2021 Second International Conference on Information Systems and Software Technologies (ICI2ST) | 978-1-6654-0411-2/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICI2ST51859.2021.00025

Empirical Strategies in Software Engineering Research: A Literature Survey

Cathy Guevara-Vega
Faculty of Engineering in Application Science, Group e-CIER
Universidad Técnica del Norte
Ibarra, Ecuador
SCORE Lab
Universidad de Sevilla
Sevilla, Spain
0000-0002-2470-8287

Beatriz Bernárdez
I3US Institute
Universidad de Sevilla
Sevilla, Spain
0000-0002-9390-3772

Amador Durán
SCORE Lab, I3US Institute
Universidad de Sevilla
Sevilla, Spain
0000-0003-3630-5511

Antonio Quiña-Mera
Faculty of Engineering in Application Science, Group e-CIER
Universidad Técnica del Norte
Ibarra, Ecuador
SCORE Lab
Universidad de Sevilla
Sevilla, Spain
0000-0003-2516-9016

Margarita Cruz
I3US Institute
Universidad de Sevilla
Sevilla, Spain
0000-0001-8334-6039

Antonio Ruiz-Cortés
SCORE Lab, I3US Institute
Universidad de Sevilla
Sevilla, Spain
0000-0001-9827-1834

***Abstract*—Context: The Software Engineering (SE) research continues to gain strength and interest for researchers considering the need to apply rigor and scientific validity to research results. Objective: Establishing an overview of the topic through a classification scheme of publications and structure the field of interest. Method: We conducted a Systematic Mapping Study, including articles published until 2019, that report at least one study of empirical strategies in SE. Results: 80 initial sets of studies were selected and analyzed, identifying: i) empirical strategy type used and ii) Software Engineering hypotheses types used. Also, 20 papers of the set of studies for mapping were selected and analyzed, identifying 17 empirical strategies and 11 main characteristics to address the empirical research inception in SE. Conclusions: We corroborate that the selection of an empirical strategy in Software Engineering research depends on the nature and scope of the research and on the resources that the researcher has at that moment, in addition to the degree of scientific and methodological knowledge that he has to carry out an empirical study. It is necessary to continue studying in-depth the behavior and nature of the empirical strategies in Software Engineering research that allows strengthening the scientific taxonomy in SE, besides walking towards the automation of the experimental process.**

***Index Terms*—empirical strategy, software engineering, systematic mapping study, empirical software engineering.**

I. Introduction

The research methodology in the last decade has been of interest to researchers in Software Engineering (SE) due to the need to give scientific validity and rigor to the results obtained in the study and analysis of research work [1]. Besides, the need to evaluate and validate new research proposals utilizing empirical studies in SE is recognized today to a greater extent than ten years ago [2]. The choice of methodology, strategy, and research method should not be considered “good” or “bad” a priori because this depends mostly on the establishment, field, approach, and objective of the research [3]; Software Engineering is a laboratory science [4].

From the conception of the researcher’s thought, there is a selection bias, which adds the degree of scientific and methodological knowledge of carrying out an empirical study [5]. Furthermore, it is influenced by the availability of alternatives for the researcher, such as the accessibility of data sources, the researcher’s knowledge and experience about the methods and methodology of the research that will influence the design of the study [6]. For SE, it is necessary to distinguish between two types of scientific knowledge: science and technology, knowledge for understanding and knowledge for a purpose [7]. Starting from the fact that the scientific approach typically consists of observation, measurement, and experimentation [8].

In this sense, researchers in SE have adopted numerous methods, approaches, and empirical strategies from other disciplines such as psychology, medicine, and biotechnology that allow us to narrow the gap in the variety of terminology that makes it difficult to establish a clear and concise taxonomy in the SE [9]; it is necessary to do more empirical research and evidence-based practice [10]. In this sense, Harrison and Basili in 1996 explained that Empirical Software Engineering (ESE) is the area of research that emphasizes the use of the empirical method in the field of Software Engineering [11].

The motivation of the present study is to strengthen the formality, standardization, and automation of the methodolog-

978-1-6654-0411-2/21/$31.00 ©2021 IEEE
DOI 10.1109/ICI2ST51859.2021.00025

ical process in empirical research for SE. In this context, we established three research questions: RQ1: What are the empirical strategies used in software engineering?. RQ2: What kind of hypotheses are handled in the empirical strategies?. RQ3: What are the main characteristics to address the empirical research inception in SE?.

The objective paper is conduct a systematic mapping study (SMS) of Empirical Strategies used in Software Engineering to establish an overview of the topic through a classification scheme of publications and structure the field of interest.

The rest of the paper is structured as follows: Section 2) Background: We establish the research paradigm for empirical study and describe the literature review methods. Section 3) Research Method: Conducting a Systematic Mapping Study (SMS). Section 4) Results: Results obtained from the SMS. Section 5) Discussion and conclusions of the study. Section 6) Appendix: presents the boundary documents included and excluded from the work.

