

## Projects VS Continuous Product Development – Does it Affect Benefits Realization?

Sinan S. Tanilkan

*Center for Effective Digitalization of the Public Sector  
Simula Metropolitan*

Pb 4, St.Olavs Plass, Oslo, Norway

Email: sinan@simula.no

0000-0003-4216-5172

Jo E. Hannay

*Center for Effective Digitalization of the Public Sector  
Simula Metropolitan*

Pb 4, St.Olavs Plass, Oslo, Norway

Email: johannay@simula.no

0000-0002-8657-7593

**Abstract**—Software investments are traditionally implemented using project organization, which often leads project participants to focus on time, cost and scope, rather than the intended benefits of the investment. We conducted a survey to compare work organized as projects against work organized as Continuous Product Development (CPD). Our results indicate that: 1. Both project organization and CPD are commonly used in practice. 2. Agile is very popular, but DevOps and the use of linear models for organizing work are also frequent. 3. CPD is perceived to outperform projects in realization of benefits. 4. We found no difference in perceived realization of benefits between those using or not using a set of ways of organizing work (including linear models, agile, DevOps, BizDev or program organization). We conclude that organizing work using CPD is a viable alternative to project organization, especially in situations where failure must be avoided. Also, we suggest that more research should be conducted to better understand what factors of the different ways of organizing work affects the realization of benefits.

**Keywords**— *Software Project; Continuous Product Development; Benefits Realization; Agile; DevOps; BizDev.*

### I. INTRODUCTION

Traditionally, software investments are implemented through projects or programs – sometimes managed using portfolio management. Once a project is finished, the solution is transferred to IT operations, who takes over responsibility for the solution, including its maintenance. This approach fits nicely into financial management, where the temporary organization of projects are considered capital expenditure, and the continuous maintenance done by IT operations are considered operational expenses.

Although there are clearly organizations that are successful in developing software solutions using project organization [1], many studies raise concerns about the low degree of success in software projects [2]–[4].

To better understand project success, Baccarini suggests that  $project\ success = project\ management\ success + product\ success$  [5]. In this way of thinking, project management success is concerned with delivering a project according to the agreed time, cost and scope, while product success is concerned with the realization of benefits of investments. When the temporary project organization is dismantled before realization of the benefits of the product begins, project participants tend to prioritize what they can be measured on [6] – which is project management success [7]. Thus, the way software investments are financed is likely to limit the realization of benefits.

An alternative approach to financing software investments, that seems to be popular when talking with practitioners, is the use of Continuous Product Development (CPD). In CPD, the team or organization is tasked to work on a product, or product area, with no defined end date, often as a solid line organization. The cost side of the investment is managed by the amount of people allocated to the product organization, allowing the people involved to focus on the product and the benefits of the product.

There are two important differences between software investments organized as projects vs. CPD. First, although the success criteria should be the same, they often end up being different. While projects have a tendency to focus on project management success, CPD organizations has to deal with the product's success or failure over time. Second, unpredictably is handled differently. While projects have a defined end date within which the agreed scope should be delivered, CPD is often used to deliver product features continuously, handling changes as they come.

As CPD seems a popular way of organizing software work, we wish to explore the degree of adoption of project and CPD. Also, we wish to explore the degree to which one way of organizing work performs better in realizing benefits of the investment. To explore these topics, we conducted a survey among practitioners in the Norwegian IT-industry.

The next section presents work relevant for the research topic, before the research questions are presented in Section III. The research method is presented in Section IV and the results in Section V. After that, we discuss and conclude.

### II. BACKGROUND

The organizing of software work has evolved over the years driven by a need to make use of the information and understanding that is gained during the software process, for the purpose of meeting stakeholder needs better. Even the earliest process models had a focus on producing a system that is useful for the customer.

One of the earliest process models for software development was described in 1956 [8], later coined the waterfall model [9]. The waterfall model is a linear model, where requirements are communicated clearly in the beginning of a project, and the project is planned from initiation to completion, providing stability, structure and predictability [10]. In 1970, Royce [11] suggested that understanding gained in one phase, can result

in the need to redo work from a previous phase. Royce's take on this problem, was that moving back to previous project phases is costly, and should be avoided by better preparations.

Iterative process models take a different approach to changing understanding. Rather than considering changed understanding as a deviation from the plan, iterative models are designed so that understanding acquired in one iteration of software development, can be utilized in the following iterations. Rather than trying to eliminate the need for adaptation, iterative approaches are designed to handle change proactively, rather than reactively. Although iterative organization of software work has reportedly taken place as early as 1957, the first publications on the topic were only due in 1988 [12].

