MAKERERE UNIVERSITY

Characterization of practices and Measurements in Software Start-ups in an Emerging Ecosystem

BY

KAMULEGEYA GRACE BUGEMBE
BSc (Mak), HDip(Wits), MSc(UCT)

2016/HD05/1896U

PHD IN DATA COMMUNICATIONS AND SOFTWARE ENGINEERING

Supervisors
REGINA HEBIG, PhD
MUGWANYA RAYMOND, PhD

A THESIS SUBMITTED TO THE DIRECTORATE OF RESEARCH AND GRADUATE TRAINING FOR THE AWARD OF THE DEGREE OF DOCTOR OF PHILOSOPHY OF MAKERERE UNIVERSITY, KAMPALA.

APRIL, 2022

Abstract

Software start-ups support the development of innovative products and services. They can support the transformation of industrial sectors in terms of new technologies and ways of working, and they can promote sustainable growth of developing economies. However, 90% of start-ups have been found to fail in their first two years yet it is beneficial to have as many successful start-ups as possible in a region in order to achieve sustainable growth or sustainable transformation. Start-up hubs are a mechanism to promote start-ups and generate new ones. The concept of start-up hubs is internationally established and there are numerous examples of success. However, there is still not much documented experience with emerging start-up hubs, especially in East Africa and Uganda in particular, where a start-up ecosystem has to be established first, such experiences are important. The question here is whether established practices will be successful with start-ups and hubs in these emerging start-up ecosystems. There is also the question of appropriate metrics to assess the progress and success of start-ups. The five empirical studies in this thesis go exactly in this direction using case studies and a survey in Uganda and Kenya. In the first study, we aimed to learn whether software start-up patterns in the Swiss-Finnish ecosystem apply to Ugandan start-ups and interviewed seven software start-ups. The Ugandan start-ups identify with similar aspects of the context and problem addressed and adopt similar solutions of the five patterns solutions found in the Swiss-Finnish start-ups. The result confirmed the generalizability of patterns across cultural and ecosystem differences. Since hubs are vital in nurturing start-ups, our second study investigated operations of hubs and metrics the hubs used to assess progress of their software start-ups. Hubs were found to lack standard start-up selection criteria, provide networking, team-building events, value addition activities, and mainly give business growth incentives to start-ups. They also notice business and organizational effects of their software start-ups' incentives, provide incentives to alumni start-ups, and measure business and scalability metrics. In our third study, we investigated metrics in early-stage start-ups and the perceived benefits of using them. We found that start-ups in East Africa use business and product-oriented metrics and expect to benefit from measuring. Despite the differences in the metrics, both start-ups and mature software companies cover similar metrics in decision making, e.g., business, organizational and technical metrics. However, what influences the choice and use of metrics in start-ups and mature software companies is unexplored. In the fourth study, we aimed at understanding the differences better by deriving ten characterizing dimensions for the metrics and surveyed 19 software start-ups and ten mature software companies in Uganda. The results show that metrics used in start-ups and mature companies differ regarding the frequency of assessment, automation of the measuring process, type of investment in the measuring tools, and the metric's final form. Metric choices for start-ups in emerging ecosystems are not necessarily the same as those in developed ecosystems. Nonetheless, metrics, often visualized with dashboards, are still considered crucial in helping start-ups to focus on the right aspects. Our fifth study aimed to identify key requirements for measurement dashboards in start-ups in emerging East Africa ecosystem and the expected benefits of using the dashboards. Results indicate that start-ups want dashboards to visualize performance and

expect performance-related benefits. The empirical studies in this thesis thus contribute to 1) the knowledge of practices, patterns, and measurements in software start-ups and hubs, 2) a new perspective to the characterization of software start-up metrics, and 3) a set of requirements that start-ups can base on to build measurement dashboards.

In conclusion, start-ups in the emerging ecosystems face similar challenges to those in developed ecosystems and largely address them similarly to those in developed ecosystems with slight variation in solutions potentially due to contextual differences. Hubs as key ecosystem actors variably select start-ups and provide incentives while focusing on business and scalability metrics to track their start-ups. The metrics used in start-ups can potentially be chosen differently compared to those used in mature companies and software start-ups like other business will want to use dashboards that visualize their performance.

Keywords: Software Start-up Patterns, Emerging Start-up Ecosystem, Hub Operations, Metrics, Metric Dimensions, Measurement Dashboards

Declaration

I, **Kamulegeya Grace Bugembe**, declare that the contents in this thesis are my original work and have never been presented to any university or institution of higher learning for any award

Kamulegeya Grace Bugembe

Signed: .....

Date: 23rd | March | 2022

This thesis has been submitted for examination with the approval of the following supervisors:

Associate Prof. Regina Hebig
B.Sc, M.Sc., Dr. rer. nat

Signed: .....

Date: 24.03.22

Raymond Mugwanya
B.Stat, M.Sc, PhD

Signed: .....

Date: 25.03.2022

Dedication

I dedicate this PhD to the memory of my beloved parents Taata Eliphaz Bugembe and Maama Justine Bugembe.

Acknowledgement

This thesis would not be called by that name without the unfaltering academic guidance of Associate Prof. Regina Hebig. I am indeed very grateful for the many skype guidance meetings, the countless hours spent with me to do revision and critique the manuscripts, the often genius pointers into the painful but rewarding depths of analysis for results and of course the support for my two hugely productive visits to Chalmers and Gothenburg University. Without that effort, I would have walked the proverbially 'long journey to complete PhDs by local students'.

I am too very thankful to Dr Raymond Mugwanya for the time he would spare to touchbase and get updates about the progress of this study. Indeed, those times were wake up calls for me to re-energize and keep my eyes on the ball to this day.

In the four-year journey of this PhD study, I will like to give special thanks to Associate Prof. Engineer Bainomugisha and Prof. Michel Chauldron for ably guiding the BRIGHT project under the Swedish International Development Agency (SIDA) that believed and recruited me as a PhD research student. Your unwavering administrative and financial support and oversight, indeed 'steadied the ship' for which I stand to be called a PhD holder.

Special thanks go to Prof. Buyinza Mukadasi, the Director of the Directorate of Research and Graduate Training (DGRT) for the often sobering but cordial conversations and numerous meetings we had about the SIDA funding Standard Operating Procedures (SOPs). Indeed, these meetings ensured that I kept my eyes on the prize of completing and walking away as a successful PhD.

In a special way, I thank my wife Winnie Kamulegeya, daughters Ellah, Kei and Kellyn for the patience and social support that has kept me sane during this journey. To this end, I am still the loving husband and father. With your support, I have achieved this personal milestone that God has blessed.

My sincere gratitude go to Dr.Fredrick Kiggundu and wife Edith Kiggundu (PhD) for the tireless effort and sacrifice they made to educate me in the best schools and universities in Africa (Kings College Budo, St. Mary's College Kisubi, Makerere University, The University of Witwatersrand and University of Cape Town). I am truly proud for the unwavering support from your family and often your wise counsel of "only the best is good enough". I also take this opportunity to acknowledge the wide ranging social support that my 'large' Bugembe family has accorded to me during my entire academic journey.

Special thanks go to Prof Imed Hommouda of Chalmers University and Mediterenean Institute of Technology (MedTech) for the academic camaraderie and the often intriguing questions about life after the PhD. For sure, 'the patterns' will go far and beyond this academic milestone.

I have studied with many students in the BRIGHT team. These I cannot list all of them, but special thanks to my friends in this struggle for academic success, Dragule Swaib, Allan Mazimwe, Hawa Nyende. For sure, the camaraderie we shared during those lonely visits to Sweden and Tunisia meant a lot.

Funding: This research was funded in part by the Swedish International Development Cooperation Agency (SIDA) and Makerere University under SIDA contribution No: 51180060; Project No. 317; Building Research Capacity in Innovative Information and Communication Technologies for Development (ICT4D) for Sustainable Social-economic Growth in Uganda (BRIGHT)

List of Publications

This thesis is a compilation of the following five (5) papers presented as Chapters 4, 5, 6, 7, 8:

[A] Kamulegeya, G., Hebig, R., Hammouda, I., Chaudron, M., & Mugwanya, R. (2017, August). Exploring the Applicability of Software Start-up Patterns in the Ugandan Context. In Software Engineering and Advanced Applications (SEAA), 2017 43rd Euromicro Conference on (pp. 116-124). IEEE.

[B] Kamulegeya, G., Mugwanya, R., & Hebig, R. (2020, August). The Character of Software Start-up Hubs in an Emerging Ecosystem. Accepted for publication in 2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA) (pp. 256-264). IEEE.

[C] Kamulegeya, G., Mugwanya, R., & Hebig, R. (2018). Measurements in the Early Stage Software Start-ups: A Multiple Case Study in a Nascent Ecosystem. Journal of Foundations of Computing and Decision Sciences, 43(4), 251-280. doi: https://doi.org/10.1515/fcds-2018-0014

[D] Kamulegeya, G., Mugwanya, R., & Hebig, R. (2020). The Character of Metrics: A Survey of Start-ups and Mature Software Companies in an Emerging Ecosystem: [In submission for review to the Journal of Foundations of Computing and Decision Sciences]

[E] Kamulegeya, G., Mugwanya, R., & Hebig, R. (2019, August). Requirements for Measurement Dashboards and Their Benefits: A Study of Start-ups in an Emerging Ecosystem. In 2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA) (pp. 300-308). IEEE.

Contents

List of Tables

List of Figures

Chapter 1

General Introduction

This chapter presents the motivation for the comprehensive study in this thesis and identifies the research problem. It also highlights the study's primary and specific objectives and the leading research questions answered by the thesis's various research publications.

1.1 Background

Software start-ups primarily develop software and depend on providing value as software products and as a service. Several authors define and describe software start-ups. Shi et al. represent software start-ups as newly created firms developing software-intensive products/services [4]. Ries [5] defines a start-up as a *" human institution designed to deliver a new product or service under conditions of extreme uncertainty"*. Blank et al.[6] define a start-up as a *"temporary organization designed to search for a repeatable and scalable business model"*. A report by the start-up Genome Project [7] on more than 3200 start-ups presents a more elaborate definition as it merges and adapts the definition of start-ups by Steve Blank and Eric Ries. It defines start-ups as *" temporary organizations designed to scale into large companies. Early-stage start-ups are designed to search for product/market fit under conditions of extreme uncertainty. Late-stage start-ups are designed to search for a repeatable and scalable business model and then scale into large companies designed to execute under conditions of high certainty"*—this definition highlights start-up goals and challenges concerning key growth phases. These varied definitions take both technical and business perspectives to software start-ups. They also imply that software start-ups apply software engineering and business practices to develop, support, and sell their software products and services. Thus, software start-ups represent a complex phenomenon that is subjected to considerable internal and external influences that may determine their overall success.

The use of software engineering practices has mostly been ad hoc in software start-ups compared to large and mature software companies. A variation in practice that could be attributed to the various challenges start-ups face. These challenges are highlighted by Paternoster et al. [8], and Sutton [9] characterization of software start-ups. The highlighted challenges indicate that start-ups have little or no operating history, are highly reactive and flexible, generally lack resources, are subject to multiple influences in often uncertain environments, work with cutting edge technologies under intense time pressures and operate in fast-growing markets. Some of the challenges are internal to start-ups, but others are influenced by the start-up ecosystems. The various difficulties could imply that the way they are solved may vary, as highlighted in our earlier study [10].

Start-ups face known challenges and are usually attracted to services in hubs that alleviate some of the challenges. The services in hubs generally include incubation, co-working spaces, and acceleration. The services nurture the start-ups and aim to grow the start-ups into sustainable businesses. In the early-stage, start-ups focus on developing, testing, and validating their products. As they grow, they develop and validate their business models and acquire paying customers.

In a broader context, start-ups exist in ecosystems. Start-up ecosystems are macro environments that influence early-stage start-up growth. Some of the characterizing traits of start-ups point to forces in their ecosystems. Tripathi et al. [11] report the composition of an ecosystem as entrepreneurs, technology, market, support factors, finance, human capital, education, and demography. A study by Giardino et al. [12] notes that 60% of the start-ups do not survive the first five years, and 75% venture capitalists funded start-ups fail in the US. Another report by the Start-up Genome [7] puts start-up failure at 90% in the first two years. Failures of the start-ups could be attributed to the challenges the start-ups face. Some studies have highlighted premature scaling[7], inconsistency in managerial strategies, and execution [12] as reasons for software start-up failure. Studies by [13][14] have documented challenges faced by start-ups. These include lack of access to initial funding, difficulty acquiring paying customers, ever-changing technology, dynamic markets, technical debt, poor infrastructure, and time pressures to deliver Minimal Viable Products (MVP), among others.

Studies by [15][16] in different countries indicated that vibrant start-up ecosystems have enablers for start-ups to survive. The enablers include policies, resources, entrepreneurship economies. According to Cukier et al., [17], the model for the maturity of software start-up ecosystems, the Ugandan ecosystem, and the wider East African ecosystem can be categorized as emerging. An emerging ecosystem is characterized as a recognizable start-up hub, with already existing start-ups, a few investment deals, and no great output in terms of jobs or worldwide penetration.

1.2 Statement of the Problem

In this section, we identify the research gaps that underpin the studies in this thesis. A number of existing studies have highlighted common [9] [8] [12] and engineering-related [18] [19] challenges in software start-ups. The challenges may be tagged to practices and patterns in software start-ups. A preliminary study [10] aimed to discover how published patterns in Swiss-Finnish start-ups apply in Uganda software start-ups. The study was limited in scope (carried out in Ugandan start-ups) but concluded that start-up patterns could often be transferred in as much as it found variations in contexts and solutions to some patterns in the Ugandan software start-ups. For example, start-ups in Uganda access initial funding through grants and self-funding, a similar funding gap identified in Kenyan start-ups[20]. In contrast, software start-ups in the Swiss-Finish context (a developed ecosystem) had no major funding gaps. The availability of funds for the start-ups in the developed ecosystems could point to the existence of angel investor and Venture Capital (VCs) networks [7]-these being very few in East Africa [21]. The evident variation of contexts and solutions in Ugandan start-ups for some known patterns elsewhere indicates potential ecosystem influences.

There are currently no known empirical studies that characterize the practices and patterns specific to software start-ups and hubs in the emerging East Africa ecosystems. Understanding these practices and patterns is an essential step in identifying and benchmarking practices that can be improved to increase software the success of software start-ups. We

define a practice as a technical process or business activity in a software start-up and a pattern as a common practice in multiple software start-ups or hubs for this study.

Hubs play an important role in nurturing early-stage start-ups into financially viable and self-supporting start-ups[20]. Currently, Uganda and the East Africa region are experiencing a proliferation of hubs-related activities. The activities include innovation competitions, hackathons, business mentoring, acceleration, co-working space services, bootcamps nurturing technical and business skills. The hubs are set up for-profit, not-for-profit, or combine the two models but generally aim to attract software and non-software intensive start-ups and Small to Medium Enterprises (SMEs). Chirchietti highlights the importance of hubs in 1) helping start-ups to develop and launch new business models, 2) linking the start-ups to their environment, and 3) promoting innovation and entrepreneurship[20]. Steve Blank distinguishes SMEs from start-ups by highlight that SMEs do not necessarily intend to grow [6]. The hubs provide enabling resources, incentives, training, and support infrastructure to the start-ups and SMEs. These enablers are needed by the early-stage start-ups to increase their chances of success since the start-ups are normally limited in resources, networks, infrastructure, and expertise.

A study in Kenya found that hubs lacked skills in monitoring and evaluating their progress [22]. A widely cited early study [22] of hubs in East Africa describes hubs based on size, age, tenants, partnerships, and funders and identified common features. This hub characterization assumes similar operations, resources, and incentives for nurturing all types of start-ups and SMEs to success. However, the assumption of hub operations may not generally hold for software start-ups. Soft start-ups produce intangible products/services using dynamic technologies and require a combination of software engineering and business skills to grow. In the case of Uganda and East Africa, hubs and start-ups operate in an emerging ecosystem - when you classified based on Cukier et al. [17] ecosystem maturity model. In this model, emerging ecosystems are characterised to comprise already existing start-ups, few investment deals, optional government initiatives to support the ecosystem's development, and no relevant output in terms of jobs or worldwide penetration. There is also low-technology literacy, mostly agricultural economies, and little government support [20].

Given the nature of the East African ecosystem and the uniqueness of software start-ups and the hubs' hybrid business models, it is inconceivable to assume that the hubs in this ecosystem operate in the same way to nurture early-stage software start-ups. There are no published studies about the East African ecosystem that focus on hubs' operations concerning nurturing software start-ups. There are no published empirical studies that focus on how the hubs measure the progress of their early-stage software start-ups. Understanding these issues will be an essential step for hub operators and the early-stage software start-ups to benchmark their practices and identify critical operational metrics.

Software start-ups founders continuously decide to build their products, learn and optimize and their business models. A study that relates software start-ups' survival to managerial actions looked at actions start-ups' management took to deploy resources and capabilities to achieve high-performance [4]. Another study by Staron and Meding [1] about Measurement Programs in large software organizations looked at the use of metrics to monitor products, processes, and projects. Metrics have widely been studied in software engineering, focusing on size, design, complexity, software process monitoring, and software quality aspects like reliability and maintainability. For large software companies, metrics to measure aspects like organization and project management have also been studied. Metrics have been mainly used for decision making in both large software companies and software start-ups and understanding what and how they are used in emerging ecosystems is important in explaining how the start-ups and hubs operate.

The environment and ecosystem in which start-ups operate potentially influences their challenges. We thus cannot assume that start-ups measure the same aspects of their businesses since metrics identified in software start-ups are mainly found in software start-ups in developed ecosystems. However, the East Africa ecosystem can be characterized as an emerging ecosystem based on Cukier et al. [17], ecosystem maturity model. A study[10] of the Ugandan ecosystem has revealed some variations in practices and patterns in software start-ups. This variation potentially points to contextual differences between the studies done in the developed start-up ecosystem and emerging ecosystems. No matter the ecosystem's context, start-ups make decisions as they pivot their technology, improve their business models, and do validated learning. Metrics form a basis for most decisions in large software companies, and metrics are used similarly in software start-ups in the developed start-ups' ecosystems. However, little is known of the metrics and the nature of metrics used in software start-ups in the emerging software start-up ecosystem in East Africa. Knowing the metrics used and their character could be an essential step in understanding what motivates the metrics' choice. The metrics' knowledge could be a basis for developing tools that can aid validated decision-making in the start-ups to track and measure their progress.

However, to the best of our knowledge, no empirical studies have explored metrics used in software start-ups and hubs in developing start-up ecosystems.

1.3 Study Objectives and Research Questions

This empirical study aims to explore practices and measurements by software start-ups in the emerging East Africa ecosystem.
To guide the study in achieving its main objective, we asked this overarching research question:

- RQ: How do software start-ups in an emerging East African ecosystem operate?

We breakdown the main objective into the following specific objectives that we achieve by answering the related research questions. The research questions are answered in the subsequent Chapters 4, 5, 6, 7, 8

- **Practices and Patterns:** [O1] To assess practices and patterns in software start-ups and hubs in East Africa
- **MRQ1**: How do software start-ups and hubs operate in emerging East Africa start-up ecosystem?
- **Measurement Character:** [O2] To characterize measurements in software start-ups and hubs in the emerging East Africa start-up ecosystem.
- **MRQ2**: What do software start-ups and hubs in emerging ecosystems measure?
- **Measurement Support:** [O3] To evaluate progress metrics in early-stage software start-ups in the emerging East Africa start-up ecosystem.
- **MRQ3**: How can tools support measurements in early-stage software start-ups?

1.4 Research Contributions

In this section, we present the contribution from each chapter presented as papers. This thesis makes the following key contributions to software start-ups research and the wider area of software engineering. We present the contribution from each of the studies presented as papers; C1- Chapter 4-, C2- Chapter 5, C3- Chapter 6, C4- Chapter 7, C5- Chapter 8

C1 ***Patterns, practices in Ugandan start-ups:*** The study in Chapter 4 validates a study by Dande et al. [14] of common software start-up practices in the Swiss-Finnish ecosystem expressed in the form of patterns. Our study in Chapter 4 identifies practices and patterns in Ugandan start-ups found in the Swiss-Finnish ecosystem and identifies context-specific variants of some practices. Given that we know the current practices in early-stage software start-ups in Uganda, future work can identify and characterize the most effective practices in the more successful software start-ups in Uganda and East Africa as an emerging software start-up ecosystem.

C2 ***Characterisation of operations in hubs in East Africa:*** In Chapter 5, we present a study that investigates how hubs work and measure the progress of their software start-ups. The study involved ten interviews with hub managers.

Studies on start-up ecosystems and players concerning software start-ups are scarce (notably [23, 24, 25]), as many studies have focused on how software start-ups operate internally. Therefore, this study makes an original knowledge contribution (through its empirical approach) to the area of software start-up ecosystems.

This empirical study within the emerging East Africa ecosystem forms a basis to document current software start-ups and hub practices. This knowledge base can be used as a benchmark in similarly emerging ecosystems elsewhere in the world.

C3 ***Key internal metrics for early-stage start-ups*** In Chapter 6 we present a study that interviewed 19 software start-ups from 6 hubs in 2 East African countries (Uganda and Kenya). The study uses the framework MeSRAM[1] to classify the start-ups and guide the analysis of the data. It sought to identify what metrics software start-ups in the emerging East African ecosystem measured and the benefits they expected from measuring. Now that we know what metrics start-ups in East Africa use, future research can investigate the most useful metrics to improve the start-ups' decision-making and why they make metric choices.

C4 ***Dimensions to characterize early-stage software start-up metrics:*** In Chapter 7, we derive ten dimensions. The dimensions are motivated by good metrics' characteristics, mapping the metrics to target users by Daskalantonakis [26], and start-up challenges to represent metrics. The dimensions are used to survey 19 software start-ups and ten mature software companies in Uganda. The study aims to characterize metrics and identify dimensions that differentiate their use in software start-ups and mature companies. The dimensions reveal similarities and differences in metrics used in start-ups and mature software companies. Understanding whether and how software start-ups use metrics can be a valuable contribution to the state-of-the-art. The use of metrics in start-ups is important since software start-ups' scope of metrics might be beyond the software engineering areas. This is because software start-ups would naturally be interested in metrics for financial and business plans. This study contributes to the first step in understanding what might significantly motivate software start-ups to use specific metrics considering their challenges compared to mature software firms. The ten dimensions contribute as an objective lens

through which start-ups can select their metrics. Further, the dimensions provide insights for future work to understand the challenges start-ups face in adopting metrics as they grow.

C5 ***Requirements for development of measurement tools*** The study presented in Chapter 8 investigated the requirements for measurement dashboards software start-ups in East Africa and their benefits to software start-ups. We conducted 36 semi-structured interviews in software start-ups in Uganda and Kenya, where we discovered a range of requirements and benefits. Dashboards have been applied in mature and large software companies [27], [3] but not in software start-ups as everyday practices. The study in Chapter 8 demonstrates that, despite this, there are needs for dashboards for various reasons, and the software start-ups do perceive the benefits of using them. Now that we know the requirements for measurement-based dashboards, we can develop information dashboards to aid software start-ups to validate decision-making. A measurement prototype tool[28] based on the requirements and metrics identified in the study in Chapter 5 exemplifies this.

1.5 Mapping Research Questions

Table 1.1 maps the specific Research Questions (MRQ1, MRQ2, MRQ3) to specific objectives (O1, O2, O3) of the study and the contributions (C1, C2, C3, C4, C5) from the five papers published from the overall study. In the following table, we map the main research

Table 1.1. Research Questions, Objectives Contribution Mapping

Research Questions	Specific Objectives	Contributions
(MRQ1) How do software start-ups and hubs operate in the emerging East Africa start-up ecosystem?	**(O1)** To assess practices and patterns in software start-ups and hubs in East Africa	**(C1)** Application of software start-ups patterns from Swiss-Finnish ecosystem (developed ecosystem) to Ugandan start-ups (an emerging start-up ecosystem). **(C2)** Characterization of operations software start-up hubs in East Africa
(MRQ2) What do software start-ups and hubs in emerging ecosystems measure?	**(O2)** To characterize measurements in software start-ups and hubs in the emerging East African ecosystem	**(C3)** Ten dimensions to characterize metrics in start-ups. **(C4)** Metrics used in software start-ups and hubs in the emerging East African ecosystem
(MRQ3) How can tools support measurements in early-stage software start-ups?	**(O3)** To evaluate progress metrics in early-stage software start-ups in the emerging East African start-up ecosystem	**(C5)** Requirements for the development of measurement-dashboards

questions, the thesis objectives, and the methods we applied to execute the studies that fulfilled the objectives and answered the research questions. In Table 1.2 we map the main research questions (MRQ1, MRQ2, MRQ3) to the sub research questions addressed by the five studies and characterises the studies. Table 1.2 aligns the main Research Questions to the study specific research questions and further characterises each study in terms of subject of study, the type and number of respondents, the study geographical scope, research methods used, sources of data, data analysis approaches and frames of discussion of the results.

1.6 Definition of Working Terms

In section, we define the following terms to clarify their meaning and use in the study.

- **Early-stage software start-up:** Start-ups in the phase of idea conceptualization to first time to market [29]
- **Mature software company:** are companies having a considerable work history, a management structure, established software products/services with customers, technical employees and have an established physical office.
- **Stage of growth:** This is a start-up growth phase where the software start-up is completing a set of milestones.
- **Practice:** is a technical or business activity/process in a start-up or hub.
- **Pattern:** is a recurrent practice in start-ups or hubs that manifests in a similar context and with similar solutions.

1.7 Thesis Outline

Table 1.3 maps the thesis chapters, publications from the study and objectives they met. Figure 1.1 outlines the thesis layout concerning the specific study objectives and thesis chapters. Chapter 1 presents the general introduction, Chapter 2 highlights the related body of literature encompassing this study, and Chapter 3 wraps the general methodology followed to achieve the entire study's objectives. We present Chapters 4, 5, 6, 7, 8 as papers for specific studies that answer the main research questions and meet the study's objectives. In Chapter 9. we present an overall discussion of the thesis and Chapter 10 presents the conclusion, highlights contributions from the studies, and points to the future research direction.

1.8 Summary

This chapter set the background to the overall study, stated the problem for the entire study, highlighted the objectives and the main research questions, mapped the research objectives to the research and defined the working terms for study. In Chapter 2, we review related literature to challenges software start-ups face, their patterns and practices, hubs as start-up actors and measurements in software engineering and software start-ups in particular.

Table 1.2. A mapping of main research questions to study specific research questions and methods

Main Research Questions	Study specific Research Questions/Objectives	Study subject	No. of subjects	Geographical scope	Research method used	Type of data collected	Sources of data	Analysis approach	Analysis and discussion frameworks
MRQ1	**Objective**: Investigate whether well-known patterns collected from software start-ups in Switzerland and Finland can be found in the same or at least similar form in Ugandan software start-ups.	software start-ups	7 software start-ups	Uganda	case studies	Qualitative (semi-structured interviews)	Start-up founders, start-up patterns by Dande et al. [14]	Thematic analysis	Dande et al. [14]
MRQ1	**RQ1**: How do software start-up hubs operate in a nascent ecosystem? **RQ2**: What metrics are used by software start-up hubs to track start-up progress?	Hubs	10 hubs	Uganda, Kenya	case study	Qualitative (semi-structured interviews)	hub managers/employees	Thematic analysis	Emerging themes
MRQ2	**RQ1**. What are software start-ups in East Africa (an emerging ecosystem) assessing and measuring? **RQ2**. What benefits of measurements do the start-ups perceive? **RQ3**. How do metrics used by East African start-ups compare to lean start-ups in developed countries?	Software start-ups in hubs	19 software start-ups in 6 Hubs	Uganda, Kenya	multiple case study	Qualitative (semi-structured interviews)	Start-up founders	Content analysis [analyzed data at different levels within case, and cross-case analysis]	MeSRAM framework[1], Croll and Yoskovitz [2]
MRQ2	**RQ1**: How do start-ups and mature companies characterize their metrics along a set of dimensions for the characterization? **RQ2**: For which dimensions can significant differences between metrics used in start-ups and mature companies be shown?	software start-ups, mature software companies	19 software start-ups, 10 mature software companies	Uganda	survey	Quantitative (questionnaire)	Start-up founders, experienced company managers	Descriptive statistics, Hypothesis testing,	Derived 10-dimensions
MRQ3	**RQ1**: What are the requirements for measurement dash-boards in software start-ups? **RQ2**: How do start-ups expect to benefit from using measurement dash-boards?	Software start-ups	36 software start-ups	Uganda, Kenya	multiple case study	Qualitative (semi-structured interviews)	start-up founders	Thematic analysis	Performance Management Dash-boards Velcu-Laitinen and Yigitbasioglu [3]

Table 1.3. Mapping of the chapters published as papers to main thesis objectives

		Research Objectives			
Publication chapters	**Main Research Questions Answered**	**Publication Type**	**[O1]** To assess practices and patterns in software start-ups and hubs in East Africa	**[O2]** To characterize measurements in software start-ups and hubs in the emerging East African start-up ecosystem	**[O3]** To evaluate progress metrics in early-stage software start-ups in the emerging East African start-up
Chapter 4: [Paper 1] Exploring the Applicability of Software Startup Patterns in the Ugandan Context	**MRQ1**	Conference	**X**	**X**	**-**
Chapter 5: [Paper 2]The Character of Software Start-up Hubs in an Emerging Ecosystem	**MRQ1**	Conference	**X**	**X**	
Chapter 6: [Paper 3] Measurements in the Early Stage Software Start-ups: A Multiple Case Study in a Nascent Ecosystem	**MRQ2**	Journal	**-**	**X**	**X**
Chapter 7: [Paper 4] The Character of Metrics: A Survey of Start-ups and Mature Software Companies in an Emerging Ecosystem **[Under review]**	**MRQ2, MRQ3**	Journal	**-**	**X**	**X**
Chapter 8: [Paper 5] Requirements for Measurement Dashboards and their Benefits: A study of Start-ups in an Emerging Ecosystem	**MRQ3**	Conference	**-**	**-**	**X**

Note **X** indicates the thesis objective the chapter achieves.

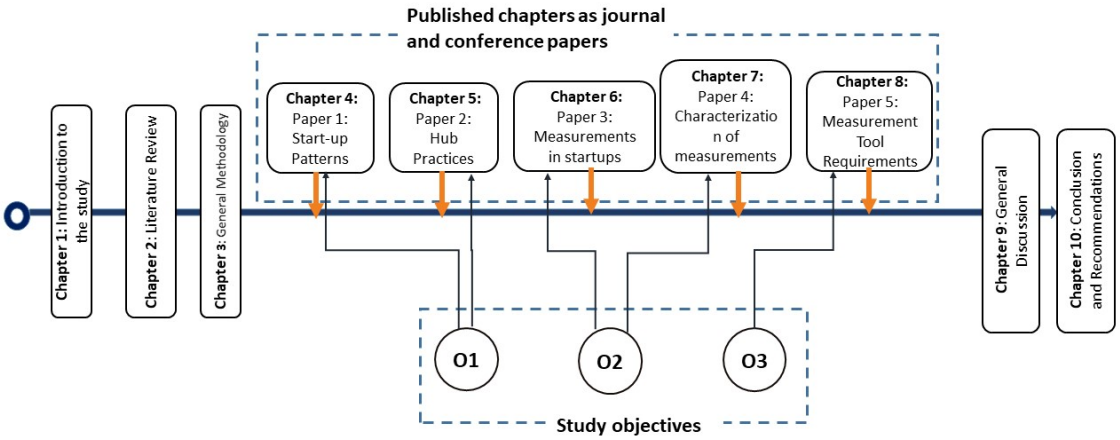

Figure 1.1. Thesis outline in relation to the study objectives

Chapter 2

Literature Review

Software start-ups spend a considerable effort in developing software products that form a cornerstone to their businesses. Existing literature has shown that this effort involves adopting software engineering processes and practices to develop successful products. Many start-ups adopt technical and business approaches that include lean and agile approaches to mitigate the known challenges. Start-ups are bound to be influenced by their ecosystems because they are often small teams, lack time, use dynamic technologies, and face market pressures coupled with a considerable lack of resources. In this chapter, we present related literature about the software start-up phenomenon to understand the work that researchers have undertaken to explore, describe, and explain it while taking varying perspectives.

