The Journal of Systems and Software 210 (2024) 111946

Contents lists available at ScienceDirect

The Journal of Systems & Software

journal homepage: www.elsevier.com/locate/jss

Thriving in the era of hybrid work: Raising cybersecurity awareness using serious games in industry trainings☆

Tiange Zhao [a,*], Tiago Gasiba [a], Ulrike Lechner [b], Maria Pinto-Albuquerque [c]

[a] Siemens AG, Otto-Hahn-Ring 6, 81739, München, Germany
[b] University of the Bundeswehr Munich, Werner-Heisenberg-Weg 39, 85579, Neubiberg, Germany
[c] Instituto Universitário de Lisboa (ISCTE-IUL), ISTAR, Av. das Forças Armadas, 1649–026 Lisboa, Portugal

ARTICLE INFO

Keywords:
Serious game
Secure coding
Cloud security
Industry
Hybrid work
Awareness of cybersecurity

ABSTRACT

The important missions of modern software engineering education are to prepare software engineers to work in a hybrid mode and to address the need to enable them to write secure code and deliver secure products and services to the customer. Providing training akin to an authentic experience poses several challenges, such as hybrid infrastructures, lack of engagement, and interactions. Cybersecurity and cybersecurity awareness have also gained importance due to the shift towards work-from-home (WFH) or work-from-anywhere (WFA): The work environment is forced to be distributed across large heterogeneous networks with different security levels. We perceive hybrid work as a work mode where the team members follow WFH or WFA and work from the office. Therefore various security levels at the workplace and restrictions on informal team communications need to be taken into account.

We report on experiences from an industrial company producing software and cyber–physical systems. Initially set to update the existing secure code guidelines, the study lead to the discovery that it is crucial to go beyond an up-to-date set of security guidelines: it is mandatory to raise the cybersecurity awareness of those who are to follow the guidelines. We present a novel approach, via serious games, to train software engineers working in the industry, which is delivered in a hybrid mode and equips practitioners to face the challenges of hybrid work. Serious games have more than just entertainment purposes. They have proven effective ways to maintain engagement and boost training, particularly in cybersecurity. We developed and used two innovative serious games to raise cybersecurity awareness: (1) CyberSecurity Challenges (CSC), about how to develop secure software; and (2) Cloud of Assets and Threats (CATS), about cloud security, including its shared responsibility model. It is decisive for the industry that the software is written, developed, and deployed securely. The cloud service has replaced many on-premises deployments. It is essential to enable hybrid work, turning knowledge and practice about cloud security into essential capacities for professional hybrid work.

We provide the theoretical foundations for the two serious games and the overall approach. We also report and analyze more than 300 industry practitioners' training experiences from 2017 to 2023 and use this to evaluate the games. By applying serious games in the industry, among practitioners, we gain valuable experience in combining the advantage of different training modes and mitigating the disadvantage of online training. We observe the impact of serious games through a scientifically-sound approach based on the data and feedback we collected systematically from the trainers', trainees', and organization's perspectives. We show through empirical evidence that serious games are a successful approach for training conducted in hybrid work mode while providing authentic and immersed experiences that empower and raise cybersecurity awareness of current and future software professionals.

1. Introduction

Even before the COVID pandemic, some companies already enabled "work from anywhere" (WFA) practices. The trend of working remotely has been called "The World's Biggest Work-From-Home Experiment" (Banjo et al., 2020). Companies started to offer/request their employees to work in hybrid mode: some days from home/elsewhere

☆ Editor: Raffaela Mirandola.

* Corresponding author.
E-mail address: tiange.zhao@siemens.com (T. Zhao).

https://doi.org/10.1016/j.jss.2023.111946
Received 30 January 2023; Received in revised form 31 October 2023; Accepted 23 December 2023
Available online 2 January 2024

and the remaining days from the office. A recent McKinsey study by De Smet et al. (2021) discovered that nearly three-quarters of around 5000 employees would like to work from home for two or more days per week, which demonstrates how the pandemic has become a booster for digital transformation (Pillai, 2020). Universities are also benefiting from this transformation, as described in the work of Hashim et al. (2022). They show digital transformation as a propelling force used to build a competitive advantage for universities.

Cybersecurity is a software engineering topic that has been gaining a lot of attention — not only in the research community but in civil society in general. One of the reasons is the rising number of cybersecurity incidents that are made public to a wide audience, but also its nefarious consequences. Cybersecurity laws and regulations in Europe demand that cybersecurity incidents in critical infrastructures, particularly those that result in breaches of personal information (under EU Regulation 1725/2018), must be disclosed within 72 h after the incident is discovered (European Data Protection Supervisor (EDPS), 2018). One way software engineers address cybersecurity is by writing code with higher security quality, which leads to software that is more resilient to cyber-attacks and unlawful usage. However, previous studies (Gasiba et al., 2021c; Patel, 2020) have shown that software developers need to gain awareness of how to write secure software. Therefore, one fundamental problem that needs to be addressed is *how to raise awareness of secure coding of software developers in the industry*. In this work, we go one step beyond software development and also look at *how to raise awareness of cybersecurity in cloud deployments*.

In this work, we present:

1. an industry case study on secure coding guidelines in the industry;
2. an industry experience report on how one of the world's leading industrial companies has coped with the challenges and worked hand-in-hand with academia in building digital serious games for industrial practitioners in cybersecurity training to overcome the disadvantage of the conventional way of training;
3. a systematic evaluation and its result in three perspectives: trainers, participants, and organization.

Our industry case study started with highlighting the need to establish secure coding guidelines in the industry. Initially set to update the existing secure code guidelines, the case study discovered the crucial need to raise software developers' awareness of secure coding guidelines and company policies. This discovery leds the authors to develop and apply the first serious game – CyberSecurity Challenges (CSC) – aimed at empowering these industry practitioners working in a hybrid environment with adequate cybersecurity awareness about secure code development. The second serious game – Cloud of Assets and Threats (CATS) – followed the example of CSC and addressed the topic of secure cloud deployment as a key enabler for hybrid work.

Both games are developed under the design science research paradigm proposed by Hevner et al. (2004). We applied the action design research method in our study, for this method was crafted to facilitate work in an industrial context with the design of artifacts and evaluation of the usefulness of the designed artifacts. The field study approach or experimental method with comparable groups is not viable due to internal constraints and the business's need to improve the training. The design and evaluation of CSC started before COVID-19 and continued during the pandemic. The game was validated and finalized at the end of 2021 and became part of the industrial companies' curriculum. The design cycle of CATS started right after the outbreak of COVID-19 and is still an ongoing research project.

This paper is organized as follows: in Section 2, we describe the context of previous work on CSC and CATS; in Section 3, the work related to our research is introduced; in Section 4, we present in detail the research design we applied in the development of the present work; in Section 5 we describe the case study to provide details regarding the environment, in which we applied the serious games; in Section 6, we introduce the key elements of the game design and the integration of CSC and CATS; in Section 7, we describe the method used to evaluate the collected results; in Section 8, the collected feedback and the results of the conducted game events in hybrid work mode are presented in the participants' perspective, the trainers' perspective, and the organization's perspective. Then we share our thoughts about the collected results in Section 9; in Section 10, we discuss the threats to validity in our study and reflect on the research design. Finally, we conclude the contribution of our work in Section 11 and give a brief outlook on future work.

2. Context of this study and the serious games CSC and CATS

Typically, a CSC game event takes a full day. Multiple categories of challenges are made available for the players to solve during the game event. Players gain points when solving the challenges and the player or team with the highest points is the winner of the game. A CATS game event takes about one hour. In CATS, we prepared six attack scenarios that simulate attackers' activity against cloud assets. The players build their defense strategy by choosing from all the available defense cards and assigning them to the correct roles. The strategy will be evaluated, and the outcome is a probability of how likely the given defense will withstand the attack. If the probability is higher than the threshold, the player wins.

CSC is a genre of serious game developed to raise awareness of industrial software developers on the topic of secure coding and secure coding guidelines (Gasiba et al., 2021b). As Graziotin et al. point out in their work (Graziotin et al., 2018), valence and dominance positively correlate with self-assessed productivity. By applying CSC in training, we empower the practitioners in the industry to learn proactively, and we empower them to produce secure code.

CSC is deployed in a software and hardware platform, initially requiring a dedicated virtual machine per participant. The development of the COVID pandemic, with the absolute need to provide training in remote mode only, forced the evolution of the deployment design, first to use server-based virtual machines, and finally to a cloud provider (Gasiba, 2021). As we come to a new normal after the COVID pandemic, we benefit from the possibility of conveying the training onsite, with the co-location of all the trainers and trainees involved, together with a much more robust, scalable, and easy-to-use online platform.

CATS (Zhao et al., 2022) is a board game dedicated to improving cloud security for industry practitioners. In recent years, the number and size of cloud assets have been increasing rapidly. For instance, in China alone, the cloud infrastructure services market grew 45 percent to a total of US $27.4 billion (Canalys, 2021) in the year 2021. The importance of cloud computing is increasing; therefore, cloud assets must be protected accordingly. Additionally, it is of great importance that a common understanding of the shared-responsibility model (Anon, 2020a) is established among practitioners. Cloud service customers must understand that it is their responsibility to configure the cloud assets securely. Especially in a hybrid setting, when the responsible persons are dislocated, the task assignment of who should do what should be conveyed clearly. The work of Travers et al. (2022) recommends guaranteeing communication of the knowledge about the architectural artifacts and the practices across sites and using a team of architects to collect and distribute this knowledge. They emphasize this to be particularly relevant in global software development. In CATS, we engage participants in activities that imply sharing knowledge about the cloud defense strategy and contribute to defending the cloud assets through training on cloud-attack scenarios in a simulated and gamified environment.

Although CATS is a complete and self-contained game, the reusability and platform characteristics of CSC allow us to integrate CATS into CSC. In some of the game events conducted in the industry, we deployed CATS as a special category of CSC challenges. This paper

reports on our experience conducting several game events under the background of hybrid work. It shares a unique insight into industrial software engineering practices and how universities and industry can work together and create useful artifacts in practice under the design science research paradigm.

CSC and CATS are presented in our previous publications. In the present work, we extend the previous work (Zhao et al., 2022, 2021; Gasiba et al., 2021b,a), which is the application of both games in a hybrid work environment, and we present the context of industrial software engineering in the form of a case study and the newly collected data and results from the recent game events which were not included in the previous publications.

We observe that serious games help convey training in the industry, which is a fundamental way of empowering practitioners. This work contributes to extending the understanding of serious game usage in an industrial environment. Scholars in the academic world benefit from the work by gaining experience in instantiating the design science paradigm in an industrial setting. Industrial practitioners collect inspiration for how serious games can be applied and improve cybersecurity awareness, in particular, in a WFA setting.

3. Related work

This section describes the related work that we found helpful in our research. We organize the related work according to different topics.

3.1. Evolving workplace conditions and hybrid work

The important missions of modern software engineering education are to prepare software engineers to work in a hybrid mode and to address the need to enable them to write secure code and deliver secure products and services to the customer. Our workplace has witnessed rapid changes in recent years. The pandemic has challenged our work life and forced many individuals to work from home (WFH). A recent study by Galanti et al. (2021) has shown that WFH can bring advantages and disadvantages to the work environment. Nevertheless, due to the global pandemic, our way of work must be adapted to the evolving landscape. As Agba et al. described in Agba et al. (2021), work processes and practices are increasingly decentralized and adjusted with more WFH workers. The global workplace is also witnessing a decongestant trend, with a few staff in most organizations directed to work from the office while others work from home.

Cybersecurity and cybersecurity awareness have also experienced the shift towards WFH, as the work environment is forced to be distributed across large heterogeneous networks with different security levels. This raises questions on the importance of adapting cybersecurity work to WFH or hybrid working conditions. Furthermore, new work environments highly depend on network connectivity, the usage of various file-sharing software, and also software for online meetings.

Hybrid work requires the adaptation of classroom training. On the one hand, hybrid work brings more flexibility and saves commuting time; on the other hand, it reduces communication within the team: casual small talk for emotional bonding and information related to the work itself. For online or web-based training, it is a bigger challenge for the participants to be engaged by the topic. Online training takes longer because of the necessary breaks in between. The trainers and trainees need to learn to use new tools from a technical perspective. We will introduce how the company, in our case study, leveraged the advantage and overcame the disadvantages in the next sections. One inspiring study is by Borges, who surveyed 323 individuals with some team virtuality. He pointed out in his work (Borges, 2022) that job engagement is a relevant indicator of job performance, gamification can positively influence engagement and satisfaction, and higher team virtuality can lead to enhanced job performance.

We consider the evolving workplace and hybrid work highly relevant to our study because it helps to contextualize the current situation of the settings in our study after the pandemic.

3.2. IT security awareness and compliance with security policies

The work of Hänsch et al. on IT security awareness (Hänsch and Benenson, 2014), and its refinement by Gasiba (2021) is a theoretical foundation for our research. In their work, Hänsch et al. define three dimensions of IT security awareness: perception, protection, and behavior. According to Hänsch et al. perception is related to knowing about IT security, protection is related to knowing how to protect IT assets, and behavior is related to the intention to protect the IT assets actively. Gasiba refined these three dimensions to secure coding as follows: perception — knowing about software security vulnerabilities; protection — knowing how to protect against these software vulnerabilities; behavior — intention to write secure code. In this experience report, the concepts presented in these three dimensions are used to evaluate our artifact in the industry in the context of hybrid work and to understand how serious games affect the cybersecurity awareness level of the participants.

In the work of Petri et al. on MEEGA+ (Petri et al., 2016), a model they proposed to evaluate the quality of educational games, proposes different factors to evaluate the player experience and perceived learning. In our work, our evaluation focus mainly on the level of awareness in the three dimensions mentioned above. We find a certain degree of overlap between those three dimensions and the decomposition of quality factors proposed by Petri et al. In Gasiba (2021), Gasiba also investigates compliance with security policies as a means of behavior analysis. Neutralization theory captures the arguments employees use to rationalize security policy non-compliance. Siponen and Vance (2010) propose in their work that the “Defense of Necessity”, the need to get work done, triggers the intent not to comply with security policies. Compliance with IT-security policies is studied by Moody et al. (2018) in the Unified Model of Security Policy Compliance. This model results from a meta-analysis that marks a milestone as it synthesizes different studies on security policy compliance. Social factors and facilitating conditions are important for the intention to comply with security policies. Various studies find that deterrence or fear does not contribute to compliance with IT security policies. Note that these studies and models generally study white-collar work and are not tailored to the particular context of software engineers. Graziotin et al. (2015, 2018) studied human factors in code quality and found consequences of happiness and unhappiness that are beneficial or detrimental to developers’ mental well-being, the software development process and the produced artifacts. Again, this indicates that empowerment is prone to be more successful to raise awareness for secure coding than deterrence.

It is mentioned in the white paper from Secure Code Warrior (Secure Code Warrior, 2021) that developers do not need to become security experts. Still, they must be empowered to be their organization’s first line of defense. The contrast mentioned above, together with the fact that developers are perceived to be generally under pressure to deliver software with as little business risk as possible. However, the time and resources to help write secure code from the start are limited. Therefore good practical knowledge and a good awareness of cybersecurity are advantageous to raise the level of security.

3.3. Gamification and serious games design

Gamification and serious games are two different methods — albeit the two terms are often used to capture games and the process to enrich work or lifestyle tasks with game elements.

The work of Nieto-Escamez and Roldán-Tapia (2021) has shown that gamification can be implemented with traditional lectures and can be a valuable instrument during post-COVID times. The work of Vold et al. reported on migrating an escape-room style education game in computer science in an online learning format. The students enjoyed the game, and all 117 participating students found the game interesting and motivating. However, most serious games in computer science, or

more specifically, cybersecurity, in our case, do not address the context of security in software development, and only a few are tailored to industrial contexts. Also, many game developments have been done without properly evaluating their usefulness.

