

Digitalization of Swedish Government Agencies – A Perspective Through the Lens of a Software Development Census

Preprint of paper accepted for the 40th International Conference on Software Engineering,

May 27–3 June 2018, Gothenburg, Sweden, Software Engineering in Society Track

Markus Borg
RISE SICS AB
P.O. Box 1212
Lund 43017-6221, Sweden
markus.borg@ri.se

Thomas Olsson
RISE SICS AB
P.O. Box 1212
Lund 43017-6221, Sweden
thomas.olsson@ri.se

Ulrik Franke
RISE SICS AB
P.O. Box 1212
Stockholm 43017-6221, Sweden
ulrik.franke@ri.se

Saïd Assar
IMT Business School
Evry, France
said.assar@telecom-em.eu

ABSTRACT

Software engineering is at the core of the digitalization of society. Ill-informed decisions can have major consequences, as made evident in the 2017 government crisis in Sweden, originating in a data breach caused by an outsourcing deal made by the Swedish Transport Agency. Many Government Agencies (GovAgs) in Sweden are rapidly undergoing a digital transition, thus it is important to overview how widespread, and mature, software development is in this part of the public sector. We present a software development census of Swedish GovAgs, complemented by document analysis and a survey. We show that 39.2% of the GovAgs develop software internally, some matching the number of developers in large companies. Our findings suggest that the development largely resembles private sector counterparts, and that established best practices are implemented. Still, we identify improvement potential in the areas of strategic sourcing, openness, collaboration across GovAgs, and quality requirements. The Swedish Government has announced the establishment of a new digitalization agency next year, and our hope is that the software engineering community will contribute its expertise with a clear voice.

KEYWORDS

Software engineering, public sector, digital government, census

1 INTRODUCTION

With the digitalization of society, more and more organizations become software intensive. Many companies must change to stay competitive on the market, but also public organizations need to improve their software maturity [11]. Government Agencies (GovAgs) are one type of public sector organizations that inevitably must adapt to the new digital era. While increased software capabilities require considerable investments, funded by tax money, successful scaling of software in GovAgs could also provide multiple benefits. According to the OECD, digital governments can lead to “more open, transparent, innovative, participatory, and trustworthy governments” [33]. Furthermore, digital governments enable

use of technology for creating a new generation of cost-efficient, interactive, and ubiquitous ICT-enabled public services [17].

Digitalization is acknowledged in the Swedish Government’s strategy on digital transformation launched in 2017, with an objective to “become the world leader in harnessing the opportunities of digital transformation”¹. Sweden has a good track record in digitalization in general, e.g., Sweden was ranked third in the World Economic Forum’s Networked Readiness Index 2016 [13], and third in the EU Digital Economy & Society Index 2017 [10]. However, the digitalization of GovAgs has just begun, and reaping the full benefits of digitization is a challenge for any organization [11].

Software development in the public sector faces unique challenges. GovAgs often have needs unlike any other actor, and when procuring specific IT services a GovAg might find itself a single buyer on the market. Moreover, sometimes only a few providers on the market offer services geared specifically towards the public sector, thus limiting procurement and sourcing options [3, 24]. Previous work on Swedish GovAgs reports that both software scaling and general software process improvement have been ongoing in recent years [6, 20], but there is no overview available of the current state-of-practice in the GovAgs’ software projects.

GovAgs’ IT is currently a hot topic in the Swedish public debate. The background is that the Swedish Transport Agency outsourced its IT to IBM in 2015. In the summer of 2017, it was revealed that as part of this procurement, the director general of the Swedish Transport Agency had authorized deviations from the applicable laws on information security. Following an investigation by the Swedish Security Service, the director general was fined and dismissed from office². As the story of insufficient and illegal information security practices at the GovAg made it into the media, questions were raised about how the government had managed the crisis. The questions about who knew what at which time, and which precautionary measures were, or were not taken, have forced two ministers and

¹<http://www.government.se/press-releases/2017/06/action-on-digital-transformation/>

²<https://transportstyrelsen.se/en/About-us/statement-about-the-information-in-media-regarding-our-it-public-procurement/>

one state secretary to resign. A third minister was subject to a vote of no-confidence in the parliament, but remained in office as the vote did not gain a majority.

The overall goal of this study is to gage the digitalization of Swedish GovAgs from a software engineering perspective. We contribute a unique view of the permeation of software development in GovAgs – as an indicator of the digitalization of society. The purpose of this article is not to relate all the details of the recent IT scandal, but it serves as a timely illustration of the importance of informed decisions. Consequently, a secondary goal of our work is to explore the current state of software development practice in Swedish GovAgs. We present a census of the 240 GovAgs in Sweden, designed to answer the following Research Question (RQ):

- RQ1. How many government agencies in Sweden develop software to support their core operations?

Among the GovAgs identified in RQ1, we continue by exploring:

- RQ2. What is the distribution of software sourcing strategies?
- RQ3. Are common software engineering best practices used?
- RQ4. How are software qualities addressed?

The rest of the paper is organized as follows: Section 2 defines digitalization and introduces related work on e-government and software engineering in the public sector. Section 3 describes our data collection and analysis. In Section 4 we present our results and discuss our findings in the light of the RQs. Finally, Section 5 summarizes the paper, puts it in the context of digitalization in Sweden, and outlines future work.

2 BACKGROUND AND RELATED WORK

In this section, we present background knowledge about digitalization and e-government. We also present related work on software development in the public sector, in Sweden and abroad.

2.1 Digitalization of Society and E-Government

Digitization is usually defined as “the integration of digital technologies into everyday life by the digitization of everything that can be digitized” [32]. This profound change in the way business is conducted is known as “digital transformation” [28] or “digitalization”. Digitalization implies disrupting organizational structures and adopting new innovative perspectives for the definition of commercial products and the creation of business value [42].

In the public sector, digitization and digitalization are generally considered as extensions of e-government. Although e-government was initially considered as a particular form of e-commerce consisting in providing online documents and services to citizens [17], its scope is much wider and includes political goals such as institutional reforms, government modernization, and the introduction of new democratic practices [1]. Denmark, another country on the Scandinavian Peninsula, established an Agency for Digitisation in 2011, with the mission to “speed up the digitalization processes required to modernize the Danish welfare society”.

The evaluation of e-government endeavors has received much research attention [18, 26]. There are multiple criteria for evaluation, e.g., quality of information, level of personalization, level of digitization, and the scope of online services. The United Nations

e-government readiness index, a large and evolving set of factors, has shifted focus from e-inclusion in 2005, to financial leverage in time of the 2010 crisis, to support of sustainable development in 2016 [34]. This shift of focus implies that a country’s position can change a lot from one year to another. For example, Sweden was in third place in 2005, then dropped from the top 10 in 2010, and is ranked sixth in the 2016 UN e-government readiness index [34].

Bridging e-government and the next subsection, there is a movement to make public *data* as well as *software* developed in publicly funded projects open [15], in Sweden and in other countries. Both the European Commission³ and the Swedish Government⁴ argue that more open data has the potential to lead to new innovations that address societal challenges, as well as increased transparency of governments.

