An Experience Report on Using Video-Creation Tasks in Requirements-Engineering Education

Citation for published version (APA):
May, R., Daher, J., Krüger, J., & Leich, T. (2024). An Experience Report on Using Video-Creation Tasks in Requirements-Engineering Education. In *ICSE-SEET '24: Proceedings of the 46th International Conference on Software Engineering: Software Engineering Education and Training* (pp. 201-211). Association for Computing Machinery, Inc. https://doi.org/10.1145/3639474.3640057

DOI:
10.1145/3639474.3640057

Document status and date:
Published: 24/05/2024

Document Version:
Accepted manuscript including changes made at the peer-review stage

An Experience Report on Using Video-Creation Tasks in Requirements-Engineering Education

Richard May
Harz University of Applied Sciences
Wernigerode, Germany
rmay@hs-harz.de

Johanna Daher
Harz University of Applied Sciences
Wernigerode, Germany
jdaher@hs-harz.de

Jacob Krüger
Eindhoven University of Technology
The Netherlands
j.kruger@tue.nl

Thomas Leich
Harz University of Applied Sciences
Wernigerode, Germany
tleich@hs-harz.de

ABSTRACT

Requirements engineering is a key skill in systems and software engineering. Educating students in the different forms and concepts of requirements engineering (e.g., traditional versus agile) is essential to prepare them for any technical job. However, requirements-engineering education can be challenging, particularly if it is not structured around a real-world project, and thus taught only conceptually. Unfortunately, when designing lectures, educators face pedagogical, technological, and content-related challenges, such as practice-orientation, student motivation, prior knowledge of students, or even emergency situations like the COVID-19 pandemic. In this paper, we report our experiences of integrating a novel pedagogical idea into a typical requirements-engineering course that builds on the increased use of multimedia communication in all parts of society: we asked students to create videos to document and communicate requirements of diverse products. Overall, we report *(i)* the general design of the course; *(ii)* why, how, and in what form we introduced video-creation tasks; as well as *(iii)* the students' feedback and our experiences. Due to mostly positive feedback and the rising demand for multimedia competences in industry, we perceive the introduction of the video-creation tasks as a success for developing key skills and improving students' motivation to learn about requirements engineering. We provide an overview of our lessons learned and discuss their implications to enable other educators to integrate similar tasks in their courses, while avoiding the pitfalls we faced.

CCS CONCEPTS

• **Applied computing → Interactive learning environments**; • **Software and its engineering → Requirements analysis**.

KEYWORDS

requirements engineering, education, multimedia competences, video-creation tasks

ACM Reference Format:
Richard May, Johanna Daher, Jacob Krüger, and Thomas Leich. 2024. An Experience Report on Using Video-Creation Tasks in Requirements-Engineering Education. In *46th International Conference on Software Engineering: : Software Engineering Education and Training (ICSE-SEET '24), April 14 –20, 2024, Lisbon, Portugal*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3639474.3640057

ICSE-SEET '24, April 14 –20, 2024, Lisbon, Portugal

ACM ISBN 979-8-4007-0498-7/24/04…$15.00
https://doi.org/10.1145/3639474.3640057

1 INTRODUCTION

Requirements represent a system's specification from the perspective of its stakeholders and are intended to guide developers in implementing that system [42]. Unfortunately, several problems in the context of systems and software engineering relate to requirements-engineering activities [8, 39], such as eliciting, documenting, or tracing requirements. A well-known cause for such problems is a lack of training leading to missing skill development or insufficient knowledge [30, 41]. Thus, it is key for systems and software engineers to understand the fundamentals of requirements engineering, recent practices, and methods that are relevant in their personal context (e.g., agile versus traditional development) to help them design and implement complex systems [41].

Requirements engineering is usually taught during software-engineering-related courses or curricula at university level or via professional training courses in industry. Educators face a variety of issues in teaching courses with such a close connection to practice as requirements engineering, for instance, designing a practice-oriented lecture, handling varying prior knowledge of students, or balancing between practice and theory [7, 11, 44]. Overall, the major challenge for any educator is conveying their content in an appropriate pedagogical and technological way [32].

Videos have emerged as a successful technology to audio-visually provide and explain information, for instance, as recorded lectures or online tutorials [36]. Additionally, videos have gained more and more popularity in companies to advertise, document, and ease the use of products, for example, by providing video-based training that explains a product's features and workflows [28, 49]. These videos essentially document and describe certain features, requirements, or use cases of a product, and are particularly helpful when describing highly innovative features that are hard to imagine [33]. In such cases, videos can also serve as a requirements specification for the respective systems and software engineers.

As an example, Perry [38] has taught short-film production classes in Malaysia, because organizations search for employees with creativity, digital, communication, and innovation skills. Particularly in digital and technological domains, using multimedia content has become increasingly relevant, with many social platforms like Instagram, TikTok, or YouTube building whole platforms around videos. As a consequence, skills for creating videos to communicate requirements to developers or users can be an immensely helpful skill for students to obtain [27, 38]. This, in turn, allows educators to prepare their students for the competitive job market and gain an advantage, with Perry finding that creating videos helped students to obtain, among others, relevant teamwork, communication, and time-management skills.

From the perspective of the established TPACK education framework [32], creating videos can be a useful pedagogical concept to design a well-founded course. The students' understanding of a topic can be significantly improved by using videos, ideally providing the right representation and adaptation of knowledge that support the students' strengths. Advantages of using videos have been demonstrated in a wide range of education areas, such as health [52], design [6], or mathematics [23]. However, we are not aware of a study that reports on using video-creation tasks (i.e., asking students to record, edit, document, and present videos) as an educational concept to teach students software-engineering-related concepts, such as requirements engineering.

In this paper, we report how we introduced video-creation tasks into a requirements-engineering course at a European university, roughly following an action-research-oriented methodology [5, 45]. We developed our concept on top of a traditional course that previously involved only basic practical training (e.g., using past projects from companies as exemplars). By incorporating video-creation tasks, we intended to educate the students on the corresponding video-creation skills, show them how to include novel practices into a workflow, increase their motivation, improve interactivity, and force them to dig deeper into requirements engineering to support their knowledge acquisition (i.e., asking them to think about the video requirements and requirements of the product that is shown). We argue that this concept is highly valuable, since we prepare students with novel and important skills, and increase the degree to which they are engaging in requirements engineering. However, setting up this course posed several challenges, such as providing the right infrastructure, teaching video-creation skills, or involving students from different nationalities.

To help educators who teach requirements engineering or want to use video-creation tasks, we contribute the following:

- We describe how we integrated video-creation tasks into a typical requirements-engineering course (Section 3).
- We report students' feedback and our own experiences of introducing the video-creation tasks (Section 4).
- We discuss the challenges we faced when changing the course and our lessons learned (Section 5).

We hope that these contributions can support educators to integrate video-creation tasks in a useful pedagogical way into their courses, avoiding the pitfalls and challenges we experienced. Moreover, the positive feedback we received indicates that the tasks have been well-received by many students, highlighting that creating videos does not only teach novel skills but can also improve students' motivation to learn and practice.

2 REQUIREMENTS EDUCATION

In the following, we first describe the fundamental background on requirements-engineering education before detailing related work on using videos for education. Then, we distinguish our paper from the related work and discuss its inherent limitations.