In the field of serious game design, it is beneficial to know about results stemming from gamification research, i.e., from enriching tasks with game elements. Serious games are defined as a type of game with more than just entertainment purposes. In the systematic literature review proposed by Subhash and Cudney (2018), 41 papers from 602 search results are studied, and they find that the successful implementation of gamification and game-based learning give reason to be enthusiastic about their application in higher education across various countries/student cultures, subjects, and formats. More specifically, the work of Markopoulos et al. (2015) study gamification in engineering education and professional training and conclude that gamification has a positive effect on engineering education by making difficult subjects more manageable, increasing intrinsic motivation, scientific knowledge, collaboration, interest and reduce or better manage the workload. In the field of serious game design, Dörner et al. established a baseline for developing serious games (Dörner et al., 2016). We followed and extended the baseline in our previous work Gasiba (2021) and Zhao et al. (2022) to design and validate our game.

Using a serious game or a gamification approach to cybersecurity does not guarantee success. In Barela et al. (2019), Barela et al. demonstrates that the game's design plays an important role. In their work, the authors applied interactive graphic storytelling to raise awareness of professionals in the industry to dangerous IT situations and scenarios. Their work shows that the methodology failed to meet its goals. The reasons given by the game participants for the failure include the usage of distracting images and preference for text-only material.

Within the scope of hybrid work, there is the trend of moving board physical games to the virtual platform. In 2022, Chukusol et al. (2022) investigate virtual board games and evaluate the virtual board games compared to the physical board game. The results of this study indicate that the overall composition of a virtual platform was comparable to that of a physical board game. Especially for the functional aspect, the study indicates that the virtual board game was as convenient as or better than the physical board game. In contrast, in other aspects, including the enjoyment aspect, virtual board games can be used the same way as physical board games. On the other hand, the social aspect was a slightly inferior one. Our work finds similar results as Chukusol et al. for the serious game with an educational purpose in the industry.

3.4. Existing serious games for cybersecurity

There are various serious games designed and implemented to raise awareness of cybersecurity. Shostack maintains a list of security tabletop games on his website (Shostack, 2021). One early example of a video game with cyber security simulation would be CyberCIEGE (Thompson and Irvine, 2011), where players purchase and configure workstations, servers, operating systems, applications, and network devices. They make trade-offs as they struggle to balance budget, productivity, and security. CyberCIEGE is mainly targeted at students in schools and institutes and successfully avoids the fear of failing by allowing the students to explore the simulated scenarios.

Another example in the company is the game Riskio designed by Hart et al. (2020). Riskio is dedicated to people without technical backgrounds and successfully increases cybersecurity awareness for people working in organizations. Riskio is a tabletop game focusing on defensive and offensive skills in IT security in general. It provides important insights into the impact of such serious games. CATS differs from Riskio because the target group is mainly industrial practitioners focusing on cloud security concepts. Comparing CSC to Riskio, the target group of CSC is people with a technical background, and the purpose is to empower the developers to write secure code. Additionally, both CATS and CSC focus on the defensive part of cybersecurity.

Another Week at the Office (AWATO) (Ferro et al., 2022) by Ferro et al. is another serious game developed based on a systematic literature review. The game focuses on the human factor by raising awareness of phishing attacks. The evaluation of the game shows that it is an effective tool for improving users' awareness of cybersecurity best practices. Their work further proves that serious games could be a useful approach to solving awareness issues.

The work of Švábenskỳ et al. (2018) shows another possibility of applying serious games in university teaching. They found that creating serious games contributes to fostering adversary thinking. Their study lasted over three semesters, and the game's purpose was to teach undergraduate students about network attack and defense by creating educational games in a cyber range environment. The students report they had a unique opportunity to understand the topic deeply. The game created is played by their college mates, who rated the quality and educational value of the games overwhelmingly positively. Their work shows exciting results in the academic environment.

One of the most obvious serious game approaches would be an online platform with programming tasks for programming education. One example is Codewars (codewars, 2023), which could improve the players' development skills by training with coding tasks that continuously challenge and pushes coding practice. This approach is helpful in self-paced training but is not applicable in our industry training setting. Another example is the capture-the-flag activity, where players play the role of an attacker in team red and the role of a defender in team blue and play against each other (HITB CyberWeek, 2020). The setting of our environment makes our study focus more on the defensive aspect of serious games.

While many games exist to raise cybersecurity awareness, the ones mentioned above can be considered to represent the variety of games and applications. These games provide evidence of the success of using serious games in cybersecurity education. In our work, we report on our experience in designing and applying serious games in training within the industry under different workplace conditions with work-from-home and hybrid work.

4. Research design

The study, consisting of its four main elements (1) the case study, (2) the design two serious games — Cybersecurity Challenges and CATS, and (3) the evaluation, reports on our experiences in empowering industrial software developers in secure coding. The Design Science Paradigm, according to Hevner (2007), guides the overall research design. The first element is the research design is a case study.

The case study on a project implementing secure coding policies identifies workplace conditions and the context of a successful Cybersecurity initiative in an international company. This study, which is presented here in a condensed and updated form, follows the guidelines on case studies (Eisenhardt, 1989), specifically by the eXperience method (Wölfe and Schubert, 2009) for developing case studies. Eisenhardt describes a case study as a research strategy focusing on understanding the dynamics within single settings. We use multiple data collection methods and continuous reflection on the insights gained in the project in the work context. The research team interpreted data with its backdrop of extensive industrial software engineering expertise. The project result is a secure coding policy adopted by the relevant stakeholders in the company. The stakeholders also agreed that more must be done for software developers' secure coding policy uptake in the software engineering processes.

The design science paradigm, according to Hevner et al. (2004), Hevner (2007), guides the two design studies, the design of the two serious games the Cybersecurity Challenges and CATS. The design of the CyberSecurity challenges aims to solve the practice-inspired problem of raising CyberSecurity awareness. This study follows the Action Design Research (ADR) method according to Sein et al. (2011).

Three design principles, namely Practice-Inspired research, Reciprocal Shaping, and Authentic and Concurrent Evaluation, in particular, characterize our research design for the CyberSecurity Challenges. The problem understanding is developed in the case study: the uptake of the secure coding policy and the three requirements of business alignment, mutual agreement and management endorsement guide the design and research activities. The ADR process for the CyberSecurity Challenges has three iterations and evaluation methods that fit the maturity of the designed artifact. Design and evaluation are done concurrently and authentically. I.e., software developers are the target group for our initiative, and in this context, focus groups to discuss the design and surveys are adequate methods. The impulses for the design gained the focus groups in the early stages of design, allowing for a fast closure on the serious game design. This allowed using summative evaluations and surveys to evaluate the stable prototype in the second iteration and the virtual prototype to respond to the changing workplace conditions with work from home due to the COVID-19 pandemic. The outcome of this design study is the serious game as the artifact; the initiative was aligned with business needs as outlined in the case study. Management, stakeholders, and most game participants consented that the CyberSecurity Challenges raise cybersecurity awareness for software developers. Positive evaluations of game design and game experience and on learning in surveys, observations by the researchers who offered the game as trainers in the company, and the management support are arguments that the CyberSecurity Challenges have the effect it is designed for.

The research design for CATS validates the findings on successful cybersecurity initiatives in organizations and the design decisions made in the CyberSecurity Challenges. A focus on security, the use of well-established sources, well-recognized best practices or standards for threats and security measures, and a target group, industrial software developers, are features of the design and the research process. Again, we have achieved an early closure on the game idea, game logic, and platform. The evaluations by players are positive in terms of gaming experience and learning. Again, players, management, and other stakeholders consent to the usefulness of the game. We adhere to the recommendations for good design science by Hevner et al. (2004), Hevner (2007): We have a viable artifact due to the research process, the CATS game; the problem is relevant, evaluating is done with rigorous methods, we formulate the research contributions, communicate our findings to academic and practitioners audiences, and follow a creative search process for the solution. We claim that CATS raises Cloud Security Awareness, works for both virtual and hybrid workplace conditions, and is successfully implemented in the company.

The design of the two serious games is shaped by the organizational context of working conditions. The CyberSecurity Challenges were designed for an in-presence experience and workshop setting and had to switch to a virtual format and work-from-home setting in the context of the COVID-19 pandemic and the CATS game began in the virtual and work-from-home experience and moved to an in-presence format. This is the setting for the empirical study with trainers and game participants on their perception of the serious games in the old on-presence and virtual and the new hybrid settings. The question is how to empower software developers to thrive in secure coding. The study with trainers and game participants to evaluate the serious game features for the Work-from-Home and the On-site games follows again guidelines in a focus group and data evaluation to ensure the validity of results.

The fourth part of the research design is an empirical study with an interview study and a survey to assess the in-presence and the virtual game formats to develop a Serious Game format that fits in the new era of Hybrid work.

The whole research design spans the time period from 2019 until 2023, and with it the pre-COVID situation of mobile work, the pandemic with its impact on digitalization and new forms of work, and the new era of Hybrid Work.

We acquire the participants' consent to participate in serious games during the training. Since the content of delivered training must remain consistent, it is hard for us to do a blind study with controlled groups, and the number of samples is limited. Since the training hour is immensely valuable and data collection tightly controlled, we cannot apply evaluation processes such as distributing pre-and-post questionnaires. However, we can use the feedback round to collect the participants' opinions on the training and organized game event.

This industrial setting is one of the main reasons we follow the action-design research method to carry out our study. The result of the case study is the research setting of the design of two cybersecurity games.

The collected results will be presented and discussed in Sections 8 and 9.

5. Research context: a case study on leveraging secure coding in the industry

This section presents a case study on a project to increase the levels of security in software engineering, carried out in the year 2019 in an international company that provides software and cyber–physical systems that are used in critical infrastructure. The case study captures an initiative to update the company's existing secure coding guidelines to reflect the latest advances in technical know-how and industry best practices. The main outcome of this project was a set of secure programming guidelines. All stakeholders approved the new set of secure coding guidelines, baselined them, and deployed them as an internal policy. The case study identifies the success factors of the project, and describes what has been done to identify secure coding guidelines that are relevant for an organization and what remains to be done to effectively implement secure coding according to the new set of guidelines in the organization for excellence in secure software development.

The case study is written from a perspective of a security expert. The case study focuses on a small department in a multi-national organization that delivers software to other parts of the organization, which then provide software and cyber–physical systems to Critical Infrastructure. The company itself wants to remain anonymous.

5.1. Motivation and trigger of activities

Software used in Critical Infrastructures needs to be secure and implemented in line with modern ways to provide products and services. Software companies that deliver software to be used in critical infrastructures need to fulfill specific IT security standards that mandate a secure software development lifecycle. In particular, both security and safety are highly relevant in software development when this software is to be part of a critical infrastructure. A big push is being felt throughout the industry to introduce measures that lead to producing high-quality, reliable, and secure software.

Over the last years, the media and security professionals have been calling attention to software vulnerabilities and the fact that software developers keep making the same programming mistakes (Poston, 2019; Vaughan-Nichols, 2019; Connory, 2019; Darling, 2018; Schneier, 2020). Acknowledging this factor and recognizing the security requirements imposed by standards, an improvement program initiative was started at the company to address the current status quo of software development and improve its current guidelines, processes, and documentation.

Since a secure software development lifecycle existed in the department, it was decided by higher management to address the documentation baseline of secure software development first. This baseline needed to be updated.

5.2. Company profile

The company in this case study provides services and solutions for Critical Infrastructures to various end-customers globally. The company operates mainly in the energy sector, according to the classification given by the German Federal Office for Information Security (Bundesamt für Sicherheit in der Informationstechnik — BSI (BSI, 2020)). In 2020, the company had over 40.000 employees in several department branches, with about 3% working for Finances, 9% for Human Resources, Legal, Supply Chain, and Solutions, about 50% for Services, 30% for Operations, and 8% for other activities. The software produced in the company may be deployed to critical infrastructure in a product or service after several years of product or service development and coding. It may run in critical infrastructure for decades.

This case study focuses on a relatively small department inside the company working on and being responsible for software analysis and simulation of instrumentation for critical infrastructures. The department is not an operator and does not interact with end customers. However, it provides software incorporated as part of end products for critical infrastructure, which is then utilized and integrated into other products by other divisions inside the company. The other divisions then offer these products directly to end customers or in a contractual agreement, in which the company operates the infrastructure directly.

The company of this case study wishes to remain anonymous. Hence it is referred to as the company. A researcher, a higher management representative, a lead developer, and an external consultant conducted the case study.

5.2.1. IT security in the company

Since the company works with and delivers products for critical infrastructures, it is very self-conscious of the need to be excellent in cybersecurity. The company is also strongly motivated to comply with cybersecurity standards, resulting in a commitment to the end customer, which translates into a low-risk appetite. This requirement applies to the whole business and business sectors of the company. The important cybersecurity standards that apply to this industry and are relevant to this case study are IEC 62443 (Anon, 2007), in particular, IEC 62443-4-1 (Anon, 2018a) and IEC 62443-4-2 (Anon, 2018b), and ISO 27001 (ISO27001, 2017). The parts of ISO 27001, which are particularly interest for this case study, are part of Annex 14 of ISO 27001.

The workforce of software developers is heterogeneous: there are employees with many years of software engineering experience and also young software developers fresh from universities. Both young and experienced developers might be reluctant to follow current computer science or cybersecurity advances and best practices in secure coding. A heterogeneity of technical backgrounds and secure coding awareness can also be found in the group of lead developers. It turned out to be a challenge to balance the different points of view between the established developers and the younger generation.

5.2.2. Risk analysis

The software developed at the department is integrated into final products used in critical infrastructures. As such, the same safety, security requirements, and goals are shared across different divisions. Regarding software development security, these requirements are also driven by the IT security standards that the company wishes to comply with and internal guidelines on software excellence.

Developing and delivering software integrated into the company's end products exposes these software components to attacks that can result in severe business consequences. From coding, it can take several years until the software is deployed to the customer to run for years, even decades. A lot can happen over the lifecycle: patches, re-configurations, and an evolving threat landscape. The risk levels are constantly monitored, and appropriate measures are taken to address weak links so that the company fulfills its accepted risk appetite established by higher management.

This case study addresses one of the actions that was started by higher management to lower the company's risk levels. These accepted risk levels by higher management consider the possible consequences of a breach in security. To address these negative consequences, the company puts much effort into secure software development to ensure that the software it produces meets high-quality standards.

5.3. Secure coding guidelines

A typical coding guideline contains instructions that show or discuss programming constructs that are valid in the programming language (i.e., do not lead to compilation errors). However, these constructs can lead to severe vulnerabilities if used in real code. Typically, software developers must spend time and effort developing software to avoid these vulnerabilities.

An example of a secure coding guideline warns the software developer to be aware of integer overflows and to protect the software to avoid this, e.g., as given by the Common Weakness Enumeration CWE-190: Integer Overflow or Wraparound (MITRE Corporation, 2006). The corresponding secure coding guideline from Carnegie Mellon's SEI-CERT is INT32-C: Ensure that operations on signed integers do not result in an overflow (Carnegie Mellon University, 2023).

Secure coding guidelines are specific for a programming language and specific for the domain: Java differs from C++ in terms of typical weaknesses, and the secure coding topics differ, e.g., between coding for an operating system or IoT device or for a business process.

For software developers to consider secure coding guidelines, they need to be aware of the guideline in the first place. Furthermore, they need to know what to do to comply with secure coding guidelines and have the necessary resources for the extra effort to provide functionality and security. Secure coding guidelines typically include examples of how to write the same code so that the issue does not occur (in this case, a possible integer overflow). In most cases, the software developer must write additional code to avoid security issues.

