classroom. The integration of Google authentication introduces an additional layer of convenience and security to the tool, allowing users to link their institutional accounts or leverage emails registered in Google Classroom for seamless access.

Figure 48 – FCTool self-registration interface (in Portuguese)

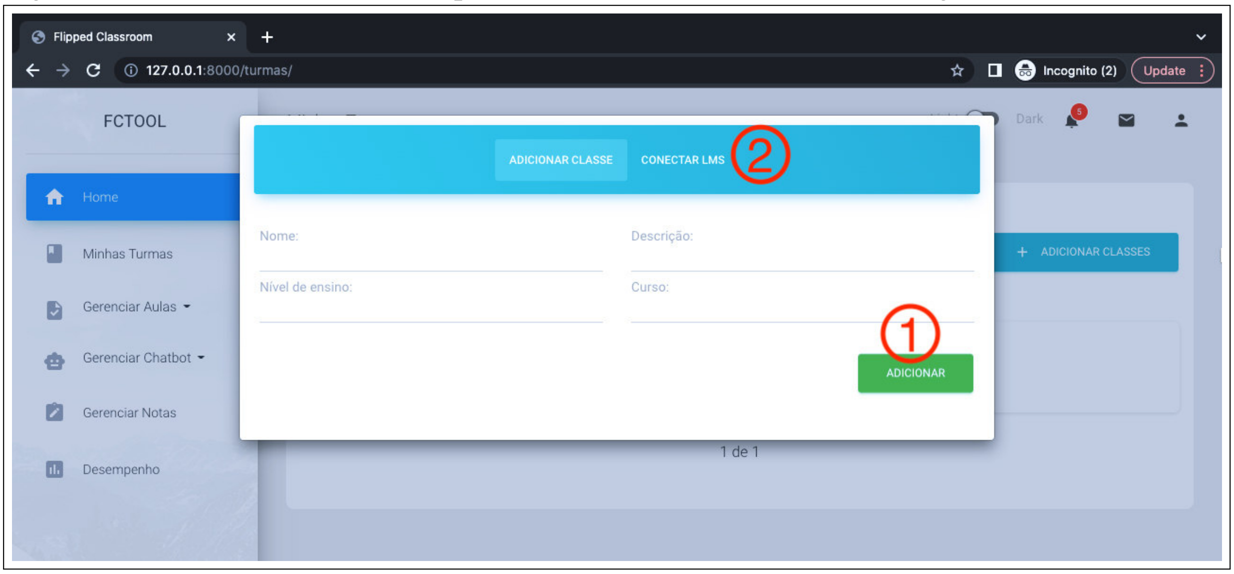


Source: Author

In feature 2 (F2), as depicted in Figure 49, teachers can create or add a new virtual classroom (numbered 1). The necessary information for creating a virtual classroom includes the name, description, teaching level, and program. The same interface also enables the connection to a Google Classroom account (numbered 2) and the importation of a virtual classroom (feature 6 - F6). During the importation process, the students in the virtual classroom are automatically added to the FCTool's virtual classroom (feature 3 - F3).

In the fourth feature (F4), students can access virtual classes through the FCTool environment. As illustrated in Figure 50, students will be presented with two types of valuable data to help them navigate their flipped lessons. Firstly, at the top of the page, students will find a summary of the number of lessons available per virtual classroom. This summary provides an overview of all the lessons available in a course, which will help students manage their time and keep track of their progress. Additionally, this overview will allow students to plan their study schedule accordingly and ensure they complete all the necessary lessons on time. Secondly, below the summary, students will find a list of all available lessons organized by virtual classrooms. This visual distribution allows the student to view all the lessons available in

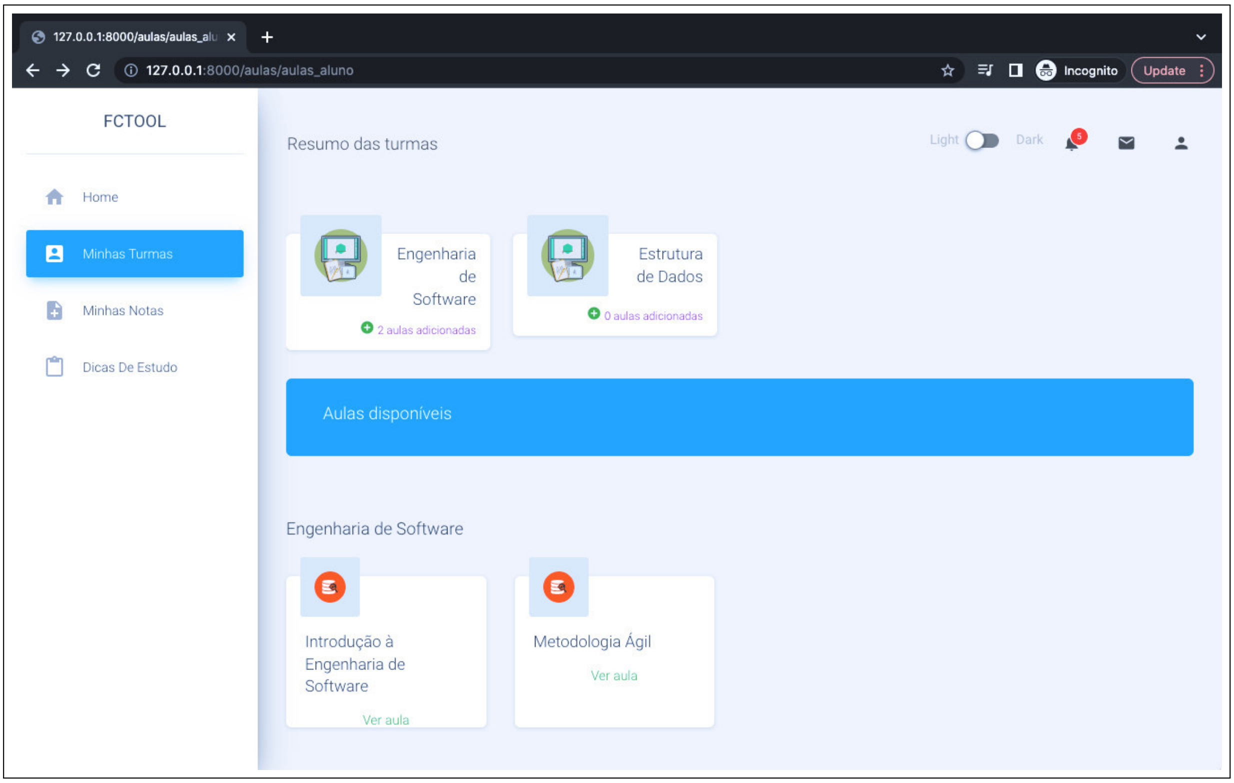
Figure 49 – Interface for add or import a new virtual classroom (in Portuguese)



Source: Author

a course or to focus on a specific lesson of their choosing.

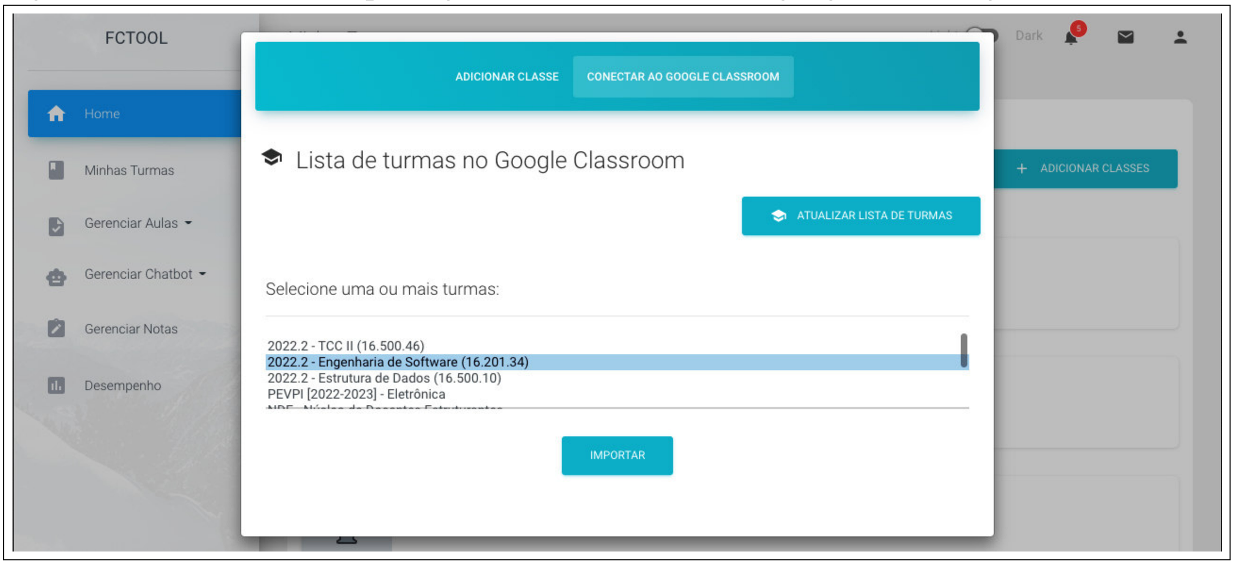
Figure 50 – Students' view of their virtual classroom (in Portuguese)



Source: Author

In FCTool, three features enable interoperability with Google Classroom (GCR). These features are named as F5, F6, and F7. With the help of these features, a teacher can easily integrate FCTool with his GCR account and import virtual classrooms with their respective students. This integration saves time and effort and provides a seamless experience for the teacher. To import a class from GCR into FCTool, the teacher can use the interface shown in Figure 51. This interface provides an intuitive way to import the class, making the entire process hassle-free.

Figure 51 – Interface for importing virtual classrooms from google (in Portuguese)

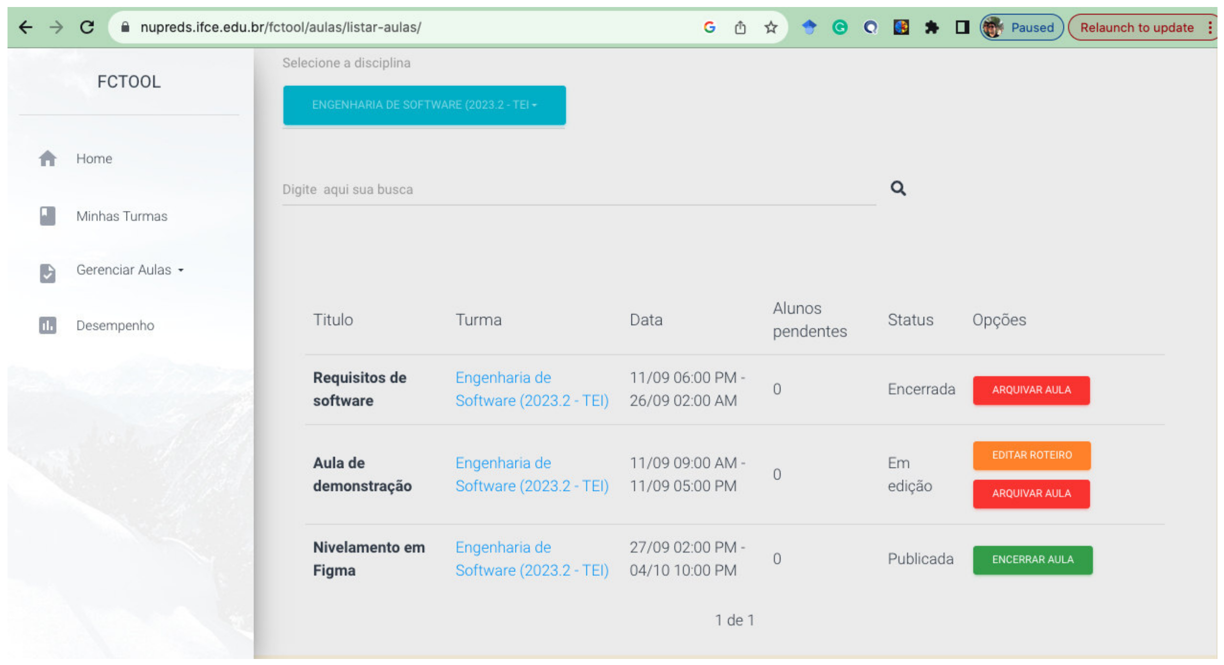


Source: Author

The eighty feature (F8) allows teachers to manage flipped lesson study guides. With the help of the FCTool, teachers can create, modify, view, end, archive, and publish different study guides for the same flipped lesson. The tool simplifies managing flipped lessons by allowing teachers to add new lessons, view the classes created, and obtain a list of archived lessons. In the list of flipped lessons, teachers can modify or archive a lesson, giving them more control over the study material they provide to their students. In addition, the FCTool enables teachers to unarchive an archived class if needed. The user interface of the tool was designed to be intuitive and user-friendly, as shown in Figure 52, which displays a list of flipped lessons created and the options to edit, end, and archive them.

The ninth feature (F9) concerns the platform's usability when designing a study guide. With the tool, teachers can visually construct a study guide that follows a learning path for students toward their educational goals. The path is curated and assembled according to the flipped lesson plan, ensuring that students can engage with the material in a structured and effective way. One of the critical components of this feature is the use of instructional cards, which are integrated into the learning path. These cards can present students with various instructional materials, including video, text (in PDF format), forms, or links to external

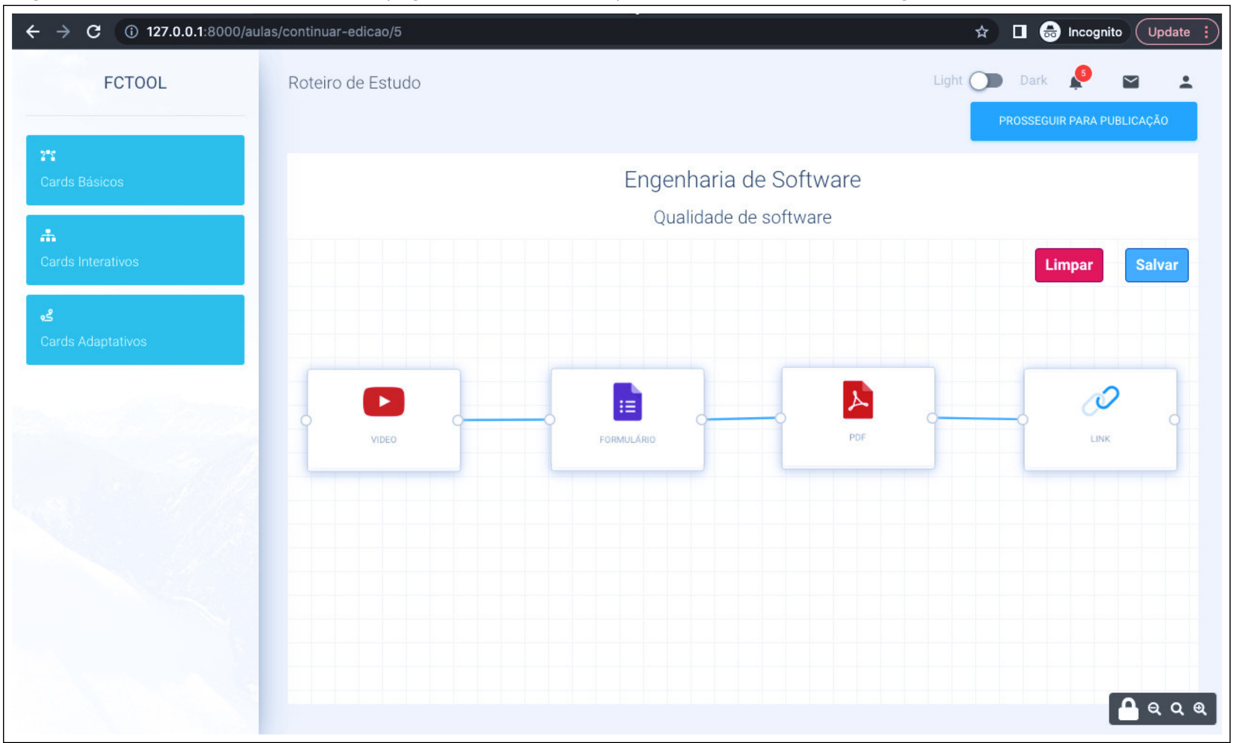
Figure 52 – Interface for study guide list from teachers' view (in Portuguese)



Source: Author

content (which is covered by feature 10 - F10). Figure 53 has been included to provide a visual representation of how this feature works. This image shows an example of a study guide that utilizes a learning path with four content cards to guide students sequentially through the material.

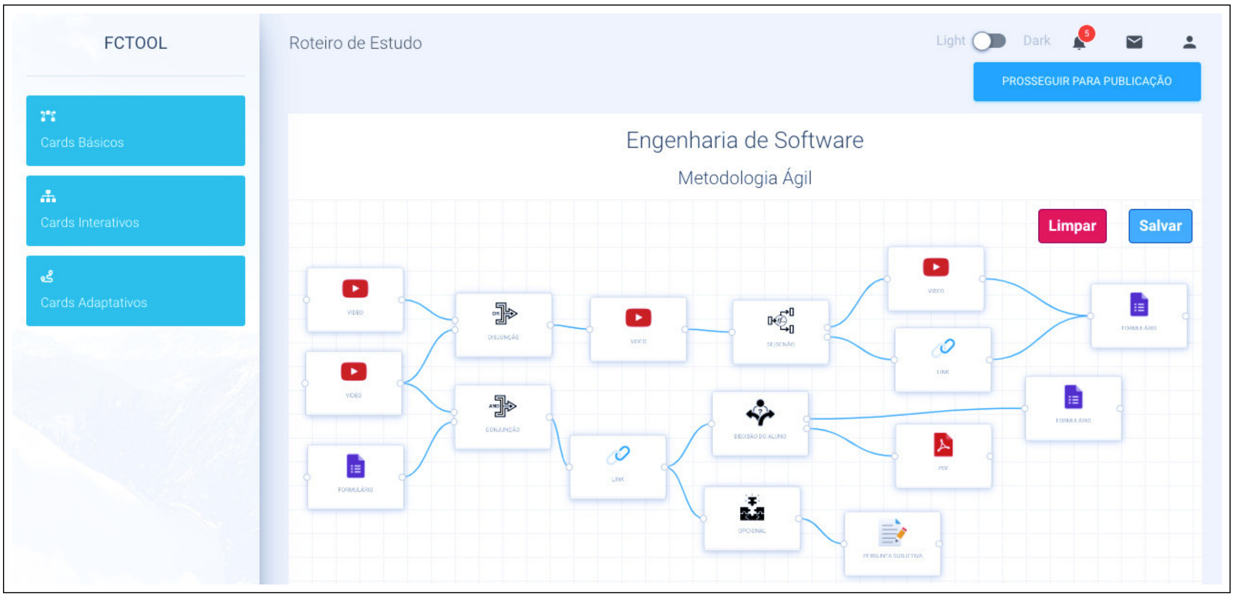
Figure 53 – Interface for study guide creation by the teacher (in Portuguese)



Source: Author

Feature 11 (F11) concerns technology use for personalization study guides (adaptive learning). FCTool allows the teacher to add and program personalization points. 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 score higher than 7 in a form *then* will follow the upper path, *else* will follow the lower path. The *conjunction* point 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 score higher than seven *then* the path will be released. The rules for the *disjunction* point resemble those of the *conjunction*, but only one of the conditions must be satisfied to release the path. The *optional* point has a single condition to release the path. The teacher can use this point to, for example, challenge the students. Finally, the *Student Decision* point allows the student to choose one of two possible paths. The teacher should explain/comment on the paths. Figure 54 illustrates a study guide with all adaptive points.