II. Background

A. Research Paradigms for Empirical Studies

The basis of empirical research is the observation of phenomena and the capture of experiences, so it is essential to plan the steps to conduct their studies [12]; this will allow the researcher to solve problems or obstacles that may arise during the experimental process. Besides, the researcher must be aware of the potential threats to his or her study and justify their findings [6]. In this sense, we consider explaining three elements that the researcher must know: 1) *The analysis unit*, is the critical object analyzed in the empirical research. It can be an artifact, a project, a specific role of a person, a group of people, or an organization. The analysis unit's decision is critical since it dictates the research question that tends to be integrated into the research methods [2]. 2) *Sampling*, is the process of selecting what portion of the population to analyze. There are several sampling strategies that a researcher can adopt, such as random sampling and snowballing. Researchers must understand the sampling process well before beginning to collect their data [13]. 3) *Reliability and validity*, are considered at each step of the research design decision points; this may include the design of a questionnaire or interviews, experimental design, sampling strategy, or how the data are analyzed. Carrying out empirical research without considering its reliability and validity does not make sense because the researcher will not generalize from the results [14].

There are several researchers' criteria about the types of research paradigms for empirical studies, from a philosophical approach Tayler and Medina in 2007 [15], Creswell and Poth in 2018 [16] and Ahrens and Zaščerinska [17] explain the following:

a) Positivist paradigm: Which tests theories or hypotheses.

b) Post-positivist paradigm: Analyzes the interaction between the researcher and his/her research participants through quantitative methods such as survey research and qualitative methods such as interviews and participant observation.

c) Interpretive paradigm: Understands other cultures from the inside through ethnographic methods such as informal interviews, participants, observation, and establishment of ethically healthy relationships.

d) Critical paradigm: Involves identifying and transforming socially unjust social structures, policies, beliefs, and practices.

e) Integral paradigm: Provides to design new hybrid methodologies that involve multiple epistemologies and their accompanying quality standards.

For Wohlin *et al.*, in 2012, there are two types of research paradigms according to their scope for empirical studies such as [2]:

f) Exploratory research: Is concerned with studying objects in their natural environment and letting the findings emerge from observations; this implies that a flexible research design is needed [18] to adapt to changes in the observed phenomenon. Flexible design research is also known as qualitative research since it is based primarily on qualitative data. Inductive research attempts to interpret a phenomenon based on explanations presented by people are concerned with discovering the causes observed by the subjects in the study and understanding their view of the problem in question. The subject is the person who participates in an empirical study to evaluate an object.

g) Explanatory research: Mainly concerned with quantifying a relationship or comparing two or more groups to identify a cause-and-effect relationship, is often carried out through controlled experiments. This study is of fixed design [18], so the factors are fixed before the study is launched. Fixed design research is also known as quantitative research since it is based mainly on quantitative data. Quantitative research is appropriate when the effect of some manipulation or activity is tested. Descriptive research is always at the base of explanatory research. The causal hypothesis cannot be formulated if the problem is not described in depth.

In general terms, from the moment a scientific problem is identified until a solution is found, the research around it goes through an exploratory, descriptive, and explanatory phase. For this reason, it may be recommended that when approaching a research topic, researchers should consider in what phase of their knowledge they are and take into account this approach to determine lines and forms of action [19].

B. Empirical Research Methods

As well as the types of paradigms, there are various criteria used by researchers to define research methods for empirical studies, according to Jiménez R. in 2018, who establishes the following [19]:

a) Inductive method: Reasoning that makes it possible to obtain general conclusions from particular facts.

b) Deductive method: Reasoning that makes it possible to establish predictions based on the general to explain the particular.

c) Hypothetical-deductive method: based on hypotheses that demonstrate or refute in order to conclude. It defines steps as posing the problem, creating hypotheses, deducing consequences from the hypothesis, and contrasting them (if it is false, it returns to creating hypotheses; otherwise, it is verified, and the Law is established).

According to Gómez & Reidl, in 2010, they established the following methods of empirical research [20].

d) Observation: Is a method of collecting information systematically, valid, reliable, and intentional.

e) Measurement: Method based on the collection of quantitative data indicating regularities.

f) Experimentation: Part of observations, and models or theories, to formulate hypotheses that are confirmed or denied by repeated testing of the behavior of elements of the phenomenon and their relationships.

According to Sandage *et al.* in 2012, they establish the following theoretical-empirical methods of scientific research [21]:

g) Hermeneutic method: Studies the symbolic construction, subjects' interpretations, internal coherence of texts, and discourses.

h) Dialectical method: Considers the relationship of the problems with the historical and social context, conceives the phenomena in movement, everything changes and is in continuous evolution.