2.1 Background

Ideally, requirements engineering is taught in an appropriate technological and pedagogical way (i.e., professional and practice oriented), aiming to prepare students for practice and help them avoid pitfalls. Requirements-engineering skills are usually taught at universities as an essential foundation for future systems and software engineers. In such courses, students typically learn about the key concepts, theoretical foundations, and practical challenges of requirements engineering, which are required as an essential entry point into practice [10, 41]. However, there are several factors that pose challenges for educators and that impact the future success of any requirements-engineering course, namely:

- *internal influences related to the lecture design*, such as the lecture style [10] and a theoretical or practical focus [44];
- *internal influences related to the students*, such as their prior knowledge and experience [7, 11], cultural and ethnic backgrounds [21, 50], or the overall number of students in a lecture or the course [7, 11];
- *external influences related to the lecture design*, such as emergency situations (e.g., the COVID-19 pandemic) changing the teaching environment (i.e., online lectures) [3, 13]; and
- *external influences related to business practices*, such as considering the latest requirements-engineering trends and needs in industry [20] or recruitment demands for graduates [40].

Taking all those influences into account, designing and implementing a valuable and successful requirements-engineering course that adequately prepares students for practice can be highly challenging. As an ideal scenario, a requirements-engineering course would involve theoretical foundations, would connect these to practice-oriented concepts that are oriented towards the latest trends in research as well as industry, and would also use practical projects. However, such an ideal scenario is hard to impossible to achieve, particularly considering the above influences and constraints of a university curriculum.

2.2 Related Work

Various concepts have been proposed to teach requirements engineering, varying in the number of lectures, use of pedagogical frameworks, focus on theory or practice, or the general form of education. Recent state-of-the-art reviews on requirements-engineering education [10, 11] highlight three major trends in this area, specifically:

- *realistic stakeholder involvement* to train soft skills and practice-oriented situations, for instance, by conducting projects in collaboration with companies [15, 18];
- *gamification and (serious) game-related simulations* to create immersive effects for students (note that a *serious* game is a game

that is designed for educating students, and not just for playing), for example, using traditional [12] or e-learning games [51]; and
- *project-oriented learning* in student teams to explain or simulate common problems requirements engineers can face, for instance, via role playing [35, 47].

As we can see, many concepts for requirements-engineering education focus on simulating real-world situations to prepare students for practice. For educational concepts building on gamification and (serious) game-related simulations or on project-oriented learning, creating and watching videos or using video-related technologies (e.g., augmented reality or virtual reality) are frequently adopted ideas. Typically, the goal of using videos is to improve students' motivation for the topic and enable them to work independent of location as well as time, particularly with the emergency shift from face-to-face towards online teaching due to the COVID-19 pandemic [4, 11, 24]. As a consequence, we distinguish between two main areas of research that are related to our paper.

Video-Related Gamification and (Serious) Games. There have been different studies on, or applications related to, game-oriented learning using video technologies, including comparative studies on serious games (usually based on virtual reality) as well as applied gamification concepts. For instance, Mayor and López-Fernández [29] or Akbulut et al. [1] propose virtual-reality applications to teach software-engineering-related topics. Mayor and López-Fernándezz focus on teaching Scrum as an agile development method in a practical environment, while Akbulut et al. introduce an application for students to learn algorithms. In both cases, the authors conclude that virtual reality is highly effective for teaching software engineering. Neffati et al. [34] have created an augmented-reality platform as an extension to a software-engineering coursebook. Specifically, they implemented virtual graphics and a multimedia application to facilitate the learning process for students.

Regarding serious games, various applications have been proposed to support students in learning software-engineering-related topics. For example, Yasin et al. [51] have created a serious game for security requirements engineering and García et al. [17] have proposed a serious game for teaching a software engineering-related standard. Moreover, Gordillo et al. [19] compare the effectiveness of video-based learning and game-based learning in software engineering. To this end, they focus on using a game created by a teacher and highlight its immersive and motivational effects compared to watching an educational video only.

In contrast to our concept, none of the previous studies builds on a video-creation or similar (e.g., for augmented or virtual reality) task. The studies incorporate showing educational videos into teaching only, or compare novel concepts to watching such videos. Moreover, the focus has typically been on other software-engineering topics than requirements engineering. So, our contributions are orthogonal, focusing on a different topic for which we see additional benefits of educating students on creating videos, for instance, to communicate innovative features and their requirements more easily to developers and other stakeholders [33].

Watching or Creating Videos. Researchers have conducted studies and proposed applications focusing either on watching or creating videos for educational purposes. Spezialetti [43] investigated video scenarios to teach requirements gathering, system-design specification, and soft skills. Precisely, their students watched and analyzed videos to simulate a common requirements-engineering-related situation. Galster et al. [16] have introduced an active video-watching platform for software-engineering education. They highlight that video watching in combination with further educational material has the advantage of helping students develop and train soft skills, and that videos are independent of time and location.

Other studies have investigated videos in the context of documenting or understanding requirements or associated requirements-engineering processes. For instance, Fricker et al. [14] and Karras et al. [25] have used videos to record requirements workshops. Nagel and Karras [33] have proposed a methodology for creating and using videos to explain the vision of the system to stakeholders. Alami and Dalpiaz [2] have used gamified tutorial videos to teach requirements engineering. All of the underlying educational concepts aim to increase the communication and understanding of requirements. However, the videos are usually created by external persons, such as a film crew, instead of the students and/or the major focus is on watching and not on creating the videos. Finally, Burch [7] have asked their students to create project-based videos to present and visualize their software products. The main objective of these videos has been to receive feedback regarding the projects themselves, whereas we are concerned with communicating novel features and requirements.

In contrast to the previous works, our concept does not primarily focus on watching videos, but on letting students create them. We argue that especially creating videos is an important skill that will also become more and more important for requirements engineers. For example, while Burch has asked students to create videos, the scope in this case was different compared to ours. The students had to focus on visualizations for the project and product in general, while our concept aims to actively enable students to describe the requirements of a product via videos.

2.3 Novelty and Limitations

The related work mainly focuses on documenting and watching educational material related to software engineering on video. However, we believe that we can further advance on this situation by educating students on how to create meaningful videos for specific situations; which is a task in which they have to reflect upon the requirements of creating a video and of the product requirements or features they want to convey. This belief is driven by the more extensive use of videos in practice, supportive indications in the related work, and the experiences we obtained. As a consequence, we argue that our experiences can be a valuable addition to the current state-of-the-art, yielding novel and complementary insights on a teaching concept for one well-suited topic.

Still, we report experiences only. Thus, our insights have inherent limitations, for instance, that they can hardly be generalized and are to some extent subjective. Such limitations mean that further research is needed to confirm and refine our experiences. We hope that this paper can serve as a starting point for moving into that direction, inspiring other educators to adopt video-creation tasks and to contribute their experiences.

3 THE COURSE

After introducing the background and related work, we now describe the design of the course itself, its context, and how we integrated video-creation tasks into the course. Note that we did not follow a full-fledged action-research methodology [5, 45], due to teaching restrictions, regulations, and time constraints. Still, introducing the tasks essentially followed that methodology: Within this section, we describe the diagnosing of the situation (cf. Section 3.1 and Section 3.2), the action planning (cf. Section 3.3), and the action taking (i.e., the intervention, cf. Section 3.4). In Section 4, we describe the evaluation of our intervention before discussing our learning in Section 5. We display a general overview of the individual lectures, tutorials, and homework assignments in Figure 1.