Not respecting the secure coding guideline does not necessarily mean that a piece of software contains a vulnerability; even if it does, it might not be exploitable. Most software vulnerabilities, however, can be traced back to the non-compliance of one or more secure coding guidelines (Department of Homeland Security, US-CERT, 2023). Knowing and correctly applying secure coding guidelines requires these guidelines to be defined and documented and their usage to be mandated by the company, e.g., through internal policy.

5.4. Updating the secure coding guidelines

The project to update the secure coding guidelines can be described by one goal and three requirements:

- **Update secure coding guidelines**: update the existing secure coding guidelines with more modern and updated guidelines that address the current state-of-the-art and industry best practices, with the following requirements,
- **Business Alignment**: in order not to cause unnecessary stress to the software developers, the number of secure coding guidelines should be kept at a minimum; the minimal number of secure coding guidelines are derived by alignment with business goals and priority of the guidelines,
- **Mutual Agreement**: the secure coding guidelines should have a mutual agreement of all the lead developers and address their experience, expertise, and concerns,
- **Management Endorsement**: the goal was that the updated document be endorsed by management to turn it into a mandatory policy for a broader number of software developers.

The previously existing secure coding guideline internal document was taken as a starting point to select the sources. The external consultant proposed using additional secure coding guidelines from external parties as a reference — instead of writing all coding guidelines from scratch.

These standards and textbooks were used to determine an initial set of coding guidelines. The decisions to select this initial set were aided by additional sources, comprising information on previous software incidents and lessons learned from previous projects. Note that, additionally, to secure coding guidelines, the lead developers also wanted to address core guidelines and clean code guidelines related to code style and software architecture. It was decided that these documents would also be evaluated and considered from a secure coding perspective. Workshops were conducted to identify relevant coding guidelines.

The guidelines were placed in the Excel tracking document and underwent several sorting, classification, prioritization, and verification iterations. In verification, it was determined how compliance with secure coding guidelines could be ensured. Options considered for verification were manual verification, automated verification, code review, and check-in trigger. The project's final activity was to present the guidelines to management, review the document and seek approval and implement it as policy in the company.

The original secure coding guideline document consisted of 12 secure coding guidelines. After all the decisions taken during the joint workshops, the total number of secure coding guidelines went up to 31, and nine were kept from the original secure coding guidelines document. In this process of defining the guidelines, a format for presentation and additional information on how to implement and verify the secure coding guideline was developed and used for a uniform and easy-to-read presentation of guidelines.

Establishing the coding guidelines included document reviews, several workshops, and a presentation to management. As per the project goal, the resulting secure coding guidelines document receives backing and support from higher management, addresses the specific business case, and distills the experience from the lead developers and the external security expert.

The main outcome is a document containing the updated secure coding guidelines in the form of a secure coding policy.

5.5. Next steps: awareness training and integration in the software development lifecycle

The project's initial goal was to update the secure coding guidelines with new state-of-the-art information and get a wide-range acceptance from management, lead developers, and software developers. From experiences with the previous set of guidelines and its adoption in the company, it was apparent that more measures are necessary to ensure the uptake of the new secure coding policy.

The main points that need to be addressed are an awareness training campaign and the secure coding guidelines integration into the company's software development lifecycle. The rationale behind these proposed action points is to (1) lower the burden of compliance towards secure coding guidelines by automating as much as possible and (2) recognize that automation tools are not sufficient to detect and eliminate weaknesses in the code; the software developers need to be trained in secure coding and in particular in the newly defined secure coding guidelines. The analyses emphasized the role of the individual and IT security awareness as well as empowering the software developers to be excellent in secure software development.

The project to create the CyberSecurity challenges as a serious game to raise awareness for secure coding was the result of the considerations to how to facilitate the uptake of the guidelines and how to empower industrial software developers for more excellence in secure coding.

5.6. Secure cloud: the follow-up security project

The project of establishing the secure coding policy and the CyberSecurity Challenges as serious games for awareness was considered a success story in the company. While the activities to implement the serious game as part of the companies training activity curriculum unfolded, the topic of cloud security became more prominent — also through the need to advance cloud security as part of hybrid work both in the company but also in critical infrastructure for which the company operates digital infrastructure and services. Shared responsibilities for a secure cloud were determined to be the topic to be addressed to increase security.

As part of the company, the department decided to follow the example of secure coding guidelines to identify cloud security rules and raise awareness for secure cloud usage through a serious game. To identify cloud vulnerabilities, information from cloud security alliance (CSA) and MITRE ATT&CK cloud matrix (Anon, 2020b) were employed. Cloud security alliance (CSA) has listed the 11 most common cloud security issues in "Top Threats to Cloud Computing The Egregious 11" (Cloud Security Alliance (CSA), 2019). The listed cloud security issues include: Data Breaches, Misconfiguration, Inadequate Change Control. To address those potential issues due to the nature of cloud deployment and the human factor, the company puts much effort into cloud security training in providing an example of implementation recommendations to ensure the cloud assets are configured securely, and the standards are met.

The guidelines were turned into elements of a serious game. This game was designed to raise awareness for secure cloud programming and deployment and to empower developers for secure coding for open architectures and business models.

5.7. Summary of the case study

This case study describes the process of how to update secure coding guidelines in an organization, some of the practices of secure software development in the industry, and, in particular, that secure coding policies need to be specific to be effective. The case study also illustrates the requirements to be aligned with the business, the mutual agreement of relevant stakeholder groups in the organization, and management endorsement to leverage the guidelines to a policy. Thirty-one guidelines form the secure coding policy of the company after the six-month project of updating the C/C++ coding guidelines.

We argue that the situation described in the case study is of general interest: organizations need to increase the security of products and code and to do so, security in software development needs to be raised. A secure coding policy is an instrument to do this, and tool support in the software lifecycle is important. In the end, software developers must be able to identify weaknesses, understand how to write secure code and they must be willing to make the necessary effort to produce secure code. This means it is necessary to raise awareness for the topic and train developers in the necessary skills — in a way that this works for software developers with different backgrounds and levels of seniority.

This case study captures the company's situation, describes the problem and the context in which the serious game is a means to raise awareness and provide training, and the research process to design the games are situated. The situation, the three requirements of business alignment, mutual agreement, and management endorsement that we identified in the case study, continue to guide the research and design activities for the two serious games in this context.

6. Design of two cybersecurity games

This section presents the design of two serious games: the CyberSecurity Challenges (CSC) and Cloud Assets and Threats (CATS). The design of CSC started in 2017 to address software developers' lack of awareness of secure coding, as we previously discussed in the industry

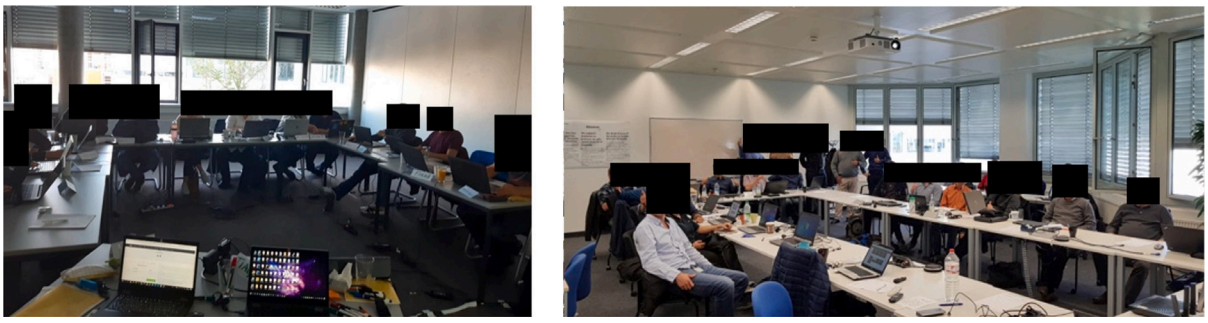

Fig. 1. CyberSecurity events — onsite events.

case study section. One of the conclusions from the case study is that software developers need guidance to understand the purpose and reasoning behind the need to follow secure coding guidelines. We use two methods to achieve this goal: (1) through a serious game and (2) through interactive discussions resulting from playing the game. This methodology has proven very successful during the design of the CyberSecurity Challenges, as shown in Gasiba (2021). As such, the game has originally been developed as an onsite event. However, in 2020, the game had to undergo significant changes to adapt to new working conditions caused by the COVID pandemic. After the adaptation, the industry used the game as a standalone event. The resulting design of the game, which can then be used in a hybrid environment (onsite, remote, or mixed), is very successful in the industry. Among others, one factor contributing to the game's success is its adoption into the standard training curriculum in the company where the game was originally developed. Furthermore, this successful design, combined with the rich set of lessons learned, led to the creation, in 2021, of another serious game — Cloud of Assets and Threats (CATS) in the same company, to raise awareness of the different roles needed in the secure deployment of cloud assets.

This section presents a detailed overview of both games and briefly discusses the design cycles undertaken during their design. We also reflect on the impact of the work environment in the design of these games and on factors that led to the successful usage of the games in the industrial context.

6.1. CyberSecurity challenges (CSC)

This game was designed in the industry, with the first design, instance, and evaluation appearing in 2017. Until the end of 2019, the game was used globally as an onsite event. In particular, the nine events carried out during this period occurred in Germany, China, and Turkey. Fig. 1 shows two examples of onsite CSC events in the industry.

This game was developed in three cycles utilizing an Action-Design Research (Sein et al., 2011) approach. In the present work, our game presentation follows the last design cycle. The game's design started with training content that had the goals of informing about threats and training secure coding, i.e., defensive skills. It moved from this Defensive/Offensive content to purely defensive content and a sole idea to focus only on the defensive security skills necessary in secure coding.

In the next sections, we will briefly describe the game's design and look at the results obtained from the evaluation of the game throughout the entire design process.

6.1.1. Game design

Fig. 2 depicts an overview of the CyberSecurity Challenges (CSC). To play the game, its participants (i.e., software developers) form teams competing against each other to solve secure coding challenges. The challenges consist of exercises related to the topic of writing secure code. In Gasiba et al. (2020c), an overview of the design of individual challenges is given. The participants must show knowledge of and apply secure coding guidelines in their solutions, e.g., through multiple-choice questions or by writing secure code on the Sifu platform, the game platforms we designed (Gasiba et al., 2020a).

In addition to the participants, trainers, and challenges, the game contains a dashboard and a countdown timer. The dashboard displays the available challenges and the number of points that can be earned by solving them. The number of points per challenge reflects its difficulty level — simpler challenges award less than harder ones. It is also used to monitor the individual team's solved challenges and amount of collected points. Fig. 3 shows an example of a dashboard. The implementation of the dashboard is based on the open-source CTFd component, which is widely used by practitioners who develop and design this type of game.

Upon solving a challenge, the team receives a random-like string (the challenge's flag) that can be pasted into the dashboard to solve and collect the challenge's points. The countdown is used to control the game's duration and automatically lock the dashboard at the end of the game, thus preventing teams from further submitting flags and collecting points. Since the game is time-limited, the players' competitiveness to solve the challenges is incentivized.

Game coaches aid the teams and participants during the game. The instruction from the coaches ensures that the team or the player can handle the challenges. The coaches also monitor the game to ensure that the desired game objectives and learning goals of the game session are achieved.

At the end of the planned game duration, the team that has achieved the highest number of points wins. Although this contracts the game's competitiveness, solving secure coding challenges, discussions, teamwork, exchange of ideas, and the fun aspect that participants experience while playing the game make all participants winners of the game. Although no long-term studies have been carried out during the evaluation of the game, literature hints at possible positive long-term effects of playing similar serious game (Zhao et al., 2023), including knowledge retention and the possibility of increased compliance to secure coding guidelines (Bulgurcu et al., 2010).

The challenges that were developed and evaluated in the design of the CSC focus on the following software development aspects:

1. secure development of web applications
2. secure development of C/C++ applications
3. secure development of Java applications
4. secure development of infrastructure as code for Cloud Environments using Terraform.

The selected secure coding guidelines standards for the different types of software development are given in the following table Table 1.

Participants solve the challenges in three steps: *Phase 1* - introduction, *Phase 2* - challenge, and *Phase 3* - conclusion. In the first step, participants are introduced to the challenge, the scenario, and the goals of the exercise. In the second step, the participants solve the exercise by the necessary means to achieve this goal (e.g., by answering multiple-choice questions or writing secure code in a special platform). In the

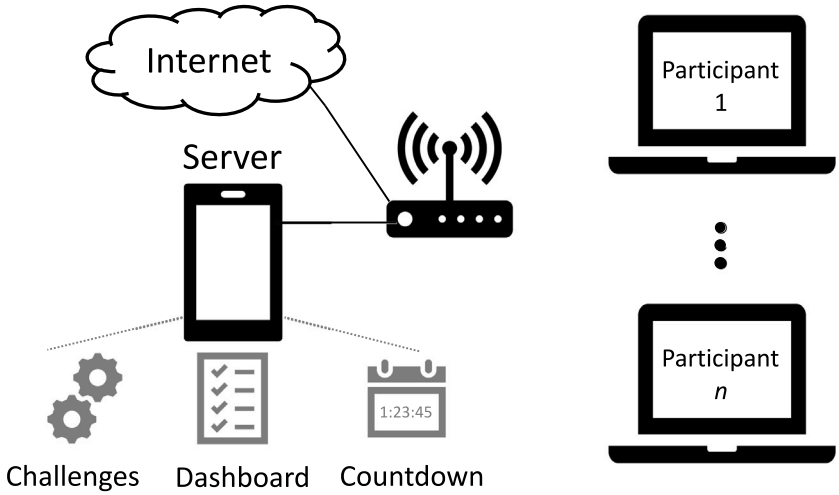

Fig. 2. Architecture of CyberSecurity Challenges infrastructure.

Table 1
Selected secure coding standards for CSC challenges.

Challenge type	Secure coding standard	Reference
Web	OWASP	OWASP Foundation (2001)
C/C++	C/C++ SEI-CERT	Software Engineering Institute, Carnegie Mellon (2018, 2023)
Java	Java SEI-CERT	Carnegie Mellon University (2020)
Terraform	Own developed guidelines	–

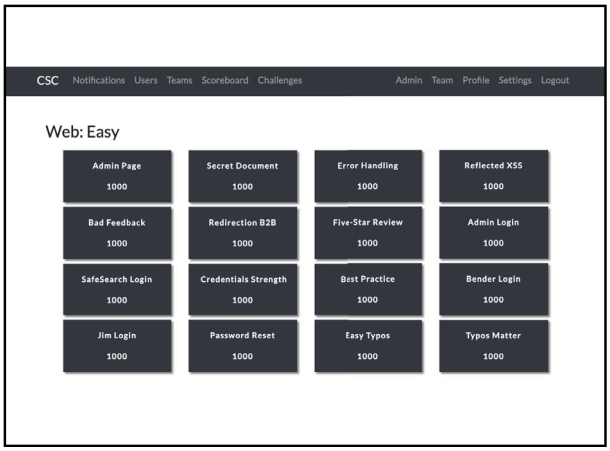

Fig. 3. Dashboard.

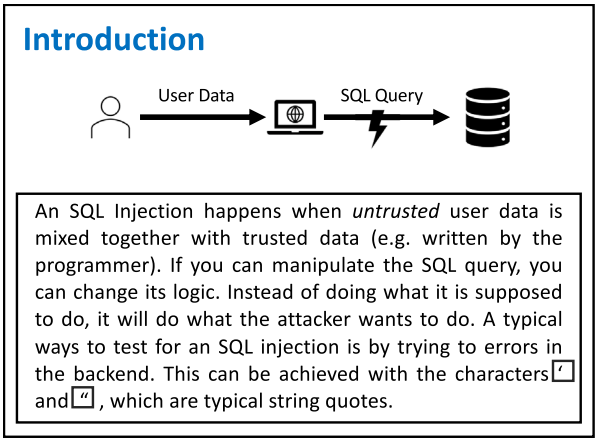

Fig. 4. Web challenge: Phase 1.