2.1 Software Start-ups: An Introduction

Blank [6] characterizes a software start-up as an organization in transition, creating innovative high-tech products, with no operating history and that seeks scalable, repeatable, and profitable business models intending to grow. Sutton [9] describes software start-ups as organizations that are constrained by limited resources, unstable markets, multiple influences, dynamic technologies, and immaturity. Coleman and Connor [30], on the other hand, describe start-ups as unique companies that develop software using various processes without a standardized methodology. Unterkalmsteiner et al.[31] further distinguish a small business from a start-up by highlighting that Steve Blank's [6] characterization of the start-up growth ambition distinguishes it from a small business since small businesses do not necessarily intend to grow.

Sutton takes a challenge-based characterization of the start-ups while Coleman and Connor's characterization focuses on the product and variation in processes used in developing the product. Both Blank and Sutton's views tend to take a business perspective to start-ups. On the other hand, Coleman and Connor's view takes a technology focus and could fit start-ups in their early growth stages.

From the various characterization of start-ups, there is generally no consensus on the definition of a software start-up. However, the descriptions and definitions generally take a product and a business leaning characterization. The product characterization highlights the product development processes and at times relate to the start-up challenges. Similarly, the business perspective also highlights business-related challenges. In this study, we characterize software start-ups as organizations that work in uncertain conditions, develop software products/services, and intend to scale. Our characterization considers the ecosystem as a context, the focus of the start-ups on technical, and business aspects and known

challenges they face.

We argue that the challenges faced by software start-ups may influence their choice of technology and business practices. As much as these known challenges are similar in various start-up ecosystems, they may be solved differently. The approaches start-ups use to solve the challenges may thus highlight the contextual influences of the ecosystems. The nature of the software products and the dynamic nature of both technical and business practices in early-stage start-ups increase the uncertainty in software start-ups. Knowing and experiencing these challenges might not be enough to keep a software start-up operational. However, the start-up being in a position monitor the health of its technical and business operations gives it a chance to know what, how, and when these challenges may manifest themselves. The monitoring gives it a chance to optimize and prioritise the often scarce resources available during its early stage.

2.2 Challenges Facing Software Start-ups

Software start-ups intend to develop technology products/services fast in usually dynamic markets. However, given their often short operational history, they face numerous challenges related to software engineering and business processes and outside influences that relate to the software start-up ecosystem in which they operate. This section presents the known challenges and highlights the current solutions that start-ups use to solve them while highlighting the gaps that warrant further research.

Software start-ups as suppliers of software-intensive products or services inevitably use engineering practices even though mostly in ad hoc ways. Given the limited resources, limited understanding of stakeholder needs, and lack of enough funds, the start-ups may not entirely execute formal software engineering processes. However, they are likely to use them in a fit-for-purpose fashion.

A case survey of 84 software start-ups by Klotins et al. [19] found that start-ups mainly faced the same challenges and used the same practices as established companies. However, their primary software engineering challenge was to evolve multiple process areas at once, with a little margin for serious errors. The study uncovered nine engineering challenges in the different engineering aspects of the software start-up in its growth life cycle stages. The start-up team faced challenges that included 1) management of a large team in the maturity phase, 2) lack of team expertise, engagement, coordination, and leadership in the inception and stabilization phase, and the 3) need for specialist skills in the growth and maturity phase. During requirements engineering in the stabilization and growth phases, start-ups faced challenges that included 4)establishment of a feedback loop and 5) feature creeping. To achieve quality goals and testing in the growth and maturity phases, 6) start-ups needed substantial effort to undertake manual regression testing. During the architecture and design in the growth and maturity phase, the start-ups faced a 7) slow down development effort due to technical debt. During project management in the inception and stabilization phase, the start-ups 9) lacked performance benchmarks and needed to 10) manage external stakeholder dependencies.

A notable early study by Block and Macmillan [32] highlights the success factors for starting a new business. These factors include completing prototypes, generating ideas for completing testing a product, and consistently making amendments or re-designs. These success factors may well apply to early-stage software start-ups, given their focus on the product. However, they may fall short in predicting success factors for start-ups as they grow and focus more on organizational and business aspects. Chorev and Anderson [33] present a model that builds on founders' experience to highlight the factors deemed critical for

success in new high-tech ventures in Israel. The model points out that the idea, strategy, core team's commitment, expertise, and marketing are critical and deems management, customer relationships, and research and development critical to the success of new high-tech ventures. This model takes a broader view of high-tech ventures to highlight the success factors. However, these factors can be deemed existent in software start-ups as high-tech businesses in as much as more success factors can be influenced by the nature of software start-ups and the challenges they face. A study by Crowne [18] investigates the critical product development issues that can lead to the failure of software start-ups. He presents an evolution model mapping and discussing underlying issues in three phases (start-up, stabilization, growth) of company maturation. The model highlights that the start-up phase presents several issues. They include 1) lack of experience by developers to meet deadlines and customer requirements, 2) unreliable product, 3) product not being a product thus requiring customization for each client, 4) lack of clear ownership of the product, for example, 5) no apparent authority to decide on additions of new features, 6) lack of a known plan for product development, 7) lack of understanding, discussion, and management of product development technologies. The issues raised in this start phase relate to the focus of early-stage software start-ups on the product. The early-stage software start-ups are prone to be influenced by the product, processes, resources, and capabilities that need the products to be delivered. The issues raised in the stabilization phase include 1) the founder's inability to let go in as much as they may have new shareholders and executives, 2) the development team failing to gel when there is new recruitment, 3) product becoming more unreliable due to more complex defects due to continued reporting from the customers, 4) raised expectations leading to customers to ask for new features or new platforms, 5) requirements becoming unmanageable, and 6) service provision to existing users delaying development activities. The growth phase brings in delays caused by the shortage of new skills, delay in development due to platform creep, empty product pipeline (the company cannot meet the demand for information on future product development), and lack of new product introduction processes.

The challenges that start-ups face are highlighted by Giardino et al.[12] as reliance on a single innovation and product, lack of resources, comprise small and usually inexperienced teams, are dependent (not self-sustaining), operate under time pressure, high risk, and uncertainty. Their study points out that the main challenges start-ups face are 1) thriving in technology uncertainty and 2) acquiring the first paying customers. A mapping study by Paternoster et al. [8] takes a more human-centered view of failure in start-ups. It unveils a behavioral framework that shows how inconsistency in managerial strategies and execution can lead to failure of software start-ups. There is thus a need to understand the existing practices used by start-ups in other ecosystems because start-ups execute different business and technical practices whose results can lead to success or failure. These strategies are likely to be influenced internally and externally by ecosystems. Understanding the practices can help start-ups to benchmark managerial and operational best practices as they consider related constraints in ecosystems like markets and policies.

Klotins et al. [34] highlight the need to identify inadequacies in applied engineering practices that significantly contribute to the high failure rates of software start-ups. They categorize these practices into knowledge areas and map them to 11 of the 15 areas of the Software Engineering Body of Knowledge (SWEBOK) [35]. Another study by Salas [36] examines lean start-up practices and their application within Finland's software companies. It looks at the positive and negative effects of these practices and makes recommendations for their application in software companies.

2.3 Software Start-up Patterns and Practices

The pattern paradigm has been borrowed from other domains [37] and active areas of software engineering [38]. Patterns apply to many aspects of the software start-ups domain. Some patterns describe the challenges of starting up a software start-up [39]. These patterns focus on start-up dynamics such as product development and the team developing it, patterns for the recruitment process of talented start-up-minded people to ensure their happiness and productivity [40]. Some patterns address start-ups as they develop software products and information services under rapidly changing social web conditions where potential customers can quickly adopt new behavior and good practices to help the start-up succeed and grow [41].

Some patterns focus on processes within start-ups, for example, software development [42]. There are also proposed patterns for strategies for building innovative software start-ups [24] and patterns identified to analyze the various software start-up ecosystem phases [14]. However, there is limited research on the transfer of patterns to start-ups in the developing world where geographical, political, social, and economic contexts vary. The contextual variation may affect the orchestration of the patterns in the emerging ecosystem. Understanding how the start-ups operate in emerging ecosystems is an essential step in the mapping and benchmarking the practices from successful start-ups in developed ecosystems. Existing studies on patterns concentrate on software start-ups and barely look at hubs practices in as much as a hubs nurture a considerable number of software start-ups and influence their early stages.

2.4 Measurement in Software Start-ups

Start-ups as young software-based businesses adopt software engineering practices that are ad hoc, agile, and to some extent, lean. Measurement as one of the vital engineering practices and is widely adopted in various mature engineering domains and large software development organisations. In software engineering, measurement and metrics are primarily used to measure software products, processes, and projects. Tahir, Rasool, Noman [43] assert that the role of software measurement as *essential to characterize, evaluate, predict and improve software products, processes, and resources.* Large software-based companies formalize measurement into measurement programs (MPs). These programs are systematic efforts carried to collect, analyze, and visualize metrics in companies. One study by Staron and Meding [1] proposes a model for assessing the robustness of these programs in large companies. A Systematic Mapping Study (SMS) by Tahir et al. [43] highlights challenges that Small and Medium Enterprises (SMEs) face in trying to use software measurements and challenges that hinder the adoption of formal MPs in SMEs. The challenges faced by the SMEs include 1) low maturity for measurement, 2) poor measurement knowledge 3) time pressures to market their businesses 4) lack of professionals experienced in measurement 5) lack of automated tools to support MPs, and 7) data collection problems that relate to unavailability of assessment data for measurement tools. Other studies have also shown SMEs to emphasize the measurement performance-related metrics. However, some studies propose the development of lightweight measurement programs, especially amongst SMEs.

Specific studies also highlight reasons for the limited use of measurements in SMEs. For example, one study highlights that SMEs use different sets of metrics [44]. A study by Claudia, Valtierra, and Mirna [45] about software process improvement (SPIs) highlights the issues of small teams, lack of infrastructure, low level of measurement maturity, high workloads as impediments to adoption of measurements in SMEs. Another study by Bhatti[46]

highlights the lack of metric selection methods. A study by [47] cites limited planning for measurements as another factor for limited use of metrics. To a certain extent, software start-ups have similar challenges to SMEs and are likely to face related challenges in adopting MPs. Studies indicate that software start-ups use various metrics [48]. However, no available studies focus on the challenges that start-ups face to characterize metrics and discover why the software start-ups chose and use the metrics compared to mature software companies. Understanding why start-ups chose to use specific metrics is the first step in rationalizing metrics decision.

An earlier Systematic Literature Review (SLR) by Gomez et al.[49] analyzed 78 primary studies investigating what, how, and when to measure. The study established that most studies discussed product metrics (78%) with relatively few looking at project and process metrics. The most measured product metrics were software size and software complexity. The mapping of the metrics to the software project lifecycle showed the initial phase and the intermediate phases used the most metrics with considerably few metrics in the final phase of the lifecycle. The study asserts the need for theoretical and empirical validation for metrics before applying them in the measurement process. Given the focus of early-stage software start-ups on developing a Minimal Viable Product (MVP), the use of product metrics would not be surprising. However, a study by [48] indicated that start-ups use various metrics, including business metrics. The general lack of focus on the same type and set metrics, even the known ones, could point to contextual influences. These influences may manifest themselves in patterns of practices [10], further pointing to potential ecosystem influences.

In the study presented in Chapter 5, we review related literature on measurements and how it relates to the broad area of software engineering. Earlier literature [50], [51] highlights that software engineering measurement focuses mainly on measurements for software products and measurements aimed at monitoring and improving the software development processes. For example, Lavazza [52] proposes an explicit model for the Goal Question Metric (GQM) plans for measured processes and guides the GQM process precisely to define the metrics involved.

A study by [1] highlights and characterizes measurement programs used by large software firms. These programs are possibly motivated by the scale and variation of metrics in large software companies. These programs are characterized by planning, coordinated execution, and control of the varied measurements. Related studies have shown software start-ups to measure in as much as they mainly do not structure their measurements into formal programs.

A study by Pino et al.[53] asserts that measurement models and standards used in large software organizations require considerable time, resources in addition to their complex recommendations. Their study recommends the implementation of separate measurement processes for SMEs. However, we still find measurement models like PRISMS[54], SQIP[55] that adopt standards and are used in SME measurement programs. Tahir et al. [56] in their systematic mapping study, observe that majority of measurement models focus on software process improvement (SPI). For example, PRISMS [54] model relates software improvement goals to business goals to prioritize key process areas for improvement. These SME models generally extend the Goal/ Question/ Metric (GQM) model that identifies measurement goals in a company, formulate questions, and the required metrics to measure the questions [57]. The most popular of these models is MIS-PyME [58]. That presents a methodological guide to defining software measurement programs based on indicators that support process improvement goals for small companies and another derived model called GQM-DFSMS [59].

The focus on Software Process Improvement (SPI) for most measurement models in start-ups means metrics like product, project, and other business metrics that are not influenced directly by technical processes are likely to be ignored. However, in as much as software start-ups use known measurements, to the best of our knowledge, there is a general lack of empirical studies that characterize the metrics they use. This characterization would influence the software start-ups on deciding what metrics to use, when to use them, and the resources required to use them.

2.5 Start-up Ecosystems

Cukier et al.[17] define a start-up ecosystem as a *limited region within 30 miles (or one hour travel) range, formed by people, their start-ups, and various types of supporting organizations, interacting as a complex system to create new start-up companies and evolve the existing ones*. Using the New York start-up ecosystem as a case study, their study highlights the stages of evolution of ecosystems that include nascent/emergent, evolving, mature, and self-sustainable. Their work points out the requirements for each of these phases and highlights the factors that may influence an ecosystem to evolve to reach a self-sustainable maturity level. Cukier et al. [17] also propose the need to measure ecosystem connectivity using social network data.

Another study by Voss et al. [60] aimed at gaining a broad understanding of the entire structure of the German innovation system highlights where improvements are necessary and proposes solutions where possible. Another study in India [16] focused on innovation as a subject in entrepreneurship [61]. Kon et al. [62] present a generalized conceptual framework that outlines the key forces leading to a successful digital entrepreneurship environment after deriving a conceptual framework using the Israeli software start-up ecosystem. Most of these studies in the different ecosystems broadly look at start-ups with an entrepreneurship lens and generally lack focus on software start-ups.

2.6 Ecosystem Models

Software start-ups as businesses usually are built on innovative solutions that address problems in geographical, domain, or technical and business contexts. Start-ups in all stages of development are influenced internally and externally by their operational ecosystem. Other actors also influence start-ups as key actors in entrepreneurial ecosystems. For example, universities provide the required affordable skills needed by early-stage software start-ups in and outside hubs. Policies influence the angel and venture capital culture and the markets in which start-ups operate. The extent of influence of various start-up actors on each other can vary but this influence generally characterizes a particular software start-up ecosystem. Understanding the start-up ecosystem dynamics is not trivial and requires evidence-based research.

Some studies propose models and frameworks to help researchers and practitioners assess and explain various aspects of ecosystems. Safiullin, Fatkhiev, Grigorian [63], present the Triple helix model, a qualitative model of strategic innovation networks that embody universities, industry, and government as key players in the initiation, creation, and dissemination of new knowledge. This model presents the interaction of the individual institutions at every stage of innovative product development. It highlights the government and university interaction in the initial stage of innovation to conceptualize the idea. The university then cooperates with industry in the transfer of the technology, and the final product is

commercialized in the market by a joint effort of government, and industry [63]. The model can be used to analyze the software start-ups from universities taking them as innovations or those innovations that a initiated through government research grants. Using the model, the start-ups in the early stages of innovation can then leverage the technical expertise and skills from the universities. The skills and expertise includes mentoring from faculty, programming skills from student interns, incubation of the innovations in university-based hubs, and commercialization of the innovations from private accelerators. The accelerators also validate the solutions in the market and finally build them as start-up businesses.

Cukier and Kon [25] present a start-up ecosystem maturity model for understanding the evolution and dynamics of software start-up ecosystems. The model based on systematic qualitative research in a multiple case study conducted in three ecosystems (Tel-Aviv, São Paulo and New York). It shows that start-up ecosystems can evolve, passing through a sequence of four levels of maturity that include 1) nascent, 2) evolving, 3) mature, 4) self-sustainable. This model is important in this study since we map the development of the East Africa software start-up ecosystems as nascent/emerging. The mapping of the East Africa ecosystem as emerging differentiates the work presented in this empirical study from software start-up studies done in developed ecosystems. Cukier and Kon [25] characterize an emerging ecosystem as having existing start-ups, few investment deals, optional government initiatives to support the ecosystem's development, and no relevant output in terms of jobs or worldwide penetration. We thus similarly characterize the East Africa ecosystem.

A case study by Stam [64] of 12 cities in the Netherlands developed a model to measure entrepreneurial ecosystem elements and compose an entrepreneurial index for the ecosystem. This model bases on ten conditions categorized as framework conditions and systemic conditions that include formal institutions, entrepreneurship culture, physical infrastructure, demand, networks, leadership, talent, finance, new knowledge, and intermediate services. They measure the output of entrepreneurial ecosystems with different indicators of high-growth firms. This model highlights the importance of the supply of support services by various intermediaries. It can substantially lower entry barriers for new entrepreneurial projects, reduce the time to market innovations, and enhance innovations' diffusion as input to the next generations of innovation. Isenberg [65] intimates that *there is no exact formula for creating an entrepreneurial economy; there are only practical, if imperfect, road maps.* He also asserts that instead of aiming to imitate thriving ecosystems, each region should identify and develop its qualities [65].

2.7 Software Start-up Hubs

Hubs are vital stakeholders in several start-up ecosystems in addition to the entrepreneurs, enabling policies and investors. We can map hubs to business service firms in the model presented by [64]. Hubs usually play a nurturing role in software start-ups in their early stages. They provide incubation, acceleration, and co-working space to start-ups as standalone services or a combination of these services. As De Beer et al.[66] asserts that *overall, hubs exist primarily to enable and support entrepreneurship and innovation, more than to create or implement them.* The hubs provide access to funds, networking opportunities, resources, and training to grow successful start-ups. Start-ups face various known challenges [9, 18, 29, 67]. These challenges inevitably lead to a high failure rate for start-ups in the first two years of their operation a cited 90% failure-rate in the study by [4]. As ecosystem actors, hubs play a crucial role in addressing some of these technical and business challenges. For example, accelerators address the start-up funding gap [68]. The hubs operate as

private businesses or public sector organizations found in universities or technology parks.

Hubs are prevalent in developed ecosystems, and a number are emerging in developing economies. The majority of the existing literature presents hubs as nurturing spaces for start-ups businesses in general. We have not encountered studies in the emerging East Africa ecosystem that study the hubs focusing on software start-ups. A case study of the Thailand ecosystem by Munkongsajurat [69] analyses the necessary business incubation activities that support start-ups and SMEs in developing ecosystems. They propose a possible operational model for business that defines 1) a host organization, 2) mode of operation, 3) the primary source of funding, 4) other sources of funding, 5), and the incubation strategy. This model generally applies to all businesses with little focus on software start-up businesses that develop software services and generic business activities.

Another study in Kenya by Chirchietti [20] examines the extent to which support provided by hubs addresses the challenges of start-ups. The work highlights that the start-ups in Kenya face diverse challenges. However, hub managers are not aware of these challenges; hence the hubs provide various services to the start-ups. The variation in services may lead managers to make assumptions about the resources, operations, and relationships that the software start-ups need to survive.

A study by De Beer et al. [66] proposes a framework to assess technology hubs in Africa systematically. This framework identifies three archetypes for African technology hubs: 1) cluster, 2) company, and 3) country. The framework uses Kenya as a case study to discuss potential collaborations, competition, and conflicts amongst the archetypes. Most of the existing studies about software hubs in East Africa are still in grey literature with limited empirical studies. To the best of our knowledge, no empirical study investigates software start-ups working in hubs as a phenomenon in East Africa as an emerging start-up ecosystem. We argue that hubs are essential ecosystem actors and influence early-stage software start-ups by providing incentives and resources. The emerging East African ecosystem has seen a proliferation of hubs [20]. Studies have investigated hubs as general nurturing spaces for business. However, there is a need to understand how the hubs operate to nurture software start-ups by investigating their processes, resources, and incentives. The hub operations may vary, but the main aim of nurturing and growing software start-ups cannot be understated. The study presented in Chapter 5 investigates hubs' operations concerning software start-ups in the emerging East Africa ecosystem. Understand how hubs work with software start-ups will be the first step to inform practices that can improve early-stage software start-ups' nurturing.

2.8 Policies and Enabling Support for Software Start-up Ecosystems in East Africa

The East African software start-up ecosystems can be largely classified as emerging with the Kenyan ecosystem being the earliest and most active and currently attracting considerable foreign investment compared to Uganda and Tanzania. The growth of the Kenyan ecosystem has been partly supported by early government interventions in terms of enabling policies for innovations and software start-ups. Kenya, too, has many ecosystem support organisations that include incubators, accelerators and co-working spaces. The emerging software ecosystem support policies in East Africa have been fueled by the need to create employment for the largely unemployed youths in East Africa in a bid to reach the specific National Development Goals (NDGs). The existing enabling government initiatives in Kenya include Konza Technology City [70] under development in the proximity of Nairobi, the Savannah Fund [71]. The Global Sustainable Development Goals (SDGs)

2015 as set out by UNDP have been adopted by Uganda and the country has set up a Roadmap for the implementation of the SDGs in Uganda 2020/21 – 2024/25. To actualise goal 9 (Industry Innovation and infrastructure) of the SDGs, Uganda has specifically set up the Ministry of ICT and National Guidance, to spur the growth of new industries and information and communication technologies and enable the country to frog-jump the economy into the Fourth Industrial Revolution (4IR). Initially, key bodies like National Information Technology Authority (NITA) was set-up to focus the country and government agencies in adopting and using ICTs and also regulate the industry and later the Ministry of ICT and National Guidance was created for oversight. The adoption and use of ICTs in the public and private sector is important in growing the start-up ecosystem in developing economies as the UNDP highlights that *"more than 4 billion people still do not have access to the Internet; 90 percent of them are in the developing world"*. Other programs like National ICT Initiatives Support Program (NISP) whose mission is to *"champion the creation of the ICT Innovation ecosystem to facilitate the growth and development of the software applications and innovations industry in Uganda"* were set up in 2018 by the Ministry to galvanise ICT innovations and commercialisation. However, the success of the polices and support organisations for the emerging East Africa ecosystems can only be realised through the growth of successful start-ups that can create jobs and sustainably add to the national gross domestic products (GDPs). The success of the start-ups hinges on the unified understanding of how the software start-ups operate and proposal of avenues to improve their current operations internally and within the ecosystems.

2.9 Theories in Start-up Research

Research about software start-ups as technology-based or focused businesses cannot be carried out in isolation. Studying the start-ups and their environments like hubs will potentially be guided by theories from diverse fields like software engineering, computer science, information systems, business, and economics. The importance of theories in many scientific fields cannot be understated. Theories form a foundation to better understand the start-up phenomenon and answer the 'why' questions that explain the phenomenon and its related events. In this section, we mention the related theoretical lenses that may be used to examine this study. This study primarily focus on the technological aspect and, to a lesser degree, the business aspect of start-ups and hubs in relation to the start-ups.

De Beer et al. [66] applied the cluster theory (adapted from the fields of economic geography and strategic management) in their framework for assessing African technology hubs. This theory posits *that advantages exist in the geographic concentration of enterprises that compete and cooperate in a specific sector*. Participation in a cluster benefits the participating members of a regional network and affords opportunities for coordination, mutual improvement, and productivity growth. This theory has been used to explain the relative successes and failures of high technology clusters in California's Silicon Valley and the Greater-Boston corridor known as Route 128 [72]. Cluster theory presents concepts that can be used to assess and understand the evolution of the East Africa start-up ecosystems as clusters in terms of software start-ups and hubs. For example, the adoption of the theory in work by Oyelaran-Oyeyinka and McCormick [73] aimed to improve the understanding of how to transform local clusters in Africa into local systems of innovation and how to better connect local to global actors. The aggregation of software start-ups, mature software companies, and hubs in capital cities like Kampala and Nairobi characterize the emerging East Africa ecosystems. These cities have established internet infrastructure, skilled talent, entrepreneurs, investors, markets, and technology literate population. To some extent, the start-ups and hubs in these cities compete for resources and the comparatively few funding

opportunities, skilled talent, and small local markets.

Nguyen-Duc, Seppänen, and Abrahamsson [74] present a conceptual model that is based on the Cynefin model and Complexity theory. The model captures how innovation and engineering activities occur and evolve in software start-ups over time. The focus of early-stage software start-ups on problem/solution fitting (product development) can call for underlying start-up research to use the Complexity theory as a basis for analyzing software design and development as also used by [75, 8].

Software projects can also be seen as a complex phenomenon that involves designers, developers, users, stakeholders, and reactions to user inputs. Steinert and Leifer [76] present the Hunter-Gatherer model that can explain how designers pursue endeavors to search for the best solution within often ambiguous problem spaces. The Hunter-Gatherer model is premised on human history that the earth was initially occupied by hunters and gatherers who killed wild animals and gathered wild plants to survive. This model helps to explain the origin of transformative or radical innovations, accounting for how design engineers skillfully find their solutions by analyzing often ambiguous problem spaces (hunting) and conceptualizing and executing them (gathering).

Decision making is a crucial aspect of software start-ups as decisions, whether informed by facts or ad hoc, must be made almost daily by start-up founders. Large companies use metrics [1], and tools like dashboards [27] to inform their decisions by applying formal measurement programs. Start-ups, on the other hand, rarely have formal measurement frameworks [48] but still make decisions that at times are used in changing strategy and growth direction (pivoting). To analyze decision making in different situations, Software start-ups can apply a decision-sensing framework named Cynefin proposed by Snowden and Boone [77]. This framework aims to help executives sense their context to make better decisions and *"also avoid the problems that arise when their preferred management style causes them to make mistakes"*. The Cynefin framework divides the world into the ordered domain (known domain or knowable after analysis) and unordered domain (a complex domain that involves situations where the cause-effect relationship can only be perceived retrospectively). The two contexts are divided into five sub-contexts. Four of these sub-contexts include simple, complicated, complex, and chaotic. These require leaders to diagnose situations and to act in contextually appropriate ways. The fifth sub-context called *disorder* is applied when there is little clarity about which of the four sub-contexts is dominant [77]. Given the short operating history of software start-ups, they are more likely to make decisions in the unordered domain.

Unterkalmsteiner et al. [31] propose using the Effectuation theory in start-ups to help better make sense of decision-making processes entrepreneurs use during uncertain situations in the growth phase of the software start-ups. They propose using this theory in situations that include problem validation, the definition of the value proposition, designing MVPs, and pivoting (radically changing some fundamental aspects of a product [78]). This theory embodies six principles 1) the bird in hand principle, 2) lemonade principle, 3) the bird-in-hand principle, 4) crazy quilt principle, 5) pilot in the plane principle, and 6) affordable loss principle [79].

Start-ups do not operate in isolation, and as businesses, markets, policies influence technology trends and global investment climate. These multiple influences are seen in the challenges that the software start-ups face. The influences are well documented to lead to start-ups' failure in their early stages of development. Boundary spanning is a concept that examines organizations that transition from a rigid hierarchical structure towards a network-based expert organization, giving rise to informal boundaries rather than structural ones [80]—boundary spanners as entities or people who bridge these boundaries

and opportunities. Software start-up founders, technical and business processes can be seen as such. Unterkalmsteiner et al. [31] in the start-up research agenda intimate that boundary-spanning helps start-up founders *in discovering how to overcome the challenges of distributed global work, where motivations, work styles, and knowledge domains vary across boundaries.* Therefore, this theory can help understand the dynamics and working of software start-ups concerning other actors like investors, universities, markets, hubs in East Africa, and other extant nascent ecosystems.

2.10 Specific Related Work to Each study

2.10.1 Patterns and practices in software start-ups

Research on software start-ups: There has been considerable research in software start-ups relating to their success [33], failure [12, 18] and challenges [9]. Paternoster et al. [8] summarize in their mapping study a behavioral framework that shows how inconsistency in managerial strategies and execution can lead to failure [12]. Klotins et al. [34] highlight the need to identify inadequacies in applied engineering practices that significantly contribute to the high failure rates of software start-ups. They categorize these practices into knowledge areas and map them to 11 of the 15 areas of the Software Engineering Body of Knowledge (SWEBOK) [35]. Salas [36] examines Lean start-up practices and how they apply within software companies in Finland. They look at the positive and negative effects of these practices and make recommendations for their software companies' applications.

Start-up patterns: The pattern paradigm has been borrowed from other domains [37] and active areas of software engineering [38], and applied in the software start-up domain. Some patterns describe the challenges of starting up a software start-up [39]; those focused on start-up dynamics such as the product being developed and the team developing it; patterns for the recruitment process of talented-start-up minded people to ensure their happiness and productivity [40]. Some patterns address start-ups as they develop software products and information services under conditions of rapidly changing social web, where potential customers can quickly adopt new behavior [81]. Others are patterns for acceptable practices that could help start-ups be successful and grow [41], patterns for processes within start-ups, for example, software development [42], patterns for strategies for building innovative software start-ups [24]. Patterns have also been identified to analyze the various software start-up ecosystem phases [14]. However, there is limited research on whether these patterns are transferable to start-ups in the developing world where geographical, political, social, and economic contextual variations may affect their orchestration.

There is considerable work done to discover software start-up patterns in various start-up ecosystems. However, most regional studies about start-ups are mainly targeting ecosystems rather than cross-cutting practices in software start-ups across regions. The pattern studies are localized, and no significant studies exist to investigate whether software start-up patterns known in western countries hold for Africa, let alone East Africa. Most studies focus on the developed world like the US and Europe (Germany, Israel, Finland, Switzerland). However, it is unclear to what extent they are generalized to the developing world.

2.10.2 Hubs operational models and characterisation frameworks

In this section, we review general existing operation models for hubs and frameworks for characterizing hubs.

Operating models for hubs: Munkongsajurat [69] in the study of the Thailand ecosystem

proposes a model for business incubators for start-ups and SMEs in developing ecosystems. This model identifies 1) host organization 2) mode of operation 3) the main source of funding, 4) other sources of funding, and 5) incubation strategy as key aspects of all business incubators. He also groups the possible activities for the model's operation as public awareness activities, pre-incubation activities, incubation activities, acceleration activities, and networking activities. This model and the related operations can generally represent the East African hubs we interviewed if we do not focus on software start-ups. However, our focus on software start-ups pinpoints the actual operations that software start-ups undergo in the nascent East Africa ecosystem.

