Educator Perceptions of DevOps Teaching Recommendations and Their Alignment with Common Challenges

Marcelo Fernandes
DIATINF, Federal Institute of Rio Grande do Norte
Natal, Brasil
marcelo.fernandes@ifrn.edu.br

Pablo Paiva
DIMAp, Federal University of Rio Grande do Norte
Natal, Brasil
pablo.paiva.123@ufrn.edu.br

Samuel Ferino
SSC, Monash University
Melbourne, Australia
samuel.demouraferino@monash.edu

Roberta Coelho
DIMAp, Federal University of Rio Grande do Norte
Natal, Brasil
roberta@dimap.ufrn.br

Eduardo Aranha
DIMAp, Federal University of Rio Grande do Norte
Natal, Brasil
eduardoaranha@dimap.ufrn.br

Uirá Kulesza
DIMAp, Federal University of Rio Grande do Norte
Natal, Brasil
uira@dimap.ufrn.br

Christoph Treude
SCIS, Singapore Management University
Singapore, Singapore
ctreude@smu.edu.sg

ABSTRACT

DevOps education presents unique pedagogical challenges due to the diversity of tools, rapid technological change, and the multidisciplinary nature of the field. Although previous work has proposed recommendations to address these challenges, it is unclear how educators perceive these recommendations and whether they align with the challenges encountered in practice. In this paper, we present a quantitative and qualitative methods study involving 11 DevOps educators who interacted with *Improve*, a tool that presents a curated set of educational challenges and recommendations derived from previous literature. Educators indicated which recommendations they already use, which they intend to use, and which challenges they experience and are motivated to address. Our findings show that 22.6% of the recommendations were new to educators and considered potentially useful, while 59.2% were already in use. Additionally, 66.3% of the challenges were considered relevant, with most of them having linked recommendations that educators were already using or willing to adopt. This study provides empirical insights into the perceived usefulness of existing recommendations, identifies gaps in challenge-recommendation mappings, and supports future efforts to design and disseminate more targeted educational guidance for DevOps teaching.

KEYWORDS

DevOps, Education, Qualitative and Quantitative Methods, User Study

1 Introduction

DevOps practice has become an essential approach to accelerate and improve continuous software delivery by fostering a collaborative culture between the development and operations teams [6, 19]. The increasing adoption of DevOps in educational settings reflects the need to prepare professionals capable of working in an increasingly dynamic and multidisciplinary technological environment [24]. However, teaching DevOps poses significant challenges due to the variety of tools (e.g., Docker, Kubernetes, GitHub Actions, Prometheus, Ansible), concepts, and practices (e.g., Versioning, Infrastructure as Code, Continuous Delivery, Observability, Chaos Engineering) involved, as well as the fast evolution of the field [1]. For example, DevOps educators [8] have mentioned *"difficulty in setting up the environment"* and also face the challenge that *"the multidisciplinary nature of DevOps is hard to deal with"*.

Although previous work [8, 12] has proposed recommendations to address these challenges, it is unclear how educators perceive these recommendations and whether they align with the challenges encountered in practice. This paper presents an empirical study with eleven DevOps educators who share both the challenges they face and the recommendations they follow for teaching and disseminating the topic through courses, lectures, workshops, or other training formats.

A central aspect of this study is to investigate the perceptions of DevOps educators about open challenges and recommendations from the DevOps Education literature [8, 12]. Furthermore, our study also discusses the remaining open challenges and provides a detailed overview of the difficulties encountered in educational practice and the strategies adopted to mitigate them.

By exploring developers' perceptions regarding the documented challenges and recommendations on DevOps Education, and by discussing their own teaching experiences, our study contributes to the development of more effective DevOps teaching methodologies aligned with the realities faced by professionals in the field. In doing so, it aims to strengthen the training of professionals who are prepared to meet the demands of today's rapidly evolving digital technology market.

The remainder of this paper is organized as follows. Section 2 details the study settings, including research questions and methodology. Section 3 presents the results of our empirical study. Section 4 discusses the implications of our findings for educators and researchers. Section 5 addresses the threats to the validity of our study. Section 6 provides an overview of related work. Finally, Section 7 concludes the paper, summarizes our contributions, and proposes directions for future work.

2 Study Settings

Our study aims to analyze the usefulness of documented challenges and recommendations related to DevOps education for educators in the context of their DevOps courses [11, 12]. We used the *Improve* system [10] to present the challenges and recommendations to deal with them. The research questions of our study are as follows.

- **RQ1**: *How helpful do DevOps educators find the reported recommendations for DevOps education?*
- **RQ2**: *To what extent do the reported challenges in DevOps education affect DevOps educators?*

To answer our research questions, we conducted an empirical study with 11 DevOps educators and used both quantitative and qualitative analysis to interpret the data collected. To support the quantitative analysis we used *Improve* tool. *Improve* is a web-based recommendation system [10] that showsllenges and their links to rtondations on DevOps EducatioEducation collects educatorseducators'ut them. Ultimately, it uses user feedback to show specific recommendations for improving the teaching of DevOps courses. Our analysis focused on (i) identifying challenges that educators want to mitigate, along with the linked recommendations they use or plan to use, and (ii) identifying recommendations and teaching methods that educators do not currently use, but intend to adopt.

The design of this study consists of two main phases. First, the study participants completed a practical session using the *Improve* system, followed by a semi-structured interview [4] with DevOps educators. The practical session presented challenges and recommendations for DevOps education [8, 12]; and the interview-based phase focused on getting in-depth descriptions of teachers' experiences and responses on phase 1.

Six researchers conducted this study: three PhD researchers, one PhD student, and two undergraduate researchers, all with experience in DevOps. The following subsections describe the two phases of our study design. In addition, we conducted a warm-up session with three other researchers from our group to validate the workflow of the practical session.

2.1 Practical Sessions and Interviews

Participant Selection Process. The study participants were selected from DevOps courses at seven Brazilian educational institutions: UFRN, UFCG, UnB, IFRN, IFPB, IFPE, and IFB. Most of the participants were recruited through advertisements sent to educators' email lists (from different institutions). Others were selected through convenience sampling based on the researchers' personal contacts.