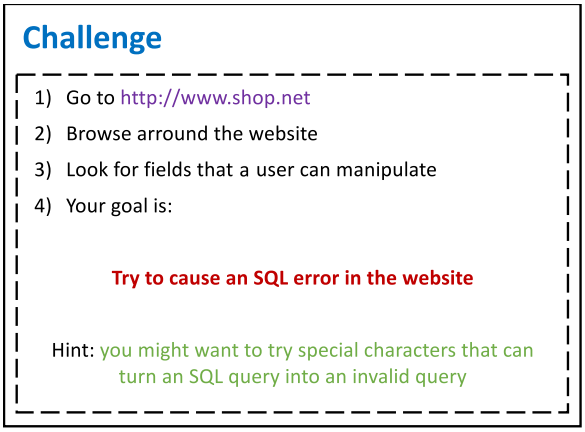

Fig. 5. Web challenge: Phase 2.

third step, an explanation of the exercise is provided to the players to solidify the lesson learned. This phase can optionally include additional questions. Upon achieving the third step, the exercise is considered solved, and a flag is shown to the player, finalizing and concluding the challenge (see Fig. 5).

Figs. 4 to 6 shows an example of a web application challenge, including the dashboard and the three exercise phases. The learning goal of the exercise in the example is to raise awareness of SQL injection vulnerabilities, how to identify them, and how to avoid them when developing web applications. Table 1 shows the mapping of all the challenge types and secure coding standard.

As previously discussed, to solve CSC challenges, the participants might also be required to interact with a unique platform. We call this the Sifu platform, as the word Sifu originates in the word *teacher*. Thus, this platform aims to teach software developers how to write secure code, i.e., to raise awareness of secure coding guidelines. Fig. 7 shows the user interface of the Sifu platform.

A software project containing one or more security vulnerabilities is hosted in the platform. Through the interface, the player can browse, read and modify the files that are part of the software project.

To solve the challenge, the players need to identify and understand the software vulnerability present in the software, and they need to fix

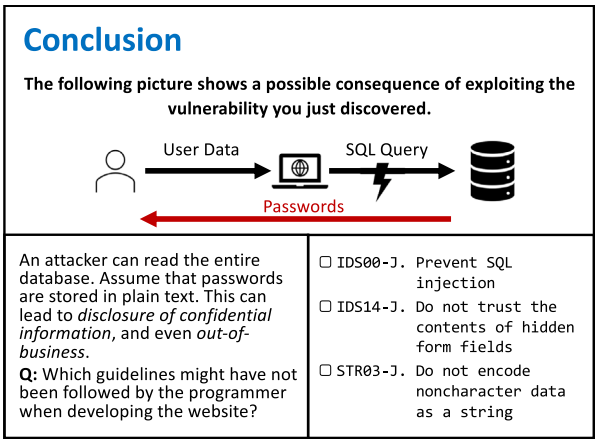

Fig. 6. Web challenge: Phase 3.

it. Fixing the vulnerability is achieved through successful interactions with the platform. The players submit their solutions to a backend component that performs several analysis steps to determine if the solution is acceptable. The condition for the acceptability of a solution includes: (1) challenge vulnerability is fixed, (2) no additional vulnerabilities are introduced, and (3) the code performs the intended functionality as stated in the introduction. The player can move to the third and final challenge phase if the solution is acceptable. If the solution is not acceptable, feedback on the reason for non-acceptability is provided to the player using an intelligent coach.

The analysis on the backend uses several techniques to determine if the submitted code can be considered an acceptable solution. In particular, the following methods are used in the backend: warnings and errors of the compiler, static code analysis, dynamic code analysis, unit tests, and security tests. The intelligent coach component is implemented through an artificial intelligence technique. The feedback provided by this component consists of hints that guide participants to understand the problem in the code and assist the participant in improving the code to bring it to an acceptable solution.

Detailed information on the implementation of the platform is available in Gasiba et al. (2020b,a). Furthermore, the authors release the Sifu platform as an open-source project, which can be downloaded from Github (Gasiba, 2020).

6.1.2. Online CSC and onsite CSC

The game's design, since it is mostly based on participants carrying out interactions through an internet-connected device, has allowed the authors to carry out the game onsite and as a fully remote event. However, the main aspects that needed to be changed and adapted between the two different work environments include (1) team formation and team discussions, (2) coaches monitoring and interaction during the game, and (3) deployment scenarios.

In the initial game design, the players were physically separated into teams by table arrangement while being placed in a single room. This allowed for quick interactions between the different members than constitute each team. Furthermore, the coaches would be moving around the room, inspecting the screens of the different team players. If players need help solving an exercise, they can easily call a coach, e.g., by simply raising their hand.

The game dynamic was adapted in the later game design, which reflects onsite events. Here, the participants need to initially connect through online meeting software — with all the participants in the same online meeting room (called the main room). Teams are formed through the assistance of a previously prepared online whiteboard (e.g., using Conceptboard). During the main game event, the participants must be placed in separate online breakout rooms according to their team members. The coaches must enter the individual online breakout rooms to monitor the game's evolution. Also, a common channel (the main room) needs to be monitored and kept open when an individual developer requires help from one of the coaches. Due to its nature, the online event is limited in the interaction between team participants, coaches and teams, and even individual teams. To guarantee the game's success, the coaches need to take a more intense and proactive role in the game — not only by listening to the discussions but also by asking questions or giving small unsolicited hints to the participants.

The deployment scenarios also needed to be adapted to the different work environments and situations. In the initial game designs, a physical machine was brought into the room where the game event took place. Therefore, it was physically necessary to be present in the room to be able to play the game. The participants needed to connect to the SSH tunnel to access the challenges, and some participants had difficulty setting up the SSH tunnel. Additionally, if the game event is to be held in countries other than Germany, export control must be in place to comply with company policies. Besides that, using a single machine to host the challenges can lead to scalability issues. The infrastructure was deployed in a cloud provider later in the game development stages. In this case, it was possible for participants coming from different parts of the world and living in different time zones to join the game event. The cloud deployment's scalability overcomes the initial game design limitations, and the players no longer need to set up an SSH tunnel since the challenges are accessible via the browser. The convenience allows the participation of players without a technical background to be able to enjoy the game. Since the cloud provider hosts the server, the deployment complies with the company's export control policy.

For more information on the game design, the reader is referred to Gasiba (2021) for a complete description of the game's design through the three design cycles. In Gasiba et al. (2020c), an overview of the design of individual challenges is given, and the Sifu platform is published under an open-source license (Gasiba et al., 2020a). Evaluating the different scenarios has shown several advantages and disadvantages for each deployment scheme. We refer the reader to Gasiba (2021) for an in-depth discussion of this aspect.

6.2. Cloud of Asset and Threats (CATS)

The next serious game we present is CATS, which stands for Cloud of Assets and Threats. It is a serious game designed to raise awareness about cloud security challenges among industrial practitioners. At the beginning of our research, we envisioned the game as a board game. However, given the pandemic was in its early phase, having a face-to-face game event was not feasible. Therefore we adapted the game as an online board game on our purposely built digital platform. The player can join as a single player or play in teams on the digital platform. We pre-defined six different attack scenarios derived from real-world cloud attack activities as reported by Homeland security (Anon, 2020b).

The flowchart in Fig. 8 shows the game process. As the game starts, a tutorial scenario is presented so the players can get familiarized with the game platform. Then the players select another attack scenario. A predefined attack is shown in each scenario, and the players are prompted to solve the scenario by building a defense plan using the available cards. The evaluator will assess the defense plan returning a success rate, which describes the probability that the submitted defense plan withstands the given attack scenario. If the calculated success rate reaches a predefined threshold, a flag is revealed to the players, and they can move to the next attack scenario. Players can adjust their defense plan according to the hints available on the game interface if the success rate does not reach the threshold. The players are allowed to repeat this step as much as needed.

An overview of CATS game features is summarized in Table 2. Players can join in teams or as single players. In our game events, we

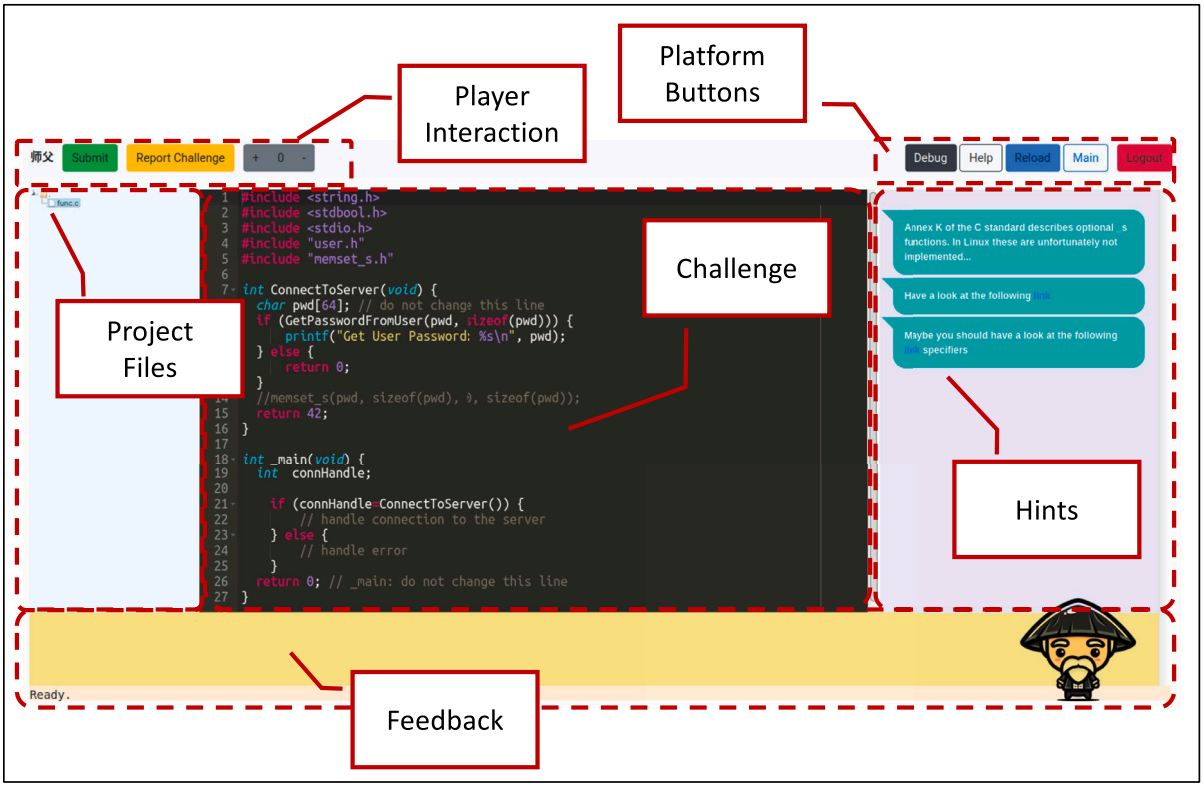

Fig. 7. Sifu platform — User interface.

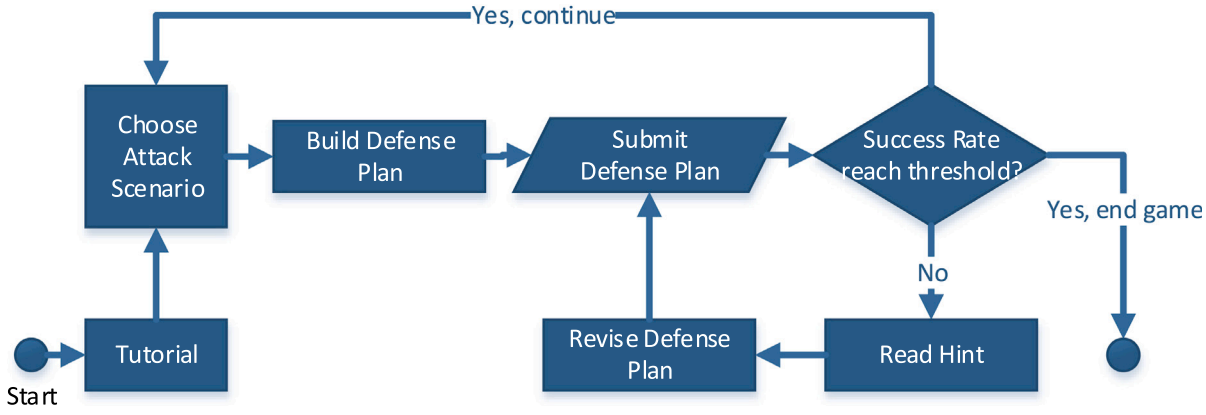

Fig. 8. The game process from player perspective.

Table 2
CATS game feature overview.

CATS game feature overview	
# of players in a team	~4
# of attack scenarios to solve	6
Average time of play	60 min
# of cards in pool	23
# of cards for a defense plan	6
Average # of submissions to solve one scenario	<13

have approximately four players in one team. Based on our observation, this is optimal to maximize communication within a team without disturbance. Solving all six attack scenarios took each team/player 60 min. The players are supposed to choose six cards out of 23 available defense cards and assign them to either "Business Responsibility" (2 cards) or "Technical Responsibility" (4 cards) to build a full defense plan.

6.2.1. Game elements

The game interface of CATS is shown in Fig. 9. On the top left corner is the area for building a defense plan. The players should pick cards in the defense pool area and assign the picked cards as "Business Responsibility" or "Technical Responsibility". The predefined attack scenario is described in the top middle area. A detailed description and manual are provided to the players for further description. On the right, there is a "submit" button. By hitting this button, the defense plan is submitted to the evaluator and assessed against the attack plan. The result can be found as a percentage number in the top right corner. If it is higher than the threshold, this scenario is solved, and the players can move on to the next one.

The attack scenarios are drawn from real-world cybersecurity attacks that have occurred and were reported. In CATS, we use the MITRE

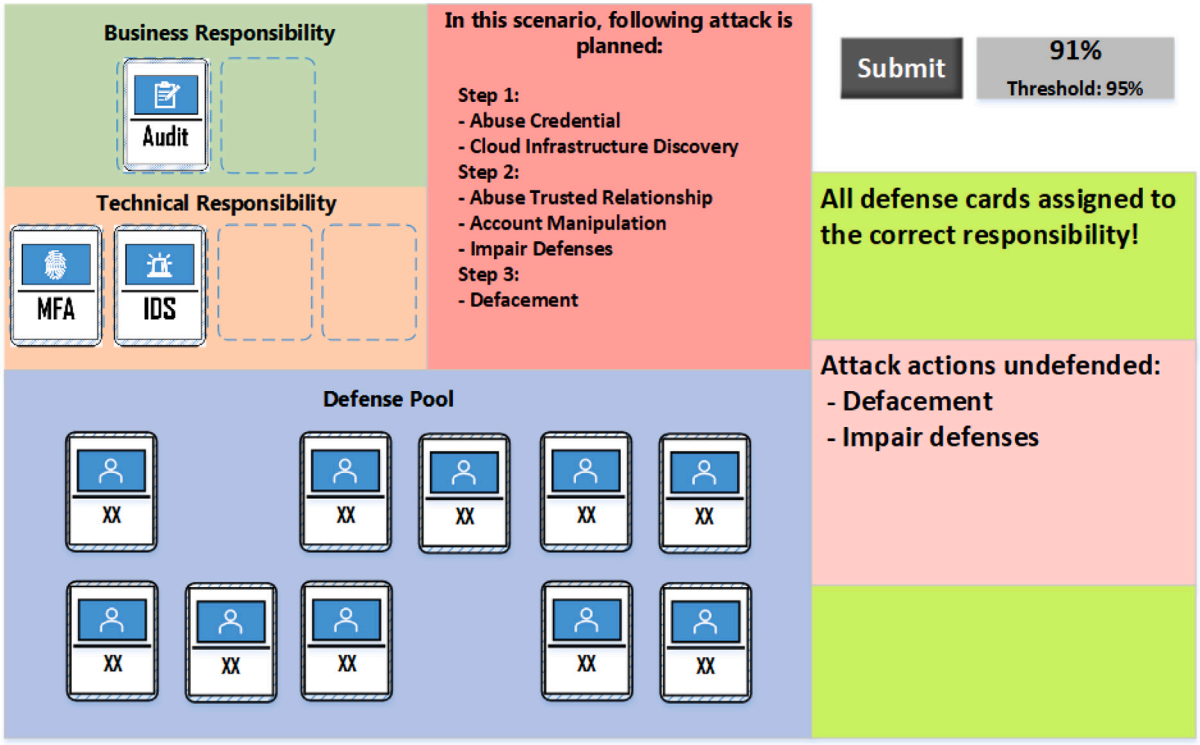

Fig. 9. Mock-up of the game interface.

