

## Spotlight on the Positives: How Do Information Technology Projects Achieve Cost Underruns?

Adam Alami  
IT University of Copenhagen  
[adaa@itu.dk](mailto:adaa@itu.dk)

Christian Østergaard Madsen  
IT University of Copenhagen  
[chrn@itu.dk](mailto:chrn@itu.dk)

Oliver Krancher  
IT University of Copenhagen  
[olik@itu.dk](mailto:olik@itu.dk)

### Abstract

*The Information Technology (IT) project literature has extensively studied cost overrun. But there is surprisingly little work on cost underrun. In this paper, we explore twelve IT projects that achieved cost underrun, using interviews and documents analysis. Our findings show that projects achieve cost underruns when they actively implement practices to enhance process efficiency. These practices are: Implementing a collaborative environment, budgetary control, capitalizing on previous knowledge, continuous learning during the project, and accommodating uncertainty. Projects also encounter conditions that contribute directly to cost underrun, like scope reduction and savings in spending. Our key contribution lies in uncovering, based on a unique sample, how a set of specific practices can lead to cost underrun.*

### 1. Introduction

Information technology (IT) projects are “interrelated set of activities” aiming to create unique IT-related outcomes such as software products, hardware installation, and business process improvement [1]. IT project management is concerned with delivering these IT-related outcomes within the agreed parameters (i.e., budget, scope, schedule and quality) [1]. IT projects enable growth, transformation and business processes streamlining, but they are often complicated by uncertain requirements, knowledge boundaries between engineers and users, and organizational and technological complexity [2]. The definition of success in project management has not been entirely unlocked [3]. The discussion of project success is 60 years old and is an evolving concept [4]. Three metrics remain popular in measuring a project’s success, cost, time and meeting business specifications [4, 5], also known as the triple constraint, iron triangle, or three-legged stool of project management [3, 5]. Although researchers and practitioners have focused

on these three and other metrics of success to various degrees [6], a key dimension of success concerns the costs of a project [7, 8]. Cost overrun is commonly used in the IT projects literature to assess their performance [7]. This has been popularized by the Chaos reports (1994 and onwards) [9]. The CHAOS reports have been criticized for being inconsistent with other studies and being influenced by political bias [10]. Budget adherence has historical and rational explanations. Although success is recently viewed broadly, historically meeting budget expectations has been at the core of this view. In addition, budget adherence is perceived as a credible indicator of the economic performance of IT projects.

A key issue for practitioners is, thus, how to manage an IT project such that the project is completed within budget. The guidance that the existing literature provides on this issue is limited in two important regards. First, while some studies (e.g., [10, 11, 12]) focus explicitly on cost overrun, most of this work analyzes outcomes in a descriptive manner, with limited attention to the antecedents that contributes to budget overrun versus underrun. Second, although other studies, e.g., [13] examined antecedents of success; but relatively few pay explicit attention to the cost dimension of success (i.e., cost overrun versus underrun), and they rarely focus explicitly on the positive cases, i.e., projects achieving cost underrun. These studies rely on the assumption that the factors contributing to cost overrun versus cost underrun are symmetrical, i.e., that the absence of a factor that causes cost overrun will cause cost underrun. Yet, as is increasingly acknowledged in the social sciences, causality can also be asymmetric [14], i.e., the factors contributing to cost overrun can be different from those contributing to cost underrun. In light of these gaps, our paper examines cost underrun as the outcome of interest, by addressing the following research question:

*How do information technology projects manage to achieve cost underruns?*

We contribute to the knowledge on IT project

management with the results of an empirical analysis of twelve IT projects in the Danish public sector that were completed under budget. Our analysis focuses on the practices and conditions that allowed these projects to achieve cost underruns.

Our findings show that in projects with cost underrun, project managers and their teams share the project’s common goals (including budget) and work toward achieving them. Project managers take positive steps to implement practices to help them to achieve cost underruns. Although, in some cases, conditions such as scope reductions and savings in spending contributed to cost underruns, project managers also attributed this outcome to the practices they put in place to execute the projects.

## 2. Research Background

Several critical success factor (CSF) (e.g., [15, 16, 17, 18]) suggest that factors important for project success include, among others, “Project mission”, having a clear and well-defined goals for the project, “client consultation”, this factor calls for the active engagement of impacted users, and “communication” [16]. Nonetheless, open questions remain with regards to how to implement these factors in specific projects and whether these factors are likely to lead to cost underrun (as opposed to other dimensions of project success). For instance, while both existing research and the popularity of agile methods attest to the criticality of collaboration, it is unclear whether and how the team-based collaboration strategies popularized by the agile movement work in outsourced projects.

After decades of continuous contributions to the CSF work, the intriguing question is how these factors have contributed to elevating the practice of IT project management. Some of these factors have become standardized practices. For example, “project mission” and “project schedule/plans” [16] are basic project

management tools in today’s standards. Others, like “effective change management” [17], may have become less important due to the increased adoption of agile methods, which advocate for accommodating changing requirements [19].