Other models have been proposed to characterize the operations of incubators. Bergek and Norrman [82] presents a framework for identifying best practice incubator models and rigorous evaluation of the performance of incubators. It presents 'howtos' for selection, business support, and mediation strategies in incubators. The study by Chirchietti [20] examines the extent to which the hubs' support addresses the challenges of start-ups. Their work highlights that the start-ups in Kenya face diverse challenges, but hub managers are not aware of these challenges besides sharing the same services. This indicates that managers are likely to make assumptions about the resources, operations, and relationships the software start-ups need to survive. Thus, there is a need for a more focused study of the hubs' actual operation concerning software start-ups.

Hub taxonomies: Dempwolf et al., [68] in their study, characterize start-up assistance organizations highlighting the differentiating characteristics between incubator hubs and accelerator hubs. They also classify the different accelerator hubs and show the metrics for measuring accelerator hubs' success and the differentiating metrics for measuring success between different accelerator hubs. Their study points out that distinction between the different types of hubs blurs as the hubs are *designed to fit local conditions and meet local opportunities*. The hubs' characterization in East Africa confirms this assertion as the hubs offered combined services possibly driven by the contextual settings. The metrics captured by this[68] study are focused on measuring the performance of the hubs, whereas in this paper, we highlight the metrics the hubs use to track the progress of the start-ups they host.

Earlier attempts at classifying incubators include Kuratko, and LaFollette [83], which takes a funding perspective to classify incubators by identifying primary sponsorship sources public, private, university, and non-profit. Grimaldi and Grandi [84] take an ownership-related view that categorizes incubators as corporate, private, independent private, university incubators and business innovation center. De Beer et al. [66] propose a framework for assessing technology hubs in Africa using Kenya as a representative case. The framework uses a three-tiered system for categorizing African hubs into 1) cluster, 2) company, and 3) an entire country. This framework may be useful as a starting point to analyze hubs

Soetanto and Jack [85] in their work, construct a framework for understanding the networks of incubators by taking a perspective that looks at incubators as 1) networking activities in terms of the resource type and 2) defines networks of the incubator as external and internal. The results indicated that incubators developed more networks to access intangible resources than tangible resources. Also, highly innovative firms and medium to low innovative firms found differences in their networking activity.

2.10.3 Measurements in software start-ups

Related work for this research stems from two partially overlapping research directions. On the one hand, there is research on measurement systems and frameworks within the broader

software engineering domain. This research focuses on the measurements used in software start-ups and their perceived benefits. On the other hand, there is research on software start-ups, which is concerned with how software start-ups are operating and thriving. Our study is at the intersection of these two areas. In the following, we summarize relevant related work in both areas and the intersection.

Measurements in software development: Software measures or metrics are omnipresent in software development [86]. Companies adopt varieties of measures, which might happen in a systematic or ad-hoc way. Several studies have explored measurements in the broader domain of software engineering. For example, very early studies looked at technical aspects such as software size estimation and measurements [87] using software function, source lines of code [88], and software productivity measurements [89]. There is also recent work around software metrics [90] and mapping studies in aspects like product size measurement methods [91]. While these measures are in theory applicable to start-ups, it is not clear whether the start-ups use them. The term measurement program is often used to describe a company's systematic effort to collect, analyze, and visualize measures. Staron and Meding [1], in their work, assess the robustness of measurement programs in large software development organizations. They use the MeSRAM method, whose aim is, as they put it, is to *"support the companies to optimize the value obtained from the measurement programs and their cost"*[1]. The MeSRAM method uses an aspect called "metrics used" that organizes the metrics found in software companies using five sub-categories (business metrics, product metrics, design metrics, organizational performance metrics, and project metrics). Other frameworks and methods aim mostly at the assessment of measurement programs and organizational infrastructures, such as Comer and Chard's framework [92], MIS-PyME [58], and the method by Daskalantonakis et al. [93]. However, in contrast to these works, we focus on software start-ups and explore measurements used and their benefits.

Software start-ups: There has been considerable research on software start-ups relating to their success factors [33], failure factors [12, 18] and challenges [9]. Paternoster et al. [8] in their behavioral framework highlight how inconsistency in managerial strategies and execution can lead to failure [12]. Klotins et al. [34] carried out a literature review and identified gaps in practices that support start-ups in successfully transitioning through the start-up life cycle. However, these studies rarely focus on measurement and related practices.

Measurements in software start-ups: Some empirical studies have been done on measurement in start-ups, especially in operational performance measurement. For example, Rompho [94] in their study on performance measurements in start-ups highlight that *"there is a positive relationship between the perceived importance and the performance of each metric"*. However, they found no significant relationship between each the importance of each metric and performance among the various stages of start-ups. Other studies focus on specific measurement approaches. For example, Paranjape et al. [95] conducted a secondary study to evaluate the Balanced Scorecard as a known performance measurement system in business and further examine problems associated with designing and implementing performance measures. They found that the Balanced Scorecard method still a dominant performance measurement system but its successful implementation within businesses is less prevalent. However, these studies focus on start-ups from a business perspective and do not consider software start-ups, consequently ignoring software measurements. A rare example of literature focusing on measurement in software start-ups is the work of Croll, and Yoskovitz [2]. In their work, Croll and Yoskovitz categorize start-ups into six types that include software as a service, e-commerce, free mobile app, media site, user-generated content, and two-sided marketplace. They also identify fine-grained metrics that are to be tracked by the lean start-ups.

Most of the empirical and secondary studies around measurements in start-ups are still not focused on software start-ups. Similarly, much of the research on software measurements happen in mature companies. Thus, little is known today about software measurements used in software start-ups. The few works existing focus on the developed world only, ignoring regions such as East Africa.

2.10.4 Characterising measurements

This section reviews the related studies about measurement in software engineering and state-of-the-art measurements in software companies and characterizing metrics.

Measurement in software engineering: Measurement is an activity done in many mature engineering fields. It is becoming more important in software engineering as the demand for software-intensive products increases. IEEE [96] defines measurement as an *"act or process of assigning a number or category to an entity to describe an attribute of that entity. A figure, extent, or amount obtained by measuring"*. Cook and Wolf [97] describe measurement in software engineering as a successive process for defining, collecting, and analyzing data in the software development process to understand and control them. Staron and Meding define measurement as the process of assigning a *"value to a variable -a metric- in a structured, repeatable and deterministic manner"*[1]. Kemell et al. describe metrics as *"quantifiable measurements of a phenomenon or object"* [98]. There are varied definitions of software engineering measurement. But overall, most emphasize process, repeatability, and quantification aspects arising from the process. A Systematic Literature Review (SLR) of 78 studies by Gómez et al. [49], on measurements used in software engineering about what, how, and when to measure, established that most companies (79%) focused on product, software complexity, and software size metrics. The review highlighted the little focus of metrics on the project (12%) and the software process (9%). The study did not specifically address the measurements and methods used in software start-ups. The product focusing metrics include software function points as a measure of software size like source lines of code [88], and metrics for productivity [89], coupling, and cohesion [99].

Measurements in software companies: There are existing studies on performance measurement in start-ups, especially in Small to Medium Enterprises (SMEs), that examine business-related aspects. For example, Rompho [94] investigated the use of performance measurements in start-ups. The study revealed no significant relationship between the importance and performance of metrics in different stages of start-ups. Another related study [100] proposes a framework to assesses performance measurement systems (PMS) in Small to Medium Enterprises. The framework characterizes and classifies SMEs, which are, at times, called small businesses. In the study, requirements for performance management systems are obtained and validated through a tool's development. Blank's [6] distinguishes a start-up from a small business by highlighting the differing growth ambition and asserting that SMEs do not necessarily intend to grow. Other work-related to performance includes Sharma et al. [44] that identified general features for performance measurement systems (PMS), tailored them to SMEs, and developed a framework to support SMEs critically reviewing their PMS. A study by Croll and Yoskovitz [2] focused on measurements in software start-ups. It presents an analytics approach that uses metrics to measure progress in start-ups. Its main goal is to help a start-up build the right product and market it before funds run out. The study identifies essential progress metrics and how they are analyzed and reported.

Measurement in large software companies is set up mainly as part of well-planned measurement programs (MPs). Measurement programs aim to understand, evaluate, improve, and predict the software processes, products, and resources [43]. An SLR of measurement

programs by Tahir et al. [43], highlights the planning models and tools for implementing MPs, their success/failure factors, and mitigation strategies. Staron and Meding [1] present the MeSRAM method used to assess the robustness of MPs in large software development organizations to support companies, optimize value from the MPs and their cost. However, measurement programs are not prevalent in software start-ups, potentially highlighting the ad hoc nature of measurement and metrics use.

In our earlier study [48], we used Cukier et al. [17] mapping to characterize the East Africa start-up ecosystem as emerging. An emerging ecosystem is characterized by existing start-ups, few investment deals, optional government initiatives to support the ecosystem's development, and no relevant output in terms of jobs or worldwide penetration. We identified metrics used in software start-ups in the ecosystem, the perceived benefits of using the metrics. We compared the metrics to those used in lean start-ups in developed ecosystems. Results showed that start-ups in the East African ecosystem used several business metrics and product-oriented metrics, while we found no evidence of design-oriented metrics.

Assessment methods for metrics in software engineering mainly take process, product, and project perspectives. A Systematic Literature Review (SLR) by [101] highlights the most measured attributes of processes to include quality, cost, and schedule. However, characterizing metrics using software engineering perspectives like process, product, and the project is still insufficient to understand why metrics are chosen and used in software start-ups and mature software companies. Studies by [29][14] [9] highlight the challenges faced by software start-ups. These include lack of access to initial funding, difficulty acquiring paying customers, ever-changing technology, dynamic markets, technical debt, poor infrastructure, and time pressures to deliver Minimal Viable Products (MVP), among others. We thus cannot assume that start-ups chose and use metrics the same way as mature software companies.

Metrics characterisation: Measurement theory examines the validity of metrics concerning the software attributes that evaluate metrics using axioms on the properties of their measurement scales [102]. A study by Wang [103] presents an axiom-based measurement theory and offers a generic measurement methodology. The study defines the fundamental ideas for engineering measurement that include an object of measurement as *"a physical entity or an abstract artifact that can be characterized or defined by a set of attributes"*. The study further defines an attribute as a fundamental characteristic or property directly or indirectly measurable and clarifies meta attribute and complex attribute as types of attributes. There are further definitions for the measurement scale and measurement unit to complete the axiom-based principles.

This axiom-based measurement theory by Wang [103] can aptly represent software companies. For this study, the measurement objects are software start-ups or mature software companies. We refer to their measured attributes (meta or complex) as metrics and definitions of their metrics' scales as dimensions and the units of measure on the scales as choices. The units of measurement for the dimensions scales are categorical. We used categorical units because we could not get absolute quantities for the units on the dimensions scale due to different types of metrics used in the companies.

According to [104], the ISO 15939 Measurement Information Model consists of two parts: classical metrology and the other on a non-metrology perspective. The metrology perspective presents a framework to identify, quantify, and collect base measures for an entity using a measurement method. The base measures are assembled using a measurement function into a derived measure. The non-metrology perspective deals with measurement results described as an indicator, analysis, and decision criteria.

An earlier study by Daskalantonakis [26] highlights characteristics of useful software metrics

that must be 1) simple to understand and define 2) as objective as possible, 3) cost-effective, and 3) informative. The study classifies metrics as a process, product, and project metrics. It also maps the metrics to target users, including software users, senior managers, software managers, software engineers, software process engineers, and software quality assurance users. The metrics classification proposed by Daskalantonakis [26] can be used to classify product and process aspects of software start-ups given the start-ups focus on product, process, and project-related aspects in early-stages. However, software start-ups also use organizational and business metrics. These metrics are also used in large software development organizations, as highlighted by Staron and Meding. They categorize the metrics as product metrics, project metrics, design metrics, business metrics, and organizational performance metrics [1]. Their categorization primarily represents the technical, organizational, and business aspects of large software development companies and can define software start-up metrics.

The axiom-based measurement theory and the ISO 15939, both base on metrology as they define object/entity of measurement, attribute/property to be measured, measurement scales, and measurement units. They both have a metrology-based aspect and a metrics application aspect that apply to developing metrology-defined measurement-based models. However, the measurement theory and standard definition of attributes do not consider potential contextual influences that may affect choice and use of attributes/properties (metrics in this study).

The existing characterization of metrics in software engineering focusing on product, process, project may be insufficient for the wide range of metrics that include technical, organizational, and business aspects in software companies. Staron and Meding [1] categories for metrics are sufficient, but they do not characterize the metrics to reveal why they are chosen and used in software start-ups and mature companies. We argue that challenges can potentially influence metric choice and use in start-ups. We believe that uncovering characteristics that relate metrics to challenges in start-ups will be a first step in understanding what affects the selection and use of the metrics in software start-ups and mature software companies.

2.10.5 Measurement dashboards, frameworks and tools

In this section, we present related studies to dashboards and software measurement practices.

Dashboards: Dashboards have been widely researched with focus on various aspects of business, such as measuring marketing effectiveness [105], performance measurement and monitoring [106], design and deployment[107, 108]. Staron et al. use the term dashboard to refer to a measurement system visualizing indicators complemented with base and derived measures. Their study highlights how dashboards are used to monitor the quality of software products under development by collecting indicators and measures, visualizing and disseminating them to show the status of these indicators in large software development organizations [27]. Dashboards are also used in large software organizations to monitor performance by managers. Velcu-Laitinen and Yigitbasioglu [3] describe dashboards as diagnostic tools meant to give busy managers a glimpse of company performance. Their study indicated that dashboards were perceived as effective tools for performance management, monitoring, problem-solving, decision rationalization, and communication.

However, no empirical studies report the use of dashboards and their perceived benefits in early-stage software start-ups in the nascent East African ecosystem. Software start-ups are characterized by dynamic business and technical processes, high failure rates, and face

widely known challenges [29]. The reason for start-up failures has also been documented [12, 8]. A study of start-ups in East Africa indicated they the measure [48], yet few are using measurement tools like dashboards that are prevalent in mature businesses and large software firms. Some of the start-up practices [10] and measurements [48] in East Africa hint to context influences when compared with practices in the start-ups in developed ecosystems. Few start-ups in East Africa use common dashboard tools like Google Analytics, Facebook analytics, and Asana. We hypothesize that the low penetration of dashboards may indicate that they do not serve East African start-up needs, although several start-ups use measurements. Moreover, existing dashboards may also suit some East African software start-up types more than the others, thus the general reluctance to use them.

Measurement frameworks, methodologies, and standards: Tyrväinen et al. [109] in their work proposed and evaluated a cycle-time-based metric framework. The framework is used during the feature development process and provides faster feedback from customers, reducing turnaround time in agile software development. The focus of this framework is optimizing efficiency in processes of feature development using metrics. This framework's implementation can benefit from this study by leveraging real-time dashboards to track the metrics' use. The Lean Startup Methodology (LSM) [5], based on the build-measure-learn paradigm, has been widely adopted by businesses and used in software start-ups. Its measure phase identifies various metrics that can be tracked during software development and other business phases. However, it is unknown to what extent software start-up firms use dashboards to integrate the different metrics from its measure phase to visualize and use the metrics to measure their progress.

The Goal Question Metric (GQM) approach focuses on quality and productivity goals for organization and project measurement. It defines questions that help in identifying the data needed to measure the goals during operations. It finally provides an interpretation framework (a set of objective and subjective measures) concerning the goals [110]. This approach can be used in defining appropriate measurements within start-ups. However, this paper focuses on the requirements for progress measuring dashboards and the perceived benefits of using the dashboards within start-ups. The requirements in start-ups may not be limited to quality and productivity but also business and technical activities. However, the GQM approach may be used in identifying the quality and productivity-related measurements in the start-ups. A dashboard will add more value in operationalizing the interpretation framework of GQM for the quality and productivity measurements in the start-ups. Kaplan and Norton introduced the Balanced Scorecard (BSC), performance measurement, and management framework for organizations that link organizational strategic goals and measures. It provides an overview of the performance of an organization in terms of financial, customer perspective, and internal business processes [111]. This framework has been adopted in many businesses, and software start-ups can adapt it for performance measurement and management. The start-ups' unique challenges have been widely documented and include the start-up lack of time and resources[29]. Thus, a dashboard to synthesize, display, and aid in interpreting performance and other metrics that matter to the start-ups may help them address these challenges.

There are other related measurement approaches that the dashboard could leverage to best capture the start-ups' requirements. These include Practical Software Measurement (PSM), whose focus is the information needs for project managers. It provides this through two components, the measurement process model and the measurement information model [112]. The dashboard for measuring progress in start-ups can leverage the measurement information model to present and visualize its outputs.

2.11 Research Gap

Start-ups as newly created firms that primarily develop innovative software products and services have been widely found to fail in their first 2 years of operations. The challenges they face are widely documented in developed ecosystems and are believed to lead to the high rates of failure. Emerging ecosystems as characterised by the ecosystem maturity model [25] that recognises ecosystems as a recognizable start-up hub, with already existing start-ups, a few investment deals, and no great output in terms of jobs or worldwide penetration may face even greater challenges in nurturing software start-ups and the practices and measurements used in software start-ups the emerging East Africa ecosystems in particular remain largely unknown. Knowing these practices could highlight how start-ups address challenges and, what and how they measure is important in understanding how the software start-ups and the hubs assess their practices to get insight and make meaningful decisions to increase their chances of survival. The ensuing chapters 4-8 present the specific research gaps addressed in each of the specific studies.

2.12 Summary

In this chapter, we reviewed the related literature to the overall study and literature specific to the five studies presented in this thesis. From re-known scholars in the field of software start-ups, we highlight varied definitions of start-ups that generally emphasize the technical and business aspects of start-ups like the product/service, organizational nature and challenges they face, their business models and their overall intention to grow/scale amidst the challenges. The literature we review highlights the known challenges facing start-ups and the practices and common practices relating to start-ups and software engineering in general. A considerable number of start-ups are generally nurtured in hubs and they operate in large dynamic systems called start-up ecosystems. We reviewed published studies about start-up ecosystems and existing models that have been used to characterise them. We reviewed known studies about at hubs and how they have been seen to nurture start-ups let alone software-startups. In the grand scheme of software engineering, we highlight existing theories that may be used to examine and explain practices, operations and measurements in software start-ups and in respect to ecosystems. From the literature, we hightlight the main research gap that we address in this overall study.

Chapter 3

General Methodology

This chapter presents the overall research design and the data collection and analysis methods used in the entire study. We also discuss the general approach we used to ensure data quality, the overall threats to the study's validity, and the challenges we faced in carrying out the study. This study explores practices and measurements in software start-ups and hubs in the emerging East Africa start-up ecosystems. We thus chose exploratory[113] empirical methods to approach to unveil the start-up phenomenon, given the limited knowledge of current practices and measurements in the start-ups and hubs in the region. This chapter presents the overall empirical research design that we followed in executing the entire study. The chapter also highlights our general rationale for the methodological choices for the compiled studies in this thesis.

3.1 Research Design/strategy

In this study, we mostly used exploratory empirical research methods that involved case studies and a survey. Overall, the studies in this thesis are guided by an interpretivist philosophy [113]. According to Robson et al.[114] classification of different purposes of research methodologies, the exploratory nature of a study serves the purposes of aiding a researcher in discovering what is happening, seeks new insight, and generates ideas and hypotheses for new research. The interpretivist philosophy is driven by truth as a social construct and highlights that there are alternative truths. The philosophy allows us to conduct an empirical study to construct an understanding and knowledge about the start-up phenomenon in a emerging ecosystems setting.

Yin [115] identifies a case study as an inquiry highlighting the overlap between the boundary and the phenomenon, while Robson [114] emphasizes the use of multiple sources of evidence in a case study. Also, Runeson and Höst highlight that a case study approach is *"well suited for many kinds of software engineering research, as the study objects are contemporary phenomena, which are hard to study in isolation"*[116]. Yin[115] also notes that case studies, unlike experiments and survey strategies, provide a non-restrictive boundary between the studied phenomenon and the study context. These different perspectives highlight a case study as an investigation of a phenomenon in a given context. Thus, we use case studies to offer a flexible design based on qualitative data and to provide a more in-depth and richer description [114].

However, the case study strategy has its critics who point out its proneness to bias by the researcher, lack of generalizability of such studies, and claim that it is of potentially less value. Flyvbjerg [117] argues that by applying proper research practices and taking

an interpretivist stand that considers knowledge to be more than statistical significance, case studies are still robust research approaches worth undertaking. Yin [118] concurs that *"qualitative research using semi-structured or in-depth interviews will not be able to be used to make statistical generalizations about the entire population where this is based on small and unrepresentative several cases"*. However, Bryman [119], and Yin [115], relate the generalizability of qualitative research or a case study to the significance of the research to its theoretical propositions. According to Saunders, Lewis, and Thornhill [113], it is up to the researcher using the case study strategy to establish this relationship to existing theory to demonstrate the broader significance of the case study findings.

We used appropriate qualitative tools for individual studies that included semi-structured interviews and a survey to collect the study data. Saunders, Lewis, and Thornhill [113] discuss the different types of interviews: 1) structured interviews - used to collect quantitatively analyzed data, 2) semi-structured and 3) in-depth interviews -used to collect data that is qualitatively analyzed. The used techniques are part of the case study research strategy. Saunders, Lewis, and Thornhill further assert that *"the data is likely to be used not only to reveal and understand the 'what' and the 'how' but also to place more emphasis on exploring the 'why'"*[113]. For interview questions, Runeson and Höst [120] propose a structure based on three general principles. The first is the funnel model that begins with open and moves to more specific questions. The second is the pyramid model that begins with more specific and ends with open questions. Lastly is the time-glass model that begins with open questions, gets more specific in between, and opens towards the end.

Saunders, Lewis, and Thornhill [113] discuss several data quality issues associated with qualitative semi-structured interviews. Firstly, reliability, which concerns whether other researchers would reveal the same information. Interviewer bias and respondent bias are key reliability issues in interviews emanating from semi-structured or unstructured interviews. For example, this arises when the respondent is unwilling to reveal some or all the information associated with an intrusive and potentially sensitive question. It can also arise when the participant reduces their willingness to participate in the study due to the time consuming nature of the interview.

Secondly, other issues that affect the quality of interviews include the validity and generalizability of the findings of the qualitative interviews. Saunders, Lewis, and Thornhill [113] explain that the validity of qualitative interviews *" refers to the extent to which the researcher gains access to their participants' knowledge and experience, and can infer a meaning that the participant intended from the language that was used by this person"*. Runeson and Höst[120] highlight that the validity of a study relates to its credibility concerning the trustworthiness of its results and not biased by the researcher's perspective. They further stress that *"validity must be addressed during all stages of a case study"*, but argue that validity cannot be evaluated until the analysis phase. For analysis, we used thematic coding for the qualitative data and hypothesis testing for the survey data.

Runeson and Höst [120] highlight that different sources of information are generally used in qualitative studies to help researchers to conclude from multiple sources of data, thus providing a broader perspective. Stake [121] highlights all the four different types of triangulation that include 1) a methodological triangulation (combining different types of data collection methods like quantitative and qualitative), 2) theory triangulation that uses alternative theories, 3) observer triangulation that uses more than one observer for a study and 4) data triangulation that uses more than one data source or collection of same data at different occasions.

This thesis explores practices and measurements in software start-ups and hubs in the emerging East Africa start-up Ecosystems. We achieve this through (1) characterizing

practices and patterns in start-ups and hubs, (2) characterizing the metrics in start-ups and hubs, and (3) collecting requirements and motivating the need for measurement dashboards in software start-ups

3.2 Research Approach

This research is qualitative and mainly uses exploratory methods. Figure 3.1 shows the research approach for this thesis. In the study presented in Chapter 4, we investigated seven software start-up cases in Uganda to ascertain the applicability of software start-ups patterns and practices found in the Swiss-Finish ecosystem (a sample developed start-up ecosystem) to the software start-ups in Uganda. We then conducted ten case studies of hubs in Uganda and Kenya in Chapter 5 to characterize operations and measurements in hubs. In the study presented in Chapter 6, we performed a multi-case study on 19 software start-ups in hubs in Uganda and Kenya to understand the use and perceived benefits of measurement in early stage start-ups. In Chapter 7, we iteratively derived ten dimensions and surveyed 19 software start-ups and 10 mature software companies in Uganda to characterize their metrics based on the dimensions. Chapter 8 presents and investigates 36 cases of early-stage software start-ups in Uganda and Kenya aiming to identify key requirements for measurement dashboards and the perceived benefits of using the measurements by the start-ups.

In the case studies, we mainly collected the raw data using semi-structured interviews administered face-to-face to the software start-up founders and hubs managers. In the study in Chapter 7, we used an online survey form to collect the data from start-ups and mature software companies in Uganda. In the interviews for the studies in Chapters 4, 5, 6, and 8, we used open-ended questions because they are flexible, thus allowing for the variation in the order and logic of questioning. The interview guides for the various case studies in this thesis are attached in Appendices A-D. The different interviews were recorded and transcribed verbatim. They were then coded and analyzed using thematic synthesis for study 4 and analyzed using content analysis for studies 5, 6 and 8. In study 7, we analyzed the survey using descriptive statistics and hypothesis testing. The instruments used for the various studies in this thesis included: 1) the interview guide to study the practices in software start-ups in Uganda shown in Appendix 10.6 2) the interview guide to investigating measurements in software start-ups and measurement dashboards in software start-ups in Kenya and Uganda shown in Appendix 10.6 3) the interview guide to investigating the operations of hubs in Kenya and Uganda in Appendix 10.6 and 4) the trial hard copy version in Appendix 10.6 that evolved into the online survey forms for software start-ups and mature software companies in Uganda. The paper form that we transformed into the online survey is shown in Appendix 10.6. In conducting and reporting the case studies, we used case study protocols. This thesis followed the case study guidelines described by Yin [115], Runeson, and Höst [120]. Yin describes a case study as a *" container for design decisions on the case study and field procedures for carrying it through"*. The description highlights that case study protocols are dynamic and need continuous updates during a study. The protocol is 1) used as a guide in data collection, keeping the researcher on track, 2) helps in concretizing the research planning that involves reviewing and refining the study questions and decisions on the data sources 3) can be used to solicit feedback from other researchers to lower the risk of missing to collect essential data and 4) act as a log for recording change decisions made during the study. In Chapter 5 and Chapter 6, we discuss the details of the case study protocols we followed in executing those studies.

Figure 3.1, outlines the overall methodology for the thesis

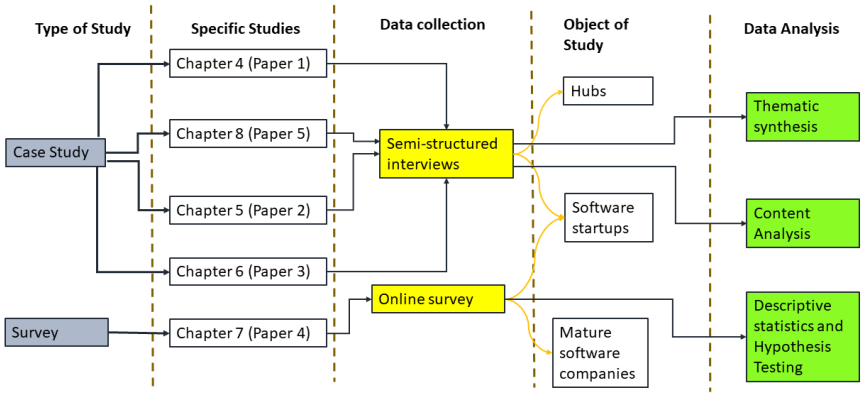

Figure 3.1. Research Methodology

3.3 Study Population and Sampling

This qualitative study used the non-probability[122], purposive sampling approach to select the case studies where we interviewed start-up founders and hub managers. Non-probability sampling is described by Etikan et al.[123] as a sampling technique. The selection process does not give all the participants in the population an equal chance of being included as randomization is not considered necessary. Creswell describes purposive sampling as homogeneous sampling where *"a researcher purposefully samples individuals or sites based on membership in a subgroup that has defining characteristics"* [124]. In this sampling, we intentionally selected early-stage software start-ups as the East African ecosystems can primarily be classified as emerging with more early-stage start-ups than mature ones. We, too, selected hubs hosting the early-stage software start-ups since the hubs host the early-stage start-ups and provide incentives. However, this sampling approach may exclude software start-ups not nurtured in hubs or those that left the hubs but still in the early stage of their development. The sampling, therefore, may have over-represented start-ups that joined the hubs in particular cohorts.

Uganda currently has six active hubs, and the number of early-stage start-ups hosted in these hubs varies in number, while some of the start-ups are not hosted in these hubs at all. Kenya has several active hubs, but we interviewed 4 of the five most active hubs and some start-ups due to logistical and scheduling constraints. We interviewed start-ups that meet the following criteria 1) early-stage software start-ups in hubs (2) early-stage start-ups not operating in hubs. We also interviewed active hubs that we have a working relationship with in Uganda and those active hubs and start-ups in Kenya that accepted our invitations to visit them.

We targeted early-stage software start-ups in and outside hubs and mature software companies in Uganda for the survey. In the survey, we considered a mature software company as one with considerable work history, a management structure, established software products/services with customers, technical employees, and an established physical office. This quantitative study used a non-probability quota sampling [122] because there is no available database listing for software start-ups and mature software companies in Uganda. We split the target population into two quotas, which included software start-ups and mature software companies. The software start-ups are presumed to be less experienced in using metrics due to their short operating history compared to mature software companies.

Due to known high failure rates, there are also more early-stage software start-ups than mature software companies. The quota sampling thus ensured a proportional number of respondents in each quota.

3.3.1 General challenges faced in the study

The interviews were deemed time-consuming by a majority of the interviewees in the start-ups and hubs. The founders and managers could have found the interviews time-consuming because they are generally busy and execute various roles. Given the early stages of the start-ups, some founders were blatantly unwilling to participate in interviews to protect their business ideas and young innovations. The founders' busy operations created issues in the scheduling of our interviews. Some key knowledgeable founders and managers declined our interviews altogether due to lack of time. The refusal potentially reduced the number of planned interviews in Kenya, where we had limited time to carry out interviews.

To interview the most knowledgeable founders and managers, we contacted the interviewees and explicitly told them about the interview's duration before scheduling the interviews. We proposed interviewing these busy managers during their convenience (even if this affected our study timelines). However, at times we failed to interview them in foreign countries due to limited stay times. This approach, however, helped in increasing the access to often busy but knowledgeable start-ups.

3.4 Data Collection and Analysis

The studies presented in Chapters 4, 5, 6, 8 relied on rich but less precise data that is primarily qualitative and captured through semi-structured interviews. The study captured in Chapter 7 used a survey to collect data. The open-ended questions in the interviews allowed us to ask wide-ranging questions to the interviewees. The interview guides' questions ensured that we consistently asked all the interviewees' relevant questions. We conducted interviews with one interviewer in the start-ups and hubs in as much as we could have conducted them in groups. We did this to increase the coverage and seek multiple views of the same subject from different start-up founders and hub managers, thus investigating the phenomena deeply.

Each interview involved presenting the interview objective and explaining how the interviewer will use the interview data. We then asked for consent by reading the consent form to the interviewees. After that, we asked the main questions. The interviews generally followed the pyramid model, which allowed us to capture enough contextual information like the background to the start-ups, experience, and roles of interviewees and enabled us to ask broader questions around the main subject areas.