Our criteria for selecting interviewees follow those established by Fernandes et al. [12]. Specifically, participants were required to have actively participated in delivering a DevOps-focused course and to have taught the course within the past two years, ensuring that their teaching experience was recent and up to date.

As a result, 14 educators accepted the invitation to participate in the study. However, three were excluded after it was determined that they did not meet the second criterion due to a lack of DevOps teaching experience. This resulted in a final sample of 11 participants. Table 1 presents key information about the DevOps educators who participated in the study, while Table 2 shows information regarding their DevOps courses.

Practical Session. In order to enable a quantitative analysis regarding the usefulness of existing challenges and recommendations every participant in our study used the *Improve* tool. *Improve* organizes the existing challenges and recommendations in seven themes: Environment Setup, Tools and Technology, Class Preparation, Curriculum, DevOps Concepts, Pedagogy, and Strategies in Course Execution.

We conducted practical sessions via *Google Meet* organized in two meetings. The activity lasted, on average, 49 minutes (min: 28 minutes; max: 74 minutes). In our first meeting, the interviewee talked about their own DevOps teaching experience (e.g., their DevOps course syllabus) and industry experience. On second meeting, the participant used *Improve* tool. S/he should chose two DevOps educational challenges and recommendations, and answer the questions related to them. All questions were Yes/No questions related to the presence of the challenge and usefulness of the related recommendations (next sections detail them). During this process, one of the authors stayed online with the participant in case of need to clear any participants' doubts, as well as to check if the interviewee provided all the required answers.

Post-Practical Session Interview. After this, we conducted interviews with the participants via *Google Meet* to complement the practical session. Before starting the recording, explicit verbal consent was required. The eleven interviews lasted, on average, 33 minutes (min: 19 minutes; max: 53 minutes). They usually occur three days after the first activity (min: 0 days; max: 6 days). The educators received recommendations to apply in their DevOps courses. They also warned us about open DevOps educational challenges, with no linked recommendations. We asked the educator to discuss his answers to questions about the recommendations, the teaching methods, and the challenges the tool shows on the Summary page. We also asked for new recommendations, teaching methods, and challenges that the educator knows and the tools that are not provided. In the end, we asked for their feedback about *Improve* usability. During the interview, the researchers identified the challenges, recommendations, and teaching methods that the interviewee talked about for the data extraction step.

2.2 Data Extraction and Analysis

We extracted the main quantitative results from the practical sessions through queries on the *Improve* database. The main qualitative results were obtained from the interview process, based on transcriptions and subsequent data analysis.

Queries on the database. We extracted the quantitative results from the practical session using the *Improve* database. Specifically, we applied queries to retrieve the number of "yes" or "no" responses to the challenge question "Will Mitigate" answered by the educators.

Table 1: Participants' Demographics (N=11).

Id	TExp	DevOpsTExp	IndEx	Source
P1	4	3	15	Personal contact
P2	12	6	0	Personal contact
P3	9	5	4	Advertisement
P4	16	3	24	Personal contact
P5	16	1	2	Advertisement
P6	14	3	5	Advertisement
P7	11	2	17	Advertisement
P8	13	4	17	Advertisement
P9	6	2	14	Advertisement
P10	12	4	10	Advertisement
P11	26	6	19	Advertisement

Key: TExp = General teaching experience (in years); DevOpsTExp = DevOps teaching experience (in years); IndEx = Industry experience (in years).

Table 2: Courses' information (N=11).

Id	Participant	Course Level	SBr	YCE
C1	P1	G	RN	2020-2022
C2	P2	U	RN	2018-2023
C3	P3	U	PB	2019-2023
C4	P4	U	RN	2021-2023
C5	P5	U	RN	2023
C6	P6	U	RN	2018; 2019; 2023
C7	P7	U	PE	2022-2023
C8	P8	G	RN	2020-2023
C9	P9	U	PB	2022-2023
C10	P10	G	RN	2020-2023
C11	P11	U	DF	2018-2023

Key: U = undergraduate degree; G = graduate degree; SBr = State in Brazil; YCE = Years of Course Editions.

In total, educators responded to this question 294 times. We also ran two additional queries on the challenge responses, grouping the results by participant and by challenge.

We also extracted the number of "yes" or "no" responses to the recommendation and teaching method questions, "Already Used" and "Will Use," answered by the educators. The "Already Used" recommendation question was answered 882 times, and the "Will Use" recommendation question was answered 360 times. For teaching methods, the educators answered the "Already Used" question 36 times and the "Will Use" question 15 times. We ran two additional queries on the recommendation responses, grouping them by participant and by recommendation. A similar analysis was performed for the teaching method responses.

We also combined some of the results, for example, linking the number of "Will Mitigate" responses for challenges with the number of "Already Used" or "Will Use" responses for the corresponding recommendations. Additionally, we queried the GUI log tables to examine the duration of each activity session and the time intervals between user interactions during the practical session.

Transcription. The interviews recorded via Google Meet were stored on Google Drive. We then used *Google Pinpoint*[1], a free tool for academic use, to streamline the transcription of the interviews conducted in Brazilian Portuguese. The automatically generated transcripts were subsequently reviewed and manually corrected using the same tool.

Data Extraction. Three researchers collaborated on the data extraction and analysis, while three additional researchers reviewed the results. At each step of the analysis, the researchers discussed and verified the results before proceeding. In cases of disagreement, researchers were encouraged to maintain their position unless one could not find the information that the other identified. If the disagreement persisted, an additional researcher was brought in to help resolve it.

We began by identifying snippets in the transcriptions that corresponded to the challenges, recommendations, teaching methods, and system usage discussed by the interviewees. These snippets were then coded, resulting in 75 comments about challenges, 123 about recommendations, 30 about teaching methods, and 26 about system usage. We employed the definition of challenge, recommendation, and teaching method from previous work [8, 12].

Ultimately, we identified and selected the snippets that explained or justified the use of each recommendation or teaching method. A similar analysis was conducted for the challenges and system usage. In total, we collected 28 helpful comments on 21 challenges, 65 comments on 44 recommendations, 19 comments on seven teaching methods, and 14 comments on system usage.

