Use and Misuse of Case Studies and Experiments in Game Software Engineering

Carlos Pérez[a,b], Isis Roca[a,b], África Domingo[a], Jorge Echeverría[a], Carlos Cetina[b], Lorena Arcega[a], Jaime Font[a]

[a] *Universidad San Jorge. SVIT Research Group, Zaragoza, Spain*
[b] *Universitat Politècnica de València. PROS Research Center, Valencia, Spain*

Abstract

Context: The growing complexity of the video game industry has led to the emergence of Game Software Engineering (GSE), a sub-discipline concerned with the development and maintenance of video game software. Unlike traditional Software Engineering, GSE is inherently interdisciplinary and places strong emphasis on delivering enjoyable user experiences. While Empirical Software Engineering has achieved increasing methodological rigor, empirical research practices within GSE remain relatively underdeveloped.

Objective: This study analyzes the use and methodological accuracy of the terms *case study* and *experiment* in GSE literature, assessing whether published studies correctly apply established empirical research definitions and criteria.

Methods: A systematic literature review identified 48 primary studies in the GSE domain. Each study was examined to determine whether its classification as a case study or experiment complied with widely accepted methodological standards. The evaluation focused on adherence to defining characteristics and essential criteria of each research strategy.

Results: The findings reveal substantial misclassification of empirical research strategies in GSE publications. Only 1 out of 37 (3%) of the studies labeled as case studies met the established criteria, whereas 36 out of 37 (97%) were incorrectly classified. Similarly, only 2 out of 11 (18%) of the studies identified as experiments satisfied all defined experimental requirements. These proportions are considerably lower than those typically observed in traditional Software Engineering research, indicating weaker methodological alignment within GSE.

Conclusion: The results highlight a significant need to strengthen methodological rigor in GSE empirical research. Beyond identifying misapplications, this study provides illustrative examples and practical recommendations to support more accurate use of empirical research strategies. Improving methodological clarity is essential.

Keywords: Empirical game software engineering, Case study, Controlled experiment, Research methodology.

1. Introduction

The video game industry is a significant and profitable sector within the software realm. The development of video games has become increasingly intricate, encompassing the creation of sophisticated software covering aspects such as game mechanics, graphics, and physics. In response to this complexity, Game Software Engineering (GSE) [1] has emerged [1] and been consolidated [2], focusing on developing and maintaining software for video games.

Video games, which are considered both advanced software products and complex works of creativity and art, have led to notable differences between traditional Software Engineering (SE) and GSE [3, 1, 4, 5]. These differences arise from the fusion of disciplines and the necessity for the product to be enjoyable.

Empirical SE studies have gained popularity and increased in maturity and rigor [6]. The empirical SE community has made efforts to develop and refine empirical methods and instruments, including experiments, case studies, surveys, and systematic literature reviews, to support experimentation in SE. A key aspect of empirical research in Software Engineering is the correct distinction between research methods and data collection techniques. Methods such as case studies, experiments, and surveys define the overall research strategy, whereas interviews, questionnaires, workshops, or archival data are instruments used within those methods. For instance, a survey is distinct from the questionnaire used to implement it, and case studies always combine multiple data sources. This distinction, emphasized in the empirical Software Engineering literature [7, 8, 9], has been framed as a longstanding methodological concern, whose increasingly explicit discussion may indicate a gradual maturation of empirical research practices.

Empirical studies have also gained popularity in the GSE community. A recent systematic literature review (SLR) on GSE reveals that 37% of the papers use case studies or experiments. While one might assume that since GSE is a branch of Software Engineering, it would adhere to established software engineering research guidelines for conducting case studies and experiments. However there is a lack of evidence supporting this assumption. To address this gap, our work assesses the us-

Email addresses: cperez@usj.es (Carlos Pérez), iroca@usj.es (Isis Roca), adomingo@usj.es (África Domingo), joigpana@uv.es (Jorge Echeverría), ccetina@dsic.upv.es (Carlos Cetina), larcega@usj.es (Lorena Arcega), jfont@usj.es (Jaime Font)

[1] Note that the acronym GSE may also refer in other works to Global Software Engineering

age and application of the terms *case study* and *experiment*. To do this, we opted to focus on the papers identified through a SLR on GSE published in 2024 [2]. We performed an in-depth manual analysis of 48 publications, 37 of which were categorized as *case studies*, and 11 of which were categorized as *experiments*.

Our analysis reveals that few GSE research works use the terms *case study* and *experiment* properly, or rather less than in traditional SE. This work reveals this issue and provides specific examples and practical guidelines for future research in GSE. Specifically, we analyze the current state of GSE case studies and experiments, outline strengths and weaknesses, propose key points to tackle these weaknesses, showcase exemplary papers, and offer recommendations and insights.

On one hand, of the 37 case studies analyzed, only one met the criteria for being considered a *case study*. We reclassified four of these 37 articles (11%) as *action research*, five articles (14%) as *archival analysis*, and 27 articles (73%) as *small-scale evaluation*. The GSE community commonly uses real-world video games in its research, but they confuse the use of a real-world video game with a case study. To help studies fulfill the criteria for being properly classified case studies, our work pinpoints key points to be addressed by *action research* (no active research role), *archival analysis* (contemporary phenomenon), and *small-scale evaluation* (not out of context). The study of Engstrom et al. [S1] can be used as a reference for inspiration on how to conduct a case study successfully in GSE.

On the other hand, of the 11 experiments analyzed, only two met all of the criteria for being an experiment (18%). For experiments, the GSE community is strong in manipulation and causality but especially weak in providing replication facilities and setting control variables. Therefore, to enhance experiments in the GSE community, it is crucial to focus on key aspects such as the replication package (including instructions, operational materials, support, and structural and usability enhancements) and minimizing bias from confounding variables. The two studies carried out by Domingo et al. [S2], [S3] achieved the highest level of compliance to date that an experiment should have. These works can inspire the community to make their experiments better.

Our work also provides insights to help follow proper guidelines. For example, the widespread playtesting techniques should not be confused with case studies or the importance of properly referencing the type of work that is carried out in the research study.

Efforts of the GSE community in empirical evaluation are growing, especially for case studies and experiments. The community has great access to real-world video games but needs improvement doing case studies and experiments correctly. Our work like this helps to correct that weakness and strengthens the empirical evaluations of the GSE community.

The rest of the paper is organized as follows: Section 2 provides background to contextualize the role of software in video game development. Section 3 summarizes the related work on the use and misuse of the terms *case study* and *experiment*. Section 4 details the research strategy and methods followed in this work. Section 5 provides an assessment of the use of the term *case study* in GSE. Section 6 provides an assessment of the use of the term *experiment* in GSE. Section 7 discusses the results. Section 8 discusses the threats to validity of this work. And, finally, Section 9 summarizes our conclusions.

2. Background

The video game industry has undergone relentless growth [10]. In 2019, the video game development industry generated more revenue than the revenue of the film and music industries combined [11]. In 2021, the video game industry generated approximately $180.3 billion [12], while, in 2022, it reached $184.4 billion [13].

Video games are intricate amalgamations where art and software intricately intertwine during development to generate the final product. Consequently, development teams require various roles, with software developers comprising the majority (24%), alongside game designers (23%), artists (15%), user interface designers (8%), and quality assurance engineers (5%) [14]. In this development context, software permeates every facet of video game development, dictating all of the elements and actions within the game. For example, it governs the logic behind non-playable character (NPC) actions, using software artifacts such as state machines or decision trees. In the advancement of video games, they evolve into increasingly complex products, thereby escalating the complexity and significance of the software they entail.

Nowadays, to decrease the difficulty in video game development, the use of game engines has become popular. Two of the most commonly used game engines are Unreal [15] and Unity [16]. Game engines are characterized by their ability to integrate graphical aspects with the physical laws throughout the video game. One can argue that game engines are software frameworks [11]. Ultimately, game engines significantly accelerate the development of video games.

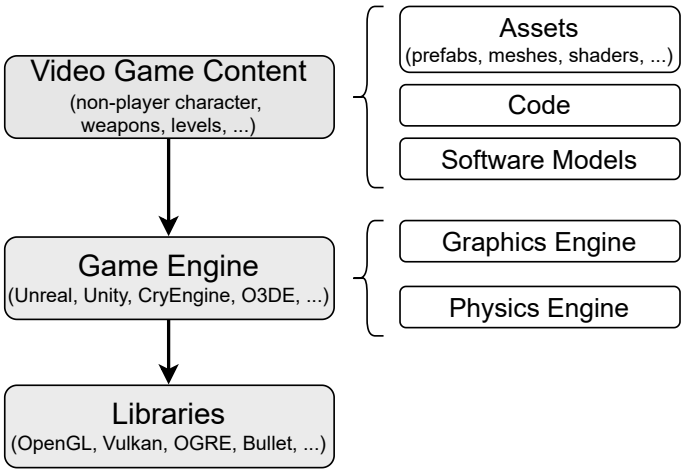

Figure 1: Artifacts of Video Game Development.

Fig. 1 shows the artifacts used in video game development. For instance, the Unity engine utilizes C# as its programming

language and proposes Unity Visual Scripting for developing video games using software models [17]. Similarly, the Unreal game engine uses C++ as its programming language and its proposal for developing video games using software models is Unreal Blueprints [18]. Using programming languages such as C++ or C# allows for more control over the content, and development with software models elevates the level of abstraction, placing greater emphasis on domain knowledge and reducing implementation and technological details.

Fig. 2 graphically represents the range of research work on GSE. The figure was generated using a react-wordcloud tool with the keywords from the GSE papers studied by the recent GSE SLR [2]. For instance, Fig. 2 includes terms related to software reuse (product line architecture, variability, and software product line engineering).

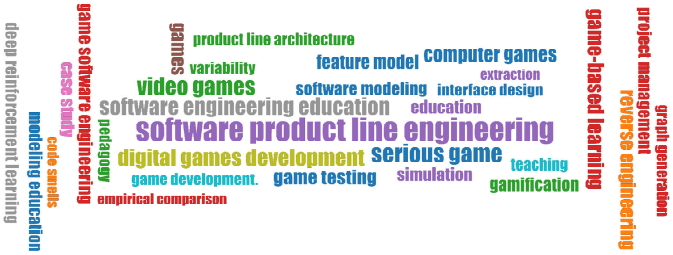

Figure 2: Keywords of GSE research work.

3. Related Work

This section summarizes the research efforts from two areas that are related to this work: the use and misuse of the term *case study* and the use and misuse of the term *experiment*.

