A structured approach to evidence-based software engineering in empirical software engineering research for students

M. Danz, T. Gräf, C. Michel*

Advisor: Andrei Miclaus†

Karlsruhe Institute of Technology (KIT)
Pervasive Computing Systems – TECO
*danz@teco.edu,tobias.graef@student.kit.edu,michel@teco.edu
†miclaus@teco.edu

Abstract.
Background: It was found, that some students struggle with scientific working.
Objective: Create a supporting document tailored for students. With a main focus on ease of use and with EBSE in mind to establish it further.
Results: Based on literature and experience, a process and supporting documents are created. The process is based on EBSE. To make the document easy to use, it incorporates simple and clear design as well as concise instructions.
Limitations: This work is mainly aimed at computer science students and needs to be evaluated in future work.

Keywords: software engineering, evidence-based software engineering, EBSE, scientific working, flowchart, checklist, student workflow

1 Introduction

Our advisor observed that most software build in a bachelor's or master's thesis is poorly evaluated or not evaluated at all. Controlled studies are needed to attain valid measurements that allow comparison. Students often struggle with obtaining this *empirical evidence* (see table 1) leading to unusable results making a proper comparison difficult or not possible. Furthermore, large pieces of software are hard to evaluate because effects can be hard to isolate.
Therefore, the aim of this work is to create a supporting system for students helping to improve the quality (in particular the substantiality) of their studies in the domain of software engineering.

The final system intended is a database containing a digitalized collection of experiments. The system is meant to simplify searching, scanning and comparing experiments. Allowing to quickly find existing evidence that can be used for software design decisions. To populate that collection, a process to create evidence correctly and a compact representation of the results are needed. In this work, two documents are proposed: *Checklist* to guide students through the scientific process and *Briefing Form* for a compact and structure resume of experiments.

First we align the proposed process with related work in section 3. In section 2 *evidence-based software engineering* (EBSE) and structured abstracts are introduced. EBSE is the foundation of the process introduced later in section 4. Therefore, the idea behind EBSE is very fundamental for this paper. Structured abstracts are used as a base for the form introduced in section 5.

2 Fundamental Principles

This chapter introduces *evidence-based software engineering* (EBSE) as well as *structured abstracts*.

EBSE introduced in 2004 as an adoption of the evidence-based approach in medicine. Kitchenham et al. thought that software engineering could profit from it in a similar manner as medicine did [24]. Finding evidence involves browsing through a lot of publications. Brereton et al. TODO: ref found that the quality of abstracts is often not sufficient to decide whether a paper is relevant in a specific context. Reading the conclusion solves this issue but increases the time needed to identify relevant studies. Structured abstracts can be used to improve the quality of abstracts, which is particularly important in the context of EBSE.

2.1 EBSE - Evidence-Based Software Engineering

The aim of EBSE is "to improve decision making related to software development and maintenance by integrating current best evidence from research with practical experience and human values" [14]. Practising EBSE includes five steps:

1. Ask an answerable question.
2. Find the best evidence that answers that question.
3. Critically appraise this evidence.
4. Apply the evidence (and critical appraisal).
5. Evaluate the performance in previous steps.

Formulating the question precisely is important for the success of the process. The question should be formulated broad enough, so important studies are not missed, but must be precise enough to cope with the amount of studies (see section 4.2).

In medicine, researchers rely heavily on already published *systematic literature reviews* (SLR) to find relevant studies. SLRs try to identify and interpret all available literature regarding a specific research question [22]. There are several organisations dedicated to conduct such reviews in medicine. The lack of this infrastructure makes applying the evidence-based approach in software engineering more difficult, but the number of existing SLRs increases steadily. For further reading on SLR, see Wohlin et al. [38].

It is important to check the quality of the identified studies, because being published is not a guarantee for absence of errors. Sometimes the integrity of results is compromised, because the research method has weaknesses or the researcher has vested interest connected to the outcome of the study.

Applying the evaluated evidence means integrating it with personal experience and other requirements. This step highly depends on the context and type of technology under evaluation.

At last the performance in previous steps is evaluated to improve applications of the EBSE process in the future.

Kitchenham et al. also identify two major problems inherent to software engineering:

1. The skill factor: Performing software engineering methods and techniques often require skilled practitioners. This prevents blinding and can therefore cause problems related to subject and experimenter bias.
2. The lifecycle issue: Prediction of behaviour of deployed technology is difficult and it is hard to isolate effects because of interaction with other methods and technologies.

Furthermore, they also state two approaches to reduce each of these effects [24].

2.2 Structured Abstract

In their guidelines for reporting experiments in software engineering Jedlitschka et al. [20] propose the use of *structured abstracts*. They adopted the idea from medicine and psychology, where structured abstracts were introduced to increase the quality of abstracts. Structured abstracts guide the writer as well as the reader by using headings. Although a variety of different elements is used (see for example TODO: "Adoption of structured abstracts by general medical journals and format for a structured abstract "), the most common elements of structured abstracts are *Background/Context*, *Objective/Aim*, *Methods*, *Results* and *Conclusion/Discussion*. Jedlitschka et al. [20] suggest the use of a sixth heading called *limitations*. This information is necessary to decide whether a result can be transfered to another context. Kitchenham et al. [23] include this information in the *conclusion* section.

The list and description of elements below closely follows the suggestion of Jedlitschka et al. [20]:

1. *Background/Context*: Briefly explains the motivation for conducting the study and refers to previous research.
2. *Objective/Aim*: Describes the purpose of the study, including the object that is studied as well as focus and perspective. This part should cover the research question.
3. *Methods*: Sums up used research methods. For example experimental design, setting, participants and selection criteria, intervention and measurement and analyzing technique.
4. *Results*: The key findings are described here in form of numerical values. Do not include interpretations here. See section 5.2 (Statistical Results) for further details.
5. *Limitations*: Describes the scope of the study to point out the limits of generalization. This element might be incorporated in the *conclusion*-element.
6. *Conclusion*: Contains the interpretation of results and puts them into larger context.

There are several studies comparing structured and unstructured abstracts in the domain of software engineering with regard to completeness and clarity:

– Structured abstracts include more relevant information and are easier to read than conventional abstracts [11,12].