Table 3
Overview of CATS game events organized in 2022.

Game event	Date	Player	Team	CSC or training	Valid submissions
1	2022–01–21	17	4	CSC	177
2	2022–03–15	14	–	Training	477
3	2022–03–22	14	–	Training	493
4	2022–03–29	13	–	Training	312
5	2022–04–14	13	4	CSC	178
6	2022–04–26	11	–	Training	100
7	2022–05–03	8	–	Training	171
8	2022–06–02	4	2	CSC	169
9	2022–09–29	14	4	CSC	298
Total number of players				108	
Total number of submissions				2375	

ATT&CK cloud matrix (Anon, 2020b) to derive the mapping between the attack and defense cards. The attack kill chain is abstracted into a three-step pattern: initial access, launch attack, and make an impact. We derived six attack scenarios in total.

6.2.2. Reusing CSC platform

CATS can and was deployed as a category of challenges in the CSC platform. The player sees a flag when an attack scenario is solved. The player can input the flag into the CSC platform and collect the points. The deployment is done with an automated Terraform script (Anon, 2021b,a).

6.2.3. Game events organized

In the year 2022, we have organized 9 CATS game events, as shown in Table 3. Four of them are integrated with the CSC platform and deployed as challenges in CSC. Five of them are included as a part of a full training day. In training, the basic concept of cloud security is introduced. After the training, the trainees are invited to join the CATS game to exercise what they learned in training.

6.3. A successful game event

CATS game integrated into CSC secure coding challenges is a balanced instantiation of the game event in a hybrid training environment. CSC focuses more on secure coding technical know-how. In training, developers find it interesting and engaging. However, in a CSC event, the participants are in mixed groups, including developers and managers. CATS complements CSC because it focuses more on strategic planning in cloud security defenses, being more relevant for managers than the secure coding challenges. When developers and managers work as a team, they find the combination of CATS and CSC engaging and helpful; both can contribute to the team's winning.

From our previous game events, the game design shows positive results for both games. We learned the following lessons:

- Participants welcome hands-on exercises.
- Serious games can keep trainees engaged during a training session.
- It is important to automate the deployment and recycling of the game infrastructure.
- It is important to align the activities with the business context regarding business alignment, mutual agreement, and management endorsement.

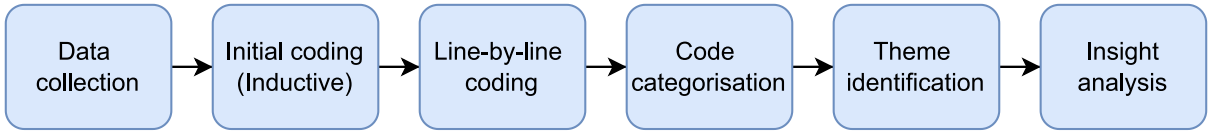

Fig. 10. Qualitative data analysis process.

CATS was developed after CSC, and we could use lessons learned in developing CSC and adapt the design to an online environment with positive preliminary results. The results of individual games are discussed in separate publications; this work focuses on applying those serious games in the industry under hybrid work.

More details about the game CATS are introduced in previous work (Zhao et al., 2022, 2021,b, 2023).

7. Evaluation methodology of games in a hybrid environment

Game design and evaluation began with face-to-face serious games, and then the games moved to a purely virtual setting. At the end of 2022 and beginning of 2023, the question is what is better for software developers: face-to-face or virtual, and how should a format look like that makes use of the best of the two worlds in the post-COVID time, when all game participants have practiced their virtual and hybrid work skills.

We wish to capture three dimensions: trainer, participant, and organizational perspective. The trainer's perspective includes the preparation, delivery, and feedback of the training, and from the trainer's perspective, we would like to compare conventional training with hybrid training. From the participant's perspective, we focus on the feedback about the game experience. The organizational perspective captures the lessons learned on what is needed to conduct training in hybrid mode.

7.1. Trainer's perspective

From the trainer's perspective, the goals of our evaluation are: (1) the effort made to adapt the on-premises training to online training. (2) the pros and cons of training in hybrid mode from the trainers' perspective.

We apply a systematic, qualitative content analysis process following the guidelines by Corbin and Strauss (2014) based on the methodology and process steps proposed in Crosley (2020), Frampton (2020). Fig. 10 describes the steps of our data analysis. The process starts with data collection through the semi-structured interview (SSI).

We conducted an online SSI with 7 trainers in our organization. All interviewees were expert trainers before the pandemic, and they experienced the digital transformation of training in hybrid work. In the interview, we asked them two questions:

- What is better in online training compared to onsite training?
- What is worse in online training compared to onsite training?

During the SSI, we transcribed the answer provided by the interviewee or let the interviewee write down their answer to the questions above as an online survey. The audio of the interview is not recorded since all the texts were saved, and an audio recording would not be necessary. We removed sensitive information from the answer we collected and applied a Python library (Mueller, 2021) to generate a word cloud as shown in Fig. 12.

Table 4 summarizes the general information regarding the conducted SSI. We invited ten trainers who hold training events about secure coding in various programming languages, operational technology (OT) security, security activities and processes, and threat & risk analysis. Seven trainers accepted the invitation and provided their answers to the questions above.

Table 4
Semi-structured interview general information.

Semi-structured interview general information	
# of invited trainers	10
# of collected result	7
# of interviewed female trainers	2 (28%)
# of interviewed male trainers	5 (72%)
# of interviewed trainers (>10 years of training experience)	3 (42%)
# of interviewed trainers (5-10 years of training experience)	3(42%)
# of interviewed trainers (<5 years of training experience)	1(16%)
Average time for the SSI	39 mins. 46 s

Original text in interview	Category Tags	Sentiment
"Participation of participants located gobally is easier...but (they) cannot chat with each other in coffee breaks."	Flexibility Networking	5/5 1/5
"...Preparation of online training is easier..."	Preparation	5/5

Fig. 11. Example showing category assignment and sentiments.

Then, we applied an initial inductive coding process to the free text we gathered during the SSI, as suggested in the process on Fig. 10. All the coding processes mentioned below are done manually because the sample size is small and manual coding provides the best precision with a small sample size. The purpose of initial coding is to identify the essence of the text and code it accordingly. The inductive approach develops the initial code set descriptively in this step. The next step is line-by-line coding, in which the text is reviewed and coded line-by-line. In this step, details will be added to each line. For example, in the initial coding step, when the interviewee mentions the new tools used in online training, it is coded as "tool/technology". In this step of line-by-line coding, it was coded more specifically as "meeting software", or "virtual whiteboard"; this allows us to dig deeper into the data we collected in SSI. After that, the code categories are created to organize the data and guide the analysis in the next step. Based on the result of code categorization, the themes are identified and articulated. In the last step, insight analysis, we derive the insights based on the steps before and produce a narrative that answers the goal of our evaluation.

In the insight analysis, we assign emotions or sentiments to each piece of data in a 5-level Likert scale (strongly positive - 5, positive - 4, neutral - 3, negative - 2 and strongly negative - 1) (Godsay, 2015) and combine the sentiments to draw a conclusion. Fig. 11 shows two examples of how the original texts are categorized and rated according to sentiment.

7.2. Participant's perspective

The evaluation methodology from the participant's perspective is presented in separate parts for CSC and CATS in this section.

7.2.1. Participant's perspective for CSC

The evaluation of the CSC game carried out between 2017 and 2020 consisted of fifteen events in the industry. Nine events were carried out onsite from November 2017 to October 2019. Six events took place

Table 5
CyberSecurity challenge events.

No.	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*
Cycle	1	1	1	1	2	2	2	2	2	3	3	3	3	3	3
Type	D/O	D/O	D/O	D/O	D/O	D/O	D/O	D/O	D/O	D	D	D	D	D	D
Date	11/17	5/18	7/18	7/18	9/18	7/19	7/19	9/19	10/19	6/20	7/20	7/20	7/20	11/20	11/20
NP	11	12	6	30	16	14	15	7	23	15	21	20	15	12	4
Where	DE	DE	DE	DE	DE	CH	CH	DE	TK	OL	OL	OL	OL	OL	OL

D/O: Defensive/Offensive Challenges, **D**: Defensive Challenges, **NP**: Number of participants, **DE**: Germany, **CH**: China, **TK**: Turkey, **OL**: Online.

online and were carried out from June 2020 to July 2020. Table 5 summarizes all evaluated events. Our results and conclusions are based on data collected from more than 220 participants.

A survey was created to evaluate and refine the CSC game. The survey is based on a 5-point Likert scale of agreement. Collected answers include two negative (-) answers (strongly disagree and disagree), a neutral (N) answer, and two positive (+) answers (agree and strongly agree). Participants in the survey, which was not mandatory, received a small briefing on the study and freely consented to participate. Furthermore, the collected answers have been anonymized. Since participation in the survey was not mandatory, the number of collected survey answers from the participants was 95.

7.2.2. Participant's perspective for CATS

From the participant's perspective, the goals of our evaluation are: (1) whether the participants find the game enjoyable, and (2) whether the participants find the game helpful in raising awareness about cybersecurity.

We invited game participants into an open discussion after each game event to evaluate from the participant's perspective. We collected the feedback in the discussion round. All the participants in one game event were in the same virtual meeting room, and we asked the question: do you have any feedback regarding the game event? The participants take turns answering this question. The length of the discussion depends on the number of participants in the group. On average, each participant took less than one minute to answer the question. The reason of directly asking is to guarantee that we get feedback from all participants and we also a chance to clarify in case of confusion. Additionally, we also provide an anonymous questionnaire with the same question: do you have any feedback regarding the game event? This helps the case when the participants do not want to share their thoughts in the group.We summarize and highlight the important feedback in Section 8.

Our evaluation uses the feedback from discussions and hereby differs from similar studies. In the work of Monasor et al. (2014a), a simulation-based training environment in global software development is evaluated. This work uses a multi-phased evaluation method in Monasor et al. (2014b). In the heuristic evaluation, a baseline, pre-training, post-training, and opinion questionnaire was used in the evaluation process. The work of Vizcaíno et al. (2019), follows a similar approach via a pre-and-post questionnaire. In our setting, we organized feedback rounds as an authentic way for evaluation due to limitations mentioned in Section 5.7. Note that we have analyzed the game dynamics to evaluate the players' performance in CATS (Zhao et al., 2023) and in CSC (Gasiba, 2021). This evaluation indicates the engagement of the players in the serious games, different strategies, and, remarkably no brute-forth strategies to win against the game engine. In this work, we focus on the participants' opinions regarding the games in the present work.

7.3. Organization's perspective

From the organization's perspective, the goals of our evaluation are: (1) whether the transformation from on-premises to online training has any negative impact on the number of training being booked (2) within an organization, how well is the training with a serious game is being accepted.

To evaluate from the organizational perspective, we counted the number of conducted game events during the pandemic outbreak (the year 2020–2022). Additionally, we provided the rating of CATS and an analysis of the requirements and the goal that are part of the problem understanding provided by the case study.

8. Results

This section provides the results and our interpretation of the trainer's, participant's, and organizational perspectives.

8.1. Trainer's perspective

We collected the answers to the SSI mentioned in Section 7 and went through an analysis process as illustrated in Fig. 10. We group the categories into seven themes and counted how many times a certain theme has occurred in the SSI and assigned a sentiment scale each time it occurred. It needs to be noted that in the interview with one training, a theme could occur multiple times. Based on the sentiment scale and the raw data, we derived insights on each theme. Table 6 provides a summary of the results.

The most frequently mentioned theme is "Tool/Technology" from the trainer's perspective. Adaptation of conventional training in a hybrid environment involves a great number of tools, such as virtual whiteboard and meeting software, etc. The new tools make the material easier to read than in the on-premises training where a poor-quality projector could leave the text barely readable. The trainers are in general positive about the improvement on "Tool/Technology", except for the fact that the training in a hybrid environment depends heavily on network quality and the inadequate webcams usage during the online session.

The next frequently-mentioned theme is "Participation", where the trainers feel slightly negative about it. In the hybrid environment, it takes more effort to keep the participants engaged and stimulate interactions. Additionally, the trainers miss the direct feedback from the participants, thus, it becomes difficult to judge whether the messages are clearly conveyed to the trainees.

The theme "Flexibility" is mentioned 16 times in the SSI, reaching 4.4 on the sentiment level. Training in hybrid environment allows greater flexibility geographically across different time zone. The reduced business trips contribute to a lower carbon footprint. However, some are missing the business trips which are perceived to be "spice" to the daily work.

"Training organization" is mentioned ten times in the SSI with a rating of 2.6 on the sentiment level. To organize the training in hybrid environment, additional work needs to be done, e.g., re-designing the group work. Some trainers are neutral about the new "Training organization", some complain about the additional effort required for adaption.

The theme "Networking" is mentioned eight times in the SSI with the lowest rating of merely 2.3 in the sentiment level. Trainers observe that in hybrid environment, networking still remains as a weak element. Due to the missing coffee breaks, participants lose the opportunity for casual information exchange.

The theme "Preparation" is mentioned only five times in the SSI, however, the sentiment level is overwhelmingly positive with 5.0 point.

Table 6
Theme identification and insights from the trainer's perspective.

Themes	# of occurring in SSI	Sentiment on average	Insights
Tool/Technology	19	4.4	Trainers are concerned with the new tooling in hybrid work and find tools helpful in training as an improvement. There are some complaints about dependencies on network quality and the availability of webcams.
Participation	16	2.5	For training in hybrid environment, it is challenging to keep the participants engaged and to remain the same level of interaction as on-premises training. Trainers no longer get direct feedback from trainees in hybrid environment.
Flexibility	16	4.4	Training in hybrid environment comes with great flexibility, allowing global participation despite the geographical and time differences. Business trips are cut regardless, which reduces carbon footprint.
Training organization	10	2.6	Work needs to be done for training the organization, e.g., re-designing the group work, introducing more breaks, etc.
Networking	8	2.3	Trainers are worried that the networking opportunities are reduced significantly when there is no coffee breaks and informal idea-exchanging channels are missing.
Preparation	5	5.0	The preparation for training in hybrid environments is easier and more efficient with higher re-usability.
Managing COVID pandemic	1	5.0	Training in hybrid environment helps managing COVID pandemic.

(a) Wordcloud: what is better in an online training

(b) Wordcloud: what is better in an onsite training

Fig. 12. Wordcloud: comparing online and onsite.

Trainers agree that the preparation of training in hybrid environment is easier. The theme "Managing COVID pandemic" is mentioned only once by one trainer interviewee, however, we consider this theme equally important for the reason that online training avoid gathering and therefore is helpful in preventing pandemic from spreading.

We generated a word cloud based on the collected text for both questions. The resulting word clouds for what is better in online training and for what is better in onsite training are shown in Fig. 12.