3 Results

We conducted 11 tool sessions and 11 complementary interviews with the selected educators. Each educator chose two themes to work with. The final distribution of selected themes was as follows: Environment Setup (36.4%), Tools and Technology (27.3%), Class Preparation (18.2%), Curriculum (9.1%), and Strategies in Course (4.5%). Together, Environment Setup, Tools and Technology, and Class Preparation accounted for 81% of the selections. Notably, the DevOps Concepts theme was not selected by any of the participants.

3.1 RQ1. How helpful do DevOps educators find the reported recommendations for DevOps education?

We categorized the responses to the "Already Use" and "Will Use" recommendation questions presented during the practical sessions into three groups:

- *Will Use*: recommendations for which the "Already Use" answer was **"no"** and the "Will Use" answer was **"yes"**. These indicate recommendations that educators have not yet used but intend to adopt in their courses.
- *Already Used*: recommendations for which the "Already Use" answer was **"yes"**. These indicate recommendations that educators are already using in their courses.

[1]Google Pinpoint at https://journaliststudio.google.com/pinpoint/about

- *Won't Use*: recommendations for which the "Already Use" answer was **"no"** and the "Will Use" answer was **"no"**. These indicate recommendations that educators are not interested in using.

We found that the *Will Use* group accounts for 22.6% of the recommendation responses. This is particularly relevant, as it indicates that educators discovered new recommendations that they intend to apply in their DevOps courses, suggesting potential for course improvement. The percentage of *Will Use* responses varies across participants. For example, P3, P5, and P7 each have values greater than 33%, while P2, P8, and P9 are below 10%.

The *Already Used* group accounts for 59.2% of the recommendation responses. In other words, educators already use most of the recommendations they reviewed, indicating that these recommendations are generally useful to them. We found that the *Already Used* group exceeds 50% for all participants except P1 and P5, and it is the largest group for each participant except P5.

The *Won't Use* group represents 18.3% of the recommendation responses, making it the smallest portion among the three categories. This suggests that a relatively small number of recommendations were considered not useful by the educators in our study. Our results show that P8 has more than 33% responses in this group, while P2 and P7 are below 10%.

3.1.1 Will Use Recommendations. The *Will Use* recommendations with the most responses are shown in Table 3. Four of them are related to tools such as Ansible[21], Terraform[16], and Katacoda. The remaining six involve different strategies for course execution, including the use of teaching assistants in labs, practices like PBL, inverted classroom, and Agile, the use of the Phoenix Project book, building whiteboard-free sessions, and teaching social coding.

Regarding the recommendation *Ansible as a deployment automation tool can be used in teaching DevOps*, P7 commented: "Ansible and Terraform are two interesting open-source and vendor-independent Infrastructure as Code tools. I still have some difficulty with them because of a lack of time."

The recommendation *Terraform as a deployment provisioning tool can be used in teaching DevOps* received two comments. P1 said, "I know Terraform. It creates the environment using configuration files. So, it would be a useful tool." P7 added, "Terraform for provisioning laboratory infrastructure automation, which is very useful for teaching."

3.1.2 Already Used Recommendations. Table 4 shows the most frequently selected *Already Used* recommendations. We highlight *Provide initial environment setup for students* and *Teaching method based on practical activities*, since both were selected by all 11 participants. The themes Environment Setup and Tool & Technology each include six of the most cited recommendations.

P8 and P9 commented on the recommendation *Teaching method based on practical activities*. P8 said, "I present DevOps concepts, so we work with some articles on DevOps deployment from the industry. We develop an application with a RESTful API using Spring Boot based on microservices concepts, and we deploy this application using Docker. The process begins manually and evolves into an automated process through CI/CD, already using associated DevOps concepts. And we continue to work on some specific tools."

P9 added, "I think the course has to be practical. I focus on applications and how they are affected by the underlying infrastructure that supports them. The course has eight practical activities, plus the project they develop in groups at the end of the course. So, they build the application, virtualize it, deploy it in a local environment, and apply it in a cloud environment. And they use it in a container environment with Kubernetes[14] and operate on top of them. So, they evolve the same application with practical activities."

P2, P5, P7, P9, and P10 commented on the recommendation *Use cloud provider services*, noting that they use AWS, Azure, and Google Cloud Platform (GCP), which offer professional resources suitable for experimentation.

P7, P9, and P10 also commented on the related recommendation *Use cloud provider services with student plans*, mentioning that AWS, GCP, Oracle, and Azure all offer free student tiers. P10 emphasized, "GCP does not require a credit card," and P7 added, "I use the AWS Academy and GCP Academy programs also."

3.1.3 Won't Use Recommendations. The *Won't Use* recommendations with the most responses are shown in Table 5. The top two are the recommendation to use the Phoenix Project book and the suggestion to use the Tuleap tool.

The recommendation *Do some research about the DevOps topic, write an essay, and if the tool is open source, contribute to it, fix some issues, and report it to the educators. The open source project should have over a hundred stars on GitHub* received two comments. P2 noted, "The students already have a project started in previous courses." P7 remarked, "I think it's irrelevant for a short-term technology course."

The recommendation *Exercise as many tools as possible* received two comments. P1 said, "There are a lot of tools, and the time is limited. It's impractical." P3 added, "It is not feasible. I believe one tool for each practice is enough for teaching purposes."

3.1.4 Conflicting Recommendation Answers. Some recommendations received a mix of *Won't Use*, *Will Use*, and *Already Used* responses from the same participant. We refer to these as conflicting answers, as they prevent us from clearly categorizing the recommendation into a single group. Although some educators find these recommendations valuable, others are not interested in using them. Table 6 lists all recommendations where such conflicts occurred. We limited the table to recommendations with up to three conflicting responses and summed the *Will Use* and *Already Used* answers.

For example, six educators indicated they would not use the recommendation *The Phoenix book by Jean Kim is a novel that covers the Ops side of DevOps*, while four others reported that they either plan to use it or are already using it.

The recommendation *Terraform as a deployment provisioning tool can be used in teaching DevOps* received 3 *Won't Use* responses and 5 *Already Used* or *Will Use* responses. P3 commented that they would not use it: "I want to focus more on using distributed and virtualized infrastructure than building it automatically." On the other hand, P1 said that they would use it: "I know Terraform. It is a tool that I already have some knowledge about. It is worth it because it already configures several things in the environment." Similarly, P4 stated: "Terraform for provisioning infrastructure, which is very useful for teaching and preparing laboratories."