– Inexperienced authors are likely to produce clearer and more complete abstracts when using a structured form [10] .
– On average structured abstracts are longer (limitations in conclusion is a good idea to prevent lengthy abstracts) and have better readability than unstructured abstracts [23].

These findings are consistent with the ones in other disciplines. It is important to mention that there are critics of structured abstracts that are supported by studies, but in general structured abstracts are considered advantageous [17,18].

A downside to structured abstracts is their length compared to unstructured ones. If the size of abstracts is limited, the abstract should still be in a structured form traditional elements should be prioritized: background (one sentence), objective, methods, result and conclusion [20].

If it is not possible to structure an abstract (e.g. due to standard of journal or supervisor, length limitations) the elements of a structured abstracts should be contained in the unstructured abstract to make sure no important information is missing. See the common structure of the clearest abstracts as found by Shaw [31]. Moreover the reader should be able to quickly identify each element by reading through the abstract.

For further information and examples of structured abstracts see the guide of C. Andrade [4] and Kitchenham et al. [24].

3 Related Work

Rainer et al. [27] released a paper about the use of EBSE by 15 under-graduate students. We have used observations listed in the paper to create design guidelines for the documents introduced in section 4 and 5 . There are seven main issues we tried to address:

1. "Students had problems constructing well-formulated EBSE questions."
2. "Students used limited criteria for identifying the best or better evidence [...]"
3. "Students used a very limited number of search terms."
4. "Students provided poor explanation in their reports of how their searches were conducted."
5. "Students varied in their use of the EBSE checklist."
6. "Some students critically appraised the technologies rather than the publications (evidence) on the technologies."
7. "But we also think that the kinds of problems students were tackling [...] are not the kinds of problems researchers commonly investigate."

Table 1. Issues with EBSE found by Rainer et al. [27]

There are already databases that are meant to ease the access to studies: SEED [19] and the *Evidence Map* [1].
SEED is a community-driven online database that has been created during graduate courses. It lists summaries of studies in a common format and grouped by topics. The studies are added by researchers, whereas users create unstructured summaries, ratings and comparison grids. There only exists a prototype [2] that has not been updated since 2009.
The Evidence Map is primarily concerned with classifying systematic literature reviews (secondary studies, see sections 2.1 and 4.3), but also provides lists of primary and tertiary studies as well as information and guidelines about EBSE. A major disadvantage is the lack of study summaries. It has not been updated since 2012.

Rainer et al. [26, p. 7] introduced a flowchart to support the use of EBSE. The chart is rather fine grained and therefore might be unsuited for students novice in EBSE and scientific working in general.

4 Research Process - Checklist

In this section, a document called *Checklist* is introduced. It is supposed to guide students through scientific working with EBSE in mind.
Rainer et al. found, that "[s]tudents varied in their use of the EBSE checklist" [27] (see issue 5 in table 1). Therefore, an important design criteria for Checklist is ease of use through clear and simple instructions. Especially tailored for students with little knowledge about scientific working in general.
The process of Checklist contains eight steps:

1. Formulate question
2. Formulate hypothesis
3. Search for existing evidence
4. Substantial evidence
5. Experiment
6. Answer question
7. Discussion
8. Evaluate process

The whole graph can be seen in figure 1. The actual document can be found in appendix A.
On the left of the document, a flow chart of the proposed process is depicted. For computer science students this should be a fast way to navigate through and orient themselves in the process. To further assist navigation visually, each process step has been assigned a unique color. This color schemes reoccurs in the tools section of the document as well as in Briefing Form .
On the right additional information is given. Each process step in Checklist contains a short description, some guidelines, and acceptance criteria. The guidelines contain methods and tools on how to process the current step. The acceptance criteria give students orientation on when a step is completed.

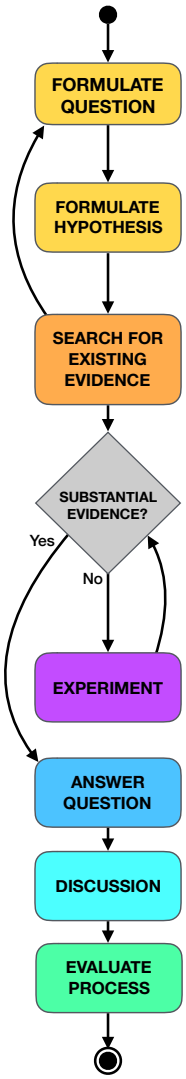


Fig. 1. Workflow Graph

4.1 Scientific Method

The scientific method is a model which describes the working process of scientific working in general. In figure 2, the Checklist process is mapped on the scientific method. The colors indicate which steps of Checklist correspond to which steps of the scientific process.
A large part of the scientific method is focused on finding research topics (white) and formulating research questions (yellow). In contrast Checklist focuses on

answering the research question. This is represented by multiple colors being mapped onto only two steps and two transitions in figure 2. The steps marked white are not contained in Checklist . The different focuses are caused by advisors usually assigning students a narrow research topic or a specific research question.