3.1. Use and misuse of case study

Case study research is a widely utilized approach in various disciplines, and there are numerous general guidelines. Yin's book [19], first published in 1984, is probably the most well-known source in software engineering research. Case studies have gained prominence in software engineering, particularly for studying software development in real-world contexts. New and more detailed guidelines have subsequently been published in software engineering by Runeson and Höst [20], Verner et al. [21], Runeson et al. [22], etc,

Wohlin [23] conducts a review of the software engineering literature based on the perception that the term *case study* is assigned to studies that are not really case studies. The result is that the term is indeed not being used correctly. Then, Wohlin and Rainer [24] uncover the frequent misapplication of the term *case study* in software engineering research. They critically analyze articles referencing case study guidelines and presenting case studies. Their work reveals that only 50% of the studies examined accurately adhered to the criteria. To address this issue, they propose measures to ensure the proper labeling of *case studies*, including the development of a checklist and a self-assessment scheme. Similar findings were reported over 15 years ago by Zannier et al. [25]. From a study of articles published at the International Conference on Software Engineering, they conclude that, *"...our sample indicated a large misuse of the term case study"*.

Software engineering is not the only discipline in which this happens. For instance, Bergen et al. [26] discuss challenges in defining and applying the case study method in community nursing research.

3.2. Use and misuse of experiment

Many works lay the groundwork for how experiments should be conducted [27, 28, 29]. However, the term *experiment* is not always used well; sometimes, it is used instead of another term, or, in other cases, because they do not apply the methodology correctly.

Ayala et al. [6] assess how the term *experiment* is employed in the context of mining software repositories (MSR) research. The authors examine the proper application of experimental methodologies and identify instances where the term is used incorrectly. They found that 19% of the papers claiming to be an experiment are indeed not an experiment at all. Their work provides valuable insights aimed at improving the rigor and clarity of experiments conducted in the MSR domain, underscoring the importance of accurate terminology for the advancement of robust scientific practices.

Other papers focus on analyzing only part of the experiment, such as Ludwig's work [30]. The study provides a comprehensive examination of the use and potential misuse of $p-values$ in both designed and observational studies. This work serves as a guide for researchers and reviewers, offering insights into the appropriate interpretation and application of $p-values$ in statistical analyses. The author discusses common pitfalls and misconceptions surrounding $p-values$, highlighting the importance of robust statistical practices.

In addition, in other disciplines, we can find works such as the one performed by Harris et al. [31]. This article focuses on the application and interpretation of quasi-experimental studies in the field of medical informatics. The authors delve into the specific challenges and considerations associated with quasi-experimental research designs, providing insights into their use in medical informatics contexts. The paper offers valuable guidance for researchers and practitioners in understanding and appropriately employing quasi-experimental methodologies.

Both the studies analyzing the use of the term *case study* and those analyzing the use of the term *experiment* aim to improve the quality of empirical research in their respective areas by highlighting the most frequent and most common errors in empirical research and providing guidelines and examples of how to do it correctly.

The classification of empirical research methods in Software Engineering is not uniform across the literature. Alternative perspectives include taxonomies rooted in social science traditions [7], inductive classifications based on author labeling [8]), and consolidated overviews of contemporary methods [9]. In this work, we adopt a criteria-based classification aligned with established guidelines to ensure consistency, while acknowledging that alternative approaches exist.

However, the papers mentioned above have not addressed studies of game software engineering. This is especially important because, in recent years, the GSE community has almost quadrupled its use of case studies, and experiments are one of the main empirical methods used by the GSE community [2]. The GSE community relies on case studies and experiments but is not aware of their maturity and rigor, which is that we address in this paper.

4. Research Method

The initial motivation for undertaking this work is to understand how empirical studies are conducted in the context of Game Software Engineering (GSE) [3]. More specifically, the focus is on gaining insights into how case studies and experiments have been performed. We have conducted a flexible exploratory research approach to answer these two research questions.

- **RQ1: Is the term *case study* properly used in GSE research?**

  To assess the use of the term *case study* in GSE research, we perform the procedure explained in Section 4.4. The answer to this research question has been detailed in Section 5.

- **RQ2: Is the term *experiment* properly used in GSE research?**

  To evaluate the application of the term *experiment* in GSE research, we perform the procedure outlined in Section 4.4. The response to this research question is detailed in Section 6.

In addition to giving us the situation in which the use of each of the terms is found, these research questions have the potential to identify representative examples of how the study should be carried out, which will also help to put empirical GSE into perspective with regard to empirical traditional SE.

The following sections describe the research methodology undertaken. They describe how the papers with case studies and experiments were selected for our research, how checklists were designed for analyzing these studies, and, finally, how they were classified. Fig. 3 graphically shows this process.

4.1. Identifying a dataset of articles

Our goal is to understand how case studies and experiments are conducted in the GSE context. For this reason, to perform this study, we considered working with the papers selected in a systematic literature review of GSE published in 2024 [2]. This systematic literature review was performed following the best practices as suggested by Kitchenhan and Charters [32].Studies were included if they focused on software engineering for industry-scale computer games and excluded if they were non-primary, non-peer-reviewed, not in English, outside the 2009–2021 range, or unrelated to SE (e.g., serious games, AI, or content creation). The selection process reduced an initial set of 3,491 records to 98 final studies through successive filtering (duplicates, title/abstract screening, and full-

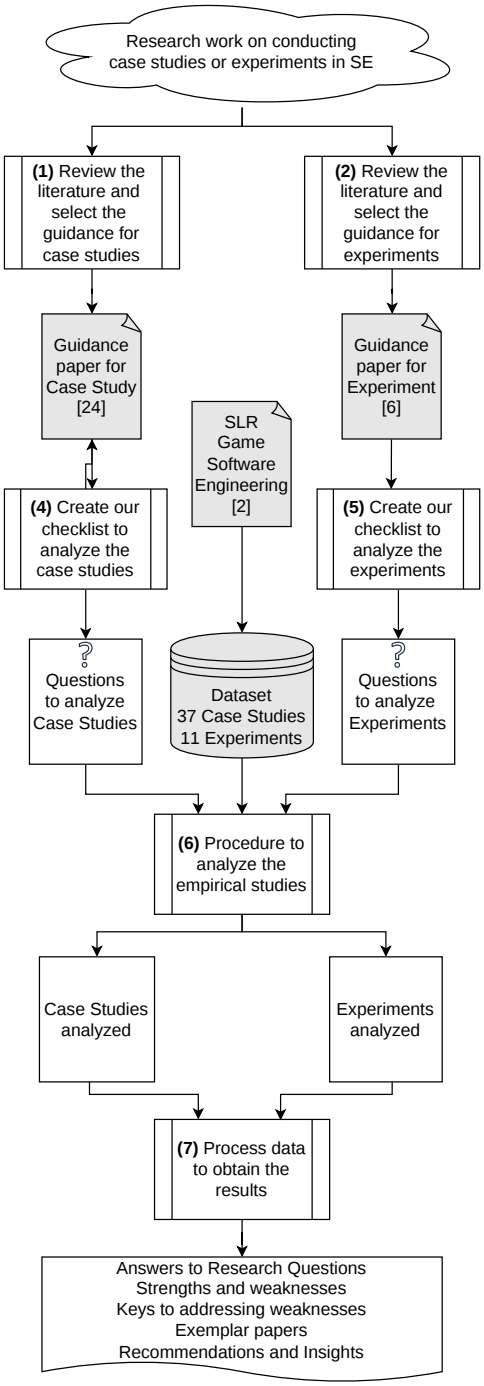

Figure 3: Process overview.

text review), complemented by snowballing, portraying the current state of software engineering for industry-scale computer games.

To generate our dataset of papers, the inclusion criterion was to select articles categorized in [2] as case studies or experiments (this step is depicted as (3) in Fig. 3). As a result of the inclusion criterion, we had 37 papers categorized as case studies and 11 papers categorized as experiments, as shown in Table 1.

All the studies analyzed in the SLR refer to commercial video games. No specific type of video game is highlighted; the papers include, for instance, action-adventure, first-person shooter, real-time strategy, multiplayer online battle arena, or 3D shooter.

4.2. How to Conduct Case Studies?

We reviewed the scientific literature to find studies that analyzed case studies in the software engineering area. Then, we selected the most recent work from the set of reviewed works. This step is depicted as (1) in Fig. 3; the selected paper was written by Wohlin and Rainer [24]. This work includes Wohlin's proposed definition of a case study. This definition is: *"A case study is an empirical investigation of a case, using multiple data collection methods, to study a contemporary phenomenon in its real-life context, and with the investigator(s) not taking an active role in the case investigated"*[23]. While this definition is widely accepted in Software Engineering, it is not shared across all disciplines. For example, some IS/IT approaches allow participatory or retrospective case studies that relax certain assumptions. In this work, we adopt Wohlin's definition as an operational reference for consistent classification, acknowledging that it reflects a specific methodological stance.

Taking the previous case study definition as a reference and the checklist proposed in [24] to check if a case study has been designed according to Wohlin's definition, we have established a set of questions as a checklist to analyze the empirical studies classified as case studies. This step is depicted as (4) in Fig. 3. The checklist that we have designed comprises the following:

- Five questions proposed in [24] to classify case studies in the software engineering context (QCS1-QCS5).
- A question to identify if the smell indicator for case studies proposed in [24] works in the GSE context (QCS6).
- A question to classify the empirical study if it is not a case study (QCS7).

The checklist we have used is the following:

- **QCS1: Will/Did the empirical study investigate an identifiable case or case(s)?** The aim of this question is to ensure that the case study should allow for the identification of what is under study and should not be confused with the context in which it takes place.
- **QCS2: Will/Was the study (be) conducted in a real-life context?** The aim of this question is to ensure that the case study unfolds in a real-world context and should

Table 1: Studies from systematic literature review of GSE categorized as a case study or experiment.

Category	Year	Studies	Total
Case Study	2009	Drachen et al. [S4]	37
	2009	Papaloukas et al. [S5]	
	2010	Vickers et al. [S6]	
	2011	Hullett et al. [S7]	
	2011	Kienzle et al. [S8]	
	2012	Hullett et al. [S9]	
	2012	Salazar et al. [S10]	
	2012	Donkervliet et al. [S11]	
	2013	Best et al. [S12]	
	2013	Ramadan et al. [S13]	
	2013	Suznjevic et al. [S14]	
	2015	Iftikhar et al. [S15]	
	2015	Ollsson et al. [S16]	
	2016	Chen et al. [S17]	
	2016	Mirza-Babaei et al. [S18]	
	2016	Kica et al. [S19]	
	2016	Guardiola et al. [S20]	
	2017	He et al. [S21]	
	2017	Sagi et al. [S22]	
	2017	Festa et al. [S23]	
	2017	Kohwalter et al. [S24]	
	2018	Engstrom et al. [S1]	
	2018	Nunnari et al. [S25]	
	2019	Seitz et al. [S26]	
	2019	Morisaki et al. [S27]	
	2019	Lynch et al. [S28]	
	2019	Zeleke et al. [S29]	
	2019	El-Nasr et al. [S30]	
	2019	Geisler et al. [S31]	
	2020	Mar'in-Lora et al. [S32]	
	2020	Yacoub et al. [S33]	
	2020	Mar'in-Lora et al. [S34]	
	2020	Mirza-Babaei et al. [S35]	
	2021	Bryant et al. [S36]	
	2021	Tong et al. [S37]	
	2021	Geisler et al. [S38]	
	2022	Mar'in-Lora et al. [S39]	
Experiment	2009	Ding et al. [S40]	11
	2010	Instance et al. [S41]	
	2011	Gonzalez et al. [S42]	
	2012	Jorgensen et al. [S43]	
	2015	Prado et al. [S44]	
	2018	Kenzel et al. [S45]	
	2019	MacCormick et al. [S46]	
	2020	Domingo et al. [S2]	
	2020	Mizutani et al. [S47]	
	2021	Domingo et al. [S3]	
	2021	Zhong et al. [S48]	

not be conducted under laboratory conditions (controlled environment).