8.2. Participant's perspective

The results from the participant's perspective is presented in separate parts for CSC and CATS in this section.

8.2.1. Participant's perspective for CSC

Table 7 shows a summary of the evaluation of the game throughout the second the design cycles, which took place from 2017 to 2019, and includes the nine events.

Note that we want the data collection to be authentic, and thus the data collection in the survey only start after the first design cycle when the game idea was validated. Nevertheless, direct feedback from the participants has shown that an improvement was achieved from the first to the second design cycle. The respondents answered the survey in a 5-point Likert scale (strongly negative, negative, neutral, positive, strongly positive) and the results are abstracted in three points: negative answers, neutral answers, and positive answers (Jacoby and Matell, 1971).

The full statement in the survey (used in the second and third design cycle) is listed as the following:

Table 7
Summary of evaluation of CyberSecurity challenges through design cycle two (2017–2019).

Measurement dimension	Negative	Neutral	Positive	Evaluation construct
1. Learned new techniques and principles	12.5%	7.1%	80.4%	Awareness - Protection
2. Understand the importance of secure coding guidelines	0.0%	5.3%	94.7%	Awareness - Perception
3. Improvement of secure coding knowledge and skills	3.6%	14.4%	82.1%	Awareness - Behavior
4. Learning goals are clearly understood	8.9%	8.9%	82.2%	Game-play
5. Help from coaches was adequate	1.8%	12.5%	85.7%	Game-play
6. Feel more prepared to handle secure coding at work	8.9%	26.8%	64.3%	Work

Table 8
Summary of evaluation of CyberSecurity challenges through design cycle three (2020).

Measurement Dimension	Negative	Neutral	Positive	Evaluation Construct
1. Learned new techniques and principles	0.0%	10.0%	90.0%	Awareness - Protection
2. Understand the importance of secure coding guidelines	0.0%	0.0%	100.0%	Awareness - Perception
3. Improvement of secure coding knowledge and skills	0.0%	0.0%	100.0%	Awareness - Behavior
4. Learning goals are clearly understood	8.0%	8.0%	84.0%	Game-play
5. Help from coaches was adequate	0.0%	0.0%	100.0%	Game-play
6. Feel more prepared to handle secure coding at work	0.0%	20.0%	80.0%	Work

1. By participating in this awareness training, I learned new techniques and principles of secure software development.
2. I understand the importance of secure coding guidelines.
3. Focusing on the challenges improves my practical secure coding skills.
4. The learning goals of the challenges were clearly explained.
5. The help from the coaches was adequate.
6. By participating in this awareness training, I feel that I am prepared to handle secure coding-related issues at work

Also presented in the tables are game measurement dimensions related to three evaluation constructs: awareness, game-play, and work-related feedback. For awareness, we differentiate the three awareness dimensions by Hänsch and Benenson (2014): protection, perception, and behavior. These dimensions relate to survey questions that were asked of the participants.

Inspecting Tables 7 and 8, we note that, in the second design cycle, the game design already shows promising results, as most of the feedback from participants was positive. The least amount of positive feedback was obtained for the behavior construct of work, with 64.3%. Furthermore, the highest amount of uncertainty through neutral answers was obtained for the same construct. This indicates that a significant amount of participants are not sure about the usefulness of the game for the work environment. However, the game designed and evaluated in the third design cycle shows an improvement in the work construct to 80.0% agreement, while one in every five participants is still unsure, and no negative feedback was obtained.

At a large scale, we can observe a significant improvement in the feedback from the participants from the second to the third design cycles. In particular, for the awareness construct of protection we obtained an improvement from 80.4% to 90.0%, for perception, an improvement from 94.7% to 100.0%, and for behavior from 82.1% to 100.0%. Furthermore, we observe a significant improvement in terms of decreased number of negative feedback and decreased amount of neutral answers.

The results presented in both tables allow us to conclude that both games, as designed in the second and third cycle, are appropriate to raise awareness of secure coding guidelines of software developers in an industrial context. Furthermore, the game designed in the third design cycle has obtained a higher amount of positive feedback, leading to the conclusion that the participants prefer this game to the game designed in the second cycle.

The results related to game-play are of special importance to the work environment under which the game is played. In particular, our results show that in both onsite events and also online events, the goals of the challenges were clearly understood by the participants (see measurement dimension 4 in Tables 7 and 8). Furthermore, for both work environments, the participants of the survey considered the help provided by the coaches as adequate (see measurement dimension 5 in Tables 7 and 8). This aspect emphasizes that in both work environment scenarios, it is important that the game coaches follow the game-play and help the participants along the way.

Although our game evaluation, which started in 2017, did not focus on the work environment, the authors collected feedback from the participants, which is relevant to understand how the work environment can affect the usage of a serious game to raise awareness of software security. The following table summarizes participants' quotes that are relevant to this aspect.

In Table 9, we show relevant quotes from participants, covering both positive and negative feedback from the participants.

In transferring from on-premises training to online training, the most obvious challenges are:

1. the network quality of the participants, since all the communication with teammates and coaches and accessing the gaming platform relies on the network quality.
2. the complexity of accessing the infrastructure since the game-play involves multiple platforms, e.g., the dashboard to keep track of the current score, the countdown for timing, the website where challenges are displayed, and the communication channel with teammates and coaches.
3. the fact that the participants can be potentially distracted.

The collected results highlight the importance of the interactions between the players and that communication can be a limiting factor. While in an onsite event, communication is easily achieved, this aspect needs to be well-planned for online events. Another interesting aspect of the participants' feedback is that they associated the game with fun and learning. This aspect requires further studies to understand the relationship between the fun of playing a serious game and the learning effects that it causes. Here, we refer to related studies (Mirkovic and Peterson, 2014; Cheung et al., 2012; Cullinane et al., 2015) that indicate strongly that a relationship exists between participating in the game event and learning. Quote number five in the same table provides further hints in this direction.

Other important aspects to the success of the game, which are related to the work environment, are those connected with infrastructure and ambience. Onsite events, since they are done in a more controlled environment, show good results for infrastructure access. However, online events have been shown to result in considerably more problems with infrastructure access. This result was surprising — since the participants in the online types of events necessarily access the game online, it was assumed that connectivity would be fine. However, the participants conducting work-from-home activities had different internet access quality (e.g., bandwidth and latency). Furthermore, the

Table 9
Collection of quotes from participants of CyberSecurity challenge game events.

ID	Feedback	Quote
1	Positive	*I really enjoyed the teamwork during the game*
2	Positive	*Playing the game was lots of fun, and I have learned many new things*
3	Positive	*The coaches did a good job in helping our team to solve the challenges*
4	Positive	*I learned a lot from the discussions we had during and after the game*
5	Positive	*I am eager to put to practice all the knowledge I have now learned*
6	Negative	*We had problems with the access to the infrastructure*
7	Negative	*The music played during the game was very distracting*
8	Negative	*I was not fully aware of what my team colleagues were doing*
9	Negative	*Communication with team members and with coaches was difficult*

Table 10
A selection of feedback we collected for CATS game event.

ID	Feedback	Quote
1	Positive	*Very nice and well-developed game.*
2	Positive	*Enjoyed. More interesting (than training without serious games).*
3	Positive	*It is great that the exercises are wide spread, so that also non-developer (application owner + tester) could be of help and had fun.*
4	Positive	*Very interesting activity! Thank you so much! It is possible to learn new technical vocabulary.*
5	Positive	*It is great to have hands-on experiences in building a cloud defense strategy! I enjoyed the game.*
6	Negative	*More time for the game.*
7	Negative	*For each measure selected, a description of its influence should be given after submission (for the player) to profit from playing.*

participants accessed the game from different parts of the world, and access to foreign resources was sometimes restricted. On the other side, in the onsite events, since the connection to the infrastructure was made locally, only a few problems occurred, and those that happened were also solved very quickly.

The ambience in which the participants are put can impact the game. In the first game version, the coaches decided to play some background music to incentivize the game-play. However, participant feedback has shown the opposite as true, i.e., participants found the background music disturbing. In terms of work environment, our experience has shown that onsite events are preferred to online events since, in the latter, the coaches have yet to foresee, plan or change the environment of the remote participant. Finally, our experience has shown clear advantages towards the onsite event, as opposed to the online event. In the onsite event, the participants reserve the time for the game and focus on it the entire time. However, some participants can get distracted in online events - e.g., with incoming emails, phone calls, etc. One of the main lessons we learned from planning both onsite and online events is the importance of planning, communication, and expectation management — particularly before the event takes place.

8.2.2. Participant's perspective for CATS

After each CATS game event, we invited the players into an open discussion to collect direct feedback, and we provided a questionnaire to them for written feedback. Table 10 provides a selection of the feedback we got for CATS. Table 9 in Section 6 provides a selection of feedback we got for CSC. As shown in Table 10, most of the game event participants could understand the game logic of CATS and found the game helpful, interesting, and well-developed. A survey focusing on the participants' feedback on CATS can be found in our previous work (Zhao et al., 2023), where 96% of the respondents agree or strongly agree that playing CATS "helps me to understand cloud attacks and defenses". The game allows participants working in different roles to cooperate and accomplish a common goal: defending their cloud assets. Some complained the time assigned to the game was too short or some explanation was missing. Note that we would improve the game artifact based on the feedback. Generally, we take both points as positive signs: the players are curious about the game and want to spend more time with the game.

8.3. Organization's perspective

The CSC game events are summarized in Table 7 in Section 6. Between 2017 and 2020, there were 15 game events organized. For CATS, we have organized nine game events and three trial runs since February 2021. CATS has a rating of 3.96 stars out of 5 stars based on the result of the questionnaire we distributed. It is important to note that this results from all the historical game events of CATS, combining all design iterations.

At last, introducing serious games into training in the industry does not impact the satisfaction of the training negatively but rather positively. Both games are adapted to online events and can be flexibly adapted to be either stand-alone or embedded in a workshop format with other training modules. From our experience, it seems that the uptake of the game worked well — the travel restrictions and the lifting of the travel restrictions did not have a significant impact on the proliferation of the game and how the company-internal bookings of the training events took up speed.

The organization itself has committed itself to excellence in cybersecurity and a low-risk appetite. The games contribute to the strategy. The organization empowers the employees in secure coding, and it, furthermore, continues to empower all employees in choosing their professional training across the organization. The contents are well-tailored to the organization's security topics. Mutual agreement among employees and management and alignment with the business needs is important to thrive for the initiative in the organization.

9. Discussions and lessons-learned

The two serious games have been evaluated, and the results were mainly positive. Players find both of the games enjoyable, as suggested in Tables 7, Tables 8 and 10, and perceive an increase in secure coding knowledge.

In this work, we have highlighted the importance of raising cybersecurity awareness in the industry. The importance of cybersecurity is not only motivated by the criticality of the products the industry develops and sells but also by security standards that must be followed and the ever-evolving threat landscape. Hybrid work adds to the challenges

of secure coding and the importance of raising awareness for coding guidelines.

Our work has started to motivate and highlight the need to raise awareness of secure coding with an industrial case study. In our case study, we have looked at the complexity to face when adopting a secure coding standard in an industry context, the need to consider the business case, the diversity of the workforce, the changing workforce, the need for compliance with standards and norms, the need to be respectful with budgets, the alignment with management and the consensus among experts. This case study also illustrates the benefit of tackling code weaknesses in coding. Having a policy as a written document backed by management and software developers is not enough: raising awareness and empowering software developers is necessary. Another important aspect highlighted in the industry case study concerns software developers' compliance to secure coding guidelines. It is necessary to convey the reasoning and motivation behind the standards and guidelines to software developers. Simply informing about secure coding is not enough: knowledge, perception, and behavior, as the three dimensions of awareness, should be addressed.

Our work understands awareness in three dimensions (perception, protection, and behavior) following the conceptualization by Hänsch and Benenson (2014) and Gasiba (2021). The dimension of perception is related to the situational awareness of software cybersecurity in knowing the possible threats to software and having general knowledge about information security. Protection relates to software developers' knowledge about the measures to protect against software threats. Finally, the behavior dimension relates to the software developers' attitude towards cybersecurity in code — this relates to how developers think and act, and comply with securing coding policies.

In our work, we studied the common weakness in coding and cloud security. Since our study is based on the public weakness repositories and common secure coding standards, and the design elements are abstracted based on publicly-known security incidents, the gaming approach can be generalized and applied in other companies, organizations and settings. Our case study describes what has been done to identify relevant challenges or tasks for designing and integrating such serious games. Other companies might find it useful to apply our game and our challenges or refer to our method to identify challenges that are more relevant to their settings. In that sense, our serious game approach can be generalized and applied in other settings.

Our work focuses on raising awareness of cybersecurity is using serious games. Towards this, we started in 2017 with the design of the CyberSecurity Challenges to address the topic of secure coding guidelines for the target group of software developers. Both junior and senior developers and industry practitioners need to be engaged in secure coding and share responsibilities in cloud security. Also, the approach needs to respect budgets and alignment with management: interactive, hands-on exercises demonstrate the problem and develop the skills to deliver the solutions.

In the design of the CyberSecurity Challenges, the shift from a game with a strong focus on understanding threats to a serious game with a focus on coding guidelines helped to increase the outcome of the game in a relatively small time frame of a couple of hours of workshop time. Software developers are playing this game, and the design process came soon to useful solutions: the feedback from industrial software developers was professional. It could be integrated into the design process very effectively.

Evaluation of the games was a core activity in the research process. The format changes from discussion rounds to surveys and focus groups to maximize the impact of the evaluation according to the maturity of design and implementation in the process.

Since our game development is scientific, our work is available via GitHub and scholarly publications. The evaluations of the games indicate success: the games are fun to play, and we have data about face-to-face and virtual settings and what works best in both worlds. Another point of success that highlights the usefulness of our games is that we have received feedback in practice from software developers who changed their minds and posture towards how to write software, particularly on the need to write secure software. This point directly relates to the problems observed in the industry case study.

The fact that we receive positive feedback in evaluation from the game participants on the evaluation of the usefulness of the game implies that our approach finds success in the industry and that serious games are helpful in secure coding education. Furthermore, another hint of the industry's success is that the game has been established as part of the company's official software developers' curriculum. Several cybersecurity experts internally scrutinized the game, and its adoption resulted from a mutual agreement. Previous studies show that increased compliance can improve overall security, i.e., more secure software.

Our work also extends the understanding of serious game design for an industrial environment. We identified a rich literature set related to the design of serious games for cybersecurity. We also identified various theories on compliance with security policies. Our work contributes to understanding the particular work setting of software developers in the industrial context. A strict focus on the topics for the daily work and immediate relevance of the skills trained in the game is important — besides the fun factor. Alignment with budgets, higher management, and consensus among lead developers on the material taught are other aspects that need to be considered.

However, only a few design studies on Serious Games in cybersecurity are being rigorously researched and evaluated, and some even need clarification on the usefulness and effectiveness of the approach. In our work, the CyberSecurity Challenges and the CATS game have been rigorously researched and evaluated and follow an Action-Design Science and Design Science approach by Sein et al. (2011) and Hevner (2007). We address in our work two important industrial topics: secure coding and secure cloud deployments.

Since the year 2020, lots of efforts have been made to adapt to hybrid work during the pandemic. The restrictions imposed during the pandemic have caused a significant impact and changes in many aspects of the way we work, train, learn and communicate. The field of software engineering was able to react very rapidly and cope with hybrid work. We attribute this ease of adaptation to the nature of software engineering, as software developers are used to interacting with computers in their daily work. What we discovered in our experience by applying serious games in training has reassured us that cybersecurity training in the industry could also be adapted relatively easily to a hybrid environment.