2.2 Software Engineering in the Public Sector

As software is the founding element for all ICT based public services, the development and acquisition of software together with the optimization of their information systems architectures and operations have acquired strategic importance for all government activities. Moreover, the spectacular failure of some large public information systems projects, e.g., the Human Resource and Payroll system in France in 2014 [30], have led public authorities to strongly enforce best practices and modern approaches to software development projects. For example, The Swedish Tax Agency recently announced a transition to agile development methods [25] and information security issues gets increasing attention by the Swedish Government⁵.

Some research claims that public sector projects tend to be poorly conducted compared to its private sector counterparts [16]. On the other hand, in a study conducted in Norway in 2012, a number of project indicators were compared between public and private organizations, and the authors found no significant differences [19]. Nonetheless, there are indeed contextual differences between software projects in the public and private sectors. Rosacker and Rosacker report three major differences [41]: 1) the private sector is market-driven, whereas the public sector has less competition, 2) managers in the private sectors are primarily accountable to immediate customers and shareholders, in the public sector the stakeholders represent a broader group of constituents, and 3) public organizations can be subject to more forceful laws and regulations than private companies.

We identified some previous work on software projects in Swedish GovAgs. Larsson and Borg compared aspects of software engineering in large companies with its counterpart in a Swedish GovAg [20]. They report that a particular challenge for the GovAg is that their goals depend on external directives that might change as political powers shift, either on the national or European Union (EU) level. Another difference is that the GovAg does not develop a solution for an open market, instead they implement a solution required by the EU commission – with substantial financial penalties if they are not fulfilled on time. The same authors later presented another

³Open data <https://ec.europa.eu/digital-single-market/en/open-data>

⁴<http://www.regeringen.se/pressmeddelanden/2016/07/regeringen-oppnar-dorren-for-mer-oppna-data>

⁵<http://www.regeringen.se/pressmeddelanden/2017/08/sakrare-statlig-it-drift-genom-okad-samordning/>

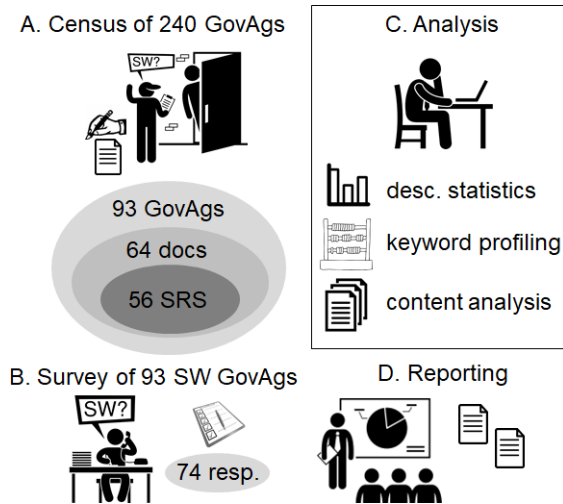


Figure 1: Overview of the research design: a census followed by a survey and an analysis based on triangulation. The reporting consists of this paper and a technical report [7].

study at the same Swedish GovAg [21], focusing on testing of QRs, but no findings appear to be unique to the public sector.

Looking beyond Sweden’s borders, Bannerman reported that software risk management in Australian GovAgs was unsystematic or informal – despite very large investments [4]. Furthermore, Bannerman reported that rigid plan-driven waterfall development dominated, thus offering limited possibilities to adapt to risks during project execution. Patanakul presented a study of problems in 14 large (€\$1B) public sector projects in the US, UK, and Australia [37]. His results corroborate Bannerman’s conclusion that risk management is inadequate, and also highlight that unclear requirements are a common cause for failed projects. Ziembra and Kolasa also addressed software risk management, but in the Polish public sector [48]. In line with Larsson and Borg [20], they argued that the public sector context introduces additional challenges due to an increased sensitivity to changing external directives, i.e., changing government processes or legal regulatory frameworks.

3 METHOD

We conducted a census of software development at Swedish GovAgs, i.e., an inquiry to gather information about every individual in a population [14]. The target population was all GovAgs listed by Statistics Sweden⁶ on January 1, 2017, in total 240 GovAgs. The data collection was based on the Swedish Freedom of the Press Act [35], stating that anyone is entitled to read the documents held by public authorities.

Figure 1 shows an overview of the research design. We initiated the data collection process by sending a formal request per email to all 240 GovAgs at 12 AM on January 1, 2017 (cf., A. in Fig. 1). With support from an archivist, we formally requested a sample of software development documentation, with an emphasis on System Requirements Specifications (SRS), from an ongoing or completed software project. We stressed that the project should

be related to the GovAg’s core operations and go beyond web content management. We distributed up to three reminders with two-month intervals, and eventually 93 GovAgs confirmed that software development occurs. As some GovAgs referred to secrecy, we obtained documents from 64 GovAgs, and 56 GovAgs provided at least one SRS.

We complemented our census with a questionnaire-based survey of the 93 GovAgs that confirmed software development (cf. B. in Fig. 1). The questionnaire (available in the accompanying technical report [7]), distributed in mid-August 2017 to contacts identified during the census, consisted of closed questions. A majority of the questions consisted of three Likert scales, addressing: 1) development process, 2) sourcing strategies, and 3) development context, in total encompassing 25 Likert items. The selection of statements was influenced by SWEBOK v. 3.0 [46], ISO/IEC 25010 on quality requirements [12], Murhphy-Hill *et al.*’s comparison between game development and traditional software engineering [31], and Badampudi *et al.*’s research on sourcing options [3]. We calculated descriptive statistics for the survey, including Spearman’s rank-order correlation coefficients.

We collected data of three types: 1) direct correspondence between GovAgs’ officials and the first author per mail and/or telephone, 2) obtained official documents from 64 GovAgs, and 3) 74 survey responses. The first author cataloged the correspondence in a spreadsheet. All written communication was saved, and telephone calls (typically resembling interview sessions) were documented in notes. Our research design enables data and method triangulation [40], i.e., we draw conclusions based on multiple sources of evidence collected using different methods. We describe the document analysis and the main threats to validity next.

3.1 Document analysis

First, we performed keyword frequency profiling [38] (cf. C. in Fig. 1), or rather keyword presence profiling, to determine which software qualities defined in ISO/IEC 25010 (from now on: keywords) occur in the GovAgs’ documents. The goal of this light-weight analysis was to identify whether concepts relating to software qualities occur in the SRSs. Second, we performed qualitative content analysis [8] of the obtained documents, predominately using predefined topics and codes [40].