- **QCS3: Will/Did the study use more than a single method of data collection?** The aim of this question is to ensure that the case study uses various different methods of data collection to ensure triangulation.
- **QCS4: Will/Did the study investigate a contemporary phenomenon, i.e., a phenomenon occurring at the time of the study?** The aim of this question is to ensure that the case study investigates a contemporary phenomenon defined as a time frame encompassing relatively recent events during which it is feasible to gather additional information to complement any existing data.
- **QCS5: Will/Did the researchers take an active part in establishing a long-term change beyond the duration of the research?** The aim of this question is to distinguish between action research and a case study. In action research, the researchers act as change agents.
- **QCS6: How many times does the word 'interview' appears in the article?** The aim of this question is to determine the value of the smell indicator defined in [24] for detecting articles defined as case studies. The smell indicator is based on *"If the main body of the article reporting the case study has less than three occurrences of the word interview then the article is NOT reporting a case study"*.
- **QCS7: If the empirical study is not a case study, what should its classification be - *small-scale evaluation*, *interview study*, *archival analysis*, *action research*, *discourse analysis*, *survey*, *workshop*, or *focus group*?** The aim of this question is to reclassify the studies. According to [24], empirical studies incorrectly classified as case studies could be some of the study types stated in the question.

To consider an empirical study as a *case study*, questions from QCS1 to QCS4 should have affirmative answers (yes), and the answer to question QCS5 should be negative (no).

4.3. How to Conduct Experiments?

We conducted a review of scientific literature to identify works analyzing experiments in the field of software engineering. Among these works, we selected the most recent one, which had been authored by Ayala et al. [6] (this step is depicted as (2) in Fig. 3). This work includes Juristo and Moreno's definition of an experiment: *"an empirical procedure where key variables of a reality are manipulated to investigate the impact of such variations [...]. It is rooted in detecting quantifiable changes as a means of comparing one unitary experiment with another in search of the difference between them and, hence, the reason for the changes"* [29]. While this definition aligns with controlled experiments, other variants exist in Software Engineering, such as field and natural experiments, which relax some of these assumptions. In this work, we focus on controlled experiments, as they most closely reflect the type of studies identified in the analyzed dataset, while acknowledging that other forms of experimentation are also relevant in practice. Building upon the previous definition and the questions to assess the use of the term experiment in the work [6], we have established a set of questions as a checklist to analyze the 11 empirical studies classified as experiments. This step is depicted as (5) in Fig. 3.

The checklist we have employed is the following:

- **QEX1: Does the researcher intervene in the study design?** The aim of this question is to determine the manipulation in the study. A study without manipulation is an observational study (where the researcher does not intervene in the study design). On the other hand, a study with manipulation (interventional study) is where the researcher actively intervenes in the study design.
- **QEX2: Does the study design contain specific control mechanisms in order to avoid biased results due to confounding variables?** The aim of this question is to determine whether the study design has incorporated mechanisms for blocking or controlling variables. These mechanisms are relevant when these variables could confound the obtained results.
- **QEX3: Do the studies cover the random assignment of treatments to experimental groups?** The aim of this question is to ensure that the studies address the random assignment of treatments to experimental groups, as it is a crucial requirement for controlled experiments to mitigate bias and facilitate the deduction of causality [33].
- **QEX4: Does the study explicitly state a hypothesis to be tested, and does the study discuss limitations or threats to validity?** The aim of this question is to determine if the investigation into the focused experiment's scope involved examining the explicit hypotheses and discussing the limitations or threats to validity of the studies.
- **QEX5: Does the study use statistical analysis?** The aim of this question is to determine whether a statistical analysis was employed to assess the acceptance or rejection of a hypothesis in the experiment.
- **QEX6: Does the study provide a replication package?** The aim of this question is to determine if the study provides replication facilities. We checked the replication package because the replication lies at the core of the experimental paradigm and is regarded as the cornerstone of scientific knowledge [29].
- **QEX7: Does the study use causality terms to discuss its findings?** The aim of this question is to gain an understanding of the concept of causality, we adopted a subjective yet practical approach by noting whether the studies utilized causality-related terms such as "our results imply/corroborate/show the effect of" to elucidate their findings [6].

All of the questions can be answered in a binary manner (yes/no). The assessment employed incorporates attributes that are associated with the experiment's hallamarks (QEX1, QEX2, and QEX3), alongside other pertinent attributes (QEX4, QEX5, QEX6, and QEX7). It is important to distinguish between the hallmarks of an experiment and additional attributes related to its quality. In this work, we consider manipulation, control, and randomization (QEX1–QEX3) as the defining hallmarks that determine whether a study can be classified as an experiment. The remaining questions (QEX4–QEX7), including the presence of a replication package (QEX6), are not used as classification criteria but rather as indicators of the quality, rigor, and reproducibility of the experimental design. In particular, while a replication package is not strictly required for a study to be considered an experiment, its absence limits the ability to reproduce the study and, consequently, weakens the reliability and impact of its findings. This distinction allows us to separate the identification of experiments from the assessment of their methodological strength.

4.4. Procedure to classify the empirical studies

The data extraction from the 37 + 11 papers of the dataset was carried out by two of the authors of this work. The process (depicted as (6) in Fig. 3.) was similar to that proposed in [24] and was as follows:

1. Two of the authors selected 10 papers to pilot the analysis. These 10 papers contained 5 case studies and 5 experiments. To select the papers, they followed the order in which papers are presented in [2]. The authors independently analyzed the 10 papers and later discussed their classifications until reaching an agreement.
2. The remaining papers (32 + 6) were then randomly assigned to each of the two authors involved in the analysis. This means one author analyzed 16 case studies and three experiments, while the other author analyzed 16 case studies and three experiments.
3. Each author then responded to the aforementioned questions for every paper included in the dataset (questions QCS1 to QCS7 if it was a paper categorized as a *case study*, and questions QEX1 to QEX7 if it was a paper categorized as an *experiment*). A specific response and an explanatory note were provided for each question to facilitate subsequent analysis. If doubts arose regarding the answer to a question for a particular article, this answer was flagged for further analysis.
4. After completing the initial round of reviews, the authors, grouped in pairs, discussed the answers to the flagged questions until a consensus was reached.
5. Finally, the data were processed to obtain the results presented in Sections 5 and 6, to answer the research questions. This process is depicted as (7) in Fig. 3.

5. Assessment on the use of the term *case study* in GSE

After analyzing the 37 primary studies with the checklist described in Section 4.2, we found that only one met the criteria to be considered a *case study*: having a (yes) answer in QCS1 - QCS4 and a (no) answer in question QCS5. Given that over 97% of the articles were misclassified as *case studies*, we looked more closely at how these articles should be classified. We relabeled four of these 37 articles (11%) as *action research*. Five articles (14%) were relabeled as *archival analysis*, and the remaining 27 (73%) articles were relabeled as *small-scale evaluation*. Table 2 presents, for each analyzed study, the responses to the research questions together with their aggregated percentages Table 3 shows these values according to the relabeling we have done to the papers.

5.1. Case study

During the analysis of the articles, we confirmed that only the article by Engstrom et al. [S1] was a *case study*. The authors present a case study as part of the evaluation of a Design Science Research Project. They evaluate the use of Text-To-Speech (TTS) Systems in a real-life context of video game production created by a university and a company. They use several methods of data collection during the development of the video game analyzed: individual field notes; project documentation (e.g., design documents, test protocols, and other reports); internal communication within the project group, mainly through Slack; and game analytics data and user reviews from Google Play and App Store.

5.2. Action research

Different authors describe *case studies* and *action research* as primary empirical research methods along with experiments and surveys [27, 24, 34]. *Action research* is also described as a research methodology that is used for academia-industry research collaborations such as Design Science or Technology Transfer Model [35, 34]. In this paper, we refer to it as an empirical qualitative research method. There are authors [36, 24] that admit that the distinction between *case study* and *action research* as qualitative methods is not clear cut. For them, the main difference between *case studies* and *action research* lies in the researchers' role and their participation as agents of change. Beyond researcher involvement, a key distinction lies in the structured intention to introduce and evaluate change in the studied context. In action research, the researcher actively seeks to influence the environment while assessing the effects of that intervention. In contrast, case studies primarily aim to observe and understand phenomena without deliberately driving change. Similarly, in this work, we relabeled the studies of a real context situation carried out by researchers involved in that situation as *action research*. In a *case study*, the role of the researcher is more like that of an external narrator in a novel, whereas in *action research*, the researcher would be one of the protagonists. The primary advantage of *action research* is that it allows the researcher to gain a thorough and firsthand understanding. However, one potential weakness is that objectivity may be compromised when researchers strive to achieve a favorable outcome for the client organization [27].