The interviews' transcription was a time-consuming exercise, but it enabled us to gain new insights, as the principal researcher did it. It is always advantageous for the interviewee to review the transcript to ascertain whether the captured information is what they said or meant to say. The interviewees thus have an opportunity to correct or update any of the information before it is analyzed. We desired this, but it was always challenging to get the interviewees to do reviews due to limited time. This challenge was because we selected to interview founders and hub the managers in the start-ups and hubs. These interviewees were generally very busy people.

To characterize the metrics used in start-ups, we iteratively derived ten dimensions. The dimensions formed the basis for the online survey tool. We administered the survey to

software start-ups and mature software companies in Uganda. We analyzed the survey responses using descriptive statistics and hypothesis testing using R-programming and report the results in Chapter 7.

The study's interviews were recorded and stored as audio files, transcribed into text and stored as transcripts. The survey's quantitative data was captured in excel sheets and stored on google drive for subsequent analysis. We coded the interview transcripts using Atlas.ti and analyzed them using Thematic and content analysis techniques. During the coding and analysis of the transcript, the priori codes and the emerging themes were shared amongst researchers using Google docs. The researchers reviewed, discussed, refined, and reached a consensus on emerging themes before analyzing and reporting. The individual studies were presented as conference and journal papers but are modified (to suit the general outlay of the thesis) in this thesis and presented as Chapters 4-8. We stored the resulting raw audio, interview transcripts, code sheets, and survey results on google drive.

3.5 General Threats to Reliability and Validity

In this section, we discuss the general threats to reliability and validity specific to the comprehensive study using the classification scheme by Yin [115]. We describe the details of exact threats for each study in the respective chapters.

3.5.1 Threats to Reliability

Participant Error: The semi-structured interviews were conducted in 30-45 minutes to avoid participant fatigue. Emerging questions were noted down by the interviewer for subsequent follow up. A second researcher reviewed the interview guides and online surveys to clarify them and improve their interpretation to reduce participant error further. For example, in the interviews, we did not combine multiple questions for one response. We avoided combining multiple questions to ensure that interviewees responded to each aspect of interest as advocated by Robson [125].

Participant Bias: To ensure that interviewees gave unbiased information, we mainly interviewed founding members of the software start-ups and managers of hubs. The assumption was that the founders of the software start-ups and managers in hubs have experience and are knowledgeable in their start-ups and hubs' various operations. To ensure that the participants give factual information, we included anonymity and non-disclosure clauses in the consent agreements. We spelt out the anonymity clauses at the start of each interview. We also gave assurance to the interviewees about the confidentiality of their business information. The confidentiality assurance aimed to make the interviewees more relaxed and open about the information they were willing to share. This assured the interviewees of anonymity and confidentiality. We did it to increase confidence and trustworthiness and reduce the possibility of interviewee bias [113].

Interviewer Error: To reduce the potential of asking the same questions differently and possibly getting varying interpretations from the interviewers, only one interviewer did all the interviews. A second researcher reviewed the interview guide's structure and questions and the themes to refine and improve them, after which we reached consensus on the themes before executing the interviews. The interviewer also scheduled 2-3 interviews per day to reduce the workload and avoid fatigue. The interview load balancing ensured no need to hire other interviewers who may have introduced procedural errors.

3.5.2 External validity

External validity is concerned with the extent to which the findings are of interest outside the study. Case studies mainly provide analytical generalizations where the findings can be extended to cases with common characteristics [120]. In this research, we interviewed software start-ups and hubs. We surveyed a relatively small number of start-ups and mature software companies. Researchers often echo the number small number of interviews as a shortcoming that leads to a lack of generalization of qualitative findings due to a statistically small and unrepresentative number of cases [113]. Since the survey used non-probabilistic sampling, we cannot make strong statistical inferences and generalize to a broader target population of software start-ups and mature software companies. However, the results are intriguing enough to motivate a large more exhaustive study of the character of metrics using the emerging 10 dimensions.

3.5.3 Construct validity

The nature of the survey instrument inherently has validity issues that we tried to address. Saunders and Thornhill [113] define construct validity as the extent to which the measurement questions measure the presence of the constructs that it intended to measure. There was a potential risk of misunderstanding the metric categories in the survey. To mitigate the risk, we adopted well defined and validated metric categories from MeSRAM [1]. The descriptive preambles had examples of increasing the category understandability.

3.6 Ethical Consideration

To respect the research ethics, we considered informed consent issues, participant data confidentiality, the handling of sensitive results, and feedback. To alleviate some of these issues, we created an informed consent form. For the interviews and pilot surveys, we attached this consent form as the first page in each interview guide that was read to each interviewee to ask for their consent. For the pilot survey, the online form, and the initial hard copy trails for the questionnaire, we attached the consent form as the first page. The consent forms generally included the following clauses. We read out the following clauses in the consent form to the interviewee:

- Name of the researchers and contact information.
- The research purpose and its current state.
- The participants' right not to respond to any question in the interview and that the interview would stop if the interviewee wishes.
- The right to confidentiality and anonymity, stating that nothing said in the interview would be attributed to the interviewer without seeking and obtaining their permission.
- The expected study output study and what will happen to the data collected during and after the study.
- The offer to provide a summary of the research findings and indicate when this will happen.
- The request to record the interview, and if agreed, recording the interviews.

Before we commenced each interview, we asked for permission to undertake the interview, spelled out the study's main objective, and briefed the interviewee about the themes the interview covered. We stated the time the interview would take and requested the interviewee to read, understand, and sign the informed consent form. Samples of these forms are attached in Appendices 10.6, 10.6, 10.6. To ensure respondents' confidentiality, we used coded company names for transcripts and did not use interviewer names during the presentation of the results and analysis of the results. The audio recordings for the interviews were also separately stored from the interview transcripts. In the coming chapters, we present individual studies as Chapters 4-8. Each subsequent chapter summarizes the study (as a Chapter Overview), the research context, related work to the study, methodology, results, discussion of the results, validity threats, and future work directions. We published the five studies as conference and journal papers, and we have modified each to fit the thesis format.

3.7 Summary

The overall study is qualitative and exploratory in nature and used case studies presented as chapters 4, 5, 6, 8, and a survey presented as 7as part of its research design. The case studies used semi-structured interviews and data was analysed using thematic synthesis and content analysis and reported interpretively in relation to the chain of evidence presented through emerging themes. The survey used an online tool and data is reported descriptively through statistical analyses using the p-values obtained in the chi-square tests. We highlight the overall threats to the validity and how we mitigated them. We too, describe the ethical considerations for the entire study. The ensuing chapters present the details of each study and the specific methodologies, results, discussions and specific threats to their validity and emerging future work from each study.

Chapter 4

Paper 1: Exploring the Applicability of Software Start-up Patterns in the Ugandan Context

Chapter Overview

Software start-ups need to tackle many challenges as they grow. Therefore, re-occurring strategies are applied that can be captured in the form of patterns. While more and more of these patterns are published, we aimed to discover to what degree they are applied within different world regions. For this study, we studied the cases of 7 software start-ups within two incubation hubs in Uganda by performing qualitative interviews. We focused on five patterns from diverse areas of concern to analyze whether the Ugandan start-ups' strategies match these patterns. Our results showed that, for most of the patterns, we found matches. However, in some cases, the start-up strategies are only partially described by the known pattern. Therefore, these findings indicate that start-up patterns can often be transferred from developed ecosystems like Switzerland and Finland to emerging ecosystems like Uganda. We also found some variations from the known patterns in the contexts and solutions applied in Ugandan start-ups.

4.1 Introduction

Software start-ups need to tackle many challenges as they grow. They apply systematic strategies captured as patterns. In the context of this research, a pattern is defined as a proven solution to a problem in a specific context, borrowing from the definition by Gamma et al. [126]. While more and more of these patterns are being published, they base on start-up ecosystems in the developed world [14]. Also, considerable amounts of research on software start-up ecosystems have focused on their success, failure factors, and challenges [33, 12, 18, 9].

Therefore, it is not clear to what degree software start-up patterns can apply within different world regions, particularly the developing world, where the social, cultural, political, and economic conditions vary considerably. For example, start-ups in Uganda are in a different geographical space and operate in a potentially different software start-up ecosystem but still face some of the challenges faced by start-ups in Switzerland and Finland. We would not expect a direct contextual mapping from a Finnish-Swiss context to a Ugandan context. The software start-ups in Uganda face similar concerns and more. However, their

reaction to these concerns is likely to be influenced by the start-up ecosystem in which they operate. Therefore, our objective is to investigate whether well-known patterns collected from software start-ups in Switzerland and Finland apply in the same or at least similar form in Ugandan software start-ups.

The remainder of this chapter is organized as follows: Section 2.10.1 describes related work on the use of patterns within the software start-up ecosystem; Section 4.2 presents the research approach used in this study; Section 4.3 then presents the results while Section 4.4 presents discussions and suggestions for future work.

4.2 Research Approach

This study's main objective is to investigate the organizational, technological, and business practices of software businesses in Uganda in the start-up stages using software start-up patterns originated by Dande et al. [14]. We aimed to see the partial or full orchestration of these patterns in Uganda and highlight emerging patterns and practices influenced by the Ugandan start-up ecosystem.

We employed a case study [33] qualitative approach in which we interviewed six founders and one developer of software start-ups. We randomly selected the start-ups from two software incubation hubs from lists provided by hubs with current and previously incubated start-ups. The hubs and start-ups are located in Kampala in Uganda. Figure 4.1, summarizes the research process for this study.

Selection of start-ups We selected start-ups from two hubs in Kampala. Hubs are spaces where several software businesses start their initial growth into fully-fledged businesses. The hubs offer several resources and capabilities [9] that support software businesses in the early growth stages. In this study, our focus was on software businesses in the innovation – start-up stages and their intermediate stages (incubation, early-stage start-up). The selected start-ups ranged from those providing software as a service/support tool to those that produced software as a product. These were classified as start-ups since they were having their core operations based in hubs and still leveraging their resources and capabilities. These start-ups are still experiencing challenges that characterize start-ups like inadequate funding, organizational challenges, competition, unclear/non-standardized technical processes, and market uncertainty.

Interviews To study the start-ups, we used interviews with semi-structured questions to capture a rich dataset for analysis. We use interviews since they are common approaches to data collection in case study research and other empirical studies [18]. We adapted an interview guide used by Dande et al. [14] that we structured into three major sections: start-up identification and role information; the second section required start-up formation information, the team, customers, product/service, and start-up challenges. The last section assessed product/service features and uniqueness, the customers, and the development process.

Selection of start-up patterns Giardino et al. [29] present diverse challenges of start-ups such as difficulty in thriving in technology uncertainty, acquiring first paying customers, acquiring initial funding, delivering customer value, and defining a minimum viable product, targeting a niche market, and staying focused and disciplined. They group these challenges into four dimensions of a product, finance, market, team using the work presented by

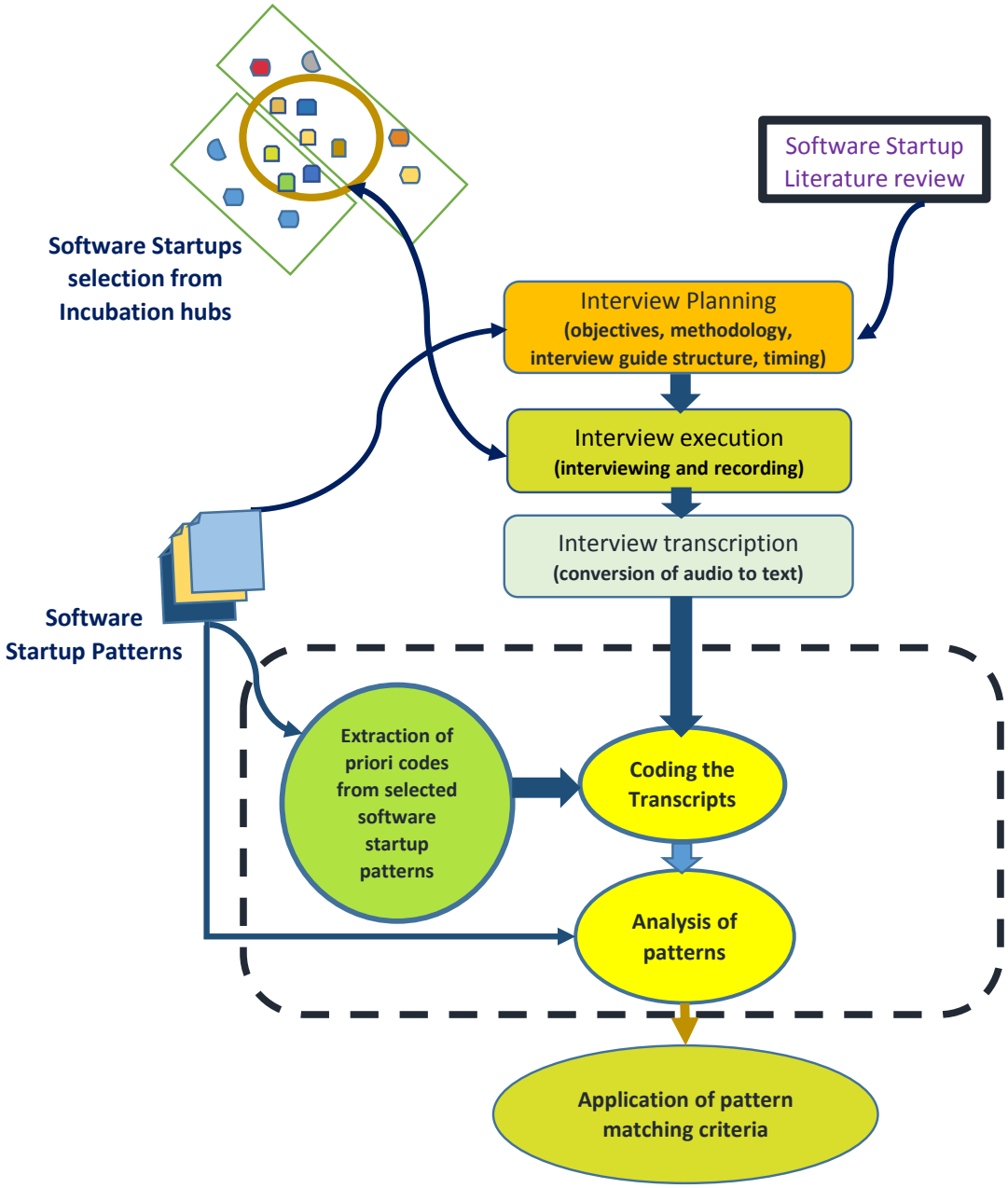

Figure 4.1. Research Process

Macmillan et al. [12]. Dande et al. [14] group their patterns into similar but more detailed categories that include funding, process, team, culture, organization, competence, design, testing, technology.

For this study, we focus on five patterns that provide broad coverage of the pattern categories. We also took into account that patterns are connected [14]. It can be assumed that the connected patterns have a high probability to be applicable in similar contexts. The selected patterns are:

- #32 Time process improvements right (Category: Process)
- #38 Unique value proposition (Category: Competence)
- #45 Keep customer communications simple and natural (Category: Customer)

- #62 Self-funding (Category: Funding)
- #64 Start with a small and experienced team and expand as needed (Category: Team)

Interview execution We recorded the interviews, and the interviewees were required to answer all questions in the guide concerning their start-ups. These interviews lasted for 40-50 minutes and were done in July and August 2016. We transcribed and later analyzed the transcripts. The interviewees included six(6) founders from each of the six(6) start-ups and one software developer from the seventh start-up. Before the seven interviews, we took the first draft interview guide through a dry run with a university resident software start-up. This run helped improve the question clarity and breadth and determine the interviews' duration before the main interviews.

Code preparation The patterns by Dande et al. [14], only highlighted the problems without the other vital sections in Coplein Format [38] thus making them incomplete for our purpose of analysis and also incomplete by the general definition of a pattern as a problem-solution set [126]. By examining the start-up category, context, problem, and forces sections of the twelve start-up patterns, we derived the first codes from our interviews. We aggregated these codes into larger codes (Priori codes) to reduce the code set. We used the priori codes and the emerging codes for qualitative analysis.

Analysis We used the transcripts for analysis by applying the thematic synthesis, a qualitative analysis technique that involved following the five steps outlined by Cruzes et al. [127]. We transcribed and coded the transcripts, created a priori codes from the patterns by [14], and used them to code the transcripts. During the coding process, we added emergent codes from the transcripts. We used the tool Atlas.ti[1] for the coding process, management of the data, and data analysis. We discovered many codes (priori codes and emergent) in the first round of coding. All codes were examined for relationships, existence or absence of software start-up patterns, and emerging practices in the software start-ups.

Conformity criteria We use a modified Coplien [38] start-up pattern format to match the patterns observed by Dande et al. [14] with those found in the Ugandan start-ups. Figure 4.2 illustrates the Coplien [38] sections of describing patterns. We consider the three sections of context, problem, and solution to be key to a pattern based on various pattern definitions [18] and pattern uses in different domains [126]. For the matching, we consider the other sections of Coplien format optional. Based on this, we define the following matching criteria that we used in this study:

Complete Pattern Match: We consider a start-up to match a pattern if, for each of the pattern's three key sections (context, problem, and solution), we can find evidence that the pattern fulfills parts of those sections.

Partial Pattern Match or Variant: We consider a pattern partially matching in a start-up (or a variant), if either 1. aspects of one, the pattern's context and problem, can be found in the start-up, but no aspects of the pattern's solution, or 2. aspects of the pattern's solution can be found in the start-up, but no aspects of either the pattern's context and problem.

[1] http://atlasti.com/

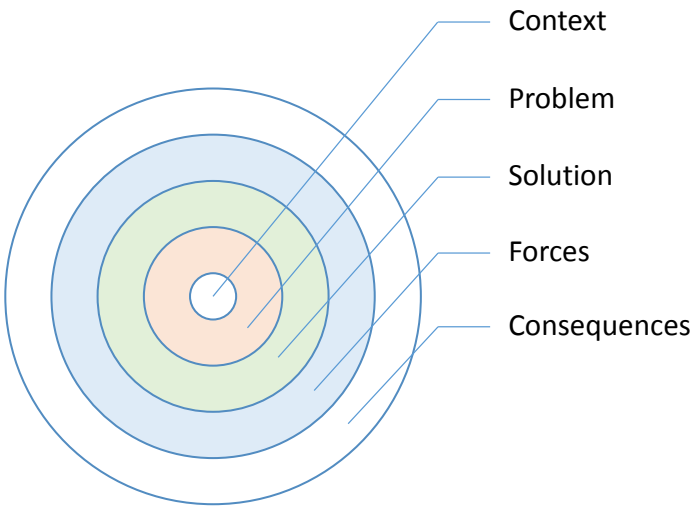

Figure 4.2. Coplien software start-up pattern synopsis

Table 4.1. Development activities and processes within the start-ups

Applied development activities, and processes	**Software start-up**
Prototyping	**A, D, E**
Waterfall model with agile phases	**A**
Agile development methodology	**B, F, G**
Iterative Evolutionary process	**A, F**
Testing product development process	**D**

No Pattern Match: We consider a start-up not to match a pattern when there is no evidence that the start-up matches parts of the context of the problem and simultaneously no evidence that the start-up matches parts of the pattern's solution.

4.3 Results

This section describes the five patterns and results obtained from our application of the pattern matching criteria to each interviewed start-up. The interviewed software start-ups' names have been abstracted using letters (A, B, C, D, E, F, G) for confidentiality.

4.3.1 Time process improvements right (#32)

There is evidence that Ugandan start-up teams are using some agile processes or adapted agile practices for developing their software. These teams pay significant attention to the product and not the process as spelled out in the pattern. In the study, we asked the start-ups to describe and characterize their service/product development processes, to state whether they are using standard or adapted development processes. The results are shown in Table 4.1.

The software start-ups adapt standard or agile processes and practices that match the start-ups' natural way of working, possibly leveraging their resources, team capabilities, and skills. The following practices in Table 4.2, where identified.

Table 4.3 summarizes how the pattern is matched by the interviewed start-ups.

Table 4.2. Software start-up practices

Start-up Practices	**Software start-ups**
Codifying business knowledge	**F, G**
Using tools to preserve and share knowledge	**A, D**
Beta testing	**B, C, G**

Table 4.3. Matches to Pattern #32 (legend: "?" = don't know, "O" = observed, "A" = solution applied, "-" = does not apply)

	Company	**A**	**B**	**C**	**D**	**E**	**F**	**G**
Context	- Growing Phase: start-up is in growing phase and both the product and the software evolve.	O	O	O	O	O	O	O
	- The development team has some processes it uses to develop the software. The process may consist of scrum like agile practices.	O	O	O	O	O	O	O
Forces	- The team is forced to improve the process at some point, for one or more of the following reasons:	O	?	O	O	O	O	O
	- The team may face difficulties with the process.	?	?	?	?	?	O	O
	- The product may not evolve quickly enough.	O	?	?	?	?	?	?
	- The quality of the software may not be on a high level.	?	?	?	?	?	?	?
Problem	- Right timing is crucial: At some point, in preparation for growth, the start-up needs to plan for better processes and example workflow tools. Nevertheless, doing this may require the focus and effort of the best people in the small company. That is why the improvements must be planned and timed carefully.	-	?	?	?	?	?	?
	- Wrong focus on the process: The process is never perfect, and the team can always improve it. In general, the team should pay attention to the product, not the process. If the product is good, the software process is mostly only a matter of programming and often releasing the software to the end-users.	O	O	O	O	O	O	O
Solution	- Plan and time process improvements right. Have a retrospective scheduled and use the outcome of the session to improve the process.	-	-	-	-	-	-	-
	- Do not improve all the time – focus on the product instead.	A	A	A	A	A	A	A
Consequences	- The process is improved, but the improvement itself does not consume all the resources of the team all the time.	A	-	A	?	?	?	?
	- The team has the focus in the product.	A	A	A	A	A	A	A
	- The software development process is good enough.	?	?	?	?	?	?	?

4.3.2 Unique value proposition (#38)

All the start-ups exhibited uniqueness by providing domain-specific products, platform-specific products, or niche products/services. Some are staying competitive by adding demand-driven features (e.g., found for A, C, F, and G) and by adapting to changing business needs as explained by start-up G: *"I would say, in the beginning, we were just washing, come to you, wash and that is all. By the way, we realized there are a lot of other services attached to the service because after you wash, you can use this water to clean*

the veranda or clean the house. So that means it is a service that's directly connected to washing, so we add it."

G, further went on to say: *"the first idea was going on the app, you find a list, click on somebody, call them, but then we got some challenges that (it) does not make money. We also saw that people need to book. I may not need to go, search for people, and everything. So we had to find a way of adding that in quickly. ..."*

Three start-ups (A, D, F) also intimated that they face external (outside country) competitors to their businesses. These start-ups exploit the market gap by providing something new within Uganda but not new as these services/products exist elsewhere in the world.

Table 4.4 summarizes what aspects of the pattern are matched by the start-ups.

4.3.3 Keep customer communications simple and natural (#45)

This pattern exists in the context that if a customer can trust in the product/service that a start-up is providing, it is easier to establish a simple and natural communication between the start-up and the customer Dande et al. [14]. This context drives the start-ups to build and establish affordable, efficient, and natural communication channels from the early stages of the product/service development to potentially throughout the lifetime of a start-up. This simple and natural communication can be through emails, one on one physical meetings, application features, prototype demos, and social media (can work in some contexts). Start-ups are generally resource-constrained and are looking for funds. Thus, they may not have resources to invest in information systems and thus opt for simple and more natural communication methods to collect feedback and build trust relationships with their customers. Start-ups, too, may integrate mechanisms in their products/services that can continuously provide feedback to and from the customers.

During the study, all seven start-ups indicated that they either have an existing customer feedback channel or have a strategy to capture customer feedback and establish customer trust. For example, start-up E said:

"... our major focus now, customer feedback and customer care. It is one of the core values of a company. So I intend to have a team follow up on these clients, so we sign up a client, but it does not end there. We make a call almost every week, at least we call you once just to find out if our system works fine, and if clients have queries, we will set up call times, we have emails."

start-up A had this to say:

"We collect feedback in two ways, we have a direct feedback form or people send us to feedback directly to our databases, then we also collect using the user interaction, from the time someone opens the application until when they press a particular feature, we record that, ..."

Generally, all start-ups use or intend to use relatively inexpensive, simple, and natural communication channels. Table 4.5 summarizes how the interviewed start-ups match the pattern.

4.3.4 Self-funding start-up pattern (#62)

Funding is one of the major challenges of start-ups. In the start-ups, this pattern is evidenced through bootstrapping from the founders (C, E, F, G) and venture capitalization

Table 4.4. Matches to Pattern #38 (legend: "?" = don't know, "O" = observed, "A" = solution applied, "-" = does not apply)

	Company	**A**	**B**	**C**	**D**	**E**	**F**	**G**
Context	- Need to make ground on a competitive market to take off in the first place and sustain.	O	O	O	O	O	O	O
	- A unique value proposition combining both technical and social values is needed to be adopted and practiced for long term sustainability.	?	?	?	O	?	?	?
Forces	- Distinctive competence is required to cope with dynamically changing market demand.	O	?	O	O	?	O	O
	- Adoption to changing market needs is essential to contend with competitors.	O	O	O	O	O	O	O
Problem	- To stand out in the competition with established market players: adopt a viable approach embedding a unique value proposition to support human-centric services along with technical competencies.	O	?	O	O	O	O	O
Solution	- Find a unique approach that differentiates the company in the market and is good for business.	A	-	A	A	-	-	A
	- Make the approach visible as a part of the brand, and cultivate it within everyday activity.	O	-	-	-	-	-	-
	- Then find a conscious way of action/competence to practice the adopted approach which would effectively separate you from the competitors.	-	-	-	-	-	-	-
	- Optional: Emphasizing on communication skills in order to better understand the customers and their needs	?	?	O	?	?	?	?
	- Market the competence as a service to the customer.	O	?	-	?	?	?	?
	- Gradually develop competences, practices and tools within everyday practices.							
Consequences	- Company stands out in the market	O	?	-	-	-	-	O
	- Easier to find customers							
	- Unique and natural way of doing things would ensure increased motivation and satisfaction to the involved personnel, e.g., employees and customers	?	?	O	?	?	?	?
	- It might be an extravagance for start-ups to focus on issues other than the product	O	O	O	O	O	O	O
	- Building such unique value proposition may require additional resources	?	O	O	?	O	O	?

(D) and through seed funding or grant (A). Start-up B's, funding sources are unknown due to its collocated nature and our inability to interview the founders. These start-ups also highlight the need for funding by drawing plans for funding, offering equity to attract needed competencies/key skills (F), and sharing operational costs (A) by starting up in incubation spaces and exploring potential investor networks.

Our observation is that the Ugandan start-ups exercise the self-funding start-up pattern, but these sources of funds vary within the start-ups. The bootstrapping model (funding

by founders) is a common one. Venture capitalization/seed funding are the other funding avenues. As a consequence of bootstrap funding, start-ups that employ this model have full control of their operations, make business and financial decisions, and even sell equity to angel investor. Table 4.6 shows what parts of the pattern are covered by the interviewed start-ups.

4.3.5 Start with a small and experienced team and expand as needed (#64)

This pattern focuses on team formation and team dynamics. To probe this pattern, we asked the start-ups about the number of their employees/founders, their roles (primary and secondary), and the key employees' experience. Small teams of 2-4 members of mostly multi-skilled founders members made up the seven start-ups. Many founders play multiple roles like programming, software development, business analysis, marketing, and user interface designing. Five start-ups (A, B, C, D, E) have technically experienced team members. The two inexperienced start-up teams (F and G) talked about the willingness to expand on their limiting skills in later stages of the start-ups' growth.

One of the forces for this start-up pattern (#64) is that increased team size increases communication and communication complexity. This increased complexity slows down development in early-stage start-ups. As shown in Table 4.7, there is reduced communication overhead for five start-up teams with small team sizes. Two of these experienced teams (A, B) and two technically inexperienced teams (F, G) intimated that they first get team consensus on the required product/service features before developing them. The other start-up did not have this consensus approach for feature addition. From table 4.7, start-up B is co-located from an offshore start-up that is developing software services in Uganda. Co-location for software development means that a team member is hired for a specific skill to execute a role for the parent company. The majority of the start-ups start with a small and experienced team and expand as needed, as pointed out by the pattern (#64). Table 4.7 summarizes how the start-ups match the pattern.

4.4 Discussion of Results

In Table 4.8 we summarize which of the start-up wholly or partially match the five patterns.

Time process improvements right (#32) Like the start-ups in the Finnish-Swiss context, the start-ups in Uganda are in their growth phases and are using agile or adapted agile processes. They focus their available skills and effort on the product and thus planning less about the processes in these early stages of their growth. All seven start-ups are using a process that suits their product. However, since they are in unique sub-stages of start-up growth, process improvement is not planned, neither is it shown to be timed. The problem that may be driving the start-ups in this early start-up phase to use agile processes could be, to have a Minimal Viable Product (MVP) ready for the market. The start-ups thus make process improvements after the product/service is operational. The start-ups are not timing process improvements, but they adapt existing processes (as shown in the results) to fit the product/services in their start-ups. The teams have not indicated that they face product delays due to lengthy or not optimized development processes. Given the early growth stages of these start-ups, it is likely that the feedback channels they currently have are being used to refine product features. There is also no clear evidence in the start-ups that their products or services' low quality is forcing them to improve

their development processes. Thus, none of the start-ups interviewed is experiencing issues related to improving processes like resource re-allocations and proper timing. Consequently, we found for none of the start-ups a solution related to issues arising from poor planning and timing of process improvements. Instead, the start-up teams balance their efforts by paying attention to the product/services and not improving their process. To sum up, all seven start-ups were found matching this pattern by Dande et al. [14].

Unique value proposition (#38) All the start-ups indicated that they offered something unique by providing domain-specific products, niche products/services, or exploiting market gaps (social uniqueness) for solutions that do not exist in Uganda but exist in some other countries. Arguably, this uniqueness will make them competitive, and that will make them stay sustainable. Start-up D also exhibited its ability to incorporate social values within its value proposition. One of its founders intimated that *"our games are usually region-specific. For example, matatu is played here [Uganda], we have another one called lost card that played in New Zealand, we have Kavata for Kenya, right, and so we try to give each market the product they want. So the players who are going to play these games give them what they already know because these games like Kavata is an already existing game in Kenya, we just put it on mobile"*. Start-up D is unlike all the other six start-ups that seem only to have a unique value proposition through their technical values. The majority of the start-ups indicated that they have something unique. However, start-up B (co-located team) was just starting an innovation cycle for its service and was still in the early stages of prototyping and getting very early feedback. None of the six start-ups indicated that they are standing out through visible branding. The lack of branding could be due to limited funding and limited resources. Start-up A is starting to brand and market its competence as a service to the customer by leveraging its early partnership with one of the top TV broadcasters in Uganda. Start-ups C is using communication skills to understand their customers better. For example, they walk into customer premises to solicit business and understand how traders order their products.

To summarize, start-ups B, E, and F, in very early stages, have not yet found a solution towards emphasizing unique values. However, we made an interesting observation that they seem to follow other strategies. For example, start-up B uses low-fidelity prototypes to clarify their new idea for innovation and then invite students to work with them. Thus, while it seems that start-ups do not fulfill this pattern in their early stages, there seem to be variant solutions that might be explored as potential patterns in future work.