Table 3: *Will Use* Recommendations with most answers (N=10).

Recommendation	Theme	Amount
-> Ansible as deployment automation tools can be used in teaching DevOps	Tool & Technology	7
-> Terraform as a deployment provisioning tool can be used in teaching DevOps	Tool & Technology	5
-> Teaching assistants are helpful with labs	Strategies in Course Execution	5
-> Use the Katacoda website to students create tutorials about tools	Tool & Technology	5
-> Merge good practices of Problem-Based Learning (PBL), inverted class, and Agile, through classroom experimentation	Strategies in Course Execution	5
-> Use a learning tool to ease the DevOps teaching	Tool & Technology	5
-> The Phoenix book by Jean Kim is a novel that covers the Ops side of DevOps	Strategies in Course Execution	4
-> Build whiteboard free sessions inspired by what students have failed and the two-hour exercise	Strategies in Course Execution	4
-> Teach social coding	Strategies in Course Execution	4
-> There is a specific support team to answer students' questions about the related infrastructure part	Strategies in Course Execution	4

Table 4: *Already Used* Recommendations with most answers (N=13).

Recommendation	Theme	Amount
-> Provide initial environment setup for students	Environment Setup	11
-> Teaching method based on practical activities	Strategies in Course Execution	11
-> Build scenarios that students can run on their own computer	Environment Setup	10
-> Try to make the environment setup minimal	Environment Setup	10
-> Teach using examples	Strategies in Course Execution	10
-> There are many free DevOps tools available	Tool & Technology	10
-> Start a class with a pre-organized structure	Environment Setup	10
-> Ask students to adopt the tools used by instructors	Tool & Technology	9
-> When using a tool to help teach, you must have a good command of it and the necessary permissions to deal well with it during its use in the discipline	Tool & Technology	9
-> Cloud computing makes it easier to stand up virtual machines	Environment Setup	9
-> Show the student that there are several ways and tools to do the task	Tool & Technology	9
-> It is necessary to choose which topics and tools are essential as the course time is limited	Class Preparation	9
-> Use cloud provider services	Tool & Technology	8

Table 5: *Won't Use* Recommendations with most answers (N=10).

Recommendation	Theme	Amount
-> The Phoenix book by Jean Kim is a novel that covers the Ops side of DevOps	Strategies in Course Execution	6
-> Use Tuleap for lifecycle management	Tool & Technology	6
-> Limit the zoom FPS rate to 10, avoiding excessive student and instructor resource consumption	Tool & Technology	5
-> DevOps tools are well integrated in Bluemix platform from IBM	Tool & Technology	5
-> There is a specific support team to answer students' questions about the related infrastructure part	Class Preparation	5
-> Teach social coding	Strategies in Course Execution	5
-> Delegate the responsibility for finding adequate infrastructure for the student	Environment Setup	5
-> Teaching assistants are helpful with labs	Strategies in Course Execution	4
-> Do some research about DevOps topic, write an essay, and if the tool is open source, contribute to that tool and fix some issues and report it to the educators. The open source project should have more than a hundred stars on Github	Class Preparation	4
-> Use the Katacoda website to students create tutorials about tools	Tool & Technology	4

Table 6: Conflicting Recommendation Answers (N=10).

Recommendation	Theme	A1	A2
-> The Phoenix book by Jean Kim is a novel that covers the Ops side of DevOps	Strategies in Course Execution	6	4
-> Teach social coding	Strategies in Course Execution	5	6
-> There is a specific support team to answer students' questions about the related infrastructure part	Class Preparation	5	4
-> Delegate the responsibility for finding adequate infrastructure for the student	Environment Setup	5	3
-> Teaching assistants are helpful with labs	Strategies in Course Execution	4	7
-> Use the Katacoda website to students create tutorials about tools	Tool & Technology	4	5
-> Use a complete example project from places such as a Java discussion forum	Strategies in Course Execution	3	7
-> Use cloud SAS providers to avoid spending a lot of time on installations and configurations	Environment Setup	3	6
-> Terraform as a deployment provisioning tool can be used in teaching DevOps	Tool & Technology	3	5
-> Build something portable that can be broken down into several pieces where one student runs one bit, and then another student runs the rest	Environment Setup	3	5

A1: Won't Use Amount; A2: Will Use plus Already Used Amount.

3.1.5 Teaching Methods Answers. We also asked recommendation questions related to the teaching methods identified by Ferino et al. [8]. We consider teaching methods as a special category of recommendations. The educators responded to the "Already Used" teaching method question 36 times and to the "Will Use" teaching method question 15 times.

We found that the *Will Use* group represents 30.6% of the teaching method responses. This is noteworthy, as it indicates that educators discovered new teaching methods they intend to apply in their DevOps courses, suggesting potential for course improvement. Six participants selected at least one teaching method in this group.

The *Already Used* group accounts for 63.9% of the teaching method responses. In other words, educators already use most of the teaching methods they reviewed, suggesting that these methods are generally useful in practice. We also found that the *Already Used* group was greater than or equal to 50% for each participant.

The *Won't Use* group represents 5.6% of the teaching method responses, making it the smallest portion. This suggests that only a few teaching methods were considered not useful by the educators who participated in our study.

Table 7 shows the distribution of teaching method responses. Collaborative Learning, Comprehensive Distance Learning, and Educational Support Tool are the most frequently mentioned methods in the *Will Use* group. In the *Already Used* group, Educational Support Tool and Collaborative Learning are the most common, with 7 and 5 responses, respectively.

Collaborative Learning received five comments. P1 and P2 stated that they plan to use it to promote collaboration among students. P5, P9, and P10 already use it in their courses. P5 said, "It is a reality they will encounter when they work." P9 noted, "They can help each other when they are part of a group. They share the activities in the laboratory. I realize that it helps them a lot to develop their knowledge." P10 added, "The course is linked to some project. The students are collaboratively learning and developing projects there."

P9 stated that he *won't use* the teaching method Comprehensive Distance Learning: "I show concepts and expose a problem they need to solve in terms of maintaining an application. In a way, they are learning remotely and in isolation, based on examples or defined problems."

