2021 IEEE/ACM 13th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE) | 978-1-6654-1409-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/CHASE52884.2021.00020

Gamification in Software Engineering: A literature Review

Carlos Futino Barreto
CESAR School
Recife, Brazil
cfb@cesar.school

César França
Departamento de Computação
Universidade Federal Rural de Pernambuco
Recife, Brazil
cesar@franssa.com

***Abstract* — Given the impact of people motivation on software engineering, it is no surprise to see that gamification techniques have been studied and applied to the field. A relevant number of studies have been reported on the subject, and different understandings of the technique are around. However, it is not clear what gamification brings to industrial software development settings. This paper intends to map the state of the art on the use and understanding of gamification in industrial software development. We aim to review the definitions of gamification adopted in software engineering studies, identify the elements of games that are used to motivate software engineers, and map the specific areas of software engineering that have been addressed so far. For that end, we conducted a mixed method literature review, covering 130 studies. From such review, we found that (1) researchers in the field tend towards a strict view of gamification, (2) the practical results of gamification are unclear and polemic, and (3) this research area has still much to improve.**

Keywords—Motivation, engagement, gamification, software engineering

I. Introduction

Among the many human aspects that affect performance in Software Engineering, motivation is reported to have the largest impact on productivity [5][6]. As such, various strategies and techniques have been studied to enhance motivation. One such strategy is known as gamification, which generally refers to a set of techniques to make activities more "fun and engaging" 0 or "game-like" [2] by using "game elements" [3].

Some definitions of gamification can be found in literature. Deterding et al. [3] defines it as "*the use of game design elements in non-gaming contexts*", while Hamari and Sarsa [4] defines it as "*the concept of applying game-design thinking through the use of game mechanics to drive game-like player behavior to non-game applications"*. Taking a different approach, Werbach [2] understands gamifications as the process of making activities more game-like. That is, it covers practices that intend to replicate the kind of experience that is usually associated with games into any activities. On a similar note, although clearly focused on the marketing use of gamification, Huotari and Hamari [7] defines it as "*a process of enhancing a service with affordances for gamefull experiences in order to support user's overall value creation*".

This short paper intends to clarify how it has been defined and used in software engineering, and what are its known impacts, which is done through a combination of an adhoc and a systematic literature review. The following sections are divided as such: Section II describes the competing definitions of gamification, section III details the methodology of our study and section IV describes and discusses the results of the research. Finally, section 5 summarizes our conclusions and points to possible questions that might lead to future research.

II. Competing views of "Gamification"

The definitions mentioned in the introduction share some similarities and differences. Deterding et al. [3] and Hamari and Sarsa [4] focus on the application, being it an activity or an artifact, while Werbach [2] and Huotari and Hamari [7] seem to pay more attention to the process of changing a given practice. While definitions such as [3] and [4] seek precision, others seek flexibility [2][7]. As there are different definitions which look at the phenomenon through significantly different lenses, it is important to shed light on how these different views can affect our practical knowledge.

These definitions also differ on how they understand the importance of game elements. Deterding et al. [3] consider the game elements crucial, while Werbach [2] claims that *"game elements are only means to the end of gamification, but what matters is how those elements are selected, deployed, implemented, and integrated"*.

Another important distinction is that Deterding et al. [3] understands that gamification exists only when applied in non-gaming contexts, while Werbach [2] understands that, by defining gamification as a process, there is no need to struggle over whether the effort is applied to a non-game context. Another point where these definitions differ is on the subject of "serious games". Deterding et al. [3] differentiates gamification from serious games noting the latter describes the creation of "*full-fledged games for non-entertainment purposes*". On the other hand, Werbach [2] understands that defining gamification as a process eliminates the need to define the borders where gamification crosses over into serious games.

III. Gamification in Software Engineering

Gamifying software engineering is not an unheard-of idea. Several studies about gamification in software engineering have been reported, and even a few commercial tools have been released [8]. Gamification in software engineering has even been subject of several secondary studies too. Pedreira et al. [9], for instance, maps the most gamified software engineering areas and most used game elements in studies ranging from 2011 to 2014 and finds implementation to be the most studied In Belinazo and Fontoura [10], the authors also offer a review of game elements and studied software areas from 2012 to 2017. Software implementation appeared then as the most common process area being gamified.

2574-1837/21/$31.00 ©2021 IEEE
DOI 10.1109/CHASE52884.2021.00020

IV. Methods

From the mentioned papers, we felt a need for further discussion on the definitions and expected results of gamification. Therefore, we questioned (RQ1) How do studies in Software Engineering understand gamification? (RQ2) In which software contexts is gamification being applied? (RQ3) What gamification practices or game elements are being used? and (RQ4) What, if any, effect is gamification achieving in software engineering? Then, our search for literature was carried out in three steps, briefly explained in the following subsections.