Figure 54 – Study guide with adaptive cards example (in Portuguese)

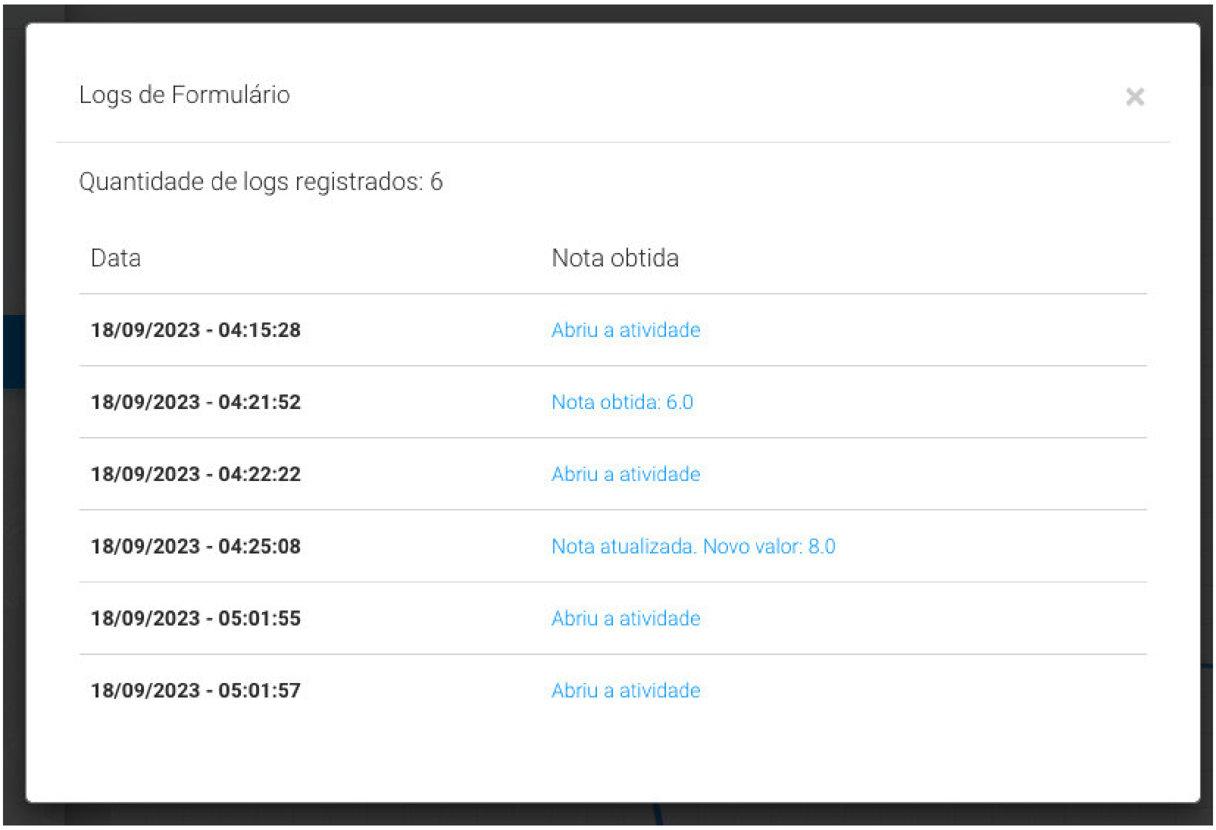


Source: Author

Feature twelve (F12) concerns the reusability of study guides by teachers. With this feature, teachers can reuse the study guides they have created in other flipped classrooms, saving them valuable time and effort. Additionally, the tool allows teachers to export or import a file corresponding to the study guide, making it easy to share with other educators.

Moving on to feature F13, the tool enables teachers to monitor the activities performed by their students in a flipped class. With this functionality, teachers can view each student's performance and access individual records (logs) of their activities. This feature gives teachers an understanding of their student's performance and where they may need additional support. To illustrate this further, Figure 55 shows how teachers can monitor a student's records, keep path of a student's progress and identify areas where they may need additional support. For instance, Figure 56 shows a student's performance on each study guide card, providing teachers with insights into how their students perform on individual topics.

Figure 55 – Monitoring of a student's records by the teacher (in Portuguese)

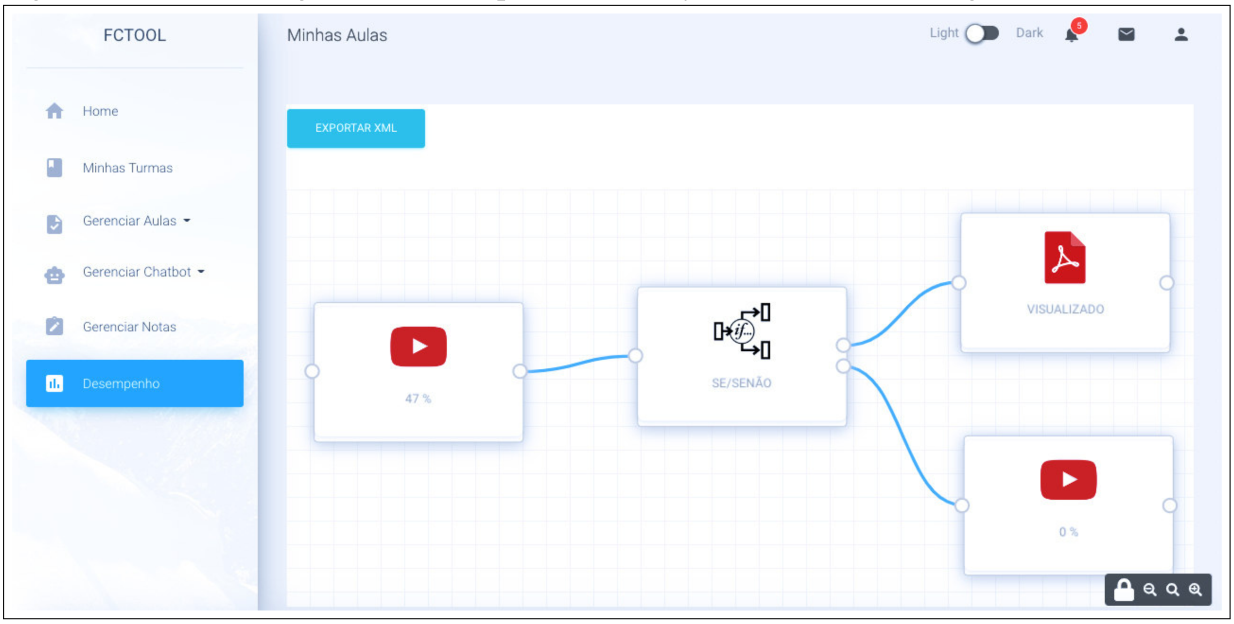


Source: Author

Students can access study guide instructional materials in the fourteenth feature (F14). Professors design these materials to give students course materials. The student can watch videos, read PDF files, answer forms, or be directed to external links within the tool. This feature is handy for those who learn better through visual or interactive means. To illustrate, Figure 57 provides an example of a student watching a video lesson.

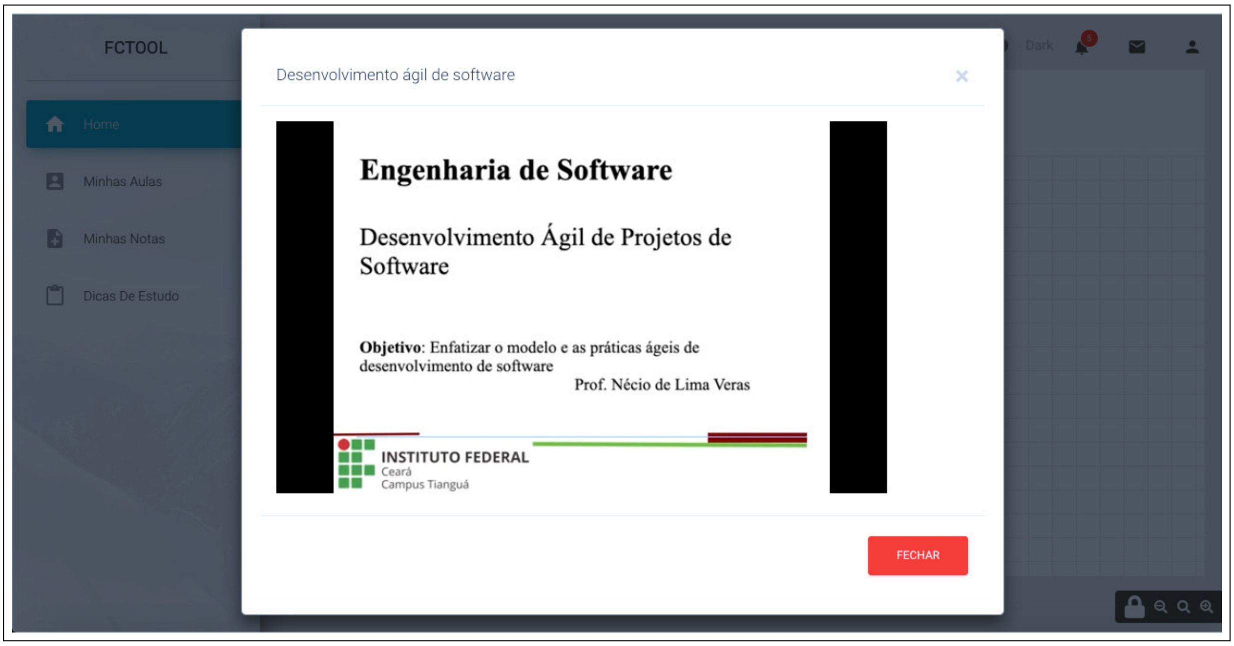
In the fifth feature (F15), students can access an overview of their progress for each study guide activity. This feature enables students to review their performance statistics for a flipped lesson, including (a) the list of activities that have been completed and the ones that are yet to be done, (b) scores per exercise, (c) time spent on each activity, (d) content watched,

Figure 56 – Monitoring of a student's performance by the teacher (in Portuguese)



Source: Author

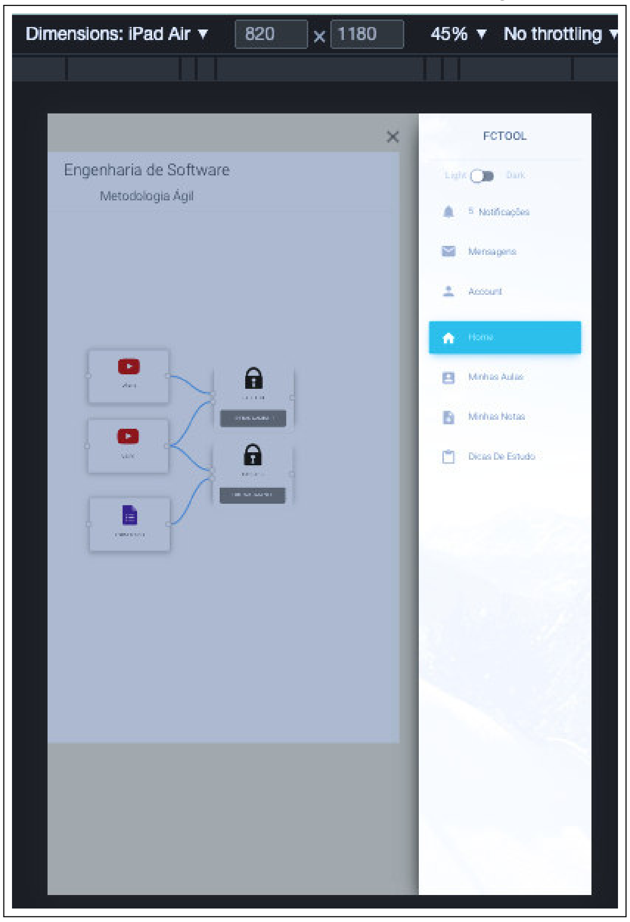
Figure 57 – Student watching video lesson (in Portuguese)



Source: Author

and (e) forms answered. This information helps students to identify areas where they need to improve and make necessary adjustments to their learning process. Moving on to feature sixteen (F16), FCTool allows students to perform study guide activities. To monitor the context of their particular studies, the student can get an overview of their progress for each study guide's content. This overview is similar to the one shown in Figure 56, and it helps students follow their progress and identify areas where they need to put in more effort. By having access to this feature, students can stay on track with their studies and ensure that they are making progress toward their goals. The tool's seventeenth feature (F17) pertains to its usability, specifically tool responsiveness. This feature means the tool is responsive and can be accessed from a mobile device while the student studies. To provide a better understanding, a simulation of accessing the tool using a mobile device, an iPad Air, is illustrated in Figure 58.

Figure 58 – A simulation of using the tool from a mobile device (in Portuguese)



Source: Author

We made two videos demonstrating the usage and monitoring from two perspectives to offer a more comprehensive version of the tool. Firstly, a video showcases the tool's usage from the student's perspective. This video offers valuable insights into how students can make the most of the tool and how it can assist them in their studies. This video is available at the following URL: <https://youtu.be/54tsr8d1yY8>. Secondly, another video illustrates student monitoring from the teacher's viewpoint. This video demonstrates how teachers can use the tool to monitor their students' progress and ensure they are on track. This second video is accessible at the following URL: <https://youtu.be/vR9Zx3tESB8>.

5.4 Chapter summary

This chapter introduces a web-based authoring tool that is intended to assist in creating personalized study guides. The chapter explains our past experiences with the flipped classroom method and its relation to the tool design. Furthermore, the chapter provides a detailed description of how the model was **partially integrated** into the tool. It lists each model element with its corresponding implementation in the tool. FCtool has four instructional contents based on types presented by the model (Reading, Multimedia, and Score-based) and all five personalization points. However, the chapter highlights the tool's limitations, which lies, mainly, in how quantitative and qualitative rules were encoded. While the rules are unlimited in the model, the tool restricts them to two rules in conjunction and disjunction and one in if/else and optional. Furthermore, we implemented into the tool only a subset of the design principles presented in Chapter 3.

In this chapter, we have included a study guide we created using a tool. We used the same example for the software requirements lesson in the previous chapter. The chapter also covers seventeen features and presents a software prototype as a minimally viable product. Each feature is explained in detail with its respective graphical interface. The chapter also includes two videos demonstrating how to use the tool. The first video shows a student using the tool, while the second video shows a teacher observing the data generated by the FCTool usage in the first video. The next chapter, Chapter 6, focuses on evaluating the the FCtool.

6 EVALUATION

This chapter aims to address our last two research questions: **RQ4** (*What are the SE professors' perceived usefulness and ease of use of the Flipped Classroom Tool?*) and **RQ5** (*What is the impact of Personalized Study Guides on students' perception, motivation, engagement, and learning gain during software engineering flipped classes?*).

We organized the evaluation into three phases. In the first phase, we interviewed software engineering professors. We introduced to them our tool to their views on FCTool and its potential usefulness. Moving to the second phase, our focus shifted to an empirical assessment of our approach's effectiveness. We adopt our tool and FC approach in a software engineering course, enrolling twenty-two students. The goal was to observe effects on the learning process and outcomes. In the third phase, we conducted a randomized controlled experiment. We planned the experiment based on software engineering flipped classes, one group with personalized study guides and the other with traditional flipped classes.

In short, the chapter was structured as follows. In Section 6.1, we present the evaluation with software engineering professors. Following that, Section 6.2 details the assessment with one-group software engineering students and Section 6.3 describes a randomized controlled experiment. Next, the Section 6.4 showcases the discussion. Limitations and Threats to validity are discussed in Section 6.5, and the concluding remarks are presented in Section 6.6.

6.1 Evaluation 1: Software engineering professors

This section presents the evaluation conducted with Software Engineering professors to assess the acceptance levels of our authoring tool for personalized study guides in flipped classrooms. Inspired by Kim *et al.* (2014), we created a protocol for the interviews (Section 6.1.3).

The interview process was structured into a three-stage flow to ensure comprehensive data collection and understanding:

- **Understanding the Professor's Experiences**. In this initial stage, we focused on gathering detailed information about the professors' backgrounds and experiences;
- **Introducing the Research and Tool Demonstration**. The second stage was dedicated to providing the professors with a thorough understanding of our research goals and how FC Tool works;

- **Collecting Tool Acceptance**. In the final stage, the professors completed a validated instrument to attest how much they accepted or liked the tool.

6.1.1 Participants

We interviewed **seven** professors from five public universities and one federal institute, representing five Brazilian states (Amazonas, Ceará, Minas Gerais, Santa Catarina, and São Paulo). The interview participants were Software Engineering professors with at least eight years of experience teaching Software Engineering. The sample was composed using a non-probabilistic, convenience sampling method within the specific profile of the group. Teacher participation was voluntary and determined through invitations sent via email and through research groups in SE on the WhatsApp application.

Among the interviewees, one professor has extensive experience with flipped classes, having conducted over 100 flipped SE classes in the last five years (since 2018). Another has been using the flipped classroom method since 2021 and has already conducted approximately 20 flipped classes. A third professor used a variation of the method during the pandemic period (remote teaching) between 2020 and 2021, having conducted between 10 and 20 (partially) flipped SE classes. A fourth professor used the blended FC method with gamification in more than 15 flipped classes. The fifth professor started testing the FC method in early 2023, blending it with hybrid approaches, such as hackathons and problem-based learning. Finally, two professors have never used the FC method in the classroom but are theoretically familiar with it.

The interviews occurred between September twenty-ninth and October thirty-first, 2023, in a remote format using the Google Meet tool.

6.1.2 Materials

We collected qualitative data through interviews and quantitative data through an electronic form[1]. Inspired by Hu *et al.* (1999), the electronic instrument followed the Technology Acceptance Model (TAM) (DAVIS, 1985) with responses using the Likert scale (LIKERT, 1932) in five levels of acceptance: Strongly Agree, Agree, Neutral, Disagree, Strongly Disagree. These responses were correspondingly mapped to 5 points, 4 points, 3 points, 2 points, and 1 point, respectively. We evaluated professors' acceptance using a set of twenty questions listed in Table 17. The questions were organized into four dimensions: Perceived usefulness (PU), Perceived

[1] Available in Portuguese at: <https://forms.gle/cMAUyRGtAJZ3Eecg7>

ease of use (PEOU), Attitude, and Intention to use (ITU). This structuring allows for assessing professors' acceptance of the tool, covering key aspects. "Perceived usefulness" assesses the extent to which professors believe the tool enhances teaching effectiveness, while "Perceived ease of use" evaluates the tool's perceived simplicity and ease of use. "Attitude" measures the general feelings (positive or negative) towards the tool, and "Intention to use" gauges the willingness of professors to integrate the tool into their future teaching activities. The multidimensional items allow an understanding of the factors influencing professors' acceptance of the tool, providing insights about their perceptions towards adopting technology in their educational environments.

Table 17 – Instrument (questions) for professors' acceptance assessment

ID	Question	Dimension measured
A1	Using FCTool technology can allow me to create a study guide more quickly	Perceived usefulness (PU)
A2	Using FCTool technology CANNOT improve my monitoring and management outside the classroom	
A3	Using FCTool technology can increase my productivity in flipped classes	
A4	The use of FCTool technology may NOT increase the effectiveness of my preparation for the flipped classroom	
A5	The use of FCTool technology can facilitate my management and my flipped classroom	
A6	I would NOT consider FCTool technology useful for my flipped classroom	
A7	Learning to operate FCTool technology would NOT be easy for me	Perceived ease of use (PEOU)
A8	I would find it easy to get FCTool technology to do what I need to do in my flipped classroom	
A9	My interaction with FCTool technology would be clear and understandable	
A10	I find FCTool technology INFLEXIBLE to interact with	
A11	It is NOT easy to become skillful in using FCTool technology	
A12	I would find FCTool technology easy to use	
A13	Using FCTool technology in flipped classes is a good idea	Attitude
A14	Using FCTool technology in flipped classes is UNPLEASANT	
A15	Using FCTool technology is beneficial for my flipped classroom and to the students	
A16	I intend to use FCTool technology in my flipped classes when it becomes available	Intention to use (ITU)
A17	I intend to use FCTool technology to provide a flipped classroom to students as often as needed	
A18	I intend NOT to use FCTool technology in my flipped classes routinely	
A19	Whenever possible, I intend NOT to use FCTool technology in my flipped classes	
A20	To the extent possible, I would use FCTool technology in my flipped classes frequently	

Source: Author (adapted from Hu *et al.* (1999))

As conducted by Hu *et al.* (1999), we balanced the questionnaire items, with nine out of the twenty items appropriately negated to draw the respondents' attention. As a result, they could become increasingly attentive to the positive question items. Additionally, we organized the questions randomly to reduce the potential ceiling (floor) effect that could induce monotonous responses to positive question items.

6.1.3 Protocol

We interviewed teachers to present our research and demonstrate the tool to gain a deeper understanding of their perceptions regarding the acceptance of the proposed technology. We used a semi-structured interview protocol applied consistently in interviews lasting approximately 60 minutes for each of the seven teachers. Inspired by Wohlin *et al.* (2012) recommendations, we created a protocol consisting of seven steps as follows:

Step 1 : At the beginning of the interview, the professors received the informed consent form via email or through the WhatsApp application (Available in Appendix B). Afterward, the first step followed this script:

"*Thank you for taking the time to speak with me today. This interview will take approximately 60 minutes to complete. You are participating in a study on the impact of the Flipped Classroom method with personalized study guides on teaching Software Engineering. This interview aims to investigate how the proposed method affects your perception regarding the use and acceptance of the FCTool technology. This interview will be used solely for this purpose and kept confidential. The interview will also be recorded.*"

To understand the teacher's experiences, the following questions were posed:

1. Comment on your experiences with the Flipped Classroom.
2. How long have you been using the method in Software Engineering teaching?
3. How many flipped classes have you conducted? Could you provide an estimate?
4. Do you employ any technology to support the implementation of the Flipped Classroom?
5. How have technologies been integrated into your flipped classroom activities?
6. How have specific technologies supported or hindered the implementation of your flipped classroom?
7. Can you monitor students' performance during home studies?
8. What are the main challenges of the method? And the main benefits?

Step 2 : In this step, we briefly explained the research context to the teachers, particularly the general idea of our personalized study guide model.

Step 3 : Subsequently, we introduced the tool and guided the interviewed professor in using the tool to create a personalized study guide. In this step, the professor accessed the platform through the link <https://nupreds.ifce.edu.br/fctool/> and shared their screen so that instructions on usage could be carried out.

Step 4 : Next, we switched the screen sharing to showcase real study guides applied in actual classes from the current semester. In this step, the professor could observe the outcomes of the model's application through the tool, especially concerning monitoring students' engagement and achievements on the platform.