Table 7: Teaching Methods with answers (N=11).

Teaching Method	WU	AU	WnU
Collaborative Learning	3	5	0
Comprehensive Distance Learning	2	1	1
Educational Support Tool	2	7	1
Problem-based Learning	1	3	0
Experimental Learning	1	0	0
Mentor	1	0	0
Personalized Learning	1	0	0
Project-based Learning	0	3	0
Lecture	0	2	0
Feedback Session	0	1	0
Labs	0	1	0

Key: WU = Will Use; AU = Already Used; WnU = Won't Use.

The Educational Support Tool received a range of comments. P5 plans to use it, stating, "I seek to work with environments that allow this process of experimentation, of carrying out practical activities so that the assimilation of the content I work on can be done in the best possible way."

P1, and P2 already use it. P1 said, "It makes it much easier to save time than when you have to configure everything manually." P2 commented, "The environment serves the purpose of training and testing various tools."

P8, on the other hand, said they *won't use* it: "I think that bringing another tool to support teaching-learning is not something that will improve the teaching-learning process."

Problem-Based Learning received two comments from educators who already use this teaching method. P2 said, "The students also

practice with me using the example I teach them. Then, they have a moment to apply it to their project." P7 explained, "I provide a list of problems, and students choose one to explore and develop. When the student is at the earliest stage, we use the project-based method. When they reach a higher maturity level, we use the problem-based approach."

P6, P7, and P9 commented that they already use Project-Based Learning. P6 said, "I like projects. I work with them in all my courses; each has a project." P7 explained, "When the student is at the earliest stage, we use the project-based method. When they are at a higher maturity level, we use the problem-based approach." P9 noted, "They develop the system together with the infrastructure, exploring the capabilities that each infrastructure or tool offers. I think this is great, as they don't need to switch between contexts and can understand the differences between each infrastructure and the scenarios they are running."

3.1.6 RQ1 Answer. Based on this study, we can conclude that the reported recommendations for DevOps education are indeed **useful** to DevOps educators. We found that **81.8%** of the responses to recommendation-related questions during the practical sessions indicated that the recommendations were **useful**. This percentage includes both the *Will Use* and *Already Used* groups.

Our study also revealed that **22.6%** of the responses indicated the recommendations were **new** to the educators and that they were **interested in applying** them to improve their courses. Additionally, **59.2%** of the responses indicated that the recommendations were **already in use**.

These findings suggest that both *Will Use* and *Already Used* recommendations could serve as valuable guidance for **new or less experienced DevOps educators**. Many comments from the interviews support and contextualize these findings.

Although there were fewer responses related to teaching methods compared to recommendations, the proportion of useful responses was even higher, at **94.5%**. Specifically, **30.6%** of the responses indicated that the teaching methods were **new** to experienced educators who **intend to apply** them to improve their courses, while **63.9%** indicated that the methods were **already in use**.

These findings suggest that both *Will Use* and *Already Used* teaching methods can serve as valuable guidance for **new or less experienced DevOps educators**.

3.2 RQ2. To what extent do the reported challenges in DevOps education affect DevOps educators?

We grouped the responses to the "Will Mitigate" challenge question and the "Already Use" and "Will Use" recommendation questions from the practical sessions into three categories:

- *Will Mitigate With Links*: challenges for which the "Will Mitigate" answer is **"yes"** and there is **at least one** linked recommendation with either "Already Use" = **"yes"** or "Will Use" = **"yes"**. These represent challenges that affect educators and are supported by useful linked recommendations for mitigation.
- *Will Mitigate Without Links*: challenges for which the "Will Mitigate" answer is **"yes"** but there are **no** linked recommendations with either "Already Use" = **"yes"** or "Will Use" = **"yes"**. These represent challenges that affect educators but currently lack useful linked recommendations.
- *Won't Mitigate*: challenges for which the "Will Mitigate" answer is **"no"**. These are considered not relevant or not impactful by the educators.

We found that the *Will Mitigate With Links* group accounts for 64.6% of the challenge responses. This is particularly relevant, as it shows that most of the challenges presented to educators were considered relevant and had useful linked recommendations. Educators can apply these recommendations in their courses to help mitigate the challenges, potentially leading to course improvement.

We also observed that the *Will Mitigate With Links* group was the largest category for all participants and exceeded 50% for each educator, except for P11. Notably, P7 had 100% of their responses in this category.

The *Will Mitigate Without Links* group accounts for 1.7% of the challenge responses. This indicates that the participants identified a small number of relevant challenges for which useful linked recommendations were not available. Only P1, P3, P4, and P6 had non-zero values in this category, and for each of them, the individual percentage was below 6%.

The *Won't Mitigate* group represents 33.7% of the challenge responses, meaning that one-third of the challenges were not considered relevant or useful by the educators in our study. P1 had a value above 50%, while P4, P5, and P7 had values below 16.5%.

3.2.1 Will Mitigate Challenges With Links. Table 8 shows the most frequently mentioned *Will Mitigate With Links* challenges. Three of the challenges, each mentioned eight times, are related to infrastructure constraints: *Limited computational resources*, *Cloud providers usage has limits*, and *Institutions' resources have limits*. The Environment Setup theme accounts for 9 of the 12 challenges listed.

Four educators commented on the challenge *Limited computational resources*. P2 and P11 mentioned that the laboratory computers are outdated and that they do not have permission to install the necessary tools. Additionally, P2, P10, and P11 reported that they did not have access to an institutional private cloud. To mitigate this challenge, P2 and P10 use public cloud services, although these come with limitations. P9 has access to an institutional private cloud but noted that it requires significant configuration effort.

Similarly, three educators discussed the challenge *Institutions' resources have limits*. P2, P10, and P11 reported facing hardware and software restrictions in their laboratories. They do not have the necessary permissions to install software and lack sufficient hardware resources to fully explore the environments in the labs.

P2 also commented, "We use the Azure cloud environment. We do not have a team to answer questions related to this infrastructure. We do not even have a budget."

To mitigate this challenge, P10 uses Docker, stating, "With Docker, I can start up a database."