The studies of Mirza-Babaei et al. [S18][S35] are examples of studies that we relabeled as *research actions* instead of *case studies*. In both studies, the researchers collaborated

Table 2: Results of Case Study Test

SLR ref	QCS1	QCS2	QCS3	QCS4	QCS5	QCS6	Is it a Case study?
S6	Yes	No	Yes	Yes	Yes	0	No
S7	Yes	No	No	No	No	0	No
S8	Yes	No	No	Yes	No	0	No
S9	Yes	No	No	No	No	0	No
S10	Yes	Yes	No	No	yes	0	No
S11	Yes	No	No	Yes	Yes	0	No
S12	No	No	Yes	Yes	Yes	0	No
S13	No	Yes	No	No	No	0	No
S14	Yes	No	Yes	Yes	No	0	No
S15	Yes	Yes	Yes	Yes	No	0	Yes
S16	Yes	Yes	No	No	No	0	No
S17	Yes	No	Yes	Yes	Yes	0	No
S18	Yes	Yes	Yes	Yes	Yes	25	No
S19	Yes	No	No	Yes	No	0	No
S20	Yes	No	No	Yes	Yes	0	No
S21	No	No	Yes	Yes	Yes	0	No
S22	Yes	No	No	Yes	Yes	0	No
S23	Yes	No	No	Yes	Yes	0	No
S24	Yes	No	No	Yes	Yes	0	No
S25	Yes	No	No	Yes	Yes	0	No
S26	Yes	No	No	Yes	Yes	0	No
S27	Yes	No	No	Yes	Yes	0	No
S28	Yes	No	No	Yes	Yes	0	No
S29	Yes	No	Yes	Yes	Yes	0	No
S30	Yes	No	No	Yes	Yes	0	No
S31	Yes	Yes	Yes	Yes	Yes	0	No
S32	Yes	No	No	Yes	Yes	0	No
S33	Yes	No	No	Yes	Yes	0	No
S34	Yes	No	No	Yes	No	0	No
S35	Yes	Yes	Yes	Yes	Yes	19	No
S36	Yes	No	Yes	Yes	No	0	No
S37	Yes	No	Yes	No	No	0	No
S38	Yes	No	Yes	Yes	Yes	0	No
S39	Yes	No	No	Yes	Yes	0	No
Yes	34 (92%)	8 (22%)	15 (41%)	31 (84%)	24(65%)	3 (8%)	1 (3%)
No	3 (8%)	29 (78%)	22 (59%)	6 (16%)	13 (35%)	34 (92%)	36(97%)

Table 3: Reclassification of the primary works and their results

	Works		QCS1	QCS2	QCS3	QCS4	QCS5
Case Study	1 (3%)	*Yes*	1 (100%)	1 (100%)	1 (100%)	1 (100%)	0 (0%)
		No	0 (0%)	0 (0%)	0 (0%)	0 (0%)	1 (100%)
Action Research	4 (11%)	*Yes*	4(100%)	4 (100%)	3 (75%)	4 (100%)	4 (100%)
		No	0 (0%)	0 (100%)	1 (25%)	0 (0%)	0 (0%)
Archival Analysis	5 (14%)	*Yes*	4 (80%)	2 (40%)	0 (0%)	1 (20%)	1 (20%)
		No	1 (20%)	3 (60%)	5 (100%)	4 (80%)	4 (80%)
Small-scale evaluation	27 (73%)	*Yes*	25 (93%)	1 (4%)	11 (41%)	25 (93%)	19 (70%)
		No	2 (7%)	26 (96%)	16 (59%)	2 (7%)	8 (30%)

with a company that provides different kinds of services (e.g., funding, mentorship, analytics services) to indie game developers. The researchers designed and conducted playtesting sessions through this company for indie developers and analyzed the results to study playtesting in indie video games. In the study [S35], one of the authors was involved in facilitating all playtesting sessions, which were conducted externaly to three development teams of three different indie video games. In the paper, they conducted a comparative analysis of review data of the three video games and their respective playtesting reports. The authors called each of the games involved in the study a case study, including the various interactions with developers, playtesting design and execution, analysis of game reviews, and comparative analysis. In this case, we consider the term *active research* to be more appropriate since at least one of the researchers participates actively as a change agent.

Another two studies were relabeled as action research. In the study of Geisler et al. [S31], the researchers create a domain-specific language and evaluate it in collaboration with an indie studio. The researchers monitored the language usage of a video game that is in active production. Magy Seif El-Nasr [S30] uses the collaborative work accomplished with two different game companies as examples to demonstrate the utility of their approach. As can be observed in Table 3, second row, all of the papers relabeled as action research used a real industrial context with which the researchers interacted at the time of the research. However, the use of different data collection methods is not explicit in all of them and only the studies of Mirza-Babaei et al. [S18][S35] are positive for the smell indicator. However, the explicit reference to the word 'interview' is because the interviews are part of some playtesting sessions.

5.3. Archival Analysis

Archival analysis is a research method that obtains data through extraction from archives such as code repositories [24]. Some authors consider it a data collection strategy, since methods such as case studies can incorporate archival data analysis as one of their multiple sources. The five papers that we relabeled as *archival analysis* obtained (no) in the question related to the contemporaneity of the phenomenon. In a software development context, we can understand 'contemporary' to refer to a period of reasonably recent events for which it is possible to collect new information (e.g., through interviews) that can complement any other collected information (e.g., repositories). When a study does not investigate a contemporary phenomenon, the study is probably an *archival analysis* or a *small-scale evaluation*, depending on the research question of the work and the level of control of the researcher in relation to context, artifacts, data, or participants.

All of the papers that are reclassified as archival analysis used existing data and only a single data collection method. Although they used different types of data (e.g., game log data and patch data), they used only one source (e.g., data from a server with real player data). Only in one of these studies does the researcher actively participate as an agent of change. The study of Ramadan and Widyan [S13] presents a new game development life cycle (GDLC), and, as an example of evaluation, they analyze the results of the proposed GDLC when it was applied in a project to create a mobile game. It is not a contemporary phenomenon because they analyzed a game development project from the past; they used a previous result of Ramadam [S13] where the GDLC was used. On the other hand, we considered that in this study [S13], the case is not clearly identifiable and the answer to the first question of the questionnaire is (no). A case might be understood as a single, empirical configuration of actors, activities, technologies, and artifacts, all within a context [37]. In [S13], the goal of the research is to propose a new game development life cycle (GDLC) and the corresponding guidelines, but the author does not define a specific context of application with specific actors, activities, or technologies.

In software engineering, *archival analysis* is often labeled data mining or mining repositories [23]. The studies of Hullett et al. [S7][S9] study how data collected from users of a released game can inform subsequent development. For the analysis presented in the paper, they mined 3.1 million entries of game log data from several thousands of users who played a release game. Kika et al. [S19] used data from real players from League of Legends servers to analyze the impact of patching on the gameplay. Salazar et al. [S10] propose an improved Game Design Document (GDD) based on a comparative analysis of GDD documents found in the literature. This study could also be labeled as a literature review.

5.4. Small-scale evaluations

We relabeled more than 70% of the articles of the case studies data set as *small-scale evaluations*. A *small-scale evaluation* is a study that comprises a single researcher or a small team, run over a short period, with limited resources, and most often occurring at a single site. This category groups together various forms of ad hoc studies. In many cases, these studies may still follow established methodologies, such as experiments, surveys, or case studies, but in a lightweight or less rigorous manner.[38]. This description is appropriate for studies related to engineering research where something novel is constructed, and its value needs to be assessed [35]. The researchers in these studies aim to illustrate, demonstrate, prove a concept, conduct a feasibility study, or demonstrate the value of a research endeavor [24]. Despite the fact that these kinds of evaluations are essential for software engineering, these evaluations are not *case studies*. The articles were relabeled as *small-scale evaluations* that conducted some kind of empirical investigation but not in a real-life context. Instead, the researchers use some kind of constructed or simulated environment or reuse an existing case study for new objectives.

In a *case study*, the level of control is limited because the study investigates a contemporary phenomenon in a real-world context. When the level of control increases, this suggests that the study is more likely to be an evaluation rather than a *case study*. Wohlin and Rainer [24] propose an assessment scheme to distinguish a *small-scale evaluation* from a field study such as a *case study*. This assessment is based on three questions related to the researchers' level of control over the study that they conduct in relation to context, artifacts or data, and participants, respectively. These questions are: (1) Does the study take place in a real-life context?; (2) Do the artifacts or data used come from the context under study?; and (3) Do the participants or collaborators in the study perform their regular work? a (no) answer to one of these questions is enough to indicate that the study is more likely a *small-scale evaluation* than a *case study* or any other type of field study. None of the papers relabeled as *small-scale evaluation* obtains a (yes) answer for the second question, 'Will the study be conducted in a real context?' (see the last two rows of Table 3). This is because all of these papers get a (no) answer to one of the three questions that determine the researchers' level of control over the study in relation to the context.

In more than 80% of the papers relabeled as *small-scale evaluations* (22 of 27), the researchers use data or developments obtained from real contexts, particularly from real video games. However, this information is used out of context since it is used to validate or evaluate the researchers' proposals. The term *case study* in these studies is linked with the video game that researchers use as an example in their evaluation. In [S15], the authors test and evaluate their approach (model-based testing approach) using data from two existing real examples. The authors call each of these two evaluations a *case study*. In [S36], the authors use real user data from a AAA [2] game (The Elder Scrolls Online) as a real online game example to evaluate the 'animation canceling' technique in terms of network traffic. For them, the evaluation with data from a real example is the *case study*. In [S4], the authors use the term *case study* to refer to the video game (Tom Ryder) used to extract the data of real players to perform spatial analysis of gameplay metrics using Geographic Information Systems as a tool. This approach is introduced as new in the toolbox of user-experience testing. In [S29], the authors create the environment for evaluating the hybrid system framework that they present in their study, following the constraints of two real games 'Flappy Birds' and 'Super Mario'. Again, the researchers referred to these two evaluations as two *case studies*.

There are five papers that are relabeled as *small-scale evaluations* that do not use data from real video games to validate or evaluate the researchers' proposals. In these papers, the researchers use their own developments to conduct the evaluation. As in [S25], the term *case study* is used as synonymous with a specific example or an application example. In [S25], the researchers propose the transpilation to reuse the implementation of motion controllers in multiple game engines. They present an example of an application that develops three common human behaviors in the Haxe programming language in two different video game engines (the open-source Blender Game Engine and the commercial Unity engine). They compare it with a manual implementation with respect to performance. They called the part of the paper where they describe this comparison a *case study*, but it is not a *case study* because the artifacts and data are not from a real-life context.

Approximately 20% of the papers that were reclassified as *small-scale evaluations* explicitly stated the experimental setting, such as in [S6][S5][S25][S39]. Although these papers could have been considered *controlled experiments*, they lacked detailed information on specific aspects of experimental design, data collection, or statistical treatment of data. The authors do not always explain which statistical tests they will use, why, or whether the conditions necessary to be able to apply these tests are verified. For example, in the work of Nunnari and Héloir [S25], they use Welch's t-test to compare the execution time of two conditions. However, there is no justification as to why this test is used instead of a Student's t-test (one or the other is used depending on whether the variances of the groups being compared are different or equal). Also, the assumptions necessary to apply Welch's t-test are not justified since, in this case, the assumption of normality is not explicitly checked.

5.5. Smell Indicator

With regard to the smell indicator for incorrect classifications of *case studies*, Table 4 presents the confusion matrix for the performance of the indicator against the evaluation dataset. We use accuracy to measure the performance of the confusion matrix, and we have also included precision and recall. Although the accuracy of the smell indicator is 92% in our data set, no

[2] A AAA (Triple-A) game is a high-budget video game developed by a major studio, known for its advanced graphics, extensive gameplay features, and significant marketing efforts. These games typically represent top-tier quality in the gaming industry.

Table 4: Confusion Matrix and Performance for the case study data set.