Step 5 : At this point in the interview, we posed open-ended questions about the tool, allowing the professor to respond based on their perceptions during the interview. The questions were:

9. Did you find any difficulties in using the platform?
10. What are your impressions regarding the user interface?
11. Did you identify any advantages in using the tool?
12. And disadvantages?
13. Any additional comments? Feel free to share your thoughts.

Step 6 : Following that, we expressed gratitude for the cooperation and highlighted the sharing of some exciting ideas.

Step 7 : Finally, we concluded the interview and provided a channel for future communications. We emphasized that, in the end, the professor could respond to the online Technology Acceptance Model (TAM) survey (available at <https://forms.gle/tq8EVcZCzZmW8KUaA>).

6.1.4 Qualitative data results

Questions 1, 2, and 3 (Step 1) aimed to understand the profile of the interviewees (described in Section 6.1.1). The fourth question provided data on technologies used to support the FC method. Among the interviewees, two mentioned using the Telegram App[2] for messaging, three used tools from Google Suites, and one extensively used the Moodle[3] learning environment. One professor mentioned game tools (Kahoot[4], Quizzis[5], Classcraft[6]), and another used the

[2] <https://telegram.org/>
[3] <https://moodle.org/>
[4] <https://kahoot.com/>
[5] <https://quizizz.com/>
[6] <https://www.classcraft.com/>

university's virtual learning environment.

We observed initiatives to organize materials delivery to students through these tools and difficulties in integrating activities with flipped classrooms (question 5 - step 1). When we asked about how specific technologies helped prepare for flipped classrooms (question 6 - step 1), there were reports of excessive work to organize, although the results were positive. No professor mentioned feeling hindered by the tool used; however, the main complaint was difficulty monitoring home studies (question 7 - step 1). We heard reports of exaggerated efforts to monitor student progress during home studies. Three professors monitor the study material by delivering reports or activities. We noticed that these deliveries occurred during the class (question and answer sessions, for example) or on the eve of the practical class, making it challenging to replan the practical class or provide individualized feedback based on the student's progress.

In the last question of step 1 (question eight), four professors reported the following main challenges: (1) curation for material preparation; (2) content personalization; (3) sequencing and organization of materials for home study; (4) workload for correcting activities; (5) monitoring students during class preparation; and (6) the lack of a centralized information environment for students. Regarding benefits, we heard reports of better use of time in the classroom (practical classes), greater autonomy and flexibility for students to direct their studies, and better results from students with higher quality than previous classes.

After explaining the research (step 2), guiding the professors in using the tool (step 3), and presenting real-use cases of the tool (step 4), we gathered perceptions from the professors regarding FCtool (step 5). Firstly, we inquired about the platform's difficulties and impressions about the user interface (questions 9 and 10). Two professors expressed concerns about the students' difficulties using the tool and suggested including explanatory tips on each content card. One professor recommended more explanations for the teacher while creating the study guides. Four professors did not mention difficulties in using the tool and emphasized the tool's user-friendliness. In questions 11 and 12, we asked about the perceived advantages and disadvantages of using the tool, as follows:

a) **Advantages**: "The study guide organized in a workflow"; "Learning path is visually presented and its facilities for the teacher and the student"; "the organization is more efficient with the tool"; "guided orientation"; "efficient monitoring and tracking of home studies"; "the method of creating and assembling the study guide allows the teacher to better reflect on the students' path".

b) **Disadvantages**: Two professors mentioned possible disadvantages regarding the use of the tool: (1) "the tool be one more tool for the teacher to use", and, in this sense, suggested a reflection on "how would students adhere to the use of the tool?"; (2) "The curation and work involved in creating a guide with a logical sequence for students could be a drawback for teachers.".

In the last item of step 5 (question 13), we requested additional comments about the tool. We received the following feedback: "The way content personalization occurs was a great idea"; "The availability of the tool would be a 'dream' for teachers who use the flipped classroom method"; "I suggest including alerts to the teacher about low student performance and engagement"; "Include in the tool the VARK model of learning styles" (PINTRICH *et al.*, 1991); "Integrate the tool with the Moodle virtual learning environment."

6.1.5 Quantitative data results

We obtained quantitative data by responding to the twenty items in Table 17. The instrument applied the Technology Acceptance Model (TAM), which is organized into four dimensions (Perceived Usefulness, Perceived Ease of Use, Attitude, and Intention to Use). We obtained an average of 3.09 (standard deviation of 1.56) for Perceived Usefulness (A1 to A6) and 2.95 (standard deviation of 1.36) for Perceived Ease of Use (A7 to A12). Regarding Attitude (A13 to A15), we calculated a mean of 3.48 (standard deviation of 1.69); for Intention to Use (A16 to A20), the mean was 3.29 (standard deviation of 1.32). Table 18 textually presents each evaluated item's mean and standard deviation, and Figure 59 graphically shows the professors' perceptions.

The figure illustrates an average acceptance rate of 93.63% for positive items (A1, A3, A5, A8, A9, A12, A13, A15, A16, A17, A20) and an average rejection rate of 81.11% for negative items (A2, A4, A6, A7, A10, A11, A14, A18, A19). The result suggests a positive acceptance of the proposed technology (above 88%), adhering to the balance between positive and negative items.

Examining each of the four evaluated dimensions individually, "Attitude" was the dimension with the highest acceptance (100% for both positive items and an 86% rejection rate for the negative item). This dimension helped measure the level of acceptance the teacher had in adopting or not adopting the technology. Thus, there is an expectation that the use of the

Table 18 – Technology Acceptance data

	Mean	Standard deviation
A1	4.28	1.11
A2	1.57	0.79
A3	4.28	0.49
A4	2.14	0.90
A5	4.71	0.49
A6	1.57	0.79
Perceived usefulness	**3.09**	**1.56**
A7	2.00	1.15
A8	4.14	0.69
A9	3.86	0.90
A10	1.57	0.79
A11	2.00	0.58
A12	4.14	0.69
Perceived ease of to use	**2.95**	**1.36**
A13	4.57	0.53
A14	1.28	0.76
A15	4.57	0.53
Attitude	**3.48**	**1.69**
A16	4.14	0.69
A17	4.28	0.49
A18	1.86	0.69
A19	1.86	0.69
A20	4.29	0.49
Intention to use	**3.29**	**1.32**

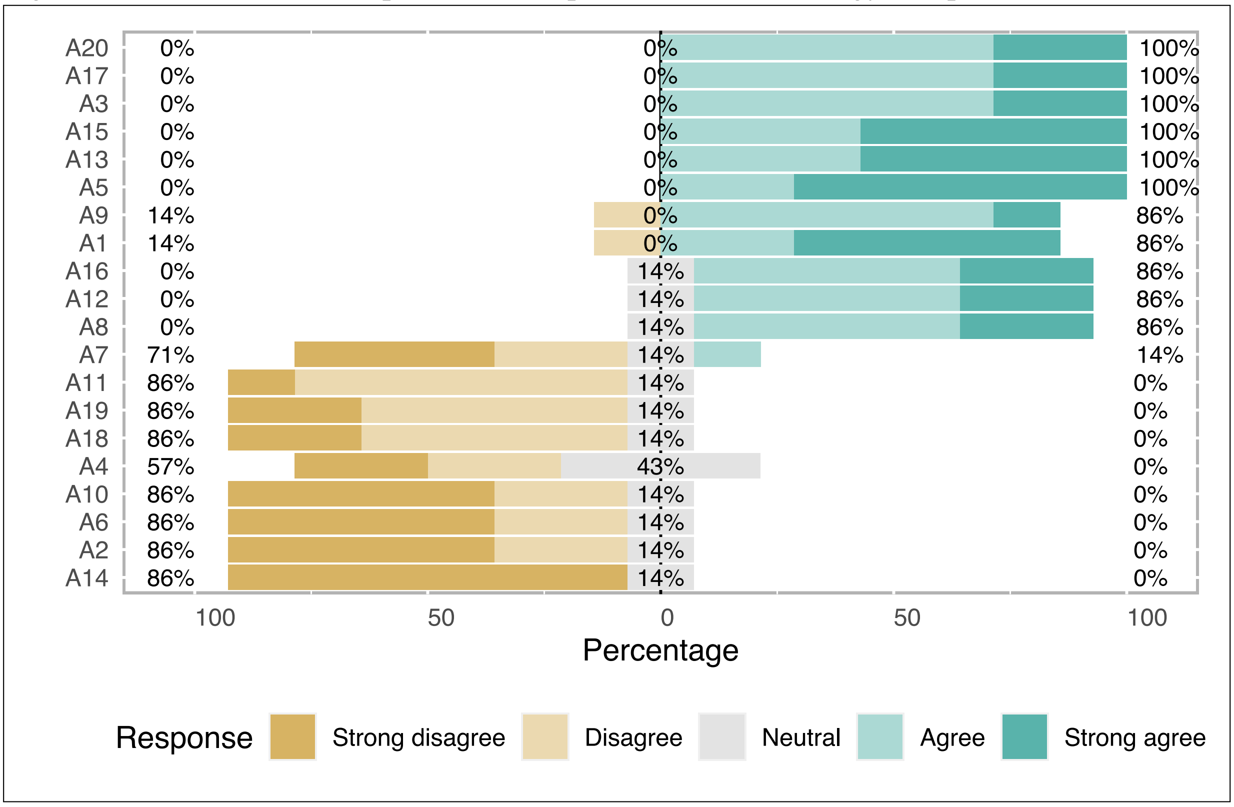
Source: Author

proposed technology can be enjoyable, beneficial, and appealing. The second dimension best accepted by teachers was “Intention to use” (95.3% positive acceptance and 86% rejection). This result suggests a strong desire to use the proposed technology. Next, the “Perceived Usefulness” dimension achieved an acceptance rate of 95.3% for positive items and a rejection of 76.3% for negative items. This dimension helped assess the teachers’ belief that using the technology could bring benefits and improve their outcomes; that is, by accepting the technology, they perceived it as applicable and offering tangible advantages. Finally, the “Perceived Ease of Use” dimension achieved an 86% acceptance rate for positive items and an 81.0% rejection for negative items. This dimension helped evaluate the teacher’s belief that the proposed technology is easy to learn, use, and integrate into their daily activities. The technology was perceived as easy to use, regardless of the teacher’s experience, making it more likely to be accepted by other users.

6.2 Evaluation 2: One-group software engineering students

This evaluation 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

Figure 59 – Likert data from professor's responses about technology acceptance



Source: Author

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 (MASON *et al.*, 2013; MAHASNEH, 2020; HAO, 2016; LIN *et al.*, 2019; HWANG *et al.*, 2022; MCLAUGHLIN *et al.*, 2013; SCHULTZ *et al.*, 2014).

We measured the impacts through students' perceptions (KIM *et al.*, 2014) (cognitive presence, learner presence, and technology use), motivation (PINTRICH *et al.*, 1991) (control of learning beliefs, self-efficacy for learning and performance, test anxiety), and achievements (engagement and score tests) during lessons. We were inspired by Schultz *et al.* (2014), which sought to investigate the impact of flipped classroom on students' performance. Also, our evaluation followed Ryan and Reid (2016), which reported a study about the impact of the flipped classroom on student performance and retention.

In our evaluation, we examined the impacts of the personalized flipped classroom in SE teaching. Three dimensions were simultaneously observed during proposed activities for students: students' (1) perception, (2) motivations, and (3) achievements.

a) **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 (KIM *et al.*, 2014).

b) **Students' Motivation:** refers to three components of motivation, such as (1) control of learning beliefs, (2) self-efficacy for learning and performance, and (3) test anxiety (PINTRICH *et al.*, 1991). 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.

c) **Students' Achievements:** refer to flipped class tool impacts on student engagement and learning (HWANG *et al.*, 2022). The measure will characterize the students' performance during flipped class activities.

6.2.1 Participants

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) adopting the **Personalized Study Guide Model (PSG-model)**. It is essential to note that all participants were beginners with no prior exposure to Software Engineering concepts.

6.2.2 Materials

We used the tool (Chapter 5) to create the study guide and implemented it during a software requirement lesson. The study guide employed adaptive learning techniques through a personalized study guide. 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 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). All the details of the software requirements lesson, the instructional contents, exercises, practical activities, and the study guide were outlined in Section

4.4.2.

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 (LIKERT, 1932). In addition, the studies by Kim *et al.* (2014), Pintrich *et al.* (1991) 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 listed in Table 19. 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 to 5 points, 4 points, 3 points, 2 points, and 1 point, respectively.

Similarly to the 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 presented by Table 20.

Additionally, we developed a consent form (Appendix A) to clarify the roles of the students in the evaluation process. Before 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.

Evaluating student engagement involved assessing their performance through formative and summative activities. The methods employed to collect information about student achievements encompass active participation, the outcomes generated, and the grades earned by the students. Furthermore, we designed two tests (pre- and post-tests) 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 students' prior understanding of the subject matter before the experiment commenced. The pre-test electronic form, available at <https://forms.gle/CVj2Gk5pJSZiJwbD8>, contains the

Table 19 – Instrument (questions) for student's perception assessment

ID	Question	Perception measured
Q1	Problems posed increased my interest in SE issues	Cognitive presence
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	
Q9	When I studied for the SE activities, I set goals for myself in order to direct my activities in each study period	Learner presence
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	
Q17	It was easy for me to find and access the out-of-class materials associated with flipped classroom activities in FCTool	Technology use
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 FCTool technology can increase my study productivity	
Q21	Using FCTool technology in flipped classrooms is a good idea	
Q22	I would like to use the FCTool in other flipped classes whenever possible	

Source: Author (adapted from Kim *et al.* (2014))

consent form, questions about motivation, and items about software requirements. The student must accept the terms to participate in the study. The post-test form contains questions about perception and motivation after the study and items about software requirements (available at <https://forms.gle/39w7Udp68gWKFnGA9>). The test results served as a baseline measure, enabling the quantification of how much knowledge students had acquired and retained throughout the evaluation.

6.2.3 Procedure

Before starting the flipped class, the teacher informed the students about the teaching model used for SE education. After that, the students answered the pre-test questionnaire

Table 20 – Instrument (questions) for student's motivation assessment

ID	Question	Motivation measured
Q23	If I study in appropriate ways, then I will be able to learn the material about SE	Control of learning beliefs
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	
Q27	I believe I will receive an excellent grade in this flipped class	Self-efficacy for learning and performance
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	
Q35	When I take a test I think about how poorly I am doing compared with other students	Test anxiety
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	

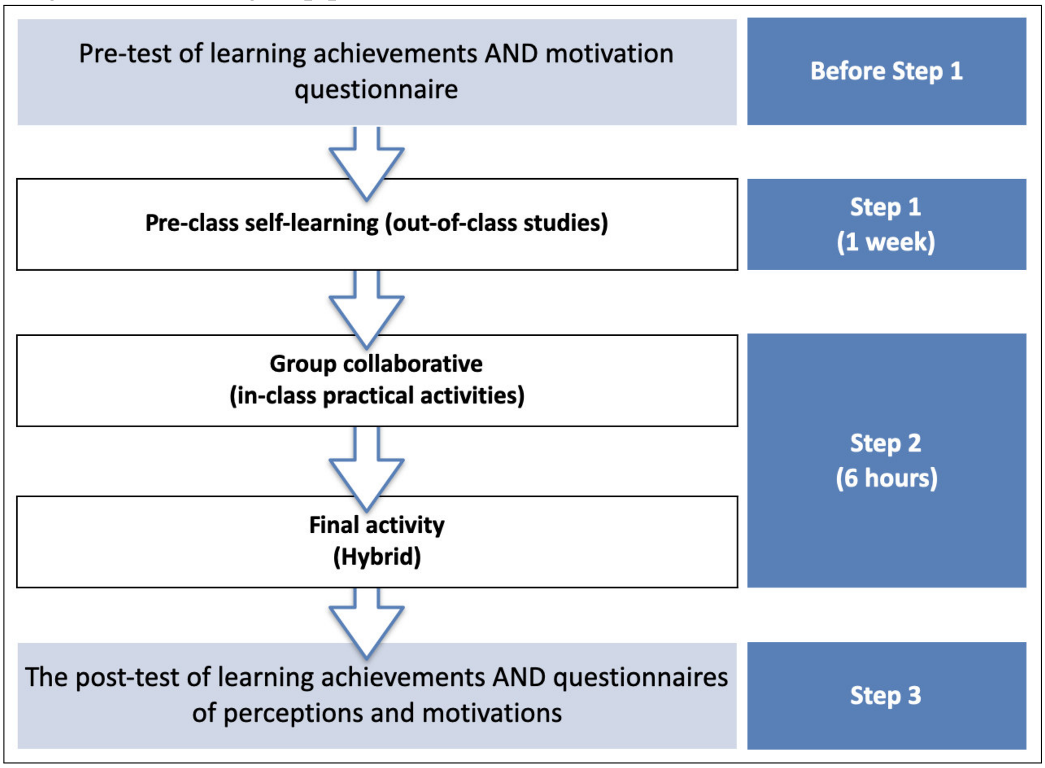
Source: Author (adapted from Pintrich *et al.* (1991))

(motivation assessment and items about software requirements). Students answered the questions individually, without consulting additional knowledge sources (e.g., the Internet and books). The questionnaires were explained to students as part of the experiment. Then, the students were required to complete the activities, although their scores would not impact their course grades. Figure 60 presents the one-group evaluation process inspired by Srichailard (2023).

6.2.4 *Student's perception analysis*

We obtained an average acceptance rate of 85.25% for cognitive presence (from Q1 to Q8). 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 the study guide. Regarding learner presence (from Q9 to Q16), 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.

Figure 60 – One-group process



Source: Author (adapted from Hwang *et al.* (2020)

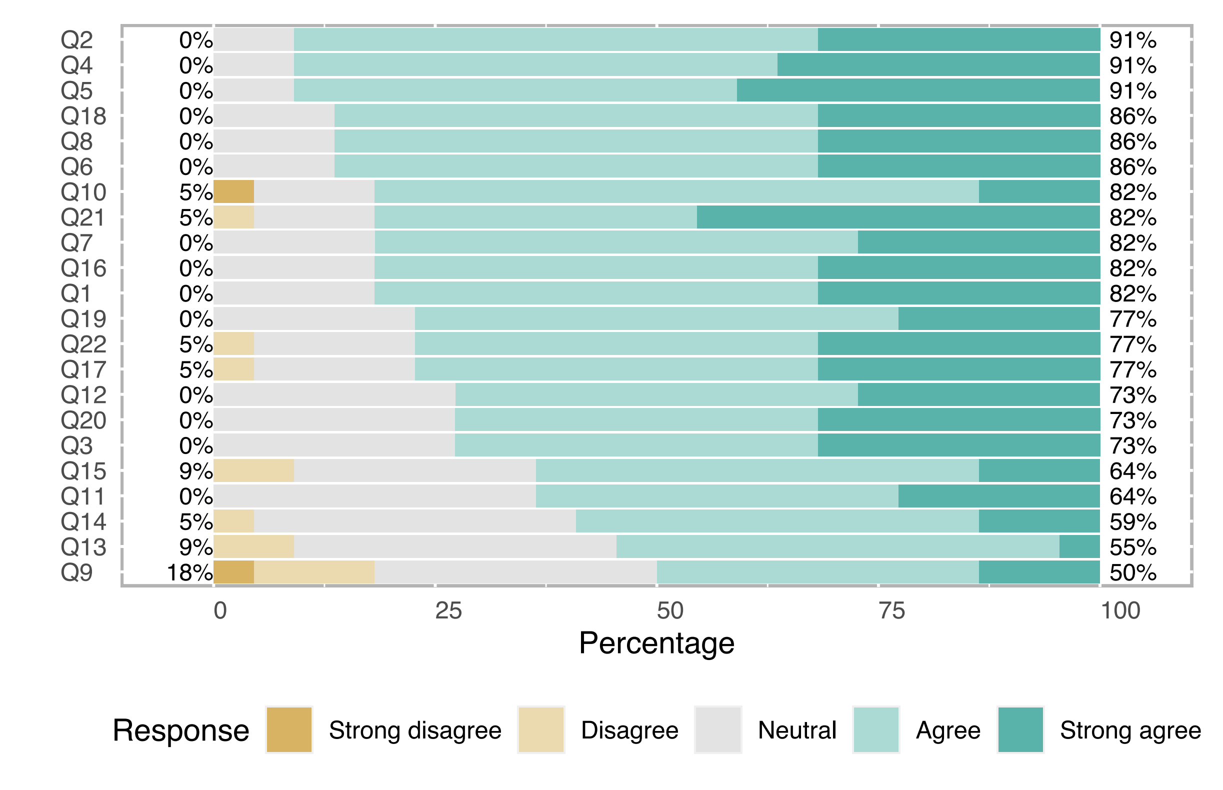
Regarding the technology acceptance rate (from Q17 to Q22), we had an average acceptance rate of 78.66%. The lowest-rated item in this dimension was Q20 with a 73% acceptance rate. The high overall acceptance indicates that students recognized the value and utility of the flipped classroom model in optimizing their study processes. The results about students' perception are presented graphically in Figure 61.

The average disagreement rate for student perception was 2.77%, while the agreement rate was 76.5% (20.73% neutral). The three highest agreement indices observed were items Q2, Q4, and Q5, representing (a) motivation to explore the contents of the flipped classroom, (b) alignment of instructional materials with the contents of theoretical activities, and (c) coherence between the new information learned during home studies and the practical activities conducted.

