

Empirical Software Engineering Report4:

Analyzing and Comparing Survey

181250060 蒋祚竑

181250066 李宾达

181250090 刘育麟

181870108 李泳劭(presenter)

group: 15 year: 18

2021 年 7 月 4 日

1 Introduction

1.1 Overview

We collected totally 27 survey studies papers published in 2018 while 7 of them from EASE, 13 from EMSE and 7 from ESEM. In this period we found some errors, finally we fix the result and select some papers.

1.2 Rectification

After re-judging the papers we selected in previous work, we find that there are some mistake of judgement about survey papers:

1.2.1 ESEM

In table 1.

表 1: ESEM - Rectification

paper name	new category	number
software analytics in continuous delivery: a case study on success factors	Case study	13
Relationship between Geographical Location and Evaluation of Developer Contributions in GitHub	Case study	29
Building a Collaborative Culture: A Grounded Theory of Well Succeeded DevOps Adoption in Practice	Case study	24

1.3 Chosen

We choose 5 papers for further analysis and comparison. They are EASE[11], EMSE[39], EMSE[41], ESEM[22], ESEM[12].

1.3.1 EASE[11] - Satisfaction, Practices, and Influences in Agile Software Development

Abstract

The principles behind the Agile Manifesto begin with "Our highest priority is to satisfy the customer...". It also states that Agile projects should be build around motivated and self-organized teams, which might also lead to more satisfied developers. Several studies indeed report an increased job satisfaction by anecdotal evidence. In this paper we address the topic of satisfaction by in-depth analysis of the results

of a nationwide survey about software development in Switzerland. We wanted to find out if satisfaction depends on the applied development method, and, more concrete, how satisfaction relates to other elements in the development process, including the use of various practices, and the influences on business, team and software issues. We found that higher satisfaction is reported more by those using Agile development than with plan-driven processes. We explored the different perspectives of developers and those with a management role and found a high consistency of satisfaction between Agile developers and Agile management, and big differences with using working plan-driven methods. We found that certain practices and influences have high correlations to satisfaction, and that collaborative processes are closely related to satisfaction, especially when combined with technical practices. Applying recursive partitioning, we found which elements were most important for satisfaction, and gained insight about how practices and influences work in combination. We also explored the relationship between satisfaction and personal experience with Agile development. Our results in this analysis are principally descriptive, but we think they can be a relevant contribution to understand the challenges for everyone involved in Agile development, and can help in the transformation to Agile.

Citation

Satisfaction, Practices, and Influences in Agile Software Development
Martin Kropp, Andreas Meier, Craig Anslow, and Robert Biddle
10.1145/3210459.3210470

1.3.2 EMSE[39] - Identification and prioritization of SLR search tool requirements: an SLR and a survey

Abstract

Context The number of published systematic literature reviews (SLRs) in software engineering venues is increasing. However, even with their high adaptation rate, the task of performing an SLR requires a large amount of effort and presents a number of barriers. Specifically, during the SLR search phase authors must expend a lot of time and overcome a large number of barriers.

Objective To help alleviate some of the barriers in the search phase, we identify and prioritize SLR search tool requirements based on input from the SLR community. These requirements will help tool builders ensure they focus their efforts appropriately.

Method We conducted an SLR and a survey of SLR authors in software engineering. In the SLR we extracted problems and solutions SLR authors reported during their search processes. In the survey we asked respondents to describe the problems they faced during SLR search and to specify any requirements they would like to see as part of an SLR search tool. We also asked survey respondents to describe any tools they use to support SLR search, along with the strengths and limitations of those tools.

Results Based on analysis of 84 studies from the literature and a qualitative analysis of 131 responses from researchers, we identified a set of functional requirements that authors, researchers, and tool builders can use as a reference. We also describe the tools currently used by SLR researchers.

Conclusions The list of SLR search tool requirements can be used by tool builders as a guide when constructing new tools. Our analysis of tools showed that more recent tools are covering more of the requirements than in the past.