Notwithstanding the insights derived from classic CSF research and from more recent research on cost overrun and project failure, a key issue is that existing work does not focus on positive cases. Most of the work correlates potential CSF with variance in cost or schedule overrun, assuming that if a factor causes cost overrun, the absence of the factor would cause cost underrun. Yet it is not granted that causality is symmetrical. Therefore, it is timely and important to revisit the question of how IT projects achieve positive outcomes. This has three goals: understand modern IT project management capabilities and techniques to deliver positive results, identify techniques and practices used by project managers to achieve cost underruns and address the gap of the deficiency on empirical reporting of positive cases.

## 3. Research Methods

We sourced our data from the Danish national IT projects reporting repository. Since 2011, all Danish government-funded IT projects with a budget of at least ten million DKK ( $\approx$  1.6 million US dollars) are required to comply with a reporting process by the Division for central government ICT management (the division). Under this mandatory reporting scheme, IT projects have to periodically submit reporting documents, including the project initiation document, a business case, half-yearly status reports and a final report upon the completion of the project.

As part of a research collaboration with the division, we were given access to reporting data of 54 IT projects that were completed between 2011 and 2020. The scope of these projects varied, but they all had a software development or tailoring component. We collected the projects’ reports and built a database to store these projects’ information for the purpose of our empirical investigation.

We calculated cost overrun for each project. We followed Flyvbjerg’s [7] recommendation of selecting the earliest point available in the project’s lifecycle, which is the data closest to the “time of decision to build” [7]. To calculate the projects’ cost overrun, we adjusted the realized costs to constant prizes, choosing the project’s initiation year as a baseline. Once the budget data was validated by the division, we calculated the cost overrun as “actual out-turn cost minus estimated costs in percent of estimated costs” [8].

**Table 1. The Study Interviewees**

Cases	Interviewee	Roles	Exp. (Yrs)
C1	I1	Project Manager	14
C2	I2	Project Manager	15
	I3	IT Department Manager	24
C3	I4	IT Project Manager	23
C4	I5	IT Project Manager	25
C5	I6	Chief Adviser	19
	I7	Head of Division	23
C6 & 7	I8	Team Leader	15
	I9	IT Director	23
C8	I10	Project Manager	20
C9	I11	IT Project Manager	23
C10	I12	Project Manager	10
C11	I13	Project Manager	26
C12	I14	IT Project Manager	18
	I15	Service Delivery Manager	8

**Table 2. The Selected Project Characteristics**

Cases	PM Methodologies	Product Acquisition Strategy	Estimated Budget (in Million DKK)	Cost Underrun	Procurement Strategy
C1	Waterfall model	Tailoring off-the-shelf software	19.3 ( $\approx$ 3.15 mil US\$)	-17.06%	Time and materials
C2	Waterfall model	New software development	40.4 ( $\approx$ 6.60 mil US\$)	-43.73%	Time and materials
C3	Agile Method	New software development	27.7 ( $\approx$ 4.52 mil US\$)	-20.03%	Time and materials
C4	Agile Method	Tailoring off-the-shelf software	44.5 ( $\approx$ 7.26 mil US\$)	-30.86%	Fixed price agreement
C5	Agile Method	Tailoring off-the-shelf software	13 ( $\approx$ 1.6 mil US\$)	-31.25%	Time and materials
C6	Waterfall model	New software development	26.3 ( $\approx$ 4.30 mil US\$)	-37.45%	Time and materials
C7	Waterfall model	New software development	96 ( $\approx$ 15.66 mil US\$)	-3.13%	Fixed price agreement
C8	Waterfall model	New software development	43.8 ( $\approx$ 7.14 mil US\$)	-13.10%	Fixed price agreement
C9	Agile Method	New software development	41.9 ( $\approx$ 6.83 mil US\$)	-7.05%	Time and materials
C10	Waterfall model	New software development	106.5 ( $\approx$ 17.38 mil US\$)	-15.90%	-
C11	Hybrid (Waterfall and agile)	Tailoring off-the-shelf software	33.6 ( $\approx$ 5.48 mil US\$)	-26.60%	Time and materials
C12	Hybrid (Waterfall and agile)	New software development	12.1 ( $\approx$ 1.97 mil US\$)	-0.10%	Fixed price agreement

Upon the completion of this analysis, we observed that a considerable number of projects in our sample were completed with cost underruns. While the cost overrun, of the whole sample ranged from -52.3% to 168.7% in the sample, 23 of the 54 projects were completed with cost underruns. To deepen our understanding of this phenomenon (i.e., completing a project with a cost underrun), we opted to investigate a subset of projects in our sample using qualitative methods. Qualitative methods allowed us to delve into the details and the intimate circumstances and conditions that facilitated this outcome [20].

### 3.1. Case Selection

We selected 12 projects from our sample with a cost underrun outcome. Our selection strategy aimed at achieving purposeful sampling [21] to ensure a wide variety of projects representation. This selection strategy allowed us to compare and contrast, i.e., to identify similarities and differences. We purposefully selected cases to represent the variety found in the whole cluster of projects under budget in our sample. To meet our sampling objective, we used four parameters to select our cases: The project management (PM) methodology used by the project, the product(s) acquisition strategy (i.e., whether new development or tailoring an off-the-shelf software), the actual cost underrun realized by the project and the procurement strategy (i.e., fixed price or time and materials). For example, we selected the project with highest and lowest cost underrun respectively (i.e., -43.73% and -0.10%). Table 2 illustrates our selection.