C. Empirical Strategies in Software Engineering

Depending on the research approaches, justification, methods, techniques, instruments, procedures, tools, the purpose of the evaluation, and the conditions for empirical research [18], there are several types of strategies used in Software Engineering. We observe that the recurrent methods are case study, survey, experiments, action research, which we describe below:

a) Case study: Is a suitable research methodology for software engineering research since it studies contemporary phenomena in its natural context within its real-life setting [22]. The information is gathering is from a few entities (people, groups, organizations), and there is a lack of experimental control [23]. The "unit of analysis" maybe some aspect of a software engineering project, a software engineering methodology and its use within an organization, a software engineering section of an organization, or the whole or a particular part of a new or ongoing development or maintenance project and its will vary depending on the research question [24].

b) Survey: This is a comprehensive process of information collection to describe, compare, and explain knowledge, attitudes, and behavior, and has a high degree of external validity [25]. Wohlin *et al.* in 2012 [2] list three types of surveys: descriptive, explanatory, and exploratory. The surveys are used as a pre-survey to determine the opportunities and risks for empirical research [26]. The survey can be applied to individuals through questionnaires and to literature bases such as Scopus, IEEE, and MCL through methods such as Systematic Mapping Study (SMS) or Systematic Literature Review (SLR) [27]. Whatever the case, it is vital to consider three aspects: knowing the scope and the resources available to the researcher to carry out the survey, selecting the parameters of interest to be examined, and identifying the sources of information needed, whether in the case of individuals or literature databases [28].

c) Experiment: Applied to control and manipulate the situation in a direct, precise, and systematic way [2], that is, where at least one treatment or controlled variable exists [3]. In SE, they are defined as controlled experiments and applied in laboratories with humans and software [29]. Several types of experiments consider the attributes defined in the research; for example, if the study presents descriptive, correlational, cause-effect results, most cause-effect studies are done with a human factor [3] if the study is done with novices, experts, or both. Suppose the study is conducted in a controlled laboratory (in vitro) or the field under normal conditions (in vivo) [30] furthermore if the study is observational [3]. Quasi-experiments are easily performed in vivo with experts. These experiments usually include a component of qualitative analysis, including at least some form of an interview. Besides, there is a lack of randomization of both subjects and objects [18].

d) Action research: Researchers conduct studies as participants in real-world projects; there is limited control over the process and the results [31]. Relevance to practice is ensured by its empirical basis and purposeful interactions with practitioners [32]. Furthermore, it focuses on change; that is, the researcher actively participates in the improvement of the software process [2].

D. Hypothesis Types in Software Engineering

Hypotheses are formulated when the research wants to test an assumption and show characteristic features of a particular situation, responding to a problem or research question [33]. Fundamentally, it reveals the cause-and-effect relationship in an investigation. Whether or not we formulate hypotheses depends on two essential factors: the study's focus and its initial scope [34].

In 2001, Zendler proposed four types of software engineering hypotheses: (1) software engineering guesses (based neither on back-ground knowledge nor on the empirical test but predominant the speculation and the early stage of theoretical work. (2) empirical software engineering hypotheses (not based on software engineering antecedent knowledge but are empirically tested). (3) plausible software engineering hypotheses (based on software engineering antecedent knowledge but are not empirically tested). (4) corroborated software engineering hypotheses (based on software engineering antecedent knowledge and are empirically tested) [35].

E. Literature Review

There are two agreed approaches to carry out literature review, SMS and SLR.

a) Systematic mapping study (SMS): Is a methodology that provides a visual summary, a map, of the research about a topic, by structuring and classifying the type of reports and research results. It provides a more comprehensive overview [36].

b) Systematic literature review (SLR): Provides a synthesis of existing knowledge from primary sources of a specific field by understanding and classifying their content. It focuses on "how" things are done in the field of study, focusing on a small set of documents from which as much knowledge as possible is extracted [37].

Given both approaches, we identified the need to conduct a systematic mapping study that classifies the existing proposals or taxonomies that establish empirical strategies, to provide an overview of the approaches, type of research, type of hypotheses, and characteristics of existing proposals [53].

III. Research Method

A. SMS process definition

The structure of the SMS was based on Petersen's guide [38]; Fig. 1 summarizes the SMS process.

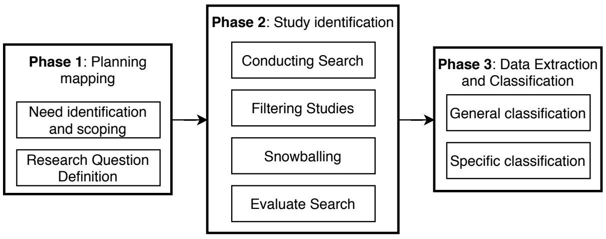
Phase 1: Planning mapping
Need identification and scoping
Research Question Definition
Phase 2: Study identification
Conducting Search
Filtering Studies
Snowballing
Evaluate Search
Phase 3: Data Extraction and Classification
General classification
Specific classification

Fig. 1. SMS Process.

B. Phase1: planning the mapping

a) Identification of need and scope: The SMS provides an overview of the proposals about existing empirical software engineering strategies to classify their main characteristics as types of research, types of hypotheses, and approaches. Therefore we established the following research questions.

- RQ1: What are the empirical strategies used in software engineering?
- RQ2: What kind of hypotheses are handled in the empirical strategies?
- RQ3: What are the main characteristics to address the empirical research inception in SE?