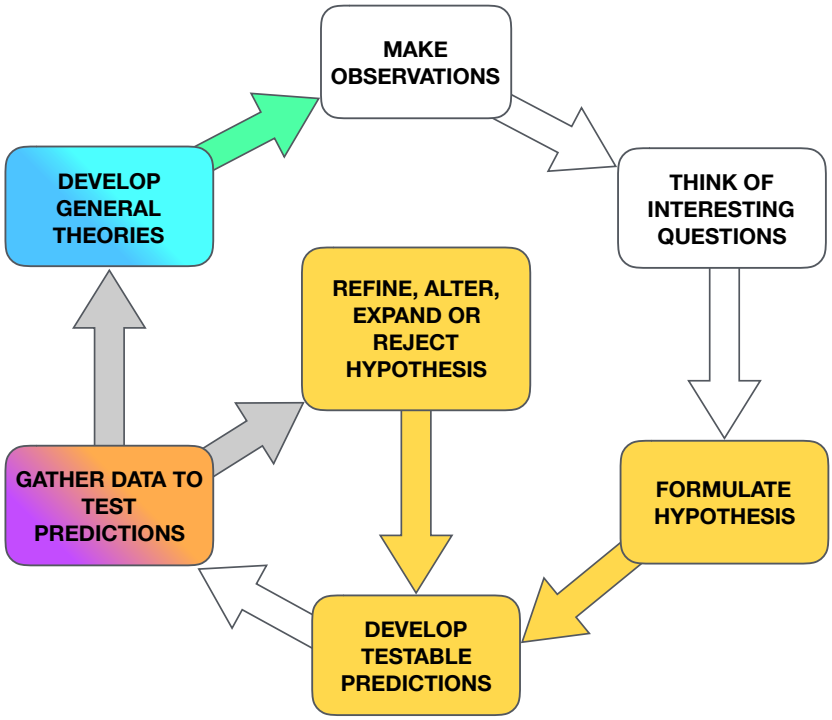


Fig. 2. Mapping of our process to the scientific method [16].

4.2 Formulating Research Question and Hypothesis

To produce relevant results and fully understand their research domain, developing a good research question is integral for researchers. A good research question is supported by a hypothesis and sometimes objectives [15]. These components should be carefully designed *before* conducting the study that tries to answer the question. Otherwise it is more likely to produce questions that are already answered, or "could potentially lead to spuriously positive findings of association through chance alone" [15, p. 280]. This seems to be especially true for students [27]. Therefore, this section addresses issue 1 from table 1.

Research Question The question a study is designed to answer is called research question [35]. It should be an answerable question that addresses a relevant issue in the research area [14]. To produce questions which drive knowledge further, a deep understanding of the topics that have already been studied is needed. The questions that arise during the acquisition of knowledge, which cannot be answered by means of EBSE, are likely appropriate questions for further research [15].

There are two general classes of research questions: qualitative and quantitative questions. Qualitative research states questions which report, describe, or explore a subject [13, p. 139-141]. The issues computer science students are confronted with are rather of quantitative nature than qualitative (e.g. "Is database engine X faster than database engine Y?"). These problems are different from the problems researchers commonly investigate (see issue 7 in table 1) [27]. Therefore, this work focuses on quantitative research questions. "Quantitative research questions inquire about the relationships among variables" [13, p. 143] and quantitative hypotheses emerge from them.

Shaw provides a model where she categorizes research questions from software engineering papers in five types to understand the structure of research questions [30].

To design a good research question, Haynes coined the acronym PICO: Population, Intervention, Comparison group, and Outcome [7]. Sometimes Time is added as fifth component, when it is important over which time frame the study is conducted. See the left box in figure 3. A research question structured with the PICOT approach supports in restricting the research question and thereby directs hypotheses and study. By restricting the research question researchers can limit bias and increase the internal validity of the study. A too narrow question may lead to decreased external validity [15].

Before PICOT Sackett and colleagues suggested that good research questions consist of three components: Intervention, Context and Outcome [28]. This is a more coarse grained decomposition than PICOT. Dybå et al. displayed a fitting example for this template in software engineering: "Does pair programming lead to improved code quality when practiced by professional software developers?" [14, p. 60] In this example, the intervention (technology) is pair programming, the context of interest are professional software developers, and the outcome (effect) is improved code quality [14]. To verify the quality of a research question Hulley et al. suggest the use of the FINER criteria. It highlights key aspects of the question and thereby provides new perspective to the proposed study. The FINER criteria consists of: Feasible, Interesting, Novel, Ethical, and Relevant [15]. A more detailed view of the FINER criteria can be seen in the right box of figure 3.

Hypothesis For each quantitative research question there should be a hypothesis - an educated guess about the outcome of the research question [9,15]. A good hypothesis needs to be a testable prediction of the studies outcome, but it

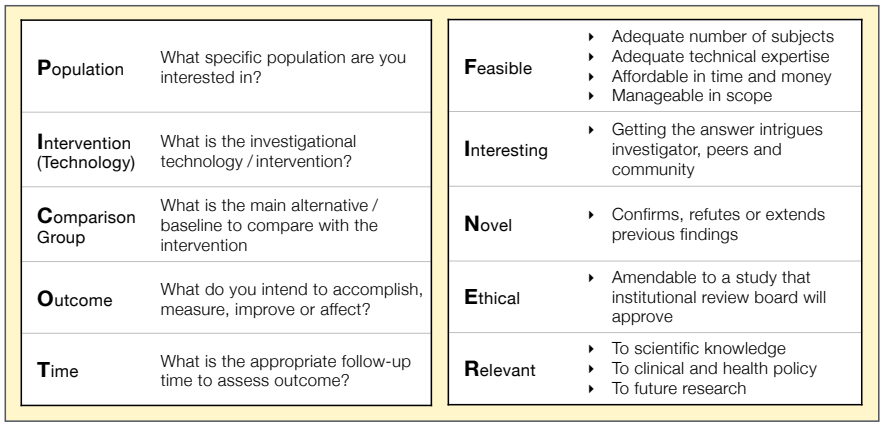


Fig. 3. PICOT criteria adjusted to computer science research [15] and FINER criteria for a good research question [15].

is important that it does not contain any interpretation [25]. A simple template for writing a hypothesis is:

> *If [I do X], then [Y] will happen.* [9]

Vickers et al. propose a more refined structure. According to that structure, a good hypothesis needs to include three components: Two or more variables, population/context, and the relationship between the variables [35]. TODO: specify the thing with the variables more. For example a good hypothesis in software engineering research could be:

> Pair programming used by professional software developers improves code quality compared to teams using conventional techniques.

Furthermore, when conducting empirical research the hypothesis should be formulated as a *null hypothesis* H_0, and be accompanied by an *alternative hypothesis* H_1 [15]. The null hypothesis is a theory that is believed to be true but not proven yet. The alternative hypothesis is the opposite prediction of the null hypothesis [25]. After the study, the null hypothesis is empirically tested, and only if it is rejected (i.e., there is a significant difference between groups) the alternative hypothesis is considered true. This confirms that effects did not show by chance alone [15]. A null hypothesis to the example above would be:

> Pair programming used by professional software developers does not affect code quality compared to teams using conventional techniques.