6.2.5 Student's motivation analysis

We gathered responses from the students before and after the flipped classroom to assess the model's impact on their motivation. The compiled student motivation responses in Table 21 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 bold P-values

Figure 61 – One-group students perception's data



Source: Author

offer evidence to support the conclusion that the data exhibit statistical dissimilarity.

Table 21 – One-group motivation's data

	Pre-class			**Post**-class			**Difference**
	Mean	SD[1]	Rate(%)	Mean	SD[1]	Rate(%)	**p-value**
Q23	4.27	0.55	95.5	3.95	0.72	72.7	0.9969
Q24	3.09	1.19	36.4	3.41	0.85	45.5	**0.0117**
Q25	4.09	0.81	81.8	3.77	0.61	68.2	0.9918
Q26	3.41	1.05	45.5	3.28	1.02	45.5	0.9599
	Average(%)		64.8	Average(%)		58.0	0.9765
Q27	3.68	0.57	63.6	3.63	0.79	54.5	0.7245
Q28	3.45	0.86	40.9	3.68	0.84	54.5	**0.0184**
Q29	4.27	0.70	86.4	4.09	0.81	72.7	0.9861
Q30	3.64	0.95	50.0	3.59	0.79	50.0	0.8068
Q31	3.77	0.75	68.2	3.41	0.66	40.9	0.9982
Q32	4.36	0.58	95.5	3.68	0.64	68.2	1.0000
Q33	3.91	0.68	72.7	3.68	0.78	59.1	0.9915
Q34	3.68	0.72	54.5	3.91	0.75	77.3	**0.0363**
	Average(%)		66.5	Average(%)		59.7	0.9996
Q35	3.14	1.24	40.9	2.95	1.09	31.8	**0.0359**
Q36	3.68	0.99	63.6	3.41	1.00	54.5	**0.0098**
Q37	3.68	1.21	72.7	2.81	1.18	36.4	**0.0002**
Q38	3.23	1.31	45.5	2.32	0.78	4.50	**0.0001**
Q39	3.09	1.23	45.5	2.36	0.90	9.10	**3.585e-05**
	Average(%)		53.6	Average(%)		27.3	**4.643e-13**

[1]Standard deviation.

Source: Author

Regarding control of learning beliefs (from Q23 to Q26), we observed a significant increase in item Q24. This growth suggests a shift in students' perception of responsibility regarding Software Engineering (SE) learning. The model might have fostered a sense of empowerment among students, making them feel more capable and responsible for their learning—a heightened belief in their ability to learn and succeed in the specific domain of Software Engineering. Concerning self-efficacy for learning and performance (from Q27 to Q34), we noted increases in items Q28 and Q34, supporting the Q24 result and indicating students' self-awareness of their shortcomings in efforts during out-of-class preparation. However, the most significant outcome was related to test anxiety (from Q35 to Q39). All analyzed items showed a substantial decrease in student anxiety. This notable reduction suggests that the model can alleviate students' fears and insecurities, reducing stress and anxiety levels during tests.

6.2.6 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. 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 22 presents textually the students' data regarding the study guide presented by Figure 33.

Figure 62 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.

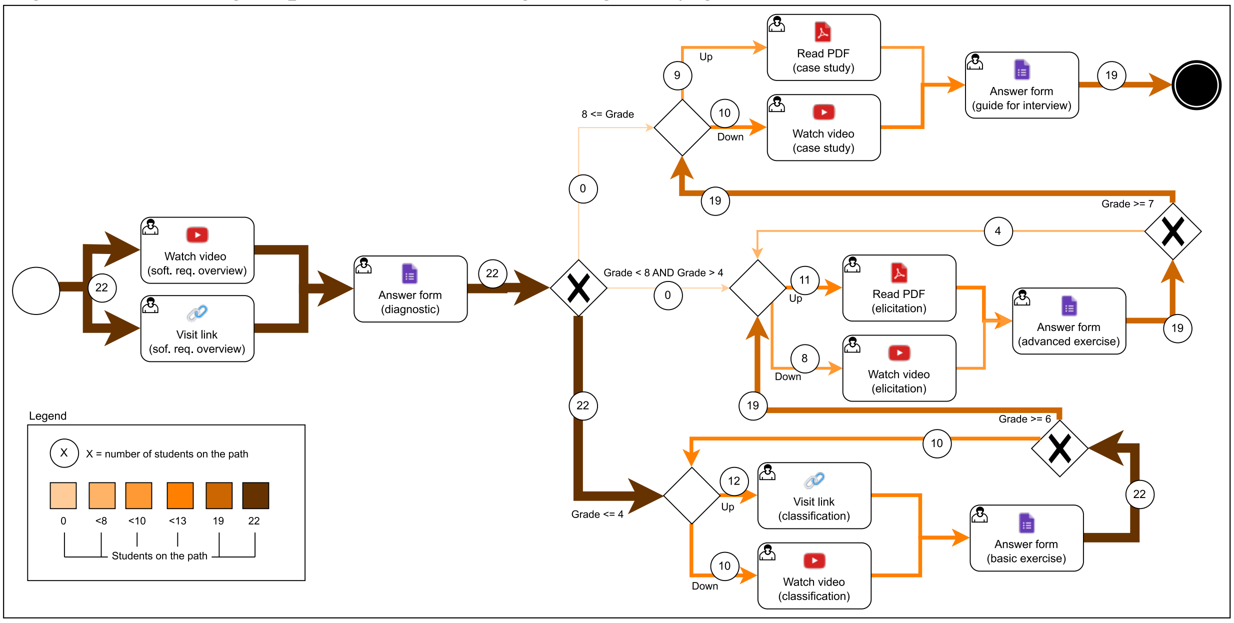
Moreover, the data presented in Figure 62 elucidates the students' transition from the "Middle grade" to the "High grade" profile. Out of the initial nineteen students in the "Middle grade" category, all successfully progressed, indicating a robust consolidation of knowledge and skills, as measured through the advanced exercise. Additionally, the four students who encountered challenges in advancing to the "High grade" profile revisited the instructional content and repeated the advanced exercise, manifesting the equitable nature of the study guide. We also noted among the initial cohort in the "Low grade" profile, three students did not progress to the

Table 22 – One-group students' engagement data

ID	$video_1$ score	$link_1$ engagement	**Diagnostic grade**	**Profile**	**Learning path engagement (%)**	**Grade average**
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

Source: Author

Figure 62 – One-group student's data regarding study guide

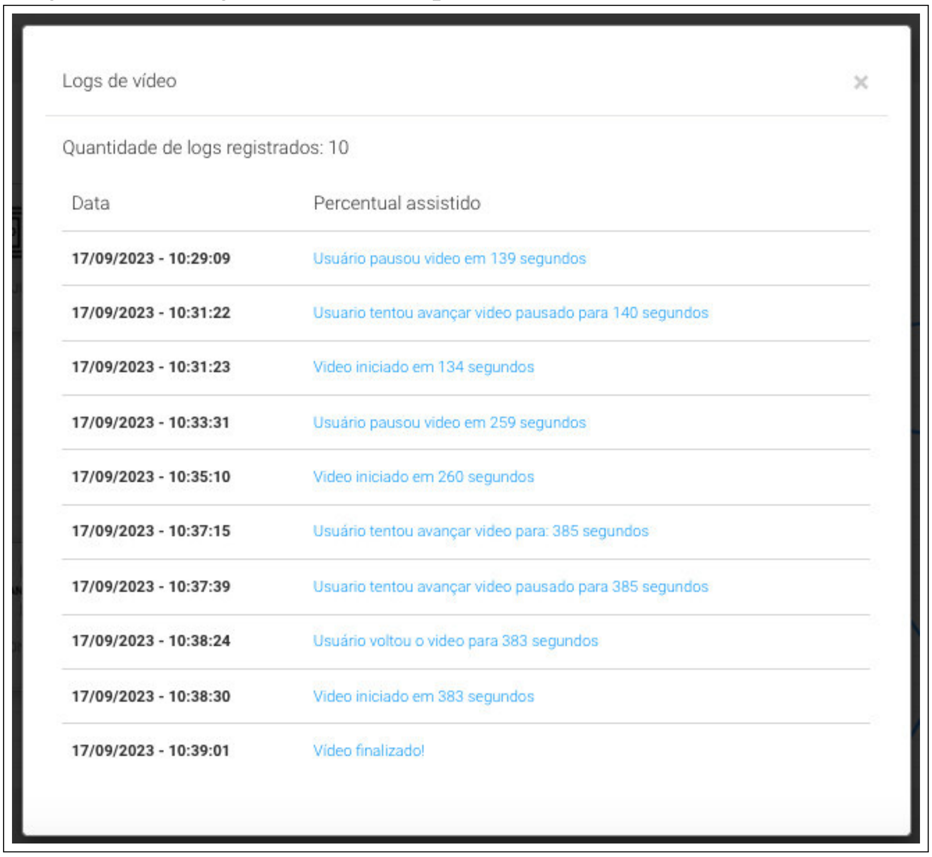


Source: Author

"Middle grade" profile despite revisiting the instructional material. In retracing their learning path within one of the profiles, the students qualify to demonstrate the voluntary and self-directed aspects of the adaptive study guide. This detailed exposition underscores the effectiveness of the personalized study guide in accommodating individual learning paces and adapting to the unique needs of each student. The transition across different proficiency levels emphasizes the model's capacity to cater to students with diverse learning requirements, fostering a targeted learning experience during software engineering studies.

We used student S20 as an example to illustrate the student's study pace. In the "Low grade" profile, the student-initiated their study on the guide at 10:29 on (September) 17th, choosing to watch a video lecture. The logs of learning monitoring (Figure 63) for the student indicate some pauses during the study, concluding at 10:39. These pauses suggest that the student engages in meticulous study, possibly making notes or jotting down content from the video. This engagement detail demonstrates the student's dedication to understanding the material.

Figure 63 – Logs of video (low profile) from student S20



Logs de vídeo

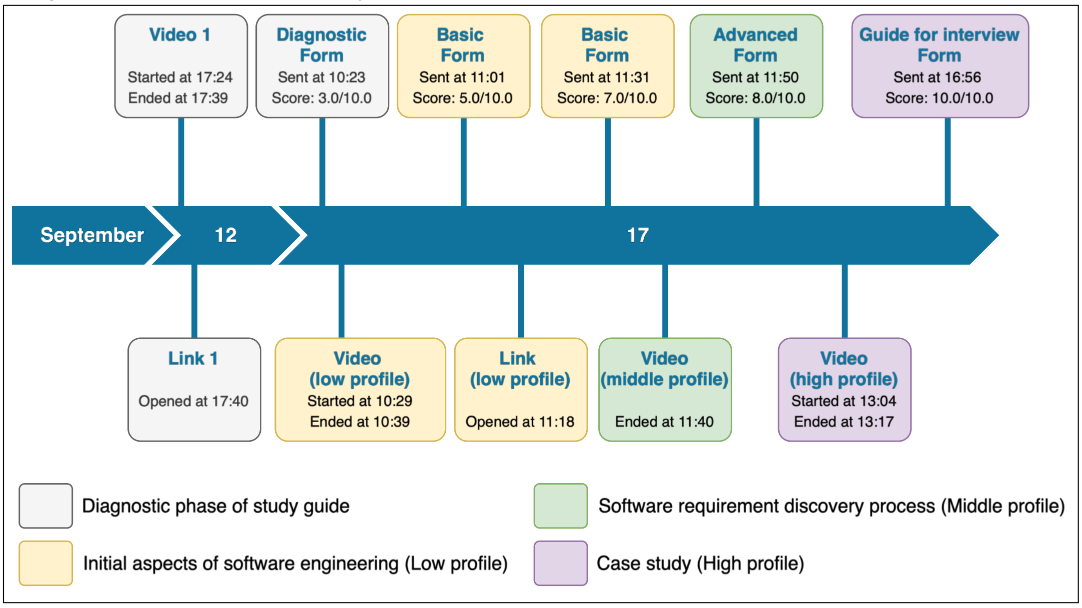
Quantidade de logs registrados: 10

Data	Percentual assistido
17/09/2023 - 10:29:09	Usuário pausou video em 139 segundos
17/09/2023 - 10:31:22	Usuario tentou avançar video pausado para 140 segundos
17/09/2023 - 10:31:23	Video iniciado em 134 segundos
17/09/2023 - 10:33:31	Usuário pausou video em 259 segundos
17/09/2023 - 10:35:10	Video iniciado em 260 segundos
17/09/2023 - 10:37:15	Usuário tentou avançar video para: 385 segundos
17/09/2023 - 10:37:39	Usuario tentou avançar video pausado para 385 segundos
17/09/2023 - 10:38:24	Usuário voltou o video para 383 segundos
17/09/2023 - 10:38:30	Video iniciado em 383 segundos
17/09/2023 - 10:39:01	Video finalizado!

Source: Author

The pauses observed during the study session, particularly in reviewing video content, align with active learning strategies in which students take responsibility for their learning. This behavior reflects a deliberate approach to the learning process, indicating a desire to absorb and internalize the presented information. The student's study pace provides insights into the differentiated learning styles and preferences the personalized study guide accommodates. Figure 64 shows the study timeline for the exemplified student (S20).

Figure 64 – Timeline study of student S20



Source: Author

Notably, at 10:40, the student started the basic exercise and submitted their response at 11:01, obtaining a grade of 5.0. Consequently, the student backtracked and altered their choice of study material to use an external resource (link) at 11:18. After studying using the recommended content reading, the student submitted a new response to the exercise at 11:31 and achieved a new grade of 7.0. This new grade allowed unlocking the learning path to advance to the “Middle grade" profile.

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 65 depicts the scores in both tests.

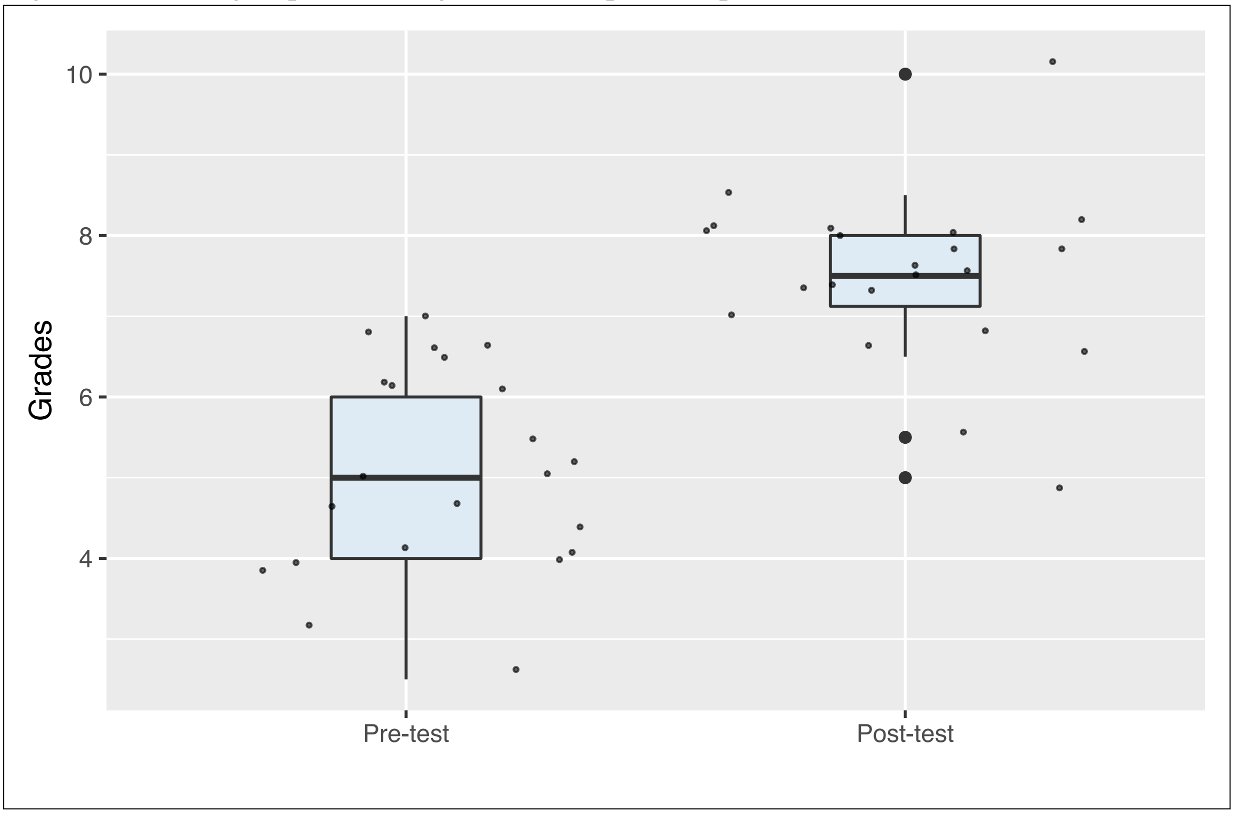
After collecting the scores for the two tests, we used the independent samples t-test as an analysis method. The analysis results are shown in Table 23. The results showed that the students obtained significant differences in learning (t = -10.58, p < 0.05). Based on the results of the analysis, it is suggested that using personalized teaching scripts during a flipped classroom improves students’ cognitive load.

Table 23 – T-test results of learning performance

Test	**N**	**Mean**	**SD**	t	p
Pre-test	22	5.04	1.27	-10.258	$6.195e-10 < 0.05$
Post-test		7.50	1.02		

Source: Author

Figure 65 – One-group students' grades in the pre- and post-tests



Source: Author

6.3 Evaluation 3: Randomized controlled experiment

This section details the evaluation of our learning model using a randomized controlled experiment. The experiment was inspired by Hwang *et al.* (2020) and adhered to the recommendations outlined in Wohlin *et al.* (2012). Similar to the one-group evaluation described in Section 6.2, we measured the effects of the personalized study guide model on SE flipped classes from the student's perspective through two groups. One group employed the personalized study guide model, while the other followed the non-personalized study guide model. We used the exact dimensions as in Section 6.2 during the experiment (perception, motivation, and achievements).

6.3.1 Participants

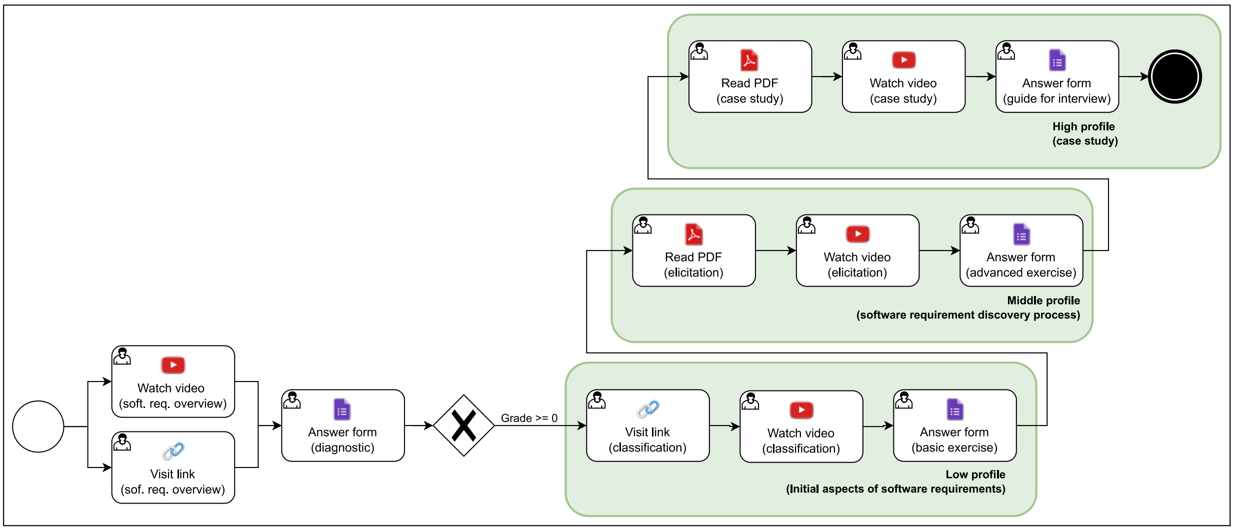
Data collection for this experiment also happened in September 2023. Participants were bachelor's degrees in computer science at the Federal Institute of Ceara, Brazil. The evaluation comprised 26 computer science students divided into two groups: an experimental group and a control group. The experimental group, consisting of 13 students, adopted the

Personalized Study Guide Model (PSG-model). The control group, consisting of 13 students, utilized the **Non-Personalized Study Guide Model (SG-model)**. The same instructor taught both groups (the author of this thesis). It is essential to note that all participants were beginners with no prior exposure to Software Engineering concepts.

6.3.2 Materials

We utilized the tool (Chapter 5) to build two study guides, implementing them during a software requirements lesson. One study guide incorporated adaptive learning techniques through the PSG-model (personalized), while the other utilized the SG-model (non-personalized). We employed identical activities and materials described in Section 6.2 for both study guides. The personalized study guide model used was the same as detailed in Section 4.4.2. The non-personalized study guide model delivered the same content as the PSG-model, but the learning path was built linearly. Figure 66 shows the non-personalized study guide following the model proposed in Chapter 4.