Citation

Identification and prioritization of SLR search tool requirements: an SLR and a survey

Ahmed Al-Zubidy · Jeffrey C. Carver

10.1007/s10664-018-9626-5

1.3.3 EMSE[41] - Investigating whether and how software developers understand open source software licensing

Abstract

Software provided under open source licenses is widely used, from forming high-profile stand-alone applications (e.g., Mozilla Firefox) to being embedded in commercial offerings (e.g., network routers). Despite the high frequency of use of open source licenses, there has been little work about whether software developers understand the open source licenses that they use. To help understand whether or not developers understand the open source licenses they use, we conducted a survey that posed development scenarios involving three popular open source licenses (GNU GPL 3.0, GNU LGPL 3.0 and MPL 2.0) both alone and in combination. The 375 respondents to the survey, who were largely developers, gave answers consistent with those of a legal expert's opinion in 62% of 42 cases. Although developers clearly understood cases involving one license, they struggled when multiple licenses were involved. To understand the context in which licensing issues arise in practice, we analyzed real-world questions posed by developers about the three licenses considered in the survey on online question-and-answer communities. We also interviewed practicing developers about license interaction problems they have faced. Among several lessons, we learnt that licensing issues can constrain software evolution and that developers are cautious of more restrictive licenses. Our results indicate a need for tool support to help guide developers in understanding the structure of the code and the technical details of a project while taking into account the exact requirements imposed by the licenses involved.

Citation

Investigating whether and how software developers understand open source software licensing

Daniel A. Almeida · Gail C. Murphy · Greg Wilson · Michael Hoyer
10.1007/s10664-018-9614-9

1.3.4 ESEM[22] - Development Processes and Practices in a Small but Growing Software Industry – a Practitioner Survey in New Zealand

Abstract

Background Development processes and practices depend on the context in which software is developed (e.g. locations, organizations, projects, developers).

Aim We aim at understanding development processes and practices in New Zealand, a country with a relatively small but growing software sector. We are particularly interested in methods and practices used in such environment, the implementation technologies software development professionals use, how professionals ensure software quality, and how they manage software release processes.

Method We conducted a descriptive survey targeting individual software development professionals working in New Zealand software companies.

Results New Zealand professionals use similar methodologies as professionals in other countries. Popular programming languages differ somewhat to popular languages in other rankings. Quality assurance is rather ad-hoc and the release process is inspired by agile software development principles.

Conclusions Our findings highlight some differences of the New Zealand software industry to other countries. Furthermore, we identified some strengths and weaknesses related to processes and practices. Our findings can help software professionals and organizations reflect on (and potentially adjust) the way they work.

Citation

Development processes and practices in a small but growing software industry: a practitioner survey in New Zealand

Di Wang, Matthias Galster

10.1145/3239235.3268926

1.3.5 ESEM[12] - Understanding the Software Development Practices of Blockchain Projects: A Survey

Abstract

Background The application of the blockchain technology has shown promises in various areas, such as smart-contracts, Internet of Things, land registry management, identity management, etc. Although Github currently hosts more than three thousand active blockchain software (BCS) projects, a few soft-

ware engineering research has been conducted on their software engineering practices.

Aims To bridge this gap, we aim to carry out the first formal survey to explore the software engineering practices including requirement analysis, task assignment, testing, and verification of blockchain software projects.

Method We sent an online survey to 1,604 active BCS developers identified via mining the Github repositories of 145 popular BCS projects. The survey received 156 responses that met our criteria for analysis.

Results We found that code review and unit testing are the two most effective software development practices among BCS developers. The results suggest that the requirements of BCS projects are mostly identified and selected by community discussion and project owners which is different from requirement collection of general OSS projects. The results also reveal that the development tasks in BCS projects are primarily assigned on voluntary basis, which is the usual task assignment practice for OSS projects.

Conclusions Our findings indicate that standard software engineering methods including testing and security best practices need to be adapted with more seriousness to address unique characteristics of blockchain and mitigate potential threats

Citation

Understanding the Software Development Practices of Blockchain Projects: A Survey

Partha Chakraborty, Rifat Shahriyar, Anindya Iqbal, Amiangshu Bosu

10.1145/3239235.3240298

1.4 Choose Reasons

[1.]It's a nationwide survey study in Switzerland and the whole study process is representative in type of survey. The main body of survey and its subject is interesting to identify. Both papers has clear structures and procedures, with carefully designed survey questions and catiously chosen participants. The characteristic of survey is obvious and easy to identify. Both two articles have clear survey architecture. Both two articles used self-administrated questionnaire, then we can compare their detailed design, such as sample selection, data analysis, which can improve our understanding towards questionnaires-based survey.

2 Study Analysis and Comparison

2.1 EASE[11]

2.1.1 Methodological characteristics

In table 2.