### 3.2. Data Collection

We used two sources of data to investigate the projects in our sample: project reporting documents and semi-structured interviews with project managers and other stakeholders. Each project submitted at

least three documents: a business case, a project initiation document and a project completion report. We examined a total of 39 reporting documents.

We conducted interviews with fifteen participants who had an active responsibility in the delivery of the selected projects. All interviews were conducted using audio-video conferencing tools (Zoom or MS Teams). The interviews were conducted in Danish and lasted between 40 to 60 minutes each. The interviews generated an average of sixteen pages when transcribed verbatim. Table 1 summarizes the interviewees' roles and experience in project management for each selected project. The interview guide is available at here <sup>1</sup>.

### 3.3. Data Analysis

Before to each interview, we examined the project documents to prepare the interview's questions. This exercise helped us to become familiar with the project and prepare detailed questions. After analyzing the interviews' data, we conducted another iteration of documents analysis to identify other possible explanations of the cost underrun.

We used thematic coding to analyze the interviews' data, following the guidelines of Braun and Clarke [22]. Our analytical approach was inductive. Given that the large share of projects with cost underrun was rather atypical according to the existing literature, we believed that an inductive approach was best suited for openly exploring patterns in our particular sample. Codes were derived from the data iteratively using the research question as an analytical lens. Once the responses were coded, patterns were identified, and similar codes were grouped to form themes.

The initial codes were generated using open coding [22]. It is a process by which raw data are systematically analyzed line-by-line to derive meanings and concepts from text. Each of the three authors conducted the initial

<sup>1</sup><https://figshare.com/s/e0864b74138f76fe4d34>

coding separately. Then, we organized two sessions to discuss and compare our codes. At the end of the second session, we had a final list of codes. Then, in the next phase of the analysis, we categorized the codes into themes and categories by comparing codes and looking for similarities among the list of codes. This step yielded a set of themes that provide explanations to our research question.

### 3.4. Validity

We used two techniques to strengthen validity: Analyst triangulation and method triangulation [20]. The three authors conducted the coding of the data separately. We used debriefing sessions to compare and contrast our codes. This exercise resulted in producing a codebook of 27 final codes. Then, the first author proposed an initial list of categorized themes. We organized a second round of debriefing sessions to reach a consensus on the themes. The second and third authors critically and constructively examined the proposed themes. We reached an agreement on the final themes and the naming of the themes after few iterations. For methods triangulation, we used two distinct sources of data: project reporting documents and interviews. Hammersley explains that “by drawing data from sources that have very different potential threats to validity it is possible reduce the chances of reaching false conclusions” [23]. We used this technique to achieve “validity-checking” and also to seek “complementary information”[23]. We asked our interviewees to confirm our findings from the initial document analysis. After the interview data was analyzed, we conducted a second iteration of documents analysis to seek complementary data to support the interviews.

## 4. Findings

Our objective is to understand how the IT projects in our sample achieved cost underrun. We identified several circumstances and conditions, which have contributed to the projects achieving cost underrun. Most importantly, we observed that these projects actively sought to elevate their performance by implementing practices that helped to achieve a cost underrun. We define practices as the application, use of ideas or methods to enhance the ability of the project to achieve cost underrun. Conversely, conditions are occurrences or events not within the control of the project.

We identified two conditions affecting cost underrun: *scope reduction* and *savings in spending*, gained by various means during the course of the project.

These conditions contributed to some extent to bring the state of cost underrun. In some instances, they had a significant influence on determining the cost underrun. However, the projects did not rely solely on these conditions to attain a cost underrun. Project managers actively sought to implement practices to assist them in achieving this particular outcome by enhancing their process efficiency, which is the success of the development process itself (i.e., the extent to which the project was delivered on schedule and within budget). We identified five practices: *Implementing a collaborative environment*, *budgetary control*, *capitalizing on previous business domain knowledge*, *continuous learning during the project* and *accommodating uncertainty*. These practices are courses of action taken by the project managers to achieve a cost underrun. Table 3 illustrates the spread of these conditions and practices in our sample.

### 4.1. Conditions

**Scope reduction:** This condition occurs when the initial and agreed requirements have been reduced (e.g., due to law changes during the project) or a subset of the requirements have been shifted to be delivered by future projects (e.g., to improve the business case of the project). Scope reduction has helped, only three projects (C2, C6 & C11) to achieve a cost underrun. However, the interviewees from these projects stressed that the implementation of several practices besides scope reduction contributed to cost underrun.

**Savings:** Some projects in our sample encountered savings in their spending. Some elements in the projects came in at a lower cost than originally estimated, or there were unused funds in the risk pools at the projects' end. These were contingency funds available for the projects in addition to the estimated costs to cover potential uncertainties. Similarly to scope reduction, the cost underrun was not contributed entirely to savings. The interviewees from the projects also attributed cost underrun to their efforts of implementing practices to drive a better outcome.