Figure 66 – Study guide flipped lesson using SG-model



Source: Author

At the outset of the study guide, the student must watch a video lecture and access an external resource (link) for reading. Upon completing these tasks, the student must respond to the diagnostic form and begin their out-of-class study starting from the "Low grade" profile (irrespective of their score in the diagnostic form). The entire learning path is available to the student, and instructional materials are presented sequentially. The student has the flexibility to choose how to explore the study materials. The data collection for evaluating students utilized the same instruments outlined in Section 6.2.2, including identical pre- and post-test forms.

6.3.3 Hypotheses definition

The implementation of this flipped classroom aimed to conduct a controlled comparison between personalized and non-personalized guides. The instructional content and activities in both guides remained consistent, ensuring a fair assessment of the impact of personalization on student outcomes. We systematically recorded students' progress via the study guides, their interactions, and responses, enabling a comprehensive analysis of the effectiveness of each model. We designed the personalized study guide (PSG-model) to adapt to individual learning preferences and pace. They were incorporating adaptive learning techniques aimed to enhance the overall learning experience by tailoring content delivery to the specific needs of each student.

On the other hand, the non-personalized study guide (SG-model) represents a more traditional and linear approach to content presentation. This difference allowed for a comparison of the impact of personalization on student engagement, perception, motivation, and achievement in the flipped classroom context. We examined the controlled experiment research questions organized in Table 24 to compare the two models.

Table 24 – Controlled experiment research questions

ID	Question	Rationale
CE-RQ1	To what extent does PSG-model improve cognitive presence compared to SG-model?	Identify critical and creative thinking presence.
CE-RQ2	To what extent does PSG-model increase learner presence compared to SG-model?	Assess self- and co-regulation learning.
CE-RQ3	To what extent does PSG-model increase learners' technology acceptance compared to SG-model?	Assess students' acceptance regarding FC-Tool uses.
CE-RQ4	To what extent does PSG-model increase learners' control of learning beliefs compared to SG-model?	Assess if students' efforts to learn result in positive outcomes.
CE-RQ5	To what extent does PSG-model improve learners' self-efficacy compared to SG-model?	Measure students' self-efficacy for learning and performance.
CE-RQ6	To what extent does PSG-model reduce learners' anxiety compared to SG-model?	Examine students' anxiety regarding tests performed.
CE-RQ7	To what extent does PSG-model improve learners' engagement in out-of-class activities compared to SG-model?	Measure the students' engagement related to the method used.
CE-RQ8	To what extent does PSG-model increase learners' skills on the score tests compared to SG-model?	Determine if personalized flipped class impacts student learning about SE skills.

Source: Author

In seeking to answer the questions, the following hypotheses were defined:

- **Null hypothesis (H_0):** The impacts of students' perceptions, motivations, and achievements are less or equal in learners who study with the personalized study guide model

(PSG-model) than in those who study with the non-personalized study guide model (SG-model).

- **Alternative hypothesis (H_a):** The impacts of students' perceptions, motivations, and achievements are greater in students who study with the personalized study guide model (PSG-model) than those who study with the non-personalized study guide model (SG-model).

6.3.4 Variables selection

This experiment aims to investigate the particular values from one variable, the **impact of the flipped classroom** (such as Schultz *et al.* (2014), Ryan and Reid (2016) and Su and Chen (2018)). To understand the variable's values, some activities were taken to control the factors that directly or indirectly influence this variable. The variables that represent the influences and can affect the experiment results are named (**dependent variables**). The variables that accommodate the experiment results' values are called (**independent variables**). The impact is the dependent variable.

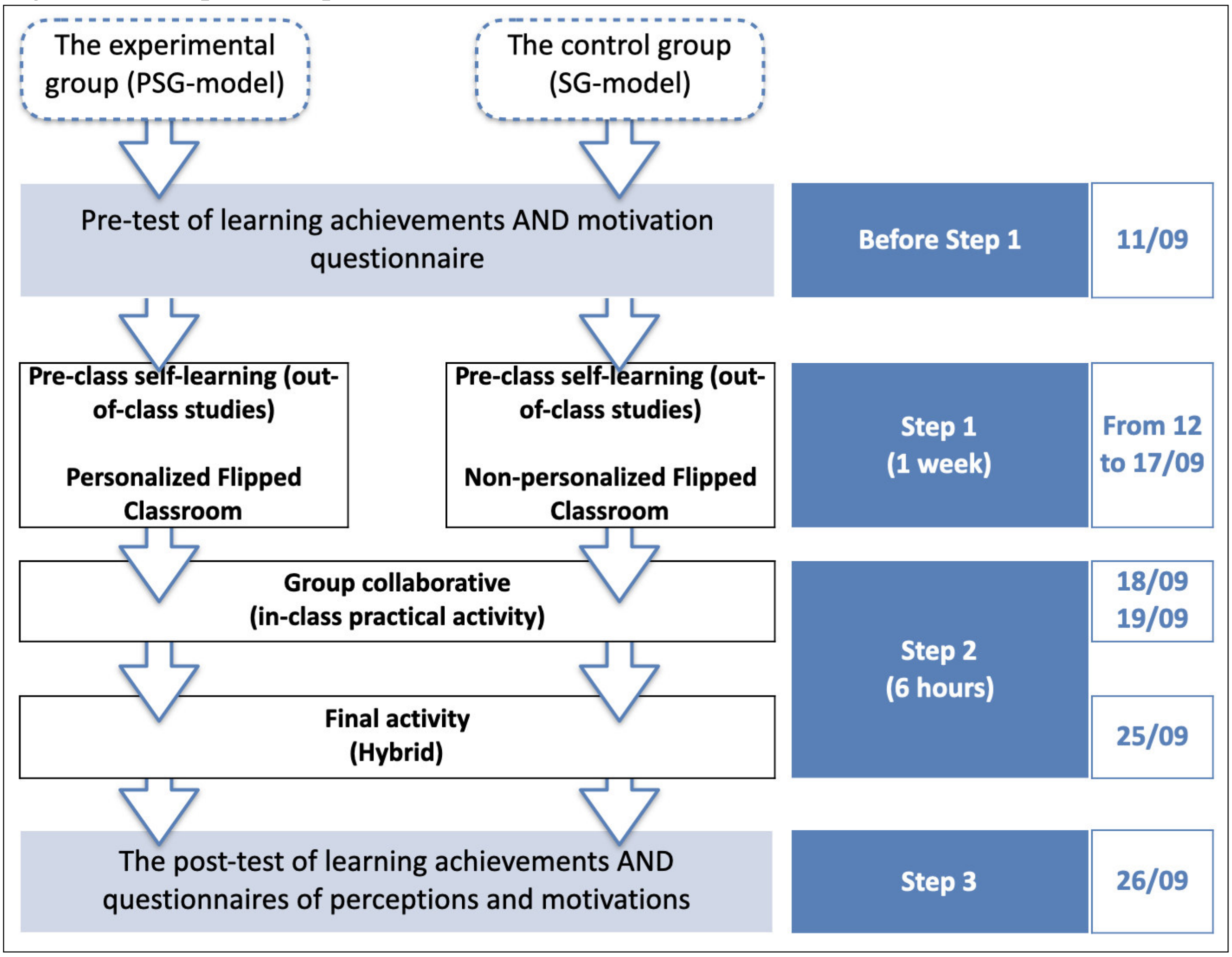
The **effects** of impact combine students' perception, motivation, and achievement results. For the independent variables of this study, the following input variables were considered:

- **Learning strategy** refers to how students study SE concepts. For this experiment, two treatments were investigated:
  Treatment A (the experimental group): students used the FC model with the **personalized** study guide (PSG-model).
  Treatment B (the control group): students using the FC model with a **non-personalized** study guide (SG-model).

6.3.5 Controlled experiment procedure

We followed the same procedural flow presented in Section 6.2.3, but using two groups (control and experimental). Figure 67 presents our experimental process inspired by Hwang *et al.* (2020).

Figure 67 – Experiment process



Source: Author (adapted from Hwang *et al.* (2020)

6.3.6 Student's perception analysis

We compared students' perceptions, grouping them according to the applied treatment (PSG-model or SG-model). Subsequently, we analyzed the data to determine the significance of statistical differences. To achieve this, we examined the data distribution. If the data to be compared followed a normal distribution, we applied the parametric T-Student test; otherwise, we applied the non-parametric Mann-Whitney test. Both tests were configured with a confidence level of 95%, and the data were non-paired, given that we have two groups of randomly selected students.

For cognitive presence (from Q1 to Q8), we obtained an average acceptance rate of 78.8% in Treatment A (PSG-model) and 66.3% in Treatment B (SG-model). Once again, item Q3 achieved the lowest acceptance rate (69.2% in Treatment A and 38.5% in B), similar to the previous evaluation with a single group. This Q3 result reinforces the idea of completeness in the materials provided to students, regardless of how they are delivered. Although we observed a difference in acceptance rates between the treatments, which could suggest lower cognitive presence for Treatment B, according to the Mann-Whitney test, this difference was not significant (*p-value*$>$ 0.05).

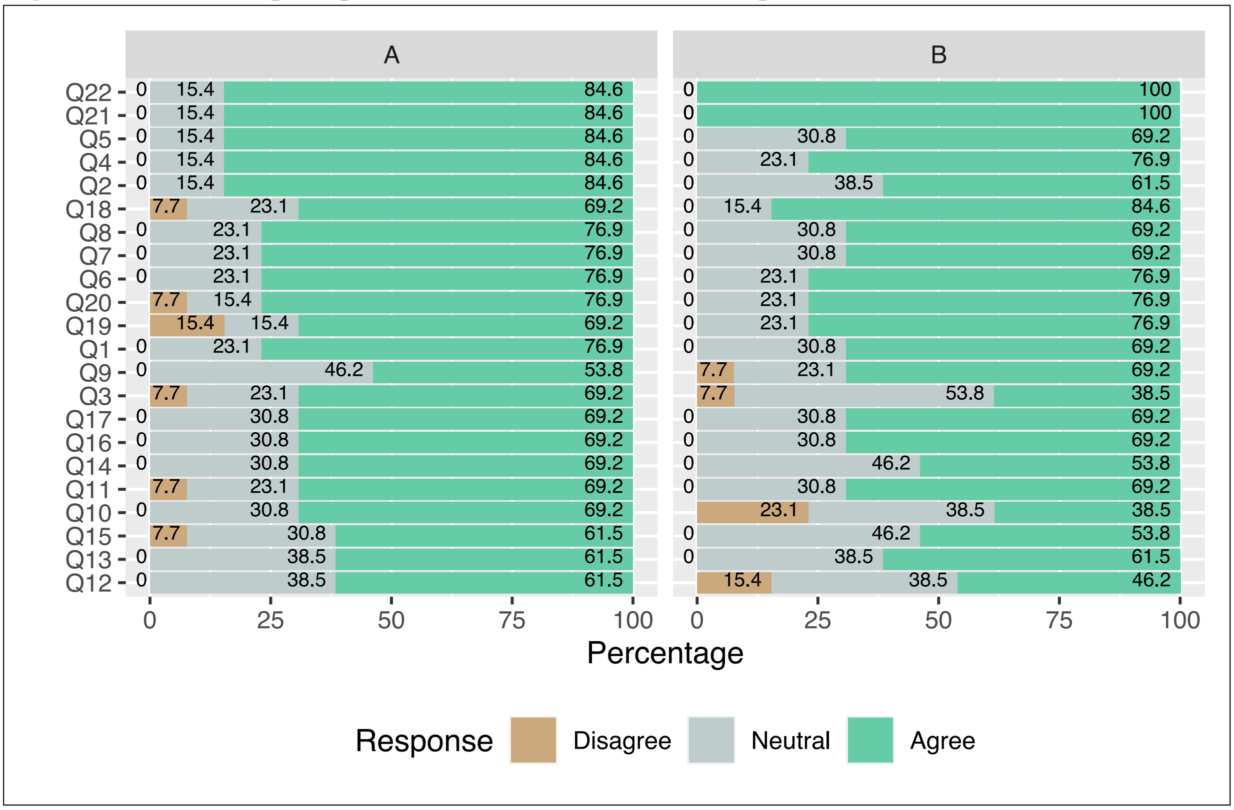
Regarding learner presence (from Q9 to Q16), the average acceptance rate for Treatment A was 64.4%, while in B, it was 57.7%. Once again, for Treatment A, item Q9 had the lowest acceptance rate (53.8%), highlighting the neutrality of some students in planning their studies. Students are likely satisfied with the dynamically constructed study plan by the personalized study guide model. The same item obtained a higher acceptance rate (69.2%) in the non-personalized study guide (Treatment B) compared to Treatment A rate, indicating that students better perceived the need to configure their goals to guide their studies. Item Q10 shows that students in Treatment B (38.5%), compared to those in Treatment A (69.2%), needed help changing their study approach to the flipped classroom style and adapting to the required activities. The linear study guide may have contributed to this difficulty. Despite the differences indicated in acceptance rates between the treatments, there was no statistical difference in terms of learner presence (*p-value* > 0.05).

For technology acceptance (from Q17 to Q22), we found an average acceptance rate of 75.8% among students in Treatment A and 84.6% for B. Notably, students in Treatment B better accepted the proposed tool in all items except for Q17 and Q20 (both treatments had equal average acceptance rates). The most significant differences between the treatments were in items Q21 and Q22 (84.6% for Treatment A and 100.0% for B). These items assessed the intention to use the tool in new flipped classes. Although the difference between the treatments was significant (*p-value* < 0.05), the main result was the acceptance above 75% in both treatments, suggesting a good acceptance of the proposed technology for supporting flipped classes in software engineering. The results of students' perceptions in both treatments are presented graphically in Figure 68.

6.3.7 Student's motivation analysis

We collected data before and after implementing the flipped classroom to analyze the impact on student motivation and compare the statistical differences between Treatments A and B with a confidence level of 95%. We repeated the analysis strategy used for students' perceptions. If the data followed a normal distribution, we applied the T-Student test; otherwise, we used the non-parametric Mann-Whitney test. We set up the tests to use a paired sample

Figure 68 – Student perception data from the controlled experiment



Source: Author

when comparing pre-test and post-test data within the same treatment group and non-paired when comparing treatments, given that the groups were randomly divided. Table 25 presents comparative data on motivation between the two treatments. In addition to the means of students' responses at each stage (pre and post-tests), the table shows the result of the sample comparison (p-value) and highlights in bold the significant differences ($p < 0.05$).

The initial findings indicate that Treatment A improved control over students' learning beliefs compared to Treatment B, particularly regarding item Q26. The results suggest that students in Treatment A better perceived the influence of their invested effort and the level of understanding of the study material in Software Engineering (SE). Overall, Treatment A contributed to a deeper understanding of control beliefs about learning.

Concerning self-efficacy, we did not observe significant differences between the treatments. Looking specifically at item Q34, both treatments significantly improved students' self-efficacy, considering the course difficulties and the student's skills. Students expressed greater confidence in their abilities after the Treatment compared to the data collected before the study began.

Regarding test anxiety, both treatments significantly reduced the levels of anxiety in

Table 25 – Motivation's data for controlled experiment

	Means of treatment A			**Means of treatment B**			**Difference**
	Pre-test	Post-test	p-value	Pre-test	Post-test	p-value	p-value
Q23	4.23	4.08	0.9703	4.38	3.92	0.9949	0.0511
Q24	3.08	3.85	**0.0023**	3.54	3.92	**0.0184**	0.0517
Q25	4.15	3.92	0.9783	4.38	3.92	0.9949	0.1187
Q26	3.00	3.92	**1.032e-05**	3.46	3.84	**0.0089**	**0.0056**
CLB[1]	3.61	3.94	**0.0011**	3.94	3.90	0.6718	**0.0050**
Q27	3.54	3.38	0.8618	3.54	3.46	0.8068	0.5415
Q28	3.38	3.61	0.0745	3.46	3.31	0.8849	**0.0350**
Q29	4.23	3.92	0.9599	4.38	4.08	0.9861	0.5598
Q30	3.61	3.69	0.3864	3.77	3.54	0.9783	0.0584
Q31	3.54	3.69	0.1729	3.77	3.84	0.3864	0.3542
Q32	4.46	3.77	0.9989	4.23	4.00	0.9783	0.9905
Q33	3.85	3.85	0.6813	3.77	3.69	0.8315	0.3056
Q34	3.61	4.00	**0.01844**	3.61	3.92	**0.0359**	0.3545
LSE[2]	3.78	3.74	0.7507	3.82	3.73	0.9653	0.2343
Q35	3.00	2.54	**0.0038**	2.69	2.69	0.5793	**0.0262**
Q36	3.38	2.61	**0.0024**	4.00	2.85	**0.0007**	0.9528
Q37	3.77	2.69	**0.0023**	3.23	2.92	0.0647	**0.0082**
Q38	3.54	2.46	**0.0012**	3.15	2.69	**0.0205**	**0.0162**
Q39	3.38	2.54	**0.0058**	2.92	2.76	0.2420	**0.0217**
LAX[3]	3.41	2.57	**5.233e-10**	3.20	2.78	**3.027e-05**	**0.0008**

[1]CLB = Control of learning beliefs
[2]LSE = Learners' self-efficacy
[3]LAX = Learners' anxiety
Source: Author

students. However, we observed a significant difference in the reduction rates between Treatment A and B. In Treatment A, the average reduction was 0.84 (from 3.41 to 2.57), while in B, it was 0.42 (3.20 to 2.78). All evaluated items (from Q35 to Q39) in Treatment A had a significant reduction. On the other hand, Treatment B reduced only items Q36 and Q38. The difference in the average reductions between Treatments A and B suggests that Treatment A significantly reduced test anxiety more than Treatment B.

6.3.8 Student's achievements analysis

We analyzed students' achievements similarly to one-group students' evaluation (Section 6.2.6), i.e., from the perspective of (1) students' engagement in study guide activities and (2) students' performance in tests (pre and post). Firstly, the Table 26 textually presents student engagement data in study guide activities from Treatment A (PSG-model). The table displays each of the thirteen students who studied using the personalized study guide (column "Student"). The columns "$video_1$ score", "$link_1$ engagement", and "Diagnostic grade" refer to the "pre-study guide activities" and serve to identify the student's profile (explained in Section 4.4.2).

The “Profile” column indicates the identified profile, with all Treatment A students identified as having the “Low” profile. The “Learning path engagement (%)” column shows the percentage of students’ engagement in study guide activities (after profile identification). Finally, the “Grade average” column presents the students’ average grades of the exercises in the study guide. We calculated data according to the equations presented in Section 4.3.2.

Table 26 – Students’ engagement data on Treatment A

Student	$video_1$ score	$link_1$ engagement	Diagnostic grade	Profile	Learning path engagement (%)	Grade average
S1	100	Visited	1.0	Low	75	7.2
S2	100	Visited	3.0	Low	66	5.2
S3	100	Visited	3.0	Low	100	7.7
S4	100	Visited	3.0	Low	100	9.0
S5	100	Visited	3.0	Low	30	2.1
S6	100	Not	3.0	Low	100	8.1
S7	100	Visited	3.0	Low	100	9.0
S8	30	Visited	1.0	Low	95	8.2
S9	100	Visited	3.0	Low	100	9.3
S10	20	Not	1.0	Low	40	4.2
S11	100	Not	1.0	Low	100	8.7
S12	100	Visited	2.0	Low	90	8.9
S13	100	Visited	3.0	Low	100	7.9
Average(%)	88.5	76.9	2.31	-	84.3	7.3

Source: Author

Next, Table 27 organizes student engagement data in Treatment B in study guide activities (SG-model).

Table 27 – Students’ engagement data on Treatment B

Student	$video_1$ score	$link_1$ engagement	Diagnostic grade	Profile	Learning path engagement (%)	Grade average
S1	100	Visited	1.0	Low	66	4.3
S2	100	Visited	1.0	Low	42	3.8
S3	100	Visited	3.0	Low	22	0.0
S4	90	Visited	3.0	Low	97	6.2
S5	100	Visited	3.0	Low	100	8.3
S6	100	Visited	1.0	Low	66	2.8
S7	0	Not	1.0	Low	15	2.8
S8	100	Visited	1.0	Low	98	4.0
S9	100	Visited	3.0	Low	11	0.0
S10	15	Not	1.0	Low	42	5.0
S11	100	Visited	2.0	Low	100	6.5
S12	100	Not	3.0	Low	75	4.6
S13	100	Visited	3.0	Low	92	6.1
Average(%)	85.0	76.9	2.00	-	63.5	4.2

Source: Author

We compared the two groups to analyze differences regarding the treatments. We used descriptive statistics (mean and median) and the non-parametric Mann-Whitney statistical test since the engagement data did not follow a normal distribution. We assessed whether Treatment A was significantly higher than B with a significance level of 95% (p-value < 0.05), i.e., whether the engagement percent and the average grade of students in Treatment A were statistically higher than those in Treatment B. Table 28 presents the comparative results in detail.