To further support the validity of the study, the hypotheses should be formulated as 2-sided hypothesis. “A 2-sided hypothesis states that there is a difference

between [groups, but without specifying the direction of the outcome]."[15, p.280] 1-sided hypotheses should only be used when there is a strong justification for one direction of the outcome [15]. A 2-sided revision of the H_1 from above would be:

> Pair programming used by professional software developers does affect code quality compared to teams using conventional techniques.

Objectives Sometimes researchers add objectives to their hypotheses. They are active statements that "define specific aims of the study and should be clearly stated" [15, p. 280] at the beginning of research. Objectives help to outline the study (e.g. helping to calculate sample size) [15,35]. Objectives are not included in Briefing Form but are mentioned here for completeness.

4.3 Search for Existing Evidence

After formulating a research question it is important to know which scientific results exist that help answering this question. Students normally perform an "ad-hoc" literature review that is prone to missing evidence and getting biased results (experimenter bias).

For that reason EBSE proposes the use of *systematic literature reviews* (SLR), a type of secondary study, to search for studies related to the research question. The aim of SLRs is to "identify, assess and combine the evidence from primary research studies using an explicit and rigorous method" [39]. Using this well-defined method might prevent biased results of the literature review, but requires more effort - both in time and skill - than traditional literature reviews. There are guidelines [22,37,39] and reports of experiences with conducting SLRs [6] that provide the means necessary to conduct a SLR. In any case published SLRs are very useful. They provide a summary of multiple studies concerning one research question even if conducting one is not feasible. Currently there are at least two projects concerned with making SLRs and finding relevant studies easier by collecting and indexing them. Namely SEED [19] and Evidence Map [1]. Both have not been updated recently. TODO: more disadv?

Students voted the SLR conducted during the EBSE process as one of the hardest steps [22]. Therefore, in this work a less formal search process to save time and effort is suggested. But it should be noted, that the search should still be planned and structured to get the desired results TODO: state it?. If it is possible conducting a SLR is always preferable.

Nevertheless, several guidelines from SLRs still can be used:

- Think about search strings and search engines beforehand and write them down. This also addresses issue 4 in table 1.
- It is unlikely to find all relevant literature using only a single search engine [6].

– *Snowballing* is useful, both in a forward and backward manner [37]. Keywords found during this process, can also be used to refine the search. This addresses issue 3 in table 1. Usually, search engines provide these features.
Backward-snowballing refers to looking at the references of a publication and therefore going back in time.
Forward-snowballing means going forward in time by looking at the papers citing the current paper.
– The search strategy depends on the research question. If the research domain provides a vast amount of studies the search can be restricted, for example by excluding older studies [6].
– Even for more experienced students it might be necessary to refine the research question based on the search result. Because their understanding of the research domain increases [6]. For example having only a few search result can be caused by research questions that are too narrow. Research Questions that are too general can lead to an abundance of search result which would take too long to examine.
– Categorize your research question using classification systems like the *ACM Computing Classification System*TODO: ref to url and search within these categories. This is especially helpful if it is problematic to limit the scope in the research area or there are uncertainties about wording and naming.

TODO: mention? systematic mapping studies (broader, but not as deep as SLRs (deep: quantitative analysis and quality assessment)) "Systematic Mapping Studies in Software Engineering", Petersen et al. 2008

4.4 Designing, Conducting And Interpreting Experiments

To generate evidence, experiments are conducted. To keep the complexity of the study manageable and to prevent side effects, it is recommended to answer only one question per experiment. If the question is too extensive for a single experiment, split up the question in sub-questions and recursively start this process for each sub-question.
Guidance and documentation through the experiment is provided by Briefing Form introduced in section 5. Since experimenting is a very complex topic and can not be fully covered in this paper, see [38,34] for further reading.

4.5 Answer Question

One way to answer the research question is using existing evidence that needs to be critically appraised prior to drawing conclusions. This includes checking study design, study quality, relevance for the research question, as well as consistency between different studies.

Rainer et al. found that students had problems critically appraising evidence. See table 1 issues 2 and 6. They suggest sensitizing students for biases to solve this issue [27]. There are several tools to support critical appraisal of studies.

Figure 4 shows a checklist containing important factors to consider when appraising a study.
Another tool is the *GRADE approach* [5], a grading system for studies that goes beyond simple hierarchies of study types. It was introduced for medicine and has been used in the software engineering domain [36,33]. GRADE is a well-defined method and differentiates between the quality of evidence and the strength of recommendation.

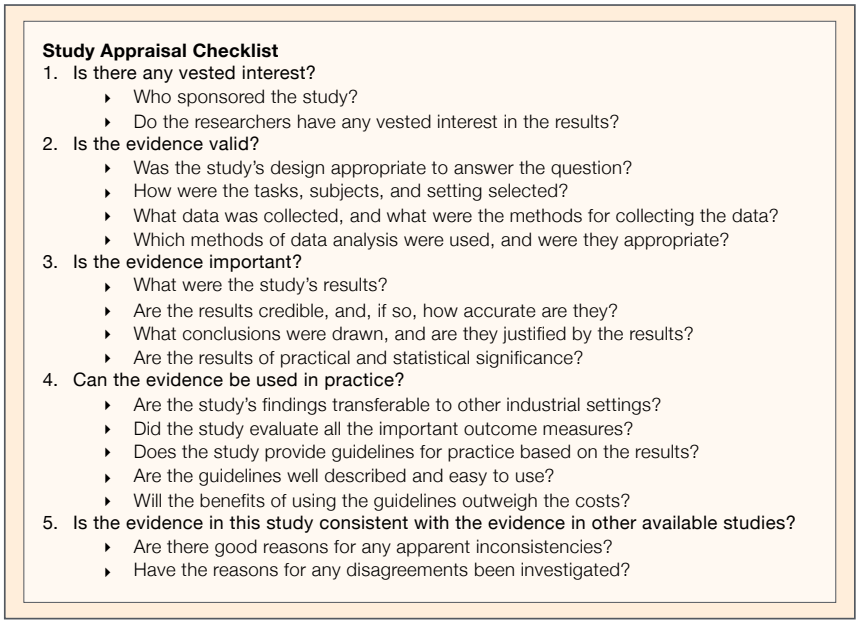


Fig. 4. Checklist for critical appraisal of studies compiled by Dybå et al. [14].