### 4.2. Practices

In our data, we have identified several practices to achieve a cost underrun. In this section, we report and discuss these practices.

All the projects in our sample opted for an acquisition strategy that involved an external vendor or multiple vendors. These projects also had the mandate to deploy IT solutions across multiple business units

**Table 3. The conditions and practices and their occurrences in our sample**

Conditions & Practices	Projects in our sample											
	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12
<b>Conditions</b>												
Scope reduction		✓				✓					✓	
Savings in project expenditures			✓		✓	✓	✓	✓	✓	✓		✓
<b>Practices</b>												
Implementing a collaborative environment	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Budgetary control	✓	✓		✓	✓	✓						
Capitalizing on previous business domain knowledge	✓		✓	✓			✓	✓	✓	✓		
Continuous learning during the project				✓			✓		✓	✓		
Accommodating uncertainty		✓						✓	✓			

and sometimes multiple government agencies across Denmark. To achieve the end results, the organizations established temporary organizations and structures to execute the projects. These temporary structures usually lacked uniformity and cohesion. Aware of these limitations, project managers ensured some level of collaboration to reduce the inherent aspects of uniformity.

**Implementing a collaborative environment:** Almost every project in our sample (eleven out of twelve) has implemented some form of collaboration between the supplier(s), impacted business units and the project team to carry out its delivery. The project managers had strong awareness of the potential benefits of collaboration. They took the necessary measures to foster a collaborative work environment for their projects. However, collaboration in these projects was not implemented to the same extent. While some projects managed to build one consolidated team to execute the project, other sufficed with an arm’s length collaboration with the supplier and the relevant business units and users.

*Building one team.* This approach of implementing collaboration (e.g., C3 & C9) relies on unifying all parties to form a consolidated team. Project managers established one team to work together and feel more connected. One project manager explained that collaboration (or “sticking together”) leads to a high performing team: “A lot of focus is on getting these teams to stick together. To become a high performing team.” (C9, I11). To achieve this quality of togetherness, some project managers intentionally collocated all project resources in the same physical location. This project manager explained: “In general, I think it is our experience, that sitting together works really well, because you can just make these clarifications quite quickly, which would otherwise require some writing or you have to plan meetings at certain times.” (C5, I6). This was echoed by another project manager. She stated: “We sat physically located next to each other and the employees from [vendor

name] sat next to us. So all the time, they could just ask us and we could ask them when there was something. And it made the communication much smoother. So I think that was a really, really big advantage.” (C1, I1). I11 further claimed that collaboration in her team facilitated openness: “We managed in many ways to create such an openness.” (C9, I11).

We also observed that projects using agile methods relied more on building one team to meet budget expectations. In addition, agile methods have helped the collaborative aspect of the teams. This interviewee claims agile facilitates “knowing each other”: “It’s clear when you run agile and you have all these meetings, it probably means something different on how well you get to know each other than if you run a little more waterfall-based projects.” (C5, I7).

*Arm’s length collaboration.* Not all projects in our sample were able to build one team. This was mainly because the relationship with the vendor was distant (lacking closeness and familiarity of each other). However, this constraint did not stop them from implementing a less intimate form of collaboration. This project manager describes the relationship with the vendor as being “reasonable”: *And then have such a reasonable collaboration with [the name of the vendor]* (C8, I10). Another project manager discusses collaboration with business users and the vendor being a contributor to the project being under budget. He states: *I think we had a really good collaboration both among us in the [name of a business unit] but also with [name of the vendor] there.* (C1, I1). Various techniques were used to facilitate this form of collaboration without being one team. For example, some interviewees talked about establishing “working groups” which met periodically to exchange information and to manage progress and issues.

Some projects in our sample were aware of the importance of including the business users’ perspectives and inputs. They implemented collaborative teams including the end users. They secured early and ongoing participation of the business users. This participation has given a voice to the users to shape and influence

the product; as explained by this project manager: “We got input quite early from some of our users from the companies, where they said if you have to make such a system, then we really want it to be able to do this and that and that. OK. It was very good to just get involved from the start, so you don’t, yes, do anything else.” (C1, I1). This was echoed by another project manager: “But a lot was done to involve people in the process, a lot was accomplished to listen to whatever input they might have.” (C10, I12).

The collaboration with the business users did not only contribute to the success of the IT solution, as per I1 and I12 testimonies, it also helped the projects’ team to mitigate the uncertainties of unknowns in the business requirements when the business users were actively providing information and inputs. This elevated the quality of communicating the business needs and subsequently the process efficiency. A project manager explained how the “competence” of the product owner had helped the team to “make it work”: “They [product owner] had a good professional understanding. A lot of professional competences ... I think we were pretty good at trying to make it work.” (C3, I4).