Table 28 – Comparison of engagement data between treatments A and B

	Treatment A		**Treatment B**		**A > B ?**
	Mean	Median	Mean	Median	p-value
$video_1$ score	88.5	100.0	85.0	100.0	0.3009
$link_1$ engagement	76.9	100.0	76.9	100.0	0.5140
Diagnostic grade	2.31	3.0	2.00	2.0	0.2177
Learning path engagement (%)	84.3	100.0	63.5	66.0	**0.0314**
Grade average	7.3	8.1	4.2	4.3	**0.0017**

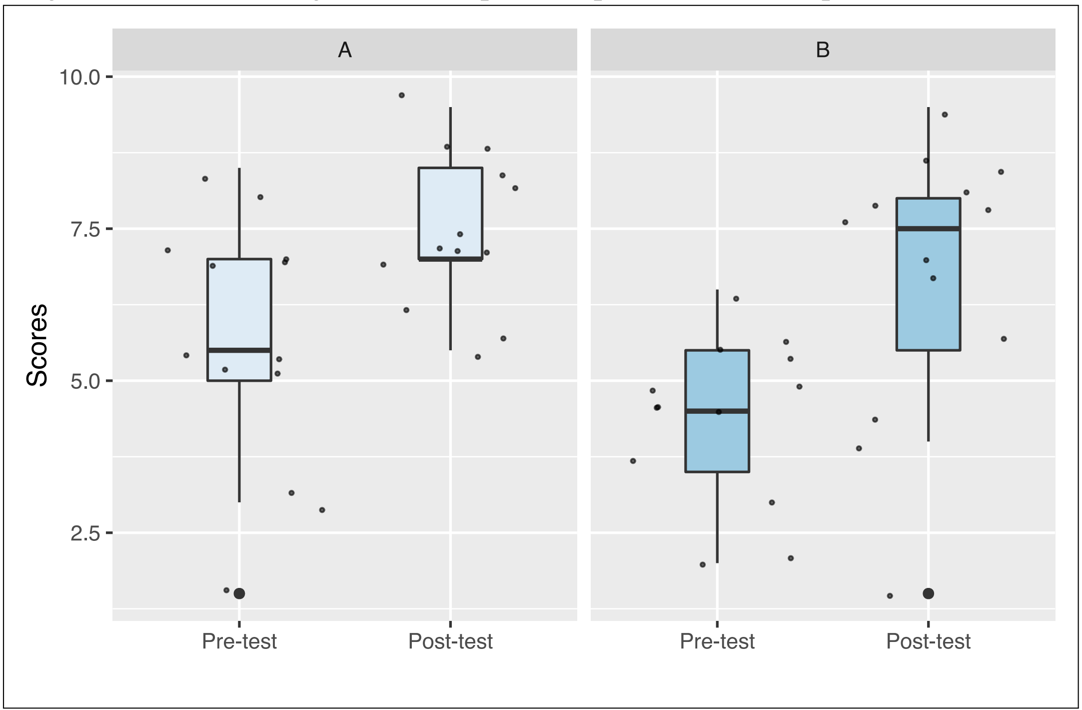
Source: Author

We observed that students who studied using Treatment A were significantly more engaged in the study guide activities and achieved higher scores in the exercises than in Treatment B. We also noted no differences between the treatments in the activities for student profile diagnosis, which were identical for both guides ($video_1$ score, $link_1$ engagement, diagnostic grade).

We assessed students' performance through tests before (pre) and after (post) the flipped lesson on software requirements. Initially, we compared the pre-test and post-test data within the same group to evaluate the level of impact of the Treatment on students' scores. The score data followed a normal distribution in both treatments, so we used the T-Student test for significance analysis (p-value < 0.05). Figure 69 displays the pre-test and post-test scores grouped by Treatment.

We observed a significant impact (p-value< 0.05) in the growth of post-test scores compared to pre-test scores in both treatments. The positive impact suggests the model's effectiveness in learning gain to the software requirements lesson, regardless of the kind of study guide (PSG-model or SG-model) applied. When comparing the impacts between treatments, we noted no significant difference in the growth of scores; that is, Treatment A was not better than B in this aspect. Table 29 presents the score data.

Figure 69 – Students' grades in the pre- and post-tests from experiment data



Source: Author

Table 29 – T-test results of experiment learning performance

		Treatment A				**Treatment B**				**A > B?**
Test	N	Mean	SD	*t*	*p-value*	Mean	SD	*t*	*p-value*	p-value
Pre-test	13	5.61	2.10	3.6030	**0.0018**	4.39	1.39	3.6236	**0.0017**	0.7282
Post-test		7.42	1.32			6.69	2.25			

Source: Author

6.4 Discussion

This section discusses the outcomes obtained for the two research questions addressed in this chapter, organized as follows:

6.4.1 Acceptance levels of the FCTool - RQ4

Summary of RQ4

(i) The technology garnered an acceptance rate of over 88%, expressing professors' belief in its potential to improve teaching efficiency and productivity; (ii) Challenges in organizing and monitoring out-of-class studies highlight the necessity for a centralized tool in flipped classroom approaches; (iii) our model showcases innovative potential by providing personalized learning paths, ensuring equitable resource delivery and enabling self-paced learning in software engineering teaching.

We conducted a technology acceptance assessment with software engineering professors through a semi-structured interview. As part of a research study, a protocol was established to conduct interviews with SE professors. A data collection instrument was designed to gather information from the interviewees and was based on the Technology Acceptance Model, a widely used theoretical framework for understanding and predicting users' acceptance of new technologies. The questionnaire included questions about the participant's perceptions of the technology, usefulness, ease of use, attitude, and intention to use. Initially, qualitative data were gathered regarding the professors' experiences using the flipped classroom method in software engineering teaching. It was observed that professors employ various tools to organize and monitor home studies. However, it was noted that organizing (curating), delivering, monitoring, and making decisions based on student performance becomes challenging without a tool that centralizes data on studies, particularly those conducted at home.

Reports indicated excessive effort in monitoring student progress while simultaneously highlighting success stories of flipped classes, demonstrating improved learning outcomes compared to previous periods without using the method, employing traditional teaching approaches. In this context, our model can improve flipped classroom management by sequentially organizing content and monitoring student progress through quantifying engagement and performance in exercises. When integrated into a tool, our model has the innovative potential to improve software engineering teaching by providing personalized learning paths to students who face significant difficulties in comprehending the content. This technology ensures that educational resources are delivered equitably to all students, regardless of their level of proficiency, by tailoring the learning experience to their specific needs. The model allows students to identify areas where they are struggling and provide them with alternative paths better suited to their learning style and abilities. This approach not only maximizes the effectiveness of the educational resources but also enables students to learn at their own pace and achieve their full potential.

After guiding the tool's usage and demonstrating its functionalities for creating and monitoring study plans, we collected data on perceived usefulness, perceived ease of use, attitude, and intention to use. Across all dimensions, we observed an acceptance rate exceeding 88%. This rate attests to the belief held by the interviewed professors that adopting the technology can yield tangible benefits in terms of efficiency, productivity, and performance. The data also indicates a positive perception regarding usability. There is a belief that technology is feasible with minimal obstacles or excessive efforts. According to the feedback from the professors, they have perceived the proposed technology (FCTool) positively. As a result of their positive perceptions, the professors have expressed a robust inclination towards adopting and using the technology in their flipped classrooms very soon. They have explicitly stated their intention to use it as they believe it can significantly benefit their teaching methodologies. The interviewees have also indicated that the FCTool is easy to use, reinforcing their positive disposition. Overall, based on the views of the professors, the proposed technology has the potential to be an effective tool for enhancing the effectiveness of flipped classrooms.

6.4.2 Personalized study guide model impact - RQ5

Summary of RQ5

(i) Average acceptance rate of 76.7%; (ii) Average reduction of 34.6% in test anxiety rate; (iii) Average engagement rate 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; (v) PSG-model proved to be more efficient in enhancing the control of learning beliefs and reducing test anxiety than SG-model; (vi) Students using PSG-model obtained an average engagement 20.8% higher and an average grade 3.1 points greater than students using SG-model.

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.

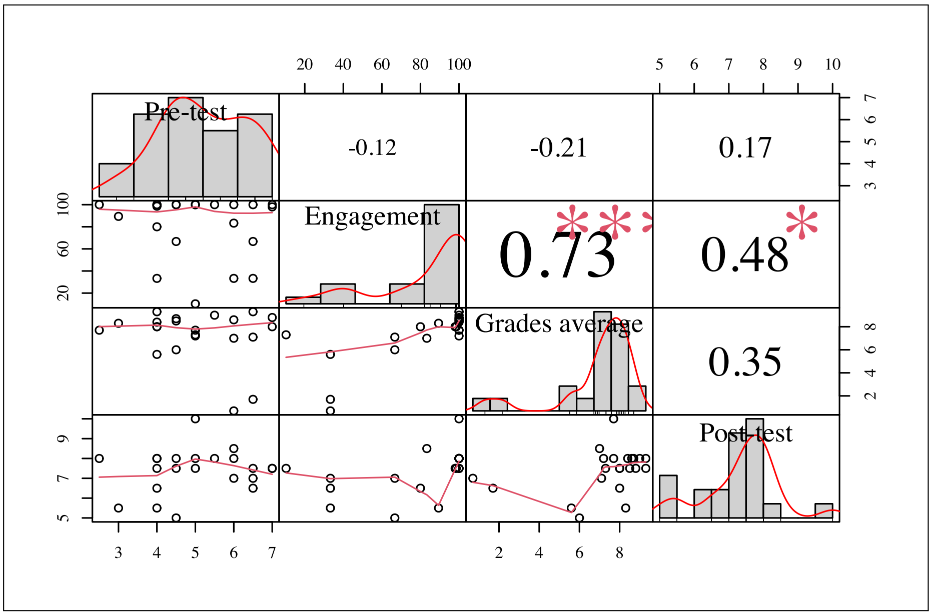
The learner presence outcome underscores and highlights the importance of goal setting for learning. Lacking this practice could affect the efficiency of realizing the study plan for some students. The technology acceptance outcomes suggest that the tool can offer tangible benefits and enhance students' 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 the 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 suggest 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. From the comparative assessment between the group of students who studied using the PSG-model and the group that used the SG-model, we noted that the personalized study guide group improved students' perception of their control of learning beliefs while significantly reducing test anxiety. These results suggest that students prepare better for in-class sessions through pre-class studies (out-of-class). This improvement may positively influence self-esteem, confidence, and well-being and potentially impact overall outcomes.

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. We noted that students utilizing the PSG-model obtained an average engagement 20.8% higher than those who studied using the SG-model. The content's adaptability may have motivated students to go further in their learning path during out-of-class preparation (study guide). The learning path taken by students using the PSG-model resulted in an average score of 3.1 points higher than that of students who studied using the SG model.

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 index (SCHOBER *et al.*, 2018). Figure 70 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 70 – Pearson's correlation coefficient



Source: Author

6.5 Limitations, Threats to validity, and Ethics considerations

6.5.1 Ethical considerations

In terms of ethical considerations, we carefully thought about our evaluation and experiments 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.

6.5.2 Limitations

It is important to note that certain factors may introduce bias into our study. This experimental research was conducted within a specific context, led by one of the professors. Additionally, there was a risk that the learning outcomes may have been impacted due to the excessive workload placed on the students. This is because the students were expected to complete some activities simultaneously, including other courses and jobs, which may have compromised their level of participation in our study.

6.5.3 Threats to validity

The discussion of threats to validity encompasses the categories of construct validity, internal validity, and reliability, as described by Runeson *et al.* (2012). Regarding construct validity in our study, the issue is associated with subjectivity in analyzing 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.

We emphasize the possibility that, during the evaluation of the student's learning impact, even though they were warned about not using reference materials during the tests, some may have searched for answers during the pre and post-tests, not relying solely on the knowledge acquired during the study. The students' performance may also have been affected by the high demand for activities from other disciplines, in addition to work, household chores, or any other factors not mentioned by the students.

Regarding external validity, the limited number of participants is noteworthy. In total, 22 students participated in the one-group experiment, 26 in the controlled experiment, and seven teachers, with these quantities constrained by the demand for students per discipline and the number of available slots. We organized details about the materials developed throughout the study (such as instruments, scripts, and instructional materials).

Considering the conclusion validity, the application of the questionnaires may reflect weaknesses in terms of the level of reliability in responses and the uniformity of measurements.

6.6 Chapter summary

This chapter evaluated the model introduced in Chapter 4. The assessment utilized the tool described in Chapter 5. It aimed to address two research questions: (RQ4 - *What are the SE professors' perceived usefulness and ease of use of the Flipped Classroom Tool?*) and (RQ5 - *What is the impact of Personalized Study Guides on students' perception, motivation, engagement, and learning gain during software engineering flipped classes?*). The first research question (RQ4) encompassed individual interview sessions with seven software engineering professors, both with and without experiences in flipped classrooms. Two evaluations were conducted with software engineering students to address the second research question (RQ5). The first involved a group of students using the PSG-model, and the second included two groups of students, one using the PSG-model and the other using the SG-model.

The chapter delineates the steps of the three evaluations. In the first one, involving software engineering professors, we devised a seven-step semi-structured protocol for conducting interviews. Ultimately, we generated qualitative and quantitative data on the professors' perceptions from interviews and an instrument inspired by the Technology Acceptance Model (TAM). Overall, the results indicated that the professors accepted the proposed technology across four dimensions: Perceived Usefulness, Perceived Ease of Use, Attitude, and Intention to Use.

Subsequently, we assessed the PSG-model with 22 software engineering students in the second evaluation. Evaluation was conducted across three dimensions: Students' Perception, Motivations, and Achievements. We observed an acceptance rate of 76.5% concerning students' perceptions. Regarding motivation, the model improved control over learning beliefs and reduced test anxiety rates by 34.6%. The model generated an average engagement of 81.5% in proposed home study activities and an average improvement of 2.46 in student grades.

In the third evaluation, we compared the PSG-model with the SG-model (without content personalization). Two groups of 13 randomly chosen students were utilized for this comparative assessment. The exact dimensions from the second evaluation were employed. It was noted that the PSG-model surpassed the SG-model in enhancing control over learning beliefs and reducing test anxiety. Additionally, the PSG-model outperformed in student engagement and average grades on preparatory exercises. However, the PSG-model did not demonstrate superiority over the SG-model concerning improvement in pre and post-test grades. The next chapter, Chapter 7, presents the conclusions of this thesis and possible directions for future works.

7 CONCLUSIONS

This chapter concludes the doctoral thesis, which aimed to investigate the impacts on students' perception, motivation, engagement, and learning using our Personalized Study Guide Model for Software Engineering Flipped Classes.

The chapter begins with an overview of the thesis in Section 7.1. It revisits the research questions introduced in Section 1.3 in Section 7.2. Following that, Section 7.2.6 discusses and checks the thesis hypothesis concerning its support from the results presented in this thesis. Section 7.2.7 summarizes the contributions achieved in this Ph.D. work. Finally, Section 7.3 outlines potential avenues for future work.

7.1 Thesis overview

This thesis explored the effectiveness of Flipped Classroom (FC) when integrated with adaptive learning techniques for teaching Software Engineering (SE). The investigation started with a systematic mapping study aiming to understand the state of the art of the FC method usage in SE teaching (Chapter 2). We identified 38 papers, extracting insights into challenges, advantages, drawbacks, methods combined with FC, active learning methodologies, tools, and the primary impacts on student learning.

We subsequently deepened our understanding of the flipped classroom process in software engineering teaching through a survey conducted with researchers knowledgeable about this method (Chapter 3). The survey aimed to discover the characteristics an authoring tool should have to help SE instructors prepare flipped classrooms using adaptive learning techniques.

Based on this understanding, we formalized a model for personalized study guides in Chapter 4 and implemented it into a tool in Chapter 5. We demonstrate the viability of the model in creating personalized study guides for Software Engineering flipped classes.

Finally, we utilized the developed tool to assess the impacts of our model on students' perception, motivation, and achievements (Chapter 6). Before this evaluation, we employed a technology acceptance model to determine our tool's acceptance and usage levels with SE professors.

7.2 Revisiting the Research Questions

This thesis defined five research questions, which are discussed in the next subsections in terms of their main results.

7.2.1 RQ1 - What is the state of the art of using the flipped classroom method in Software Engineering Teaching?

We conducted exploratory research through a literature review to answer this research question. We conducted a mapping study followed by a snowballing technique (described in Chapter 2).

The state of the art of using the FC method in teaching SE was described in terms of (1) active methods used in-class activity, (2) pedagogical strategies that were blended with the flipped classroom method, (3) tools to support the flipped classroom method, (4) assessment methods, (5) student learning impact, and (6) advantages and challenges about flipped classroom usage. Besides, advantages and challenges regarding the FC method usage in SE teaching were identified and listed as tools for producing, consuming, and delivering out-of-class content. Other teaching methods used in combination with a flipped classroom were also found. Gamification was the most used to amplify flipped class potential.

We learned that (1) diversification in Software Engineering teaching suggests that the flipped classroom method can be adapted to various aspects of SE; (2) using active learning strategies combined with the flipped classroom method can improve learning, especially during in-class practical sessions; (3) the quality of out-class content and resources influences student's motivation and satisfaction, underscoring the importance of investing time and effort into developing adequate resources; (4) There are positive outcomes in the impact on learning associated with flipped classrooms, including students' active participation, critical thinking promotion, social skills development, and contact with real-world projects.

7.2.2 RQ2 - What are the characteristics that an authoring tool should have to help Software Engineering professors during the preparation of flipped classes?

A descriptive survey was carried out with Software Engineering (SE) researchers to address the second research question. Chapter 3 gathered insights and data related to specific features necessary for constructing a flipped classroom tool tailored for personalized study guides in SE education. The primary objective of the chapter was to deepen our understanding of the flipped classroom process in software engineering education. The findings assisted in formulating design principles for the preparation of flipped classes. Twenty design principles were elaborated on and grouped as follows:

a) Eight for facing challenges and difficulties.

b) Eight for strengthening advantages and addressing drawbacks.

c) Four for emphasizing resources and evaluation.

In addition, five other universal principles were presented for personalized learning design. The design principles helped us design a tool for the flipped classroom with a personalized study guide.

7.2.3 RQ3 - Is feasible to design, formalize, and implement a Personalized Study Guide Model to represent out-of-class material for SE flipped classes?

The third research question pertained to the solution stage, where we explored the practical implementation of the model. To this end, we divided our response into two chapters. Chapter 4 focused on formally defining the model using Business Process Model and Notation (BPMN) notation. The chapter detailed the model elements and how they could be evaluated from the student's perspective. To demonstrate the feasibility of the model, we presented a proof of concept on software requirements in the form of a flipped classroom. The example showcased how the model could be applied in practice and its potential benefits to students. Finally, we conducted a computer-based simulation to validate the model's design and formalization feasibility further. The simulation results revealed different learning paths that hypothetical students could take, highlighting the model's flexibility and adaptability.

Chapter 5 complemented the response to the third research question by demonstrating the practical applicability of the model. The chapter introduced a web-based tool that implemented the model and compared it with the theoretical model previously presented. The chapter then described the tool's functionalities and showcased a prototype through a minimum-viable product. The primary contribution of the chapter was to demonstrate the feasibility of implementing the model to represent out-of-class material for Software Engineering flipped classes. The tool's implementation showed that the model could be applied in practice and offered a user-friendly interface for students to engage with the flipped classroom materials.

7.2.4 RQ4 - What are the SE professors' perceived usefulness and ease of use of the Flipped Classroom Tool?

For our fourth research question, we interviewed seven Software Engineering professors. Our objective was to evaluate our flipped classroom tool's perceived usefulness and ease of use. We followed a semi-structured interview protocol to ensure that we covered all the relevant aspects of the tool. Upon analysis of the data collected, we found that the professors had an acceptance rate of over 88%. The results also showed that they believed in the tool's potential to enhance efficiency and productivity in teaching. They reported challenges in organizing and monitoring student data related to out-of-class studies. Professors felt the need for a centralized tool for flipped classroom approaches to overcome this challenge.

7.2.5 RQ5 - What is the impact of Personalized Study Guides on students' perception, motivation, engagement, and learning gain during software engineering flipped classes?

We conducted two evaluations with software engineering students to address our fifth research question. The first evaluation involved a one-group assessment with 22 students, while the second comprised an empirical study using a controlled experiment with 26 students. These evaluations aimed to investigate the impacts of personalized study guides on students' perception, motivation, engagement, and learning gain during software engineering flipped lessons.

The results of the evaluations showed that the average acceptance rate was 76.7%, indicating that the students found the personalized learning approach acceptable. There was also a significant reduction in test anxiety, with a mean reduction of 34.6%. The study guide activities had an average engagement rate of 81.5%, indicating that students found the activities exciting and engaging. Furthermore, the mean performance score of students in the study guide exercises was 7.3, which suggests that students learned and retained the material effectively.

The use of personalized study guides demonstrated higher efficiency in enhancing learning and reducing test anxiety compared to the control group. On average, students who used personalized study guides exhibited 20.8% higher engagement and a 3.1-point higher score than those who used the other approach. 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.

7.2.6 *Revisiting the Thesis Hypothesis*

At the beginning of this research, this work established the following research hypothesis:

Research Hypothesis

Software Engineering Flipped Classes using Personalized Study Guides positively impact students' perception, motivation, engagement, and learning.