C. Phase2: Study identification

Below we detail the steps to identify the primary studies used in the SMS.

a) Search Strategies: We used two search strategies: a database search and Snowballing, which is to search for new primary studies in the references of the studies found with the database search strategy.

The search process conducts as follows: Step 1) Conducting search the scientific databases. Then join the studies found and eliminate the repeated studies. Step 2) Filter the studies by applying inclusion and exclusion criteria. Step 3) Apply the snowball strategy to include quality primary studies to the set of studies for the SMS. Step 4) Evaluate the search by comparing the search result with a set of relevant studies known to researchers in the field.

b) Development of the search: We conduct the database search applying the following search string: TITLE-ABS-KEY (selecting AND empirical AND methods AND for AND software AND engineering AND research). The scientific databases chosen were Scopus, IEEE Explore (IEEE), ACM Digital Library (ACM), Web of Science (WoS); besides, we selected the specialized Journal Empirical Software Engineering (EMSE), due to the relevance of the subject of study. We include studies up to 2019 and do not use an initial date. In the initial search, we found 333 studies, and after eliminating the duplicate studies, 80 studies remained. Then in the filtering process, there were 8 studies. Furthermore, after applying the snowballing strategy, it was possible to retrieve 11 studies relevant to SMS's set of primary studies, see Fig. 2.

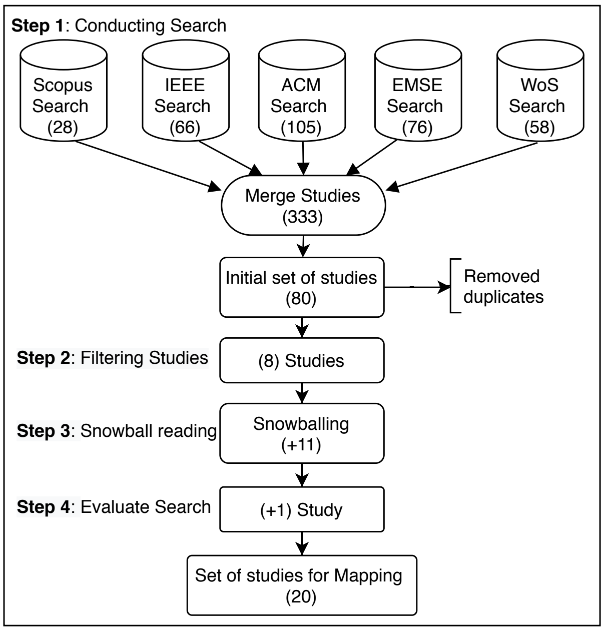

Fig. 2. SMS Search Process.

c) Filtering Studies: We defined the following inclusion and exclusion criteria to be applied to the set of initial studies established in Step 1 of the search. With this step, we identified the appropriate and relevant studies for mapping in the full text of the studies.

Inclusion criteria:

- Publications in the English language.
- Peer-reviewed studies in journals or conferences
- Studies that propose a description, classification, or taxonomy of the empirical strategies applied in software engineering.

- Studies containing an adequate research structure as context, objective, research method, and results.

Exclusion criteria:

- Studies that only refer to empirical strategies in software engineering.
- Non-peer-reviewed studies, books, book chapters, or gray literature.
- Studies that are not in English.
- Studies without access.

d) Snowballing: We applied the snowballing search strategy to add quality primary studies to the SMS. This strategy consisted of reviewing the references for the studies established in step 2 of the filtering process. We recovered 8 studies, plus the 11 studies of the initial set, leaving 19 studies.

e) Evaluate of search: To assess the quality of the set of primary studies found, we compared them with the following set of relevant studies known to the researchers: Stol and Fitzgerald [39],Wohlin and Aurum [6], Easterbrook *et al.* [40], Perry *et al.* [10], Wieringa [41], Zelkowitz and Wallace [42], Glass *et al.* [43], Höfer and Tichy [44], Sjøberg *et al.* [45], Zannier *et al.* [46], Jain *et al.* [47], Dybå *et al.* [48], Carvalho João [49]. After the comparison, we recovered the study of Professor Carvalho João [49], leaving a set of 20 primary studies for SMS.

D. Phase3: Data extraction and classification

a) General classification: The fields for the general classification of primary studies were: author, year, keywords, abstract, forums, affiliation.

b) Specific classification: The fields for the specific classification were: empirical strategy type, Software Engineering hypotheses types, and main characteristics of the empirical strategies used.

IV. Results

A. RQ1: What are the empirical strategies used in software engineering?

Based on the 20 primary studies found in the SMS, we observe that the researchers propose multiple criteria to establish SE's empirical strategies. We identified 17 empirical strategies, see Table I. Furthermore, we observe that Controlled experiment (human/software) stands out and is commonly used by researchers with 95%, followed by Case study with 90% and Survey with 85%, see Fig. 3.

B. RQ2: What kind of hypotheses are handled in the empirical strategies?

To identify the hypotheses used in the empirical strategies, we used the initial set of 80 primary studies. We classified them by the type of hypotheses proposed by Zendler [35]. We observe that most of the studies used the Empirical Software Engineering Hypothesis type (51.25%), see Fig. 4.