Jørgensen et al. [21] reported indications of research and publication bias being quite common in the domain of software engineering. So, an important aspect of appraising studies is checking if they are biased. Shepperd presents similar findings and gives a good and short overview of the problem [32].

The critically appraised evidence (and the result of the conducted study) is used as foundation to accept or reject the hypothesis. Followed by answering the research question accordingly.

4.6 Discussion

In this step the whole study should be discussed with respect to limitations and the scope of generalization. The study should be put in larger context of the concerning field of research. Additionally future approaches and increments can

be discussed here. For guidance in analyzing studies it is recommended to use a checklist as proposed by Dybå et al. (see fig. 4).

Make sure to only discuss the content of the study. For reflection on the process, see section 4.7.

4.7 Evaluate Process

The last step is to reflect on the whole process. After discussing the study, Dybå et al. recommend to reflect on how well each step was performed and to find improvements for the next iteration. To support this they recommend using *after-action reviews* (AAR) and *postmortem analysis* (PMA) [14].

AAR is a short meeting of 10 to 20 minutes to answer the four questions in figure 5, left box. This method can also be used at any other point during the process if there is a need for reflection or rethinking the current situation [14].

PMA is similar to AAR but with a deeper insight. Therefore it lasts for several hours up to a full day. It answers the questions in figure 5, right box [14]. Here, conclusions can be drawn to improve upcoming EBSE cycles and further research in general. This method can give insight in issues. For example:

- Which methods worked out as planned and which did not?
- Where was time lost and deadlines missed?
- Which technique proves to be used in upcoming studies?

After this step the current iteration of the process is finished. It can be started over again with the next research question.

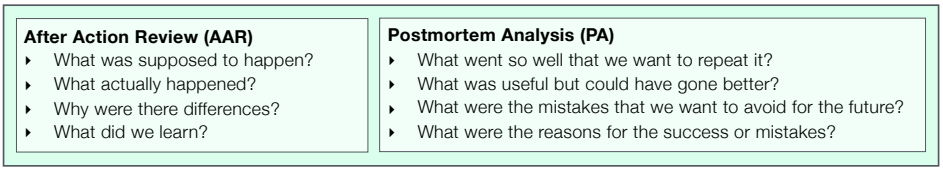


Fig. 5. After-action reviews (AAR) left box, and postmortem analysis (PMA) right box [14].

TODO: remove repetition in heading if we do not find better name for briefing form.

5 Briefing Form - Briefing Form

In this section, *Briefing Form* is introduced (see appendix B). It is a one page sheet designed for two purposes: Guiding users through the design of an experiment and provide summaries of experiments that are easier searchable and screenable. It is similar to SEED but has a more detailed structure to additionally support the search for existing evidence.

Experiments are used to obtain scientific evidence. To increase the reliability of evidence, experiments needs to be designed and conducted with minimal flaws. This can be a very difficult task because experiments and their interpretation can be prone to errors or mistakes. Especially for people new to experimenting, an awareness for common mistakes and best practices can be very beneficial. By supporting the researcher in understanding the study thoroughly mistakes can be discovered early in the process.
On a small scale, Briefing Form is also meant to be a supporting framework for a systematic workflow.

For software practitioners, it is important to quickly find solutions for a problem. Reading through papers can be very time consuming. Briefing Form can help speed up the search by providing a clear structure for a quick overview of an experiment.

Since Briefing Form is meant to implicitly guide the user through experimenting and thereby preventing mistakes. It consists of three parts: Research Question/Hypothesis, Experiment and Conclusion. These are explained in the following.

5.1 Research Question and Hypothesis

In many cases question and hypothesis are rather redundant. Nonetheless, both are included in Briefing Form . The research question is included, because researchers use to search for existing evidence by entering their question into search engines. Whereas the hypothesis has its right to exist in Briefing Form to support evaluation of relevance and validity of found evidence.

Therefore the form should be filled in with the research question and hypothesis constructed earlier (see section 4.2). The research question contains at least technology, context and effect. The hypothesis is in form of a testable prediction (e.g., “If [I do X], then [Y] will happen.” [9]).

5.2 Experiment

To find cause and effect relationships between variables experiments are conducted [8]. An experiment consists of: Variables, techniques, and statistical results. These are integral components to describe a study in brief. They also

enhance the ability to search for, and evaluate the relevance and validity of a study. For further reading see section 4.4.

Variables Variables are operationalizations of concepts. They are split in three types: Dependent, independent and controlled variables [8,29]. A good variable must be measurable by any means [8]. Seltman additionally defines several qualities of good variables: "high reliability, absence of bias, low cost, practicability, objectivity, high acceptance, and high concept validity" [29, p. 10]. The variables can also be classified by their statistical type. Which influences the statistical methods used to evaluate the experiment outcome [29, p. 12-16].
The variables are contained in Briefing Form to decompose an experiment for quicker overview while searching. Additionally the variables make it easier to evaluate the validity of the evidence.

Independent Variables The variables that are changed by the researcher during the experiment are called independent variables. In the pair programming example in section 4.2, the independent variable is the technique utilized by the software developers (e.g., pair programming and conventional techniques). It is advisable to only measure one independent variable during one experiment. Unless there is justifiable evidence that two or more independent variables are not effecting the same dependent variable [8]. Otherwise it is not clear which of the independent variables caused the observed behavior of the measured dependent variable.

Dependent Variables An experiment focuses on the effect of independent variables on dependent variables. Therefore dependent variables are the variables that are measured. For example the code quality of a software project is the dependent variable used in the pair programming example from 4.2.

Controlled Variables The third variable category are controlled variables. They have or may have an effect on the dependent variables, but are not focus of the study. Therefore they are controlled by the researcher and kept constant throughout the experiment [8]. Examples for controlled variables in software engineering research are: the test environment of a software, the gender of participants, or the number of trials per group.