The keyword profiling evolved during the process. We validated the keywords by assessing the resulting frequency profiles and by manually investigating a subset by reading keywords in the context of the SRSs. As a result, some keywords required special treatment. “Security” and its Swedish translation “säkerhet” are too general concepts that also are used in everyday expressions. On top of that, “säkerhet” is used to denote both safety and security in Swedish. Hence, we decided to refine the security concept into: 1) confidentiality, 2) integrity, and 3) availability – motivated by the fact that Swedish GovAgs are mandated to classify their information assets accordingly⁷. Moreover, the Swedish translations of “availability” (“tillgänglighet”, also meaning “accessability”) and “reliability” (“tillförlitlighet”, also used for trust in general) suffer from similar vagueness problems, thus these keywords were restricted

⁶<http://www.scb.se/myndighetsregistret>

⁷<https://www.msb.se/externdata/rs/94a3d208-2ac4-48a1-84f2-208268f5767e.pdf>

to English. For the keyword “confidentiality”, we use the two keywords “konfidentialitet” and “sekretess” and for “portability” we used both “portabilitet” and “flyttbarhet” in Swedish to capture all relevant occurrences of the concepts. All translations are available in the accompanying technical report [7].

Content analysis in documents is often used in combination with other research methods as a means of triangulation [8]. We used content analysis for this purpose mainly in relation to RQ3 and RQ4, i.e., data triangulation to identify evidence, or at least indications, of implemented best practice presence and quality foci. Due to the large amounts of documents obtained from the GovAgs, a comprehensive document analysis was not within the scope of this study. Consequently, the content analysis can only be used to draw conclusions on the presence of phenomena, and not absence.

3.2 Threats to Validity

This section presents the main threats to the validity of our findings, organized in three parts. *External validity* concerns the generalizability of our results. First of all, the census part of our study does not suffer from issues related the sampling and response rate. We sent formal requests to the entire population of Swedish GovAgs and everyone was required to answer, thus we consider it highly reliable. Through the census, we collected firsthand contacts with government officials responsible for software development, which we later used to reach a very high response rate (76.6%) in the survey part. As our respondents are GovAgs and not individuals, we respond to the call by Stavru that more surveys should target organizations instead of personal opinions [47]. That said, we cannot be sure that the entire GovAgs is behind the individual answers.

Content validity concerns how much a measure represents every single element of a construct. For the survey, we had to limit the number of Likert items to keep the response times reasonable. We selected statements based on related work, incl. SWEBOK [46], but obviously more Likert items would have captured additional aspects of software development. Understanding software development requires more than a list of closed questions, thus we plan for future work with in-depth case studies at a selection of GovAgs.

Finally, *construct validity* refers to how an operational definition of a variable actually reflects the true theoretical meaning of a concept. The major threat to our study is whether our inquiry about software development was properly interpreted by the respondents. Many GovAgs appeared to lump together all matters related to IT, not distinguishing software development from general IT operations. However, we believe that the document analysis filtered out any GovAgs that do not develop software internally. The keyword profiling also introduces threats to construct validity, as the approach cannot completely capture all concepts of software qualities. First, there might be false positives, i.e., wrongly identifying a quality focus as present in an SRS. Second, we could have false negatives, i.e., failing to identify a quality focus that indeed was described in an SRS. We mitigated both threats by manual assessment of a sample of both keywords and SRSs.

4 RESULTS AND DISCUSSION

All GovAgs in Sweden answered the census, i.e., whether software development related to the GovAg’s core operations occurs.

Seventy-four of the 93 GovAgs that confirmed software development answered our survey (cf. Table 2), i.e., a response rate of 79.6%. Five GovAgs could not answer the survey for confidentiality reasons and three GovAgs declined to answer. We are still expecting answers from two GovAgs, but the remaining 9 GovAgs did not reply to our emails. This section synthesizes findings from the census, survey, and document analysis to answer the RQs.

4.1 Software development at Swedish Government Agencies – A Census (RQ1)

All GovAgs in Sweden answered the census, but three GovAgs were discontinued during the execution of the study, resulting in a total population of 237 GovAgs. Ninety-three out of 237 GovAgs (39.2%) answer that they have in-house software development, performed either with employed developers or contractors. However, several GovAgs answer that another GovAg provides custom software solutions for them, e.g., all 21 County Administrative Boards have centralized the IT development organization to the agency in Västra Götalands Län. Furthermore, seven GovAgs report that their respective parent GovAgs provide all software, e.g., the National Board of Health and Welfare provides the smaller National Medical Responsibility Board with all software solutions.

Most GovAgs disclosed documentation from in-house software projects. We obtained software documentation from 64 out of 93 GovAgs (68.8%). Nine GovAgs answered that the documentation was secret, e.g., the Prosecution Authority, the Coast Guard, the Election Authority, and the Armed Forces. By the time of this writing, we are still waiting for documentation from 20 GovAgs, including some large GovAgs such as the National Property Board.

The census also revealed that several GovAgs have considerable software engineering know-how despite having no in-house development. Sixteen of the 237 GovAgs report that they develop software and systems requirements in-house, which then are used for public procurement. We obtained SRSs from all these 16 GovAgs, and conclude that software engineering knowledge exists also in GovAgs not covered among the 93 that have internal development. Considerable requirements engineering skills are needed when specifying large systems in the public sector, and several GovAgs without internal development do it well – however, none of these 16 GovAgs are part of the analysis in the remainder of the paper.

Table 1 shows an overview of the volume of software development conducted at Swedish GovAgs. The most common number of developers is between 5 and 19, but several GovAgs report having large development organizations – some with more than 100 software developers. Note that GovAgs such as the Armed Forces refer to secrecy, and that we are still waiting for figures from some large GovAgs, e.g., the Public Employment Agency and Statistics Sweden, thus the number of large development organizations is likely to be even higher.

Table 1 also shows an overview of the proportion of employed development resources, as opposed to contractors, in the GovAgs’ development organizations. The proportion varies from no employed software developers at all (e.g., the Energy Markets Inspectorate and the Environmental Protection Agency) to having nothing but employed developers (e.g., the Swedish University of Agricultural Sciences and the National Veterinary Institute). Typically,

Table 1: Number of software developers at Swedish GovAgs and the proportion of employed development resources.

#Developers		%Employed	
1-4	9	0-20%	10
5-19	30	21-40%	11
20-49	13	41-60%	15
50-99	8	61-80%	24
100-199	8	81-100%	11
>199	6	Missing	3

development at Swedish GovAgs involves a mix between in-house resources and contractors, i.e., most often between 40-80% of the developers are employed by the GovAg. These figures apply to all respondents with 50 or more developers, except the Prison and Probation Service and the Council for Higher Education (70% contractors each) and the Meteorological and Hydrological Institute (85% employed developers).

Table 2 shows the responses to the three Likert scales collected as part of the survey. In the rest of the paper, we refer to individual statements using their IDs in bold font, e.g., “**S3d**”. Based on the survey, we continue by providing a general overview of software development at Swedish GovAgs.

Agile development is an important contextual factor when describing contemporary software engineering, thus it receives particular attention in our study. A clear majority of the GovAgs (65 out of 73, 87.8%) report that their software development is more agile than plan-driven (**S3b**). Four other statements are also related to agile development, and can thus be used to corroborate the level of agility. Lean software development principles (**S3c**, 70.3%) are common among the GovAgs, and a majority also confirm implementing DevOps (**S3d**, 73.0%). Note, however, that Lwakatara *et al.* recently showed that practitioners have different interpretations of what constitutes DevOps [27]. Finally, continuous integration (**S3m**) is used in 68.9% (46 out of 74) of the GovAgs, but test automation (**S3n**) is not a standard practice; only 32 out of 74 (43.2%) agree or strongly agree to the statement.