Although early iterative models helped practitioners to utilize new understanding acquired during development, feedback from users often came late, due to infrequent, perhaps only yearly, releases. From around the year 2000, release cycles started to shorten, and by 2010 companies were releasing software multiple times a day [13]. It is likely that this change has been aided both by agile development and management, and the emergence of approaches such as DevOps.

Agile software development is all about feedback and change [14]. This is aided by principles such as "early and continuous delivery of valuable software", "welcoming changing requirements", and "business people and developers working together daily" [15]. Projects using agile practices, have been found to see themselves as more successful in realizing benefits than other projects [16], especially those with flexible scope and frequent delivery to the client.

DevOps, and later BizDev, were designed to increase information sharing and collaboration among organizational units: "DevOps integrates the two worlds of development and operations, using automated development, deployment, and infrastructure monitoring. It's an organizational shift in which, instead of distributed siloed groups performing functions separately, cross-functional teams work on continuous operational feature deliveries" [17]. This means that those who develop become integrated into where benefit are experienced, which ostensibly, should foster increased understanding of benefits during development. Moreover, just as DevOps brings the organizational units responsible for development and operations closer together, BizDev suggests to bring those making business decisions closer to those developing software solutions, through continuous planning and continuous budgeting. On the face of it, this is perfect for evolving and utilizing understanding of benefits. Combining DevOps and BizDev into BizDevOps [18] would seem better still.

Practices such as agile, DevOps and BizDev fit nicely into CPD organization, because the practices are geared towards continuously learning and delivering value, without the constraints imposed by project organization and more linear approaches. An approach that shares many similarities with CPD is Continuous Software Engineering (CSE) [18]. In CSE it is suggested that software engineering should be considered as a set of continuous processes, including continuous planning, budgeting, integration, delivery, deployment, verification,

testing, compliance, security, evolution, use, trust, run-time monitoring, improvement, innovation and experimentation. While all of these processes fit nicely into CPD, they are not a requirement for CPD. What CPD adds to the picture is the explicit lifecycle-focus on product and the discard of the time-bounded project as organizational form.

### III. RESEARCH QUESTIONS

Our first objective is to understand how common different ways of organizing software work are. We look into the adoption of project vs. CPD, and the adoption of ways of organizing, such as the use of linear models, agile, DevOps, BizDev and program organization.

Our second objective is to understand if there are differences in the perceived realization of benefits when using the above ways of organizing.

We pose the following research questions:

- RQ1 How common is CPD compared to project organization?
- RQ2 How common are the following ways of organizing work: linear model (waterfall, v-model, etc.), agile, DevOps, BizDev and program organization?
- RQ3 Is the realization of benefits perceived to be higher in work organized as project or as CPD?
- RQ4 Are there differences in the perceived benefits among those organizing work using linear models, agile, DevOps, BizDev or program organization?

### IV. RESEARCH METHOD

We conducted a survey consisting of an online questionnaire to address the research questions. Data was collected during a webinar titled *Digitalization as Continuous Product Development* in June 2021. During this webinar selected IT-professionals presented experiences and reflections on CPD.

Respondents were asked to base all answers on the latest IT system product development they had taken part in (either organized as project or CPD), where the product or part of the product had been taken into use. As discussed in [19], selecting the last project (in this case the last IT product development) "... reduces the risk that the sample of projects is biased towards the most successful or the largest software projects ..." [19]. This is relevant, because we want to compare characteristics of the reported work with perceived success in realizing benefits.

Comparing the sizes of projects and CPD is not straightforward, since CPD does not have a defined end-date, and consequently not a comparable amount of man-hours to compare with projects. Nevertheless, for a description of the size of the work, we asked respondents to provide the number of people involved in the work at most. Then, when including size in the analysis, we used the categories of organization sizes proposed by the European Commission [20].

#### A. Survey Questions

The survey questions most relevant to answering the research questions are presented in Table I. A complete list of survey questions and responses are available at [21].