Research Techniques For researchers that sift through existing evidence it is important to know by which means a hypothesis was tested. Therefore a section for research techniques is included in Briefing Form . Here the conducting researcher gives a brief resume about which techniques are used. For example: the experiment is conducted as double blind experiment, using A/B testing, with a think aloud session and data is also measured via EEG (Electroencephalography). The term techniques is used for everything ranging from methods that tell how an experiment is conducted, over measurement techniques, through to the hardware that is used. For further reading and more details on research techniques, measurement methods and study design see [3,38].

Statistical Results The statistical results help validating the solidity of evidence found. They should never contain any interpretation. Interpretations should be given in a conclusion or discussion section separately from the found data.

This field should at least contain the used statistic measurement, method, or model and their resulting parameters (e.g. Analysis of Variance: $F(2.57) = 211.496, p < .001$). It is advisable to use established methods to ensure comparability and reproducibility. More on statistical analysis can be found in books like [38], or [3] as already mentioned in section 4.4.

5.3 Conclusion

This part contains the findings and conclusion of the study. The conclusion indicates whether the question is answered or if more research has to be done on this issue. This includes an interpretation of the experiment results, and the verification of H_0 which is either accepting or rejecting H_0. Accepting H_0 leads to rejection of H_1 and vice versa. Also a short statement on the scope of generalization should be given.

6 Discussion

To support students in scientific working, this paper introduced a process along with guiding documents. The process is mainly based on EBSE.
The workflow and documents were not evaluated in this work. Therefore, they need to be tested in future work. This should be done with students in real world scenarios such as bachelor and master theses. Comparing multiple theses is a difficult task. There are many influencing factors and already a certain variance in the quality of theses requiring a lot of subjects. Also finding a suiting metric to classify and compare two theses that might not be particularly similar is not trivial. Whereas asking students in a qualitative study can provide useful insight whether the documents were actually helpful or not.
When formatively evaluated, a digitalized version of Briefing Form should be implemented to allow state of the art data collection and searching. This is a step towards an infrastructure which supports the use of EBSE on a larger scale. That kind of infrastructure might also be of interest for software practitioners and scientists.

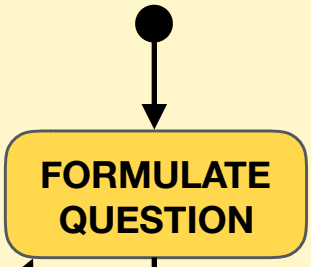
References

1. Evidence Based Software Engineering. https://community.dur.ac.uk/ebse, accessed 2017-02-25
2. SEED. http://evidencebasedse.com, 2017-2-25
3. Albert, B., Tullis, T.: Measuring the user experience. Collecting, Analyzing, and Presenting Usability ... pp. 1–17 (2008), http://www2.engr.arizona.edu/{~}ece596c/lysecky/uploads/Main/Lec11.pdf
4. Andrade, C.: How to write a good abstract for a scientific paper or conference presentation. Indian Journal of Psychiatry 53(2), 172 (2011), http://www.ncbi.nlm.nih.gov/pubmed/3136027{\%}5Cnhttp://www.indianjpsychiatry.org/text.asp?2011/53/2/172/82558
5. Atkins, D., Best, D., Briss, P.A., Eccles, M., Falck-Ytter, Y., Flottorp, S., Guyatt, G.H., Harbour, R.T., Haugh, M.C., Henry, D., Hill, S., Jaeschke, R., Leng, G., Liberati, A., Magrini, N., Mason, J., Middleton, P., Mrukowicz, J., O'Connell, D., Oxman, A.D., Phillips, B., Schünemann, H.J., Edejer, T.T.T., Varonen, H., Vist, G.E., Williams, J.W., Zaza, S., GRADE Working Group: Grading quality of evidence and strength of recommendations. BMJ (Clinical research ed.) 328(7454), 1490 (2004), http://www.ncbi.nlm.nih.gov/pubmed/15205295{\%}5Cnhttp://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC428525
6. Brereton, P., Kitchenham, B.A., Budgen, D., Turner, M., Khalil, M.: Lessons from applying the systematic literature review process within the software engineering domain. Journal of Systems and Software 80(4), 571–583 (2007)
7. Brian Haynes, R.: Forming research questions. Journal of Clinical Epidemiology 59(9), 881–886 (2006)
8. Buddies, S.: Variables in your science fair project. http://www.sciencebuddies.org/science-fair-projects/project_variables.shtml, accessed 2017-01-06
9. Buddies, S.: A Strong Hypothesis. http://www.sciencebuddies.org/blog/2010/02/a-strong-hypothesis.php (2010), accessed 2017-01-02