3.1 Diagnosis: Study Program

Goals and Setting. The requirements-engineering course is a mandatory part of a master's program that is concerned with *technology and innovation management* at a more practice-oriented European university, namely the Harz University of Applied Sciences, Germany. In this study program, students deepen their competencies on a variety of areas, particularly (software) engineering, computer science, project management, business economics, and scientific working. The students shall learn to analyze, evaluate, and develop technical innovations, as well as how these innovations can be used in companies that move towards more digitization (e.g., introducing innovative software features into typically hardware-centred products). Consequently, in addition to the requirements-engineering course, the students attend various interdisciplinary lectures, such as strategic innovation management, information retrieval, cyber security and functional safety, or entrepreneurship. The university achieves a high level of practical relevance for the program through a *research and development project*, in which the students have to employ their developed skills (e.g., in requirements engineering) in practical research projects, often in collaboration with regional or national companies. Due to the close collaboration with practice, the focus on industrial requirements, and the unique skills obtained during the program, most of the students are directly hired by small to large international companies afterwards.

Students. The program is split into two tracks: a national track for native speakers and an English track for international students. Both tracks offer a range of lectures that are available in both languages (i.e., native and English), ensuring that every student can develop their skills appropriately. Due to the interdisciplinary nature of the program, it is available to students with a wide range of bachelor degrees in engineering (e.g., industrial engineering, mechanical engineering), computer science (e.g., business informatics, software engineering), and potentially other programs (e.g., economics). However, students from certain bachelor programs, such as economics, have to coordinate with the program head regarding what courses they previously attended and what additional courses they have to participate in before or during the master program to acquire missing knowledge (e.g., on software engineering).

Concept of the Course. Since the beginning of the program in 2016, both tracks involved a mandatory course focusing on (agile) requirements engineering (5 ECTS points). Note that although the course is mandatory, there is no specification of when within the study program the course must be taken. Only a certain number of ECTS points is prescribed, which must be provided before the start of the master thesis. However, the study guide recommends to take this course in the first or second semester. Due to national regulations, the course originally started as a typical lecture series with tutorials running in parallel. When the program gained more and more traction and interest from industry, more international students also asked to participate in the program, leading to the introduction of the international track in 2018, which does not differ from the native one except for the language. This posed novel challenges for the involved educators, for example, to account for cultural and language differences or missing prior knowledge, skills, and experiences of students. In the beginning, this situation caused various difficulties in the lectures and the tutorials, especially in the international track in which some students lacked a detailed understanding of certain technologies and their features. For instance, when explaining requirements-engineering activities on the examples of partially automated cars or modern driver-assistance systems, we noticed that background knowledge was lacking due to differences in the students' bachelor programs, different contents of their lectures, or certain technologies not yet being relevant or established in some countries. Furthermore, we experienced a general problem regarding the motivation to document complex systems with classical requirements-engineering techniques.

3.2 Diagnosis: Design of the Lectures

The lectures are organized along five primary topics of requirements-engineering basics and related concepts, which the primary lecturer introduces and discusses in twelve 90-minutes lessons (see Figure 1):

Motivation (lessons 1 & 2). In the first two lectures, we motivate the need for managing requirements in a digitized and software-based world. Precisely, we aim that students understand recent trends in research as well as practice (e.g., big data, machine learning), and become aware of the corresponding challenges (e.g., software complexity). So, the learning goal is for the students to understand the use of requirements in systems and software engineering, particularly for innovative products.

Requirements (lessons 3 & 4). In the next two lectures, we introduce the basic concepts and terms related to requirements engineering, such as goal, vision, use case, or functional and non-functional requirements. These basics are structured around the requirements-engineering framework by Pohl [39]. So, the learning goal is for the students to obtain the understanding required to describe and use requirements as well as related artifacts.

Goals, activities, framework (lessons 5, 6, & 7). Next, we introduce goals, scenarios, the detailed requirements-engineering framework as well as all of its activities, namely eliciting, documenting, negotiating, validating, and managing (e.g., change management, configurations, conflict resolution) requirements. The learning goal is to enable students to perform a goal-oriented requirements-engineering process as well as to understand the pros and cons of the involved activities and artifacts.

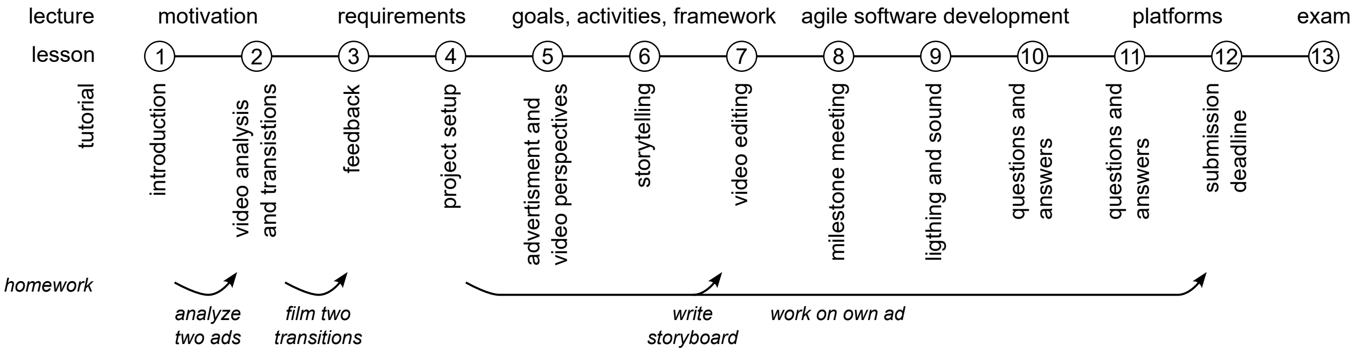

Figure 1: Overview of the course schedule, representing each lesson (i.e., week) in terms of lectures, tutorials, and homework.

Agile software development (lessons 8, 9, & 10). Since agile software development has become more and more prominent in practice (particularly for innovative projects), we incorporated three lectures on agile software development and how requirements are managed in this context [22, 31, 46]. Specifically, we introduce and exemplify how requirements are used within popular agile development methods (i.e., Scrum, Extreme Programming, feature-driven development, Crystal) and discuss the involved roles, artifacts, as well as events. Moreover, we analyze the pros and cons of agile methods from a practical point of view, aiming to achieve the learning goal of enabling students to decide whether an agile method is feasible for their project and to consequently employ it.

Platforms (lessons 11 & 12). Finally, in the last two lectures, we introduce the ideas of software platforms and configurability, which are particularly challenging topics for requirements engineering, but also key for the long-term success of (larger) companies [9, 26, 48]. In particular, we introduce the fundamental concepts and methods of software configurability based on software product-line engineering (e.g., features, software reuse, domain and application engineering), highlighting the various benefits and challenges that arise for companies—particularly in the context of requirements engineering. The learning goal is to enable the students to decide whether a software platform may be helpful to advance innovations further and to establish them in a market.