Confusion Matrix

		Smell indicator suggests the study is:	
		Case study	Not case study
The study is	Case study	0	1
	Not case study	2	34

Performance measure

Accuraccy	(0+34)/(0+1+2+34) =0.92
Precision	(0)/(0+2) = 0
Recall	(0)/(0+1) = 0

true positives were obtained. Therefore, both recall and precision obtained values of zero. The only paper confirmed as a *case study* makes no reference to interviews. On the other hand, two false positives were detected corresponding to papers that conducted interviews as part of playtesting sessions. The work where the smell indicator was proposed obtained accuracy values of 77% and 68% for the smell indicator in the datasets used for its development and evaluation, respectively. These datasets consist of journal articles that explicitly cite the guidelines of Runeson and Höst (2009) [20], and the authors note that the indicator may not perform as effectively for other types of datasets. Note that none of the papers discussed in this section made explicit reference to a formal definition of *case study* such as those provided by Runeson et al. [22] or Wohlin et al. [23].

6. Assessment on the use of the term *experiment* in GSE

After analyzing the 11 primary studies with the checklist described in Section 4.3, we found that only two met the criteria to be considered an experiment: having a (yes) answer to all of the questions. Note that the assessment used includes characteristics related to the experiment hallmarks (QEX1, QEX2, and QEX3) and other relevant characteristics (QEX4, QEX5, QEX6, and QEX7).

Table 5 shows the individual answers to each of the questions from the different studies, together with the number and percentage of the papers that meet each criterion at the bottom of the table.The last column shows the studies that meet all of the criteria (less than 20%). Taking into account the questions individually, we notice that all of the papers meet the criteria for questions QEX1 and QEX7, related to manipulation and causality, respectively. For QEX1, the authors have introduced a treatment or independent variable in all of the papers. Similarly, for QEX7, all of the papers declared causality. In other words, all of the studies discuss their results or use terms related to causality, such as imply, show the effect of, or corroborate [6].

However, on the other hand, it can be observed that questions QEX6 and QEX2 obtain the lowest compliance related to the replication package (27%) and control (36%), respectively. Replication is at the heart of the experimental paradigm [29]. For an idea to become established, experiments must be repeated under similar circumstances multiple times. Without replication, it is impossible to tell if the results happened by chance [29]. Question QEX6 (replication facilities) stands out with the lowest compliance rate (27%), underscoring the importance of highlighting the need to provide a replication package when conducting an *experiment*. Notably, two of the papers meet all of the criteria except this one. One of them is the experiment conducted by Zhong and Xu [S48] in which they performed an experiment to study how game updates affect players' game engagement. This experiment could serve as a commendable example due to its clarity (well-defined and ordered sections), except for the absence of a replication package, which penalized it. The other *experiment*, which is the experiment conducted by Prado et al. [S44], fulfills everything except that it does not have a replication package. They conducted an experiment to validate their approach, which uses a game engine, multiple DSLs, code generation, and design patterns to integrate manual code with generated code.

A good replication package should include all of the necessary materials and documentation in order to enable others to replicate the experiment and verify the results. An example is the replication package provided by Domingo et al. in [S2][S3], and by Prado et al.[S44], . They provided a link where you can find all of the files of the replication package: consent process data, the questionnaires, the tasks, and the statistical analysis. Hence, they did everything necessary to replicate the experiment and check the results available.

The aim of question QEX2 is to ascertain whether control mechanisms have been employed in the experiment to control the effect of confounding factors. According to [39], confounding factors may affect the dependent variables without the researcher's knowledge. It is, hence, crucial to try to identify the factors that otherwise may affect the outcome in an undesirable way. The researcher should block or control these factors in order to prevent their potential effect and avoid contamination of the values obtained from the dependent variable. In order to counteract their influence, constant values can be assumed, or randomness can be introduced. Of the 11 studied experiments, only four (36%) applied mechanisms to control undesired effects. It is noteworthy that three of the studies that apply control mechanisms use guidelines as a basis for experiment design. Another notable aspect is that these three experiments consider and control participants' experience as a confounding factor. The paper by Prado et al. [S44] is the only one that controls confounding factors; however, it does not have a statistical significance study. This paper exclusively uses descriptive statistics. Considering the seven studies that do not use mechanisms to control confounding variables, there is a possibility that their potential effects have been described as threats to validity or similar. Of these seven studies, only one of them contains threats to validity and does not reference the existence of confounding factors. An example of how to present this information can be seen in Domingo et al. in their studies [S2] [S3] in the "Variables" section. Another example can be found in Zhong and Xu's study [S48] in the "Summary statistics" section.

Table 5: Results of Experiment Test

SLR ref	QEX1	QEX2	QEX3	QEX4	QEX5	QEX6	QEX7	Meets all criteria
S2	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
S3	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
S40	Yes	No	No	No	No	No	Yes	No
S41	Yes	No	Yes	No	Yes	No	Yes	No
S42	Yes	No	No	No	No	No	Yes	No
S43	Yes	No	Yes	No	No	No	Yes	No
S44	Yes	Yes	Yes	Yes	Yes	No	Yes	No
S45	Yes	No	No	No	Yes	Yes	Yes	No
S46	Yes	No	Yes	No	Yes	No	Yes	No
S47	Yes	No	Yes	Yes	No	No	Yes	No
S48	Yes	Yes	Yes	Yes	Yes	No	Yes	No
Yes	11 (100%)	4 (36%)	8 (73%)	5 (45%)	7 (64%)	3 (27%)	11 (100%)	2 (18%)
No	0 (0%)	7 (64%)	3 (27%)	6 (55%)	4 (36%)	8 (73%)	0 (0%)	9 (82%)

The question QEX3 is related to randomization. Randomization is a process where study units are allocated to receive either the treatment or an alternative condition through a random procedure. It is worth noting that randomization can be extended to sample selection to enhance external validity in various empirical studies. Although random sampling selection is considered beneficial, the random assignment of treatments to entities is imperative in *controlled experiments*. This serves as the default control to mitigate bias and establish causality [33]. Eight out of the 11 experiments used randomization, representing 73% of the experiments. Specifically, the three studies that used guidelines for their design incorporated randomization. One of these three studies used a randomization process but did not explicitly state it in the paper. In total, we have found four studies that implemented randomization but did not explicitly detail or explain the reasons for using it. Of the three studies that did not introduce randomization, none of them examined the statistical significance of treatments or described validity threats where the absence of randomization could be considered. An example of how to present this information can be seen in Zhong and Xu's study [S48], which explains that the participants were randomly selected in the "Data collection" section. They emphasize this again in the "Model" section, as does Domingo et al. [S2][S3] in the "Procedure" section.

For questions QEX4 and QEX5, which are related to focused scope and statistics, respectively, approximately half of the papers fulfill the criteria, and the other half do not. A focused scope enables researchers to define the objectives and parameters of the study precisely. A good focused scope eases the identification of causal relationships and enhances the validity of the findings. By establishing a clear hypothesis, researchers can guide their efforts toward specific goals and determine the appropriate controls needed for rigorous experimentation. Additionally, addressing limitations or threats to validity provides practical insights into the focused nature of the study. Our research reveals that most of the papers do not make the hypothesis explicit, nor do they offer a validity analysis. An example of how to present a focus scope is the study of Zhong and Xu, who evaluate how game updates affect players' game engagement. This paper proposes a "Review and Hypothesis" section that delineates the research scope and presents the specific hypothesis to be tested through the study [S48]. Other examples are proposed by Domingo et al. in [S2] [S3], in which they relate one or more hypotheses to each proposed research question to evaluate each of the answers. On the other hand, statistical analysis in empirical studies provides researchers with the tools to analyze and interpret collected data. Statistical analysis is essential for establishing the validity of findings and adjusting for potential confounding effects when dealing with vast amounts of data. In addition, the chosen statistical methods must align with the study's design in order to avoid inappropriate usage. However, our research reveals that most of the papers only use descriptive analysis, do not justify the test used, and do not offer information about normality or other conditions to execute this test. An example of how to explain the procedure carried out in the statistical analysis is the one made by Domingo et al. in [S3] in the "Analysis procedure" section.

It is worth noting that the two papers that meet all of the criteria that make them categorized as *experiments* have referenced and followed the guidelines of Wohlin [28], and Basili [27]. This highlights the importance of documenting and following the appropriate guidelines to avoid misusing the term *experiment*.

7. Main Recommendations and Insights

Our work reveals the strengths and weaknesses of the GSE community. With regard to case studies, the GSE community is adept at using real-world video games in its research. However, it tends to confuse the use of a real-world video game with a

case study. In other words, it is a widespread view among GSE researchers that using a real-world video game is a sufficient condition for the work to be considered a case study.

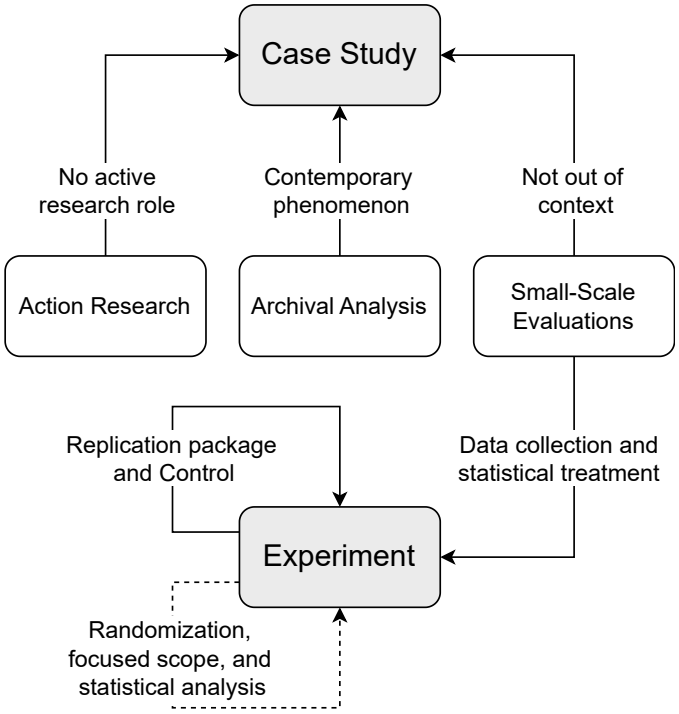

Figure 4: Key points for fulfilling the proper criteria as *case studies* or *experiments*.

To help the GSE community, our work pinpoints key points for fulfilling the proper criteria as *case studies*. Fig. 4 shows the key points to be addressed by *action research* (no active research role), *archival analysis* (contemporary phenomenon), and *small-scale evaluation* (not out of context).

For *action research* in the GSE community to be *case studies*, it is key for researchers to pay attention to the role of the researcher. They should avoid any researcher having an active role (such as a protagonist). Instead, they should have a neutral role (such as that of a narrator). For example, if the case study is to evaluate a certain aspect of video game development, the researcher must not have participated in the development of the video game.