TABLE I: SURVEY QUESTIONS

Question	Answer options
SQ1 How was this work organized?	Select one: -Project -CPD -Other, explain: [text field]
SQ2 Approximately how many people was/are actively involved in the work at most? - both from the product owner side and the product developer side	[text field]
SQ3 By and large, how well do you consider that you succeeded in realizing benefits as a result of the product?	Five-point ordinal: (Very Successful 1–5 Very Unsuccessful + 6 Don't know)
SQ4 Was the product owned by public or private sector?	Select one: -Public sector -Private Sector
SQ5 Which ways of working/organizing was used?	Multiple choice: -Linear model (waterfall, V-model, etc.) -Agile -DevOps -BizDev -Organized as program -Other: [text field]

### B. Respondents and Response Rate

A total of 140 people were present at the seminar at the point in time when the survey started. Of these, 131 participated in the survey, but 19 stopped after providing demographic data only. These 19 are not included in the survey results. In total  $n_{\text{partial}}=112$  (85%) people completed the first three pages of the survey, and  $n_{\text{complete}}=94$  (72%) finished the entire survey. The size of work reported on are in the following categories: 20% micro (<10 people involved), 55% small (10–49 people involved), 20% medium (50–249 people involved) and 4% large (>250 people involved).

Figure 1 shows an overview of the success in realizing benefits. Compared to what is reported in similar studies (see [19]), it seems that respondents in this survey reported on work that is more successful in realizing benefits.

Among the  $n_{\text{partial}}$  respondents, 36% represented the product owner side exclusively, 49% represented the product developer side exclusively, while 15% represented both sides. The respondents' average experience with creation of digital solutions ranged from under a year to 50 years, with a mean of 17.6 years and median of 20 years. The number of years

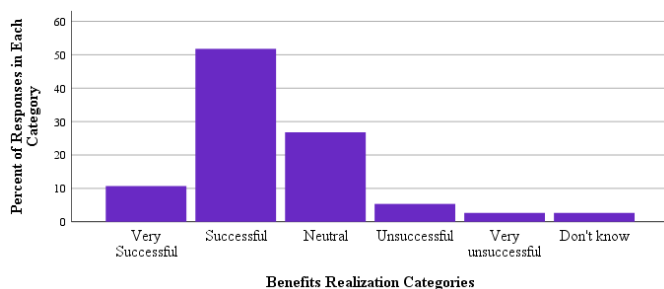


Fig. 1: Degree of Realization of Benefits

of experience as a manager in this field ranged from one to 30 years, with a mean of 10.5 years and median of 9 years. Further, 51.8% of the respondents reported to organize work as projects, while 40.2% reported to organize work as CPD, and 8% reported to organize work as “other” (most of these were combinations of project and CPD). Finally, 68% of the respondents reported on products owned by the public sector, while 32% reported on products owned by the private sector.

### C. Analysis

For each of the four research questions, we present descriptive statistics for the corresponding survey questions. For RQ3 and RQ4, where we look at relations between variables, we also present significance values and effect sizes for the comparisons.

When comparing projects and CPD with respect to the realization of benefits (RQ3 and RQ4), we exclude data from work organized as “other” and data where the respondents reported not to know the degree of benefits realization. We use a t-test to calculate significance values and effect sizes [22]. Although the data is not normally distributed, the sample size is large enough for the t-test even with a skewed sample (condition:  $n \geq 40$  [22, p. 516], current sample:  $n_{\text{project}}=56$ ,  $n_{\text{CPD}}=45$ ). For effect sizes we use Cohen's  $d$  with the following rules of thumb [23]: <0.1 (very small), 0.1 – <0.3 (small), 0.3 – <0.5 (medium), 0.5 – <1.2 (large), 1.2 – <2.0 (very large) and  $\geq 2.0$  (huge). We use a two-tailed test [22], because we make no assumptions on projects performing better or worse than CPD.

When comparing other ways of organizing work with respect to the realization of benefits, we exclude only the data where respondents reported not to know the degree of benefits realization. Because some of the ways of organizing work have few reported occurrences, we use the Fisher's exact test [24] to calculate significance values.

## V. RESULTS

### A. Organization of work (RQ1 and RQ2)

Figure 2 summarizes the results on the different ways of organizing work. Figure 2a shows the data from SQ1, where we see that project organization is more common than CPD. Those selecting the “other”-category, were combining project organization and CPD, or using program organization; often organizing work using CPD-like aspects within a project.

Figure 2b shows work organization (SQ1) differentiated by size (SQ2). We observe that the micro and large endeavors use CPD more often than project organization. When looking at the small- and medium-sized endeavors, project organization is more common than CPD. It is interesting to note that large endeavors observed here are either organized as CPD or using program organization (reported as other, with program organization written in the freetext field).