The students have to pass an oral exam (i.e., lesson 13) on the content of these lectures. A student's performance in the exam represents 50 % of their final grade to motivate them to learn about the introduced concepts. The tutorials contribute the remaining 50 % to also motivate the practical work.

3.3 Planning: Course Development

The switch towards online teaching, which was triggered by the COVID-19 pandemic in 2020, added further challenges regarding teaching and communication, mainly due to social distancing and the use of new tools for teaching [13, 37]. To tackle these specific problems, instead of only conducting and recording lectures in long videos (i.e., based on video conferences), the primary lecturer decided to create shorter videos as a pedagogical concept to explain certain topics via video tutorials. In these videos, the course educators explained specific topics of requirements engineering and aspects of certain technologies in a simple and clear manner. We took advantage of pedagogical and technological opportunities opened by online teaching, which relaxed regulations (e.g., on course structures and teaching formats), and thus allowed us to provide lectures in a more flexible style to enable students to attend the course independent of time or location [3].

When face-to-face teaching slowly started again in 2021, the primary lecturer kept the concept of creating short video sequences, due to overwhelmingly positive feedback by students. Interestingly, the students themselves further expanded the technology-driven concept by creating short self-made videos to present requirements and requirements-related artifacts, such as objectives or scenarios. This inspired the primary lecturer to extend the traditional format of the course to increase the motivation of students and to integrate them more. For this purpose, we reorganized the tutorials towards teaching requirements-engineering activities through a video-creation task.

Originally, the tutorials involved practical exercises on finished industry projects from regional companies, which we planned to change into a stand-alone tutorial series on video editing and documenting a product with its requirements as a video-creation task. Referring to the TPACK framework [32], we intended to combine teaching video-creation skills and their application (i.e., technological pedagogical knowledge) in the context of requirements engineering (i.e., content knowledge). More specifically, we aimed to relate the video creation to common requirements-engineering activities [39] as follows:

- *Elicitation*: The students have to think about the requirements of (1) the video and (2) the recorded product. This includes analyzing existing product ads to identify common elements and best practices, for their video and their product. Moreover, they have to discuss with stakeholders (i.e., students, lecturers) to understand their needs and expectations, involving the definition of user personas if needed.
- *Analysis*: Once the requirements are gathered, the students have to thoroughly analyze them to ensure clarity, consistency, and feasibility. This also entails defining a clear goal and product scenarios, classifying video requirements (i.e., functional requirements, quality attributes, constraints), and prioritizing the features they want to convey about the product they record.

- *Specification*: The students have to formally specify and document the analyzed requirements to create detailed guidelines for the video-creation process. For example, this includes a detailed style guide, comprehensive storyboards, and editing guidelines.
- *Validation*: To validate the requirements of the video and shown product, the students share their specifications with stakeholders (i.e., students, lecturers) for feedback and revisions. In this context, early video versions or mockups are useful to provide first impressions on the fulfillment of requirements.
- *Management*: Throughout the whole video-creation task, the students are encouraged to document their videos; and to store as well as maintain their requirements in a consistent way. Particularly, they have to continuously manage and trace adaptations to their requierements during feedback loops.

For a successful introduction of the video-creation tasks, the primary lecturer (i.e., the last author) decided to hire an online journalist (i.e., the second author) with in-depth knowledge on media and game conception (i.e., having a corresponding master degree) as well as experiences in the context of social media video-creation (i.e., own YouTube channel, YouTube partner program). Via this new design, we aimed to provide the students a detailed understanding of video-creation skills, which not only expands their own skillset but also forces them to integrate innovative concepts into established workflows (i.e., creating videos in a traditional course and for requirements engineering).

3.4 Action: Redesign of the Tutorials

Based on the planning, we decided to redesign the tutorials into twelve 90-minutes lessons that introduce different topics on creating videos, as well as homework assignments related to creating the student-made videos (i.e., the final outcome of the tutorials, cf. Figure 1). Due to time restrictions, all homework assignments are intentionally designed to support the final outcome, giving the students more space and time to work on their own videos and trying out different ways of working (e.g. applying agile project-management methods introduced in the lectures). While the students get inspired and create ideas during the course together with the lecturers, they have the freedom to film and try out video-related methods outside of the course setting, for example, they are free to try out other methods, products, and requirements. This way, they can also practice requirements elicitation and analysis in the context of planning their videos and describing different products. Through this freedom and tutorial-like structure, we aim to develop the skills for innovative video creation and editing that more and more companies are looking for in an increasingly digitized and video-centered world (e.g., in the context of social-media platforms, such as YouTube, Instagram, or TikTok).

Structure. The newly designed tutorials involve the following ten topics spanning eleven lessons (cf. Figure 1):

(1) We introduce the tutorial structure, its topics, the final outcome, as well as a first product-ad impression. As a first homework, we ask the students to analyze two video-based product ads of their choice. During this lesson, we aim to provide a high-level understanding of the components of videos, such as product presentations, stories, transitions, or camera perspectives as well as relevant requirements for creating videos.
(2) We start the second lesson by discussing the homework assignment, YouTube and its properties (e.g., from "how to" until video-metric insights), as well as video transitions. The students have to film two seamless transitions of their choice as a homework assignment.
(3) In the third lesson, we focus only on the homework, asking each student to present their transitions and discussing the outcomes with everyone to provide feedback.
(4) After learning the basics of creating videos, the students start to set up their final video-creation project, which includes finding a team, choosing a product and an appropriate editing software, as well as getting inspired for the first brainstorming sessions. In detail, the students build groups of three to five students within which they have to create an advertisement video as a team. We specify the technicalities of the videos to define the boundaries of the project (e.g., length of 2 to 3 minutes, format of 16:9). As a camera, the students use (their) smartphones, which usually have a sufficient image quality. The video is intended to promote a technological product chosen by the team (e.g., a smartphone, speakers), focusing on its features and functionalities. Each team can decide how they want to record and show the product, and how they want to work in their team—allowing them to freely find their own creative solution for telling a story and working collaboratively. A key requirement for every video is that it has to include the elements introduced during the lessons, such as seamless transitions, different camera perspectives, movements, and sounds (i.e., voice). Consequently, the project is running in parallel to the remaining lessons, enabling students to continuously ask questions, have an in-between presentation (lessons 7, 8), and get detailed feedback at the end if needed (lessons 10, 11).
(5) In this lesson, we introduce the concepts of advertisement and video perspectives.
(6) Next, we introduce the basics of storytelling in videos. Subsequently, the students have to start working on storyboards for their videos, which is a concrete homework assignment.
(7) We start this lesson by discussing the storyboards of the teams, providing feedback on their progress and stories. Also, we teach the fundamentals of video editing and corresponding software tools, such as Adobe Premiere.
(8) After the previous four lessons, we conduct milestone presentations in which each team presents the current status of their video and its editing.
(9) As two of the last concepts on video creation, we introduce lighting and sound to help the students improve their videos further.
(10) Finally, we provide the opportunity for each team to voluntarily attend two more lessons (10, 11) to receive individual feedback on the videos from the lecturer.

To ensure that we have enough time to view and properly grade each video (as mentioned, accounting for 50 % of the course grade), the students have to submit these before the last lesson (i.e., 12). During that lesson, we conduct a get together in which all teams watch their videos, receive feedback from the lecturers as well as other teams, and anonymously evaluate the course based on a template of the university.