The question is how collaboration has helped these projects to achieve cost underruns. Contrary to the conditions we presented above, collaboration had a less tangible impact on project cost. This does not make it of secondary importance. Collaboration elevated the teams’ performance and subsequently the process efficiency, i.e., the amount of effort required to produce the IT solution and implement the change. Project teams worked collectively toward achieving the project goals and achieving “success.” This is supported by a project manager’s claim: “Collaboration is probably an important word in here, for things to succeed, you are nerdy and you want to collaborate.” (C6 & 7, I9). According to her testimony, collaboration is a condition to “success”, which is synonymous to delivering the IT solution, meeting quality requirements, schedule and budget expectations. Both projects (C6 & C7,) closing reports’ documents describe implementing the IT solutions and changes “successfully” and within the agreed quality criteria. Our analysis of the reported schedule and budget data shows that both projects have achieved cost underruns and were completed within schedule.

**Budgetary control:** Half of the project managers showed meticulous attention to controlling the project budget. This practice was not merely a process; it was also an accountability. With respect to the process, projects implement procedures and tools to track the accuracy of spending. This interviewee states: “...

*It [the project] was very focused on the financial management and on the delivery management that we kind of got things delivered at the time that had been agreed, at the prices that had been agreed. [name of the organization] had a very good financial management system and was very conscientious about managing the finances.”* (C2, I3). Some project managers implied accountability for the budget. For example, this project manager indicates that it is an obligation (he used “must”) to remain within budget: “...Of course we should look at the budget ... which is such a stricter duty, that we must try to stay within ... so of course there was focus on the budget.” (C4, I5). Another project manager implied an obligation or willingness to accept responsibility and commitment to the budget. He explained: *It was such a thing that I have to get a handle on all the expenses in this project, I am simply out with the magnifying glass and find all the expenses.* (C1, I1). This interviewee also demonstrated responsibility (“no waste”): *“It is important that you are really loyal to not waste money away”* (C5, I6). Accountability for the budget coupled with good control processes enabled projects in our sample to achieve cost underrun.

Then, how did budgetary control help these projects to achieve cost underruns? By accurately measuring the actual project costs against the estimated budget, projects were able to continuously assess potential deviations and respond swiftly if necessary. This practice provided project managers with a sense of control over their spending. One project manager explains that “attention to all expenses” (i.e., control) allowed him to make judgements on “putting out” (i.e., put a stop to) other less important expenses: *“There was a lot of attention to all expenses, where we in the [name of the organization] and I think that applies, i.e., in all sorts of other projects, well (...) we had to put out all those little fires of expense.”* (C1, I1).

**Capitalizing on previous business domain knowledge:** Knowledge of the organization’s business domain helped projects to navigate through the complexity of the business requirements. Several (six of the interviewees) project managers were aware of this valuable quality. They capitalized on previous business domain knowledge of teams gained from delivering projects within the same organization. In some instances, this was implemented by re-using teams that participated in executing projects together in the same organization. For example, one project manager associated the “success” of her project to the “learnings” from a previous project: *“We had also learned a lot, also from [the name of a previous project] ... and that is actually perhaps not entirely irrelevant to*

one of the success criteria of the project.” (C11, I13). In the same vein, another project manager thought that it was “advantageous” to leverage her previous team to run a new project. She stated: “Of course it [the team familiarity with the business domain knowledge] is also an advantage, yes. Definitely.” (C3, I4).

There is evidence in our data that this practice gave some projects an advantageous instrument to deliver, helped them achieve process efficiency and subsequently achieve a cost underrun. This practice has mainly helped the project teams to understand and interpret the requirements, which allowed them to be in control and fully informed. This project manager explains how “having knowledge” has helped his team’s efficiency in specifying and implementing the requirements: *So there we had quite a lot of soil under the nails and knowledge of what is being demanded at institutions, where are the issues ... And part of the requirements specification then turns into making or discussing, how to develop an implementation plan.* (C4, I5).

**Continuous learning during the project:** Four projects were conscious that new knowledge should be translated into actionable improvements in order to increase their efficiency. While some projects adopted an iterative rollout of the IT solution in order to learn and adapt, others simply claimed that they became “wiser” during the course of the project. For example, this interviewee explained the learning from the pilot phase and the subsequent implementation phases: *So the pilot, it’s super much our implementation method that we try out there. When the pilot is over, we go in and revise, we do so after each phase, but we look at what went well in this implementation, what needs to be adjusted.* (C4, I5). This project manager explained that “becoming wiser”, i.e. acquiring knowledge and information, helped the project: *I think we have become a lot wiser on the project afterwards. And that also suggests that it can be a bit difficult, when you develop a completely new gadget, or a completely new solution ... It’s partly about getting wiser.* (C12, I14).

**Accommodating uncertainty:** Three projects in our sample faced particularly strong uncertainty in the business needs. This uncertainty was manifested because the business requirements at the time of the estimation were characterized by inaccuracy, unknowns or unreliable. Carrying a project with unknowns presents some risks. Project managers mitigated these risks by generously estimating the budget. This project manager explained how he identified uncertainty in the project idea and responded by “budgeting high”:

*“So you see, you have some kind of idea of where this carries you and what you are looking at. And then there is something in it [the project idea] where you say, this is uncertain ... We expected that it would be quite expensive. So, therefore, it has been budgeted quite high.”* (C2, I2). Similar action was taken by this project manager: *“So let’s make sure we have room to go if something goes wrong. So I also think it was a bit out of the consideration one might have over budgeted a bit.”* (C8, I10). She mitigated the risk of unknowns (“something goes wrong”) by “over budgeting.” While accommodating uncertainty occurred at the outset of the project, savings in project’s expenditures were encountered during the execution of the project, once the real costs became known.