A. Step 1: The Ad-hoc Review

First, we ran an ad-hoc search of papers, reading selected papers and following references. We started the research by reading a few selected papers on motivation and gamification, such as [5] and [11]. This step included the reading of a few secondary papers, such as reviews and systematic mappings [9][10]. For those, we also read some potentially useful references. We also ran free searches on known databases. The aim of this step was to help us to build a general understanding of the field of gamification and its application in software engineering, and to give us enough experience on the topic to plan the systematic review. Overall, in this step, we analysed 56 unique studies.

B. Step 2: The Systematic Research

In a second moment, we ran a systematic literature review based on Kitchenham and Charter's guidelines [35], which is a broadly accepted procedure in software engineering [36].

We divided the search string in two main components: "Software engineering" and "gamification". The first component was then broken into different terms that could reference to the different processes, while the second component was broken down into possible synonyms. The final search string was as follows: ("software engineering" OR "software process" OR "software requirements" OR "software planning" OR "software risk" OR "software configuration" OR "software design" OR "software construction" OR "software integration" OR "software maintenance" OR "software verification" OR "software validation" OR "software metrics" OR "software management" OR "software testing" OR "software implementation" OR "software development") AND (gamification" OR "gamifying" OR "gamify" OR "gamified" OR "gameful" OR "funware").

The selected data sources were the ACM Digital Library, IEEE Xplore and ScienceDirect. We decided to filter for studies published from 2018 to 2020, because there were already known secondary studies covering the previous years [9][10], from which we had dug references in the first step of our research. A total of 401 unique studies were retrieved from the academic databases. These studies were distributed as such: 226 studies from the ACM digital Library, 73 studies from IEEE Xplore and 102 studies from Science Direct.

From these, we excluded those that were wither not peer reviewed or book chapter, that did not focus on gamification in software engineering, that focused exclusively on teaching or training, and that focused on the development of serious games. We also excluded one paper because, while it was returned in the search it had been removed from the academic library. Furthermore, we excluded another paper because we were not able to access its digital version.

The whole selection process was conducted in pairs, and every conflict of opinion was discussed, and found consensual decision. After finishing the selection, we were left with 43 papers, distributed as such: 15 from the ACM Digital Library, 26 from IEEE Xplore and 2 from Science Direct.

C. Step 3: The Snowball

Finally, we ran a snowball study of the secondary papers found in the systematic review. We added to the study the primary resources of five secondary studies found in the systematic research . One among these studies, while claiming to have studied "*over 80*" primary studies, did not list its primary references. Thus, we were unable to follow these papers to our snowball research. The same selection criteria described in the previous section was followed in this step. As a result, 32 new unique papers were added to our study.

D. Data Analysis and Synthesis

With the four research questions in mind, our analysis and synthesis followed the Thematic Synthesis process described in Cruzes and Dyba [37]. First, we mapped the segments in the papers that provided answers to our questions. Then, we coded and organized those items into more general categories. Finally, we adopted the current version of SWEBOK guide [38] to organize the articles according to their software engineering area.

E. Threats to Validity

The non-systematic nature of the ad-hoc research makes it difficult to replicate an, as such, to validate. Although, it has been used with the aim to give us the necessary grounds to conduct the systematic review, most of the studies in our list come from that step. We have not looked specifically at the proceeding of more specialized conferences and publications, so we cannot assure that conferences that are not indexed in ACM DL and IEEE Xplore have been covered in our scrutinization process. Also, the secondary studies that we used as seed to find older studies may have addressed different issues and their search strings may not be fully compatible with ours, which was a risk we decided to take.

V. Results

At the end of the selection process, we had a total of 130 papers, divided as such: 56 from the ad-hoc study, 43 from the systematic study and 32 from the snowball. The full list of selected titles is available at http://bit.ly/396txCB. Unsurprisingly, the oldest studies included came from the ad-hoc research. The oldest study, [34], is from 1987. In contrast, there are 3 studies from 2020. The full distribution of studies by year of publication can be seen in Fig 1.