3.2.2 Will Mitigate Challenges Without Links. We found only two challenges that educators wanted to mitigate but for which they did

Table 8: Challenges with most *Will Mitigate With Links* answers (N=12).

Challenge	Theme	Amount
-> Limited computational resources	Environment Setup	8
-> Cloud providers usage has limits	Environment Setup	8
-> Institutions' resources have limits	Environment Setup	8
-> Setting up the infrastructure is difficult	Environment Setup	7
-> There is a large number of DevOps tools	Tool & Technology	6
-> It's challenging to find the right sized examples to teach DevOps	Tool & Technology	6
-> Environment set up in a cloud service cost money	Environment Setup	6
-> Debugging lab sessions are very difficult	Environment Setup	6
-> There was no automated environment setup tool to support the student	Environment Setup	5
-> Difficulty in understanding environment, tools and network configuration	Environment Setup	5
-> Lab environment preparation requires a lot of time	Environment Setup	5
-> It is hard to prepare a robust and simple technology stack	Tool & Technology	5

not find any useful linked recommendations. P1, P3, and P6 selected the challenge *Small examples were really unsatisfactory*, which is linked to the recommendation *Use a complete example project from places such as a Java discussion forum*. P4 selected the challenge *Katacoda is incompatible with some specific tools*, which is linked to the recommendation *Let the students choose the technology stack used in their systems*.

In both cases, each challenge had only one linked recommendation, and the educators did not select that recommendation as useful for mitigation.

3.2.3 Won't Mitigate Challenges. Table 9 presents the *Won't Mitigate* challenges that received the most responses. Three challenges received six responses each: *Teaching operational activities are ignored because it is hard*, *You need a lot of interconnected machines running different services with visibility on each other to perform continuous deployment*, and *Bamboo continuous integration fails with 120 students running pipelines at the same time*.

P10 commented on the challenge *You need a lot of interconnected machines running different services with visibility on each other to do continuous deployment*, stating, "For example, we can create a cluster with one click using Google Cloud Platform. It creates the cluster for us automatically, so it hasn't been a problem."

The challenge *Preparing the lab environment requires a lot of time* was considered irrelevant by P10, who said, "It hasn't bothered me because I've been using cloud infrastructure a lot. So, the student rarely needs to set up a local environment."

3.2.4 Conflicting Challenge Answers. Some challenges received both *Won't Mitigate* and *Will Mitigate With Links* responses from different participants. We refer to these as conflicting answers because they prevent us from clearly categorizing the challenge. These cases highlight challenges that some educators do not consider relevant, while others actively seek to address them.

Due to space limitations, we cannot present all challenges with conflicting responses here; additional details can be found in [13]. Below, we provide a few examples of conflicting challenge responses.

For example, four educators indicated they would not mitigate the challenge *There was no script for the student on how to install the tools used during the course*, while four others indicated they would.

P2 commented, "The configured environment of a cloud service costs money; it is expensive," in reference to the challenge *There was no automated environment setup tool to support the student.*

Regarding the challenge *Difficulty in understanding environment, tools, and network configuration*, P11 said, "Our curriculum has to evolve. Students haven't studied networks. They don't know protocols."

P4 commented on the challenge *Getting all DevOps tools to work together is challenging*, stating, "I must understand Git, integration, deployment, and testing. This is tight within the course schedule to work with all the tools."

3.2.5 RQ2 Answer. Our study investigated how the reported challenges **impact** DevOps educators.

A total of **66.3%** of the responses to challenge-related questions during the practical sessions indicated that the challenges **do impact** DevOps educators. This percentage represents the combined total of the *Will Mitigate With Links* and *Will Mitigate Without Links* groups.

Specifically, **64.6%** of the responses show that educators **want to mitigate** the challenges and either **already use** or **plan to use** linked recommendations. An additional **1.7%** of responses indicate that educators **want to mitigate** the challenges but **do not currently use** and **do not intend to use** any linked recommendations. Comments from the interviews support and help contextualize these findings.

3.3 Feedback for the *Improve* Tool

We received feedback on the use of the *Improve* tool from study participants, including several suggestions for improvement.

P2 requested "more detailed tips, for example, a specific section regarding the creation of pipelines and environment configuration with specific tools."

P5 and P7 suggested providing complementary materials for each recommendation. P5 elaborated: "For example, I will catalog a good practice for teaching Kanban. When I click there, I will see

Table 9: Challenges with most *Won't Mitigate* answers (N=9).

Challenge	Theme	Amount
-> Teach operational activities are ignored because it is hard	Environment Setup	6
-> You need a lot of interconnected machines running different services with visibility on each other to do continuous deployment	Environment Setup	6
-> Bamboo continuous integration fails with 120 students running pipeline at the same time	Tool & Technology	6
Katacoda is incompatible with some specific tools	Tool & Technology	5
-> There was no script for the student on how to install the tools used during the course	Environment Setup	4
-> The process of making students migrate to other tools is hard	Environment Setup	4
-> The student has difficulty realizing the importance of setting the environment	Environment Setup	4
-> VirtualBox has limitation in MacOS	Tool & Technology	4
-> It's difficult to use Jira lifecycle management tool because of its license model	Tool & Technology	4

examples, videos, links, and materials to help me. After identifying the problem, I will see proposed solutions with examples or materials I could consult. A compendium of repositories that will help the educator apply a practice."

P6 requested the availability of a Portuguese version of the tool.

There were also several positive comments from participants about the tool. P2 noted, "Very easy to answer because they were basically yes or no," and also commented, "You realize that the information about challenges and recommendations is well linked. The system saves the answers you have already given before." P11 remarked, "The form is very complete. It worked well."

Some educators also reported difficulties using the tool. P9 said, "Sometimes it was slow to load, but nothing that bothered me." P1 and P3 shared similar concerns about the system's responsiveness. P4 commented, "Some usability difficulties for those using it for the first time. The system has many options."

4 Discussion

In this section, we discuss our results and their implications for both educators and researchers.

Implications for Educators. The results of the study highlight the usefulness of the extracted recommendations in supporting the improvement of DevOps courses, with 23% of the responses falling into the *Will Use* category and 59% into the *Already Used* category. Educators can refer to the recommendations and teaching methods ranked by *Will Use* and *Already Used* responses, as presented in Tables 3 and 4, to explore ideas that may benefit their own courses.