Implementation. We have implemented the video-creation tasks for three iterations of the course. The newly designed course took first place in 2021/22 (winter term) with 64 international students that were split into two groups. We used two groups because we found that the students' levels of experience with creating and editing videos differed heavily from absolute beginners to experienced. More experienced students were actively involved in social media (i.e., they interacted on such platforms every day and released own videos), and thus they had a closer connection to creating videos. As an immediate consequence, this group of students was highly motivated in the tutorial and required less support.

In the second iteration in 2022 (summer term), we conducted the course with 13 national students, also a mixture of beginners and more experienced users. For instance, one student filmed videos with a Go-Pro, which their team used during the tutorials. We experienced one expected problem that did not manifest at all in the previous, international iteration: One student was uninterested and unmotivated in creating a video. It was challenging for us to convey the meaning of the task and how the obtained skills could benefit their future career. After many conversations, the student created a video with their team, and their perspective as well as attitude changed during the course towards a more positive side.

Lastly, the third iteration took place in the winter term 2022/23 with 37 international students. Overall, we experienced similar patterns as in the first iteration regarding experiences and motivation. Still, while we kept the course structure due to the mostly positive feedback, we built on the experiences of the second iteration to update our material to fix ambiguities and provide a better reasoning to the students why video-creation skills are valuable.

4 EVALUATION

Next, we summarize the feedback we received (e.g., the summarized anonymous evaluations provided to us by the university, personal comments) and the outcomes of the course (e.g., videos). Please note that we are neither allowed (legally and by university regulations) to disclose the complete evaluation forms nor to map the feedback to exam grades, which is why we focus on students' comments.

4.1 Experiences and Course Evaluations

Every group in the three iterations created an own video advertisement using their smartphones and the two Gimbals the university provided; presenting, for instance, speakers, innovative glasses, or the connection between Apple devices. Matching our experiences that the international students were more motivated than the national ones, the feedback also differed between the three iterations. In total, 24 of the 101 international students answered the anonymous evaluation questionnaire of the university, including statements like:

- "Really liked the practical approach of this seminar. Rather than being theoretical."
- "The video making. It was so fun and also learned a new tool. In the end, enjoyed watching all the videos together."
- "Well firstly a very realistic approach to learn doing videography, properly organized step by step interpretation. The final task of making a product ad was the best experience in my view and must be retained."
- "Teaching method and explanation was excellent and should be done the same way."

Such comments represent the majority and indicate that the redesign of the tutorials was successful and well-perceived by the students. This also matches our personal perceptions during the course iterations.

Still, we also experienced challenges, for instance, COVID-19 requiring some internationals to stay in their home countries and some Zoom lessons during the first iteration. This is why some students criticized the internet, because the university's Wifi was not stable at all times. Moreover, some comments clearly highlighted potential for improving the course infrastructure as well as valuable design improvements:

- "Sadly this semester, some of the equipment arrived really late in the semester, e.g. the Gimbal etc. These type of equipment could be organized a bit more properly so that students can use them more efficiently."
- "To sum it up adding more practicality to the sessions will just make it perfect in my perspective."
- "Instead of ads we can think more in digital transformation."

Note for the second comment that time and COVID-19 restrictions in the transitioning to the video-creation task resulted in the assignments being actual homework to let every student practice at their own speed. Setting up a lab with a proper technical infrastructure and editing software at the university in the future will improve this setup and allow students to easily join collaborative sessions for their homework assignments and projects. Finally, a few international students in the third iteration suggested that it may be even better to focus on creating videos for explaining digital technologies instead of creating a product advertisement. We consider this an interesting alternative that we aim to offer in the future. Still, the negative feedback in the international groups has been primarily on technicalities, not on the course structure, content, effort, or task—all of which were perceived positive.

Compared to the international groups, the national group involved more students that were less interested in the course, did not like the goals of the task, or constantly questioned why they should produce an advertisement video. Seven of 13 students answered the anonymous evaluation, with roughly half of them writing negative comments, such as:

- "Don't see the connection with the main course."
- "Very time-consuming."

In contrast, the remaining half of the students wrote all positive comments, such as:

- "Very interesting concept."
- "The taught contents should remain."

So, the overall picture in the national group varies more, and we particularly used the critique to strengthen the motivation for creating the advertisements in the third iteration.

We also got the impression that some students just did not want to spend effort on the task, which is further supported when taking into account the feedback from previous terms (using company exemplars instead of the video-creation task): National students preferred agile methods and employed them for typical homework assignments (i.e., exercises on finished reports), but seemed to dislike using them for creating videos. For instance, before introducing the video-creation task into the course, national students in the summer term of 2021 wrote comments, such as:

- "Very great practical references with vivid examples. The exercises were very helpful and supported the learned theory."
- "The taught content was great! Especially the opportunity, watching the whole lecture (or parts of it) in a recorded video for the final exam, should maintain."
- "The lecture and exercise have been very exciting and informative. The lecture was mediated in a good way. The exercises with their practical examples with real cases helped a lot."

Since such feedback indicates that the (identical) content of the course is interesting, the more negative feedback by the national students in the summer term of 2022 may have been caused by less willingness to spend additional time on the video-creation task. For us, this implies that it is important to reflect on using a combination of typical assignments and a shorter video-creation task.

Experiences and Course Evaluation

To summarize our experiences and the feedback we received:
- *It can be challenging to communicate the benefits of video-creation tasks to reason on the additional effort to some students, which can impair their motivation.*
- *It is key to provide the necessary technical infrastructure to the students early on.*
- *The video-creation task was perceived very positive by most students, viewed as a means to transfer important skills that are becoming more and more relevant in practice.*

4.2 Further Course Development

Although some national students have criticized the new concept, we intend to keep it due to the primarily positive feedback from the others and almost all international students. In fact, the videos and grades of the international students have been very good on average, showing that all students were able and had the required infrastructure as well as knowledge to create videos for describing a product and its features. However, we are improving our materials based on the students' feedback, particularly aiming to provide a better motivation for the task and making it more practical. In addition, there will be an increased focus on creating videos not only to describe products, but also to show and perform actual requirements-engineering activities, which has essentially been a secondary goal of the video-creation tasks so far. In this context, a shift in focus to the creation of videos primarily for requirements-engineering activities, such as eliciting and documenting requirements, is currently planned. Another primary point we aim to improve in the future is to introduce a studio for the course with all required infrastructure (cf. Figure 2). While the university owns a professional film studio for courses related to the program media informatics, this studio is overpowered and involves many tools that can easily overwhelm students without assistance and the proper background knowledge (e.g., on cameras, lighting, screens). We have not used this studio, since we aimed to provide an easily accessible introduction into creating videos.

To provide a better technical infrastructure and collaborative environment, we plan to build a small studio with the essential technical infrastructure and green screens that all students of our course can use. We are convinced that this will contribute to more fairness for the students (e.g., same technical starting point), ensure that they have an appropriate setup (e.g., providing all tools and software required), improve their motivation (e.g., facilitating collaborations at the location), and help us tackle further issues, for instance, whether we should use a professional (expensive) editing tool or a free alternative. As we sketch in Figure 2, the plan is to have a small setup with only a few workstations and a green-screen wall to not overwhelm students as with the university's film studio.

Figure 2: Conceptual sketch of the media studio.