Figure 2c shows differences between endeavors in the public sector versus the private sector (SQ4) in selection of work organization (SQ1). Visual inspection indicates no substantial

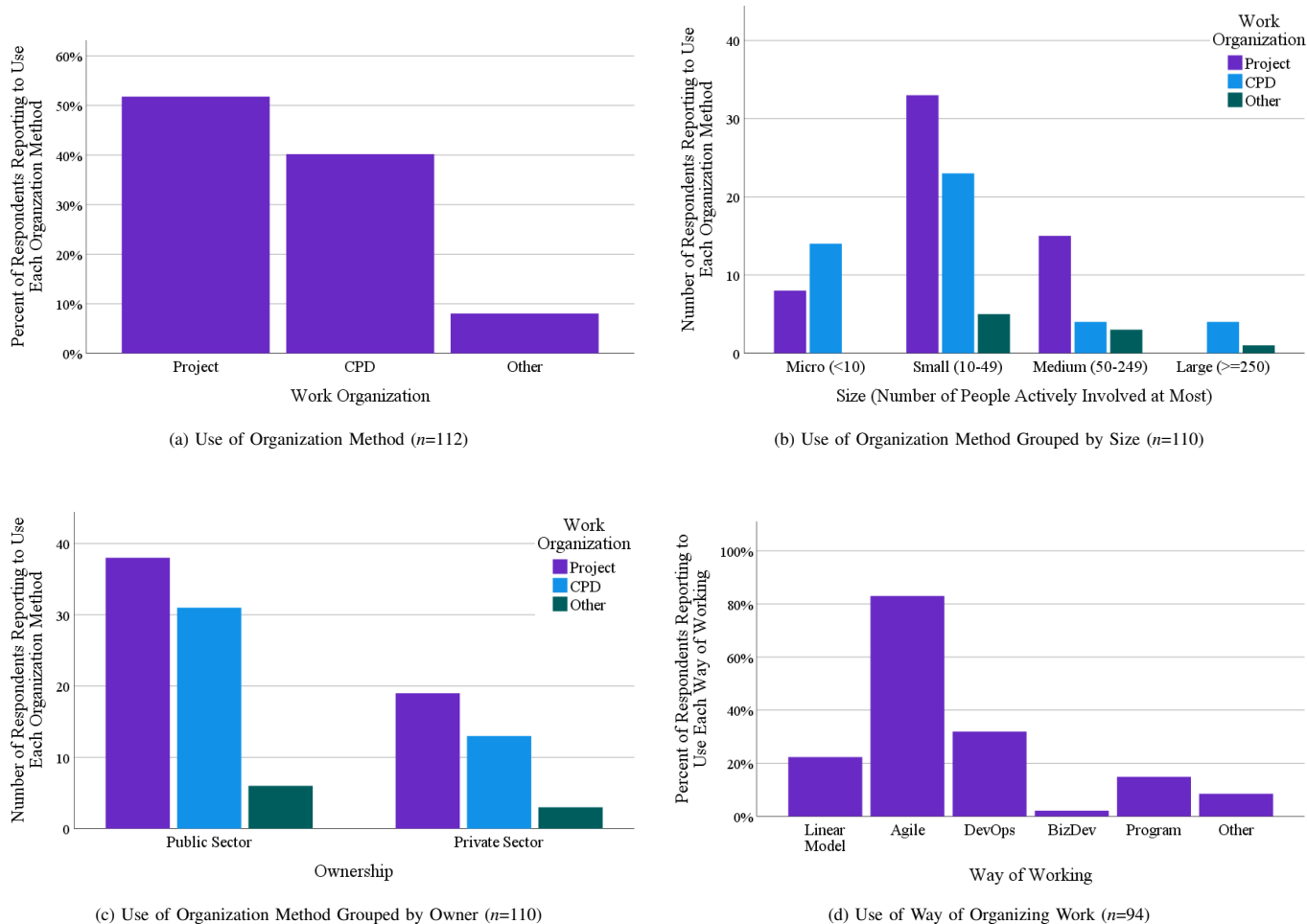


Fig. 2: Organization of work

differences between work organization in efforts owned by the public versus the private sector.