For *archival analysis* of the GSE community to be *case studies*, it is key for researchers to pay attention to the timing of the phenomenon they are studying. They should avoid extracting information from archives as repositories. Instead, they should use phenomena that are sufficiently recent so that it is possible to collect new information. For example, they should not use log entries from the past but rather logs that have recently been collected, with the possibility of more if necessary.

For *small-scale evaluations* of the GSE community to be *case studies*, it is key for researchers to pay attention to the context of the study. They should avoid taking the study out of context through simulated settings. Instead, they should maintain the original context. For example, they should not design specific playtesting sessions but rather observe the players in their real game environment.

In the case of *small-scale evaluations*, the GSE community should also reflect on whether it is really aiming at a *case study* or an *experiment*. One out of five small-scale evaluations makes the design of the experiment explicit. This work can help the community to better understand both small-scale evaluations and experiments in order to make an informed decision. For the GSE community's *small-scale evaluations* to be *experiments*, researchers must pay more attention to data collection and the statistical treatment of data. The statistical study we perform on the data in an experiment depends both on the experimental design used to collect those data and on the characteristics or statistical properties of those data. The authors should clearly define the experimental design they have used to obtain the data and justify the type of statistical tests they can use based on the characteristics of their data. On the other hand, they should check whether the data to which these tests are applied verify the assumptions necessary to consider the results valid (e.g., sample size, normality, or homogeneity of variances). Otherwise, the statistical study performed will lose its value. Fig. 4 shows graphically this dichotomy between small-scale evaluation and the GSE community experiment.

In the case of *experiments*, the GSE community is proficient in manipulation and causality (solid lines in Fig. 4). However, it is especially weak in replication package and control, and to a lesser extent, in randomization, focused scope, and statistical analysis (dashed lines in Fig. 4).

It is important to note that action research, archival research, and small-scale evaluations are not inherently of lower quality than case studies or experiments. This work focuses on methodological characteristics and classification rather than quality assessment. Investigating potential relationships between method choice and study quality remains an interesting avenue for future research. To improve the GSE community's experiments, they must focus on and enhance each of the above. According to [40], the packaging replication should contain 1) instructions for the replicating experimenter, 2) operational materials, 3) support for the experimental research process, and 4) structural and usability enhancements. On the other hand, it should minimize biased results due to confounding variables. Confounding variables are those that interfere with the relationship between the independent and dependent variables in research. To minimize the effect of confounding variables, control techniques should be employed, such as: random assignment, matching, counterbalancing, blocking, or repeated measures [41].

Our work also reveals studies that are exemplary of both *case studies* and *experiments* for GSE. In the case of the study of Engstrom et al. [S1], it fulfills all of the case study questions. The community can use it as a reference for inspiration on how to answer these questions successfully in GSE. The two studies carried out by Domingo et al. [S2] [S3] achieved the highest level of compliance to date that an *experiment* should have. These works can be an inspiration for the community to make their experiments better.

Based on our results, we highlight seven main recommendations and insights below.

1. How to classify GSE work

The use of the questionnaires outlined in Sections 4.2 and 4.3 to self-assess GSE research work can be helpful in determining to what extent an empirical GSE work can be classified in one way or another and how one should refine empirical methods to meet the standards expected by the scientific community.

This paper gathers guidelines for *case studies* and *experiments*. These are particularly relevant to the GSE community because: (1) case studies and experiments are the most commonly used empirical methods by the GSE community [2]; and (2) we have identified small-scale evaluations that lie on the borderline between case studies and experiments that can potentially be improved in both directions as we show in this paper.

Beyond its use for classifying individual studies, the proposed approach can also support a broader assessment of the maturity of empirical research in GSE. By reapplying the same classification framework and checklists to updated sets of publications over time, it would be possible to track the evolution of methodological rigor in the field, identifying trends, improvements, or persistent weaknesses. In this sense, our approach can also serve as a benchmarking instrument for future GSE research. Researchers may use the checklists as a self-assessment tool during study design, while reviewers may adopt them as a reference to evaluate methodological soundness. This dual use positions the framework not only as an analytical tool, but also as a practical mechanism to foster continuous methodological improvement in the GSE community.

2. A case study must be conducted in a real context

Specifically, to call a paper a *case study*, we must ask ourselves whether we are actually using a real context (real artifacts with real people in the real problem environment). Otherwise, our paper is not a *case study*. On the other hand, we should avoid calling the examples on which we develop our evaluation *case studies*.

The results suggest a significant portion (78%) of the analyzed case studies did not utilize a real-world context for their research. Most of these studies have been reclassified as *small–scale evaluations*. A potential drawback of the term *small–scale evaluation* is that it may not be perceived as sufficiently impressive. However, the *small-scale evaluation* is a natural and valuable stepping stone in the quest to propose new research solutions [24].

3. Weakness of the experiments

To improve *experiments* in GSE, we should be able to develop them in such a way that they are replicable. As described in Section 6, the possibility of replication in empirical studies is a fundamental characteristic. The absence of a methodology and a replication package is even more crucial in video game development, given that empirical studies often rely on a single game. To enable result extrapolation, it is advisable for empirical studies in the context of video games to include a detailed description of the methodology and an accessible replication package.

The design of an *experiment* should be characterized by objectives, hypotheses related to these objectives, metrics to validate these hypotheses, and subsequent statistical analysis. In Section 6, we found that 50% of the experiments do not explicitly state these characteristics. The objectives of the experiments should be clearly defined, as well as their hypotheses and the variables to be used. These improvements would help to define and use experimental designs that allow for more robust statistical analyses of the data obtained, and thus improve the validity of the conclusions drawn from the experimentation.

4. GSE is an active community that takes into account real developments and data

Our work shows that, despite the lack of experience of the GSE community, it is an active community that takes into account the contact with real developments to carry out its empirical evaluations, either through action research or by taking real developments or data from real videogames as examples in its evaluations.

5. Playtesting

The process for developing video games, similar to software development, requires testing the generated video game. In this development process, there are different types of testing [42]: focus groups, QA testing, usability testing, and playtesting. The aim of playtesting is to verify whether, when playing the video game, players have a gaming experience for which the video game was designed. The players' experience is a consequence of the aesthetic aspect of the video game, the game mechanics, and the storytelling [2]. In other words, in addition to the technical aspects, playtesting considers other characteristics of the video game, due to the relevance of playtesting in video game development, addressing playtesting techniques has become a significant topic in the scientific literature in the GSE field. Playtesting techniques are not described as a specific type of empirical evaluation. As we have described in Section 5, the studies that describe playtesting are not *case studies*, but they cannot be considered *experiments* either since the number of participants or conditions are not always appropriate. We consider that: 1) it is useful for the GSE community to know how playtesting studies are conducted in the industry due to their peculiarities; and 2) the GSE community should make an effort to ensure that the terminology and the methodologies related to empirical evaluation are used correctly when performing and documenting playtesting tests.

6. The importance of proper referencing

Most of the papers we have highlighted as good practices in experiments use clearly referenced guidelines both in the study and in structuring the written work. However, none of the papers in the data set of case studies use specific guidelines on how to conduct and narrate case studies. The recommendations and guidelines offer valuable support for designing, executing, and reporting empirical studies with rigor and systematic methodology [6].

The GSE community is encouraged to select appropriate guidelines for developing their empirical work. On the other hand, as far as we know, there are currently no comprehensive methodological guidelines available to assist researchers in the design and execution of empirical studies in the context of video game development. It would be valuable for researchers in empirical GSE studies to recognize the need for specific guidelines to promote the proper utilization of SE empirical methods and to establish a common terminology between academia and the video game industry.

7. Smell indicator in GSE

The smell indicator proposed by Wohlin and Rainer [24] may not be appropriate in GSE. For example, playtesting usually involves interviews, but playtesting is not a case study (out of context), which generates false positives. In fact, we found more false positives than true positives. Defining a smell indicator for experiments was not the goal of this work; we only evaluated the indicator proposed by [24] for case studies. Developing a similar indicator for experiments would require dedicated effort and likely vary depending on the type of experiment (e.g., controlled, quasi-, or field). We consider this an interesting direction for future work that will help with the comparability between the case study and the experiment checklists. Furthermore, given the GSE community's weaknesses in empirical evaluation, our recommendation is that researchers be guided by the questions rather than by a smell indicator. The questions have a didactic component that can benefit the GSE community at this level of empirical maturity.

8. Threats to validity

In this section, we describe threats to validity that we have been unable to eliminate, that is, threats that have been mitigated or could not be avoided. To describe these threats, we have used the classification proposed in [28] and the study of threats to validity analyzed in [43].

Construct validity: This facet of validity gauges how well the measures used in a study accurately capture what the researchers intend to study. Issues may arise regarding whether the concepts are sufficiently defined before measurements are established. We have mitigated the threat of an incomplete dataset because the dataset was generated with a systematic literature review in the field of GSE following best practices as suggested by [32]. The threat of an incorrectly classified dataset was mitigated because the dataset authors contacted the authors of the papers that were included in the dataset via email to confirm the classification of each paper. Finally, to mitigate the threat of formulating inappropriate research questions, dedicated sessions were conducted with all of the co-authors. These sessions aimed to discuss and evaluate the most suitable research questions, ensuring the creation of a comprehensive overview of *case studies* and *experiments* in GSE that would benefit both researchers and practitioners.

Internal Validity: This facet of validity becomes relevant when examining causal relationships. The concern here is the potential influence of other unaccounted factors on the investigated factor. In this work, the size and number of samples pose a threat, which has been mitigated due to the focus on articles in software engineering within the video game industry. Another threat is the subjective quality assessment, which was mitigated by pooling the classification of the 10 initial articles, as described in subsection 4.4. Finally, the threat of an absence of expert evaluation has been reduced, as two authors of this work perform research in the context of empirical software engineering.

External Validity: This facet of validity examines the degree to which the findings can be applied beyond the specific study context and whether they hold relevance for other scenarios. There is a potential risk that the 37 + 11 papers retrieved may not accurately represent the target population in GSE. To minimize this risk, the selected papers are focused on industry-scale computer games. The study is affected by the threat of a restricted timespan because the 37 + 11 papers span from 2009 to 2021.

Conclusion Validity: This facet focuses on the extent to which the research process can be reproduced with consistent results. Researchers may inadvertently bias the results by seeking a specific outcome. To minimize the threat of subjective interpretation of the extracted data, we clearly separated the assessment (Sections 5 and 6) from the recommendations and insights (Section 7). The assessments show extracted data after the papers' classification, and the recommendations and insights provide different points regarding the state of *case studies* and *experiments* in GSE according to the authors' interpretation. To mitigate the threat of bias in data extraction, a protocol was designed during the planning phase. Two of the authors answered the questions to extract the data, and, then all of the authors (in pairs) discussed the doubtful answers until a consensus was reached. The overlap in authorship between the original SLR and this work may introduce bias. However, independent validation steps were applied to mitigate this risk. Finally, to improve this validity facet, we provide a detailed description of the procedure followed in this work (see Fig. 4) and the dataset used to allow the replication study.