The unknowns in the business needs eventually became knowns, as more details came to light and the project teams learned more. This implied more understanding of the real cost and in some instances less expenditure than what was originally estimated. This project manager stated: *“It just turned out that it was not quite as complicated and that we did not spend as many estimated implementation costs as we had originally anticipated.”* (C8, I10).

## 5. Discussion

Our findings show a significant emphasis on practices that foster an execution to achieve cost underruns. Some of these practices have been long called for by researchers, e.g., budgetary control. So, what is new? The distinguishing novelty in these findings is the social and human-related qualities invested in the implementation and execution of these practices. Individuals’ and teams’ actions were directed towards the implementation of some of these practices, which helped them to work better.

**Collaboration.** Collaboration brings together individuals with different skills and perspectives to complete a project. They not only have to work together but also need to think together, and share in the responsibilities, rather than work as individuals. Although several studies [24, 25] pointed out the positive effect of collaboration on project success, there is little work that establishes empirical evidence of the link between this construct and success [24]. Bond-Barnard et al. conclude that project success is more likely to materialize as the degree of collaboration increases [24]. Hoegl and Gemuenden propose a “teamwork quality” construct, which postulates that highly collaborative teams display behaviors related to six facets: communication, coordination, balance

of member contributions, mutual support, effort and cohesion [25]. Their study suggests a positive influence of “teamwork quality” on team performance [25]. Our study builds on this work by providing empirical evidence of project outcomes (i.e., achieving cost underrun) and implementing highly collaborative teams to achieve such outcomes. Our findings also suggest that various degrees of collaboration were implemented across our cases. In some cases, when establishing one consolidated team was not feasible, project managers used other means of collaboration, such as regular meetings and working groups. This finding shows how collaboration may contribute to facilitating cost underrun in IT projects.

While previous literature highlights the importance of collaboration, we extend the literature by revealing two different approaches for implementing a collaborative environment conducive to high project efficiency: building one team and arm’s length collaboration. Especially the building one team strategy goes beyond the existing literature on outsourced projects, which typically conceives of clients and vendors as separate parties [26], which are thus unlikely to form a team. We also point to context factors - closeness and familiarity between clients and vendors - that may lead projects to choose among these strategies.

**Capitalizing on previous knowledge.** Knowledge integration entails sharing and contributing individually held information and know-how to become common knowledge [27]. This quality has been linked to IT projects performance by several studies [27, 28, 29]. Mitchell found that access to external knowledge and internal knowledge integration in IT project teams significantly reduce the risk of project delays, thus promoting timely project completion [27]. Patnayakuni et al. concur that integration of explicit knowledge and collaboratively exchanging this knowledge positively influence the performance of the team [28]. Tesch et al. present similar conclusions [29]. Their results suggest that developers knowledge of the application domains had significant influence on project performance. We build on this work by contributing with cases where we made the link between project outcomes, i.e., cost underrun, and implementing the practice of knowledge sharing. In particular, our finding suggests that projects also capitalize on their teams’ previously acquired knowledge to run other projects. Our results suggest that prior knowledge may play a more important role if one looks at cost underrun as the dependent variable. This finding implies that IT project teams’ knowledge of the business domain is a valuable asset that should be leveraged as a capability in projects to achieve cost

underruns.

Although the role of domain knowledge in software development is widely acknowledged, evidence of the benefits of continuity in client-vendor relationships shows mixed findings. While Ethiraj et al. find that such continuity helps vendors to save costs [30], Gefen et al. did not find correlations between continuity and payments to the vendor [31]. Going beyond these findings, our study reveals that firms can capitalize on prior knowledge by ensuring stability at the team level. That is, projects that ensured continuity of joint client-vendor teams across projects were able to achieve cost underruns.

**Continuous learning.** Cultivating learning within projects is an “under-developed” area of research in project management [32]. Traditionally, project management seems to assume the certainty of the knowns at the inception of the project, which is not always true. Projects face a great degree of uncertainty and complexity. Ahern et al. suggest “modes of organizing and learning” to navigate through projects’ complexity and uncertainty. Ahern et al. argue that projects are means of organizing and learning to achieve an undertaking [32]. They explain that project management methods should acknowledge that knowledge formation during the project life cycle is a fundamental part of the delivery process instead of presuming sufficient knowledge at the outset with little learning during the journey [32]. Other studies [33, 26] suggest that techniques such as continuous feedback, self-organization and cross-functional teams promote collective learning processes in software development teams. Our findings show that project teams learn in-flight and invest the newly acquired knowledge to promote their processes’ efficiency. This has influenced their performance and contributed to achieving cost underrun. The implication of this finding is that IT projects should continuously generate knowledge during the execution that is not specifiable at the outset and invest the new knowledge in improving the efficiency of the process. Continuous learning during the project helped to achieve cost underrun in the cases we studied. Previous research shows that strategies for continuous learning, such as iterative development, can positively affect the business impact of a software system [34] and project duration [35], our study presents cost underrun as another potential positive consequence of these strategies.