TABLE I
Empirical Strategies in Software Engineering

Empirical Strategy	***Primary studies***
Controlled experiment (human/software)	[6], [8], [10], [12], [31], [34], [39], [40], [41], [42], [43], [44], [45], [46], [48], [49], [50], [51], [52]
Case study (restrospective / post-morten, current / non-intrusive, participatory) (in situ)	[6], [8], [10], [12], [31], [39], [40], [41], [42], [43], [44], [45], [46], [47], [48], [49], [50], [51]
Survey (descriptive and exploratory)	[6], [8], [12], [31], [34], [39], [40], [41], [43], [44], [45], [46], [47], [48], [49], [51], [52]
Quasi-experiment	[6], [8], [12], [34], [39], [40], [45], [46], [48], [49]
Action research	[6], [12], [39], [40], [41], [43], [45], [48], [49], [50]
Ethnography	[12], [39], [40], [43], [44], [48], [49], [50], [52]
Computational simulation (in silico)	[6], [8], [31], [39], [42], [43], [49]
Secondary-data studies	[12], [31], [41], [42], [43], [45], [49]
Simulated experiment	[6], [31], [39], [42], [43], [49]
Meta-analysis	[10], [43], [44], [46], [48]
Phenomenology	[43], [44], [49], [50]
Grounded theory	[6], [10], [43], [50]
Hermeneutics	[6], [43], [50]
Archive research	[6], [49]
Delphic study	[39], [49]
Focus group (with a panel of specialists and with key informats)	[39], [49]
Conceptual analysis	[43]

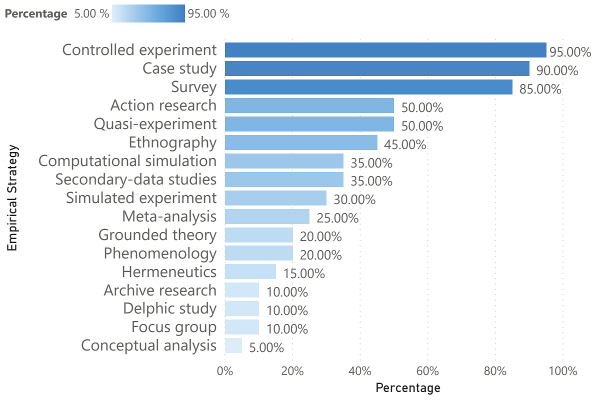

Fig. 3. Empirical strategies used in studies.

C. RQ3: What are the main characteristics to address the empirical research inception in SE?

Based on the 20 primary studies found in the SMS, we have identified a set of good practices to address the empirical research inception, which will be studied/analyzed before choosing the empirical strategy, see Table II.

V. Discussion and Conclusions

This paper establishes a literary overview of the empirical strategies in Software Engineering research through a systematic mapping study. We consider discussing our research with two related studies on the need that researchers have when

TABLE II
CHARACTERISTICS TO ADDRESS THE EMPIRICAL RESEARCH INCEPTION IN SOFTWARE ENGINEERING

Characteristic	*Rationale*	*Classification*
General environment [49].	The initial vision of the research.	1) Study, 2) Experiment
Specific environment [49].	The environment where the research will be conducted.	1) Field, 2) Library, archive, 3) Laboratory.
Study type [39].	General scheme defines the process and procedure of activities undertaken to solve problems and answer research questions.	1) Field study, 2) Judgment study (to interview study), 3) Sample studies (i. e., to interview, questionnaires), 4) Field experiments, 5) Formal theory, 6) Laboratory experiment, 7) Experimental simulation.
Metaphor and Setting [39].	Direct comparison the relationship of similarity or analogy between the place and the real environment where the behavior of the phenomenon is established.	1) Jungle (to Field study), 2) Courtroom (to Judgment study), 3) Referendum (to Sample studies), 4) Nature reserve (to Field experiment), 5) Jigsaw puzzle (Formal theory), 6) Cleanroom, test tube (to Laboratory experiment), 7) Greenhouse, flight simulator (to Experimental simulation).
Exploratory questions to understand the phenomenon [6], [40], [48].	Basic questions about the normal patterns of occurrence of the phenomena within the study.	1) "what's going on" (to Field study), 2) "How", "Why", "Which is better" (to Field study), 3) "how things work" (to Judgment study), 4) "what, how much, and how many, as well as questions about how and why" (to Sample study), 5) "What if" (to Laboratory experiment).
Research outcome [6].	It is based on the types of final results obtained in the research.	1) Basic research, 2) Applied research.
Research design [18].	Design is concerned with turning research questions into projects [18]. Establish various things and resources which should be thought about and kept in mind when carrying out a research project.	1) Fixed design, 2) Flexible design.
Research Logic [6].	Based on the direction, the research is conducted, i.e., whether it goes from the general to the specific or vice versa.	1) Inductive research, 2) Deductive research.
Research Purpose [6].	Based on the scope and limitation that the research will have.	1) Exploratory, 2) Descriptive, 3) Explanatory, 4) Evaluation.
Research approach from a philosophical perspective [40].	Philosophical stance of knowledge that leads to logical inference from a set of basic observable facts within the research.	1) Positivitism, 2) Constructivism, 3) Critical Theory, 4) Pragmatism.
Research according to the nature of the data [6].	The natural way the data behaves.	1) Cualitative, 2) Cuantitative, 3) Mixed Approach.