9. Conclusion

This work examines how the terms *case study* and *experiment* are utilized within the Game Software Engineering (GSE) community.

Our analysis indicates that only a small percentage of the studies analyzed employ these terms correctly, or rather less than in traditional software engineering. Specifically, merely 3% of case studies accurately labeled the study as such, while the remaining 97% misclassified them. Similarly, only 18% of experiments met all of the defined criteria for what constitutes an experiment.

Furthermore, this work not only identifies instances of misapplication, but it also provides key points for tackling the weaknesses of the GSE community's case studies and experiments as well as illustrative examples. Furthermore, this work

provides recommendations and insights for future research in empirical GSE.

Research on GSE continues increasing in interest and the industry benefits from it [2]. To conduct research, the GSE community depends on case studies and experiments. Consequently, the aim of our work is relevant in order to raise awareness of the current situation and the proposed key points, exemplars, recommendations, and insights. Given the interdisciplinary nature of GSE, our work could be complemented by analyzing game specific publication outlets that reflect experimental practices not fully captured in Software Engineering research.

Acknowledgments

This work was supported in part by the Ministry of Economy and Competitiveness (MINECO) through the Spanish National R+D+i Plan and ERDF funds under the Project VARIATIVA under Grant PID2021-128695OB-I00 and in part by the Gobierno de Aragón (Spain) (Research Group S05_20D).

Studies analyzed

[S1] H. Engstrom, P. A. Ostblad, Using text-to-speech to prototype game dialog, Computers in Entertainment (CIE) 16 (2018) 1–16.

[S2] Á. Domingo, J. Echeverría, Ó. Pastor, C. Cetina, Evaluating the benefits of model-driven development - empirical evaluation paper, in: S. Dustdar, E. Yu, C. Salinesi, D. Rieu, V. Pant (Eds.), Advanced Information Systems Engineering - 32nd International Conference, CAiSE 2020, Grenoble, France, June 8-12, 2020, Proceedings, volume 12127 of *Lecture Notes in Computer Science*, Springer, 2020, pp. 353–367.

[S3] Á. Domingo, J. Echeverría, Ó. Pastor, C. Cetina, Comparing uml-based and dsl-based modeling from subjective and objective perspectives, in: M. La Rosa, S. Sadiq, E. Teniente (Eds.), Advanced Information Systems Engineering, Springer International Publishing, Cham, 2021, pp. 483–498.

[S4] A. Drachen, A. Canossa, Analyzing spatial user behavior in computer games using geographic information systems, in: Proceedings of the 13th international MindTrek conference Everyday life in the ubiquitous era, 2009, pp. 182–189.

[S5] S. Papaloukas, K. Patriarcheas, M. Xenos, Usability assessment heuristics in new genre videogames, in: 2009 13th Panhellenic Conference on Informatics, IEEE, 2009, pp. 202–206.

[S6] S. Vickers, H. Istance, M. Smalley, Eyeguitar making rhythm based music video games accessible using only eye movements, in: Proceedings of the 7th international conference on advances in computer entertainment technology, 2010, pp. 36–39.

[S7] K. Hullett, N. Nagappan, E. Schuh, J. Hopson, Data analytics for game development nier track, in: 2011 33rd International Conference on Software Engineering (ICSE), IEEE, 2011, pp. 940–943.

[S8] J. Kienzle, A. Denault, Journey a massively multiplayer online game middleware, IEEE Software 28 (2011) 38–44.

[S9] K. Hullett, N. Nagappan, E. Schuh, J. Hopson, Empirical analysis of user data in game software development, in: Proceedings of the 2012 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, IEEE, 2012, pp. 89–98.

[S10] M. G. Salazar, H. A. Mitre, C. L. Olalde, J. L. G. Sánchez, Proposal of game design document from software engineering requirements perspective, in: 2012 17th International Conference on Computer Games (CGAMES), IEEE, 2012, pp. 81–85.

[S11] J. Donkervliet, J. Cuijpers, A. Iosup, Dyconits scaling minecraft-like services through dynamically managed inconsistency, in: 2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS), IEEE, 2021, pp. 126–137.

[S12] M. J. Best, D. Jacobsen, N. Vining, A. Fedorova, Collection-focused parallelism, in: 5th USENIX Workshop on Hot Topics in Parallelism (HotPar 13), 2013.

[S13] R. Ramadan, Y. Widyani, Game development life cycle guidelines, in: 2013 International Conference on Advanced Computer Science and Information Systems (ICACSIS), IEEE, 2013, pp. 95–100.

[S14] M. Suznjevic, I. Stupar, M. Matijasevic, A model and software architecture for mmorpg traffic generation based on player behavior, Multimedia systems 19 (2013) 231–253.

[S15] S. Iftikhar, M. Z. Iqbal, M. U. Khan, W. Mahmood, An automated model based testing approach for platform games, in: 2015 ACMIEEE 18th International Conference on Model Driven Engineering Languages and Systems (MODELS), IEEE, 2015, pp. 426–435.

[S16] T. Ollsson, D. Toll, A. Wingkvist, M. Ericsson, Evolution and evaluation of the model-view-controller architecture in games, in: 2015 IEEEACM 4th International Workshop on Games and Software Engineering, IEEE, 2015, pp. 8–14.

[S17] L. Chen, N. Shashidhar, D. Rawat, M. Yang, C. Kadlec, Investigating the security and digital forensics of video games and gaming systems a study of pc games and ps4 console, in: 2016 International Conference on Computing, Networking and Communications (ICNC), IEEE, 2016, pp. 1–5.

[S18] P. Mirza-Babaei, N. Moosajee, B. Drenikow, Playtesting for indie studios, in: Proceedings of the 20th International Academic Mindtrek Conference, 2016, pp. 366–374.

[S19] A. Kica, A. La Manna, L. O'Donnell, T. Paolillo, M. Claypool, Nerfs, buffs and bugs-analysis of the impact of patching on league of legends, in: 2016 International Conference on Collaboration Technologies and Systems (CTS), IEEE, 2016, pp. 128–135.

[S20] E. Guardiola, The gameplay loop a player activity model for game design and analysis, in: Proceedings of the 13th International Conference on Advances in Computer Entertainment Technology, 2016, pp. 1–7.

[S21] Y. He, T. Foley, T. Hofstee, H. Long, K. Fatahalian, Shader components modular and high performance shader development, ACM Transactions on Graphics (TOG) 36 (2017) 1–11.

[S22] B. R. Sagi, R. Silvestrini, Application of combinatorial tests in video game testing, Quality Engineering 29 (2017) 745–759.

[S23] D. Festa, D. Maggiorini, L. A. Ripamonti, A. Bujari, Supporting distributed real-time debugging in online games, in: 2017 14th IEEE Annual Consumer Communications & Networking Conference (CCNC), IEEE, 2017, pp. 737–740.

[S24] T. C. Kohwalter, L. G. P. Murta, E. W. G. Clua, Capturing game telemetry with provenance, in: 2017 16th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames), IEEE, 2017, pp. 66–75.

[S25] F. Nunnari, A. Héloir, Write-Once, Transpile-Everywhere Re-using Motion Controllers of Virtual Humans Across Multiple Game Engines, 2018, pp. 435–446.

[S26] K. A. Seitz Jr, T. Foley, S. D. Porumbescu, J. D. Owens, Staged metaprogramming for shader system development, ACM Transactions on Graphics (TOG) 38 (2019) 1–15.

[S27] S. Morisaki, N. Kasai, K. Kanamori, S. Yamamoto, Detecting source code hotspot in games software using call flow analysis, in: 2019 20th IEEEACIS International Conference on Software Engineering, Artificial Intelligence, Networking and ParallelDistributed Computing (SNPD), IEEE, 2019, pp. 484–489.

[S28] S. Lynch, K. Pedersen, F. Charles, C. Hargood, M22-a modern visual novel framework, in: Proceedings of the 8th International Workshop on Narrative and Hypertext, 2019, pp. 9–13.

[S29] Y. Zeleke, J. C. Osborn, R. G. Sanfelice, Analyzing action games a hybrid systems approach, in: Proceedings of the 14th International Conference on the Foundations of Digital Games, 2019, pp. 1–11.

[S30] M. S. El-Nasr, Developing games that capture and engage users, in: 2019 IEEEACM 41st International Conference on Software Engineering Companion Proceedings (ICSE-Companion), IEEE, 2019, pp. 9–10.

[S31] B. J. Geisler, F. J. Mitropoulos, S. Kavage, Gamespect aspect oriented programming for a video game engine using meta-languages, in: 2019 SoutheastCon, 2019, pp. 1–8.

[S32] C. Marín-Lora, A. Cercós, M. Chover, J. M. Sotoca, A first step to specify arcade games as multi-agent systems, in: World Conference on Information Systems and Technologies, Springer, 2020, pp. 369–379.

[S33] A. Yacoub, M. E. A. Hamri, C. Frydman, Dev-promela modeling, verification, and validation of a video game by combining model-checking and simulation, Simulation 96 (2020) 881–910.

[S34] C. Marín-Lora, M. Chover, J. M. Sotoca, A game logic specification proposal for 2d video games, in: World Conference on Information Systems and Technologies, Springer, 2020, pp. 494–504.

[S35] P. Mirza-Babaei, S. Stahlke, G. Wallner, A. Nova, A postmortem on playtesting exploring the impact of playtesting on the critical reception of video games, in: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, 2020, pp. 1–12.

[S36] B. D. Bryant, H. Saiedian, A state saturation attack against massively multiplayer online videogames., ICISSP (2021) 217–225.

[S37] X. Tong, Positioning game review as a crucial element of game user feedback in the ongoing development of independent video games, Computers in Human Behavior Reports 3 (2021) 100077.

[S38] B. J. Geisler, S. L. Kavage, A multi-engine aspect-oriented language with modeling integration for video game design, in: R. Ali, H. Kaindl, L. A. Maciaszek (Eds.), Evaluation of Novel Approaches to Software Engineering, Springer International Publishing, Cham, 2021, pp. 336–359.

[S39] C. Marín-Lora, M. Chover, J. M. Sotoca, A multi-agent specification for the tetris game, in: International Symposium on Distributed Computing and Artificial Intelligence, Springer, 2022, pp. 169–178.

[S40] S. Ding, N. Tang, T. Lin, S. Zhao, Rts-gameflow a new evaluation framework for rts games, in: 2009 International Conference on Computational Intelligence and Software Engineering, IEEE, 2009, pp. 1–4.

[S41] H. Istance, A. Hyrskykari, L. Immonen, S. Mansikkamaa, S. Vickers, Designing gaze gestures for gaming an investigation of performance, in: Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications, 2010, pp. 323–330.