表 2: EASE[11] - Methodological characteristics

Characteristics	study
type	Cross-sectional
Objective	We wanted to find out if satisfaction depends on the applied development method, and, more concrete, how satisfaction relates to other elements in the development process, including the use of various practices, and the influences on business, team and software issues.
format	Self-administrated questionnaires
questions	closed questions with few open questions
type of answer	yes/no answers + textual answer + numeric values
instrument evaluation	not mention
response rate	142 companies and 185 IT professionals filled out the complete survey. Response rate: 18.16% 62.00% & Completion rate: 10.15% 31.19% for company survey and anonymous survey
sampling	Cluster-base sampling
length	not mention
study	quantitative analysis + qualitative analysis

2.1.2 Validity

External Validity

One is the coverage of the survey. We found that the company types and job roles reflected our software industry well, but more careful coverage would be beneficial, especially to attempt representative balance across organizations and domains.

Internal Validity

Similarly, we must be cautious because the data is self-reported, and indeed self-selected. For example, it is possible that professionals might be more likely to self-select if they were interested in, or even advocates of, Agile methods. In future would we might be better to include questions to detect such bias to improve the validity of our results.

Finally, and of particular importance to the topic of this paper, is that we cannot assume correlation reflects causality.

Construct Validity

Our emphasis on Agile methods might also dissuade proponents of more planned approaches from participating, so we should be especially hesitant about any negative findings about planned approaches.

2.2 EMSE[39]

2.2.1 Methodological characteristics

In table 3.

表 3: EMSE[39] - Methodological characteristics

Characteristics	study
type	Cross-sectional
Objective	To help alleviate some of the barriers in the SLR search phase
format	Self-administrated questionnaires
questions	closed questions with few open questions
type of answer	yes/no answers + textual answer + numeric values
instrument evaluation	focus groups
response rate	566 potential respondents out of 638 identified authors 297 clicked on the survey link (52%) 91 began the survey (33%) 131 completed the survey (23%)
sampling	Cluster-base sampling
length	not mention
study	qualitative analysis

2.2.2 Validity

External Validity

The main threat to external validity is the representativeness of the sample. It is possible that we missed SLR authors if they published their SLR in a venue other than the ones we searched. Since we had a broad sample, this threat does not appear to be serious in this survey.

Internal Validity

it is possible that some of the recipients shared the survey with students or colleagues who did not have

the requisite experience. Therefore, we asked respondents to report their experience with SLRs. Based on that data, this threat is not present.

We had no way of assessing the severity of each barrier or priority of each requirement. We used frequency of report as a proxy for importance, but future studies should focus specifically on identifying priorities.

Construct Validity

The most critical construct validity threat was that survey respondents may have misunderstood the survey questions and provided incorrect information. To reduce this threat, we pilot tested the survey with experts and with software engineering students (who had performed SLRs themselves) and updated the survey based on their feedback. Also our analysis of the responses indicates that overall the participants understood the questions as we had very few answers that were completely irrelevant.

2.3 EMSE[41]

2.3.1 Methodological characteristics

In table 4.

表 4: EMSE[41] - Methodological characteristics

Characteristics	study
type	Cross-sectional
Objective	To help understand whether or not developers understand the open source licenses they use
format	Self-adminstrated questionnaires and interviews
questions	closed questions with few open questions
type of answer	yes/no answers + textual answer + response categories
instrument evaluation	focus groups
response rate	825 started the survey with 45% completing the survey;
sampling	Cluster-base sampling
length	not mention
study	quantitative analysis + qualitative analysis

2.3.2 Validity

External Validity

Another issue we faced in the design of the survey was the specificity to provide in the scenarios posed

in the survey questions. As some of the participants noted, the wording of the scenarios had some ambiguity.

Content Validity

The survey has limitations with regards to content validity, which considers the degree to which the survey investigates developers knowledge of open source license use. The survey is limited in what can be concluded about the knowledge of individual open source licenses.

Construct Validity

The survey provided links to the licenses referred to in the survey but did not require participants to answer questions to validate their understanding of individual licenses, which may have affected the construct validity of the survey.

The construct validity may also have been affected by the particular three licenses we chose to use in the survey.

2.4 ESEM[22]

2.4.1 Methodological characteristics

In table 5.