**Accommodating uncertainty.** Requirements uncertainty occurs when the information necessary to identify and understand user requirements is not

sufficient for the project team to estimate, plan and develop the IT product and other subsequent changes. It has been shown to have a negative effect on project performance [13]. While strategies for accommodating uncertainty have been suggested by Flyvbjerg and colleagues [8, 8], there is little evidence to date about the use of these strategies in practice. In our sample, we see that these strategies can indeed help to avoid cost overrun.

We contribute to the literature of IT project performance with an investigation of twelve cases. We establish empirical links between a set of practices and achieving cost overrun. The implication of our findings is that the set-up of IT projects should aim at balancing process and human-focused practices and techniques to achieve cost overrun. IT Project management is more than planning and monitoring. It is a socially loaded practice that demands a great degree of collaboration among the parties involved and, in some instances, establishing a consolidated team to carry out the execution. This collaborative execution also necessitates capitalizing and sharing knowledge and continuous learning to navigate through the inherent complexity of the IT solution that the project seeks to deliver.

## 6. Limitations

Our study is not without limitations. First, our sample contains only government IT projects. Private sector projects may bring an element of diversity to the sample. Second, our study focused on exploring patterns among projects with cost overrun. Future work could take a comparative approach, contrasting projects with cost overrun to those with cost overrun. Third, while we focused on cost overrun during the life of a project, we did not examine costs during the maintenance phase. Some projects may reduce scope and costs during the life of a project by moving scope into the maintenance phase. Our analysis did not capture these costs. Finally, the majority of our participants had comparatively long experience in project management. This may limit the generalizability of our findings to such settings. In addition, we have not explored cost-increasing conditions in our study. We recognize that projects may experience both conditions that increase and reduce costs.

Although the iron triangle suggests three parameters (cost, time and scope) to measure performance, we opted only for one, i.e., cost. Our focus is motivated by the popularity of this metric in the literature and our genuine motivation to shed light on the cost overrun of IT projects. Our study contributes to the shortage

of empirical studies of IT projects performance with positive cases.

## 7. Conclusion

This study is motivated by the aspiration to understand how IT projects manage to achieve cost overruns. It also aims at minimizing the imbalance in IT projects performance reporting. There is a tendency to report underperforming cases.

To deepen our understanding of how IT projects manage to achieve cost overruns, we investigated twelve IT projects using two qualitative techniques, documents analysis and interviews. Our cases have achieved cost overruns, which we have validated before selection. We conclude that the projects in our sample have greatly focused on the combination of process and human-focused practices to achieve their goals, including cost overrun.

## Acknowledgements

The authors would like to thank the interviewees and their respective organizations for participating in this study. Further, the authors would like to thank the anonymous reviewers for their valuable feedback. Adam Alami and Christian Madsen are employed by the Research Centre for Government IT, which is a co-funded collaboration between the IT University of Copenhagen and the Danish Ministry for Finance. The ministry has provided the documentation on IT projects, but had no part in defining, guiding or performing the research. The ministry has made no textual, editorial, or other contributions to the paper but has received an earlier draft version to correct possible factual errors concerning the ministry and Danish council for ICT.

## References

- [1] H. Benbya, D. E. Leidner, and D. Preston, "Mis quarterly research curation on information systems alignment research curation team," *MIS Quarterly Research Curations*, pp. 1–19, 2019.
- [2] K. Dongus, S. Ebert, M. Schermann, and H. Kremer, "What determines information systems project performance? a narrative review and meta-analysis," in *2015 48th Hawaii International Conference on System Sciences*, pp. 4483–4492, IEEE, 2015.
- [3] P. L. Bannerman, "Defining project success: a multilevel framework," in *Proceedings of the Project Management Institute Research Conference*, pp. 1–14, Citeseer, 2008.
- [4] K. Jugdev and R. Müller, "A retrospective look at our evolving understanding of project success," *Project management journal*, vol. 36, no. 4, pp. 19–31, 2005.
- [5] J. Pollack, J. Helm, and D. Adler, "What is the iron triangle, and how has it changed?," *International journal of managing projects in business*, 2018.