Analyzing the five statements related to agile, we find that many GovAgs get high agility ratings in our study. Most of them are small, including several universities, but also several large GovAgs appear to adhere to agile development practices, e.g., the Board of Agriculture and the Swedish Customs. We identified evidence of agile practices in the document analysis, such as Scrum backlogs from Linnaeus University and the Companies Registration Office, but also examples of traditional plan-driven documents with agile elements such as user stories. Our finding is in line with the dominance of agile and iterative methods reported by Scott *et al.* in the HELENA survey [43], i.e., Scrum, Kanban, and DevOps are all more popular than waterfall processes, both in Sweden and worldwide.

A recent example of the agile movement influencing Swedish GovAgs is that the Tax Agency, one of the most software-intensive GovAgs, announced a transition to agile methods in September 2017 [25]. The level of agility we report for Swedish GovAgs contrasts with findings by Bannerman from 2007, who reported that Australian GovAgs were mainly plan-driven [4].

A rationale for GovAgs to use agile methods is that a majority (45 out of 75, 60.8%) report having flexible deadlines (**S3k**). This suggests that it might be more important for a GovAg to be responsive to change rather than to follow a traditional plan, in line with the Agile Manifesto. The reported flexibility was unexpected, as previous work rather emphasized that software projects in the public sector often are subject to legislative changes with major consequences if not implemented in time [20].

We designed three questions geared at gaging the nature of the products and services developed by the GovAgs. Sixty-five out of 74 (87.8%) answer that the agency mainly develops information systems (**S3f**), i.e., systems intended to store, manage, and present information. The respondents are split in two groups regarding the statement on integration projects (**S3g**), such as described by Larsson *et al.* [21]: 29 GovAgs agree and 25 disagree. Finally, a majority of the respondents (44 out of 74, 59.5%) answer that test and debug are time consuming activities due to the complexity of the systems (**S3j**). Thus, our results indicate that the systems developed in the public sector are far from trivial, and in comparison to Murphy-Hill *et al.*'s study on development at Microsoft [31], they could be even more complex than private sector counterparts.

4.2 Sourcing strategies (RQ2)

A recurring strategic consideration in software projects is whether to develop an asset internally or acquire it from external sources. In software projects, the decision involves choosing between several different sourcing options [3]. First, a GovAg can develop a software asset in-house. Second, software can be acquired externally by 1) specifying requirements and outsourcing of development through public procurement, 2) purchasing commercial-off-the-shelf (COTS) software through public procurement, and 3) acquiring existing open source software (OSS). Note that previous research on strategic software sourcing mainly focused on market-driven contexts in the private sector; public-sector organizations are primarily designed to be “fair, open, objective, and accountable” rather than efficient [9].

Most GovAgs respond that a majority of the software development is conducted by employed resources (45 out of 74, 60.8%), while only 21 disagree with the corresponding statement (**S4a**) (28.4%). On the other hand, roughly the same proportion of respondents agree and disagree with the statement that most development is conducted by contractors (**S4b**, 31 vs. 34 out of 74). Thus, as the statements **S4a** and **S4b** were designed to expose an inverse relationship, the results are contradicting. We believe that a fraction of the respondents considered contractors working on the GovAgs premises as in-house resources, possibly due to long contracts. Regarding development abroad, currently a highly sensitive topic in the Swedish press, all respondents (but a missing answer) report that the development is conducted in Sweden: 67 (90.5%) even strongly agree to the statement (**S4f**). Regarding operations, however, Ottso and Kristensson conducted a survey after the Transport Agency scandal revealing that 11 Swedish GovAgs manage data in other countries [36], i.e., the solution that caused the 2017 crisis.

Our survey identifies a handful of relevant correlations between a focus on contractors (**S4f**) and the characteristics of software projects. The following statements are all correlated with **S4b** (statistically significant moderate correlations, $0.31 \leq |\rho| \leq 0.44$).

Table 2: Results from the survey. The Likert distributions show “Strongly disagree” to the left and “Strongly agree” to the right.

Statement	ID	Likert distribution	Median	Mode	Missing
Development process					
Software development at the agency follows a documented process.	a)		4	5	2
The software development is agile rather than plan-driven.	b)		4	4	1
The agency adheres to lean software development principles.	c)		4	4	3
The agency seeks a DevOps process to close the gap between development and operations.	d)		4	4	4
Software developed by the agency is typically made available under an open source license.	e)		2	1	3
Software developed at the agency can primarily be categorized as information systems.	f)		5	5	2
Development at the agency primarily relate to integration projects.	g)		3	4	2
The agency systematically reuses source code from previous projects.	h)		4	4	5
The systems maintained by the agency often have high technical debt.	i)		3	4	2
Due to the complexity of the systems, test and debug are often time consuming activities.	j)		4	4	2
The development projects at the agency often have flexible deadlines.	k)		4	4	2
Developers at the agency are supported by modern development tools.	l)		5	5	3
The development projects at the agency use continuous integration.	m)		4	4	4
Test automation is a natural part of development at the agency.	n)		3	4	3
Sourcing strategies					
Most of the development is conducted in-house.	a)		4	5	1
Most of the development at the agency is done contractors.	b)		3	4	3
The agency primarily focuses on requirements engineering followed by public procurement.	c)		2.5	1	2
The agency strives to purchase commercial-off-the-shelf products.	d)		4	4	1
The agency seeks to evolve existing systems by integrating available open source software.	e)		3	2	2
The agency's software development is conducted in Sweden.	f)		5	5	1
Development context					
Development projects at the agency are conducted in tight communication with our end-users.	a)		4	4	1
The agency coordinates development with projects undertaken at other government agencies.	b)		4	4	3
Information security is constantly present throughout the development process.	c)		4	5	3
Software developed at the agency has clear functional requirements.	d)		4	5	1
Different quality requirements are prioritized against each other in a systematic way.	e)		2	3	19

Software development primarily conducted by contractors is correlated with: 1) high technical debt (**S3i**, $\rho = 0.31$) and negatively correlated with 2) adherence to lean principles (**S3c**, $\rho = -0.44$), 3) security awareness throughout development (**S5c**, $\rho = -0.36$), 4) releasing source code as OSS (**S3e**, $\rho = -0.33$), and 5) coordinated development across GovAgs (**S5b**, $\rho = -0.31$).

While we cannot make causal claims, our findings suggest that contractors and technical debt co-occur in software projects at Swedish GovAgs. We found no previous work on the connection between technical debt and contractors in the public sector, the closest result instead targeting outsourcing. Assuncao *et al.* stressed the need for Brazilian GovAgs to check the quality of source code developed by third parties [2]. Lin *et al.* concluded that outsourcing in the public sector threatens the organizational memory, which we believe is fundamental to avoid technical debt. Another correlation suggests that GovAgs with a high proportion of contractors focus less on security. A noteworthy exception is The National Board of Institutional Care, a GovAg with a small development organizations consisting almost exclusively of contractors, that ranks high on security focus and awareness despite the external resources.