Future Improvements

We plan to intensify the focus on using the video-creating tasks for describing actual requirements-engineering activities, and not only for describing products as part of executing certain activities. Moreover, we will set up a video laboratory, with our experiences indicating that a well-equipped and collaborative environment will greatly help students compared to working from home.

5 LESSONS LEARNED

Next, we present our lessons learned, providing insights for supporting technological and pedagogical knowledge [32]. We hope that these help educators adopt video-creation concepts in their courses if feasible.

5.1 Building and Organizing Student Teams

During the tutorials, all students learned the basics of creating videos. Interestingly, we experienced that previous experiences or prior knowledge do not play a major role regarding the video creation, because the students did work in mixed teams with different levels of experiences (i.e., highly experienced students worked together with inexperienced students). So, the students learned potentially missing skills together and from each other (e.g., regarding video editing or cutting). Nevertheless, a team's skill level in creating videos is visible in the final outcome, leading to a slightly varying professionalism of the resulting video. Although the course and its conveyed content is restricted due to the limited number of lessons, the video-creation tasks were always successfully and satisfactory completed by every team.

The overall number of students in the course was varying, depending on the number of new students in the study program.

Considering both tracks (i.e., national and international), we experienced that a team size of three to five students was suitable in either case. With that team size, the students had neither too many nor too few tasks, and exactly knew what their exact roles in their team were. Not surprisingly, in general, teaching the course with fewer students makes it easier to work with them and discuss the tasks on a deeper level. Consequently, the more groups there are, the more time passes to discuss everyone's questions and problems, which can become critical in the context of the limited number of lessons. So, we learned that while video-creation tasks are an interesting and well-perceived concept for team settings, they do not necessarily solve the problem of effort required for setting up practical projects.

Lessons Learned: Building and Organizing Student Teams

- *Every student can learn the basics of video-creation, regardless of previous experiences or prior knowledge.*
- *Mixed teams of beginners and experts are ideal to create learning effects between the students.*
- *Groups of three to five students per teams worked best.*
- *A smaller number of teams allows for more in-depths feedback and limits the effort.*

In summary, we recommend to work with mixed teams regarding the expertise of creating videos, involving three to five students in each group. Also, educators should not underestimate the effort video-creation tasks require and aim to work with fewer teams.

5.2 Increasing Student Motivation

We perceived a tendency that whether students like or dislike the video-creation task depends on their personal vision and social-media usage. In this context, one of the national students criticized the video-creation task, because it did not relate to their personal career perspective, and thus did not create any value in their perception. However, most students were highly motivated to create a video, especially because they were creating and watching videos daily (e.g., on YouTube, Instagram, TikTok); thereby also motivating other students. So, highly motivated students are an ideal part of any team to bring their experience and enthusiasm to lift the motivation of the whole group. However, it is also the responsibility of the lecturers to clearly motivate why the task is relevant for their course (i.e., requirements engineering) and how the obtained skills benefit the students. We aimed to improve this motivation in the third iteration by discussing the increased use of media (and particularly videos) in all forms of communication, and as a means to explain innovative features as well as requirements.

Moreover, we experienced that letting teams choose their own product for their videos was well appreciated by the students, mainly due to varying interests among the teams. To support the students' creativity, the lecturer showed inspirational video advertisements during the tutorial, aiming to help the students find their own product ideas. This way, we noticed that students became more creative and created a wide variety of different approaches, video styles, and contents. In contrast, we had to teach the fundamentals of video creation during the tutorials to ensure that all students had the skills to fulfill the task. Still, freedom in the creative process of selecting a product with innovative requirements helped to improve motivation and to successfully transfer our learning goals.

Although watching videos can tend to make students inactive, unfocused, and bored because of their passive role [33], we have not noticed a similar effect with our students. While watching the self-made videos of the teams together, the students were motivated, applauded, and gave feedback in a constructive way. The students excitedly watched all videos and supported the other teams. Arguably, it is much more motivating to students to actually create their videos and see how the others performed, compared to simply watching recorded lectures. We consider this a great benefit of using video-creation tasks.

Lessons Learned: Increasing Student Motivation

- *Giving students freedom to explore creative video-creation tasks and products is essential to improve their motivation and yield positive learning effects.*
- *Watching intermediate results and final videos of all teams together helps to further motivate students and improve the videos based on feedback.*
- *Creating own videos seems more motivational to students than simply watching them.*

In summary, we recommend to have at least one highly motivated student in each group and allow for freedom in the concrete solutions to the video-creation tasks. Also, creating videos seems much more engaging to students than simply watching them, which will also result in more detailed and constructive feedback during recommended milestone meetings. Based on these experiences, we perceive video-creation tasks as valuable and interesting for (requirements-engineering) education.

5.3 Project Planing and Management

Despite some negative feedback, we experienced that both parts of the course, namely learning theory in the lectures (e.g., agile methods) and using video-creation tasks in the tutorials, strongly support each other. Especially agile methods were usually adopted by the students for producing their videos in a more efficient way. The students organized their project and teams in agile-like ways, for instance, by applying Kanban, and found creative ways to plan their project as well as the requirements they wanted to present by using tools like Trello or Miro. As a means to connect lectures and tutorials even more (as demanded by some national students), we are exploring ways to more clearly communicate the video-creation task as a practical exemplar. For this purpose, we envision that students describe their requirements-engineering activities, such as eliciting, documenting, and validating the video requirements in more detail than before.

Additionally, we experienced that defining a well-planned and student-oriented time schedule of the tutorials that is openly communicated to the students is key to the success of a course. The time schedule supports the students in planing their video-creation task, knowing all relevant steps, milestones, and submission deadlines. Specifically, every homework assignment helped them with their own project (e.g., creating a storyboard, identifying requirements), while milestone meetings helped them to get feedback from the lecturer and all other teams. However, although we organized milestone meetings to support the project progress, some teams had time issues due to time-consuming video editing. Depending on their previous experiences or background, some students may need more time or a more structured plan than others. This is why it is essential that the lecturer monitors the progress of all teams to help them more quickly and to provide enough time to work. So, we also argue that it is important to avoid additional tasks that do not contribute to achieving the actual main video-creation task.

Lessons Learned: Project Planing and Management

- *Combining requirements-engineering theory in a lecture with creating a feasible video contributes to students engaging with the theoretical concepts.*
- *Video-creation tasks seem to promote agile working methods, which is an ideal opportunity for students to transfer their theoretical knowledge into practice.*
- *Providing a student-oriented time schedule contributes to the success of the course.*
- *By monitoring the students' progress, educators can detect problems early on and provide enough time to resolve these.*

In summary, we strongly advise educators to monitor the progress of students regarding their video-creation tasks, since problems can have significant impact and may be the result of missing experience or knowledge. So, regular milestones and a clear schedule against which the students can compare their work are important. Interestingly, video-creation tasks seem to be a great opportunity to transfer some theoretical concepts into practice (e.g., agile methods), which again underpins the value this concept can have.

5.4 Choosing Video Equipment

Based on our positive experiences regarding the use of smartphones, we highlight that professional camera equipment is not needed to successfully create useful videos with a great learning effect for students. Precisely, modern smartphones already offer high-quality cameras and sufficient technological opportunities. In this context, nearly every student of the course owned a smartphone, knew how to use its camera functions, and was motivated enough to discover more camera functions. However, to obtain an even higher quality of the videos, ensure fair starting points, and facilitate the conduct of the tasks, the course should ensure that enough high-quality hardware and software is available to the teams.