Self-funding (#62) We observe that Ugandan start-ups are using multiple funding avenues. Some are using savings from founders as a bootstrapping option in their earlier stages, and as they grow, they increasingly become funded by their acquired customers. Since most of the interviewed start-ups have been self-funded in their early stages, they conform to the pattern's context.

Solution Variants

- We, however, see some variation in the solution suggested here. Start-up A is also partly funded by a venture capitalist (as part of their funding model), which deviates from the pattern's solution. Consequently, early venture capitalization in start-up A creates dependence and potential early external control, thus stifling future growth. This assertion is intimated by founders of A, that say *"In a bid to reach our desired future goals, as a company, we want to diversify, as a company in products. Having (Company X) as a client (venture capitalist) is good, but we want to be independent. We do not want to ride with (Company X) for the business model to work, so we look*

to have 100 percent independence from all other stakeholders. The business model is fully working, and our products are being used actively. That is our desired future goal".

- As a consequence, this lack of funding by the Ugandan start-ups is forcing them to draw funding plans (E and F) and look for ways to get more funding through equity offering (F) and exploring investor networks (D).

The self-funding pattern is fulfilled by the software start-ups A, C, F, G with some variation in the solutions for start-ups A, E, F, and D when we use our pattern matching criteria.

Start with a small and experienced team and expand as needed (#64) All the interviewed start-ups are small teams, with the majority of them multi-skilled and multi-roled. As a solution in this pattern, the small teams inherently have reduced communication overhead. From our results, it is apparent that the experienced teams appreciate the work of the experienced team members, and teams use the skills to execute multiple roles in these start-ups. Consequently, with the small teams and inherent simple communication, all the start-ups inherently have less operational overheads. However, the two teams whose members arrived from different organizations can potentially clash in work culture as they grow. All the seven interviewed start-ups match the start-up pattern (#64).

Keep customer communications simple and natural (#45) All these start-ups are establishing communication channels in their early stages of growth. The force to this could be primary to get early feedback on the products/services they are developing but not being forced by the customer culture, by customers wanting to talk about their business ideas or organizational issues as spelled out by the Dande et al. patterns. The early feedback may help establish early customer trust in the product/service since the customers have been part of its development. Due to their limited resources in terms of staff and finances, all the seven start-ups establish or use cheap communication channels that include emails, social media, forms, user trackers on web pages, and direct walk-in customers. Three start-ups (A, C, G) are building a trust relationship with their customers. The start-ups with considerable customer numbers (A, C) employing web technology to keep in touch with their clients. All the seven start-ups have evidence to match at least one of the solutions described by Dande et al.[14] start-up pattern (#45). All the start-ups conform to this pattern in the early stages, and possibly they will orchestrate it more as they acquire customers.

Wrap-up discussion To sum up, we see that Ugandan start-ups are mostly applying solutions known from Dande et al. However, many Ugandan start-ups do not fulfill many parts (e.g. solutions) of the patterns. Thus, with a few exceptions, the Ugandan start-ups conform in the way they fulfill the patterns and use solutions. That might be a hint on the influences of the context. It might be possible that a specific solution is more comfortable to apply in Uganda while others are not feasible. Consequently, there is a need for future research on whether these solutions that we have not seen are really out of reach in the context of Uganda or whether they present a so far unused potential that can help start-ups in the future to be more successful.

4.5 Limitations of the Study

In this preliminary study, we interviewed start-ups that are still in incubation hubs. We did not interview start-ups that graduated from the hubs or may not have been part of the hubs in as much as they exist. This exclusion of start-ups may affect the generalizability of some of the patterns to all start-ups in Uganda since hubs' processes and resources may influence the hosted start-ups. Due to this study's qualitative nature, which incorporates a sample size of seven start-ups, we cannot make statistical generalizations of our findings for all the software start-ups in Uganda. Finally, while Dande et al.'s work on a pattern language has been peer-reviewed, the report on the pattern themselves has not [14]. The lack of peer-review for the patterns represents a threat to validity for comparing Ugandan start-ups with Swiss-Finish start-ups. Nonetheless, the results for the Ugandan start-ups are valid. Future work can further investigate what degree solutions presented in the pattern of Dande et al. occur in Europe.

4.6 Conclusion and Future Work

We used the software start-up pattern descriptions by Dande et al. [14] and examined start-ups in Uganda to see how well they conform to patterns in the Finnish-Swiss context. To our surprise, we found that all five patterns have matches within the seven interviewed start-ups in Uganda. The matching is good news since it confirms that these patterns' generalizability ranges across cultural and ecosystem differences. Furthermore, we identified that the interviewed companies often adopt similar aspects of the patterns' solutions while others are left out. The context and problems are often similar, which is an exciting finding as it hints at specifics of the Ugandan context. However, future studies should evaluate this. Finally, we found that some companies have only partial matches of the pattern, and this provides a starting point for us to search for pattern variants specific to the local context. We will also like to explore alternative pattern conformity criteria for future work and draft a framework for validating pattern conformity for various start-up patterns.

Table 4.5. Matches to Pattern #45 (legend: "?" = don't know, "O" = observed, "A" = solution applied, "-" = does not apply)

	Company	**A**	**B**	**C**	**D**	**E**	**F**	**G**
Context	- When a customer can trust in a product/service a start-up is providing to her/him, it is easier to establish a natural and simple customer communication between the start-up and her/his customer. When communication is natural, it is easier to learn about the customer's way of thinking and the culture to which the start-up is developing its product.	-	-	O	-	-	-	-
	- Added to that, when the communication is also simple enough, setting it up does not take too much effort from the busy start-up. This pattern is an essential issue throughout the life cycle of start-ups.	-	-	O	-	-	-	-
Forces	- Sometimes the start-up can be forced to communication manners coming from the customer's culture which lead to formal or distant communication. Filling forms is not as natural as sending emails or talking face-to-face.	-	-	O	-	-	-	-
	- Very informal communication can lead to ambiguous requirements or decisions when it is difficult to have a consensus at the critical moment (e.g. changes needed in the product development)	?	?	?	?	?	?	?
	- Using a social media can work in some context but is not a solution to all.	-	-	O	-	-	-	-
	- In some cases, a customer might want to talk about her/his business ideas or organizational issues, which are not a focus of customer product development. Being a good listener and a guide for relevant activities within the product development both are included in successful customer communication.	?	?	?	?	?	?	?
Problem	- start-ups are mostly getting feedback to development.	O	O	O	O	O	O	O
	- They have limited resources to put effort into customer relationship management, although this is important for establishing long-term customer relationships. How to keep the customer communication natural and straightforward within these constraints	O	?	O	O	O	O	O
Solution	- Build a trust relationship with the customer.	A	-	A	?	?	?	A
	- Give customers natural ways to communicate continuously. As a main point of customer contact, be present at the customer's office often and communicate face-to-face. Informal communication should be unambiguous.	?	?	A	?	?	?	?
	- Allow direct contacts by email or some other communication channels with which customers feel comfortable.	A	A	A	A	A	A	A
	- When the start-up has a larger customer base, a web communication technology probably is needed	A	-	A	-	-	-	-
	- Focus on the relevant and efficient communication forms.	A	A	A	A	A	A	A
Consequences	When behaving naturally and openly in communication with a customer, it is easier to establish a trust relationship and have a full understanding of the customer's requests.	?	?	?	O	?	?	?

Table 4.6. Matches to Pattern #62 (legend: "?" = don't know, "O" = observed, "A" = solution applied, "-" = does not apply)

	Company	**A**	**B**	**C**	**D**	**E**	**F**	**G**
Context	- Funding required (when starting up a new company or entering a new phase)	O	?	O	O	O	O	O
	- The idea/product/domain is such that self-funding is possible. One of the following situations: 1) for **early stage** start-ups, 2) when the idea is difficult to fund, 3) when founders want freedom in making a decision, or 4) when you want to encourage the creation of something financially viable.	O	?	O	O	O	O	O
Forces	- Money required to start-up, to build, to live and to grow (with few exceptions, but always need for money to live).	O	?	O	O	O	O	O
	- Getting funding is difficult	?	?	?	?	?	O	?
	- External funding comes with external influences; Self-funding retains these freedoms.	O	O	-	-	-	-	-
	- With self-funding, the need to create a financially viable company/product is greater than normal. This need creates drive, which encourages creating something that sells and is thus financially viable.	?	?	?	?	O	O	?
Problem	- Lack of funding ("How to create a product when there is a lack of funding?")	O	O	O	O	O	O	O
	- Secondary: "How to keep freedom in decision making/control over the company?"	O	-	-	-	-	-	-
	- Secondary: "How to encourage yourself to build something that can be successful?"	-	-	-	-	O	-	-
Solution	- option 1) work only part-time on the product,	-	-	-	-	-	-	-
	- option 2) get money from consulting,	-	-	-	-	-	-	-
	- option 3) use savings, or	-	?	A	A	A	A	A
	- option 4) use other means of self-funding.	A	A	-	-	-	-	-
	- Later stage: get funds from acquired customers (= customer funding).	A	-	A	A	-	-	A
Consequences	- Freedom: By bootstrapping, you do not rely on external funding, and there is no-one else controlling anything, such as steering decision towards their interests	-	-	O	O	O	O	O
	- Lack of funding forces you to build something that works, something that generates revenue and funding. It forces you to find ways to earn money.	-	-	-	O	O	O	-
	- Bootstrapping for too long: you need to set a bootstrapping deadline because it might hinder your growth and let the competition run you down.	-	?	-	-	O	O	?
	- Worst-case scenario: Wasted all your savings, wasted time and money trying to get the company off the ground.	-	-	-	-	?	?	?
	- Competitors might get ahead, because they might have the funding -	-	-	-	-	O	O	?

Table 4.7. Matches to Pattern #64 (legend: "?" = don't know, "O" = observed, "A" = solution applied, "-" = does not apply)

	Company	A	B	C	D	E	F	G
Context	- Market full with competitors timing is essential; no time can be wasted (focus on product development)	O	O	O	O	O	O	?
	- good communication and experience crucial to avoid operational overhead	O	O	O	O	O	?	O
Forces	- The amount communication increases with the size of team.	O	O	O	O	O	O	O
	- An increased communication complexity can slow down the development.	?	?	?	?	?	?	O
	- An inexperienced team member will require time to learn about the focused area and has to acquire skills.	-	-	-	-	?	O	O
Problem	- Too much overhead (waste of time). How to enable smooth product development with fewer overheads	O	O	O	O	O	O	O
Solution	- Small Team - to reduce communication overhead (normally 4 to 5 people, depending on context)	A	A	A	A	A	A	A
	- Co-location of team members - to preserve tacit communication and understandings	-	A	-	-	-	-	-
	- Experienced team members - to smoothen the work flow	A	A	A	A	A	-	-
	- Multi-skilled people	A	A	A	A	A	-	A
	- Growth phase: provide team members the opportunity to grow their skills	-	-	-	?	?	?	A
	- Growth phase: appreciate work of experienced members	A	A	A	A	A	?	?
	- Growth phase: lower risk of hiring inexperienced team members	A	A	?	A	?	?	?
Consequences	- Few operational overhead	O	O	O	O	O	O	O
	- A good quality product can be achieved by a starting company within reasonable time.	O	O	O	O	O	?	?
	- The experienced team members may have arrived from different organizations with different cultures. A conflict of cultures may arise.	O	?	O	?	-	-	O

Table 4.8. Summary of the pattern matches

	Full Matches	Partial Matches	No Matches
#32	A, B, C, D, E, F ,G	[None]	[None]
#38	A, C, D,G	B,E,F	[None]
#45	A, B,C, D, E, F ,G	[None]	[None]
#62	A, C, D, E, F ,G	B	[None]
#64	A, B, C, D, E ,G, F	[None]	[None]

Chapter 5

Paper 2: The Character of Software Start-up Hubs in an Emerging Ecosystem

Chapter Overview

Software start-ups face numerous challenges, and many fail in the first two years. As nurturing spaces, hubs provide incubation, acceleration, and co-working space as services to start-ups to alleviate these challenges. Previous studies have highlighted what early-stage software start-ups measure. However, given that they are mostly nurtured in hubs, there is a need to understand how the hub environment affects the early-stage start-ups through understanding the hub operations and measurements. Using semi-structured interviews with ten hubs in Uganda and Kenya, we characterize and analyze hubs' current practices and processes. The results show that most hubs combine incubation, acceleration, and co-working space as services and offer networking and team building events, in addition to value addition activities. Hubs provide mainly business growth incentives and notice the business and organizational effects of their incentives. They, too, have various selection checklists, provide incentives to alumni start-ups, and measure business and scalability metrics. Start-up hubs in East Africa are therefore prepared to address start-ups' business aspects but may need to improve technical mentorship and learn from each other's practices.

5.1 Introduction

Hubs nurture a considerable number of early-stage software start-ups. The hubs are normally identified by their services that include incubation, co-working space, and acceleration. These services may be offered to start-ups in combination by providing resources, incentives, and events to increase the start-ups' survival rates and generate revenues for the hub. Software start-ups face several challenges that have been highlighted by[9, 18, 29, 67] and up to 90% have been found to fail in the first two years of operations [4]. Through the different services they offer, Hubs have stepped in to address some of the challenges the early-stage software start-ups face. For example, accelerators mainly address the funding gap that start-ups face [68]. Incubators help the early start-ups address challenges related to the technical development of a solution, team formation, problem/solution fitting by providing resources and services. Co-working spaces create a community that harnesses

collaboration, sharing, trust, accessibility, and sustainability for software start-ups [128]. Several studies have identified the software engineering [129] and business practices in start-ups. However, not many studies have characterized how hubs in nascent ecosystems work with software start-ups in as much as an earlier study [10] found some variations in start-up practices in hubs. Understanding how the hubs operate concerning their software start-ups will provide an avenue for developing, bench-marking, and optimizing practices and processes for nurturing software start-ups successfully in hubs for emerging start-up ecosystems.

Hubs are popular in the developed ecosystems for nurturing start-ups, and their growing popularity in the emerging start-up ecosystems like East Africa is not surprising. Much of the literature explores hubs in the context of nurturing start-up businesses in general terms. As defined by [129] as temporary organizations, software start-ups seek scalable, repeatable, and profitable business models often operating in uncertain markets can be treated as unique businesses given their known challenges. Few empirical studies in emerging ecosystems characterize the practices and processes in hubs with a particular focus on software start-ups. There have been studies of the highly successful start-up ecosystems like Israel [23], Finland [130], New York [131]. In as much as these studies may have inspired the emergence of start-up ecosystems elsewhere, the operations of key ecosystem players like hubs deserve more focused empirical studies to understand them. Thus focus calls for a deeper study and understanding of the general practices and processes in these hubs that will result in an objective perspective to identify the important gaps for improvement, similarities, and generally successful practices that are used or must be adopted or adapted. This study aims to characterize hubs' operations concerning how they work with software start-ups in East Africa. Understanding operations of hubs in the East Africa ecosystems concerning specifically software start-ups, will enable researchers and practitioners to unveil current practices that can form a basis for the hubs to tailor their operations to nurture more successful software start-ups. The characterization of hubs will also lead to a better understanding of current practices in hubs. It may lead to better support structures that fit the needs of early-stage software start-ups within the hubs.

5.2 Research Methodology

Research questions To characterise the hubs, we ask the following research questions.

- RQ1: How do software start-up hubs operate in a nascent ecosystem?
- RQ2: What do software start-up hubs measure about the progress of their start-ups?

In answering RQ1, will provide a validated understanding of hubs' operation in the emerging East African. RQ2 will enable hubs to learn from each other to provide a basis for metrics that hubs can use to track early-stage start-ups' progress.

5.2.1 Case study protocol

We designed and followed a case study protocol as a desirable instrument that contains the general procedures for conducting the entire study, as advocated by Yin [115]. In this study, we used a convenient sampling where we identified hubs that incubated early-stage software start-ups in the past two or more years and were accessible to the researchers through previously working with the university or a credible known contact. We set up appointments with the hubs' managers by writing letters and calling them on the phone. We targeted the

hub managers involved in the day-to-day running of the hubs to ensure that the interviewees were knowledgeable about the hub's different aspects. We used a pre-designed interview guide with 16 questions grouped into three areas 1) hub practices, 2) activities/resources offered to the start-ups, and 3) measurements. The conducted interviews lasted 30-40 minutes in the hubs. We followed the protocol to ensure consistency in selecting the cases, data collection, analysis, and final interpretation of our results.

5.2.2 Case selection

In this study, we interviewed ten hubs hosting software start-ups. These hubs provided incubation, co-working space, and some incubation, acceleration, or co-working space) as services to start-ups. There have been broader definitions for hubs. For example, a UNICEF guide [132] describes hubs in terms of whom they work with (diverse stakeholders), their aim (lowering barriers to solutions from co-creators), location (embedded in local contexts) and operations (encourage collaboration through shared space). Chirchietti [20], defines a hub, an umbrella term for pre-incubators, accelerators, hackerspaces, and co-working spaces a physical environment that support start-ups and individuals at different stages of development. In this study, we generally adopt this definition [20] of hubs by focusing on the services they provide and taking a contextual view of the East Africa start-up ecosystem classified as nascent[17]. The hubs are not very many, and most offer more than one service to start-ups. We thus found it reasonable to study them without discriminating the services they offer.
The hubs were located in Kampala, Uganda, and in Nairobi, Kenya. We targeted these locations because most of the hubs are in capital cities due to the amenities, potential markets, skilled talent and entrepreneurs, and better internet connectivity. According to a GSMA report [21], of 2016, Uganda had 12 hubs, and Kenya had 27 hubs, most of them based in Kampala's cities Nairobi. In the hubs, we specifically interviewed managers whom we deemed to be knowledgeable and experienced and could thus give detailed accounts of the hubs' daily operations. We used purposive sampling [123] to select the hubs we interviewed. It involved two criteria; (1) In Kenya, we interviewed Nairobi hubs based on a known contact reference. (2) In Uganda, we interviewed active hubs where the university had alumni or some earlier working relationship.

5.2.3 Data collection

This study was mainly exploratory, and we collected qualitative data by conducting semi-structured interviews with the hub managers. According to Yin, well-conducted interviews can provide important insights into the phenomenon [115]. Evidence in case studies can is solicited from six sources: interviews, documentation, archived records, direct and participant observations, and physical artifacts. To execute the interviews, we followed the case study protocol and used semi-structured interviews, where we had an interview guide as described in the protocol. This flexibility in the interview structure enabled us to change the course of interviews and follow interesting directions [114]. This approach potentially gave us a deeper understanding of the current hub operations from the practitioners (hub managers).

5.2.4 Data analysis method

We recorded all the interviews and transcribed them verbatim [133]. We then used content analysis, which is one of the known approaches suitable for analyzing interview data

Table 5.1. Characterisation of the hubs

Hub	Country	Age (years)	Hub services	Capacity	Interviewee role
HCH	Kenya	8	Incubation, acceleration	100 start-ups	operations manager
HME*	Kenya	2	Co-working space	110 start-ups	operations manager
HNG*	Kenya	6	Co-working space	120 businesses	manager
HWO	Kenya	2	Co-working space	10 start-ups, 9 companies (SMEs)	operations manager
HMI	Uganda	3	Incubation, acceleration	23 start-ups	manager
HOU	Uganda	7	Incubation, co-working space	15 start-ups	operations and finance
HVC*	Uganda	9	Incubation, acceleration, co-working space	35 start-ups	programs manager
HIN	Uganda	4	Incubation, acceleration, co-working space	30 start-ups	operations manager
HRA*	Uganda	7	Incubation, acceleration	190 innovators	innovations officer
HSA**	Uganda	2	Incubation	20 start-ups	co-founder/ pitch event manager

Hubs marked with * have distributed operations **hosts does not physically host start-ups.

[134] for each of the transcripts. Content analysis can also be applied to a wide array of unstructured information like text and images. We coded the transcripts using Atlas.ti. The transcripts were uploaded into the tool and systematically coded as we read them line by line. The reading helped us to familiarise ourselves with the content further. We then identified the emerging codes by selecting and tagging text with appropriate codes. We tracked the original quotes to help us to maintain the context of a discussion of the results because one of the main criticism of coding is directed towards its potential for loss of context [135]

5.2.5 Content analysis, coding, and themes

One of the main features of content analysis is to use categories that are derived out of the data or brought to the empirical data [134]. These categories may be based on the research questions or emerge from common concepts (codes) repeated in the interviews. It is common to combine these approaches, as highlighted by Flick [134]. We followed this approach and derived the initial categories from the research questions. The categories included 1) hub work with alumni start-ups 2) hub acceptance criteria for start-ups 3) hub incentives and resources to start-ups 4) effects of hub activities and incentives on start-ups 5) measuring the progress of start-ups by hubs. We organized the initial/raw codes emerging from the transcripts around those categories. We further organized codes in each category into sets of related themes (thematic coding) [134]. These code categories and emerging themes formed the starting point for our analysis. We used spreadsheets to analyze the codes.

5.3 Results

This section presents the key results from the interviews. Table 5.1 characterizes the hubs in terms of their country of location, operational age, services it provides to hubs, capacity, and the interviewees' role. Note that the capacity of some hubs is approximated because they operate in more than one location. The time for hosting the start-ups varies from hub to hub, with very few hubs hosting start-ups for a definite time and the majority letting the start-ups stay and leave at their accord or after the expiry of their lease terms. The figure 5.1 below illustrates our envisaged hub operation model by interlinks the some main themes with emerging themes.

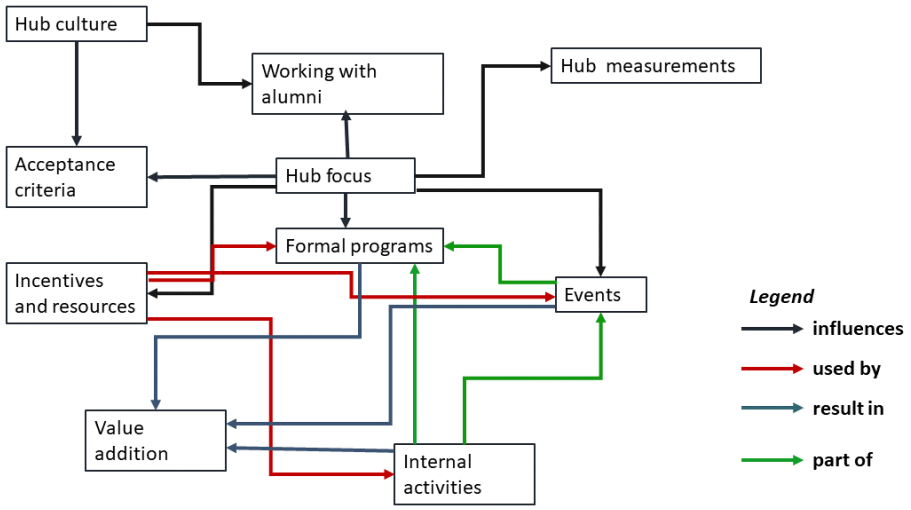

Figure 5.1. Hub operation model

5.3.1 RQ1: Hub operations

During the interviews, we explicitly asked the hub managers how they operate the hubs concerning software start-ups. We asked other questions that delved deeper into the operations of the hubs. These inquired about the 1) incentives, activities, and resources they provide to the start-ups 2) the effects of these incentives, activities on the start-ups, 3) the criteria for accepting the start-ups into the hubs, 4) how hubs work with their alumni start-ups and 5) how the hubs measure the progress of their hubs. In the end, we provide a conceptual understanding of hub operation by discussing the emerging themes from those leading lines of inquiry.

Hub operations

The results in table 5.2 revealed that hubs to a small extent had an existing culture, organised and ran events, had established formal programs, added value to their start-ups through training and incentives, executed internal activities, and focused on certain aspects of start-up businesses.

The co-working space [HWO, HNG, HOU] hubs respected the privacy of their start-ups and other businesses they hosted. They also catered for evolving business needs and provided differentiating service to the start-ups, as evidenced by hub HNG.

> *"...since everybody is their own company, we do not want to impose policies on them. So, we need to have those boundaries with our clients"* [HWO]

> *"A lot of the start-ups have their network of whom they work with, we do not really get so involved. It's a point to consider, we do not really offer their businesses that exist the services we provide for start-ups "* [HNG]

Table 5.2. Hub operations

Themes	Codes	Hubs
Hub culture	respect for business privacy	HWO, HNG, HOU
	catering for evolving start-up needs	HME, HNG
	service differentiation	HNG
Events	networking events (workshops, seminars)	HWO, HRA, HME, HMI, HCH, HNG, HVC, HIN
	team building events (pitches, hackathons, boot-camps)	HMI, HNG, HOU, HIN, HAS, HME, HVC, HCH
	collaboration events	HMI
Formal Programs	accelerator programs	HVC, HME, HIN
	catalyser program	HSA
	programming skills development program	HOU
Value addition	software development skills training	HOU
	business training activities (entrepreneurship, coaching, pivoting, marketing)	HCH, HVC, HRA, HMI, HME, HNG, HIN
	connecting with investors	HNG, HME, HVC
	idea development	HIN
	professional activities (business talks, legal advisory)	HWO, HVC
	access to the e-commerce platform	HSA
	financial modelling for start-ups	HME
	validating entrepreneurship skills	HSA
Internal activities	3D- approach (Discovery, Development and Deployment)	HIN
	carryout due diligence on start-ups	HME
	operations and finance	HOU
	start-up peer reviews	HVC
	six stage incubation process	HMI
	hub engages scaling partners	HRA
	project planning, implementation and management	HIN
	multinational operations	HME
	human-centered design approach	HRA, HOU
	activity workplans	HIN
	scheduled conferences	HCH
	business matchmaking/matching start-ups with opportunities	HCH, HME, HNG, HIN
	problem/solution fitting at ideation stage	HRA
	collaboration with development organisations	HOU
Hub focus	incubation of start-ups	HRA, HCH, HMI, HSA, HIN
	innovations support	HVC, HRA
	support entreprenuership	HME, HWO
	maturing businesses	HNG

> *"... you must want us to get involved in your work. So we do not force ourselves into you work but when you want to inquire on something, we always ready offer them with assistance. "* [HOU]

The majority of the hubs the provided incubation as one of their services did not identify a hub culture. A majority of the hubs held networking events (8/10) and team building events (8/10), with a notable exception of hub HSA, which held no such events. The exception with hub HAS could point to the fact that hub HSA does not physically host the start-ups. HSA, whose main aim as an incubator is to create jobs, has a unique operating model that trains university students in entrepreneurship skills, and the trainees then form start-ups. This operational model may not easily support holding events held regularly in physical hub spaces. All the ten hubs offered at least some value addition to their start-ups. The most popular of the value addition was business training done by seven hubs. Interestingly, hubs that provide co-working space [HME, HNG, HVC, and HWO] as one of their services connected their start-ups to investors and offered professional guidance through legal advisory and business talks.

The majority of hubs (8/10) carried out various internal activities (evidenced by 14 dif-

Table 5.3. Hub incentives and resources

Themes	Incentive types	Hubs
Business enablers	free/serviced working space	HRA, HME, HWO, HCH, HMI, HNG, HIN
	internet connectivity	HNG, HWO, HVC, HRA, HMI, HIN
	digital marketing	HME
	access to technology incentives	HOU
Domain expert incentives	linking the start-ups to experts/talks from experts	HRA, HIN
	legal services	HVC, HCH
	patent services	HCH
Business growth incentives	cross-cutting training (business, product development, branding, financial)	HVC
	seed funding	HRA
	grant competition	HSA
	assistance in grant/funding applications	HCH
	entrepreneurship skills training	HMI
	free start-up business registration	HVC
	access to technology credits	HOU
	access to investor panels	HMI
	discounted business services/partner discounts (legal, accounting, business coaching)	HNG, HME, HVC
	access to potential customers, suppliers	HNG
	connection to investors and potential partners	HVC
	solution/market validation	HME
	subsidized costs to start-ups	HIN
	mentorship	HME, HMI, HOU, HRA, HSA, HWO, HIN, HVC
	equipment financing	HSA
Social incentives	alumni inspiration talks	HOU, HIN
	free access to internal events	HOU
	publicity of start-ups	HVC
	start-up conferences/innovation festivals	HCH
	in-house affordable restaurant service	HIN
	flexible office relations/free interactions	HNG, HWO

ferent codes), with only business matchmaking being the most shared by the four hubs [HCH, HME, HNG, HIN]. Hubs HWO and HSA did not define any processes/activities, possibly due to their short existence of 2 years; hence the possible lack of ingrained or routine identifiable operations. All the hubs identified a focus, with the majority of hubs (7/10) [HRA, HCH, HMI, HSA, HIN] focusing on incubation of start-ups and supporting innovations [HVC, HRA]. The purely co-working spaces [HME, HWO, HNG] focused mainly on entrepreneurship support.

Hub incentives and resources

Table 5.3 highlights the incentives and resources that hubs provide to their start-ups in East African hubs. Early-stage start-ups are usually attracted to hubs by the resources and incentives with the hope of leveraging them to grow into viable businesses. The majority of the hubs save for HMI and HCH are privately owned and thus are potentially profit-driven. We, therefore, cannot assume that the hubs provide the same incentives and resources to start-ups.

Hubs provided the majority of their incentives as business growth incentives (15 of 28 incentives) in comparison to other incentives. The others include social incentives (6/28), business enablers (4/28) that mainly included serviced working space provided by (7/10) hubs and internet connectivity provided by 8/10) hubs except for HSA (which did not physically host hubs) and domain expert incentives (3/28). The mentorship was the most common business growth incentive provided by (8/10). Hub HNG notably did not provide mentorship, possibly due to its strict co-working space model that minimized interference

Table 5.4. Effects of hub incentives on the start-ups

Themes	Effects	Hubs
Business effects	building social capital/business networks	**HVC, HMI**
	more confident in attracting investors	**HCH**
	development of new or spin off ideas	**HSA**
	feedback from actual users	**HOU**
	more start-ups are succeeding in the market/survival rate improvement/improved start-up growth rates	**HVC**
	start-ups save on operational costs/cost cutting	**HVC, HME**
	reporting routines slow down work for start-ups	**HRA**
	attain financial and business maturity	**HRA**
	growth of ideas into start-ups	**HOU**
	learn to develop for right customers	**HVC**
	more start-ups are raising funds	**HVC**
	running many programs lead to loss of focus by founders	**HMI**
Organisational effects	improved/increased social interactions	**HWO, HOU**
	networks expose them to initiatives	**HCH**
	more collaboration through sharing	**HNG**
	appreciate what it takes to grow/change in mindset	**HIN**
	start-ups depend on each other	**HNG**
	learning through networking	**HWO, HMI**
	start-ups grow through the different stages as they hit objectives	**HNG**
Effect on skills	improvement in their skills e.g accounting, financial management	**HME, HVC, HRA**
	start-ups learn different ways of doing things	**HIN**
	founders learn business skill	**HMI**

in the hosted businesses. Only (3/10) hubs offered discounts on business services as a growth incentive. Business growth incentives also showed the most variation amongst the hubs' incentives, with 15 different incentives.

Effects of incentives on the start-ups

We also asked hubs what effects the incentives and activities have on their start-ups. Table 5.4 shows that hubs noticed mostly business effects (12/22) and to a certain extent organizational effects (7/22), with few skills effects (3/22) seen in the start-ups they host. Hub HVC observes the most business effects (6/12) compared to the other hubs that averagely observed 1 to 2 effects. It is also interesting to note that hubs HNG and HWO observed only organizational aspects of their start-ups.