Nineteen out of 74 respondents claim that the GovAg mainly does requirements engineering followed by public procurement

(**S4c**, 25.7%), most of them having fewer than 20 developers. Exceptions include the Customs Service and the Government Offices of Sweden, both having 50-99 developers but still focusing on procurement of development effort. Lin *et al.* reported in 2007 that 92.3% of the public sector organizations in Australia outsourced at least parts of their IT functions [24]. They found a negative correlation between the percentage of outsourcing and organization size, i.e., a result in line with our findings, although our focus is on software development rather than IT functions. Finally, a majority of the GovAgs has an ambition to buy COTS systems rather than developing their own solutions (**S4d**, 58.1%), only nine GovAgs disagree with the statement.

Few GovAgs develop a majority of their software under open source licenses (6 out of 74, 8.1%). Instead, as many as 54 out of 74 (73.0%) disagree to the corresponding statement (**S3e**). This is interesting in the light of calls for more open data and open software in the public sector. It may be that even though the intention is to go in the direction of openness, the pace is limited by technical and organizational inertia. This hypothesis gains some support from the fact that while only six of the responding Swedish GovAgs predominantly develop software under open source licenses, considerably more GovAgs strive to acquire OSS solutions when investing in *new* systems (**S4e**, 31 out of 74, 41.9%). If this observation is indeed an

indicator of an ongoing development, it seems likely that we will see more open source software in the public sector in the future.

Many GovAgs develop and maintain software solutions offering similar features. The potential for commodity components and services became evident as our understanding grew during the study. Statement **S5b** reflects this, as 46 out of 74 GovAgs (62.2%) respond that they align development efforts with other GovAgs, e.g., through knowledge exchange activities and shared solutions. There is still a potential for improvement, however, as 13 GovAgs do not collaborate much with others. In addition, only half of the GovAgs report a systematic approach to source code reuse (**S3h**, see Section 4.3) – an approach that could facilitate collaboration across GovAgs.

There are current initiatives to improve GovAg collaboration, at least on a strategic level. In February 2017, Statens servicecenter, a GovAg supporting GovAg efficiency, concluded that a common cloud solution for Swedish GovAgs could reduce costs by 30% [45]. Similar initiatives have been reported also in Korea [22] and China [23]. In August, in the aftermath of the IT scandal, the Swedish Government commissioned the Social Insurance Agency to host a secure cloud solution for other GovAgs⁸. The Social Insurance Agency is among the most digitally mature GovAgs in Sweden, and enabling both shared technical solutions and knowledge transfer appears promising. However, as a focus on contractors is negatively correlated with cross-GovAg collaboration, our recommendation for Swedish GovAgs is to maintain an employee vs. contractor ratio that still ensures sufficient internal competency.

4.3 Software engineering best practices (RQ3)

The survey encompasses a self-assessment for GovAgs in relation to a selection of acknowledged software engineering best practices. Inspired by SWEBOK [46], we designed a number of Likert items that reflect generally accepted practices.

Requirements engineering is known as a foundation for software quality, and poor requirements have turned many software projects into failures [46]. Sixty-four out of 74 GovAgs (86.5%) report that the software development is guided by clear functional requirements (**S5d**). The document analysis of the 56 GovAgs that provided SRS partly supports this – the requirements specification practices appear generally mature. For example, The Board of Student Finance and Swedish University of Agricultural Sciences provided structured text requirements with unique identifiers and combine them with allocation to development sprints. Several GovAgs, e.g., the Swedish ESF Council and the Environmental Protection Agency, use textual use case descriptions, sometimes in combination with UML diagrams.

The finding that software development in GovAgs is supported by clear functional requirements corroborates and generalizes results by Larsson and Borg, who reported that one specific large Swedish GovAg had well-defined and verifiable functional requirements [20]. It had not always been that way though, as the authors reported that the requirements engineering practices had matured considerably in recent years. On the contrary, Patankul reported that unclear requirements often plague software projects in the public sector [37]. In the same vein, Mendez-Fernandez *et al.* report in

the NaPiRE survey that the incomplete and/or hidden requirements is a major challenge to all types of software projects, no matter size or development agility [29]. Thus, we see a need for further studies on the nature of the functional requirements in Swedish GovAgs – how do they manage to capture them with such precision?

Software processes, no matter if they are agile or plan-driven, facilitate communication and coordination, and support development of high-quality software products[46]. Most GovAgs (62 out of 74, 83.8%) agree that their development is guided by documented processes (**S3a**). Note that while the first generation of agile methods (as captured in the Agile Manifesto) downplays the role of processes, later agile constructs such as Scrum and DevOps have well-defined processes. As a majority of the GovAgs consider themselves agile, it appears likely that more recent agile processes are implemented, or possibly hybrid approaches [43].

Software reuse is recognized as a key factor in improving productivity and competitiveness [46], but it requires strategic vision and supporting processes. Roughly half of the GovAgs report a systematic approach to source code reuse (**S3h**). Our results identify software reuse as an improvement potential, possibly in combination with collaboration across GovAgs as discussed in Section 4.2.

Regarding development **tool support**, as many as 86.5% (64 out of 74) agree that appropriate tool support is available (**S3l**) – a rather high number given that surveys often indicate that practitioners call for better tools. Consequently, it appears that Swedish GovAgs are not obstructed by a lack of modern development tools.

GovAgs are not spared from challenging legacies and short-termed solutions. Thirty-five out of 74 GovAgs (47.3%) report that their systems have considerable **technical debt** (**S3i**). Assuncao *et al.* claims that the issue of technical debt is increasing in the public sector, and investigated the phenomenon in software projects run by the Brazilian Ministry of Communications [2]. Whether the technical debt is increasing in Swedish GovAgs we cannot say, but it is clearly an issue that should be considered – perhaps GovAgs should avoid hiring too many contractors as discussed in Section 4.2.

We regard two statements used in the agility assessment in Section 4.1 to be relevant also as software engineering best practices. Both are widespread approaches to limit negative consequences of big bang integration, namely **continuous integration** (**S3m**) and **test automation** (**S3n**). While the practices are complementary in nature, 62.6% of the GovAgs use the former practice and only 43.2% the latter.

A successful software development project requires interactions with stakeholders, in particular **feedback from end-users** [46]. As many as 67 out of 74 GovAgs (90.5%) agree that they communicate frequently with the future end-users, and only two single GovAgs disagree with the corresponding statement (**S5a**). This is a high number, suggesting that software projects in Swedish GovAgs are well connected with end-users such as GovAg officials or the Swedish citizens. Also, collaboration with end-users is a cornerstone in agile development, which a majority of the GovAgs adhere to. Moreover, Larsson and Borg reported that close collaboration between end-users and the software developers was one of six agile practices followed in a large Swedish GovAg [20].