The study's outcomes positively impact students' perceptions regarding their acceptance of cognitive presence, learner presence, and technology use. The mean percentage of students with a positive perception of cognitive presence was 82%. In comparison, 65.3% of students had a positive perception of learner presence, and 77.2% had a positive perception of technology use.

Regarding the motivations, the study found significant results regarding the decrease in test anxiety, with an average percentage of 34%. Additionally, the study found a positive impact on increasing control of learning beliefs, with a mean percentage of 8.4%.

Moreover, the results positively impacted the average engagement of students with a percentage of 84.3%. The study also recorded an average learning gain of 3.1%. These outcomes suggest that the intervention positively impacted students' perceptions, motivations, engagement, and learning outcomes using personalized study guides on software engineering flipped classes.

7.2.7 *Contributions and Publications*

The following papers are directly related to the thesis and were published during its development:

- Necio Veras, Lincoln S. Rocha, and Windson Viana. "A Personalized-based Study Guides Tool for Software Engineering Flipped Classes" Proceedings of IV Brazilian Symposium on Computer Education (**Accepted, pending publication**), 2024.
- Veras, Nécio L., Lincoln S. Rocha, and Windson Viana. "An Authoring Tool to Support Flipped Classroom in Software Engineering Teaching." Proceedings of the 17th ACM Conference on International Computing Education Research. 2021.
- Veras, Nécio L., Lincoln S. Rocha, and Windson Viana. "Flipped Classroom in Software Engineering: A Systematic Mapping Study." Proceedings of the 34th Brazilian Symposium

on Software Engineering. 2020.

We have recently submitted an article to a journal. Experts in the field then reviewed the paper, and we received valuable feedback from them. We are currently working on incorporating the suggested changes and updates to the paper to improve its quality. Furthermore, we are in the final stages of completing another paper on the model. It will also be submitted to a journal soon. We hope both papers will contribute to the academic community and provide insights that can benefit future research.

The following papers have been **co-authored** and are indirectly related to this thesis.

- Rodrigues, Maria Elanne, Gabriela Nayara Damazio, Necio Veras, Anna Beatriz Marques, and Windson Viana. “Gamificando Aulas Invertidas no Ensino de Engenharia de Requisitos: Um Relato de Experiência.” Anais do III Simpósio Brasileiro de Educação em Computação, Evento Online, 2023. SBC, 2023.
- Alves, Gabriela., Soares, Pamella., Souza, Bianca., Veras, Nécio., AraúJo, Allysson. “What is students’ perception of Flipped Classroom usage during Emergency Remote Teaching of Software Engineering subjects? An Experience Report in Brazil.” Proceedings of the XXXVI Brazilian Symposium on Software Engineering. 2022.
- Souza, Bianca., Veras, Nécio., Olivindo, Mayara., Mendes, Elanne., Rocha, Lincoln., Viana, Windson. “Ensino invertido de Estrutura de Dados no contexto do Ensino Remoto Emergencial.” Anais do II Simpósio Brasileiro de Educação em Computação. SBC, 2022.
- Olivindo, Mayara., Veras, Nécio., Viana, Windson., Cortés, Mariela., Rocha, Lincoln. “Gamifying flipped classes: An experience report in software engineering remote teaching.” Brazilian Symposium on Software Engineering. 2021.
- Araújo, Pedro., Costa, Cecília., Viana, Windwson., Veras, Nécio., Farias, Éder. “Automatic Personalisation of Study Guides in Flipped Classroom: A Case Study in the course of Distributed Systems” Frontiers in Education. IEEE, 2020. p.1-9.
- Quarto, Cicero., Busson, Antonio., Veras, Nécio., Fonseca, Luis., Nascimento, Edson., Bercht, Magda. “A Multiagent System in Aid to the Formation of Groups for Collaborative Learning.” 2019 IEEE 19th International Conference on Advanced Learning Technologies (ICALT). Vol. 2161. IEEE, 2019.
- Veras, Nécio., Viana, Windson., Passos, Anderson., Miranda, Moisés., Rocha, Antônio Carlos., Gomes, Fernando. “Jogos sérios nos cuidados com Diabetes: Um App Review.” Anais da VII Escola Regional de Computação Aplicada à Saúde. SBC, 2019.

- Araujo, Pedro., Viana, Windson., Veras, Nécio., Farias, Eder., Castro, José Aires. “Exploring Students Perceptions and Performance in Flipped Classroom Designed with Adaptive Learning Techniques: A Study in Distributed Systems Courses.” Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE). Vol. 30. No. 1. 2019.
- Rodrigues, Willian Almeida., Costa, Cecília., Veras, Nécio., Cortez, Vitor., Andrade, Ricardo., Trinta, Fernando., Mendonça, Glaudiney., Viana, Windson. “Avaliando o Tardigrade: um framework para o desenvolvimento de jogos de cartas ubíquos.” Anais do X Simpósio Brasileiro de Computação Ubíqua e Pervasiva. SBC, 2018.
- Araujo, Pedro., Viana, Windson., Veras, Nécio., Castro, José Aires. “Aprendizagem Adaptativa em Aulas Invertidas de Sistemas Distribuídos: um estudo de caso apoiado no G Suite.” Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE). Vol. 29. No. 1. 2018.

7.3 Future Work

This research suggests future work considering three main topics: (1) model, (2) implementation, and (3) evaluation. Firstly, concerning the model, possible future work includes:

a) Adding new elements to the model, such as the automatic formation of groups for collaborative learning and generative artificial intelligence to provide online support (chatbots). Investigating the impacts of these additional elements in learning gain;
b) Comparatively investigating the effects of content personalization for each personalization point present in the model;
c) Integrating gamification elements into the model;
d) Create a repository of templates with personalized study guides for software engineering classes. The templates will serve as a usage guide for SE instructors.

We defined a Personalized Study Guide model to support the flipped classroom method in SE teaching. Adaptive learning is based on literature and grounded theory, but it is only one of the various possibilities for integration with the flipped classroom. Other knowledge sources and techniques can be used, and future work can explore complementary approaches to the proposed model. For example, future work could consider: (1)Incorporating real-world software project-based learning experiences; (2) Exploring case studies with industry practices using concrete examples; and (3) Implementing social learning, where students can collaborate, discuss, and share knowledge among themselves.

Secondly, regarding the implementation of the model, opportunities for future work include:

a) Increase the MVP presented in Chapter 5 by implementing all the design principles described in Chapter 3 and all model elements presented in Chapter 4;
b) Exploring alternative implementations of the proposed model, such as creating a plugin for the Moodle environment, or one page website-based;
c) Assessing and improve the accessibility of FCtool.

The model was implemented based on features discovered through literature review and a survey with researchers, but other functionalities may exist in the literature or even in successful experiences. A future work idea is to investigate additional functionalities not covered in this thesis. An interesting approach is to catalog best practices for flipped classroom teaching from experiences not disclosed in the scientific literature.

Finally, concerning future work based on the evaluations conducted in Chapter 6, potential suggestions include:

a) Evaluating the model for other quality characteristics beyond the scope of this thesis, as others could replace the used instruments;
b) Obtaining qualitative and quantitative feedback from other professors using our Personalized Study Guide Model to design Software Engineering Flipped Classes;
c) A case study involving other SE professors applying our tool in their classes. From this study, investigate how students adhere to the use of the tool and how teachers adapt to the practical use of the tool;
d) Investigating more deeply possible negative effects of the adaptive approach, such as the novelty effect that may lead to a reduction in engagement over time;
e) Evaluating ways to adapt content to students: manual, automatic, or semi-automatic;
f) Another future work related to this thesis is its application and evaluation in large-scale class scenarios, such as online courses.

REFERENCES

ACHARYA, S.; MANOHAR, P.; WU, P.; MAXIM, B. Strategies for delivering active learning tools in software verification & validation education. In: **124th ASEE Annual Conference & Exposition**. [S.l.: s.n.], 2017.

AKÇAYIR, G.; AKÇAYIR, M. The flipped classroom: A review of its advantages and challenges. **Computers & Education**, Elsevier, v. 126, p. 334–345, 2018.

AKDUR, D. Analysis of software engineering skills gap in the industry. **ACM Transactions on Computing Education**, ACM New York, NY, v. 23, n. 1, p. 1–28, 2022.

AMRESH, A.; CARBERRY, A. R.; FEMIANI, J. Evaluating the effectiveness of flipped classrooms for teaching cs1. In: IEEE. **2013 IEEE Frontiers in Education Conference (FIE)**. [S.l.], 2013. p. 733–735.

ARAÚJO, P.; COSTA, C.; VIANA, W.; VERAS, N. de L.; FARIAS, E. J. Automatic personalisation of study guides in flipped classroom: A case study in a distributed systems course. In: IEEE. **2020 IEEE Frontiers in Education Conference (FIE)**. [S.l.], 2020. p. 1–9.

ARAUJO, P.; VIANA, W.; VERAS, N.; FILHO, J. A. de C. 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)**. [S.l.: s.n.], 2018. v. 29, n. 1, p. 328.

ARAUJO, P.; VIANA, W.; VERAS, N.; FARIAS, E. J.; FILHO, J. A. de C. 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)**. [S.l.: s.n.], 2019. v. 30, n. 1, p. 219.

ARAÚJO, P. T. d. Fctool: uma ferramenta para a geração de roteiros adaptativos de aulas invertidas de sistemas distribuídos. Fortaleza, 2019.

BARBOSA, E. F.; MOURA, D. G. de. Metodologias ativas de aprendizagem na educação profissional e tecnológica. **Boletim Técnico do Senac**, v. 39, n. 2, p. 48–67, 2013.

BERGMANN, J.; SAMS, A. **Flip your classroom: Reach every student in every class every day**. International society for technology in education, 2012. ISBN ISBN 9781564843159. Available at: <http://books.google.com/books?id=nBi2pwAACAAJ>.

BOURQUE, P.; FAIRLEY, R. E. *et al.* **Guide to the software engineering body of knowledge (SWEBOK (R)): Version 3.0**. [S.l.]: IEEE Computer Society Press, 2014.

BRAME, C. Flipping the classroom. Retrieved from The Vanderbilt University Center for Teaching website, 2022. Available at: <https://cft.vanderbilt.edu/guides-sub-pages/flipping-the-classroom/>.

BROWN, S. **Likert scale examples for surveys**. [S.l.]: IA:Iowa State University, 2010.

CARCHIOLO, V.; LONGHEU, A.; MALGERI, M. Adaptive formative paths in a web-based learning environment. **Journal of Educational Technology & Society**, JSTOR, v. 5, n. 4, p. 64–75, 2002.

CARVALHO, R. M. **Correlate & lead: process and catalog of non-functional requirements correlations in ubicomp and iot systems**. Phd Thesis (Doutorado em Ciência da Computação) — Centro de Ciências e Tecnologia, Programa de Mestrado e Doutorado em Ciência da Computação, Universidade Federal do Ceará, 2019.

CEVIKBAS, M.; KAISER, G. Promoting personalized learning in flipped classrooms: A systematic review study. **Sustainability**, MDPI, v. 14, n. 18, p. 11393, 2022.

CHANIN, R.; SALES, A.; POMPERMAIER, L.; PRIKLADNICKI, R. A systematic mapping study on software startups education. In: **Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018**. [S.l.: s.n.], 2018. p. 163–168.

CHETLUR, M.; TAMHANE, A.; REDDY, V. K.; SENGUPTA, B.; JAIN, M.; SUKJUNNIMIT, P.; WAGH, R. Edupal: Enabling blended learning in resource constrained environments. In: **Proceedings of the Fifth ACM Symposium on Computing for Development**. [S.l.: s.n.], 2014. p. 73–82.

CHOI, E. M. Applying inverted classroom to software engineering education. **International Journal of e-Education, e-Business, e-Management and e-Learning**, IACSIT Press, v. 3, n. 2, p. 121, 2013.

DAVIS, F. D. **A technology acceptance model for empirically testing new end-user information systems: Theory and results**. Phd Thesis (PhD Thesis) — Massachusetts Institute of Technology, 1985.

EASTERBROOK, S.; SINGER, J.; STOREY, M.-A.; DAMIAN, D. Selecting empirical methods for software engineering research. In: **Guide to advanced empirical software engineering**. [S.l.]: Springer, 2008. p. 285–311.

ELLIOTT, R. Do students like the flipped classroom? an investigation of student reaction to a flipped undergraduate it course. In: IEEE. **2014 IEEE Frontiers in Education Conference (FIE) Proceedings**. [S.l.], 2014. p. 1–7.

ERDOGMUS, H.; GADGIL, S.; PERAIRE, C. Introducing low-stakes just-in-time assessments to a flipped software engineering course. In: **Proceedings of the 52nd Hawaii International Conference on System Sciences**. [S.l.: s.n.], 2019.

ERDOGMUS, H.; PÉRAIRE, C. Flipping a graduate-level software engineering foundations course. In: IEEE. **2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering Education and Training Track (ICSE-SEET)**. [S.l.], 2017. p. 23–32.

FASSBINDER, A. G. de O.; FASSBINDER, M.; BARBOSA, E. F. From flipped classroom theory to the personalized design of learning experiences in moocs. In: IEEE. **2015 IEEE Frontiers in Education Conference (FIE)**. [S.l.], 2015. p. 1–8.

FETAJI, M.; FETAJI, B.; EBIBI, M. Analyses of possibilities of flipped classroom in teaching computer science courses. In: IEEE. **2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)**. [S.l.], 2019. p. 747–752.

FINK, A. **How To Design Surveys. The Survey Kit, Volume 5.** [S.l.]: ERIC, 1995.

FIQRI, M.; NURJANAH, D. Graph-based domain model for adaptive learning path recommendation. In: IEEE. **2017 IEEE Global Engineering Education Conference (EDUCON)**. [S.l.], 2017. p. 375–380.

FORTUNA, J.; JUSTASON, M. D.; SINGH, I. Conversion of a software engineering technology program to an online format: A work in progress and lessons learned. In: **Online Engineering & Internet of Things**. [S.l.]: Springer, 2018. p. 851–858.

GANNOD, G.; BURGE, J.; HELMICK, M. Using the inverted classroom to teach software engineering. In: IEEE. **2008 ACM/IEEE 30th International Conference on Software Engineering**. [S.l.], 2008. p. 777–786.

GHAZI, P.; GLINZ, M. An experimental comparison of two navigation techniques for requirements modeling tools. In: IEEE. **2018 IEEE 26th international requirements engineering conference (RE)**. [S.l.], 2018. p. 240–250.

GIANNAKOS, M. N.; KROGSTIE, J.; CHRISOCHOIDES, N. Reviewing the flipped classroom research: reflections for computer science education. In: **Proceedings of the computer science education research conference**. [S.l.: s.n.], 2014. p. 23–29.

GIANNAKOS, M. N.; KROGSTIE, J.; SAMPSON, D. Putting flipped classroom into practice: A comprehensive review of empirical research. In: **Digital technologies: Sustainable innovations for improving teaching and learning**. [S.l.]: Springer, 2018. p. 27–44.

GLAZUNOVA, O.; VOLOSHYNA, T.; KOROLCHUK, V.; PARHOMENKO, O. Cloud-oriented environment for flipped learning of the future it specialists. **E3SWC**, v. 166, p. 10014, 2020.

GREN, L. A flipped classroom approach to teaching empirical software engineering. **IEEE Transactions on Education**, IEEE, 2020.

HAIDER, M.; SINHA, A.; CHAUDHARY, B. An investigation of relationship between learning styles and performance of learners. **International Journal of Engineering Science and Technology**, v. 2, n. 7, p. 2813–2819, 2010.

HAMMAMI, S.; MATHKOUR, H. Adaptive e-learning system based on agents and object petri nets (aels-a/opn). **Computer applications in engineering education**, Wiley Online Library, v. 23, n. 2, p. 170–190, 2015.

HAN, H.; RØKENES, F. M. Flipped classroom in teacher education: A scoping review. In: FRONTIERS MEDIA SA. **Frontiers in Education**. [S.l.], 2020. v. 5, p. 601593.

HAO, Y. Exploring undergraduates' perspectives and flipped learning readiness in their flipped classrooms. **Computers in Human Behavior**, Elsevier, v. 59, p. 82–92, 2016.

HARRIS, B. F.; HARRIS, J.; REED, L.; ZELIHIC, M. M. Flipped classroom: Another tool for your pedagogy tool box. In: **Developments in Business Simulation and Experiential Learning: Proceedings of the Annual ABSEL conference**. [S.l.: s.n.], 2016. v. 43, n. 1.

HEROLD, M.; BOLINGER, J.; RAMNATH, R.; BIHARI, T.; RAMANATHAN, J. Providing end-to-end perspectives in software engineering. In: IEEE. **2011 Frontiers in Education Conference (FIE)**. [S.l.], 2011. p. S4B–1.

HEROLD, M. J.; LYNCH, T. D.; RAMNATH, R.; RAMANATHAN, J. Student and instructor experiences in the inverted classroom. In: IEEE. **2012 Frontiers in Education Conference Proceedings**. [S.l.], 2012. p. 1–6.

HSIA, L.-H.; LIN, Y.-N.; HWANG, G.-J. 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**, Wiley Online Library, v. 52, n. 4, p. 1771–1787, 2021.

HU, P. J.; CHAU, P. Y.; SHENG, O. R. L.; TAM, K. Y. Examining the technology acceptance model using physician acceptance of telemedicine technology. **Journal of management information systems**, Taylor & Francis, v. 16, n. 2, p. 91–112, 1999.

HWANG, G.-J.; CHANG, S.-C.; SONG, Y.; HSIEH, M.-C. Powering up flipped learning: An online learning environment with a concept map-guided problem-posing strategy. **Journal of Computer Assisted Learning**, Wiley Online Library, 2020.

HWANG, G.-J.; CHEN, C.-H.; CHEN, W.-H. A concept mapping-based prediction-observation-explanation approach to promoting students' flipped learning achievements and perceptions. **Educational technology research and development**, Springer, p. 1–20, 2022.

JOHNSON, P. Design and evaluation of an "athletic" approach to software engineering education. **ACM Transactions on Computing Education (TOCE)**, ACM New York, NY, USA, v. 19, n. 4, p. 1–25, 2019.

JOHNSON, P.; PORT, D.; HILL, E. An athletic approach to software engineering education. In: IEEE. **2016 IEEE 29th International Conference on Software Engineering Education and Training (CSEET)**. [S.l.], 2016. p. 8–17.

KAVCIC, A. The role of user models in adaptive hypermedia systems. In: IEEE. **2000 10th Mediterranean Electrotechnical Conference. Information Technology and Electrotechnology for the Mediterranean Countries. Proceedings. MeleCon 2000 (Cat. No. 00CH37099)**. [S.l.], 2000. v. 1, p. 119–122.

KEEFE, J. W. What is personalization? **Phi Delta Kappan**, SAGE Publications Sage CA: Los Angeles, CA, v. 89, n. 3, p. 217–223, 2007.

KERR, P. Adaptive learning. **Elt Journal**, Oxford University Press UK, v. 70, n. 1, p. 88–93, 2016.

KHOGALI, S. E.; LAIDLAW, J. M.; HARDEN, R. M. Study guides: a study of different formats. **Medical teacher**, Taylor & Francis, v. 28, n. 4, p. 375–377, 2006.

KIAT, P. N.; KWONG, Y. T. The flipped classroom experience. In: IEEE. **2014 IEEE 27th Conference on Software Engineering Education and Training (CSEE&T)**. [S.l.], 2014. p. 39–43.

KIM, M. K.; KIM, S. M.; KHERA, O.; GETMAN, J. The experience of three flipped classrooms in an urban university: An exploration of design principles. **The Internet and Higher Education**, Elsevier, v. 22, p. 37–50, 2014.

KING, A. From sage on the stage to guide on the side. **College teaching**, Taylor & Francis, v. 41, n. 1, p. 30–35, 1993.

KITCHENHAM, B. Procedures for performing systematic reviews. **Keele, UK, Keele University**, v. 33, n. 2004, p. 1–26, 2004.

KITCHENHAM, B.; MENDES, E.; TRAVASSOS, G. H. A systematic review of cross-vs. within-company cost estimation studies. In: BCS LEARNING & DEVELOPMENT LTD. **Proceedings of the 10th international conference on Evaluation and Assessment in Software Engineering**. [S.l.], 2006. p. 81–90.

KITCHENHAM, B. A.; PFLEEGER, S. L. Personal opinion surveys. In: **Guide to advanced empirical software engineering**. [S.l.]: Springer, 2008. p. 63–92.

KOCHUMARANGOLIL, J.; RENUMOL, G. Activity oriented teaching strategy for software engineering course: An experience report. **Journal of Information Technology Education: Innovations in Practice**, Informing Science Institute, v. 17, n. 1, p. 181–200, 2018.

KRATHWOHL, D. R. A revision of bloom's taxonomy: An overview. **Theory into practice**, Taylor & Francis, v. 41, n. 4, p. 212–218, 2002.

LEE, J.; KOTONYA, G.; WHITTLE, J.; BULL, C. Software design studio: a practical example. In: IEEE. **2015 IEEE/ACM 37th IEEE International Conference on Software Engineering**. [S.l.], 2015. v. 2, p. 389–397.