As a trainer, in the beginning, it is stressful to cope with the changes and transform the training material to an online version to allow hybrid work. Once we establish an online training framework, it becomes easier to set up online training than transitional ones. Our experience has shown that it is important to keep the trainees engaged and motivated in online training in a hybrid environment, to which the existing serious games have contributed tremendously.

From the participant's perspective, serious games are an innovative approach to learning about cybersecurity facts and raising awareness about cloud security and secure coding issues. It helps to motivate the training participants and keep them focused and engaged during the online-training session. In some cases, it can compensate, to some extent, the missing group spirit and encourage discussions within the group in an online breakout room. For organizations, developing serious games and integrating them into training improves the quality of training and the participants' satisfaction. Under the background of hybrid work, efforts need to be made to adapt the existing training framework to online training. Our experience reported in this work provides a successful story of how this can be achieved with serious games. Once the adaptation is made, preparing for the individual training sessions will take less time. Participants, trainers, and organizations could benefit in the long run from work.

Our continuous evaluation of the usage of serious games in the industry started in an onsite environment and later moved to an online event. Although the environments changed during the evaluation

process, we still received positive feedback and have seen success in our game. These facts indicate that serious games are a good approach to raising awareness of cybersecurity in both types of environments and raising awareness on software engineering.

From our experience, onsite training events still provide unique networking and social interaction opportunities that cannot be easily replaced in remote environments. In the now emerging hybrid environment, some participants are still attending events remotely, and another is already attending events onsite. The opportunities created by the development of fully remote events have allowed addressing the hybrid workplace more naturally. Early indicators in our continued efforts highlight the continued success of our approach. Our experience allows us to conclude that onsite events are preferable (in terms of learning effect) to online events, as there is no possible perfect replacement for social interactions. During onsite events, the participants are focused and only concentrate on the training content. However, during remote events, the participants can be easily distracted by, e.g. emails, calls, etc., which can impact the effectiveness of the learning process. On the other side, online and hybrid events have the advantage of being flexible compared to onsite events. Participants from different time zones can easily and cost-effectively participate in the joint event. Physical and mentally fit participants can participate in online events without any issues. Moving from onsite, face-to-face to virtual, and now hybrid, is a relatively seamless process, and the game design transitioned quite easily.

Upon comparing both types of events, providing an excellent solution and training environment with online events is possible. It is more effective to do the training as an onsite event. There are virtual elements that most likely will be kept in the options on how serious games in training can be played.

In our experience of adapting on-premises cybersecurity training to online training in hybrid work mode with the enhancement of serious games, we have learned the following lessons:

- Transforming from onsite training to online training could be stressful for the trainers due to the disadvantages (see Table 6).
- Serious games help mitigate those disadvantages, e.g., lacking engagement, communication, and interaction (see Table 10).
- By following the action-design research methodology and the guidance of the design science paradigm, successful serious games can be designed and integrated into the organizational training framework in the industry.

10. Reflection on the research design and threats to validity

The study, with its main elements (1) the case study, (2) the design of the two serious games — CyberSecurity Challenges and CATS, and (3) the evaluation, reports on our experiences in empowering industrial software developers in secure coding. We claim that our approach with secure coding guidelines, a secure coding policy, and two serious games have three effects. Firstly, it raises cybersecurity awareness. Secondly, it raises the security level of products and services deployed to critical infrastructures and contributes to the low-risk appetite of the company. Finally, it empowers the software developers and contributes to their productivity. Our evaluation contributes to the understanding of training measures under various workplace conditions and on how to empower workers in a hybrid workplace setting to address cybersecurity - a topic important to the company and society as a whole. Training akin to workplace conditions facilitates skill transfer from the games to actual work. The serious game contributes to a positive workplace atmosphere and contributes to productivity in coding. Distinguishing to our approach is that we work in the context of an international company with industrial software developers.

Research in the context of an international company and for industrial software engineers is a particular challenge. We claim to follow guidelines in case study research, design science, action design research, and qualitative content analysis for our findings' (internal) validity. To be aligned with business goals, mutual agreement, and management endorsement were crucial for the success of our design activities in the organization. The team of researchers has extensive expertise in industrial software engineering, which facilitates authentic feedback on game designs. Both the researchers expertise and the professional skills of software developers on giving feedback on design contributed to an early closure and the necessary stability for significant evaluation.

Software developers providing feedback and expertise among the researchers facilitated a comprehensive understanding of the problem, early closure on the design of the games, and authentic feedback in the evaluations.

While the evaluations indicate a positive gaming experience and the management endorsement and mutual agreement of all stakeholders indicate that the serious games have the desired effect, it is necessary to reflect on limitations and threats to validity.

First, we claim that the problem analyzed in the case study is not uncommon. Following our approach of defining guidelines, getting them adopted as a policy, and using serious games to raise awareness is a viable solution to the problem also for other organizations in the cyber–physical systems business.

Second, we do not quantify the actual effect of the games as an intervention. This is inherent to the methods used. We have a research design with intervention but without a non-intervened control group to assess the impact of the serious games and the impact under different workplace conditions. In this very industrial context, experimental research with groups and control groups is hardly feasible — one has to respect the workload of the software developers and the company's resources. To assess the effect of our intervention, we rely on surveys and feedback in focus groups and on mutual agreement among the stakeholders and argue that this is authentic and valid. Moreover, measuring the effect, e.g., through code analysis, is theoretically feasible; in practice, the company has a high level of tool support in the software lifecycle with elaborating quality controls for code. With the games, we address the weaknesses that are typically not found by code quality measures in place. Also, employees' representatives and compliance with workplace regulations are barriers to assessing employees' performance in the workplace. So we argue again that surveys and focus groups are authentic methods of evaluation, whereas code reviews or experimental settings would not work in this research setting. Also, we argue that we face a wicked problem — according to Hevner's design science paradigm. For a wicked problem, human cognitive and social limitations have an impact on the perception of the usefulness of the design for a given problem. In the case of serious games, the effect of the games depends on the individual capabilities of the software developers as well as on the organization of the software lifecycle. Measuring the effect quantitatively is, in this case, a tedious topic. This is inherent in cybersecurity, software engineering, and the methods applied in this study.

Third, we need to argue whether the increase in the level of awareness is an effect of our intervention or due to the Hawthorne effect (McCarney et al., 2007), i.e., the attention and presence of the researchers. The researchers with experience in industrial software engineering and, later, members of the organization delivered the games. We argue that this yields a low power distance, which is an argument against the Hawthorne effect. We have no evidence that the effect is not due to the game but to some other phenomenon in the context.

Forth, we discuss the quality of our instruments. Concerning reliability of our research instruments, we argue that consistency of findings across time, across variations of the serious game formats, and independent of the trainer has been demonstrated. We utilize the awareness concept, an established model for evaluation with a grounding in literature, and the specific setting of Action Design Research with the experience of the researchers ensures the internal validity of constructs. Throughout and in triangulation of experiments, settings,

and instruments, we have no evidence to challenge the internal validity of constructs.

The strength of our design and empirical work lies in the authentic setting and the different data collection methods that contribute to the evaluation. In such a professional setting, mutual agreement, obtaining management support, and becoming part of the company's continuing education program strongly indicate that the serious games are valuable and contribute to the organizational strategy for excellence in cybersecurity.

11. Conclusions

Software development is a fundamental activity of software engineering, and the development of secure software is gaining more and more attention. The need to develop secure software is rooted in the evolving threat landscape, which has seen several serious cybersecurity incidents with serious consequences. This need is also motivated by existing security standards that aim to raise the level of cybersecurity of existing software. There are several methods to enhance software cybersecurity, e.g., static code analysis (SCA) or code reviews. However, SCA tools could generate false-positive and false-negative results. Methods like SCA tools along are not adequate for achieving security requirements. The developers and practitioners in the industry need to raise awareness about cybersecurity and exercise their know-hows on interpreting the findings from SCA tools. Our work focuses on serious games to raise cybersecurity awareness among software developers in the industry. As such, we give an overview and practical advice on developing serious games for the industrial context.

Our work starts with an industry case study regarding establishing secure coding guidelines and policies in a company. As a result, we found a sense of urgency to raise awareness about cybersecurity among practitioners in the industry but also highlighted the complexity of the topic. We designed and developed a serious game, the CyberSecurity Challenges (CSC), to raise awareness among software developers of secure coding guidelines. We have thoroughly and systematically evaluated the game in the industry, and our results clearly show its usefulness. Our validation of the game is based on collecting participant feedback and the iterative design based on this feedback. Since CSC has successfully raised awareness about secure coding guidelines, we followed a similar design science research approach and continued with the design of the second game to raise awareness about cloud security. Towards this goal, we have designed the Cloud of Assets and Threats (CATS) game, which can reuse the existing infrastructure of CSC and be deployed within the CSC framework as a category of challenges or deployed as a stand-alone online board game.

We find it beneficial from the trainers, participants, and organization's perspectives to involve different types of serious games in training under the background of hybrid work. The conclusion is supported by feedback from experienced cybersecurity trainers and the data we collected in our research. Applying serious games on such occasions could maximize the advantage of online training and compensate for the disadvantages.

This experience report provides important insight into the design, implementation, and evaluation of two instances of serious games. It could inspire further research in cybersecurity education and developer cybersecurity empowerment with serious games.

CRediT authorship contribution statement

Tiange Zhao: Conceptualization, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization. **Tiago Gasiba:** Conceptualization, Software, Validation, Writing – original draft, Writing – review & editing. **Ulrike Lechner:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Supervision. **Maria Pinto-Albuquerque:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

Acknowledgments

This research task was partially supported by Fundação para a Ciência e a Tecnologia, I.P. (FCT), Portugal [ISTAR Projects: UIDB/04466/2020 and UIDP/04466/2020]. Ulrike Lechner acknowledges funding by dtec.bw for project LIONS and dtec.bw is funded by the European Union — NextGenerationEU and for project CONTAIN by the Bundesministerium für Bildung und Forschung, Germany (FKZ 13N16581). Tiange Zhao and Tiago Gasiba acknowledge the funding provided by the Bundesministerium für Bildung und Forschung (BMBF), Germany for the project CONTAIN with the number 13N16585.

The authors would also like to thank Ece Ata, Kristian Beckers, Luís Afonso Casqueiro, Jorge Cuellar, Tilman Dewes, Thomas Diefenbach, Holger Dreger, Samra Hodzic, Andrei-Cristian Iosif, Akram Louati, Daniel Mendez, Didem Oengue, Kaan Oguzhan, Anmoal Porwal, Filip Rezabek, Santiago Suppan, Thomas Wakim and Alae Zouitni for valuable discussions, contributions to game designs and studies. We thank all game participants and trainers for their support and insights. We thank the anonymous reviewers and the editors for their helpful and fruitful comments and guidance in the review process. We are indebted to all players, study participants who took part in the game events, and empirical studies during this research endeavor.

References

Agba, M.S., Agba, A.O., Chukwurah, Jr., D.C., 2021. COVID-19 pandemic and workplace adjustments/decentralization: A focus on teleworking in the new normal. BRAIN. Broad Res. Artif. Intell. Neurosci. 11 (4), 185–200.

Anon, 2007, ISA/IEC 62443 Series of Standards, https://www.isa.org/standards-and-publications/isa-standards/isa-iec-62443-series-of-standards.

Anon, 2018, ANSI/ISA-62443–4–1–2018, Security for industrial automation and control systems - Part 4-1: Secure product development lifecycle requirementshttps://www.isa.org/products/ansi-isa-62443-4-1-2018-security-for-industrial-au.

Anon, 2018, ANSI/ISA-62443-4-2-2018, Security for industrial automation and control systems, Part 4-2: Technical security requirements for IACS components, https://www.isa.org/products/ansi-isa-62443-4-2-2018-security-for-industrial-au.

Anon, 2020a. Cloud Security Alliance, Shared Responsibility Model Explained. https://cloudsecurityalliance.org/blog/2020/08/26/shared-responsibility-model-explained/.

Anon, 2020b. MITRE ATT&CK cloud matrix. https://attack.mitre.org/versions/v8/matrices/enterprise/cloud/.

Anon, 2021a. Amazon web service. https://aws.amazon.com/.

Anon, 2021b. Terraform by HashiCorp. https://www.terraform.io/.

Banjo, S., Yap, L., Murphy, C., Chan, V., 2020, The World's Biggest Work-From-Home Experiment, https://www.bloomberg.com/news/articles/2020-02-02/coronavirus-forces-world-s-largest-work-from-home-experiment.

Barela, J., Gasiba, T., Suppan, S., Berges, M., Beckers, K., 2019. When interactive graphic storytelling fails. In: 2019 IEEE 27th International Requirements Engineering Conference Workshops. REW, pp. 164–169. http://dx.doi.org/10.1109/REW.2019.00034.

Borges, F.C.L., 2022. Employee engagement in virtual teams: The role of gamification. NOVA Information Management School (NIMS).

BSI, 2020. BSI IT-Grundschutz-Kompendium. Tech. rep, Bundesamt für Sicherheit in der Informationstechnik, Reguvis Fachmedien GmbH, Köln, Germany, pp. 1–816, ISBN: 978-3-8462-0906-6, URL https://tinyurl.com/BSI-Grundschutz-Kompendium.

Bulgurcu, B., Cavusoglu, H., Benbasat, I., 2010. Information Security Policy Compliance: an Empirical Study of Rationality-Based Beliefs and Information Security Awareness. MIS Q. 34 (3), 523–548. http://dx.doi.org/10.2307/25750690, URL https://misq.org/misq/downloads/download/article/872/.

Canalys, 2021. China's cloud spend up 45% in 2021 bringing high expectations for 2022. https://canalys.com/newsroom/china-cloud-market-q4-2021.

Carnegie Mellon University, 2020, Secure Coding Standards, Software Engineering Institute, https://wiki.sei.cmu.edu/confluence/display/seccode.

Carnegie Mellon University, 2023, INT32-C. Ensure that operations on signed integers do not result in overflow, Software Engineering Institute, https://tinyurl.com/46mz225n.

Cheung, R.S., Cohen, J.P., Lo, H.Z., Elia, F., Carrillo-Marquez, V., 2012. Effectiveness of cybersecurity competitions. In: The Steering Committee of The World Congress in Computer Science (Ed.), Proceedings of the International Conference on Security and Management. SAM, Las Vegas, USA, pp. 1–5.

Chukusol, C., Nilsook, P., Wannapiroon, P., 2022. Virtual board games platform. In: 2022 Research, Invention, and Innovation Congress: Innovative Electricals and Electronics. RI2C, pp. 273–277. http://dx.doi.org/10.1109/RI2C56397.2022.9910289.

Cloud Security Alliance (CSA), 2019. Top threats to cloud computing: The egregious 11. BLACKHAT2019.

codewars, 2023, Achieve mastery through challenge - Improve your development skills by training with your peers on code kata that continuously challenge and push your coding practice, https://www.codewars.com/.

Connory, M., 2019. Software companies keep making these same cyber security mistakes. https://isuggi.com/software-companies-keep-making-these-same-cyber-security-mistakes.

Corbin, J., Strauss, A., 2014. Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory. SAGE Publications, URL https://books.google.de/books?id=hZ6kBQAAQBAJ.

Crosley, J., 2020. Qualitative data coding 101. https://gradcoach.com/qualitative-data-coding-101/.

Cullinane, I., Huang, C., Sharkey, T., Moussavi, S., 2015. Cyber security education through gaming cybersecurity games can be interactive, fun, educational and engaging. In: J. Comput. Sci. Colleges. Evansville, IN, USA, pp. 75–81, 30.6 (June 2015).

Darling, E., 2018. Why SQL Developers Keep Making The Same Mistakes. https://www.brentozar.com/archive/2018/07/why-sql-developers-keep-making-the-same-mistakes.