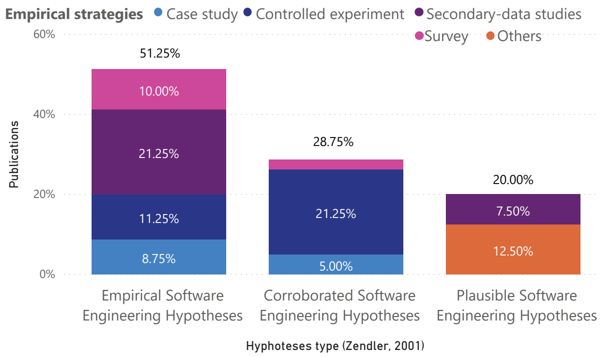

Fig. 4. Publications by Hyphoteses and Empirical Strategies.

selecting the type of study and the decisions that should be made during the research design. The first study is by Stol *et al*. in 2018 [39], who make an essential contribution through a proposal of an ABC framework for SE research that addresses behavior, scenarios, and a holistic view of eight archetypal research strategies. The second is by Wohlin and Aurum [6] in 2015; they propose a decision-making structure that contains some decision points to help researchers in the starting point for the research design before going into the research details design chosen. In a discussion with our study, we classified into 17 empirical strategies based on 20 primary studies resulting from our SMS, this we report in Table I. We also reported the current trend of application of these strategies. We synthesized 11 main characteristics that the researcher should consider when empirical research inception in Software Engineering, this we report in Table II. As conclusions, we establish the following: We observe that the researchers in Software Engineering continue applying in their majority three empirical strategies: Controlled experiment, Case study, and Survey, respectively, this we report in Fig. 3. These three strategies appear in the great majority of the proposals on classifications, which shows their importance with respect for the rest (which appear in at most 50%). We identified the level of application of the kinds of hypotheses handled in the empirical strategies in Software Engineering for Controlled experiment most apply Corroborated Software Engineering Hypotheses, for Case Study, Survey and Secondary-data studies most apply Empirical Software Engineering Hypotheses, this we report in Fig 4. We conclude that it is essential to establish from the inception of the research the type of study that will be carried out and consider the main characteristics that the empirical strategies have in SE. We corroborate that the selection of an empirical strategy in Software Engineering research depends on the nature and scope of the research and the resources that the researcher has at that moment, and the degree of scientific and methodological knowledge that he has to carry out an empirical study. Finally, we found a gap in the definition of the characteristics of empirical methods in SE research. Therefore, it is a challenge for new researchers in SE to select a correct empirical strategy for their research.

Automation in this sense would help to minimize this gap. As future work, it is necessary to continue studying in-depth the behavior and nature of the empirical strategies in Software Engineering research that allow us to contribute to the body of knowledge and strengthen the scientific taxonomy in SE and walk towards the automation of the experimental process.

VI. Appendices

The initial set of studies (80 studies) are available in [54] The list of mapped primary studies (20 studies) is available in [55]

Acknowledgment

This work is partially supported by the Universidad Técnica del Norte, Ecuador (UTN), the European Commission (FEDER) and the Spanish Government under projects HORATIO and OPHELIA (RTI2018-101204-B-C21/C22), and also by the Andalusian Administration under projects APOLO (US-1264651), and EKIPMENT-PLUS (P18-FR-2895).

References

[1] V. R. Basili, "A Personal Perspective on the Evolution of Empirical Software Engineering," in Perspectives on the Future of Software Engineering, 2013.

[2] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, Experimentation in software engineering, vol. 9783642290, no. 7. London: Springer Heidelberg New York Dordrecht London, 2012.

[3] B. Boehm, H. Dieter, M. V. Z. Eds, T. Legacy, B. Boehm, and H. D. Rombach, Foundations of Empirical Software Engineering. 2005.

[4] V. R. Basili and M. V. Zelkowitz, "Empirical studies to build a science of computer science," Commun. ACM, vol. 50, no. 11, p. 33, 2007.

[5] S. L. Pfleeger, "Experimental design and analysis in software engineering," Ann. Softw. Eng., vol. 1, no. 1, pp. 219–253, 1995.

[6] C. Wohlin and A. Aurum, "Towards a decision-making structure for selecting a research design in empirical software engineering," Empir. Softw. Eng., vol. 20, no. 6, pp. 1427–1455, 2015.

[7] K. J. Stol, M. A. Babar, B. Russo, and B. Fitzgerald, "The use of empirical methods in open source software research: Facts, trends and future directions," Proc. 2009 ICSE Work. Emerg. Trends Free. Source Softw. Res. Dev. FLOSS 2009, pp. 19–24, 2009.