Educators may also review participant comments to gain insights into how certain recommendations can be applied in practice and why they are considered relevant. For example, P8 commented on the recommendation *Teaching method based on practical activities*: "We work on some articles about DevOps deployment in the industry, we develop a RESTful API with Spring Boot based on microservices concepts, and we deploy this API through Docker. The process begins manually and evolves into an automated process through CI/CD, using associated DevOps concepts."

The results also suggest that the reported challenges are useful for identifying areas of improvement in DevOps courses, with 65% of the responses falling into the *Will Mitigate With Links* category. Educators can review the challenges ranked by *Will Mitigate With Links* responses in Table 8 to assess their relevance to their own courses. They may then consider using the linked recommendations to help mitigate these challenges.

Educators can also gain insights into how to address specific challenges by reviewing participant comments. For example, regarding the challenge *Difficulty in using multiple materials to create the classes*, P9 commented: "I think the DevOps and Site Reliability Engineer (SRE) areas have a lot of information spread out. You have company materials, you have books, and it depends on your local problem and how you apply DevOps and SRE techniques. I think, in essence, I shouldn't use a core book. I even said that I wasn't going to use a basic book because I believe that there are several very good materials in specific areas. I prefer to work with various materials."

We also identified that the Katacoda platform [3] is no longer publicly available. This represents new challenges to educators in setting up a suitable learning environment for the students.

Implications for Researchers. Our study focused on evaluating the usefulness of reported challenges and recommendations in contributing to the improvement of DevOps courses. The complementary interviews provided additional insights regarding: (i) the difficulty of addressing DevOps-specific challenges; (ii) the perceived usefulness and relevance of the recommendations in mitigating those challenges; and (iii) the effectiveness of the association links between challenges and recommendations. These aspects merit further investigation to deepen our understanding of DevOps education. Although we relied on a consistent set of challenges, recommendations, and association links derived from existing studies, there is still room to improve this dataset. In particular, we identified two challenges for which participants selected *Will Mitigate* but did not select any linked recommendation as *Already Used* or *Will Use* (see Section 3.2). This likely occurred because each of these challenges had only one linked recommendation. Expanding the list of recommendations for these challenges may improve coverage and usefulness.

We identified ten conflicting recommendations (Table 6) and nine conflicting challenges [13]. These represent cases where participants had differing opinions, which can be further explored in

the context of DevOps courses. Researchers should note that such conflicts do not necessarily need to be resolved universally—one side of the conflict may be more appropriate depending on the specific characteristics and context of a given course.

For example, the recommendation *Teaching social coding* received 5 *Won't Use* responses and 6 *Already Used* or *Will Use* responses. In this case, it is important to understand why some educators choose not to teach social coding, despite others finding it useful. Notably, only the recommendation *Terraform as a deployment provisioning tool can be used in teaching DevOps* includes comments from both the *Won't Use* and *Already Used plus Will Use* groups, providing insights into the reasons behind these differing perspectives.

Researchers could enhance the *Improve* tool by adding supplementary materials such as examples, videos, and links. This has the potential to transform the tool into a compendium of resources that can help educators apply the recommendations to specific challenges, as suggested by P5.

The number of responses regarding the use of recommendations and challenges presented in this study is substantial but not exhaustive. Researchers could conduct additional tool sessions and interviews with educators from other countries and explore the "DevOps Concepts" theme, which was not selected by any participants in this study, to gather further insights and enhance the findings.

5 Threats to Validity

We discuss threats to the validity of this study in the context of qualitative research [15, 17, 18].

Transferability. Transferability refers to the extent to which our findings can be applied to other contexts. This study is based on practical sessions and semi-structured interviews with 11 participants from Brazilian educational institutions. Most of the DevOps courses they teach (91%) are academic in nature.

Credibility. Credibility refers to whether the research findings are accurately derived from the original data. We employed several strategies to ensure credibility: (a) the construction and execution of queries were reviewed collaboratively by the researchers involved in the study; (b) all steps related to data extraction from interview transcriptions underwent double review, with many also subjected to blind review; and (c) the findings were discussed multiple times among the authors to mitigate potential researcher bias.

Confirmability. Confirmability refers to the extent to which the findings can be verified by other researchers. While we do not have participants' permission to share the full interview transcripts, we ensured transparency by identifying each piece of feedback related to challenges, recommendations, and teaching methods through direct quotations from participants whenever possible.

We also made the source code of the *Improve* application publicly available [10]. The repository includes the queries used to extract the quantitative results.

6 Related Work

In the area of recommendation systems for education, Almudena et al. [23] proposed a knowledge-based strategy to recommend educational resources, such as worked problems, exercises, quiz questions, and lecture notes, to students in the first two courses of a computer science major (CS1 and CS2). The knowledge base was constructed by computer science educators from various universities, and the system allows users to search using a query string. In contrast, our tool is specifically designed for DevOps education and enables users to search using themes related to challenges and recommendations. Additionally, our recommendation system is grounded in domain-specific knowledge collected from existing studies in DevOps education.

Ferino et al. [9] compared five DevOps-supporting environments: DevOps Lab Platform [2], ALECSS [20], Crunch [5], DevOpsEnvy [22], and CDEP [7]. These tools offer useful functionalities for both educators and students, such as automated assessment, integration with industry tools (e.g., Git, Jenkins), evaluation of team workload, and support for collaborative project work. In contrast, our tool provides educators with curated recommendations from expert DevOps instructors to support the improvement of their courses, including suggestions for teaching methods.

Zarour et al. [25] developed and applied a system-based learning tool to teach DevOps to students. In contrast, our tool is designed to help educators improve their DevOps courses. These educators were the focus of our study.

7 Conclusion

This work aimed to analyze the usefulness of challenges and recommendations in DevOps education, as identified in previous studies [8, 12], from the perspective of educators involved in DevOps courses. We conducted practical sessions, complemented by interviews, with 11 DevOps educators from various universities and institutes using the *Improve* tool.

The study found that 22.6% of the recommendations presented were new to educators and that they expressed interest in using them to improve their courses. Additionally, 64.6% of the challenges were considered relevant and had useful linked recommendations to address them. Participants also provided comments highlighting the value of the recommendations and the importance of addressing the challenges.