When developing secure software, **security** must be built into the development process from the start [46]. Fifty-nine out of 74

⁸<https://computersweden.idg.se/2.2683/1.687526/it-drift-forsakringskassan>

GovAgs (79.7%) agree that security awareness permeates the entire development process (**S5c**) (32 GovAgs even strongly agree, 43.2%). Only six GovAgs disagree with the statement, including three GovAgs involved in the finance sector (the Financial Supervisory Authority, the Enforcement Authority, and the National Government Employee Pensions Board). On the other hand, disagreement might indicate a particular maturity, i.e., these GovAgs might be well aware of the need to build in security from the start, and thus emphasize that even more security awareness would be beneficial to their software projects.

Finally, we report four correlations between software projects guided by clear functional requirements (**S5d**) and other statements. The following statements are all correlated with **S4b** (statistically significant strong or moderate correlations, $0.31 \leq \rho \leq 0.59$). Software development primarily conducted by contractors is strongly correlated with: 1) following a documented process (**S3a**, $\rho = 0.59$) and moderately correlated with 2) coordination with other GovAgs (**S5b**, $\rho = 0.40$), 3) close communication with end-users (**S5a**, $\rho = 0.36$), and 4) adherence to lean principles (**S3c**, $\rho = 0.31$).

Without claiming any causality, we notice that GovAgs guided by clear functional requirements also report other advantages. Clear requirements go hand in hand with other types of clarity, including processes and communication both with end-users and other GovAgs. Requirements are known to be a vehicle for communication [46], and properly managed they might support collaboration across GovAgs. The correlation between clear requirements and lean principles is less obvious, but perhaps lean principles force requirements into brief constructs with no wasteful description. Alternatively, clear requirements might help a GovAg to focus on core development activities that bring value.

Our general impression is that the maturity of software development in the public sector resembles the software projects we have studied in the private sector. Most GovAgs with 50 or more developers implement established best practices. Two exceptions, both in the financial sector, are the Financial Supervisory Authority and the National Government Employee Pensions Board – two anomalies that constitute interesting directions for future work. Nevertheless, our results confirm the conclusion by Krogstie [19], i.e., software development in the public appears as mature as in the private sector.

4.4 Software qualities (RQ4)

Quality requirements are a well-known software engineering challenge [5]. In the survey, we explicitly asked the respondents to select the three most important qualities as defined by ISO/IEC 25010 [12], listed in Table 3. In addition, 53 GovAgs ranked the three qualities selected. In the keyword profiling, we found that 24 out of 56 GovAgs (42.9%) provided SRSs in which one or more of the keywords occur.

By far the most commonly selected quality in the survey was **functional suitability**, listed by 54 out of 74 GovAgs (73.0%), and also ranked as the most important quality by 29 GovAgs. The results suggest that GovAgs are highly results-oriented, for most respondents it is more important that software provides all functions required, rather than *how* it is provided, i.e., performance, usability,

security etc. and the rest of the qualities tend to be secondary considerations. The focus on functional suitability is not surprising, however, but in line with tendencies identified by Sentilles *et al.* in component-based software engineering [44]. On the other hand, the keyword profiling showed that the functional suitability occurs only in an SRS from a single GovAg. Thus, it is evident that other keywords are required to capture this concept.

Security was listed by 46 out of 74 GovAgs (62.2%), and is also ranked as the most important quality by 12 GovAgs (cf. Table 3). Moreover, the security related keywords (confidentiality, integrity, and availability) are the most prevalent in the SRS as they occur in SRSs from 20 GovAgs. For some GovAgs, both the survey results and the keyword profiling show that security is central, e.g., the Medical Products Agency and the Land Registration Authority. For others, the obtained SRS did not support the claimed focus on security, including the Tax Agency and the Transport Agency. In most cases the discrepancy can probably be explained by the limited SRS sample obtained, but it might also indicate that the security focus has not been fully operationalized.

Roughly half of the respondents list **usability** (34 out of 74, 45.9%) and **reliability** (30 out of 74, 40.5%) among the top three most important qualities. Seven GovAgs stand out by listing usability as the single most important quality, including the Swedish Police. The keyword profiling showed that usability occurs in SRSs from 16 of the 55 GovAgs, i.e., both the survey and the SRSs indicate its importance. The term reliability was only found in one single SRS, since we could not search for the Swedish translation as described in Section 3.1.

Performance efficiency appeared more important in the keyword profiling than in the survey. Performance keywords occur in SRSs from 17 out of 56 GovAgs, but performance was only prioritized by 11 out of 74 GovAgs (14.9%) in the survey – none of them ranking it as the most important quality. A possible explanation for this is that it is fairly easy to write performance QRs in comparison to the other qualities. However, if the SRSs specify unnecessarily strict performance QRs, the overall development costs might increase. We speculate that there is a lack of strategic direction leading to unnecessary performance QRs. Thus, there might be a potential for GovAgs to improve how quality targets are set, e.g., using the QUPER model proposed by Regnell *et al.* [39].

Three of the ISO/IEC 25010 qualities appear to be less important to Swedish GovAgs. **Maintainability** was listed among the top three qualities by 16 out of 74 GovAgs (21.6%), but its keywords occur only in SRSs from three GovAgs. **Compatibility** and **portability** were both only listed by a handful of GovAgs each, although the related keywords occur in SRSs from 10 and six GovAgs, respectively. As we argued for performance QRs, this might indicate that compatibility and portability QRs are comparatively straightforward to specify.

Finally, we explored how many GovAgs use systematic approaches to prioritize different QRs, e.g., the analytic hierarchy process or the \$100 test. Only nine out of 74 GovAgs (12.2%) agree to the statement (**S5e**) and as many as 28 disagree (37.8%). However, roughly a quarter of the respondents neither agree nor disagree (18 out of 73, 24.3%) – possibly indicating limited understanding of what a systematic method means. This is interesting and also a bit worrying, as all software projects involve trade-offs, and a

lack of systematic methods to do so should lead to worse quality of software and services in the end. Furthermore, we identified a strong correlation between clear functional requirements (**S5d**) and systematic QR trade-offs (**S5e**) ($\rho = 0.59$). This might, at least in part, be understood by the observation that clear requirements are a prerequisite for systematic trade-offs – it is impossible to do systematic trade-offs with undefined qualities.

Table 3: Ranking of ISO/IEC 25010 qualities based on the survey and results from the keyword profiling. EN denotes keywords searched for in English only.

Qualities	Survey		Docs
	#Mentions	#Prio 1s	#GovAgs
functional suitability	54	29	1
performance efficiency	11	0	19
compatibility	4	0	10
usability	34	9	16
reliability	30	4	1 (EN)
security	46	12	20 ⁹
- confidentiality	N/A	N/A	18
- integrity	N/A	N/A	3
- availability	N/A	N/A	4 (EN)
maintainability	16	1	4
portability	2	0	7

5 CONCLUSION

Societal digitalization is happening at a rapid pace worldwide, and Sweden is no exception. Digital solutions enable several benefits, but also introduce novel risks as shown by the IT scandal that surfaced in Sweden during 2017. To reap the benefits of digital solutions, and to mitigate the risks involved, considerable understanding of both software and software engineering is essential. Software is continuously scaling in society, and the public sector must ensure that its software maturity matches the new era.