10. Budgen, D., Burn, A.J., Kitchenham, B.: Reporting computing projects through structured abstracts: A quasi-experiment. Empirical Software Engineering 16(2), 244–277 (2011)
11. Budgen, D., Kitchenham, B., Charters, S., Turner, M., Brereton, P., Linkman, S.: Preliminary results of a study of the completeness and clarity of structured abstracts. Proc. of the 11th Int. Conf. on Evaluation and Assessment in Software Engineering pp. 64–72 (2007)
12. Budgen, D., Kitchenham, B.A., Charters, S.M., Turner, M., Brereton, P., Linkman, S.G.: Presenting software engineering results using structured abstracts: A randomised experiment. Empirical Software Engineering 13(4), 435–468 (2008)
13. Creswell, J.W.: Research Design: Qualitative, Quantitative, and Mixed Methods Approaches (2014)
14. Dybå, T., Kitchenham, B.A., Jørgensen, M.: Evidence-based software engineering for practitioners. IEEE Software 22(1), 58–65 (2005)
15. Farrugia, P., Petrisor, B.A., Farrokhyar, F., Bhandari, M.: Practical tips for surgical research: Research questions, hypotheses and objectives. Canadian journal of surgery. Journal canadien de chirurgie 53(4), 278–281 (2009)
16. Garland, Jr., T.: The Scientific Method as an Ongoing Process. http://idea.ucr.edu/documents/flash/scientific_method/story_html5.html (2015), accessed 2017-03-24
17. Hartley, J.: Current findings from research on structured abstracts. Journal of the Medical Library Association 92(3), 368 (2004)
18. Hartley, J.: Current findings from research on structured abstracts: an update. Journal of the Medical Library Association: JMLA 102(3), 146 (2014)
19. Janzen, D.S., Ryoo, J.: Seeds of Evidence: Integrating Evidence-Based Software Engineering. In: Software Engineering Education Conference, Proceedings. pp. 223–232 (2008)
20. Jedlitschka, A., Ciolkowski, M., Pfahl, D.: Reporting experiments in software engineering. In: Guide to advanced empirical software engineering, pp. 201–228. Springer (2008)
21. Jørgensen, M., Dybå, T., Liestøl, K., Sjøberg, D.I.K.: Incorrect results in software engineering experiments: How to improve research practices. Journal of Systems and Software 116, 133–145 (2016)
22. Keele, S.: Guidelines for performing systematic literature reviews in software engineering. In: Technical report, Ver. 2.3 EBSE Technical Report. EBSE (2007)
23. Kitchenham, B.A., Brereton, O.P., Owen, S., Butcher, J., Jefferies, C.: Length and readability of structured software engineering abstracts. IET Software 2, 37 – 45 (2008), http://www.redi-bw.de/db/ebsco.php/search.ebscohost.com/login.aspx%3fdirect%3dtrue%26db%3daph%26AN%3d30193038%26site%3dehost-live
24. Kitchenham, B.A., Dyba, T., Jørgensen, M.: Evidence-based software engineering. In: Proceedings of the 26th international conference on software engineering. pp. 273–281. IEEE Computer Society (2004)
25. Prasad, S., Rao, A., Rehani, E.: Developing hypothesis and research question. 500 Research Methods pp. 1–30 (2001), http://www.public.asu.edu/{~}kroel/www500/hypothesis.pdf
26. Rainer, A., Beecham, S.: Supplementary Guidelines , Assessment Scheme and evidence-based evaluations of the use of Evidence Based Software Engineering, Version 2 (February 2008), 1–27 (2008)
27. Rainer, A., Hall, T., Baddoo, N.: A preliminary empirical investigation of the use of evidence based software engineering by under-graduate students. 10th Interna-

tional Conference on Evaluation and Assessment in Software Engineering (EASE 2006) (2006)
28. Sackett, D.: Evidence-based medicine : how to practice and teach EBM (2000), http://www.ncbi.nlm.nih.gov/pubmed/12037026
29. Seltman, H.: Experimental Design and Analysis. Online Book p. 428 (2015)
30. Shaw, M.: What makes good research in software engineering? International Journal on Software Tools for Technology ... 4(1), 1–7 (2002), http://link.springer.com/article/10.1007/s10009-002-0083-4
31. Shaw, M.: Writing good software engineering research papers: minitutorial. In: Proceedings of the 25th international conference on software engineering. pp. 726–736. IEEE Computer Society (2003)
32. Shepperd, M.: How Do I Know Whether to Trust a Research Result? IEEE Software 32(1), 106–109 (jan 2015), http://ieeexplore.ieee.org/document/7030205/
33. Tore, D., Dingsøyr, T.: Empirical studies of agile software development: A systematic review. Information and Software Technology 50(9-10), 833–859 (2008)
34. Tullis, T., Albert, B.: Measuring the user experience (2013)
35. Vickers, P., Offredy, M.: Developing a Healthcare Research Proposal: An Interactive Student Guide. http://www.health.herts.ac.uk/immunology/Web%20programme%20-%20Researchhealthprofessionals/hypothesisresearch_question.htm, accessed 2017-01-06
36. Wohlin, C.: An Evidence Profile for Software Engineering Research and Practice. Perspectives on the Future of Software Engineering (2013), http://link.springer.com/chapter/10.1007/978-3-642-37395-4{_}10
37. Wohlin, C.: Guidelines for Snowballing in Systematic Literature Studies and a Replication in Software Engineering. 18th International Conference on Evaluation and Assessment in Software Engineering (EASE 2014) pp. 1–10 (2014), http://dl.acm.org/citation.cfm?doid=2601248.2601268
38. Wohlin, C., Runeson, P., Höst, M., Ohlsson, M.C., Regnell, B., Wesslén, A.: Experimentation in software engineering, vol. 9783642290 (2012)
39. Zhang, H., Babar, M.A., Tell, P.: Identifying relevant studies in software engineering. Information and Software Technology 53(6), 625–637 (2011)

A Checklist



FORMULATE QUESTION

DESCRIPTION

- Contains **technology** in a **context** showing an **effect**.

GUIDELINES

- Create question using a structured approach such as **PICOT**[1].
- Validate question with **FINER**[1].

ACCEPTANCE CRITERIA

- [ ] Question systematically constructed
- [ ] Formulation of question validated

FORMULATE HYPOTHESIS

DESCRIPTION

- A testable prediction based on the question.

GUIDELINES

- Contains: two or more variables, context, relationship between the variables
- *'If I [do X], then [Y] will happen.'*[2]

ACCEPTANCE CRITERIA

- [ ] Hypothesis is formulated as prediction
- [ ] Prediction is testable

SEARCH FOR EXISTING EVIDENCE

DESCRIPTION

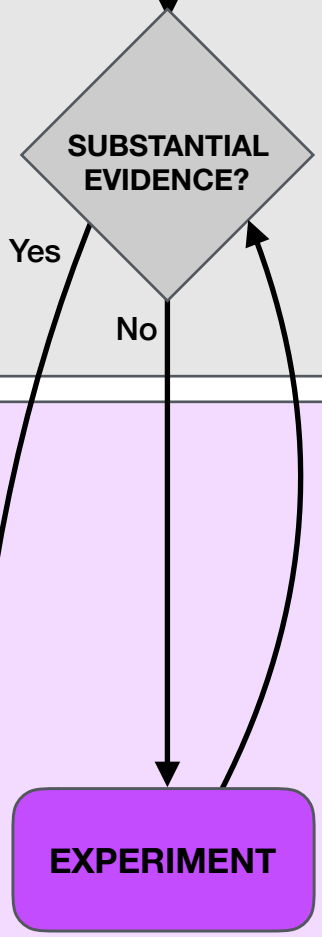
- Search for evidence related to the research question.
- Enlarge your Personal Evidence Pool.
- Deepen your understanding of the research domain.
- If necessary, incrementally refine the research question.