Hub acceptance criteria for start-up

The hubs that were not purely co-working spaces mainly pre-selected for the start-ups from the various sectors (see Table 5.5) software being one of the domains. The participants included innovators and start-ups that were are externally incubated or at post-incubation stages. The majority of the start-ups came from different sectors, as evidenced by (6/10) hubs [HWO, HVC, HME, HNG, HIN, HVC]. The hubs pre-selected the start-ups by letting them apply online, register, or participate in challenge-based calls. The hubs then applied a selection checklist to chose amongst the software start-ups. The hubs that offered only co-working space [HWO, HME, HNG] as a service and the few hubs [HVC, HIN, HCH] that offered other services were not sector-specific to the start-ups they onboarded. Some hubs also onboarded start-ups at different stages of their growth [HME, HRA], at different times, and some assessed by panels [HCH, HRA, HME] and later signed contractual agreements [HMI, HWO].

The results showed that most hubs shared no standard criteria for selecting start-ups, as

Table 5.5. Acceptance criteria for start-ups in hubs

Themes	Codes	Hubs
Pre-selection	start-ups walk-in and freely register	HVC
	request for proposals from innovators	HRA
	proof of concept	HSA
	free start-up registration	HVC
	challenge based calls/run a call	HRA, HSA
	apply online	HCH, HMI, HNG
	technology related start-ups	HOU, HVC
Participants	innovators and start-ups	HRA
	post incubation stage start-ups/externally incubated start-ups	HMI
	sector agnostic/open to all businesses	HWO, HVC, HME, HNG, HIN, HCH
Selection checklist	commitment to participate in activities e.g mentorship sessions	HVC, HMI
	potential to scale	HRA
	assess idea feasibility	HMI
	team knowledge of start-up needs	HME
	validated problem/solution fitting	HRA, HIN
	solution/market need assessment	HMI
	potential to change lives/impact it creates	HCH, HRA
	team composition	HIN, HRA
	pick teams to incubate from participants	HVC
	passion	HCH, HOU
	appropriate business model	HCH, HRA
	originality of the idea	HCH, HRA
	entrepreneurial mindset	HSA
	legality of the business	HWO, HOU
	variation in team skill set	HIN
	ability to pay fees	HOU
	relevant ideas in any five key vertical areas	HVC
	random selection of signed up entrants	HSA
Onboarding	hold an assessment panel/run diagnostic panel	HCH, HRA, HME
	business join at different times	HNG
	innovators sign acceptance form	HSA
	contractual agreement	HMI, HWO
	start-ups join at various stages	HME, HRA

evidenced in the variation of the selection checklist that had (18/34). However, on average, the hubs have more than one aspect they use to select their start-ups. For example, hub HVC picks teams to incubate from participants in its events and checks to see that the start-ups fit in any of the five vertical areas it supports and ensures that start-ups can commit to participating in the incubation activities. On the other hand, hub HRA looks at the start-up's potential to scale and change/impact lives, its business model's appropriateness, the originality of its idea, and its validated problem/solution fitting.

Hubs work with alumni start-ups

We examined how the hubs work with previously hosted start-ups. Table 5.6 reveals that almost all hubs provide some incentives to their alumni start-ups. A notable exception is hub HNG, which does not provide any incentives. Hubs HMI and HIN provide the most incentives to their start-ups (4 incentives each), and hub HRA provides three incentives. To a small extent, alumni start-ups also provide some value to the hubs includes providing some shareholding in their start-ups to the hubs like HVC, helping in the mentoring of current start-ups in the hub [HCH], holding talks to inspire early-stage software start-ups that occupy the hubs and facilitating events and workshops [HNG] in the hubs. However, only (4/10) hubs recognize the value that alumni start-ups provide to them. However, there seems to be no common communication strategy by the hubs to their alumni 6/10 hubs all have different approach.

Table 5.6. Hubs work with the alumni start-ups

Themes	Codes	Hubs
Communication activities	maintain alumni database and filter start-up profiles	HME, HIN
	emailing alumni	HOU
	regular contact and followup of alumni	HCH
	put start-ups in contact with investment clubs	HSA
	reference point for communication with alumni	HRA
Incentives to alumni	subsidized external space	HMI
	post incubation accelerator program	HMI
	facilitate alumni meetings with potential scaling and funding partners	HRA
	mentor-ship (venture advisory, catalyer program)	HMI, HSA
	repayment grace period for start-ups	HMI
	continuous training(business, pivoting, marketing)	HVC
	proposal writing for scalable innovations for scaling partners	HRA
	offer facilities for business networking and interactions	HWO
	matching partners with alumni start-ups of interest	HIN
	extended membership to alumni	HIN
	present opportunities to alumni	HOU, HVC, HIN
	sharing experiences with current start-ups	HIN, HRA
	matching alumni with events of interest	HME
Alumni value to hub	equity in the alumni start-up	HVC
	alumni mentoring inhouse start-ups	HCH
	inspirational talks to start-ups	HOU, HVC
	alumni outreach for events, workshops	HNG

5.3.2 RQ2: Hub measurements for the progress of start-up

Table 5.7 presents the measurements hubs use to track their software start-ups' progress. The hubs mainly track business metrics (9/10 hubs) and scalability metrics (7/10 hubs). The most used scaleability metric was the number of employees/jobs created by hiring the employee base's people/growth. This metric was tracked five hubs [HME, HSA, HVC, HOU, HWO]. Revenue raised was the most common business metric tracked by four hubs [HOU, HVC, HCH, HSA]. Interestingly, each of these four hubs provides incubation as one of their main services.

To a small extent, other metrics include team metrics (5/29 metrics tracked by four hubs), funding metrics (3/29 metrics tracked by three hubs), and product line metrics (3/29 metrics tracked by four hubs). Hub, HVC tracks the most number of metrics (11/30) that spread out in all the five metric themes. Other hubs that track at least 4 of the five different types of metrics include HIN and HMI. These two hubs track a small number of metrics considerably compared to hub HVC. In the co-working space, HNG did not track any metrics about its start-ups, possibly because of its operational model. For example, HNG had this to say about measuring:

> *"Again, because we are a co-working space, we do not actually have a way to quantify or measure. [...] It's not the core of our business. [...] The information we need to know is that are we doing our job for the start-ups, rather than what is the revenue or the profit's, that not helping our business. If you are an incubator, then it would be beneficial to stand up and track, but as a co-working space, it's not so important for the business to understand that, so we do not really take much interest in knowing the information."* [HNG]

5.4 Discussion of Results

This section discusses the key results from the study of operations and measurements in hubs for software start-ups.

Table 5.7. Hub metrics for tracking progress of start-ups

Themes	Metrics	Hubs
Scalability metrics	customer reach	**HOU**
	number of market penetrated	**HIN**
	start-up impact	**HVC, HIN**
	number of employees/number of people hired/number of jobs created/growth of employee base	**HME, HSA, HVC, HOU, HWO**
	product acceptance by the market/product uptake	**HCH, HVC**
	new areas start-up product reaches/start-up growth in market/recognition of new products	**HVC, HIN**
	start-up affinity for professional services (e.g legal,accounting)	**HVC, HWO**
	transitioning from shared space to private offices	**HVC, HIN**
Business metrics	number of customers acquired	**HMI**
	number of business partnerships/business deals created	**HME, HIN**
	growing user/customer bases/number of customers acquired	**HVC, HMI**
	onboarding investors or venture capitalists/number of partners working with start-up	**HWO, HVC**
	number of users willing to buy the product	**HRA**
	returns filled	**HMI**
	revenue raised (turnover)	**HOU, HVC, HCH, HSA**
	time a start-up survives after exit	**HOU**
	business outgrowing the hub	**HWO**
	launching of products	**HWO**
Team metrics	increase in number of active roles/team growth	**HVC, HME**
	ability to employ more staff	**HCH**
	employee turnover	**HME**
	attendance of start-ups to programs	**HMI**
	relevance of hosted events to a start-up	**HME**
Funding metrics	number of support grants got/won	**HIN**
	funds burn rate	**HMI**
	amount in grants	**HVC**
Productline metrics	development of new or spin off products/services	**HVC, HIN**
	measure stage output (e.g, prototype a minimum testable product)	**HMI**
	problem/solution refining at ideation stage	**HRA**

5.4.1 Hub character

A notable number of hubs provided a combination of incubation, acceleration, incubation, and co-working space as services to the start-ups. Most East African hubs' combined services could imply that the resources and incentives provided are similar for early-stage start-ups. The combined services in each hub would also help start-ups to realize their technologies in the incubation phase and evolve into the acceleration phase in one place. Therefore, the hubs serve their primary purpose of enabling the start-ups to grow and bring their products to the marketplace [68] all in one place. The co-working space service in a good number of hubs could also point to a profitability motive where the start-ups that have been incubated and accelerated finally pay rent and services, thus sustaining the hubs in the long run.

5.4.2 Hub operations

Hubs that provided a purely co-working space as a service exercised a culture of respect to privacy for the software start-ups they hosted. This privacy could imply the general lack of closer supervision for the hosted businesses within the co-working spaces. This lack of closer supervision could also be because co-working spaces are sector agnostic to businesses they host and may not have standard programs for the software start-ups and other SMEs they provide. Future work could ascertain how the operations in purely co-working space

hubs influence early-stage software start-ups' growth.
The prevalence of value addition in hubs that include popular business training and the internal activities that argue the value addition like matchmaking start-ups with opportunities is not entirely surprising. The activities could potentially be supporting the incubation and acceleration services in the hubs. These value addition and related internal activities enable the start-ups to create viable products during incubation. They also ensure that the start-ups push the products to their markets during the acceleration.
Completing an incubation-acceleration cycle could be an indicator of success for the hubs that provide these combined services. The co-working space service could be an approach that keeps the accelerated start-ups to pay for the often subsidized space and keep using the resources available to them (these usually are still priced slightly below the market rates).
The results indicated that hubs that provide co-working space as one of the services connected their start-ups to investors and offer professional activities like legal advisory and business talks, pointing to a potential differentiating character of co-working space as a service in hubs. The co-working space service may provide long-term sustainability to the hubs since it may guarantee long term financial value through rental fees and payment for more advanced but premium services as the start-ups mature.
There was variation in the internal activities hubs potentially due to the absence of shared best practices and known standard activities for incubating, accelerating, and co-working services in East Africa. This result may be surprising given that these hubs may provide common incentives, resources, and focus to the software start-ups as seen in table 5.3.

5.4.3 Hub incentives and resources

The majority of hubs provided business enablers like serviced working space with internet connectivity. These enablers may be key incentives to software start-ups to hubs potentially only differentiated by their quality and pricing in hubs. The mentorship was the main business growth incentive, and this is not surprising given that hubs, even in developed ecosystems, are known to provide mentorship [136]. Hubs showed marked variation in business growth incentives, indicating that these incentives are offered as a competitive advantage by the hubs. The purely co-working spaces HNG and HME surprisingly provided professional business services like legal, accounting, and business coaching is usually seen in accelerators [136]. This service in these two hubs could point to an extended business model that aims to sustain these co-working spaces by adding another revenue stream besides rent.

5.4.4 Effects of incentives on the start-ups

Hubs mostly noticed business effects and, to a reasonable extent, organizational effects of incentives on their start-ups. This result may not be surprising given that most hubs provide common business enablers and various business growth incentives, as indicated in earlier results. Interestingly, HNG and HWO observed only organizational effects, potentially pointing to the hub culture of non-interference in the start-ups. They thus only observed the start-ups from social interactions as evidenced by both indicating that they had flexible office relations/free interactions as one of their social incentives. These two hubs are thus unlikely to witness internal effects that impact the start-ups' business and skills.

5.4.5 Hub acceptance criteria for start-ups

Most hubs that offered incubation as one of their services had a pre-selection phase (as described by [137] as a stage for evaluating the start-ups' potential for the start-ups before onboarding. The result is not surprising given that the resources like space usually limit the hubs. Mentors will want to onboard start-ups with a high chance of becoming a sustainable business. This result contrasts with that of hubs that offered mainly co-working space as service. For example, co-working spaces [HWO, HNG] did not have an existing criterion for pre-selecting the start-ups. The co-working spaces' business model may be simply profit-driven on a first-come, first-serve basis so long as a start-up can be the fees, and because they never attract start-ups in cohorts like incubators and accelerators [138].
The start-ups' selection list varied amongst all the hubs indicating the lack of standard metrics to judge the potential for start-ups' success before they are onboarded. The varied criteria may also indicate the need for hubs to objectively select start-ups considering different aspects to gauge their potential for success.

5.4.6 Hubs work with alumni start-ups

Most hubs provided some incentives to their alumni start-ups, possibly because they provide incubation support to the start-ups in their early stage. After this phase, hubs can provide value to start-ups after formal incubation or acceleration, potentially for future mutually beneficial engagements. As evidence, some of the hubs maintained alumni databases that they could use to reach out to the alumni.
Noticeably mature hubs [HVC, HCH, HOU, HNG] also acknowledged their alumni start-ups' value. This value is possibly from their successful start-ups or start-ups that have binding terms to the hubs. For example, accelerators typically have a stake in the start-ups by holding equity.

5.4.7 Hub measurements for the progress of start-up

The hubs mainly tracked business and scalability metrics. The most popular of these metrics included employees/jobs created by hiring people/growth of the employee base and the revenue raised metric. The popularity of that scalability metric in hubs is surprising, given that early stage software start-ups mainly focused on developing a viable product to take to markets and evidenced by the presence of scalability metrics focusing on product and business. These business metrics' popularity may indicate the hubs' lack of interest in the technical metrics that early start-ups measure to track their internal growth. On the other hand, hubs' may be interested in the business metrics because they can see and potentially influence them through training and mentorships.
The hubs noticeably tracked a few start-up team metrics, funding metrics, and product line metrics. This may be surprising given that most hubs provide incubation as one of the services. Incubation activities would generally have a strong focus on developing a viable software product and thus should influence the tracking product line metrics. However, product and team related metrics may be of more interest to the early stage software start-ups than hubs as start-ups use them to track their internal progress.

5.5 Threats to Validity

While case studies are hailed as the most suitable method of exploring a phenomenon in a real-life context, inherent problems are still associated with it, and our study is not spared of these. In some interviews, the hub managers were hesitant about divulging much information about how they work with the start-ups. We encountered this when we first tried to interview the hub manager for one of the branches of hub HNG. However, she referred us to the overall manager in the parent hub, whom we successfully interviewed.
The chosen study design may have weaknesses in the case selection, data collection, data validation, case synthesis. The study design may raise issues about the study credibility, dependability, transferability, and confirmability [139]. We describe some of the issues that may have arisen during the study that may have affected its trustworthiness and how we mitigated them

5.5.1 Internal validity

During the coding process, there was a risk of coding substantial texts to have various meanings of the same text. However, coding snippets of text could result in fragmentation, thus losing the text's meaning during the coding and abstraction process [139]. We thus coded text between 1-3 lines to preserve coding meaning integrity and reviewed the codes to reach consensus on the meanings to mitigate this pitfall. However, the content analysis approach we used is known to fall short in answering the 'why' questions [139]. We referred to related published studies to provide plausible reasons for some of those emerging observations from the codes as a complementary way to enrich the analysis to mitigate this shortcoming during content analysis.

5.5.2 External validity

The hubs were domain agnostic and thus hosted software start-ups and other types of start-ups. They at times offered a combination of incubation, acceleration, or co-working space. This study focused on operations of the hubs concerning software start-ups. The hubs hosting other start-ups and SMEs could have led to a bias in explaining the hub operating models. Thus our results may reveal differences when subjected to hubs specifically hosting software start-ups. However, studies in the developed ecosystems indicate that specialized hubs are more of an exception than the norm. We thus take our results to be representative of a typical hub.

5.6 Conclusion and Future work

Answering RQI provides us with a better understanding of how innovation hubs and co-working spaces work, thus enabling the exchange of best practices amongst hubs within the East African ecosystem. This question will also enable the hub to understand the most significant value-adding activities, resources, and incentives to enable more sustainable start-ups. Future work may be needed to identify the motivation behind particular incentives, resources, and operations in hubs. All the hubs aim to create sustainable software start-ups. Answering RQ2 will enable different hubs to track and measure relevant metrics used by more successful hubs in the region. Future work will investigate how the hubs use metrics to assess and improve their operational and business practices.

Chapter 6

Paper 3: Measurements in the Early Stage Software Start-ups: A Multiple Case Study in a Nascent Ecosystem

Chapter Overview

Software measurement is crucial to stay competitive and deliver quality software products. While much research has been done on measurement in large companies in developed countries, there is limited research on measurement in start-ups. There are no studies on whether these results apply to nascent ecosystems, such as those in East Africa. This study aims to understand the use and perceived benefits of measurement in software start-ups in East Africa. Here, we performed a multi-case study on 19 software start-ups in hubs in Uganda and Kenya through conducting semi-structured interviews. We transcribed and analyzed them using the content analysis technique. In our results, we identified that start-ups are using several business metrics and product-oriented metrics. Furthermore, we found no evidence of the use of design-oriented metrics. Nonetheless, start-ups have expectations for the benefits of measuring. Finally, metrics found in this study partially differ from metrics used in start-ups in developed countries. Therefore, there is a need to create a more inclusive characterization for measurement as early start-ups in East Africa cannot yet be represented with known models.

6.1 Introduction

Measuring is a crosscutting activity within the software life cycle. As found by Staron et al. [1], measurement activities affect everything from the technical to business aspects of software companies. Measurement activities provide results that give insight, support decision-making, or provide actionable alerts [1] into companies' technical and business aspects. Especially in large software companies, established measurement programs can be found [1]. However, while most research on measurements is done in large companies, little is known about measurement in software start-ups [31].

Software start-ups are newly created firms developing software-intensive products or services [4]. These start-ups are usually disruptive organizations operating under uncertain conditions and face numerous challenges to grow and succeed as mature businesses. Studies have shown that many start-up ecosystems are experiencing failure rates of up to 90% [4] of their software start-ups in the first two years [17]. Thus, it is crucial to help founders

to understand and measure the state of their start-ups. As Croll and Yoskovitz [2] put it: *"If you measure better, you are more likely to succeed"* [2]. Providing better support for start-ups is especially crucial in emerging regions, such as East Africa.

According to the maturity model for so software start-ups proposed by Cukier et al. [17], the East African ecosystem is in the nascent/emerging phase. This phase is characterized by already existing start-ups, few investment deals, optional government initiatives to spur the ecosystem's development, and no relevant output in terms of jobs or worldwide penetration. However, to the best of our knowledge, the existing research on start-up measurements is in ecosystems can be considered evolving, mature, or self-sustaining[17].

This paper investigates how and what software start-ups in East African measure and the benefits of measurement they perceive and experience. We compare the results to the work of Croll and Yoskovitz [2] who propose measurements for software start-ups in more mature ecosystems. The remainder of this paper is organized as follows. In Section 2.10.3 we discuss related work. Section 6.2 presents the methodology used in this study. Section 6.3 reports the results of the multi-case study. Section 6.4 presents discussions that answer the research questions. We summarize our results and discuss future work in Section 6.6.

6.2 Research Methodology

To learn about the measurements in East African start-ups, we performed an exploratory case study [115] that involved conducting interviews for start-ups located in different hubs in Uganda and Kenya. The main motivation for this method was that case studies, unlike experiments and survey strategies, allow the study of phenomena with unclear or unknown extent and boundary in their context [115]. Performing an exploratory study allows us to learn more about these boundaries, i.e., what comprises measurements in East African software start-ups. Our research design follows the guidelines suggested by Runeson and Höst [116] for conducting and reporting case studies.

6.2.1 Research questions

To investigate the assessment and measurements within start-ups, we raise and answer the following research questions.

- RQ1: What are software start-ups in East Africa (a nascent ecosystem) assessing and measuring?
- RQ2: What benefits of measurements do the start-ups perceive?
- RQ3: How do metrics used in East African start-ups compare to those in lean start-ups in developed countries?

We define an assessment as an activity that potentially leads to a measure and a measurement as a qualitative or quantitative value used for insight, decision making, or alert within a start-up.

Data collection: We conducted semi-structured interviews with start-up founders. The semi-structured interviews allowed us to steer the conversation in the study's direction while providing the freedom to explore upcoming topics during the interview. Thus, this method provided us with the potential to discover unknown and unexpected aspects of measurement within early-stage software start-ups.

We grouped the interview questions into three themes, namely (1) start-up challenges, (2) start-up growth milestones, and (3) measurements in start-ups. All questions were independently reviewed by the co-authors and refined in several iterations to ensure that they are understandable and cover the aspects relevant to this study. We firstly conducted four interviews in Uganda and refined a few questions. We had three themes for the interviews, but in this paper, we focus, analyze, and report the measurement theme for the early-stage software start-ups in East Africa. Each interview lasted on average 45 minutes. We conducted each interview for at most 45 minutes, recorded, and transcribed it.

Case and subject selection: To get in contact with start-ups for the study, we approached hubs in Kampala, Uganda, and Nairobi, Kenya using convenience sampling [140]. These cities have most of the active hubs in Uganda and Kenya - probably, because of the high concentration of universities (where young developers come from) and the availability of quality amenities and internet infrastructure. We contacted hub managers and later asked them to link us to the start-ups in their hubs. This contact ensured that the choice of start-ups was not driven by us but by the hub managers. When a start-up agreed to participate, we scheduled an interview with one of the active founders. Interviewing active founders was motivated by knowledgeable founders' that knew about the start-up's past and current operations.

We approached three hubs in Kampala that had a previous collaboration with Makerere University. We refer to the hubs as R, O, I to maintain their anonymity. The two hubs, R and O, are incubation hubs, R incubating any type of start-up, and O mainly incubating software start-ups. Hub I is a hybrid (a co-working space and an incubation hub) but predominantly targets software businesses. In Kenya, we made contact with hubs with the help of a local researcher. After initial contact with six hubs, three agreed to participate. We refer to these hubs as C, W, and N. Hub C is an incubation hub for all types of start-ups, while W and N are co-working spaces. Thus, we visited six hubs in total, which included two co-working spaces (W, N), three incubation spaces (R, O, C), and one hybrid hub (I). We initially carried out 23 interviews in all the six hubs. In hub I, we interviewed ten start-ups because it is the largest hub in Kampala, Uganda, two from O, four from R, four from C, one from N, and two from W. We later excluded four interviews from these interviews (one from hub I and three from hub R) for this study during the analysis phase. It turned out that they were not developing software or using any software in their business. We interviewed fewer start-ups in Kenya compared to Uganda since we were there for only one week. Table 6.1 characterizes 19 start-ups included in this study. We indicate the type of the hosting hub (incubation hub, co-working space, or hybrid), the start-ups' age in months, the start-up type, and whether the start-up is measuring. We use codes to represent the start-ups and hubs to maintain their anonymity. Table 6.2 summarizes the age structure of our start-ups.

Analysis: We imported all interview transcripts into Excel spreadsheets. In each excel sheet, we organized each question's responses under the corresponding question in the interview guide. The organization allowed for tracking of the emerging codes and quotations in the raw transcripts. We used thematic coding that involved reading through each transcript, line by line, to derive the emerging codes. Runeson and Höst [116] point out that this analysis is a suitable approach for analyzing software engineering case studies. After the first round of coding, we grouped the emerging codes from the 19 transcripts, resulting in high-level groups that included, "progress assessment", "what to measure", "benefits of measurement", "use of the measurement dashboard", and "benefits of using the dashboard". We identified matching start-ups for each emerging code by examining each start-up transcript

Table 6.1. Characterization of Interviewed software start-ups (Uganda = UG, Kenya = KE)

No	Hub ID	Start-up ID	Host Country	Hosting Hub Type	Age (Months)	Start-up Type	State of assessment/ Measuring
1	**C**	CB	KE	Incubation Hub	24	Software as a Service	Measuring
2		CZ	KE	Incubation Hub	11	Software as a Service	Measuring
3		CES	KE	Incubation Hub	12	Software as Service	Measuring
4		CE	KE	Incubation Hub	10	Mobile App	Measuring
5	**R**	RI	UG	Incubation Hub	51	e-commerce	Measuring
6	**O**	OI	UG	Incubation Hub	48	Using software in their workflow	Measuring
7		OC	UG	Incubation Hub	15	Fintech	Measuring
8	**W**	WT	KE	Co-working space	10	Two-sided marketplace	Measuring
9		WE	KE	Co-working space	12	Two-sided marketplace	Measuring
10	**N**	NF	KE	Co-working space	43	Software as a Service	Measuring
11	**I**	IH	UG	Hybrid	7	e-commerce	Measuring
12		IY	UG	Hybrid	27	User-generated content	Measuring
13		IT	UG	Hybrid	14	Software development	Measuring
14		ID	UG	Hybrid	12	Media Site	Measuring
15		IA	UG	Hybrid	9	Software as a service	Measuring
16		IR	UG	Hybrid	8	Mobile App	Measuring
17		IN	UG	Hybrid	15	Fintech	Measuring
18		IP	UG	Hybrid	32	Two-sided marketplace	Measuring
19		IS	UG	Hybrid	39	Software as a service	Measuring

Table 6.2. Start-up age characterization

Age Bracket (Months)	Start-ups
0-6	None
7-12	CZ, CES, CE, IH, ID, IA, IR, WT, WE
13-18	IT, IN, OC
19-24	CB
25-30	IY, IP
31-36	IS
>**36**	RI, NF, OI

in a related interview question. We reviewed the grouped codes to clarify the ambiguous ones, merge, or split them. Afterward, we performed a second round of coding, using the merged codes. With the help of codes, we analyzed and discussed the interviews to identify and characterize measurements used and benefits perceived by the start-ups. To analyze and discuss the results emerging from the coding, we make use of parts of MeSRAM [1]. This method is one of the most complete and recent approaches to assessing large software firms' measurement programs. So far, it has been applied in large software companies. However, MeSRAM's aspect "metrics used" provide sub-categories of metric types (business metrics, product metrics, design metrics, organizational performance metrics, and project metrics), which are also applicable to software start-ups. We apply these sub-categories to systematize the measurements from the codes' synthesis from the start-up interviews in this study.

To gain further insight into the use of measurements and their benefits in early-stage start-ups, we performed a simple cross-case analysis [141] to identify whether variables such as start-up age, hub type, and country affect 1) how many metrics are used, 2) how many metrics are wished for, 3) how many benefits are experienced, and 4) how many benefits are expected by the start-ups. Since the number of data-points is limited, we decided to group the start-ups for each variable we studied so that the gained groups are meaningful and not too small for statistical testing. As a result we work with the following groups:

Variable 'start-up age' (3 groups): young (0-12 months, n = 9); medium (13-24 months, n = 4); old (≥ 25 months, n = 6)

Variable 'hub type' (3 groups): Incubation hub (n = 7); Co-working space (n = 3); Hybrid (n = 9)

Variable 'country' (2 groups): Kenya (n = 7); Uganda (n = 12)

We do not consider the start-up type as a variable here since we have, for most types, only one or two start-ups, which is not enough to form a representative sample per type.

For the test, we used the Wilcoxon Signed-Rank Test [142] to test the probability that two samples belong to the same population. Furthermore, the Wilcoxon test can be applied to data with a normal distribution and not normally distributed data. We focus on the following null hypotheses:

- $H0_1$: The studied variables have no impact on the number of measurements used by a start-up.
  - $H0_{1a}$: The start-up age has no impact on the number of measurements used by a start-up.
  - $H0_{1b}$: The hub type has no impact on the number of measurements used by a start-up.
  - $H0_{1c}$: The country has no impact on the number of measurements used by a start-up.
- $H0_2$: The studied variables have no impact on the number of measurements wished for by a start-up.
  - $H0_{2a}$: The start-up age has no impact on the number of measurements wished for by a start-up.
  - $H0_{2b}$: The hub type has no impact on the number of measurements wished for by a start-up.
  - $H0_{2c}$: The country has no impact on the number of measurements wished for by a start-up.
- $H0_3$: The studied variables have no impact on the number of benefits experienced by a start-up.
  - $H0_{3a}$: The start-up age has no impact on the number of benefits experienced by a start-up.
  - $H0_{3b}$: The hub type has no impact on the number of benefits experienced by a start-up.
  - $H0_{3c}$: The country has no impact on the number of benefits experienced by a start-up.
- $H0_4$: The studied variables have no impact on the number of benefits expected by a start-up.
  - $H0_{4a}$: The start-up age has no impact on the number of benefits expected by a start-up.
  - $H0_{4b}$: The hub type has no impact on the number of benefits expected by a start-up.

  - $H0_{4c}$: The country has no impact on the number of benefits expected by a start-up.

Besides, we investigate whether the number of benefits experienced is related to the number of measurements used, the number of measurements wished for, or the number of benefits expected:

- $H0_5$: Whether experienced benefits do not correlate with the number of benefits expected, measurements used, or measurements wished for by a start-up.
  - $H0_{5used}$: Whether benefits experienced do not correlate with the number of measurements used by a start-up.
  - $H0_{5wished}$: Whether benefits experienced do not correlate with the number of measurements wished for by a start-up.
  - $H0_{5expected}$: Whether benefits experienced do not correlate with the number of benefits expected by a start-up.

Finally, we compared the measurements in early-stage software start-ups in East Africa to the lean metrics presented by Croll and Yoskovitz [2]. For that, we use Croll's categories to group the East Africa start-ups and match the found metrics to the lean start-up metrics described by Croll and Yoskovitz. We discuss the results of the comparison of these metrics in the start-ups and existing lean start-ups in Section 6.3 and 6.4.

6.3 Results

This section presents the study results. It highlights observations on measurements in the East African software start-ups, what measurement benefits these start-ups perceive, and the degree to which the found measurements match those found in Croll and Yoskovitz [2] work.

6.3.1 Assessments and measurements in East African software start-ups

In Tables 6.3 and 6.4 we summarized what assessments and measurements we found in the early stage start-ups in East Africa. Our interviewees mentioned metrics that they use and they would wish to use in the future (probably starting to use them as they grow or as they get resources and capabilities).

Altogether, we found 28 assessments and measurements (also called metrics) from the 19 start-ups. Each of the 19 start-ups had at least one way of assessing progress and some additional measurements they use or wish to use.

To understand these metrics, we categorized them based on the sub-categories in the "metric used" aspect of the MeSRAM model [1]. This model targets the robustness of a companies measurement program and is often applied to a large software company. There are five sub-categories: business, product, design, organizational performance, and project metrics (Staron and Meding [1]). Thus, categorizing the metrics found in start-ups using the "metrics used" sub-categories provides a first idea of how start-up metrics may relate to metrics used in more mature companies.