In fact, even though we bought additional equipment for students that they could borrow (e.g., two Gimbals), the students0 missed a work environment with sufficient work places for editing their videos in a higher quality and with a better performance (i.e., reducing the time required). This problem also led to the situation that the more experienced students already had license-based editing software (e.g., Adobe Premiere), while others had to use free software (e.g., iMovie) on their personal computers. As a consequence, we decided and (due to the positive feedback) received funding from the university to build an independent lab (cf. Figure 2) that provides this technical infrastructure and a dedicated meeting place for the students—enabling future students to produce their videos more easily together and improve the quality of these videos further. Still, we are convinced that video-creation tasks can be implemented with a small technical setup (e.g., providing a few cameras/smartphones and workstations with free software), and thus are feasible for most lectures and universities.

Lessons Learned: Choosing Video Equipment

- *Smartphones are well-suited to create student videos.*
- *To enable higher quality videos, professional technical equipment (i.e., hardware, software) and work places are helpful.*
- *A dedicated place for working on the videos helps the teams collaborate and coordinate.*

In summary, we recommend to (if possible) provide a dedicated space for students to create (film, edit, etc.) their videos, in which also workstations and the required software are available. Based on the feedback we received and our experiences, we argue that this would greatly benefit the students' collaborations, and also improve the fairness for all of them by defining the same technical setup. However, if such equipment is not available, there are free software solutions and smartphones are an ideal starting point to initiate a video-creation task to explore its feasibility for a course. Later on (as we did), educators can then argue for funding based on the students' feedback and the success of the course.

6 CONCLUSION

In this paper, we shared our experiences of introducing a concept for requirements-engineering education based on video-creation tasks and described its implementation in three iterations of a university course. We highlighted that although this concept is applied in a similar way in some courses and is a major trend in practice due to social media, it is still somewhat underdeveloped in education—even though it promises several benefits. Based on our students' feedback, we perceive video-creation tasks as a valuable pedagogical technology-based concept for (requirements-engineering) education, and recommend other educators who want to introduce this concept to (among others):

- Establish teams with mixed experiences in creating videos, typically involving three to five students (cf. Section 5.1).
- Allow the students freedom regarding the concrete solutions for creating the videos (cf. Section 5.2).
- Define and publish a clear schedule for monitoring how students are proceeding towards the goal, letting them employ their own methods while using milestones and intermediate presentations to evaluate their progress (cf. Section 5.3).
- Provide a dedicated work environment with the required hardware and software infrastructure (cf. Section 5.4).

In general, we experienced that the video-creation task facilitated the transfer of theoretical requirements-engineering knowledge into practice and was motivating most students. The negative feedback provided further insights on how to improve the course design, and to communicate clearer how the video-creation task is connected to requirements engineering.

In the future, we aim to further improve the course and incorporate the students' feedback. Particularly, we are interested to see the impact of the media studio as a place for collaborative working. We hope that our contributions help researchers and educators experiment with video-creation tasks in requirements- and software-engineering education themselves; which we also aim to extend more systematically.

REFERENCES

[1] A. Akbulut, C. Catal, and B. Yıldız. 2018. On the Effectiveness of Virtual Reality in the Education of Software Engineering. *Computer Applications in Engineering Education* 26, 4 (2018), 918–927.

[2] D. Alami and F. Dalpiaz. 2017. A Gamified Tutorial for Learning about Security Requirements Engineering. In *International Requirements Engineering Conference (RE)*. IEEE, 418–423.

[3] M. Barr, S. W. Nabir, and D. Somerville. 2020. Online Delivery of Intensive Software Engineering Education During the COVID-19 Pandemic. In *Conference on Software Engineering Education and Training (CSEE&T)*. IEEE, 1–6.

[4] E. Barut Tugtekin and O. O. Dursun. 2022. Effect of Animated and Interactive Video Variations on Learners' Motivation in Distance Education. *Education and Information Technologies* 27, 3 (2022), 3247–3276.

[5] R. L. Baskerville and A. T. Wood-Harper. 1996. A Critical Perspective on Action Research as a Method for Information Systems Research. *Journal of Information Technology* 11, 3 (1996), 235–246.

[6] D. Brothers and A. Wendell. 2016. Design Films-Implementing Video Creation Techniques into Undergraduate Design Education. In *International Conference of The Arab Society for Computation in Architecture, Art, and Design (ASCAAD)*. CUMINCAD, 321–330.

[7] M. Burch. 2020. The Importance of Requirements Engineering for Teaching Large Visualization Courses. In *International Workshop on Learning from other Disciplines for Requirements Engineering (D4RE)*. IEEE, 6–10.

[8] M. Chemuturi. 2012. *Requirements Engineering and Management for Software Development Projects*. Springer.

[9] P. Clements and L. Northrop. 2002. *Software Product Lines*. Addison-Wesley.

[10] M. Daun, A. Grubb, V. Stenkova, and B. Tenbergen. 2023. A Systematic Literature Review of Requirements Engineering Education. *Requirements Engineering* (2023), 1–31.

[11] M. Daun, A. M. Grubb, and B. Tenbergen. 2021. A Survey of Instructional Approaches in the Requirements Engineering Education Literature. In *International Requirements Engineering Conference (RE)*. IEEE, 257–268.

[12] A. de Pádua Albuquerque Oliveira, V. R. Werneck, J. C. S. do Prado Leite, and L. M. Cysneiros. 2015. The Monopoly Game to Teach ERi*c – Intentional Requirements Engineering. In *International Conference on Advanced Information Systems Engineering (CAiSE)*. 49–54.

[13] R. Fitoussi and H. Chassidim. 2021. Teaching Software Engineering During Covid-19 Constraint or Opportunity?. In *Global Engineering Education Conference (EDUCON)*. IEEE, 1727–1731.

[14] S. A. Fricker, K. Schneider, F. Fotrousi, and C. Thuemmler. 2016. Workshop Videos for Requirements Communication. *Requirements Engineering* 21, 4 (2016), 521–552.

[15] G. Gabrysiak, R. Hebig, L. Pirl, and H. Giese. 2013. Cooperating with a Non-Governmental Organization to Teach Gathering and Implementation of Requirements. In *Conference on Software Engineering Education and Training (CSEE&T)*. IEEE, 11–20.

[16] M. Galster, A. Mitrovic, and M. Gordon. 2018. Toward Enhancing the Training of Software Engineering Students and Professionals Using Active Video Watching. In *International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*. ACM, 5–8.

[17] I. García, C. Pacheco, A. León, and J. A. Calvo-Manzano. 2020. A Serious Game for Teaching the Fundamentals of ISO/IEC/IEEE 29148 Systems and Software Engineering – Lifecycle Processes – Requirements Engineering at Undergraduate Level. *Computer Standards & Interfaces* 67 (2020), 103377.

[18] K. A. Gary. 2009. Contextual Requirements Experiences within the Software Enterprise. In *Requirements Engineering Education and Training (REET)*. IEEE, 12–19.