GUIDELINES

- Try to find as much relevant evidence as possible
- Use 'Cited by' function of search engines
- Read through found work's bibliography
- Refer to Systematic Literature Reviews for a wide comparison of related work
- Find similar work using classification systems such as CCS[3]
- Use EBSE Checklist[4] for critical appraisal of related work
- Note search strings for more structured search approach
- Too few search results? Make Research Question more general
- Too many/general search results? Narrow down Research Question

ACCEPTANCE CRITERIA

- [ ] Rough understanding/overview of research domain
- [ ] Deep understanding of research domain
- [ ] Critical appraisal of relevant related work



DESCRIPTION

Check if Personal Evidence Pool contains enough substantial evidence to answer the Research Question.

DESCRIPTION

- Design, Conduct and Evaluate Experiment
- If the Research Question can not be answered in one experiment: Split up the question. Recursively start a new process instance in FORMULATE QUESTION using a sub-question.

GUIDELINES

- Use guidelines from literature like, *Experimenting in Software Engineering'*[5] to design and conduct experiments.
- Use Briefing Form

ACCEPTANCE CRITERIA

- [ ] Experiment Designed
- [ ] Experiment Conducted
- [ ] Experiment Evaluated
- [ ] Briefing Form filled in

DESCRIPTION

- Accept or reject Hypothesis based on evidence.
- Answer the Research Question accordingly.

GUIDELINES

- Make sure evidence is substantial enough to answer the Research Question.
- Try to avoid bias. (Confirmation Bias, Information Expectancy Bias, ...)

ACCEPTANCE CRITERIA

- [ ] Validity of Hypothesis is evaluated
- [ ] Research Question is answered

DISCUSSION

DESCRIPTION

- Discuss the whole study, suggest future approaches and limits.

GUIDELINES

- Critically assess study using **EBSE Checklist**[4]
- Put study in relation to larger context.

ACCEPTANCE CRITERIA

- [ ] Content of study critically assessed
- [ ] Study put in larger context

EVALUATE PROCESS

DESCRIPTION

- Reflect on your work in the previous steps and seek ways to improve your future performance.

GUIDELINES

- Use **After Action Review** and **Postmortem Analysis**[4]

ACCEPTANCE CRITERIA

- [ ] AAR done
- [ ] PMA done
- [ ] Conclusions drawn for future processes

REFERENCES

[1] *Farrugia, P., Petrisor, B.A., Farrokhyar, F., Bhandari, M.: Practical tips for surgical research: Research questions, hypotheses and objectives. Canadian journal of surgery. Journal canadien de chirurgie 53(4), 278–281 (2009)*

[2] *Buddies,S.: A Strong Hypothesis (2010), http://www.sciencebuddies.org/blog/ 2010/02/a-strong-hypothesis.php*

[3] *http://dl.acm.org/ccs/ccs.cfm*

[4] *Dybå, T., Kitchenham, B.A., Jorgensen, M.: Evidence-based software engineering for practitioners. IEEE Software 22(1), 58–65 (2005)*

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

Population	What specific population are you interested in?
Intervention (Technology)	What is the investigational technology / intervention?
Comparison Group	What is the main alternative / baseline to compare with the intervention
Outcome	What do you intend to accomplish, measure, improve or affect?
Time	What is the appropriate follow-up time to assess outcome?

Feasible	▸ Adequate number of subjects ▸ Adequate technical expertise ▸ Affordable in time and money ▸ Manageable in scope
Interesting	▸ Getting the answer intrigues investigator, peers and community
Novel	▸ Confirms, refutes or extends previous findings
Ethical	▸ Amendable to a study that institutional review board will approve
Relevant	▸ To scientific knowledge ▸ To clinical and health policy ▸ To future research

Study Appraisal Checklist

1. Is there any vested interest?
   - Who sponsored the study?
   - Do the researchers have any vested interest in the results?
2. Is the evidence valid?
   - Was the study's design appropriate to answer the question?
   - How were the tasks, subjects, and setting selected?
   - What data was collected, and what were the methods for collecting the data?
   - Which methods of data analysis were used, and were they appropriate?
3. Is the evidence important?
   - What were the study's results?
   - Are the results credible, and, if so, how accurate are they?
   - What conclusions were drawn, and are they justified by the results?
   - Are the results of practical and statistical significance?
4. Can the evidence be used in practice?
   - Are the study's findings transferable to other industrial settings?
   - Did the study evaluate all the important outcome measures?
   - Does the study provide guidelines for practice based on the results?
   - Are the guidelines well described and easy to use?
   - Will the benefits of using the guidelines outweigh the costs?
5. Is the evidence in this study consistent with the evidence in other available studies?
   - Are there good reasons for any apparent inconsistencies?
   - Have the reasons for any disagreements been investigated?

After Action Review (AAR)

- What was supposed to happen?
- What actually happened?
- Why were there differences?
- What did we learn?

Postmortem Analysis (PA)

- What went so well that we want to repeat it?
- What was useful but could have gone better?
- What were the mistakes that we want to avoid for the future?
- What were the reasons for the success or mistakes?

REFERENCES

FINER, PICOT: *Farrugia, P., Petrisor, B.A., Farrokhyar, F., Bhandari, M.: Practical tips for surgical research: Research questions, hypotheses and objectives. Canadian journal of surgery. Journal canadien de chirurgie 53(4), 278–281 (2009)*

Checklist, AAR, PA: *Dybå, T., Kitchenham, B.A., Jorgensen, M.: Evidence-based software engineering for practitioners. IEEE Software 22(1), 58–65 (2005)*

B Briefing Form

QUESTION

HYPOTHESIS

EXPERIMENT

VARIABLES

Dependent

Independent

Control

Technique

Statistical Results

CONCLUSION