[S42] J. L. González Sánchez, R. M. Gil Iranzo, F. L. Gutiérrez Vela, Enriching evaluation in video games, in: IFIP Conference on Human-Computer Interaction, Springer, 2011, pp. 519–522.

[S43] K. Jorgensen, Between the game system and the fictional world a study of computer game interfaces, Games and Culture 7 (2012) 142–163.

[S44] E. F. do Prado, D. Lucrédio, A flexible model-driven game development approach, in: 2015 IX Brazilian Symposium on Components, Architectures and Reuse Software, IEEE, 2015, pp. 130–139.

[S45] M. Kenzel, B. Kerbl, D. Schmalstieg, M. Steinberger, A high-performance software graphics pipeline architecture for the gpu, ACM Transactions on Graphics (TOG) 37 (2018) 1–15.

[S46] D. MacCormick, L. Zaman, Subvis the use of subjunctive visual programming environments for exploring alternatives in game development, in: Proceedings of the 14th International Conference on the Foundations of Digital Games, 2019, pp. 1–11.

[S47] W. K. Mizutani, F. Kon, Unlimited rulebook a reference architecture for economy mechanics in digital games, in: 2020 IEEE International Conference on Software Architecture (ICSA), IEEE, 2020, pp. 58–68.

[S48] X. Zhong, J. Xu, Game updates enhance players' engagement a case of dota2, in: 2021 4th International Conference on Information Management and Management Science, 2021, pp. 117–123.

References

[1] A. Ampatzoglou, I. Stamelos, Software engineering research for computer games: A systematic review, Information and Software Technology 52 (2010) 888–901. doi:https://doi.org/10.1016/j.infsof.2010.05.004.

[2] J. Chueca, J. Verón, J. Font, F. Pérez, C. Cetina, The consolidation of game software engineering: A systematic literature review of software engineering for industry-scale computer games, Information and Software Technology 165 (2024) 107330. URL: https://www.sciencedirect.com/science/article/pii/S0950584923001854. doi:https://doi.org/10.1016/j.infsof.2023.107330.

[3] C. Lewis, J. Whitehead, The whats and the whys of games and software engineering, in: Proceedings of the 1st International Workshop on Games and Software Engineering, GAS '11, Association for Computing Machinery, New York, NY, USA, 2011, p. 1–4. URL: https://doi.org/10.1145/1984674.1984676. doi:10.1145/1984674.1984676.

[4] J. Kasurinen, V. Palacin, E. Vanhala, What concerns game developers? a study on game development processes, sustainability and metrics, 2017, pp. 15–21. doi:10.1109/WETSoM.2017.3.

[5] J. Kasurinen, Games as software: Similarities and differences between the implementation projects, 2016, pp. 33–40. doi:10.1145/2983468.2983501.

[6] C. Ayala, B. Turhan, X. Franch, N. Juristo, Use and misuse of the term "experiment" in mining software repositories research, IEEE Transactions on Software Engineering 48 (2022) 4229–4248. doi:10.1109/TSE.2021.3113558.

[7] K.-J. Stol, B. Fitzgerald, The abc of software engineering research, ACM Transactions on Software Engineering and Methodology (TOSEM) 27 (2018) 1–51. doi:10.1145/3241743.

[8] J. S. Molléri, K. Petersen, E. Mendes, Cerse-catalog for empirical research in software engineering: A systematic mapping study, Information and Software Technology 105 (2019) 117–149. doi:10.1016/j.infsof.2018.08.008.

[9] D. Mendez, P. Avgeriou, M. Kalinowski, N. B. Ali (Eds.), Handbook on Teaching Empirical Software Engineering, Springer, New York, NY, USA, 2024.

[10] P. Rykała, The growth of the gaming industry in the context of creative industries, Biblioteka Regionalisty (2020) 124–136.

[11] C. Politowski, F. Petrillo, J. Montandon, M. Valente, Y.-G. Guéhéneuc, Are game engines software frameworks? a three-perspective study, Journal of Systems and Software 171 (2021) 110846. doi:10.1016/j.jss.2020.110846.

[12] T. Wijman, The games market and beyond in 2021: The year in numbers, https://newzoo.com/insights/articles/the-gamesmarket-in-2021-the-year-in-numbers-esports 2021. [Online; accessed 15-April-2024].

[13] T. Wijman, The games market in 2022: The year in numbers, https://newzoo.com/resources/blog/the-games-market-in-2022-the-year-in-numbers/, 2022. [Online; accessed 15-April-2024].

[14] SlashData, State of the developer nation 23rd edition, https://slashdata-website-cms.s3.amazonaws.com/sample_reports/dsIe6JlZge_KsHWt.pdf, 2022. [Online; accessed 15-April-2024].

[15] Unreal, Unreal game engine, https://www.unrealengine.com/, 2024. [Online; accessed 15-April-2024].

[16] Unity, Unity game engine, https://unity.com, 2024. [Online; accessed 15-April-2024].

[17] Unity, Unity visual scripting, https://unity.com/features/unity-visual-scripting, 2024. [Online; accessed 15-April-2024].

[18] Unreal, Unreal blueprint visual scripting, https://docs.unrealengine.com/4.27/en-US/ProgrammingAndScripting/Blueprints/, 2024. [Online; accessed 15-April-2024].

[19] R. K. Yin, Case study research and applications, volume 6, Sage Thousand Oaks, CA, 2018.

[20] P. Runeson, M. Höst, Guidelines for conducting and reporting case study research in software engineering, Empirical Software Engineering 14 (2009) 131–164. doi:10.1007/s10664-008-9102-8.

[21] J. Verner, J. Sampson, V. Tosic, N. A. Bakar, B. Kitchenham, Guidelines for industrially-based multiple case studies in software engineering, in: 2009 Third International Conference on Research Challenges in Information Science, 2009, pp. 313–324. doi:10.1109/RCIS.2009.5089295.

[22] P. Runeson, M. Host, A. Rainer, B. Regnell, Case study research in software engineering: Guidelines and examples, John Wiley & Sons, 2012.

[23] C. Wohlin, Case study research in software engineering—it is a case, and it is a study, but is it a case study?, Information and Software Technology 133 (2021) 106514. URL: https://www.sciencedirect.com/science/article/pii/S0950584921000033. doi:https://doi.org/10.1016/j.infsof.2021.106514.

[24] C. Wohlin, A. Rainer, Is it a case study?—a critical analysis and guidance, Journal of Systems and Software 192 (2022) 111395. URL: https://www.sciencedirect.com/science/article/pii/S0164121222001121. doi:https://doi.org/10.1016/j.jss.2022.111395.

[25] C. Zannier, G. Melnik, F. Maurer, On the success of empirical studies in the international conference on software engineering, in: Proceedings of the 28th International Conference on Software Engineering, ICSE '06, Association for Computing Machinery, New York, NY, USA, 2006, p. 341–350. URL: https://doi.org/10.1145/1134285.1134333. doi:10.1145/1134285.1134333.

[26] A. Bergen, A. While, A case for case studies: exploring the use of case study design in community nursing research, Journal of Advanced Nursing 31 (2000) 926–934. URL: https://onlinelibrary.wiley.com/doi/abs/10.1046/j.1365-2648.2000.01356.x. doi:https://doi.org/10.1046/j.1365-2648.2000.01356.x. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1046/j.1365-2648.2000.01356.x.

[27] V. R. Basili, R. W. Selby, D. H. Hutchens, Experimentation in software engineering, IEEE Transactions on Software Engineering SE-12 (1986) 733–743. doi:10.1109/TSE.1986.6312975.

[28] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, A. Wesslén, Experimentation in software engineering, Springer Science & Business Media, 2012.

[29] N. Juristo, A. M. Moreno, Basics of software engineering experimentation, Springer Science & Business Media, 2013.

[30] D. A. Ludwig, Use and misuse of p-values in designed and observational studies: Guide for researchers and reviewers, Aviation, Space, and Environmental Medicine 76 (2005) 675–680(6).

[31] A. D. Harris, J. C. McGregor, E. N. Perencevich, J. P. Furuno, J. Zhu, D. E. Peterson, J. Finkelstein, The Use and Interpretation of Quasi-Experimental Studies in Medical Informatics, Journal of the American Medical Informatics Association 13 (2006) 16–23. URL: https://doi.org/10.1197/jamia.M1749. doi:10.1197/jamia.M1749. arXiv:https://academic.oup.com/jamia/article-pdf/13/

[32] B. Kitchenham, S. Charters, Guidelines for performing systematic literature reviews in software engineering, Technical report, EBSE Technical Report EBSE-2007-01 (2007). URL: https://www.cs.auckland.ac.nz/~norsaremah/2007%20Guidelines%20for%20performing%20SLR%20in%20SE%20v2.3.pdf.

[33] K. Suresh, An overview of randomization techniques: an unbiased assessment of outcome in clinical research, Journal of human reproductive sciences 4 (2011) 8.

[34] M. Staron, Action research in software engineering, Springer, 2020.

[35] C. Wohlin, P. Runeson, Guiding the selection of research methodology in industry–academia collaboration in software engineering, Information and Software Technology 140 (2021) 106678. URL: https://www.sciencedirect.com/science/article/pii/S0950584921001361. doi:https://doi.org/10.1016/j.infsof.2021.106678.

[36] D. I. K. Sjoberg, T. Dyba, M. Jorgensen, The future of empirical methods in software engineering research, in: Future of Software Engineering (FOSE '07), 2007, pp. 358–378. doi:10.1109/FOSE.2007.30.

[37] D. Sjøberg, T. Dybå, B. Anda, J. Hannay, Building Theories in Software Engineering, 2008, pp. 312–336. doi:10.1007/978-1-84800-044-5_12.

[38] C. Robson, Small-scale evaluation: Principles and practice, Small-Scale Evaluation (01 2002).

[39] C. Wohlin, M. Höst, K. Henningsson, Empirical Research Methods in Software Engineering, Springer Berlin Heidelberg, Berlin, Heidelberg, 2003, pp. 7–23. URL: https://doi.org/10.1007/978-3-540-45143-3_2. doi:10.1007/978-3-540-45143-3_2.

[40] M. Solari, S. Vegas, N. Juristo, Content and structure of laboratory packages for software engineering experiments, Information and Software Technology 97 (2018) 64–79. URL: https://www.sciencedirect.com/science/article/pii/S0950584916304220. doi:https://doi.org/10.1016/j.infsof.2017.12.016.

[41] L. B. Christensen, B. Johnson, L. A. Turner, L. B. Christensen, Research methods, design, and analysis (2011).

[42] J. Schell, The Art of Game Design: A book of lenses, CRC press, 2008.

[43] X. Zhou, Y. Jin, H. Zhang, S. Li, X. Huang, A map of threats to validity of systematic literature reviews in software engineering, in: 2016 23rd Asia-Pacific Software Engineering Conference (APSEC), 2016, pp. 153–160. doi:10.1109/APSEC.2016.031.