[19] A. Gordillo, D. López-Fernández, and E. Tovar. 2022. Comparing the Effectiveness of Video-Based Learning and Game-Based Learning Using Teacher-Authored Video Games for Online Software Engineering Education. *IEEE Transactions on Education* (2022).

[20] K. Hertz and P. Spoletini. 2018. Are Requirements Engineering Courses Covering What Industry Needs? A Preliminary Analysis of the United States Situation. In *Requirements Engineering Education and Training (REET)*. IEEE, 20–23.

[21] D. Hidellaarachchi, J. Grundy, R. Hoda, and I. Mueller. 2022. The Influence of Human Aspects on Requirements Engineering-related Activities: Software Practitioners' Perspective. *ACM Transactions on Software Engineering and Methodology* 1, 1 (2022).

[22] J. Highsmith and A. Cockburn. 2001. Agile Software Development: The Business of Innovation. *Computer* 34, 9 (2001), 120–127.

[23] M. C.-L. Huang, C.-Y. Chou, Y.-T. Wu, J.-L. Shih, C. Y. C. Yeh, A. C. C. Lao, H. Fong, Y.-F. Lin, and T.-W. Chan. 2020. Interest-Driven Video Creation for Learning Mathematics. *Journal of Computers in Education* 7, 3 (2020), 395–433.

[24] J. Jang, Y. Ko, W. S. Shin, and I. Han. 2021. Augmented Reality and Virtual Reality for Learning: An Examination Using an Extended Technology Acceptance Model. *IEEE Access* 9 (2021), 6798–6809.

[25] O. Karras, S. Kiesling, and K. Schneider. 2016. Supporting Requirements Elicitation by Tool-Supported Video Analysis. In *International Requirements Engineering Conference (RE)*. IEEE, 146–155.

[26] J. Krüger and T. Berger. 2020. An Empirical Analysis of the Costs of Clone- and Platform-Oriented Software Reuse. In *Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE)*. 432–444.

[27] G.-F. Liu, P.-C. Gao, Y.-C. Li, and Z.-P. Zhang. 2019. Research on the Influence of Social Media Short Video Marketing on Consumer Brand Attitude. In *International Conference on Social science, Humanities and Education (ICSSHE)*. Atlantis Press, 433–438.

[28] J. Liu, J. Wu, H. Gan, T. Zhang, and C. Ma. 2023. Diversified Teaching Reform of Software Requirement Engineering Based on Case. In *International Conference on Computational Structures Technology (CST)*. IEEE, 30–34.

[29] J. Mayor and D. López-Fernández. 2021. Scrum VR: Virtual Reality Serious Video Game to Learn Scrum. *Applied Sciences* 11, 19 (2021), 9015.

[30] R. N. Memon, R. Ahmad, and S. S. Salim. 2010. Problems in Requirements Engineering Education: A Survey. In *International Conference on Frontiers of Intelligent Technology (ICFIT)*. ACM, 1–6.

[31] B. Meyer. 2014. *Agile!* Springer.

[32] P. Mishra and M. J. Koehler. 2006. Technological Pedagogical Content Knowledge: A Framework for Teacher Knowledge. *Teachers College Record* 108, 6 (2006), 1017–1054.

[33] L. Nagel and O. Karras. 2021. Keep Your Stakeholders Engaged: Interactive Vision Videos in Requirements Engineering. In *International Requirements Engineering Conference Workshops (REW)*. 51–57.

[34] O. S. Neffati, R. Setiawan, P. Jayanthi, S. Vanithamani, D. K. Sharma, R. Regin, D. Mani, and S. Sengan. 2021. An Educational Tool for Enhanced Mobile e-Learning for Technical Higher Education Using Mobile Devices for Augmented Reality. *Microprocessors and Microsystems* 83 (2021), 104030.

[35] T. Nkamaura and Y. Tachikawa. 2016. Requirements Engineering Education Using Role-Play Training. In *International Conference on. Teaching, Assessment and Learning for Engineering (TALE)*. IEEE, 231–238.

[36] M. Noetel, S. Griffith, O. Delaney, T. Sanders, P. Parker, B. del Pozo Cruz, and C. Lonsdale. 2021. Video Improves Learning in Higher Education: A Systematic Review. *Review of Educational Research* 91, 2 (2021), 204–236.

[37] S. Ouhbi and N. Pombo. 2022. Teaching Software Engineering at the Beginning of the COVID-19 Pandemic: Experience Paper. In *World Engineering Education Conference (EDUNINE)*. IEEE, 1–6.

[38] M. S. Perry. 2018. 21st Century Skills through Film Production in Tertiary Education: A Transformative Assessment in a Literature and Media Course. *Southeast Asian Journal of English Language Studies* 24, 4 (2018).

[39] K. Pohl. 2010. *Requirements Engineering: Fundamentals, Principles, and Techniques*. Springer.

[40] A. Pradela. 2012. Engineering Education in the Context of Labor Market Requirements and Expectations-Polish Experiences. *Global Journal of Engineering Education* 14, 2 (2012).

[41] G. Regev, D. C. Gause, and A. Wegmann. 2008. Requirements Engineering Education in the 21st Century, an Experiential Learning Approach. In *International Requirements Engineering Conference (RE)*. IEEE, 85–94.

[42] K. Schmid. 2014. Challenges and Solutions in Global Requirements Engineering – A Literature Survey. In *Software Quality Days (SWQD)*. Springer, 85–99.

[43] M. Spezialetti. 2021. Putting People in the Picture: Building Requirements Gathering, Design Specification and Communication Skills with Video Scenarios. In *Technical Symposium on Computer Science Education (SIGCSE)*. ACM, 129–135.

[44] M. Spichkova. 2019. Industry-Oriented Project-Based Learning of Software Engineering. In *International Conference on Engineering of Complex Computer Systems (ICECCS)*. IEEE, 51–60.

[45] M. Staron. 2020. *Action Research in Software Engineering*. Springer.

[46] T. Stober and U. Hansmann. 2010. *Agile Software Development*. Springer.

[47] R. B. Svensson and B. Regnell. 2017. Is Role Playing in Requirements Engineering Education Increasing Learning Outcome? *Requirements Engineering* 22, 4 (2017), 475–489.

[48] F. J. van der Linden, K. Schmid, and E. Rommes. 2007. *Software Product Lines in Action: The Best Industrial Practice in Product Line Engineering*. Springer.

[49] H. van der Meij and J. van der Meij. 2013. Eight Guidelines for the Design of Instructional Videos for Software Training. *Technical Communication* 60, 3 (2013).

[50] B. Waizmann, H. Schuhbauer, and P. Brockmann. 2020. Smart Technology to Improve Cultural and Gender Diversity in Engineering Education. In *Global Engineering Education Conference (EDUCON)*. IEEE, 61–65.

[51] A. Yasin, L. Liu, T. Li, J. Wang, and D. Zowghi. 2018. Design and Preliminary Evaluation of a Cyber Security Requirements Education Game (SREG). *Information and Software Technology* 95 (2018), 179–200.

[52] C. Zahn, N. Schaeffeler, K. E. Giel, D. Wessel, A. Thiel, S. Zipfel, and F. W. Hesse. 2014. Video Clips for YouTube: Collaborative Video Creation as an Educational Concept for Knowledge Acquisition and Attitude Change Related to Obesity Stigmatization. *Education and Information Technologies* 19, 3 (2014), 603–621.