We conducted a census of Swedish Government Agencies (GovAgs) to overview the extent of internal software development projects – as an indicator of the digitalization of society. Ninety-three GovAgs (39.2%) confirmed conducting software development (**RQ1**). While most such GovAgs have developers in the magnitude of tens, several GovAgs have hundreds of development resources. Swedish GovAgs often complement the employed developers with roughly the same number of contractors. However, our survey suggests that relying heavily on contractors is correlated with high technical debt and negatively correlated with coordinated development efforts with other GovAgs – thus we recommend GovAgs to maintain sufficient software know-how in-house.

GovAgs must make strategic decisions on software sourcing, e.g., whether to develop software internally or acquire it externally using public procurement (**RQ2**). Most GovAgs focus on in-house development (60.8%), but outsourcing of development is widespread – however rarely to organizations outside of Sweden. If possible, a

majority of the GovAgs aim to purchase software solutions off-the-shelf (58.1%). Furthermore, few GovAgs routinely develop software under open source licenses (8.1%), despite recent calls that software development funded by tax money should be publicly available. On the other hand, somewhat hypocritically, 41.9% of the GovAgs strive to integrate available OSS into their own solutions. Open innovation initiatives require domain knowledge, and we recommend GovAgs to find inspiration in successful software ecosystems in the private sector. Moreover, GovAgs spearheading openness rather than lagging behind would perfectly match OECD’s promises of transparency and trustworthiness through digital governments. Opposed to private companies struggling with making profits on the market, GovAgs are not competitors – thus it seems like a real opportunity for them to openly collaborate.

Software development in the public sector offers a different context than the market-driven development typically studied by the software engineering community. We provide insights into state-of-practice development in Swedish GovAgs and the prevalence of a selection of software engineering best practices (**RQ3**). In line with previous work, we conclude that public sector development resembles its private sector counterpart. A majority of the GovAgs develop software iteratively using modern development tools and several best practices are implemented, e.g., software processes, requirements engineering, close communication with end-users.

Software quality requirements are an acknowledged development challenge that inevitably introduce trade-offs. Based on the qualities defined by ISO/IEC 25010 [12], our survey shows that GovAgs, in line with private sector companies, tend to prioritize functional suitability. Several GovAgs instead consider security to be the most important (**RQ4**), and also usability and reliability are frequently highlighted as important qualities. Performance efficiency is rarely a primary concern, but related keywords often occur in the GovAgs’ System Requirements Specifications (SRS). More importantly, few GovAgs use systematic approaches to prioritize different qualities. Based on the lack of systematic approaches, combined with indications of poor alignment between strategic goals and operationalized reality, we recommend GovAgs to focus process improvements on software qualities – to ensure future trust in the software that drive the digitalization of Swedish society.

This paper summarizes the first step in a larger ambition to study software development during the digitalization of the Swedish public sector. Our planned research is needed, as the digital transformation is happening fast and large amounts of tax money are at stake – the software engineering community ought to contribute its experience and support decision-makers as consultation bodies.

The Swedish Government has an appointed minister for Housing and Digital Development, but the double duty shows that the question has not been given sufficient weight. There is a silver lining, however, as the Swedish Government announced in the budget bill for 2018 that a new GovAg will be commissioned from September 2018. The new GovAg will shoulder an overall responsibility to coordinate and support the governmental digitalization at large, possibly in line with the Danish Agency for Digitisation – established already in 2011. We are eagerly awaiting the new GovAg, and hope to contribute our research perspectives for a successful – and accelerating – digitalization of the Swedish public sector.

⁹In the documents, security was not searched for directly, and the number given is the union of the documents where confidentiality, integrity, and availability occurred.

ACKNOWLEDGMENTS

Thanks go to all respondents. The work is partially supported by a research grant for the ORION project (reference number 20140218) from The Knowledge Foundation in Sweden and by a grant for the DRISTIG project by the Swedish Civil Contingencies Agency, MSB (agreement no. 2015-6986).