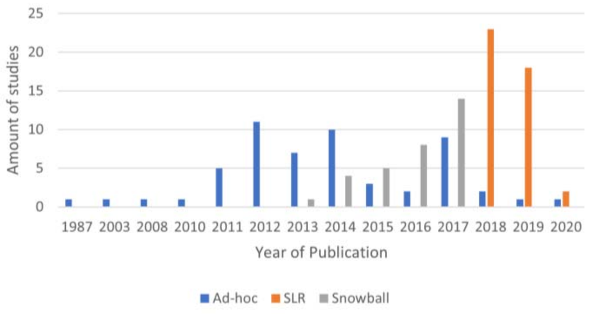

Fig 1 Studies by year of publication

A. *RQ1: How do studies in Software Engineering define gamification?*

Most studies (56%; 73/130) define gamification as the use, addition, or incorporation of "game elements" to the software engineering process, defining gamification more closely to Deterding [3]. However, only 46% (60/130) studies specifically mention the application of game elements in "non-game contexts". Furthermore, not always the term "game elements" is explicitly used. For instance, Melo [19], mentions "game metaphors", while 8 other papers mention "game mechanics".

In contrast, 11% of the studies (14/130) use terms referring to game thinking. These terms vary from actual "game thinking" [16] to "gamefullness" [17], or "theory of games" [18]. Among these, only 6 refer specifically to non-game environments.

Of the studied papers, 40 (31%) do not offer any definition of gamification, for they apparently consider the reader knows what the term may mean. Rodrigues et al. [13], Maarek et al. [14] and Marques et al. [15] claim gamification involves the development of a game or the transformation of an activity in a game. This puts those papers on the edge of what Deterding et al. [3] defines as "serious games".

B. *RQ2: In which software contexts is gamification being applied?*

Only 60% (79/130) studies looked at a specific area of software engineering. The areas with most studies were software construction, with 19 studies, and software requirements, with 17 (Fig. 2). On the other end, only Bounov et al. [22] focuses on computing fundamentals.

Among the remaining studies, 30% (39/130) do not focus on any particular software engineering area, including theoretical studies. França et al. [21] for example, focus on motivation in general software engineering. Furthermore, 10% (12/130) studies focus either on the proposition, or in the adoption of tools that can be used on all areas of software engineering.

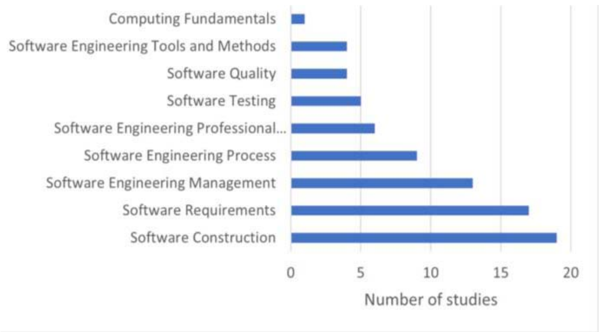

Fig 2 Studies per Software Engineering Area

C. *RQ3: What gamification practices or game elements are being used?*

We found that 42% (55/130) studies do not explicit the studied game elements. Among the gamification techniques discussed, the use of points was the one most mentioned, appearing in a total of 46% (60/130) studies. The use of badges and the use of leaderboards appear in 28% (36/130) and 27% (35/130) studies, respectively. This seems to be indicative of a strong tendency towards the most traditional, and simple, game practices. The use of what Chou [23] defines as PBLs is common in many gamification efforts. On the other extreme, only Pimentel et al. [25] mentions the use of luck. Fig 3. discloses all the game elements mapped.

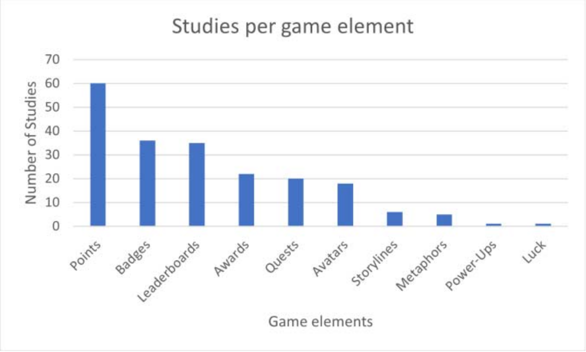

Fig 3 Studies per Game Element Used

D. *RQ4: What, if any, effect is gamification achieving?*

Among the 28% (36/130) studies reporting positive effects of gamification, 11.5% (15/130) reported improvements in behavioral aspects such as collaboration, motivation, or acceptance of new practices, for software engineers. Other 17% (23/130) studies reported technical benefits, such as performance, product quality, and learning. We found only 3% (4/130) studies that reported negative effects, like increased perceived difficulty of the tasks [28] or undermined satisfaction [29]. Grant and Betts, for example, [30] reported that participants tended to contribute less after badges are acquired. Four other studies (3%) found no relevant effect on the application of gamification, while Sammut et al. [32] pointed out to controversial effects of badges, being positively evaluated by developers but negatively evaluated by managers.