- [6] A. Badewi, "The impact of project management (pm) and benefits management (bm) practices on project success: Towards developing a project benefits governance framework," *International Journal of Project Management*, vol. 34, no. 4, pp. 761–778, 2016.
- [7] B. Flyvbjerg, A. Ansar, A. Budzier, S. Buhl, C. Cantarelli, M. Garbuio, C. Glenting, M. S. Holm, D. Lovallo, D. Lunn, *et al.*, "Five things you should know about cost overrun," *Transportation Research Part A: Policy and Practice*, vol. 118, pp. 174–190, 2018.
- [8] B. Flyvbjerg, "Policy and planning for large-infrastructure projects: problems, causes, cures," *Environment and Planning B: planning and design*, vol. 34, no. 4, pp. 578–597, 2007.
- [9] T. S. Group, "Chaos report 1994-2020," [https://www.standishgroup.com/sample\\_research](https://www.standishgroup.com/sample_research), 2020.
- [10] K. Molokken and M. Jorgensen, "A review of software surveys on software effort estimation," in *2003 International Symposium on Empirical Software Engineering, 2003. ISESE 2003. Proceedings.*, pp. 223–230, IEEE, 2003.
- [11] A. Budzier and B. Flyvbjerg, "Why your it project might be riskier than you think," *Harvard Business Review*, vol. 89, no. 9, 2011.
- [12] R. L. Glass, "The standish report: does it really describe a software crisis?," *Communications of the ACM*, vol. 49, no. 8, pp. 15–16, 2006.
- [13] S. Nidumolu, "The effect of coordination and uncertainty on software project performance: residual performance risk as an intervening variable," *Information systems research*, vol. 6, no. 3, pp. 191–219, 1995.
- [14] P. C. Fiss, "A set-theoretic approach to organizational configurations," *Academy of management review*, vol. 32, no. 4, pp. 1180–1198, 2007.
- [15] T. Cooke-Davies, "The "real" success factors on projects," *International journal of project management*, vol. 20, no. 3, pp. 185–190, 2002.
- [16] D. P. Slevin and J. K. Pinto, "Critical success factors across the project life cycle," *Project Management Journal*, vol. 19, no. 3, pp. 67–75, 1988.
- [17] J. Fortune and D. White, "Framing of project critical success factors by a systems model," *International journal of project management*, vol. 24, no. 1, pp. 53–65, 2006.
- [18] W. Belassi and O. I. Tukel, "A new framework for determining critical success/failure factors in projects," *International journal of project management*, vol. 14, no. 3, pp. 141–151, 1996.
- [19] M. Fowler, J. Highsmith, *et al.*, "The agile manifesto," *Software Development*, vol. 9, no. 8, pp. 28–35, 2001.
- [20] J. W. Creswell and D. L. Miller, "Determining validity in qualitative inquiry," *Theory into practice*, vol. 39, no. 3, pp. 124–130, 2000.
- [21] L. A. Palinkas, S. M. Horwitz, C. A. Green, J. P. Wisdom, N. Duan, and K. Hoagwood, "Purposeful sampling for qualitative data collection and analysis in mixed method implementation research," *Administration and policy in mental health and mental health services research*, vol. 42, no. 5, pp. 533–544, 2015.
- [22] V. Braun and V. Clarke, "Using thematic analysis in psychology," *Qualitative research in psychology*, vol. 3, no. 2, pp. 77–101, 2006.
- [23] M. Hammersley, "Troubles with triangulation," *Advances in mixed methods research*, pp. 22–36, 2008.
- [24] T. J. Bond-Barnard, L. Fletcher, and H. Steyn, "Linking trust and collaboration in project teams to project management success," *International Journal of Managing Projects in Business*, 2018.
- [25] M. Hoegl and H. G. Gemuenden, "Teamwork quality and the success of innovative projects: A theoretical concept and empirical evidence," *Organization science*, vol. 12, no. 4, pp. 435–449, 2001.
- [26] O. Krancher, "Agile software development practices and success in outsourced projects: The moderating role of requirements risk," in *International Conference on Agile Software Development*, pp. 56–72, Springer, 2020.
- [27] V. L. Mitchell, "Knowledge integration and information technology project performance," *Mis Quarterly*, pp. 919–939, 2006.
- [28] R. Patnayakuni and C. P. Ruppel, "Managing the complementarity of knowledge integration and process formalization for systems development performance," *Journal of the association for information systems*, vol. 7, no. 1, p. 21, 2006.
- [29] D. Tesch, M. G. Sobol, G. Klein, and J. J. Jiang, "User and developer common knowledge: Effect on the success of information system development projects," *International Journal of Project Management*, vol. 27, no. 7, pp. 657–664, 2009.
- [30] S. K. Ethiraj, P. Kale, M. S. Krishnan, and J. V. Singh, "Where do capabilities come from and how do they matter? a study in the software services industry," *Strategic management journal*, vol. 26, no. 1, pp. 25–45, 2005.
- [31] D. Gefen, S. Wyss, and Y. Lichtenstein, "Business familiarity as risk mitigation in software development outsourcing contracts," *MIS quarterly*, pp. 531–551, 2008.
- [32] T. Ahern, B. Leavy, and P. Byrne, "Knowledge formation and learning in the management of projects: A problem solving perspective," *International Journal of Project Management*, vol. 32, no. 8, pp. 1423–1431, 2014.
- [33] O. Krancher, P. Luther, and M. Jost, "Key affordances of platform-as-a-service: Self-organization and continuous feedback," *Journal of Management Information Systems*, vol. 35, no. 3, pp. 776–812, 2018.
- [34] D. Robey, J. W. Ross, and M.-C. Boudreau, "Learning to implement enterprise systems: An exploratory study of the dialectics of change," *Journal of management information systems*, vol. 19, no. 1, pp. 17–46, 2002.
- [35] K. M. Eisenhardt and B. N. Tabrizi, "Accelerating adaptive processes: Product innovation in the global computer industry," *Administrative science quarterly*, pp. 84–110, 1995.