Figure 2d shows which ways of organizing are most commonly applied (SQ5). SQ5 is a multiple choice question, allowing respondents to select several ways of organizing work. Respondents report the use of agile to be very common, with 84% of respondents reporting to use agile in their latest completed endeavor. DevOps is the second most common way of working (32% adoption). Interestingly, linear models are also somewhat frequently used (23% adoption), and 21.5% of those using agile, also use a linear model.

*B. Comparing the Use of Practices with Realization of Benefits (RQ3 and RQ4)*

Table II shows the realization of benefits (SQ3) when organizing work as projects versus CPD (SQ1), by percentage of responses in each benefits realization category. Our data shows that work organized as CPD is perceived to be significantly (two-tailed t-test  $p=.020$ ) more successful in realizing benefits than work organized as projects. The effect size of the comparison is Cohen’s  $d=0.475$ , which is considered a medium effect size. Although this does suggest that CPD

outperforms projects, there are other takeaways from Table II worth highlighting:

- 55% of the projects were reported to be successful or very successful (78% for CPD).
- Projects are more distributed on the success-scale than CPD.
- Work organized as CPD was always reported to be neutral or better.

In summary, work organized as CPD was perceived to be more successful than projects, but there are also many successful projects. Among the work reported to use CPD, there were no unsuccessful occurrences.

TABLE II: PERCEIVED REALIZATION OF BENEFITS FOR WORK ORGANIZED AS PROJECT VS CPD (n=101)

Benefits realization	Project	CPD
Very Successful	~ 12%	~ 9%
Successful	~ 43%	~ 69%
Neutral	~ 29%	~ 22%
Unsuccessful	~ 11%	0%
Very Unsuccessful	~ 5%	0%
Two tailed t-test	$p=.020$	
Effect size	$d=0.475$	

TABLE III: REALIZATION OF BENEFITS FOR EACH WAY OF ORGANIZING WORK ( $n=93$ )

<i>Benefits realization</i>	<i>Linear</i>		<i>Agile</i>		<i>DevOps</i>		<i>Program</i>	
	No (n=71)	Yes (n=22)	No (n=14)	Yes (n=79)	No (n=63)	Yes (n=30)	No (n=80)	Yes (n=13)
<i>Very Successful</i>	13%	9%	7%	13%	14%	7%	13%	8%
<i>Successful</i>	58%	50%	57%	56%	51%	67%	55%	62%
<i>Neutral</i>	24%	27%	29%	24%	27%	20%	26%	15%
<i>Unsuccessful</i>	5%	5%	0%	6%	6%	3%	5%	8%
<i>Very Unsuccessful</i>	0%	9%	7%	1%	2%	3%	1%	8%
<i>Fisher's exact test</i>	$p=0.208$		$p=0.594$		$p=0.589$		$p=0.462$	

Table III shows comparisons between the other categories for organizing work (SQ5) with respect to benefits realization (SQ3). Using the Fisher's exact test (bottom row in Table III), we see that our data does not show significant differences in the realization of benefits between those using or not using any of the ways of organizing work. Thus, our data does not support the current trends in software engineering, where linear models are considered inferior, and agile and DevOps has become very popular.

## VI. DISCUSSION

The observation that CPD outperforms projects when it comes to the realization benefits, is relevant for several groups.

First, it is relevant for those making investment decisions. By organizing investment into projects there might be a perceived predictability of time, cost and scope, but this seems to come at the expense of reduced benefits realization.

Second, it is relevant for those working to create software products and realize the benefits of those products. Organizing work in a manner that increases the probability of realizing benefits, helps team members achieving the purpose of their work, which is likely to provide improved job satisfaction.

Third, it is relevant for researchers, because it raises the awareness that several ways of organizing IT development and lifecycle work can lead to success; even though some modes of organizing work are currently in vogue. Researchers should not become evangelists for one approach or the other. Rather, one should conduct research that helps us understand the characteristics of situations where various ways of organizing work – and in particular, project or CPD – is most suitable.

We are puzzled to observe that neither the use, nor non-use, of linear models, agile, DevOps or program organization, had any significant effect on the realization of benefits. Given that further studies with larger power (see next section) corroborate this, one might contrast this to what seems to be the mainstream opinions in the software industry, where linear models are considered bad, and agile is very popular. We speculate that this could be due, either to practitioners making good choices about various ways of organizing work, or, conversely, that practitioners are not successful in using the different ways of organizing work. Both of these situations could explain the lack of differences, and more in-depth studies are called for to unravel the connections between various nuances in ways of organizing work and success in benefits realization. In the mean time one might speculate as follows:

Looking at the results under the assumption that practitioners are making good choices of ways of organizing work, one can speculate that practitioners employ linear models in situations of low uncertainty (where this approach would fit) and agile approaches when there is more uncertainty. This assumption is supported by the observation that the overall degree of realization of benefits reported here, is higher than in similar studies [19] (see Section IV-B and Figure 1).