On the other hand, an impressive number of 72% (94/130) have not reported any practical results. Other studies such as McGregor [26] were in their initial stages of research and, although promising, still did not have significant results to present.

VI. Discussion and Conclusions

This study intended to study how software engineering researchers understand gamification, and what practical results they have been reaching so far. After reviewing 130 studies, we found that this field tends towards points, badges and leaderboards as game elements. These findings seem to point to a strict view of gamification in the area, while there are much more sophisticated views for this phenomenon ([7][23]) that have been systematically ignored in this area.

Deterding's definition [3] on the "*use of game elements in non-gaming environments*" seems to be the most popular understanding in software engineering. However, in fact, most studies do not even discuss the actual phenomenon under investigation. Overall, there is a general lack of theoretical underpinnings, which is a clear weakness in this area.

As a result, although gamification is expected to produce positive results in both the behavioral and technical dimensions, the available evidence is scarce, and even negative impacts have been reported. We also felt a lack of depth on the discussion of these effects. Motivation research show [21], for example, that some effects may represent short-term improvements, at the cost of long-term performance.

As expected, software construction is still the most investigated area among the specific areas of study, in line with the data from previous literature review studies [9][10]. However, our data show that gamification practices are also spreading to many other areas of software engineering.

In a future iterations of this research, we plan to integrate more studies in the analysis, and to run a Qualitative Metasummary [39] of the literature in order to cross the data in order to synthesize evidence for what gamification practices have been more or less responsible for each evidenced effect, if possible.

References

[1] A. Dorling and F. Mc Caffery, “The Gamification of SPICE.” Accessed: Nov. 15, 2019. [Online].

[2] K. Werbach, “(Re) Defining Gamification : A Process Approach Gamification as a Process,” pp. 266–272, 2014.

[3] S. Deterding, D. Dixon, R. Khaled, and L. Nacke, “From game design elements to gamefulness: Defining ‘gamification,’” in Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments, MindTrek 2011, 2011, pp. 9–15, doi: 10.1145/2181037.2181040.

[4] J. Hamari and H. Sarsa, “Does Gamification Work ? — A Literature Review of Empirical Studies on Gamification,” 2014.

[5] C. Franca, F. Q. B. da Silva, and H. Sharp, “Motivation and Satisfaction of Software Engineers,” IEEE Transactions on Software Engineering, vol. 46, no. 2, Feb. 2020, doi: 10.1109/TSE.2018.2842201.

[6] T. Hall, N. Baddoo, S. Beecham, H. Robinson, and H. Sharp, “A systematic review of theory use in studies investigating the motivations of software engineers,” ACM Transactions on Software Engineering and Methodology, vol. 18, no. 3, May 2009, doi: 10.1145/1525880.1525883.

[7] K. Huotari and J. Hamari, “Defining gamification,” 2012, doi: 10.1145/2393132.2393137.

[8] E. B. Passos, I. S. Inf, and P. A. S. Neto, “Turning Real-World Software Development into a Game,” pp. 1–8, 2011.

[9] O. Pedreira, F. García, N. Brisaboa, and M. Piattini, “Gamification in software engineering - A systematic mapping,” in Information and Software Technology, 2015, vol. 57, no. 1, pp. 157–168, doi: 10.1016/j.infsof.2014.08.007.

[10] G. Belinazo and L. M. Fontoura, “Mapeamento Sistemático de Gamificação em Engenharia de Software,” no. Eres, pp. 191–200, 2017.

[11] S. Deterding, “Gamification : Designing for motivation Gamification : Designing for Motivation,” no. June, 2014, doi: 10.1145/2212877.2212883.

[12] N. Unkelos-Shpigel and I. Hadar, “Leveraging motivational theories for designing gamification for RE,” Proceedings - International Conference on Software Engineering, pp. 69–72, 2018, doi: 10.1145/3195836.3195843.

[13] L. F. Rodrigues, C. J. Costa, and A. Oliveira, “Gamification: A framework for designing software in e-banking,” Computers in Human Behavior, vol. 62, Sep. 2016, doi: 10.1016/j.chb.2016.04.035.

[14] M. Maarek, L. McGregor, S. Louchart, and R. McMenemy, “How Could Serious Games Support Secure Programming? Designing a Study Replication and Intervention,” Jun. 2019, doi: 10.1109/EuroSPW.2019.00022.

[15] R. Marques, G. Costa, M. Mira da Silva, and P. Goncalves, “Gamifying software development scrum projects,” Sep. 2017, doi: 10.1109/VS-GAMES.2017.8056584.