De Smet, A., Dowling, B., Mysore, M., Reich, A., 2021, It's time for leaders to get real about hybridhttps://www.mckinsey.com/capabilities/people-and-organizational-performance/our-insights/its-time-for-leaders-to-get-real-about-hybrid.

Department of Homeland Security, US-CERT, 2023, Software Assurance, https://tinyurl.com/y6pr9v42.

Dörner, R., Göbel, S., Effelsberg, W., Wiemeyer, J., 2016. Serious Games: Foundations, Concepts and Practice. Springer.

Eisenhardt, K.M., 1989. Building theories from case study research. Acad. Manag. Rev. 14 (4), 532–550.

European Data Protection Supervisor (EDPS), 2018. Personal Data Breach. https://edps.europa.eu/data-protection/our-role-supervisor/personal-data-breach_en.

Ferro, L.S., Marrella, A., Catarci, T., Sapio, F., Parenti, A., De Santis, M., 2022. AWATO: A Serious Game to Improve Cybersecurity Awareness. In: Fang, X. (Ed.), HCI in Games. Springer International Publishing, Cham, pp. 508–529. http://dx.doi.org/10.1007/978-3-031-05637-6_33.

Frampton, S., 2020. Coding qualitative data: A beginner's how-to + examples. https://chattermill.com/blog/coding-qualitative-data#3-steps-for-coding-qualitative-data-from-the-top-down.

Galanti, T., Guidetti, G., Mazzei, E., Zappalà, S., Toscano, F., 2021. Work from home during the COVID-19 outbreak: The impact on employees' remote work productivity, engagement, and stress. J. Occup. Environ. Med. 63 (7), e426–e432. http://dx.doi.org/10.1097/JOM.0000000000002236.

Gasiba, T., 2020. Sifu Platform. Siemens AG, MIT License, https://github.com/saucec0de/sifu.

Gasiba, T., 2021. Raising Awareness on Secure Coding in the Industry through CyberSecurity Challenges (Ph.D. thesis). Universität der Bundeswehr München, URL: https://athene-forschung.unibw.de/85257?show_id=140142.

Gasiba, T., Lechner, U., Pinto-Albuquerque, M., 2020a. Sifu - A CyberSecurity Awareness Platform with Challenge Assessment and Intelligent Coach. In: Cybersecurity Journal, Special Issue on Cyber-Physical System Security. SpringerOpen, pp. 1–23. http://dx.doi.org/10.1186/s42400-020-00064-4.

Gasiba, T., Lechner, U., Pinto-Albuquerque, M., 2021a. CyberSecurity challenges for software developer awareness training in industrial environments. In: Ahlemann, F., Schütte, R., Stieglitz, S. (Eds.), Innovation Through Information Systems. In: Lecture Notes in Information Systems and Organisation, Springer International Publishing, Cham, pp. 370–387.

Gasiba, T., Lechner, U., Pinto-Albuquerque, M., 2021b. CyberSecurity challenges: Serious games for awareness training in industrial environments. pp. 1–15. http://dx.doi.org/10.48550/arXiv.2102.10432, Federal Office for Information Security (Ed.): Germany. Digital. Secure. 30 Years BSI - Proceedings of the 17th German IT Security Congress 2021.

Gasiba, T., Lechner, U., Pinto-Albuquerque, M., Mendez, D., 2021c. Is Secure Coding Education in the Industry Needed? An Investigation Through a Large Scale Survey. In: Erdogmus, H., Moreno, A.M. (Eds.), 43rd International Conference on Software Engineering. pp. 1–12, URL https://arxiv.org/abs/2102.05343.

Gasiba, T., Lechner, U., Pinto-Albuquerque, M., Porwal, A., 2020b. Cybersecurity Awareness Platform with Virtual Coach and Automated Challenge Assessment. In: 6th Workshop on the Security of Industrial Control Systems & of Cyber-Physical Systems. CyberICPS, Springer, Cham, Online, pp. 67–83, https://doi.org/10.48550/arXiv.2102.10430.

Gasiba, T., Lechner, U., Pinto-Albuquerque, M., Zouitni, A., 2020c. Design of Secure Coding Challenges for Cybersecurity Education in the Industry. In: 13th International Conference on the Quality of Information and Communications Technology. Springer, pp. 223–237, Online.

Godsay, M., 2015. Article: The process of sentiment analysis: A study. Int. J. Comput. Appl. 126 (7), 26–30, Published by Foundation of Computer Science (FCS), NY, USA.

Graziotin, D., Fagerholm, F., Wang, X., Abrahamsson, P., 2018. What happens when software developers are (un)happy. J. Syst. Softw. 140, 32–47. http://dx.doi.org/10.1016/j.jss.2018.02.041.

Graziotin, D., Wang, X., Abrahamsson, P., 2015. Do feelings matter? On the correlation of affects and the self-assessed productivity in software engineering. J. Softw. Evol. Process 27 (7), 467–487. http://dx.doi.org/10.1002/smr.1673.

Hänsch, N., Benenson, Z., 2014. Specifying IT security awareness. In: 25th International Workshop on Database and Expert Systems Applications. IEEE, pp. 326–330. http://dx.doi.org/10.1109/DEXA.2014.71.

Hart, S., Margheri, A., Paci, F., Sassone, V., 2020. Riskio: A serious game for cyber security awareness and education. Comput. Secur. 95, 101827. http://dx.doi.org/10.1016/j.cose.2020.101827.

Hashim, M., Ashmel, M., Tlemsani, I., Matthews, R., 2022. Higher education strategy in digital transformation. Educ. Inf. Technol. 27 (3), 3171–3195. http://dx.doi.org/10.1007/s10639-021-10739-1.

Hevner, A., 2007. A three cycle view of design science research. Scand. J. Inf. Syst. 19, 1–6, URL http://aisel.aisnet.org/sjis/vol19/iss2/4.

Hevner, A., March, S., Park, J., 2004. Design science in information systems research. MIS Q. 28 (1), 75–105. http://dx.doi.org/10.2307/25148625.

HITB CyberWeek, 2020, Third edition of Adversaries Vs Defenders ctf Competition - Nov 18, 19 Welcoming Red Teams and Blue Teams Upcoming village and CTF at HITB CyberWeek, https://redteamvillage.org/HITB-CyberWeek-2020-Red-vs-Blue-CTF/.

ISO27001, 2017. ISO/IEC 27001 Information Security Management. https://www.iso.org/isoiec-27001-information-security.html.

Jacoby, J., Matell, M.S., 1971. Three-point Likert scales are good enough. J. Mar. Res. 8 (4), 495–500. http://dx.doi.org/10.1177/002224377100800414, SAGE Publications Sage CA: Los Angeles, CA.

Markopoulos, A.P., Fragkou, A., Kasidiaris, P.D., Davim, J.P., 2015. Gamification in engineering education and professional training. Int. J. Mech. Eng. Edu. 43 (2), 118–131. http://dx.doi.org/10.1177/0306419015591324.

McCarney, R., Warner, J., Iliffe, S., van Haselen, R., Griffin, M., Fisher, P., 2007. The hawthorne effect: a randomised, controlled trial. BMC Med. Res. Methodol. 7 (1), 30.

Mirkovic, J., Peterson, P., 2014. Class capture-the-flag exercises. In: USENIX Association (Ed.), 2014 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 14). San Diego, CA, USA, pp. 1–8.

MITRE Corporation, 2006, CWE-190 – Integer Overflow or Wraparound, https://cwe.mitre.org/data/definitions/190.html.

Monasor, M.J., Vizcaíno, A., Piattini, M., Noll, J., Beecham, S., 2014a. Assessment process for a simulation-based training environment in global software development. In: Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education. ITiCSE '14, Association for Computing Machinery, New York, NY, USA, pp. 231–236. http://dx.doi.org/10.1145/2591708.2591747.

Monasor, M.J., Vizcaino, A., Piattini, M., Noll, J., Beecham, S., 2014b. Evaluation of a simulation platform for interaction training: A multi-phased methodology. In: 2014 IEEE Frontiers in Education Conference (FIE) Proceedings. pp. 1–8. http://dx.doi.org/10.1109/FIE.2014.7044255.

Moody, G., Siponen, M., Pahnila, S., 2018. Toward a Unified Model of Information Security Policy Compliance. MIS Q. 42 (1), 285–311. http://dx.doi.org/10.25300/MISQ/2018/13853, URL https://misq.org/misq/downloads/download/article/1302/.

Mueller, A., 2021. Word-cloud. https://pypi.org/project/wordcloud/.

Nieto-Escamez, F.A., Roldán-Tapia, M.D., 2021. Gamification as online teaching strategy during COVID-19: A mini-review. Front. Psychol. 12, http://dx.doi.org/10.3389/fpsyg.2021.648552.

OWASP Foundation, 2001, Open Web Application Security Project, https://owasp.org/.

Patel, S., 2020. 2019 Global Developer Report: DevSecOps finds security roadblocks divide teams. https://about.gitlab.com/blog/2019/07/15/global-developer-report/.

Petri, G., von Wangenheim, C.G., Borgatto, A.F., 2016. MEEGA+: an evolution of a model for the evaluation of educational games. INCoD/GQS 3, 1–40.

Pillai, R.S., 2020. Covid 19- a booster for digital transformation!!. https://www.finextra.com/blogposting/18626/covid-19--a-booster-for-digital-transformation.

Poston, H., 2019. The Need For Secure Coding. https://securityboulevard.com/2019/11/the-need-for-secure-coding/.

Schneier, B., 2020. Software Developers and Security. https://www.schneier.com/blog/archives/2019/07/software_develo.html.

Secure Code Warrior, 2021. Whitepaper: Empowering developers to write secure code. Tech. rep., Secure Code Warrior, URL https://discover.securecodewarrior.com/Empowering-Developers.html.

Sein, M.K., Henfridsson, O., Purao, S., Rossi, M., Lindgren, R., 2011. Action design research. MIS Q. 35 (1), 37–56. http://dx.doi.org/10.2307/23043488.

Shostack, A., 2021. Tabletop security games & cards. https://https://shostack.org/games.html.

Siponen, M., Vance, A., 2010. Neutralization: New insights into the problem of employee information systems security policy violations. MIS Q. 34 (3), 487–502. http://dx.doi.org/10.2307/25750688.

Software Engineering Institute, Carnegie Mellon, 2018, SEI CERT C Coding Standardhttps://wiki.sei.cmu.edu/confluence/display/c/SEI+CERT+C+Coding+Standard.

Software Engineering Institute, Carnegie Mellon, 2023, SEI CERT C++ Coding Standard, https://wiki.sei.cmu.edu/confluence/pages/viewpage.action?pageId=88046682.

Subhash, S., Cudney, E.A., 2018. Gamified learning in higher education: A systematic review of the literature. Comput. Hum. Behav. 87, 192–206. http://dx.doi.org/10.1016/j.chb.2018.05.028, URL https://www.sciencedirect.com/science/article/pii/S0747563218302541.

Švábenskỳ, V., Vykopal, J., Cermak, M., Laštovička, M., 2018. Enhancing cybersecurity skills by creating serious games. In: Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education. pp. 194–199. http://dx.doi.org/10.48550/arXiv.1804.03567.

Thompson, M., Irvine, C., 2011. Active learning with the cyberciege video game. In: Proceedings of the 4th Conference on Cyber Security Experimentation and Test. CSET '11, USENIX Association, USA, p. 10.

Travers, M., Richardson, I., Higgins, L., 2022. Challenges and opportunities when deploying a gender STEM intervention during a pandemic. In: 2022 IEEE/ACM 3rd International Workshop on Gender Equality, Diversity and Inclusion in Software Engineering. GEICSE, pp. 59–66. http://dx.doi.org/10.1145/3524501.3527596.

Vaughan-Nichols, S., 2019. No Love Lost Between Security Specialists and Developers. https://www.zdnet.com/article/no-love-lost-between-security-specialists-and-developers/.

Vizcaíno, A., García, F., Guzmán, I.G.R.D., Moraga, M.Á., 2019. Evaluating GSD-aware: A serious game for discovering global software development challenges. ACM Trans. Comput. Educ. 19 (2), http://dx.doi.org/10.1145/3218279.

Wölfe, R., Schubert, P., 2009. Dauerhafter Erfolg Mit Business Software: 10 Jahre Fallstudien Nach Der EXperience Methodik. Carl Hanser Verlag, Germany.

Zhao, T., Gasiba, T., Lechner, U., Pinto-Albuquerque, M., 2021. Exploring a Board Game to Improve Cloud Security Training in Industry. In: Henriques, P.R., Portela, F., Queirós, R., Simões, A. (Eds.), Second International Computer Programming Education Conference. ICPEC 2021, In: Open Access Series in Informatics (OASIcs), vol. 91, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, pp. 11:1–11:8. http://dx.doi.org/10.4230/OASIcs.ICPEC.2021.11.

Zhao, T., Gasiba, T., Lechner, U., Pinto-Albuquerque, M., 2021b. Raising awareness about cloud security in industry through a board game. Inf. Special Issue Future Trends Comput. Program. Edu. 12 (11), http://dx.doi.org/10.3390/info12110482.

Zhao, T., Lechner, U., Pinto-Albuquerque, M., Ata, E., 2022. Cloud of Assets and Threats: A Playful Method to Raise Awareness for Cloud Security in Industry. In: Simões, A., Silva, J.a.C. (Eds.), Third International Computer Programming Education Conference. ICPEC 2022, In: Open Access Series in Informatics (OASIcs), vol. 102, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, pp. 6:1–6:13. http://dx.doi.org/10.4230/OASIcs.ICPEC.2022.6.

Zhao, T., Lechner, U., Pinto-Albuquerque, M., Ata, E., Gasiba, T., 2023. CATS: A serious game in industry towards stronger cloud security. In: Wang, G., Choo, K.-K.R., Wu, J., Damiani, E. (Eds.), Ubiquitous Security. Springer Nature Singapore, Singapore, pp. 64–82.

Tiange Zhao was born in Beijing, China. She received her M.Sc. degree in School of Computation, Information and Technology from Technical University of Munich (TUM) Germany in 2017 and her Bachelor degree in School of Software Engineering at Beijing University of Technology in 2014. She is currently working toward her Ph.D. under the supervision of Prof. Lechner and Prof. Pinto-Albuquerque. She works at Siemens AG since 2017 as a security consultant and deliver training to internal customers. Her current research interest includes developing and evaluating serious games in industry to help raising awareness about cloud security.

Dr. Tiago Gasiba was born in Oporto, Portugal. He did his Ph.D. at the Universität der Bundeswehr München in 2021 on the topic of secure coding awareness. He received his M. Sc. degree in telecommunication engineering from the Technical University of Munich (TUM) Germany in 2004, and his Eng. degree in electrical engineering and computer science from the Faculdade de Engenharia da Universidade do Porto in 2002. He is currently working for Siemens AG as the senior key expert for developer enablement. His current research interest includes secure coding guideline in industry and designing serious games for raising awareness in cybersecurity.

Prof. Dr. Ulrike Lechner holds the Chair of Information Systems at the Universität der Bundeswehr München. She studied Computer Science at the Universität Passau and did her Ph.D. at the Universität Passau. She held positions as professor at the Universität St. Gallen and the Universität Bremen. Her research and teaching interests are IT-Security for Critical Infrastructures, Enterprise Architectures and Digital Business Models.

Prof. Dr. Maria Pinto-Albuquerque is an Assistant Professor at Iscte - Instituto Universitário de Lisboa and a researcher at Istar-Iscte.

Her research work focuses on the relationship of the person, as user or creator, with the computational system. Her research on the person–system relationship has been developed in the topics of cybersecurity awareness, security and usability alignment, and requirements engineering.

She has developed tools and techniques, such as serious games and techniques that use creativity, to promote the efficient, responsible and safe use and development of computer systems, both by users, system engineers and all kinds of stakeholders (co-creators of these systems).