LIKERT, R. A technique for the measurement of attitudes. **Archives of psychology**, 1932.

LIMA, J.; ALENCAR, F.; SANTOS, W. A preliminary guide for assertive selection of active methodologies in software engineering education. In: **Brazilian Symposium on Software Engineering**. [S.l.: s.n.], 2021. p. 170–179.

LIN, H.-C.; HWANG, G.-J.; HSU, Y.-D. Effects of asq-based flipped learning on nurse practitioner learners' nursing skills, learning achievement and learning perceptions. **Computers & Education**, Elsevier, v. 139, p. 207–221, 2019.

LIN, Y.-N.; HSIA, L.-H.; HWANG, G.-J. Promoting pre-class guidance and in-class reflection: A sqirc-based mobile flipped learning approach to promoting students' billiards skills, strategies, motivation and self-efficacy. **Computers & Education**, Elsevier, v. 160, p. 104035, 2021.

LIN, Y.-T. Impacts of a flipped classroom with a smart learning diagnosis system on students' learning performance, perception, and problem solving ability in a software engineering course. **Computers in Human Behavior**, Elsevier, v. 95, p. 187–196, 2019.

LUBURIĆ, N.; SLADIĆ, G.; SLIVKA, J.; MILOSAVLJEVIĆ, B. A framework for teaching security design analysis using case studies and the hybrid flipped classroom. **ACM Transactions on Computing Education (TOCE)**, ACM New York, NY, USA, v. 19, n. 3, p. 1–19, 2019.

MAHASNEH, O. M. 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**, v. 28, 2020.

MAHER, M. L.; LATULIPE, C.; LIPFORD, H.; RORRER, A. Flipped classroom strategies for cs education. In: ACM. **Proceedings of the 46th ACM Technical Symposium on Computer Science Education**. [S.l.], 2015. p. 218–223.

MARQUES, M.; ROBLEDO, J. What software engineering “best practices” are we teaching students-a systematic literature review. In: IEEE. **2018 IEEE Frontiers in Education Conference (FIE)**. [S.l.], 2018. p. 1–8.

MARQUES, M. R.; QUISPE, A.; OCHOA, S. F. A systematic mapping study on practical approaches to teaching software engineering. In: IEEE. **2014 IEEE Frontiers in Education Conference (FIE) Proceedings**. [S.l.], 2014. p. 1–8.

MARTÍN, J. L.; FRÍAS, Z.; GALLARDO, C.; LÓPEZ, J. S.; GARCÍA, P. M.; ABAD, C. G.; DURTUS, B. S.; LECEA, M. L. P. M. d. Learning to develop software with flipped classroom. **Advanced Science Letters**, American Scientific Publishers, v. 24, n. 4, p. 2625–2629, 2017.

MASON, G. S.; SHUMAN, T. R.; COOK, K. E. Comparing the effectiveness of an inverted classroom to a traditional classroom in an upper-division engineering course. **IEEE transactions on education**, IEEE, v. 56, n. 4, p. 430–435, 2013.

MATALONGA, S.; MOUSQUÉS, G.; BIA, A. Deploying team-based learning at undergraduate software engineering courses. In: IEEE. **2017 IEEE/ACM 1st International Workshop on Software Engineering Curricula for Millennials (SECM)**. [S.l.], 2017. p. 9–15.

MAXIM, B.; LIMBAUGH, T.; YACKLEY, J. Socially distant active learning and student engagement in software engineering courses. In: **2022 ASEE Annual Conference & Exposition**. [S.l.: s.n.], 2022.

MAXIM, B. R.; LIMBAUGH, T.; YACKLEY, J. J. Student engagement in an online software engineering course. In: IEEE. **2021 IEEE Frontiers in Education Conference (FIE)**. [S.l.], 2021. p. 1–9.

MCCONNELL, J. J. Active learning and its use in computer science. **ACM SIGCSE Bulletin**, ACM, v. 28, n. SI, p. 52–54, 1996.

MCLAUGHLIN, J. E.; GRIFFIN, L. M.; ESSERMAN, D. A.; DAVIDSON, C. A.; GLATT, D. M.; ROTH, M. T.; GHARKHOLONAREHE, N.; MUMPER, R. J. Pharmacy student engagement, performance, and perception in a flipped satellite classroom. **American journal of pharmaceutical education**, American Journal of Pharmaceutical Education, v. 77, n. 9, 2013.

MUÑOZ, J. L. R.; OJEDA, F. M.; JURADO, D. L. A.; PEÑA, P. F. P.; CARRANZA, C. P. M.; BERRÍOS, H. Q.; MOLINA, S. U.; FARFAN, A. R. M.; ARIAS-GONZÁLES, J. L.; VASQUEZ-PAUCA, M. J. Systematic review of adaptive learning technology for learning in higher education. **Eurasian Journal of Educational Research**, v. 98, n. 98, p. 221–233, 2022.

OUHBI, S.; POMBO, N. Software engineering education: Challenges and perspectives. In: IEEE. **2020 IEEE Global Engineering Education Conference (EDUCON)**. [S.l.], 2020. p. 202–209.

ÖZYURT, H.; ÖZYURT, Ö. Analyzing the effects of adapted flipped classroom approach on computer programming success, attitude toward programming, and programming self-efficacy. **Computer Applications in Engineering Education**, Wiley Online Library, v. 26, n. 6, p. 2036–2046, 2018.

PAEZ, N. M. A flipped classroom experience teaching software engineering. In: IEEE. **2017 IEEE/ACM 1st International Workshop on Software Engineering Curricula for Millennials (SECM)**. [S.l.], 2017. p. 16–20.

PASCHOAL, L. N.; OLIVEIRA, B. R.; NAKAGAWA, E. Y.; SOUZA, S. R. Can we use the flipped classroom model to teach black-box testing to computer students? In: **Proceedings of the XVIII Brazilian Symposium on Software Quality**. [S.l.: s.n.], 2019. p. 158–167.

PASCHOAL, L. N.; OLIVEIRA, M. M.; MELO, S. M.; BARBOSA, E. F.; SOUZA, S. R. Evaluating the impact of software testing education through the flipped classroom model in deriving test requirements. In: **Proceedings of the 34th Brazilian Symposium on Software Engineering**. [S.l.: s.n.], 2020. p. 570–579.

PATRICK, S.; KENNEDY, K.; POWELL, A. Mean what you say: Defining and integrating personalized, blended and competency education. **International Association for K-12 Online Learning**, ERIC, 2013.

PÉRAIRE, C. Dual-track agile in software engineering education. In: IEEE. **2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)**. [S.l.], 2019. p. 38–49.

PINTRICH, P. R. *et al.* A manual for the use of the motivated strategies for learning questionnaire (mslq). ERIC, 1991.

PUGSEE, P. Effects of using flipped classroom learning in object-oriented analysis and design course. In: IEEE. **2017 10th International Conference on Ubi-media Computing and Workshops (Ubi-Media)**. [S.l.], 2017. p. 1–6.

PUGSEE, P. Using of the flipped classroom learning with a workshop activity in object-oriented analysis and design course. **Adv Sci Technol Engin Syst J**, v. 3, p. 52–63, 2018.

RAJ, N. S.; RENUMOL, V. A systematic literature review on adaptive content recommenders in personalized learning environments from 2015 to 2020. **Journal of Computers in Education**, Springer, v. 9, n. 1, p. 113–148, 2022.

ROEHL, A.; REDDY, S. L.; SHANNON, G. J. The flipped classroom: An opportunity to engage millennial students through active learning strategies. **Journal of Family & Consumer Sciences**, v. 105, n. 2, p. 44–49, 2013.

ROSING, M. von; WHITE, S.; CUMMINS, F.; MAN, H. de. **Business Process Model and Notation-BPMN.** 2015.

RUNESON, P.; HOST, M.; RAINER, A.; REGNELL, B. **Case study research in software engineering: Guidelines and examples**. [S.l.]: John Wiley & Sons, 2012.

RYAN, M. D.; REID, S. A. Impact of the flipped classroom on student performance and retention: A parallel controlled study in general chemistry. **Journal of Chemical Education**, ACS Publications, v. 93, n. 1, p. 13–23, 2016.

SCHEFER-WENZL, S.; MILADINOVIC, I. Game changing mobile learning based method mix for teaching software development. In: **Proceedings of the 16th World Conference on Mobile and Contextual Learning**. [S.l.: s.n.], 2017. p. 1–7.

SCHEFER-WENZL, S.; MILADINOVIC, I. Leveraging collaborative mobile learning for sustained software development skills. In: SPRINGER. **International Conference on Interactive Collaborative Learning**. [S.l.], 2018. p. 157–166.

SCHEFER-WENZL, S.; MILADINOVIC, I. Mobile distance learning driven software development education. **International Journal of Information and Education Technology**, v. 8, n. 5, 2018.

SCHOBER, P.; BOER, C.; SCHWARTE, L. A. Correlation coefficients: appropriate use and interpretation. **Anesthesia & analgesia**, Wolters Kluwer, v. 126, n. 5, p. 1763–1768, 2018.

SCHULTZ, D.; DUFFIELD, S.; RASMUSSEN, S. C.; WAGEMAN, J. Effects of the flipped classroom model on student performance for advanced placement high school chemistry students. **Journal of chemical education**, ACS Publications, v. 91, n. 9, p. 1334–1339, 2014.

SETERS, J. V.; OSSEVOORT, M.; TRAMPER, J.; GOEDHART, M. J. The influence of student characteristics on the use of adaptive e-learning material. **Computers & Education**, Elsevier, v. 58, n. 3, p. 942–952, 2012.

SHI, Y.; YANG, H.; MACLEOD, J.; ZHANG, J.; YANG, H. H. College students' cognitive learning outcomes in technology-enabled active learning environments: A meta-analysis of the empirical literature. **Journal of Educational Computing Research**, SAGE Publications Sage CA: Los Angeles, CA, p. 0735633119881477, 2019.

SIMPSON, R.; STORER, T. Experimenting with realism in software engineering team projects: An experience report. In: IEEE. **2017 IEEE 30th Conference on Software Engineering Education and Training (CSEE&T)**. [S.l.], 2017. p. 87–96.

SRICHAILARD, P. The development learning model of flipped classroom with digital storytelling for department of computer education. **Kasetsart Journal of Social Sciences**, v. 44, n. 3, p. 929–938, 2023.

STRELAN, P.; OSBORN, A.; PALMER, E. The flipped classroom: A meta-analysis of effects on student performance across disciplines and education levels. **Educational Research Review**, Elsevier, p. 100314, 2020.

SU, C.-Y.; CHEN, C.-H. Investigating the effects of flipped learning, student question generation, and instant response technologies on students' learning motivation, attitudes, and engagement: A structural equation modeling. **EURASIA Journal of Mathematics, Science and Technology Education**, Modestum Publishing LTD, v. 14, n. 6, p. 2453–2466, 2018.

SU, T.; DENG, S.; XU, X.; LI, D.; TU, Z. Principled flipped learning paradigm for laboratory courses in software engineering. In: **Software Engineering Education Going Agile**. [S.l.]: Springer, 2016. p. 123–128.

SUREKA, A.; GUPTA, M.; SARKAR, D.; CHAUDHARY, V. A case-study on teaching undergraduate-level software engineering course using inverted-classroom, large-group, real-client and studio-based instruction model. **arXiv preprint arXiv:1309.0714**, 2013.

TEINIKER, E.; SEUCHTER, G. Improving the flipped classroom model by the use of inductive learning. In: IEEE. **2020 IEEE Global Engineering Education Conference (EDUCON)**. [S.l.], 2020. p. 512–520.

TOWEY, D. Lessons from a failed flipped classroom: The hacked computer science teacher. In: IEEE. **2015 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)**. [S.l.], 2015. p. 11–15.

TROYA, J.; PAREJO, J. A.; SEGURA, S.; GÁMEZ-DÍAZ, A.; MÁRQUEZ-CHAMORRO, A. E.; RÍO-ORTEGA, A. del. Flipping laboratory sessions in a computer science course: An experience report. **IEEE Transactions on Education**, IEEE, v. 64, n. 2, p. 139–146, 2021.

VASILACHE, S. Suddenly online: Active learning implementation strategies during remote teaching of a software engineering course. In: SPRINGER. **International Conference on Interactive Collaborative Learning**. [S.l.], 2021. p. 395–402.

VERAS, N. L.; ROCHA, L. S.; VIANA, W. Flipped classroom in software engineering: A systematic mapping study. In: **Proceedings of the 34th Brazilian Symposium on Software Engineering**. [S.l.: s.n.], 2020. p. 720–729.

VUJOVIĆ, V.; MAKSIMOVIĆ, M.; PERIŠIĆ, B. The different active learning strategies in software engineering and their effectiveness. In: **Proc. 7th ICERI**. [S.l.: s.n.], 2014. p. 3183–3193.

WATSON, W. R.; WATSON, S. L. Principles for personalized instruction. In: **Instructional-Design Theories and Models, Volume IV**. [S.l.]: Routledge, 2016. p. 93–120.

WIERINGA, R.; MAIDEN, N.; MEAD, N.; ROLLAND, C. Requirements engineering paper classification and evaluation criteria: a proposal and a discussion. **Requirements engineering**, Springer, v. 11, n. 1, p. 102–107, 2006.

WOHLIN, C. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In: CITESEER. **Proceedings of the 18th international conference on evaluation and assessment in software engineering**. [S.l.], 2014. p. 38.

WOHLIN, C.; RUNESON, P.; HÖST, M.; OHLSSON, M. C.; REGNELL, B.; WESSLÉN, A. **Experimentation in software engineering**. [S.l.]: Springer Science & Business Media, 2012.

WRIGHT, R. W.; BRAND, R. A.; DUNN, W.; SPINDLER, K. P. How to write a systematic review. **Clinical Orthopaedics and Related Research (1976-2007)**, LWW, v. 455, p. 23–29, 2007.

XIE, H.; CHU, H.-C.; HWANG, G.-J.; WANG, C.-C. Trends and development in technology-enhanced adaptive/personalized learning: A systematic review of journal publications from 2007 to 2017. **Computers & Education**, Elsevier, v. 140, p. 103599, 2019.

ZHAO, J.; HWANG, G.-J.; CHANG, S.-C.; YANG, Q.-f.; NOKKAEW, A. Effects of gamified interactive e-books on students' flipped learning performance, motivation, and meta-cognition tendency in a mathematics course. **Educational Technology Research and Development**, Springer, v. 69, n. 6, p. 3255–3280, 2021.

APPENDIX A – INFORMED TERM OF CONSENT (STUDENTS)

A informed term of consent was developed to clarify the roles of the students in the evaluation process (in Portuguese).

Termo de Consentimento Livre e Esclarecido

Título do estudo: ***Modelagem, desenvolvimento, avaliação e investigação empírica do ensino de Engenharia de Software por meio de aulas invertidas baseadas em roteiros personalizados de estudo.***

Pesquisador responsável: Nécio de Lima Veras
Orientador: Prof. Dr. Windson Viana
Co-orientador: Prof. Dr. Lincoln Rocha

Introdução
Você está sendo convidado(a) a participar de um estudo sobre o impacto do método da Sala de Aula Invertida com roteiros personalizados no ensino de Engenharia de Software. Este estudo tem como objetivo investigar como esse método afeta sua percepção, motivação, engajamento e aprendizagem. Antes de decidir se deseja participar, é importante que você compreenda o propósito da pesquisa e o que será esperado de você como participante.

Procedimentos

- Você participará de uma aula que emprega o método da Sala de Aula Invertida com roteiros de estudo personalizados no contexto do ensino de Engenharia de Software.
- Serão coletados dados relacionados às suas percepções, motivação, auto-eficácia e desempenho adquiridas a partir das experiências vivenciadas durante as aulas invertidas.
- Sua participação envolverá a conclusão de questionários, estudos em casa, participação em atividades práticas ou outras atividades de coleta de dados, conforme especificado ao longo do estudo.
- Seu envolvimento neste estudo pode incluir a utilização de materiais de ensino, bem como o compartilhamento de suas experiências nas aulas invertidas.
- Pede-se que as suas respostas sejam **sinceras, honestas** e reflitam suas percepções e aprendizagem obtidas durante o estudo.

Confidencialidade
Todos os dados coletados serão mantidos estritamente confidenciais e serão usados apenas para fins de pesquisa. Seus dados não serão compartilhados com terceiros sem sua permissão explícita. Todos os dados serão armazenados de forma segura e não identificável. Nenhuma informação pessoal será armazenada e todos os dados serão tratadas com cuidado.

Riscos e Benefícios
Não são esperados riscos significativos ao participar deste estudo. No entanto, os benefícios potenciais incluem uma melhor compreensão da Engenharia de Software, especificamente nos assuntos relacionados com a aula invertida, além da oportunidade de contribuir para o avanço da pesquisa educacional.

Participação Voluntária
Sua participação neste estudo é totalmente voluntária. Você tem o direito de recusar-se a participar ou retirar-se a qualquer momento, sem qualquer penalização. Sua decisão de participar ou não, ou de retirar-se, não terá impacto nas suas relações com o curso ou com os pesquisadores envolvidos. Os resultados farão parte da tese de doutorado do professor Me. Nécio de Lima Veras, bem como, de um relato de experiência sobre o ensino de Engenharia de Software usando o método da Sala de Aula Invertida.

Contato
Se você tiver alguma dúvida sobre o estudo ou seus direitos como participante, entre em contato com o pesquisador responsável, Nécio de Lima Veras, pelo e-mail necio.veras@ifce.edu.br.

Consentimento
Eu li e entendi as informações acima e concordo em participar do estudo. Eu entendo que minha participação é voluntária e que posso retirar meu consentimento a qualquer momento, sem qualquer penalização. Ao continuar, você garante que concorda com os termos acima.

( ) Concordo em participar do estudo.
( ) Não concordo em participar do estudo.

Assinatura do Participante: ________________________________ Data: ______________

APPENDIX B – INFORMED TERM OF CONSENT (PROFESSORS)

A informed term of consent was developed to clarify the roles of the professors in the interview process (in Portuguese).

Termo de Consentimento Livre e Esclarecido

Título do estudo: **Modelagem, desenvolvimento, avaliação e investigação empírica do ensino de Engenharia de Software por meio de aulas invertidas baseadas em roteiros personalizados de estudo.**

Pesquisador responsável: Prof. Me. Nécio de Lima Veras
Orientador: Prof. Dr. Windson Viana
Co-orientador: Prof. Dr. Lincoln Rocha

Introdução
Você está sendo convidado(a) a participar de um estudo sobre o impacto do método da Sala de Aula Invertida com roteiros personalizados no ensino de Engenharia de Software. Este estudo tem como objetivo investigar como esse método afeta sua percepção em relação ao uso e aceitação da tecnologia FCTool. Antes de decidir se deseja participar, é importante que você compreenda o propósito da pesquisa e o que será esperado de você como participante.

Procedimentos

- Você participará de uma entrevista sobre o emprego do método da Sala de Aula Invertida no contexto do ensino de Engenharia de Software.
- Serão coletados dados relacionados às suas experiências anteriores adquiridas a partir de vivências durante aulas invertidas.
- Sua participação envolverá a construção de um roteiro de estudo fictício usando a ferramenta FCTool e, posteriormente, o preenchimento de um formulário online de aceitação de tecnologia
- Seu envolvimento neste estudo pode incluir o compartilhamento de suas experiências nas aulas invertidas.
- Pede-se que as suas respostas sejam sinceras, honestas e reflitam suas percepções obtidas durante a entrevista.

Confidencialidade
Todos os dados coletados serão mantidos estritamente confidenciais e serão usados apenas para fins de pesquisa. Seus dados não serão compartilhados com terceiros sem sua permissão explícita. Todos os dados serão armazenados de forma segura e não identificável (**anonimizados**). Nenhuma informação pessoal será armazenada e todos os dados serão tratadas com cuidado.

Riscos e Benefícios
Não são esperados riscos significativos ao participar deste estudo. No entanto, os benefícios potenciais incluem a oportunidade de contribuir para o avanço da pesquisa educacional no ensino de Engenharia de Software, bem como, a chance de usar a ferramenta FCTool de forma prioritária quando a mesma estiver disponível para amplo uso.

Participação Voluntária
Sua participação neste estudo é totalmente voluntária. Você tem o direito de recusar-se a participar ou retirar-se a qualquer momento, sem qualquer penalização. Sua decisão de participar ou não, ou de retirar-se, não terá impacto nas suas relações com os pesquisadores envolvidos. Os resultados farão parte da tese de doutorado do professor Me. Nécio de Lima Veras sobre o ensino de Engenharia de Software usando o método da Sala de Aula Invertida.

Contato
Se você tiver alguma dúvida sobre o estudo ou seus direitos como participante, entre em contato com o pesquisador responsável, Nécio de Lima Veras, pelo e-mail necio.veras@ifce.edu.br.

Consentimento
Eu li e entendi as informações acima e concordo em participar do estudo. Eu entendo que minha participação é voluntária e que posso retirar meu consentimento a qualquer momento, sem qualquer penalização. Ao continuar, você garante que concorda com os termos acima.

( ) Concordo em participar do estudo.
( ) Não concordo em participar do estudo.

Assinatura do Participante: ________________________________ Data: ______________