Table 6.3. Measurements and assessments in start-ups categorized based on MeSRAM[1] , ["-" = No Start-ups, "Y" = Yes, "N" = No]

No	**Metric Category**	**Found Measurements/Assessments**	**Start-ups using the metric**	**Start-ups wishing for the metric**	**Used in Start-ups**
1	Business Metrics	Customer analytic (Number of people using platform, customer behaviour)	OI, WT, IY, IR, IP, NF	WE, IH, CE, ID , IA	Y
2		Product delivery process time estimation (Delivery reliability measurement)	-	IH	N
3		Rate of customer/partner acquisition/-growing customer base	CES, NF	IS	Y
4		Revenue growth/generated revenue/activities that generate revenue	CES, IN	RI	Y
5		Using a telemetry tool	IN	-	Y
6		Tracking market indicators/market events	-	ID	N
7		Ability to close a business deal	CES	-	Y
8		Set and review business targets	IH	-	Y
9		Product awareness/customer interest	IY, IP	-	Y
10		Using market as a benchmark	OC	-	Y
11		Customer feedback measurement	OC	IY	N
12		Reaching key business milestones (patents, tax registration, incorporation)	WE	-	Y

Table 6.4. Measurements and assessments in start-ups categorized based on MeSRAM[1] (contin.), ["-" = No Start-ups, "Y" = Yes, "N" = No]

No.	**Metric Category**	**Found Measurements/ Assessments**	**Start-ups Using the metric**	**Start-ups Wishing for the metric**	**Used in Start-ups**
13	Product Metrics	Product/feature usage	-	IN, IY, CE, OC	N
14		Production process time estimation	IH	-	Y
15		System reliability	-	RI	N
16		Ability to build a complete product	CB, CES	-	Y
17		Feedback from friends about product features (peer endorsement)	CZ	-	Y
18		Product maintenance/support	CES	-	Y
19		Comparing product versions (added features)	CE	-	Y
20	Organizational Performance Metrics	Set and evaluate Key Performance Indicators (KPIs)	IP, ID, WT	-	Y
21		Time-based task setting, tracking and review for progress of project/staff	NF, WT	RI	Y
22		Time-based project performance appraisal	NF	-	Y
23	Project Metrics	Monetary value of time spent on task/activity	-	IT	N
24		Set and evaluate tasks	CZ	-	Y
25		Activity completion time	-	IT	N
26		Process adherence by the team	-	OC	N
27		Tools usage by team	-	WT	N
28		Documenting and reviewing activities for progress	IS, WE, WT	-	Y

Business metrics: 12 of the 28 metrics are business-oriented, and most of the start-ups (17 of 19) are using or wish to use them. Twelve start-ups are already using at least one business metric. Furthermore, eight start-ups wish to use business metrics. For example, customer analytic is a metric that consists of more fine-grained metrics like the number of downloads, number of user accounts, number of people using the platform, or customer

behavior. This metric is the most popular business metric within the interviewed start-ups. Six start-ups (OI, WT, IY, IR, IP, NF) are already using it, and five start-ups (WE, IH, CE, ID, IA) are wishing to use it. Of the 12 business metrics found, ten are already used by different start-ups. Only two metrics are just wished for (Product delivery process time estimation and tracking market indicators/market events).

Product metrics: Product-related metrics are the second most popular metrics, with 7 out of 28. These metrics are used or wished for by nine different start-ups. 5 of the 7 product metrics are also already in use (by five different start-ups). There are, however, two metrics (Product/feature usage and System reliability) that are not used at all but only wished by some start-ups in this category. The product/feature usage metric is wished for by 4 of the nine start-ups but not used.

Organizational performance metrics: The organizational performance metrics category has 3 of the 28 metrics we found in the start-ups. It is the category with the least number of used and wished for metrics in the interviewed start-ups. However, all three metrics in this group are used by at least one of the four start-ups (IP, ID, WT, NF). Only one start-up (RI) wishes to use a metric in this category in the future ("Time-based task setting, tracking and review for the progress of project/staff").

Project metrics: We found six project-oriented metrics from the start-ups. However, in contrast to organizational performance metrics, project metrics are mostly wished for by the start-ups. Only 2 of those six metrics are used by the start-ups. Of the six different start-ups using and wishing to use these project metrics, four are using at least one of them. The most popular used project metric is documenting and reviewing activities for progress (IS, WE, WT).

Design metrics: Most interestingly, none of the start-ups uses or wishes to use design metrics.

Table 6.5. Perceived sufficiency of measuring in start-ups

Perceived sufficiency of Measuring	Start-ups
Enough	NF, IN, IS, IP
Not Enough	CB, CZ, CES, CE, IO, OC, WT, WE, IY, IT, ID, IR
Not Sure	RI, IH, IA

Perceived sufficiency of measuring by start-ups: We also asked whether the start-ups perceived their measuring as sufficient. Table 6.5 and Figure 6.1 summarize the answers. Of the 19 start-ups, four start-ups (NF, IP, IN, IS) affirmatively said that they measured enough, 12 start-ups do not think they measure enough, and three start-ups were unsure whether they measure enough. Interestingly, 3 of the start-ups were not using any measures (RI, IT, and IA). All of those claimed that they are not measuring enough or are unsure whether they measure enough.

To sum up, most start-ups use at least one or more metrics, and most start-ups conceded that they do not measure enough. Our results have shown that there is a wide variation in the used and wished measures. This wide variety may be a symptom of missing best practices for measurement in start-ups. Future research will have to show whether common knowledge about best practices would reduce the visible variety.

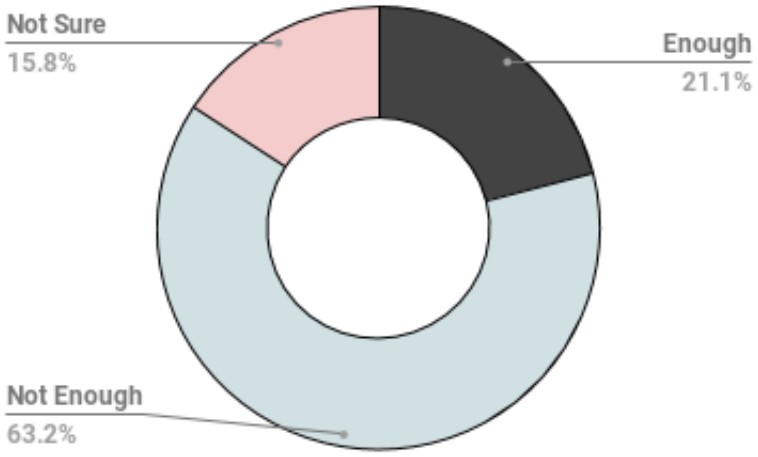

Figure 6.1. Perceived Sufficiency of Measuring by start-ups

6.3.2 Perceived benefits of measurements

Given that some start-ups were measuring and some wishing to measure aspects of their businesses, we found it important to determine what they expected to benefit or experience from measuring. All, but one, of the start-ups clearly stated what they do or would like to use metrics (see Tables 6.3 and 6.4). Table 6.6 lists the benefits that the start-ups are already experiencing or expecting when they measure.

Our results reveal 22 benefits experienced and/or expected by the 19 start-ups. The majority (17 of the 22 benefits) are expected and only 5, are experienced by the start-ups. The five experienced benefits are expressed by three start-ups (CZ, NF, IS), which are all from different hubs in Kenya and Uganda.

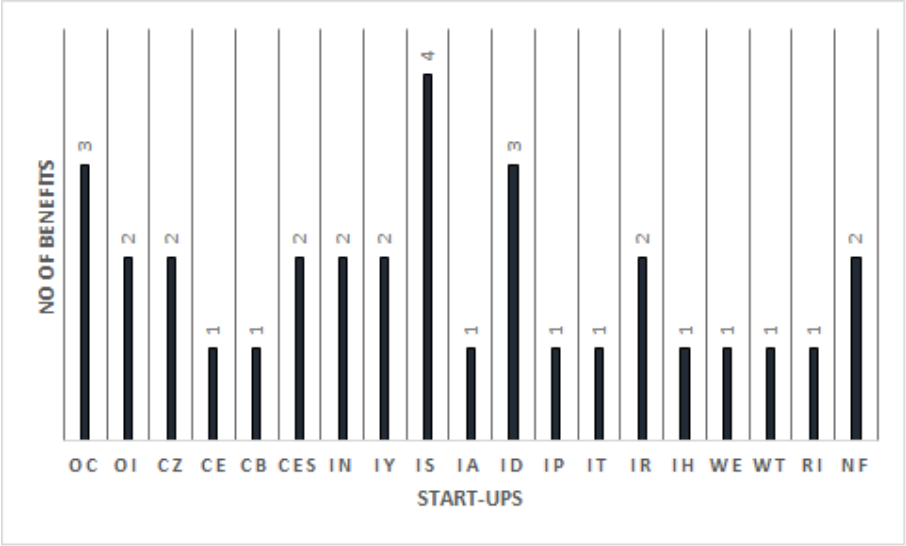

Figure 6.2. Number of Measuring Benefits Expected and/or Experienced per Start-up

Figure 6.2, shows the number of expected or experienced benefits per start-up. A majority of the start-ups (16 of 19) have experienced or expect two or fewer benefits from measuring. There are, however, three start-ups (OC, IS, ID) that expect or have experienced three or more benefits of measuring. OC, a Financial Technology (fintech) start-up of 12 months,

Table 6.6. Perceived Benefits of Measuring in Start-ups

No.	Benefits	Start-ups Expressing Benefit	Experienced or Expected
1	Proof that the product is being used versus just downloading	CE	Expected
2	Reflect on how to earn(activities that generate revenue)/measuring revenue growth	CES, IN	Expected
3	Notice Change (growth)	CB, CES	Expected
4	**Gauging ability to deliver working product/service**	**CZ**	**Experienced**
5	**Profiling growth(areas of growth, comparing growth)**	**CZ, NF**	**Experienced**
6	Know what to focus on (based on success or failure)/where to focus(resources)/direction	OI, RI, IY, IS, IA, ID	Expected
7	Negotiating deals with reliable Information	OI, IS	Expected
8	Detect areas of success/strength and failure/weaknesses	OC, IY, IP	Expected
9	Indicate progress of team members	OC	Expected
10	Show managerial/administrative ability	OC	Expected
11	Provide a history of solutions that can aid solve current start-up problems	WT	Expected
12	It helps in providing evidence for securing start-up funding from investors	WE	Expected
13	**Show areas of positive growth/success**	**NF**	**Experienced**
14	Aids in timely delivery of projects and ensuring value for money	IT	Expected
15	**Determining the team size**	**IS**	**Experienced**
16	**Knowing how to grow and scale**	**IS**	**Experienced**
17	Helps in knowing revenue generating activities in a start-up	IN	Expected
18	Improving product based on measuring usage	IR	Expected
19	Can help in product feature-based pricing	IR	Expected
20	Understanding and optimizing product value chains	IH	Expected
21	Providing insights into operations	ID	Expected
22	Aids in making appropriate resource allocation	ID	Expected

expects measuring to help a) detecting areas of success/strength and failure/weaknesses, b) indicate the progress of its team members, and c) unveil its managerial/administrative ability. They had this to say:

"We shall be able to detect failure early in time, meaning you can see them before they become catastrophic. If you can do that, then you can work on them. They also show you where you are making great success, and within the team, they show you who is moving on well and who is not. It will show you, in terms of management, can someone handle a given area, or a change is required to be able to match our competition"[OC].

Start-up, ID, a 12 months media site, expects measuring to enable it a) to know what to focus on, b) to provide insights into the start-ups' operations, and c) to make appropriate resource allocations:

"We can know where to focus our attention and also know how to plan for our time especially when attending events, you can know who is coming for an event or not from experience. [...] So tracking helps us know and give insights for certain events and their turn up. [...] There is a lot of misallocation of resources, so my start-up will help allocate resources appropriately. Why would I talk about machine learning, yet no one has a start-up in that field?" [ID].

Finally, a 39 months old Software as a service (SaaS) start-up, IS, has experienced two measuring benefits that included are enabling to determine the team size and how to grow and scale. It expects two other benefits from measuring. One is to negotiate deals using reliable information, and the other is to know what to focus on.

"[...] The biggest benefit is that we can know how to grow and how to scale and where to focus our resources. For example, we are not focusing on developing a lot now because we want to first grow to a certain point and then come back and add features. Secondly, it helps us determine the size of our team. Thirdly to make business decisions like should we get someone to invest in us or should we use what is coming in for a longer time" [IS].

The most popular expected benefit is knowing what to focus on (based on the success or failure of operations of the start-up) or where to focus resources, and six start-ups mentioned it (OI, RI, IY, IS, IA, ID). These start-ups come from the three hubs (O, I, R) we visited in Kampala. Following are some example quotes expressing this expectation:

"Strategy: we shall know what to focus on, but also that is key in new markets. Because if you know what to focus on, you don't have to start from scratch. Our measurements will help us understand what we need to focus on and help us in our partnerships. For example, I can tell the courier that for me, I will bring mostly this amount of goods so give me a deal, and it helps us work with the stalls" [OI].

"Well because we are all over the place, you have to spend your efforts in doing the right thing, and the only way you are doing the right thing is if you are measuring. If you have a goal like moving from here to Mbarara, you have to know that you are 50km in, moving at 20km/hr, and you will arrive at 4:00 pm. So, it gives you a chance to know what the right directions are so that you do not go astray and speed towards like the Mabira forest. Measuring is key" [RI].

"We can know where to focus our attention and also know how to plan for our time especially when attending events, you can know who is coming for an event or not from experience" [ID].

Finally, our results also show that 16 of the 22 benefits were mentioned by a single start-up only. Thus, only a few benefits are shared among the start-ups.

6.3.3 Cross-case analysis

Table 6.7 summarizes the data used as input for the cross-case analysis. The Tables 6.8, 6.9, 6.10, and 6.11 below show the results of the Wilcoxon signed-rank test for the different variable groups. We work with a threshold of 0.05 for the p-value.

Concerning hypothesis $H0_1$ and its sub-hypotheses, we compared each pair of groups within the three variables (start-up age, hub type, and country) as shown in Table 6.8. As there are no significant differences between the different start-up ages, we conclude that we cannot reject hypothesis $H0_{1a}$ (the start-up age has no impact on the number of used metrics). A similar result is found when comparing the different hub types. However, when comparing metrics used in co-working spaces and hybrid hubs, we have a p-value below 0.05. In itself, this would be a significant result. However, we have to take into account that hypothesis $H0_{1b}$ is tested three times ($h01b_{ic}$, $h01b_{ih}$, and $h01b_{ch}$), causing a multiple comparison problem (the more often we test a hypothesis, the higher the probability to

Table 6.7. Data by variable groups

Variable	**Group**	**n**	**average # used metrics**	**average # # wished metrics**	**average # experienced benefits**	**average # expected benefits**
Start-up age	young	9	1.89	1.00	1.22	0.33
	medium	4	1.25	1.25	1.75	0.00
	old	6	1.83	1.00	1.33	0.67
Hub type	Incub. hub	7	1.57	1.00	1.43	0.29
	Hybrid	9	1.33	1.22	1.56	0.33
	Co-work.	3	3.33	0.67	0.67	0.67
Country	Kenya	7	2.57	0.57	0.86	0.57
	Uganda	12	1.25	1.33	1.67	0.25
Benefits Experienced	Yes	4	2.00	0.75	-	0.50
	No	15	1.67	1.13	-	1.60

Table 6.8. A Wilcoxon Signed-Rank test for used measurements.

Variable	**x**	**y**	**Hypothesis (used_metrics(x) = used_metrics(y))**	**W**	**p-value**	**Result**
Start-up age	young	medium	$h01a_{ym}$	20.5	0.75	$h01a_{ym}$ not rejected
	medium	old	$h01a_{mo}$	9.5	0.66	$h01a_{mo}$ not rejected
	young	old	$h01a_{yo}$	27	1	$h01a_{yo}$ not rejected
Hub type	Incub. hub	Co-work.	$h01b_{ic}$	3.5	0.12	$h01b_{ic}$ not rejected
	Incub. hub	Hybrid	$h01b_{ih}$	30.5	0.96	$h01b_{ih}$ not rejected
	Co-work.	Hybrid	$h01b_{ch}$	2.5	0.045	$h01b_{ch}$ not rejected (Bonferroni corr.)
Country	Kenya	Uganda	$h01c_{ku}$	60	0.12	$h01c_{ku}$ not rejected
Benefits Experienced	Yes	No	$h05_{used}$	34.5	0.68	$h05_{used}$ not rejected

Table 6.9. A Wilcoxon Signed-Rank test for wished measurements.

Variable	**x**	**y**	**hypothesis (wished_metrics(x) = wished_metrics(y))**	**W**	**p-value**	**Result**
Start-up age	young	medium	$h02a_{ym}$	15	0.68	$h02a_{ym}$ not rejected
	medium	old	$h02a_{mo}$	14	0.74	$h02a_{mo}$ not rejected
	young	old	$h02a_{yo}$	28.5	0.90	$h02a_{yo}$ not rejected
Hub type	Incub. hub	Co-work.	$h02b_{ic}$	11	1	$h02b_{ic}$ not rejected
	Incub. hub	Hybrid	$h02b_{ih}$	27	0.65	$h02b_{ih}$ not rejected
	Co-work.	Hybrid	$h02b_{ch}$	19	0.32	$h02b_{ch}$ not rejected
Country	Kenya	Uganda	$h02c_{ku}$	23.5	0.11	$h02c_{ku}$ not rejected
Benefits Experienced	Yes	No	$h05_{wished}$	23.5	0.53	$h05_{wished}$ not rejected

Table 6.10. A Wilcoxon Signed-Rank test for experienced benefits.

Variable	**x**	**y**	**hypothesis (experienced_benefits(x) = experienced_benefits(y))**	**W**	**p-value**	**Result**
Start-up age	young	medium	$h03a_{ym}$	14	0.39	$h03a_{ym}$ not rejected
	medium	old	$h03a_{mo}$	8	0.28	$h03a_{mo}$ not rejected
	young	old	$h03a_{yo}$	23	0.59	$h03a_{yo}$ not rejected
Hub type	Incub. hub	Co-work.	$h03b_{ic}$	8.5	0.62	$h03b_{ic}$ not rejected
	Incub. hub	Hybrid	$h03b_{ih}$	29.5	0.81	$h03b_{ih}$ not rejected
	Co-work.	Hybrid	$h03b_{ch}$	11.5	0.71	$h03b_{ch}$ not rejected
Country	Kenya	Uganda	$h03c_{ku}$	48	0.51	$h03c_{ku}$ not rejected

Table 6.11. A Wilcoxon Signed-Rank test for expected benefits.

Variable	**x**	**y**	**hypothesis (expected_benefits(x) = expected_benefits(y))**	**W**	**p-value**	**Result**
Start-up age	young	medium	$h04a_{ym}$	12.5	0.41	$h04a_{ym}$ not rejected
	medium	old	$h04a_{mo}$	9.5	0.65	$h04a_{mo}$ not rejected
	young	old	$h04a_{yo}$	24	0.75	$h04a_{yo}$ not rejected
Hub type	Incub. hub	Co-work.	$h04b_{ic}$	15.5	0.27	$h04b_{ic}$ not rejected
	Incub. hub	Hybrid	$h04b_{ih}$	28.5	0.78	$h04b_{ih}$ not rejected
	Co-work.	Hybrid	$h04b_{ch}$	21.5	0.14	$h04b_{ch}$ not rejected
Country	Kenya	Uganda	$h04c_{ku}$	20.5	0.06	$h04c_{ku}$ not rejected
Benefits Experienced	Yes	No	$h05_{expected}$	10.5	0.04	$h05_{expected}$ rejected

get a significant result). To avoid that problem, we perform a Bonferroni correction on the threshold by dividing the threshold by the number of tests, i.e., $0.05/3 = 0.016$. Given that adjusted threshold, $h01b_{ch}$ is not rejected as well. Thus, our data do not indicate any impact of the hub type on the number of used metrics ($H0_{1b}$ is not rejected). Finally, we do not find an impact of the country on the number of used metrics ($H0_{1c}$ is not rejected).

We approached hypothesis $H0_2$ and its sub hypotheses similarly to hypothesis $H0_1$. As Table 6.9 illustrates, neither of the sub-hypotheses can be rejected. Thus, our data do not impact the variables (start-up age, hub type, and country) on the number of metrics wished for by a start-up.

We find the same results when investigating the number of benefits experienced ($H0_3$) and the number of benefits expected ($H0_4$) by the start-ups. None of the sub-hypotheses can be rejected (as shown in Tables 6.10 and 6.11), indicating that the three variables have no impact on the number of benefits experienced or expected.

Finally, we tested hypothesis $H0_5$, by comparing start-ups that experience benefits with start-ups that experience no benefits, with regards to the number used metrics ($h05_{used}$), the number of wished-for metrics ($h05_{wished}$), and the number of expected benefits ($h05_{expected}$) (shown in Tables 6.8, 6.9,and 6.11). The results do not allow us to reject $H0_{5used}$ and $H0_{5wished}$. Thus, it seems that there is no relationship between the number of used or wished for metrics and the experienced benefit of metric use. However, we can reject $H0_{5expected}$, with an average number of expected benefits of 0.5 for start-ups already experiencing benefits and 1.6 for start-ups that do not yet experience benefits. Thus, results indicate a relation between the number of benefits expected and the number of benefits experienced.

6.3.4 Comparison to lean start-ups in developed countries

To better understand how the metrics found in the East African context relate to the other contexts, we decided to compare them to the metrics for lean start-ups described by Croll and Yoskovitz [2] which were collected in the context of developed countries. In Tables 6.12, 6.13, and 6.14, we present the comparison matrix. The lean metrics presented by Croll and Yoskovitz [2], are associated to start-up categories and are more fine grained/specific than our aggregated list in Tables 6.3 and 6.4.

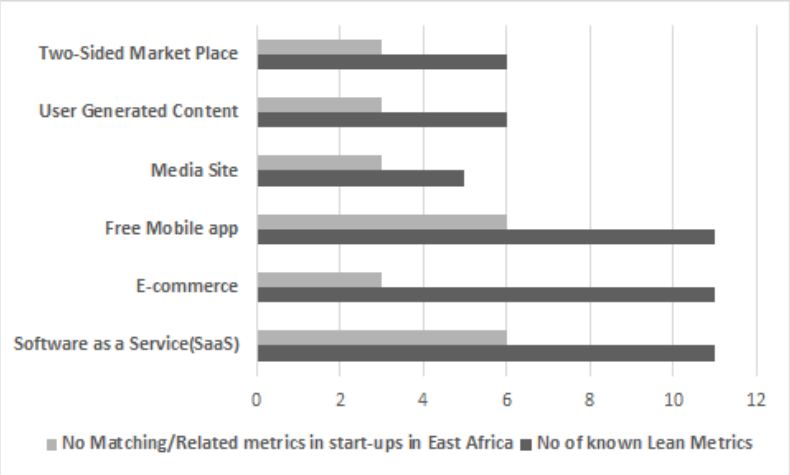

Figure 6.3. Number of Lean metrics and matching/related start-up metrics

Thus, we first mapped our start-ups to Croll and Yoskovitz [2] start-ups categories (shown in column 7 in Table 6.1). In a second step, we matched the metrics used/wished for by the start-ups to the metrics listed by Croll and Yoskovitz. Note that we refined the description of the metrics here to reflect more specifically what happens in the start-ups and enable the mapping. This matching goal was to establish the common metrics and unused metrics in East Africa start-ups compared to known lean start-up metrics in the developing world.

Start-up categories: The categories by Croll and Yoskovitz [2] include e-commerce, software as a Service (SaaS), free mobile app, two-sided marketplaces, media site, and user-generated content. We already categorized the start-ups in Table 6.1 based on these categories. However, it was challenging to fit some start-ups found in East Africa in these categories. IN and OC are better categorized as Fintechs (start-ups that provide payment and financing solutions to customers, using existing telecommunication and banking infrastructure). OI can be categorized as a start-up that uses software as part of its workflow, and IT can be categorized as a software development start-up (developing software solutions for other companies).

Metrics comparison: In Figure 6.3, we illustrate the number of matching/related metrics found in the start-ups in East Africa in comparison to those already known in Lean start-ups presented by Croll and Yoskovitz [2].

Overall, we matched 21 of the metrics found in East African start-ups to the 71 metrics listed by Croll and Yoskovitz. The matches included Software as a Service (SaaS)(6/11), E-commerce (3/11), Free mobile app (6/11), media site (3/5), user-generated content (3/6), and Two-sided market place (3/6). We found matching metrics in the start-ups for at least half of Croll's and Yoskovitz's metrics, except for E-commerce. However, we also found

Table 6.12. A match of Croll's and Yoskovitz's Lean start-up metrics [2] to the metrics found in our study.

Type and Start-ups	**Lean Start-up Metrics**	**Metric Description**	**Matching/Related Lean and Start-up Metrics**
Software as a service (SaaS): CB, CZ, CES, NF, IA, IS	Attention	How effectively the business attracts visitors.	-
	Enrollment	How many visitors become free or trial users.	3 (Rate of user acquisition new users and merchants - IS)
	Stickiness	How much the customers use the product.	1 (Web page usage tracking - NF)
	Conversion	How many users become paying customers and switch to a higher-paying tier.	-
	Revenue per customer	How much money a customer brings in a given time period.	4 (Amount of generated Revenue - CES)
	Customer acquisition cost	How much it costs to get a paying user.	3 (Customer growth-CES)
	Virality	How likely customers are to invite others and spread the word, and how long it takes them to do so.	-
	Up-selling	How often and why customers increase their spending.	-
	Up-time and reliability	Number of complaints, problem escalations, or outages.	18 (Product maintenance - CES)
	Churn	How many users and customers leave in a given time period.	3 (Customer Retention-NF),
	Lifetime value	How much customers are worth from cradle to grave.	-
E-commerce: RI, IH	Purchases per year	The number of purchases made by each customer per year.	4 (Transaction revenue volume - RI)
	Average shopping cart size	The amount of money spent on a purchase.	4 (Transaction revenue volume - RI)
	Abandonment	The percentage of people who don't finish a begun purchase.	-
	Cost of customer acquisition	The money spent to get someone to buy something.	-
	Revenue per customer	The lifetime value of each customer.	-
	Top keywords driving traffic to the site	Those terms that people are looking for, and associate with you—a clue to adjacent products or markets.	-
	Top search terms	Both those that lead to revenue, and those that don't have any results.	-
	Effectiveness of recommendation engines	How likely a visitor is to add a recommended product to the shopping cart.	-
	Virality	Word of mouth, and sharing per visitor.	1 (customer behavior analytic -IH)
	Conversion rate	The number of visitors who buy something.	-
	Mailing list effectiveness	Click-through rates and rates of buyers returning.	-

Table 6.13. A match of Croll's and Yoskovitz's Lean start-up metrics [2] to the metrics found in our study. (contin.)

Type and Start-ups	**Lean Start-up Metrics**	**Metric Description**	**Matching/Related Lean and Start-up Metrics**
Free Mobile App: CE, IR	Downloads	How many people have downloaded the application, as well as related metrics such as app store placement, and ratings.	1 (Number of downloads - CE),
	Customer acquisition cost (CAC)	How much it costs to get a user and to get a paying customer.	-
	Launch rate	The percentage of people who download the app, actually launch it, and create an account.	1 (Number of new accounts - IR), 1 (Number of activations - IR),
	Percent of active users/-players	The percentage of users who've launched the application and use it on a daily and monthly basis: these are your daily active users (DAU) and monthly active users (MAU).	1 (Number of daily active users - CE), 1 (number of users - CE), 1 (Number of application installations - CE, IR)
	Percentage of users who pay	How many of your users ever pay for anything.	1 (Frequency of platform usage - CE)
	Time to first purchase	Time between account activation and first purchase.	-
	Monthly average revenue per user (ARPU)	Revenue per user through purchases and watched ads.	1 (Frequency of platform usage - CE)
	Ratings click-through	Percentage of users adding ratings or reviews.	-
	Virality	On average, how many other users a user invites.	-
	Churn	How many customers have uninstalled the application, or haven't launched it in a certain time period.	1 (Number of un-installations - IR)
	Customer lifetime value	How much a user is worth from cradle to grave.	-
Media site: ID	Audience and churn	How many people visit the site and how loyal they are.	1 (Number of followers on social media platforms - ID)
	Ad inventory	The number of impressions that can be monetized.	-
	Ad rates	Sometimes measured in cost per engagement—essentially how much a site can make from those impressions based on the content it covers and the people who visit.	1 (Tracking articles - ID)
	Click-through rates	How many of the impressions actually turn into money.	-
	Content/ advertising balance	The balance of ad inventory rates and content that maximizes overall performance.	1 (Key words -ID),

Table 6.14. A match of Croll's and Yoskovitz's Lean start-up metrics [2] to the metrics found in our study. (contin.)

Type and Start-ups	**Lean Start-up Metrics**	**Metric Description**	**Matching/Related Lean and Start-up Metrics**
User-Generated Content: IY	Number of engaged visitors	Frequency and length of customer visits.	1 (Number of returning customers - IY)
	Content creation	The percentage of visitors who interact with content in some way, from creating it to voting on it.	-
	Engagement funnel changes	How well the site moves people to more engaged levels of content over time.	1 (Number of people using the platform - IY)
	Value of created content	Business benefit of content, from donations to media clicks.	-
	Content sharing and virality	How content gets shared, and how this drives growth.	-
	Notification effectiveness	The percentage of users who, when told something by push, email, or another means, act on it.	9 (Product awareness by public - IY)
Two-sided Market Place: WT, WE, IP	Buyer and seller growth	The rate at which you're adding new buyers and sellers, as measured by return visitors.	1 (Number of organizations contacted in a quarter - IP), 1 (Number of walk-in clients - IP)
	Inventory growth	The rate at which sellers are adding inventory—such as new listings—as well as completeness of those listings.	20 (Performance of retailers and wholesalers - WT)
	Search effectiveness	What buyers are searching for, and whether it matches the inventory you're building.	1 (Number of site visits - IP)
	Conversion funnels	The conversion rates for items sold, and any segmentation that reveals what helps sell items	-
	Ratings and signs of fraud	The ratings for buyers and sellers, signs of fraud, and tone of the comments.	-
	Pricing metrics	If you have a bidding method in place (as eBay does),then you care whether sellers are setting prices too high or leaving money on the table.	-

Table 6.15. Unique metrics found in East Africa start-ups in comparison to Lean start-ups metrics [2]

Start-up Categorization	Comparatively unique metrics found in East Africa start-ups
Software as a service (SaaS)	16 (Ability to build a complete product - CB and CES), 24 (Set and evaluate tasks - CZ), 17 (Feedback from friends about product features - CZ), 7 (Ability to close a business deal - CES), 21 (Time-based task setting, tracking and review for progress of project/staff - NF), 22 (Time-based project performance appraisal - NF), 1 (Track impact of advice-IA)
E-commerce	15 (System reliability - RI), 21 (Time-based task setting, tracking and review for progress of project/staff - RI), 6 (Tracking market indicators/market events - RI), 8 (Set and review business targets - IH), 14 (Production process time estimation - IH), 2 (Product delivery process time estimation - IH)
Free Mobile App	13 (Usage per product feature - IR)
Media site	6 (Number of events attended - ID), 1 (Posting time by customers - ID)
User-Generated Content	1 (call given number of customers - IY), 11 (get product feedback-IY)
Two-sided Market Place	27 (Document and tracking progress of team members - WT), 12 (Reaching key business milestones,like tax registration, incorporation, patents -WE), 9 (Number of customers aware of the product - IP)

metrics in the East African start-ups that could not be matched or related to the known lean start-up metrics. Table 6.15 summarizes these metrics. Note that the table shows the metrics found only in the listed East African start-ups. We provide the ID from the corresponding aggregated metric in Tables 6.3 and 6.4.
On matching with lean start-up metrics, we found 21 new metrics in the studied early-stage start-ups. These include 7 in the SaaS start-ups, 6 in e-commerce start-ups, one in free mobile apps, 2 in media site start-ups, 2 in user-generated content start-ups, and 3 in two-sided market places.

6.4 Discussion of Results

This section discusses the results concerning our main research questions RQ1, RQ2, and RQ3.