Alternatively, looking at the results from the view that practitioners are not able to utilize the different ways of organizing work, could help explain why using agile, DevOps and program organization does not seem to lead to higher realization of benefits. This view finds support in challenges reported when introducing agile, DevOps and program organization. Introducing new ways of working can be a challenging task [25], including resistance to change [26] and pressure to use traditional approaches [27]. For DevOps, there is lacking consensus on the best way of organizing work to ensure collaboration between development and operations [28].

## VII. LIMITATIONS

### A. Statistical Conclusion Validity

The low number of respondents (112) in this survey, gives low statistical power with a low probability of observing significant results that are actually present in the population. Replicating studies using a larger sample may find effects that were not uncovered in our data.

It is possible that a webinar on CPD attracts people who are already using CPD. If this is the case, the percentage of respondents using CPD would be higher here than in the population of software product development initiatives. However, the resulting near equal group sizes for projects versus CPD was beneficial for the purpose of answering our research questions via the present survey.

### B. External Validity

Based on demographic data in this study, one can generalize the results to populations with similar characteristics. For our sample, this can be problematic for three reasons:

- 1) We have limited demographic information of the work reported on in this study. Limited information was collected due to time limitations duration the webinar.
- 2) It is possible that practitioners attending a seminar on CPD think differently about organization of software work than others. As such, it is likely that the number of respondents reporting to organize work using CPD is

higher in our sample than in the population of software development endeavors in the industry.

- 3) A webinar and survey conducted in Norwegian, limits the effects of cultural difference, and difference in respondents background. It is possible that respondents with a different background would yield different results.

### C. Construct Validity

A recent study of the use of linear models and agile methodology [29] found that work organization only accounts for 40% of observed activities in organizations, while the remaining 60% are a result of method incompleteness, peoples skills and habits, organizational noise and similar factors. Thus, when practitioners report to use a linear model, agile, DevOps, BizDev or program organization, it is not clear exactly to what extent, or how, these ways of organizing are implemented.

We observed that the number of respondents reporting to use BizDev was very small. Our impression after talking with practitioners, is that many of the practices in BizDev are commonly used. If this is the case, it seems BizDev is not known to practitioners, resulting in a low count in the survey.

## VIII. CONCLUSION AND FURTHER RESEARCH

Based on the above results and discussion, we conclude that CPD is a very viable alternative to organizing software investments as projects, especially in situations where failure to realize benefits must be avoided. Also, we believe that more research is needed to understand in what situations practitioners would benefit from selecting either project or CPD organization, or a combination of both.

We did not find any evidence for or against the use of linear models, agile, DevOps, BizDev or program organization. Still, we believe that all of these has previously been used to realize the benefits of software investments successfully. We suggest that more research is needed to understand better these ways of organizing work, especially focusing on how the different ways of organizing work perform in different situations. This we hope will provide practitioners with actionable guidance on the selection of ways of organizing work.

### ACKNOWLEDGMENTS

The authors are grateful to the survey respondents for their time and experience.