[16] D. Ašeriškis and R. Damaševičius, “Gamification Patterns for Gamification Applications,” Procedia Computer Science, vol. 39, 2014, doi: 10.1016/j.procs.2014.11.013.

[17] F. García, O. Pedreira, M. Piattini, A. Cerdeira-Pena, and M. Penabad, “A framework for gamification in software engineering,” Journal of Systems and Software, vol. 132, Oct. 2017, doi: 10.1016/j.jss.2017.06.021.

[18] Ç. Üsfekes, E. Tüzün, M. Yılmaz, Y. Macit, and P. Clarke, “Auction‑based serious game for bug tracking,” IET Software, vol. 13, no. 5, Oct. 2019, doi: 10.1049/iet-sen.2018.5144.

[19] A. de Melo et al., “LNCS 8531 - Version Control System Gamification: A Proposal to Encourage the Engagement of Developers to Collaborate in Software Projects.”

[20] R. C. Callan, K. N. Bauer, and R. N. Landers, “How to avoid the dark side of gamification: Ten business scenarios and their unintended consequences,” in Gamification in Education and Business, Springer International Publishing, 2015, pp. 553–568.

[21] A. C. C. França, T. B. Gouveia, P. C. F. Santos, C. A. Santana, and F. Q. B. Silva, “Motivation in Software Engineering : A Systematic Review Update,” pp. 154–163, 2011.

[22] D. Bounov, A. DeRossi, M. Menarini, W. G. Griswold, and S. Lerner, “Inferring Loop Invariants through Gamification,” Apr. 2018, doi: 10.1145/3173574.3173805.

[23] Y. Chou, Actionable gamification: Beyond points, badges, and leaderboards. Packt Publishing Ltd, 2019.

[24] D. Bounov, A. DeRossi, M. Menarini, W. G. Griswold, and S. Lerner, “Inferring Loop Invariants through Gamification,” Apr. 2018, doi: 10.1145/3173574.3173805.

[25] J. Pimentel, E. Santos, T. Pereira, D. Ferreira, and J. Castro, “A gamified requirements inspection process for goal models,” Apr. 2018, doi: 10.1145/3167132.3167272.

[26] L. McGregor, “Gamification and Collaboration to Evaluate and Improve the Security Mindset of Developers,” Jul. 2019, doi: 10.1145/3304221.3325593.

[27] D. Ašeriškis and R. Damasevicius, “Gamification of a Project Management System,” Jan. 2014.

[28] R. Snijders, F. Dalpiaz, M. Hosseini, and R. Ali, “REfine : A Gamified Platform for Participatory Requirements Engineering,” pp. 1–6, 2015.

[29] C. R. Prause and M. Jarke, “Gamification for enforcing coding conventions,” Aug. 2015, doi: 10.1145/2786805.2786806.

[30] S. Grant and B. Betts, “Encouraging User Behaviour with Achievements: An Empirical Study.” Accessed: Nov. 15, 2019. [Online]. Available: http://www.xbox.com/.

[31] J. Thom, D. Millen, and J. DiMicco, “Removing gamification from an enterprise SNS,” 2012, doi: 10.1145/2145204.2145362.

[32] R. Sammut, D. Seychell, and N. Attard, “Gamification of Project Management within a Corporate Environment An Exploratory Study,” pp. 1–2, 2014, doi: 10.1145/2181037.2181040.

[33] Cruzes, Daniela S., and Tore Dyba. "Recommended steps for thematic synthesis in software engineering." 2011 international symposium on empirical software engineering and measurement. IEEE, 2011.

[34] Boehm, Barry W, “Improving Software Productivity”, 1987, doi: https://doi.org/10.1109/MC.1987.1663694

[35] B. Kitchenham, and S. Charters. Guidelines for performing systematic literature reviews in software engineering. 2007.

[36] F. Q. Da Silva, A. L.Santos, S. Soares, C. França, C. Monteiro, F. Maciel. Six years of systematic literature reviews in software engineering: An updated tertiary study. Information and Software Technology, 2011, 53(9), 899-913.

[37] D. S. Cruzes and T. Dyba. "Recommended steps for thematic synthesis in software engineering." 2011 international symposium on empirical software engineering and measurement. IEEE, 2011.

[38] P. Bourque, R. Dupuis, A. Abran, J. W. Moore, and L. Tripp (1999). The guide to the software engineering body of knowledge. IEEE software, 16(6), 35-44.

[39] D. M. Ribeiro, M. Cardoso, F. Q. da Silva, and C. França (2014). Using qualitative metasummary to synthesize empirical findings in literature reviews. In Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (pp. 1-4).