[8] K. J. Stol and B. Fitzgerald, "A Holistic Overview of Software Engineering Research Strategies," Proc. - 3rd Int. Work. Conduct. Empir. Stud. Ind. CESI 2015, pp. 47–54, 2015.

[9] D. Falessi, M. A. Babar, G. Cantone, and P. Kruchten, "Applying empirical software engineering to software architecture: Challenges and lessons learned," Empir. Softw. Eng., vol. 15, no. 3, pp. 250–276, 2010.

[10] D. E. Perry, A. a Porter, and L. G. Votta, Empirical Studies of Software Engineering: A Roadmap, Proc. Conf. Futur. Softw. Eng. ICSE 00, pp. 345–355, 2000.

[11] W. Harrison and V. R. Basili, "Editorial EMSE," Empirical Software Engineering, vol. 1, no. 1, pp. 5–10, 1996.

[12] A. Borges et al., "Support mechanisms to conduct empirical studies in software engineering: A systematic mapping study," ACM Int. Conf. Proceeding Ser., vol. 27-29-Apri, 2015.

[13] B. Kitchenham et al., "Robust Statistical Methods for Empirical Software Engineering," Empir. Softw. Eng., vol. 22, no. 2, pp. 579–630, 2017.

[14] D. Falessi et al., "Empirical software engineering experts on the use of students and professionals in experiments," Empir. Softw. Eng., vol. 23, no. 1, pp. 452–489, 2018.

[15] P. C. Taylor and M. N. D. Medina, "Educational research paradigms: from positivism to multiparadigmatic," J. Meaning-Centered Educ., vol. 1, no. 2007, pp. 1–16, 2013.

[16] J. W. Creswell and C. N. Poth, Qualitative inquiry and research design: Choosing among five approaches. Thousand Oaks, CA: SAGE, 2018.

[17] A. Ahrens and J. Zaščerinska, "A Framework for Selecting Sample Size in Educational Research on e-business Application," in ICE-B2014-International Conferenceone-Business, 2014, pp. 39–46.

[18] C. Robson, Real World Research: A Resource for Social Scientists and Practitioners- Researchers, 2nd ed. Blackwell, Oxford/Madden, 2002.

[19] R. Jiménez Paneque, Metodología de la Investigación. Elementos básicos para la investigación clínica. 1998.

[20] G. Gómez and L. Reidl, "Métodos de investigación," Metodol. lInvestigación en Ciencias Soc., pp. 16–40, 2010.

[21] S. Sandage, K. Cook, P. Hill, S. Brad, R Kevin, "Hermeneutics and psychology: A review and dialectical model," Review of General Psychology, 12(4), 344–364. doi:10.1037/1089-2680.12.4.344

[22] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," Empir. Softw. Eng., vol. 14, no. 2, pp. 131–164, 2009.

[23] I. Benbasat, D. K. Goldstein, and M. Mead, "The case research strategy in studies of information systems," MIS Q. Manag. Inf. Syst., vol. 11, no. 3, pp. 369–386, 1987.

[24] P. Runeson, M. H¨ost, A. W. Rainer, and B. Regnell, Case Study Research in Software Engineering. Guidelines and Examples. Wiley, Hoboken, 2012.

[25] M. Ciolkowski, O. Laitenberger, S. Vegas, and S. Biffl, "Practical experiences in the design and conduct of surveys in empirical software engineering," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 2765, pp. 104–128, 2003.

[26] S. L. Pfleeger and B. A. Kitchenham, "Principles of Survey Research: Part 1: Turning Lemons into Lemonade," SIGSOFT Softw. Eng. Notes, vol. 26, no. 6, pp. 16–18, 2001.

[27] A. N. Ghazi, K. Petersen, S. S. V. R. Reddy, and H. Nekkanti, "Survey research in software engineering: Problems and mitigation strategies," IEEE Access, vol. 7, pp. 24703–24718, 2019.

[28] J. S. Molléri, K. Petersen, and E. Mendes, "Survey Guidelines in Software Engineering: An Annotated Review," Int. Symp. Empir. Softw. Eng. Meas., vol. 08-09-Sept, 2016.

[29] N. Juristo and A. M. Moreno, Basics of Software Engineering Experimentation. Boston: Springer, Boston, MA, 2001.

[30] J. Tisseau, "In vivo , in vitro , in silico , in virtuo The virtuoscope," 1st Work. SMA Biol. meso or Macrosc. scales, pp. 1–17, 2008.

[31] L. Zhang, J. H. Tian, J. Jiang, Y. J. Liu, M. Y. Pu, and T. Yue, "Empirical Research in Software Engineering — A Literature Survey," J. Comput. Sci. Technol., vol. 33, no. 5, pp. 876–899, 2018.

[32] L. T. Heeager and J. Rose, "Optimising agile development practices for the maintenance operation: nine heuristics," Empir. Softw. Eng., vol. 20, no. 6, pp. 1762–1784, 2015.

[33] M. Icart Isern and J. Canela Soler, "The use of hypothesis in scientific research," Atención Primaria, vol. 21, pp. 172–178, 1998.