### REFERENCES

- [1] R. Berntsson-Svensson and A. Aurum, "Successful software project and products: An empirical investigation," in *Proceedings of the 2006 ACM/IEEE international symposium on Empirical software engineering*, pp. 144–153, 2006.
- [2] S. Goldfinch, "Pessimism, computer failure, and information systems development in the public sector," *Public Administration Review*, vol. 67, no. 4, pp. 917–929, 2007.
- [3] A. Bharadwaj, M. Keil, and M. Mähring, "Effects of information technology failures on the market value of firms," *The Journal of Strategic Information Systems*, vol. 18, no. 2, pp. 66–79, 2009.
- [4] K. Conboy, "Project failure en masse: a study of loose budgetary control in isd projects," *European Journal of Information Systems*, vol. 19, no. 3, pp. 273–287, 2010.
- [5] D. Baccharini, "The logical framework method for defining project success," *Project management journal*, vol. 30, no. 4, pp. 25–32, 1999.
- [6] E. Bouwers, J. Visser, and A. Van Deursen, "Getting what you measure," *Communications of the ACM*, vol. 55, no. 7, pp. 54–59, 2012.
- [7] S. S. Tanilkan and J. E. Hannay, "Benefit considerations in project decisions," in *International Conference on Product-Focused Software Process Improvement*, pp. 217–234, Springer, 2022.
- [8] R. Kneuper, "Sixty years of software development life cycle models," *IEEE Annals of the History of Computing*, vol. 39, no. 3, pp. 41–54, 2017.
- [9] T. E. Bell and T. A. Thayer, "Software requirements: Are they really a problem?," in *Proceedings of the 2nd International Conference on Software Engineering*, pp. 61–68, 1976.
- [10] T. Thesing, C. Feldmann, and M. Burchardt, "Agile versus waterfall project management: Decision model for selecting the appropriate approach to a project," *Procedia Computer Science*, vol. 181, pp. 746–756, 2021.
- [11] W. Royce, "Managing the development of large software systems: concepts and techniques," in *Proceedings of the 9th International Conference on Software Engineering*, pp. 328–338, 1987.
- [12] C. Larman and V. Basili, "Iterative and incremental developments. a brief history," *Computer*, vol. 36, no. 6, pp. 47–56, 2003.
- [13] J. Bosch, *Continuous software engineering*, ch. Continuous Software Engineering: An Introduction, pp. 3–13. Springer International Publishing, 01 2014.
- [14] L. Williams and A. Cockburn, "Agile software development: it's about feedback and change," *Computer*, vol. 36, no. 6, pp. 39–43, 2003.
- [15] M. Fowler *et al.*, "The agile manifesto," *Software development*, vol. 9, no. 8, pp. 28–35, 2001.
- [16] K. K. Holgeid and M. Jørgensen, "Benefits management and agile practices in software projects: how perceived benefits are impacted," *IEEE 22nd Conference on Business Informatics (CBI)*, vol. 2, pp. 48–56, 2020.
- [17] C. Ebert, G. Gallardo, J. Hernantes, and N. Serrano, "Devops," *IEEE Software*, vol. 33, no. 3, pp. 94–100, 2016.
- [18] B. Fitzgerald and K.-J. Stol, "Continuous software engineering: A roadmap and agenda," *Journal of Systems and Software*, vol. 123, pp. 176–189, 2017.
- [19] M. Jørgensen, "A survey on the characteristics of projects with success in delivering client benefits," *Information and Software Technology*, vol. 78, pp. 83–94, 2016.
- [20] European Commission and Directorate-General for Internal Market, Industry, Entrepreneurship and SMEs, *User guide to the SME definition*. Publications Office, 2020.
- [21] S. S. Tanilkan, "Organization as project vs continuous product development." <https://tinyurl.com/pvscpd>. Accessed: 2023-04-24.
- [22] D. S. Moore and G. P. McCabe, *Introduction to the Practice of Statistics*. W. H. Freeman and Company, 2001.
- [23] S. S. Sawilowsky, "New effect size rules of thumb," *Journal of Modern Applied Statistical Methods*, vol. 8, no. 2, pp. 596–599, 2009.
- [24] S. Weerahandi, *Exact statistical methods for data analysis*. Springer Science & Business Media, 2003.
- [25] Z. Shehu and A. Akintoye, "Major challenges to the successful implementation and practice of programme management in the construction environment: A critical analysis," *International Journal of Project Management*, vol. 28, no. 1, pp. 26–39, 2010.
- [26] T. J. Gandomani, H. Zulzalil, A. A. Ghani, A. B. M. Sultan, and K. Y. Sharif, "How human aspects impress agile software development transition and adoption," *International Journal of Software Engineering and its Applications*, vol. 8, no. 1, pp. 129–148, 2014.
- [27] C. de O. Melo *et al.*, "The evolution of agile software development in brazil," *Journal of the Brazilian Computer Society*, vol. 19, no. 4, pp. 523–552, 2013.
- [28] L. Leite, C. Rocha, F. Kon, D. Milojicic, and P. Meirelles, "A survey of devops concepts and challenges," *ACM Comput. Surv.*, vol. 52, pp. 1–35, Nov. 2019.
- [29] B. V. Thummadi and K. Lyytinen, "How much method-in-use matters? a case study of agile and waterfall software projects and their design routine variation," *Journal of the Association for Information Systems*, vol. 21, no. 4, pp. 863–900, 2020.