表 5: ESEM[22] - Methodological characteristics

Characteristics	study
type	Cross-sectional
Objective	To understand the development processes and practices in New Zealand. Particularly interested in methods and practices used in such environment, the implementation technologies software development professionals use, how professionals ensure software quality, and how they manage software release processes.
format	self-administrated questionnaire(web based)
questions	21 closed questions with one open question in the end
type of answer	numeric values + response categories + yes/no answers + ordinal scales
instrument evaluation	focus groups: The authors of the questionnaire were one industry practitioner (lead) and one researcher. We also asked several practitioners and researchers not involved in the research to review the questionnaire.
response rate	received 101 complete responses.
sampling	Cluster-base sampling
length	the questionnaire contains totally 22 questions
study	descriptive statistics and quantitative data analysis

2.4.2 Validity

External Validity

Regarding external validity, it is difficult to claim statistical generalizability of our findings, given that our sample included 101 respondents. We cannot claim that those who responded are representative of the entire software engineering industry in New Zealand. Given an approximate number of 27,000 software professionals in New Zealand (in technical and non-technical roles), we surveyed a rather small subset. Also, many of the findings (a majority of them) are not statistically significant. More participants are needed to draw more statistically significant conclusions.

Internal Validity

Regarding internal validity, we tried to account for confounding factors that impact our results and causal relationships by conducting several statistical analyses. Also, by giving options to respondents, we did not need to determine what respondents might have meant by an answer.

Construct Validity

Regarding construct validity, the questionnaire might not provide enough details for our research questions. While interviews would have allowed more in-depth discussions and targeted questions about the use of practices, we chose an online questionnaire because a) it allowed us to obtain data from a larger number of respondents since participants could complete the questionnaire independent from us, b) it allowed us to collect data from practitioners independent from their location and without time synchronization, and c) it reduced potential errors in data that could occur when manually entering data from paper-based questionnaires. Also, it may have been possible that the questionnaire's terminology was difficult to comprehend. Furthermore, respondents may have interpreted questions or answer options differently from the intended interpretation.

2.5 ESEM[12]

2.5.1 Methodological characteristics

In table 6.

表 6: ESEM[12] - Methodological characteristics

Characteristics	study
type	Cross-sectional
Objective	aim to carry out the first formal survey to explore the software engineering practices including requirement analysis, task assignment, testing, and verification of blockchain software projects
format	self-administrated questionnaire
questions	9 closed questions and 4 open questions
type of answer	numeric values + response categories + textual answers
instrument evaluation	focus group: To help ensure the understandability of the survey, we asked Computer Science professors and graduate students with experience in SE and experience in survey design to review the survey to ensure the questions were clear and complete.
response rate	Sent the survey to 1,604 active BCS developers, received 200 responses(13%), and they used 156 responses for analysis
sampling	Cluster-base sampling
length	13 questions
study	qualitative analysis

2.5.2 Validity

External Validity

The primary threats to validity are related to sample selection. The respondents of our survey may not adequately represent all BCS developers. While our respondents come from 61 different BCS projects, they primarily represent the top ones. Therefore, some of the opinions, especially, the software development practices used by BCS developers in smaller projects may be different from those included in this study. However, the software development practices mentioned by our respondents cover a broad range, so software development practices in smaller BCS projects may be similar to the practices our study revealed.

2.6 Comparison

Type All of papers are cross-sectional type. It means their participants are asked for information at one fixed point in time.

Format All of papers use self-administrated questionnaire. Then, EMSE[41] also used interviews.

Questions These paper used some closed questions and some open questions to compose their questionnaire. All of the papers in EASE and EMSE didn't mention the number of question they asked in surveys but the two papers in ESEM mention this.

Type of answer These papers didn't used only one type of answers. The 'numeric values' way was used in four papers ESEM[22], ESEM[12], EASE[11], and EMSE[39]. The 'yes/no answers' way was used in four papers ESEM[22], EASE[11], EMSE[39], and EMSE[41]. The 'response categories' way was used in three papers ESEM[22], ESEM[12], and EMSE[41]. The 'textual answer' way was used in three papers ESEM[12], EASE[11], EMSE[39], and EMSE[41]. The 'ordinal scales' way was used in one paper ESEM[22].

Response Rate Most of these paper split response rate to two indicator, response rate and completion rate. The response rate means interviewees start surveys and answer some questions. Then, the completion rate was the percentages that interviewees complete this survey. The highest response rate in these five papers is 62% in EASE[11], and the lowest rate is 13% in ESEM[12]. Obviously, The completion rate in these five papers are lower than 50%. The most highest completion rate is 45 % in EMSE[41], and the lowest rate is 9%(the surveys they finally used) in ESEM[12].

Sampling All of papers used Cluster-base sampling.