We identified conflicting responses related to ten recommendations and nine challenges, suggesting opportunities for further research into context-specific educational strategies and techniques. Our findings contribute to the improvement of educational guidelines for DevOps Education.

Based on the findings and limitations of this study, several avenues for future work can be explored. A natural next step is to replicate this study with educators from other countries. It can contribute to assess the transferability of our findings and identify challenges and recommendations that may be unique to different educational contexts.

ARTIFACT AVAILABILITY

All artifacts produced during this research, like data coding sheets and interview scripts, are available in [13].

ACKNOWLEDGMENTS

This work is partially supported by IFRN, INES.IA (www.ines.org.br), and CNPq grant 408817/2024-0.

REFERENCES

[1] Muhammad Azeem Akbar, Saima Rafi, Abeer Abdulaziz Alsanad, Syed Furqan Qadri, Ahmed Alsanad, and Abdulrahman Alothaim. 2022. toward successful DevOps: a decision-making framework. *IEEE Access* 10 (2022), 51343–51362.

[2] X. Bai, D. Pei, M. Li, and S. Li. 2018. The DevOps Lab Platform for Managing Diversified Projects in Educating Agile Software Engineering. In *2018 IEEE Frontiers in Education Conference (FIE)*. IEEE, 1–5.

[3] Laura Baldwin. 2022. Leveraging Katacoda technology exclusively within O'Reilly. https://www.oreilly.com/online-learning/leveraging-katacoda-technology.html

[4] HU Chih-Pei and Yan-Yi Chang. 2017. John W. Creswell, research design: Qualitative, quantitative, and mixed methods approaches. *Journal of Social and Administrative Sciences* 4, 2 (2017), 205–207.

[5] H. Christensen. 2016. Teaching DevOps and cloud computing using a cognitive apprenticeship and story-telling approach. In *Proceedings of the 2016 ACM conference on innovation and technology in computer science education*. 174–179.

[6] Christof Ebert, Gorka Gallardo, Josune Hernantes, and Nicolas Serrano. 2016. DevOps. *Ieee Software* 33, 3 (2016), 94–100.

[7] B. Eddy, N. Wilde, N. Cooper, B. Mishra, V. Gamboa, K. Shah, A. Deleon, and N. Shields. 2017. A pilot study on introducing continuous integration and delivery into undergraduate software engineering courses. In *2017 IEEE 30th Conference on Software Engineering Education and Training (CSEE&T)*. IEEE, 47–56.

[8] Samuel Ferino, Marcelo Fernandes, Elder Cirilo, Lucas Agnez, Bruno Batista, Uirá Kulesza, Eduardo Aranha, and Christoph Treude. 2023. Overcoming Challenges in DevOps Education through Teaching Methods. In *Proceedings of the 45th International Conference on Software Engineering*.

[9] Samuel Ferino, Marcelo Fernandes, Anny Fernandes, Uirá Kulesza, Eduardo Aranha, and Christoph Treude. 2021. Analyzing DevOps teaching strategies: An initial study. In *Proceedings of the XXXV Brazilian Symposium on Software Engineering*. 180–185.

[10] Marcelo Fernandes. 2025. *Improve Tool*. https://github.com/devops-education/improve

[11] Marcelo Fernandes et al. 2020. Challenges and recommendations in devops education: A systematic literature review. In *Proceedings of the 34th Brazilian Symposium on Software Engineering*. 648–657.

[12] Marcelo Fernandes et al. 2022. DevOps Education: An Interview Study of Challenges and Recommendations. In *Proceedings of the 44th International Conference on Software Engineering*.

[13] Marcelo Fernandes, Samuel Ferino, Uirá Kulesza, Eduardo Aranha, and Christoph Treude. 2023. *Educator Perceptions of DevOps Teaching Recommendations and Their Alignment with Common Challenges*. https://zenodo.org/doi/10.5281/zenodo.10059119

[14] The Linux Foundation. 2025. Production-Grade Container Orchestration. https://kubernetes.io

[15] Egon G Guba. 1981. Criteria for assessing the trustworthiness of naturalistic inquiries. *Ectj* 29, 2 (1981), 75–91.

[16] Hashicorp. 2025. Automate Infrastructure on Any Cloud. https://developer.hashicorp.com/terraform

[17] Irene Korstjens et al. 2018. Series: Practical guidance to qualitative research. Part 4: Trustworthiness and publishing. *European Journal of General Practice* 24, 1 (2018), 120–124.

[18] Enrique Larios Vargas et al. 2020. Selecting third-party libraries: The practitioners' perspective. In *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 245–256.

[19] Leonardo Leite, Carla Rocha, Fabio Kon, Dejan Milojicic, and Paulo Meirelles. 2019. A survey of DevOps concepts and challenges. *ACM Computing Surveys (CSUR)* 52, 6 (2019), 1–35.

[20] M. Ohtsuki and T. Kakeshita. 2019. Utilizing Software Engineering Education Support System ALECSS at an Actual Software Development Experiment: A Case Study. (2019), 367–375.

[21] RedHat. 2025. Ansible Collaborative. https://www.redhat.com/en/ansible-collaborative

[22] G. Rong, S. Gu, H. Zhang, and D. Shao. 2017. DevOpsEnvy: an education support system for DevOps. In *2017 IEEE 30th Conference on Software Engineering Education and Training (CSEE&T)*. IEEE, 37–46.

[23] Almudena Ruiz-Iniesta, Guillermo Jimenez-Diaz, and Mercedes Gomez-Albarran. 2014. A semantically enriched context-aware OER recommendation strategy and its application to a computer science OER repository. *IEEE Transactions on Education* 57, 4 (2014), 255–260.

[24] Miguel Ángel Sánchez-Cifo, Pablo Bermejo, and Elena Navarro. 2023. DevOps: Is there a gap between education and industry? *Journal of Software: Evolution and Process* 35, 12 (2023), e2534.

[25] Mohammad Zarour, Mohammed Akour, and Mamdouh Alenezi. 2024. Enhancing DevOps engineering education through system-based learning approach. *Open Education Studies* 6, 1 (2024), 20240012.