[34] V. Mandić, J. Markkula, and M. Oivo, "Towards multi-method research approach in empirical software engineering," Lect. Notes Bus. Inf. Process., vol. 32 LNBIP, pp. 96–110, 2009.

[35] A. Zendler, "A preliminary software engineering theory as investigated by published experiments," Empir. Softw. Eng., vol. 6, no. 2, pp. 161–180, 2001.

[36] B. Kitchenham and S. Charters, "Guidelines for performing Systematic Literature Reviews in SE," 2007.

[37] B. Kitchenham, O. Pearl Brereton, D. Budgen, M. Turner, J. Bailey, and S. Linkman, "Systematic literature reviews in software engineering - A systematic literature review," Inf. Softw. Technol., vol. 51, no. 1, pp. 7–15, 2009.

[38] K. Petersen, S. Vakkalanka, and L. Kuzniarz, "Guidelines for conducting systematic mapping studies in software engineering: An update," Inf. Softw. Technol., vol. 64, pp. 1–18, 2015.

[39] K. J. Stol and B. Fitzgerald, "The ABC of software engineering research," ACM Trans. Softw. Eng. Methodol., vol. 27, no. 3, 2018.

[40] S. Easterbrook, J. Singer, M.-A. Storey, and D. Damian, "Selecting Empirical Methods for Software Engineering Research Guide to Advanced Empirical Software Engineering," Guid. to Adv. Empir. Softw. Eng., pp. 285–311, 2008.

[41] R. Wieringa, "Empirical research methods for technology validation: Scaling up to practice," J. Syst. Softw., vol. 95, pp. 19–31, 2014.

[42] M. V. Zelkowitz, D. R. Wallace, and D. W. Binkley, "Experimental Validation of New Software Technology," pp. 229–263, 2003.

[43] R. L. Glass, I. Vessey, and V. Ramesh, "Research in software engineering: An analysis of the literature," Inf. Softw. Technol., vol. 44, no. 8, pp. 491–506, 2002.

[44] A. Höfer and W. F. Tichy, "Status of empirical research in software engineering," Lect. Notes Comput. Sci. (including Subser. Lect. Notes

Artif. Intell. Lect. Notes Bioinformatics), vol. 4336 LNCS, no. 1, pp. 10–19, 2007.

[45] D. I. K. Sjøberg, T. Dybå, M. Jørgensen, and T. Dybå, “The Future of Empirical Methods in SE Research,” 2007.

[46] C. Zannier, G. Melnik, and F. Maurer, “On the success of empirical studies in the international conference on software engineering,” Proc. - Int. Conf. Softw. Eng., vol. 2006, pp. 341–350, 2006.

[47] S. Jain, M. A. Babar, and J. Fernandez, “Conducting empirical studies in industry: Balancing rigor and relevance,” 2013 1st Int. Work. Conduct. Empir. Stud. Ind. CESI 2013 - Proc., pp. 9–14, 2013.

[48] T. Dybå, D. I. K. Sjøberg, and D. S. Cruzes, “What Works for Whom, Where, When, and Why? On the Role of Context in Empirical Software Engineering,” ESEM ’12 Proc. ACM-IEEE Int. Symp. Empir. Softw. Eng. Meas., no. 7465, pp. 19–28, 2012.

[49] J. A. Carvalho, “Research Methods in Engineering, Methods for empirical research,” Portugal, 2018.

[50] M. D. Myers, “Qualitative Research in Information Systems,” Handb. Inf. Syst. Res., no. May, pp. 291–304, 2011.

[51] R. Harrison et al., “Directions and methodologies for empirical software engineering research,” Empir. Softw. Eng., vol. 4, no. 4, pp. 405–410, 1999.

[52] S. E. Sim, J. Singer, and M. A. Storey, “Beg, borrow, or steal: using multidisciplinary approaches in empirical software engineering research,” Empir. Softw. Eng., vol. 6, no. 1, pp. 85–93, 2001.

[53] D. Šmite, C. Wohlin, T. Gorschek, R. Feldt, ”Empirical evidence in global software engineering: a systematic review,” Empir. Softw. Eng., 15(1), 91–118. doi:10.1007/s10664-009-9123-y, 2010.

[54] C. Guevara-Vega, B. Bernárdez, A. Durán, A. Quiña-Mera, M. Cruz, A. Ruiz-Cortés. (2021). 80 Initial Data-set Studies SMS Strategy CG [Data set]. Presented at the Second International Conference on Information Systems and Software Technologies (ICI2ST 2021), Quito, Ecuador: Zenodo. http://doi.org/10.5281/zenodo.4456034

[55] C. Guevara-Vega, B. Bernárdez, A. Durán, A. Quiña-Mera, M. Cruz, A. Ruiz-Cortés. (2021). 20 Primary Studies SMS Strategy CG [Data set]. Presented at the Second International Conference on Information Systems and Software Technologies (ICI2ST 2021), Quito, Ecuador: Zenodo. http://doi.org/10.5281/zenodo.4455951