REFERENCES

- [1] S. Assar, I. Boughzala, and I. Boydens. 2011. Back to Practice, a Decade of Research in E-Government. In *Practical Studies in E-Government: Best Practices from Around the World*, S. Assar et al. (Ed.). Springer Science+Business Media, 1–12.
- [2] T. Assuncao, I. Rodrigues, E. Venson, R. Figueredo, and T. Sousa. 2015. Technical Debt Management in the Brazilian Federal Administration. In *Proc of the 6th Brazilian Workshop on Agile Methods*. 6–9.
- [3] D. Badampudi, K. Wnuk, C. Wohlin, U. Franke, D. Smitte, and A. Cicchetti. 2017. A Decision-making Process-line for Selection of Software Asset Origins and Components. *Journal of Systems and Software* (2017).
- [4] P. Bannerman. 2007. Software Project Risk in the Public Sector. In *Proc. of the 18th Australian Software Engineering Conference*. 389–398.
- [5] R. Bertsson Svensson, T. Gorschek, B. Regnell, R. Torkar, A. Shahrokni, and R. Feldt. 2011. Quality Requirements in Industrial Practice - An Extended Interview Study at Eleven Companies. *Trans. on Software Engineering* 38, 4 (2011), 923–935.
- [6] A. Beskow and K. Wnuk. 2016. *SCALARE Deliverable WP4 - Government Agency*. Project Deliverable. http://book.scalare.org/wp-content/uploads/2016/11/CaseStudy_GovernmentAgency.pdf
- [7] M. Borg, T. Olsson, U. Franke, and S. Assar. 2018. *Digitalization of Swedish Government Agencies - Detailed Census Description and Analysis*. Technical Report SICS T2018:02. diva2:1179537
- [8] G. Bowen. 2009. Document Analysis as a Qualitative Research Method. *Qualitative Research Journal* 9, 2 (2009), 27–40.
- [9] P. Cilek, W. Janko, S. Koch, A. Mild, and A. Taudes. 2004. A Hedonic Wage Model-based Methodology for Evaluating the Benefits of IT Investments in Public-sector Organisations. *Journal of Enterprise Information Management* 17, 4 (2004), 269–275.
- [10] European Commission. 2017. *Europe's Digital Progress Report 2017*. Technical Report SWD(2017) 160 final.
- [11] B. Fitzgerald, K. Stol, S. Minör, and H. Cosmo. 2017. *Scaling a Software Business: The Digitalization Journey*. Springer.
- [12] International Organization for Standardization. 2011. *Systems and Software Engineering - Systems and Software Quality Requirements and Evaluation (SquaRE)*. Technical Report ISO/IEC 25010:2011(E).
- [13] World Economic Forum. 2016. *The Global Information Technology Report 2016 - Innovating in the Digital Economy*. Technical Report ISBN: 978-1-944835-03-3.
- [14] F. Fowler. 2013. *Survey Research Methods*. SAGE Publications.
- [15] Free Software Foundation Europe (FSFE). 2017. *Public Money, Public Code*. <https://publiccode.eu/>
- [16] S. Goldfinch. 2007. Pessimism, Computer Failure, and Information Systems Development in the Public Sector. *Public Administration Review* 67, 5 (2007), 917–929.
- [17] A. Grönlund and T. Horan. 2005. Introducing e-Gov: History, Definitions, and Issues. *Comm. of the AIS* 15, 1 (2005).
- [18] M. Gupta and D. Jana. 2003. E-government Evaluation: a Framework and Case Study. *Government Information Quarterly* 20, 4 (Jan. 2003), 365–387.
- [19] J. Krogstie. 2012. Comparing Private and Public Sector on Information Systems Development and Maintenance Efficiency. In *Electronic Government*, H. Scholl et al. (Ed.). Springer Berlin Heidelberg, 222–233.
- [20] J. Larsson and M. Borg. 2014. Revisiting the Challenges in Aligning RE and V&V: Experiences from the Public Sector. In *Proc. of the 1st International Workshop on Requirements Engineering and Testing*. 4–11.
- [21] J. Larsson, M. Borg, and T. Olsson. 2016. Testing Quality Requirements of a System-of-Systems in the Public Sector - Challenges and Potential Remedies. In *Proc. of the 3rd International Workshop on Requirements Engineering and Testing*.
- [22] H. Lee and M. Kim. 2013. Implementing Cloud Computing in the Current IT Environments of Korean Government Agencies. *International Journal of Software Engineering and Its Applications* 7, 1 (2013), 149–160.
- [23] J. Liang and H. Jin. 2013. Integrating Local E-Governments of China to Provide Better Public Services Based on Cloud Computing. *SpringerLink* (2013), 893–898.
- [24] C. Lin, G. Pervan, and D. McDermid. 2007. Issues and Recommendations in Evaluating and Managing the Benefits of Public Sector IS/IT Outsourcing. *Information Technology & People* 20, 2 (2007), 161–183.
- [25] K. Lindström. 2017. Nu ska IT-kolossen Skatteverket bli agil. *Computer Sweden* (Sept. 2017). <https://computersweden.idg.se/2.2683/1.688986/skatteverket-agil-koloss>
- [26] L. Luna-Reyes, R. Gil-Garcia, and G. Romero. 2012. Towards a Multidimensional Model for Evaluating Electronic Government: Proposing a More Comprehensive and Integrative Perspective. *Government Information Quarterly* 29, 3 (2012), 324–334.
- [27] L. Lwakatara, P. Kuvaja, and M. Oivo. 2016. An Exploratory Study of DevOps Extending the Dimensions of DevOps with Practices. In *Proc. of the 11th International Conference on Software Engineering Advances*.
- [28] C. Matt, T. Hess, and A. Benlian. 2015. Digital Transformation Strategies. *Business & Information Systems Engineering* 57, 5 (2015), 339–343.
- [29] D. Mendez Fernandez, S. Wagner, M. Kalinowski, M. Felderer, P. Mafra, A. Vetro, T. Conte, M. Christiansson, D. Greer, C. Lassenius, T. Mannisto, M. Nayabi, M. Oivo, B. Penzenstadler, D. Pfahl, R. Prikladnicki, G. Ruhe, A. Schekelmann, S. Sen, R. Spinola, A. Tuzcu, J. de la Vara, and R. Wieringa. 2017. Naming the Pain in Requirements Engineering. *Empirical Software Engineering* 22, 5 (2017), 2298–2338.
- [30] R. Moreaux. 2014. Logiciel de paie des fonctionnaires: les raisons de la debacle. *acteurs publics* (April 2014). <https://www.acteurspublics.com/2014/04/29/logiciel-de-paie-des-fonctionnaires-les-raisons-de-la-debacle>
- [31] E. Murphy-Hill, T. Zimmermann, and N. Nagappan. 2014. Cowboys, Ankle Sprains, and Keepers of Quality: How is Video Game Development Different from Software Development? ACM, 1–11.
- [32] T. Ochs and U. Riemann. 2018. *IT Strategy Follows Digitalization*. IGI Global. <https://www.igi-global.com/chapter/it-strategy-follows-digitalization/183799>
- [33] OECD Council. 2014. *OECD Recommendation on Digital Government Strategies - OECD*. <http://www.oecd.org/gov/digital-government/recommendation-on-digital-government-strategies.htm>
- [34] United Nations Department of Economic and Social Affairs. 2005. *United Nations e-Government Survey*. Technical Report. <http://publicadministration.un.org/egovkb/en-us/>
- [35] Swedish Ministry of Justice. 2009. *Public Access to Information and Secrecy Act*. Technical Report ISBN 978-91-38-23271-2.
- [36] P. Ottsjö and J. Kristensson. 2017. Myndigheternas data stannar kvar i Sverige. *Ny Teknik* (Aug. 2017). <https://www.nyteknik.se/digitalisering/myndigheternas-data-stannar-kvar-i-sverige-6867327>
- [37] P. Patanakul. 2014. Managing Large-scale IS/IT Projects in the Public Sector: Problems and Causes Leading to Poor Performance. *The Journal of High Technology Management Research* 25, 1 (2014), 21–35.
- [38] P. Rayson and R. Garside. 2000. Comparing Corpora Using Frequency Profiling. Association for Computational Linguistics, 1–6.
- [39] B. Regnell, R. Bertsson Svensson, and T. Olsson. 2008. Supporting Roadmapping of Quality Requirements. *IEEE Software* 25, 2 (2008), 42–47.
- [40] B. Robson. 2002. *Real World Research* (2nd ed.). Blackwell.
- [41] K. Rosacker and R. Rosacker. 2010. Information Technology Project Management Within Public Sector Organizations. *Journal of Enterprise Information Management* 23, 5 (2010), 587–594.
- [42] R. Schmidt, A. Zimmermann, M. Möhring, S. Nurcan, B. Keller, and F. Bär. 2015. Digitization - Perspectives for Conceptualization. In *Advances in Service-Oriented and Cloud Computing*, A. Celesti and P. Leitner (Eds.). Springer, 263–275.
- [43] E. Scott, D. Pfahl, R. Hebig, R. Heldal, and E. Knauss. 2017. Initial Results of the HELENA Survey Conducted in Estonia with Comparison to Results from Sweden and Worldwide. In *Proc. of the 18th International Conference on Product-Focused Process Improvement*. 404–412.
- [44] S. Sentilles, E. Papatheocharous, F. Ciccozzi, and K. Petersen. 2016. A Property Model Ontology. In *Proc. of the 42th EuroMicro Conference on Software Engineering and Advanced Applications*. 165–172.
- [45] Statens servicecenter. 2017. *En gemensam ställing molntjänst för myndigheternas IT-drift*. Technical Report R:001.
- [46] IEEE Computer Society. 2015. *Guide to the Software Engineering Body of Knowledge (SWEBOK Guide)*. Technical Report ISO/IEC TR 19759:2015.
- [47] S. Stavru. 2014. A Critical Examination of Recent Industrial Surveys on Agile Method Usage. *Journal of Systems and Software* 94 (2014), 87–97.
- [48] E. Ziemba and I. Kolasa. 2015. Risk Factors Framework for Information Systems Projects in Public Organizations - Insight from Poland. In *Proc. of the Federated Conference on Computer Science and Information Systems*. 1575–1583.