6.4.1 RQ1: What are software start-ups in East Africa (a nascent ecosystem) assessing and measuring?

Our results indicate that even early-stage start-ups in East Africa see value in measuring. Some are experiencing or expecting benefits. However, we did not find any formal measurement programs in these start-ups.

Business metrics: Our results indicate that business metrics were the most popular among the start-ups in East Africa. The customer analytic metric is the most used and wished for by the start-ups of the business metrics. It is an aggregated metric with several sub metrics specific to a niche or particular business. One of the possible reasons this metric is popular could be that it is widely supported by existing expensive and affordable tools. For example, Facebook and Google offer affordable customer-related analytic tools. The other could be that the customer analytic metric also provides quantifiable values (like the number of customers, number of active accounts, and number of followers) that start-ups may use in decision making or get insight into their young businesses. Early-stage start-ups are trying to get the product/market fit[5] right as one of the ways to prove that they are

providing value to customers. To have confidence that they will commercialize, they need to analyze customer-related aspects of their businesses. Finally, the popularity of business metrics, in general, may be caused by the start-ups' need to become profitable in a short time.

Product metrics: Product-related metrics were the second most popular, and it may not be surprising given that most of the start-ups (13/19) are two years old and below. During the early stages, several start-ups are trying to develop the right product to fit the market (product/market fitting[5]). However, it is interesting to see that start-ups of hub C mostly use these metrics, while start-ups from other hubs are instead of wishing to use these metrics, hinting at a potential common mentor in hub C for its start-ups.

The product/feature usage metric is a popular wished for metric by the start-ups (IN, IY, CE, OC). It is somewhat surprising that no start-up uses this metric, possibly because it requires customers' information that the start-ups may not have in their early stages. This could also be why start-up CZ (11 months old) uses the feedback from friends about product features (peer endorsement) metric instead. Three of the four start-ups (IN, CE, OC) that wish to use this metric are below two years. It is harder to say why older start-ups are not interested in this metric. A probable reason could be a shift in focus, e.g., towards business aspects, favoring customer analytic metrics. In contrast to the product/feature usage metric, those product metrics that are in use appear to need low staff effort, require little or no tooling, and are convenient, a potential reason why start-ups opt for them.

Finally, system reliability is mentioned by RI, the oldest start-up (51 months). A reason for that might be that this metric becomes only meaningful once there is a working deployed system. Again, we do not know why this start-up is not using the metric. However, a possible reason could be that it is difficult to measure, requiring an expertise that the start-up may not have.

Organizational metrics: Organizational metrics are the least mentioned category by the start-ups. We identified only three of such metrics. These include Key Performance Indicators (KPIs), time-based task setting, tracking and review for project/staff progress, and time-based project performance appraisal from the start-ups. Four start-ups use these metrics, two relatively young start-ups ID (12 months), WT (10 months), and two older start-ups IP (32 months), NF (43 months), and wished for by another older start-up RI (51 months). The motivation for the older start-ups could be that they have started to look at more prominent solutions other than the Minimal Viable Product (MVP)[143] [144]. They are thus managing their solutions as projects with project teams, unlike the younger start-ups. Start-up ID is a younger start-up using metrics in this category. However, this start-up has some experienced partners, as clarified during the interview:

"One of my partners is an ardent investor, he has made past mistakes in investments, raising money and also setting KPIs making priorities focus and brings such expertise in the team [...] so we have a team of advisers one of them is the CEO of company x"

The youngest start-up in this metrics category (WT) had just successfully gone through its funding round. Its preparation for funding may be the reason to use these organizational metrics to convince the investors. It would be interesting to study investors' role in metrics that start-ups adopt in future research.

Project metrics: Project metrics were mainly wished for by (4/6) start-ups. The most popular used metric in this category is documenting and reviewing activities for progress.

Three start-ups use it (IS, WE, WT). When comparing used and wished for metrics in this category, it can be seen that used metrics can be assessed manually and without special tooling. In contrast, the wished-for metrics, such as process adherence, seem more difficult to assess,

Design metrics: Our results showed that there are no metrics targeted at product design by any start-up. The lack of metrics on product design could indicate a (1) lack of knowledge on what to measure during design or (2) the lack of a formal measurement program (like MeSRAM) that explicitly states what aspects of product design to measure. It is alternatively possible that the start-ups are aware of such metrics but consider their code-base to be too small to have a reason for such metrics.

Perceived sufficiency: The results showed that four start-ups NF (43 months), IN(15 months), IP (32 months), and IS (39 months) perceive their measuring to be enough. It is interesting to observe that three of the four start-ups (NF, IP, IS) are over two years old. Three of these start-ups (IN, IP, IS) come from the same hub (I). However, they use different measurements, possibly, because they operate different start-up types: IN is a Fintech, IP is a two-sided marketplace, IS is a Software as a Service(SaaS). Thus, we see again a hint that hubs are having an impact on what start-ups measure. However, the general trend shows that start-ups would like to have better measurement programs. This is visible in the high number of start-ups reporting that they do not perceive their measurement as sufficient and the high number of wished for measurements reported.

Future research: Finally, it would be interesting to know why start-ups decide the metrics they use. The cross-analysis could not prove a relation of metric choice and factors such as hub-type or country. We suspect that more external factors are involved, such as the start-up's relationship with other entities, e.g., mother companies or investors, or factors such as costs (some metrics are more expensive to implement than others). We plan to address this question in future work.

6.4.2 RQ2: What are the perceived benefits of measurements to start-ups?

Results indicate that most start-ups have not experienced real benefits of measuring but have expectations of benefits if they measure enough. This result is in tandem because although all the start-ups' measure, 84% of the start-ups perceive not to measure enough, and some are not even sure of measuring. The results compare well with 82% of the 22 benefits that the start-ups just expected rather than experienced. Three start-ups (IS, NF, CZ) have experienced the benefits of measuring. For the two older start-ups, IS (39 months) and NF (43 months), we might expect that they had enough time to use the measurements and experience their effects. It is also possible that these companies already learned what measures work best for them. Both companies also consider themselves to measure enough. Start-up CZ (11 months) is young but experienced some benefits of measuring because it actively measures its important business aspects enabling it to make decisions and provide insight, as exemplified by the following quote:

> *"For a given month everyone is given tasks, and we evaluate these tasks, and if by the end of the month the tasks have not been completed then we realize we are stuck."*[CZ]

It has been shown that start-ups have big hope in measuring since the majority expressed their expectations from measuring. There seems to be a relationship between the perceived sufficiency of measuring and expected and experienced benefits in start-ups. Start-ups that are not measuring enough or not measuring may not experience the benefits of measuring, although they may have expectations of the positive benefits of measuring. Start-ups have shown hope in the benefits of measuring, and they generally appreciate that they do not measure enough. One interesting question is why start-ups are not experiencing more benefits of measuring. The cross-case analysis revealed no significant differences in start-ups' characteristics that experience benefits and those that do not. The lack of benefits may be due to ill-fitting measurements. However, it is also possible that some start-ups are applying good measures but cannot react to the measurement results. Future work will have to investigate the reasons for that further.

6.4.3 RQ3: How do metrics used in East African start-ups compare to those in lean start-ups in developed countries?

Start-up Categories: The nascent East Africa start-up ecosystem has all types of the start-ups found in lean start-ups summarized by Croll and Yoskovitz [2]. Software as a service is a popular start-up model in the East African ecosystem, potentially due to its ease of scalability and its potential to be self-sustain (fewer overheads) after the initial investment. We observed that not all the start-ups interviewed in East Africa could be categorized using the lean start-up categorization. The limitation may indicate the need for a broader categorization of the start-ups in East Africa, such as fintech. It also directly indicates the need to investigate essential metrics for these types of start-ups in the nascent East Africa ecosystem.

Metrics: Taken together, the East Africa start-ups used or wished to use at least half of the known lean start-up metrics in each category. However, single start-ups have much lower coverage of those metrics. The choice of the lean metrics to use may be influenced by the start-up growth stage, as our earlier results showed that most of these interviewed start-ups are two years and younger. Another reason for the low coverage is that start-ups may not know all the essential metrics to capture their particular type of start-up. Furthermore, the start-ups are also using or wishing for some unique metrics which have not been listed by Croll, and Yoskovitz [2]. Future work will have to show whether this is due to the East Africa ecosystem or due to the relatively young age of start-ups interviewed in this study.

Summary: Our results indicate the need for additional lean start-up categories that include fintech, software development, and software as part of work-flow. These new proposed categories mean that more lean start-up metrics need to be identified to enable start-ups in these categories to measure more effectively. We found new metrics in the start-ups in East Africa. This may be an indicator of contextual influences on the start-ups and may need more contextualized research on lean metrics or more inclusive metrics for start-ups in East Africa.

6.4.4 Cross-case analysis

The cross-case analysis delivered surprisingly few results. It is exciting to see that variables, such as start-ups age, country, or hub-type, have no significant impact on the used metrics

or experienced measurement benefits. Primarily, we would have expected an increase in metrics use with age since large companies use much more metrics than start-ups. There are possible alternative explanations. One is that our data-set is simply too small to prove such effects. An alternative interpretation could be that differences in metric use are visible only after a much longer time than the 1 to 3 years of age difference in the studied start-ups. The differences may not be in the number of metrics used or benefits experienced, but in the type of metrics used/benefits experienced. Future studies will have to investigate this possibility more in detail. Finally, the significant result of $H0_{5expected}$ seems logical, as benefits that are already experienced are not counted as expected anymore. Again, future studies will need to confirm or refute this finding.

6.5 Threats to Validity

For the discussion of this study's validity, we follow a classification scheme used by Runeson [116], and Yin [115].

Construct validity: The main threat to construct validity is the risk that we have asked the wrong questions to assess what is measured. To mitigate this threat, we created the interview guide in several iterations to ensure that all three authors had the same understanding of the questions. Furthermore, we clarified and merged some interview guide questions after interviewing the first four start-ups (RD, RK, RW, and R1). To further increase the chances to get a complete picture of the measurements used, the interviewer used reformulated and repeated questions when the interviewee indicated that they did not fully understand a question or when the discussion deviated from the original topic.

Internal validity: A common risk during the analysis of data is a misinterpretation of what the interviewees said. We recorded and transcribed all the interviews and listened to them again in case of doubt. Besides, we iterated together over the used codes and observations to make sure all measures were identified. These mitigation steps turned out crucial. We initially omitted many of the unusual measures as they were only mentioned by interviewees when asked how they assess their progress, but not when asked for what they measure.

External validity: Regarding external validity, our data has some limitations. First of all, we studied start-ups older than six months and younger than 4,5 years. While we believe that start-ups that are slightly younger or older than that might have a similar use of metrics, we do not expect the metrics to be the same across all company ages and maturities. Similarly, our study focuses on a specific emerging ecosystem, namely East Africa. It is difficult to predict to what degree our findings can hold for software start-ups outside this regional context, e.g., in Europe. A final threat to generalizability stems from the fact that we only interviewed start-ups working within hubs. However, to the best of our knowledge, there are only a few early-stage software start-ups in East Africa working independently of hubs. Therefore, the 19 interviewed early-stage start-ups allow a representative assessment of measurement in software start-ups in East Africa.

6.6 Conclusions and Future Work

Early-stage software start-ups in the nascent East Africa ecosystem measure some of their technical and business aspects. Using the MeSRAM[1] group for "metrics used," the start-ups have been seen to preferably using business and product metrics and, to a lesser extent, organizational performance metrics. There has been no evidence of use or wish for design metrics (which exist in large software organizations) in these start-ups. The older start-ups have been shown to also wish for project-oriented metrics. Organizational metrics are the least used metrics in the studied start-ups.
Start-ups have also shown considerable expectations in the benefits of measuring, although a number of them perceive their measuring activities as not enough. Several start-ups in East Africa can be categorized using the lean start-up framework[2]. They are also using or wishing to use some known lean start-up metrics, but the lean start-up framework does not capture some metrics. The lean analytic framework also fails to categorize some types of software start-ups in East Africa, although they exist and use or wish to use some metrics. In the future, we will like to investigate the mapping between the used and wished for metrics and specific known growth stages of early-stage start-ups. We would also like to examine the suitability of these metrics in measuring certain aspects of a start-up. Using the metrics is one thing, but having a quality metric is another. Therefore, we will want to investigate the various properties of these metrics so that they can be qualitatively compared to metrics known elsewhere. We will also want to ascertain the extent of the use of metrics within start-ups. Finally, we will like to propose an extension to the Lean Analytic framework that will incorporate the currently unclassified start-up types in East Africa and common metrics they must measure as they grow.

Chapter 7

Paper 4: The Character of Metrics: A Survey of Start-ups and Mature Software Companies in an Emerging Ecosystem

Chapter Overview

Measurement and metrics play an important role in software engineering in monitoring and evaluating processes, products, and resources. Start-ups and mature software companies use metrics in decision making and thus perceive the metrics to be beneficial. However, there is limited knowledge of metrics that may influence their choice and use in software start-ups. This study aims to derive characterizing dimensions for metrics and survey software start-ups and mature companies. We surveyed 19 software start-ups and ten mature software companies in Uganda. Results show no significant difference in six metric dimensions that include: use of a metric, phase of assessment, target users of the metric, time spent on assessing a metric, time used to set up tools, and objectivity/subjectivity of the metric. Four dimensions that significantly differentiate the metrics include frequency of assessment, automation of the measuring process, type of investment in the measuring tools, and the final form of the metric. Start-ups and mature software companies can choose to use metrics differently based on four dimensions.

7.1 Introduction

Measurement is a key activity in software engineering. In software engineering, measurement is used to characterize, evaluate, improve, and predict software engineering aspects like processes, products, and resources [56]. Measurement uses metrics to monitor, control, and improve software processes and product quality [101] and has diverse software engineering applications. For example, measurement is used for (a) the assessment of the complexity of software products [145] [146], (b) estimating software cost [147], (c) product size measurement [148] (d) determination and evaluation of the design of software [149], (e) defect prediction [150], and (f) monitoring of software processes [151]. Wang presents an axiom-based measurement theory for software engineering that defines what can be measured and presents a generic measurement methodology. The theory explains an object of measurement, attributes of an object to be measured, measurement scales, and

measurement units of measurement [103].

Existing studies have indicated the metrics used in software start-ups. For example, Croll and Yoskovitz [2], in their work, identify fine-grained metrics tracked by lean start-ups. Our previous work highlighted metrics used by early-stage software start-ups and their perceived benefits in an emerging ecosystem [48]. Large software companies mainly use metrics for insight and decision making using well-structured Measurement Programs (MP) [152]. We found that early-stage software start-ups companies [48] similarly used metrics for decision making and insight. The metrics, their use and benefits in software start-ups, and mature and large software organizations are known. However, not much is known about the dimensions that potentially influence the metrics' choice and use.

This study aims to derive dimensions for metrics and use them to survey start-ups and mature software companies. The following research questions guide us:
RQ1: How do start-ups and mature companies characterize their metrics along with a set of dimensions?
RQ2: Which dimensions show significant differences between metrics used in start-ups and mature companies?

To answer the questions, we iteratively identified and refined ten dimensions of interest and surveyed software start-ups and mature software companies in Uganda. This study is a first step in understanding what might stop software start-ups from using the same metrics used in mature software companies. Further, the dimensions provide a starting step for future studies on what influences start-ups to adopt and use metrics.

The remainder of the paper is organized as follows. In section 2.10.4 we discuss related work. Section 7.2 presents the methodology used in this study. Section 7.3 reports the results of the survey. Section 7.4 presents the discussions of key results, their implication to research, and validity threats to the study, while Section 7.6 presents the conclusion.

7.2 Research Method

This section presents the methodology we followed to identify the ten dimensions for the survey. Table 7.2 shows the resulting final ten dimensions, their definitions, scales, and type of expected responses used in the online survey.

7.2.1 Development of the survey instrument

Wang asserts [103] that measurement is a quantifying process of four steps that include: 1) categorization -that involves the identification and elicitation of common attributes among objects to be measured 2) yardsticking - defining scales of measurements for the identified attributes 3) metrization - finding relationships between same attributes of two objects, or between an attribute and a given measurement scale and 4) interpretation - the explanation of the physical or the cognitive meaning of the measurement result defined by the attribute, scale, and unit of measure.

In the following subsections, we follow steps 1-2 by Wang to iteratively identify the metrics' dimensions and develop and refine their categorical scales. The metrization and interpretation steps are presented in analyzing the survey results using hypothesis testing and discussion sections of this paper. Figure 7.1, summarises the main steps in the derivation of dimensions that form the basis for the survey.

The dimensions were motivated by good metrics' characteristics and the mapping of the

metrics to target users by Daskalantonakis [26] and known start-up challenges include lack of funds, severe lack of resources, and time pressure highlighted by Giardino et al. [12].

Initialization and trial 0 (Self-Experimentation)

Initialization of dimensions: The challenge was to identify as many dimensions as possible, considering the target users, stages of growth, and the challenges software start-ups faced. Therefore, we started with a brainstorming session taking as input 28 metrics found in East African start-ups in a previous study [48]. During the session, we focused on identifying aspects of the metrics' assessment and use, i.e., the "how".

As a result, we formulated 15 potential dimensions related to the required effort for assessment in terms of time, required skills, the investment required, use of the metric, the form of the metric, technical-business phase to assess the metric, and tools used for the assessment. Note that this session's goal was not to guarantee that all identified dimensions impact the likelihood of using a metric in a start-up. Instead, the goal was to develop a rich set that was likely to contain relevant dimensions concerning known challenges faced by start-ups, the operations of the start-ups, and the metrics' users. For each dimension, we formulated a question, e.g., "How much dedicated time does the team use to measure this metric?" and added a categorical scale of possible answers to the question, e.g., 1) A lot of time, 2) Moderate time, 3) a little time.

Run Trial 0: As a first pre-trial, we used the dimensions to characterize the 28 metrics found in our previous study.

Observations: The main observation from this exercise was that the same metric could score differently on a dimension depending on its use in the company. For example, a metric about customer awareness based on social media mentions could be assessed manually or automatically, depending on the companies infrastructure. A second observation was that several dimensions needed rework as their definitions still allowed too much room for interpretation.

Trial 1 (Paper pilot I)

After reformulating the dimensions and questions to make them more precise, we performed the first trial. We maintained the 15 questions that we had in the pre-trial.

Run Trial 1: We piloted the dimensions with a paper form. Each dimension was formulated as a question with a numbered categorical scale. We included demographic questions about the type of start-up in the form—for example, the services/products offered by the start-up and the company's operational age. Further, the form contained a multiple-choice list of 16 metrics identified in an earlier empirical study of start-ups in Kenya and Uganda [48]. The start-ups were only required to choose the metrics they recognized from this list. The main reason for showing this list to start-ups was to gauge the familiarity of the start-ups' metrics to validate our earlier findings. This study focuses on the metrics used in software start-ups, and we do not further analyze and report the responses of the start-ups on familiar metrics here. The start-ups were required to fill the metrics they used and characterize each along the ten dimensions. We administered the paper pilot to four start-ups. Three of the start-ups were based in a university hub, and the fourth was an independent start-up. We asked the start-ups to read the questions and scales first, then fill a metric in the form and for each item, use the dimensions and their accompanying scales. While we did this, we minimally interacted with the start-up founders. The four start-ups characterized 17 metrics. Ten (10) of the metrics were customer-related business

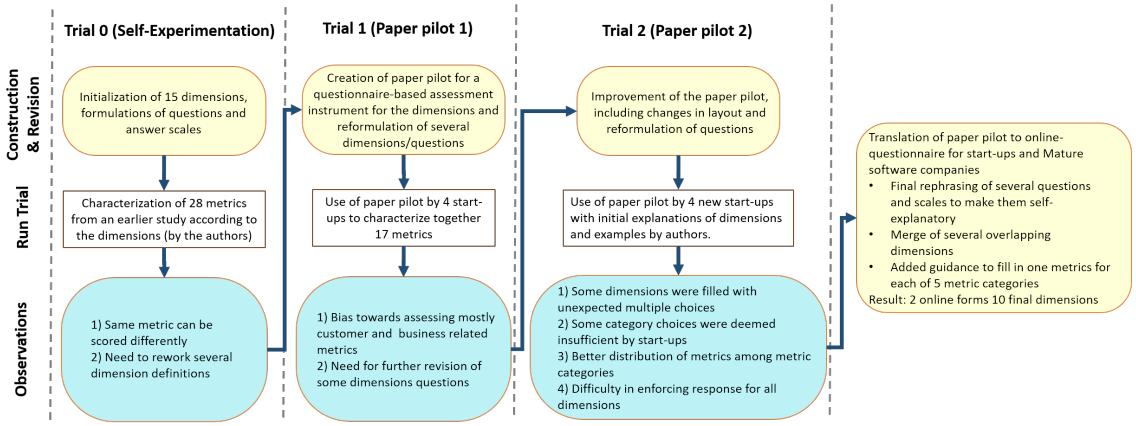

Figure 7.1. Iterations to derive the 10 dimensions for the metrics used in the survey tool

metrics, 4 product metrics, and three project metrics.

Observations: The metrics were biased to customer-related metrics and product metrics. Thus, to ensure a more balanced assessment of the different metrics, including project metrics, the questionnaire needed to be revised. The observations also indicated that some questions might not have been easy to understand. The difficulty in understanding the problems motivated us to carry out a second paper trial. In this second paper trail, we encouraged the respondents to give us feedback as they filled the paper form and ask for clarity on any aspect of the form.

Trial 2 (Paper pilot II)

To capture better feedback about the dimensions, we improved the paper form's visual layout by adding more explicit borders between different form sections.

Run Trial 2: We administered the form to four new start-ups, one of those was hosted in a hub, and the others were not. This time we explained the questions and provided some examples where start-ups requested them. We let two of the start-ups fill the form in our absence, and another two start-ups filled the form in our presence. We allowed the two start-ups that we engaged to ask questions about the form. The form had space to fill in at most eight metrics.

Observations: We closely interacted with the two start-ups that filled in multiple responses for some dimensions that we thought required single responses. The interaction was to guide the start-ups on the expected number of responses for the categorical scales. For example, for the dimension question: *"What is the source of information used in this measurement?"* with single options *[(1) Other companies, (2) Team, (3) Business partners, (4) External friends/peers]*, the start-ups filled in more than one choice for some metrics. In some dimensions, the trial uncovered that the given categorical answers were not sufficient. Take, for example, the question *"Do you use any tools or know any tools that can automate this measurement?* with options *[(1) Yes, (2) No, (3) Not sure]"*. One of the start-ups wrote on the paper form that *" different parts of the metric could be automated, others cannot"* for their metric named *'time required to carry out a process'*. There was a better spread of metrics in the metric categories of business, product, project, and organizational performance metrics in this trial. However, one of the start-ups listed six metrics but did not fill in all the dimensions. With a paper form, it is not possible to make responses mandatory.

Table 7.1. Metric types definitions adapted from MeSRAM framework by Staron [1].

Metric type	Definition
Business Metric:	This is a metric used to assess and quantify such aspects of company operations as customers, value, financial performance, product/service delivery efficiency, and product quality.
Product Metric:	Quantifies how good a product is, its potential, or the readiness of the product for release. For example, product quality, performance, and management; the speed of development; continuous product improvement; product maintenance; product monitoring, etc.
Organizational Performance Metric:	Shows how fast the organization is (velocity), its efficiency, and its response to customers. For example, measures related to employees' skill set, the organization's speed, measures for the customer, and financial perspective.
Design metric:	Captures the internal quality of the software. For example, design stability, product/code stability, product size, design debt/technical debt, and defects.
Project metric:	Captures measures and indicators for project managers for monitoring progress. For example, progress, status, quality of products, team, test effectiveness, and efficiency, etc.

Resulting online survey tool

The observations in the previous trials and limitations of the paper form motivated refinements and development of an online survey form for software start-ups and mature software companies. The online survey had the additional advantage of scaling better.

Refining the dimensions: Based on the input from the previous paper trials, (1) we rephrased several dimension questions to make them more precise, (2) we adopted new scales and added examples to some questions to make them self guiding, (3) we enforced restrictions on the scales, and (4) questions that probe similar aspects of metrics were discussed and subsequently merged and rephrased with revised or new scales. The refinement resulted in a final set of 10 dimensions. Table 7.2 indicates the final ten dimensions, their definitions, scales, and categorical responses for the online survey.

Balanced metric types: To ensure the balance of captured metric types, we structured the online survey form into sections to capture the six different types of metrics. The introductory section contained demographic questions (e.g., age, number of employees, type of products). Each of the subsequent sections was a metric category and accompanied by ten dimensions questions and their scales. We adopted the MeSRAM metric [1] categories that are a broader set of categories that encompass an earlier but smaller group of metric types by Daskalantonakis [26]. The MeSRAM categories described in Table 7.1 Each section started with a definition of the metric type and invited the respondents to provide one metric of that type used in their company. Subsequently, the respondents used the dimensions and associated scales on the named metric. The main aim of all these changes for the online form was to make the tool self-guiding.

Extension to mature companies: Finally, we modified the online survey to suit mature software companies. For the mature software companies, we revised the demographics section to capture the number of employees and the company's age. We required mature companies to state the type of software products/services they develop and provide. In the mature companies' form, we did not include the list of start-up metrics. The list had been included in the start-up form because we were validating a previous study of start-

up metrics. We do not report the results of this validation in this study. The rest of the sections captured the different metrics and used the same ten dimensions as those used for the start-up survey form.

7.2.2 Population and sampling

The target population for the survey was early-stage software start-ups and mature software companies in Uganda. In this study, we look at software start-ups as newly created firms developing software-intensive products/services [4] and early-stage software start-ups as those in the idea conceptualization phase to first time to market [29]. This study describes mature software companies as companies with a working history, many employees, a management structure, established products/services with customers, established technical and business roles, and physical offices.

The study used the non-probabilistic quota sampling [122] because we could not find a database that listed software start-ups and mature software companies in Uganda. The target population was split into two groups that included software start-ups and mature companies. We used quota sampling because we expected the start-ups and mature companies to have different experiences on how they initiate and use metrics. Given the known mortality for software start-ups, we expected more software start-ups than mature companies. Thus the splitting of the sample ensured a proportional number of respondents in each stratum. We used both convenience and snowballing approaches [122] to reach out to respondents in each stratum. To reach out to the software start-ups, we first listed the active hubs in Uganda that hosted start-ups. The hubs allowed us to access the founders of the start-ups they hosted. The hubs also recommended their alumni software start-ups. The start-ups we met also recommended other start-ups they knew. For each identified start-up, we emailed the survey form to one founder in a software start-up. For mature software companies, we approached technical managers who we had initially met in different forums or those known to alumni that we had contacted. Due to the multiple roles start-ups' founders play, they were savvy to both the technical and business metrics. Technical managers in mature companies suited the survey because we assumed that an experienced technical manager could identify 4 out of the 5 (products, project, design, and organizational performance) types of metrics. These 4 metric types saddled the technical and team related aspects of companies. Given the managers' administrative role, we assumed that they liaise with the business teams and are savvy to the business teams' business metrics. We met each start-up founder and technical manager in a mature company, established their roles, and the survey's objective explained, after which we asked for consent. The online survey was sent to 45 software start-ups and 20 mature software companies.

7.3 Results

This section presents the ten dimensions for 19 software start-ups and ten mature software companies. We received 42% respondents with 82 start-up metrics and 50% respondents for 50 mature company metrics.

7.3.1 Character of the software start-ups

Table 7.3 characterizes the software start-ups in terms of the number of employees, the company age and types of software products/service that they offer. We used start-up categories identified by Croll and Yoskovitz [2], and added two categories for Fintechs and

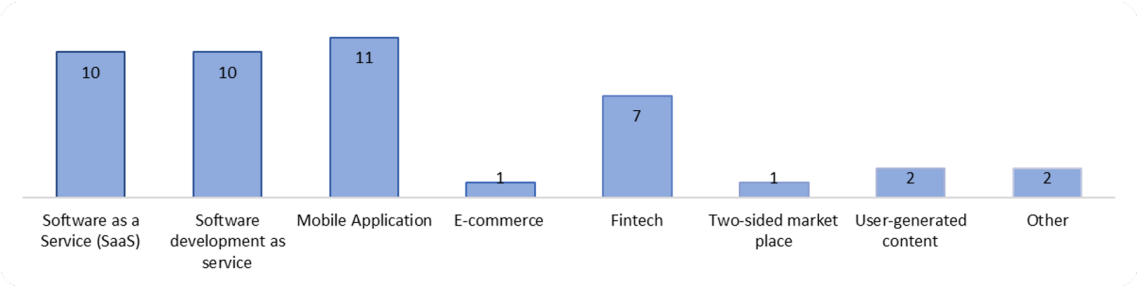

Figure 7.2. Products/services in surveyed start-ups.

Software development as a service. These extra start-ups types were identified in the East Africa ecosystem and documented in an earlier study by [48]. 14/19 (87%) of start-ups surveyed employed at most 5 people with the rest having 6 or more employees. The start-ups with 6 or more employees (S01, S03, S09, S19) generally offered more than one type of product or service. Figure 7.2 highlights the products and services in start-ups.

Most of the start-ups offered software as a service (SaaS) (10/19), software development as a service (10/19), mobile applications (11/19), and Fintechs (7/19).

7.3.2 Character of the mature software companies

Table 7.4 characterizes the surveyed mature software companies in Uganda. Most of the mature companies (6/8) developed/used financial/banking software. 9/10 of the mature companies employed less than 50 staff, and the majority (7/9) had operated for 5-10 years. Only two companies (MC03 and MC06) over ten years old.

7.3.3 Analysis

To get results, each metric, and each multiple-choice dimension (D01, D02, D03), we counted each dimension's responses. For each dimension, we counted the responses (per choice) and also counted the no-responses. For example, in D01, the business team had 52 responses and 30 no-responses in start-ups and 30 responses and 20 non-responses for the mature software companies.

Because of the categorical nature of the responses, we ran Chi-square tests for each set of responses. The tests had two independent sample populations for comparison (software start-ups and mature software companies). Each of the ten dimensions had a null hypothesis following the pattern:*"<<*dimension of metric*>>" does not differ between start-ups and mature software companies"*. The resulting null hypotheses are shown in Table 7.5. For multiple-choice responses, we performed Chi-square tests, one test for each response on the answer scale, except "not sure". To compensate for multiple tests on multiple-choice responses, we performed a Bonferroni correction [153] on the thresholds for the p-values. The single response dimensions (D04, D05, D06, D07, D08, D09, D10) used two Chi-square tests, i.e., in 1) we tested the dimensions for all values including the "not sure" responses, 2) performed the chi-test without the "not sure" response. To double-check whether "not sure" response creates significance. In this paper, we primarily report the test results that exclude the "not sure" responses. The Chi-square tests worked with a statistical threshold of $p < 0.05$.

Also, we used Cramer's V to determine the effect size of the relationships for each dimension. All the Chi-square and Cramer's V tests were done with R software [154]. For the Cramer's V effect size of the relationships, we adopted the definition by Cohen [155] that

defines and describes the different effect sizes: "Small effect size: r = .10; r = .012. Relationships of this size would not be perceptible based on casual observation; many relationships pursued in “soft” behavioral science are of this order of magnitude. Medium effect size: r = .30; r = .092. This degree of relationship would be perceptible to the naked eye of a reasonably sensitive observer. Large effect size: r = .50; r = .252. This magnitude of the relationship falls around the upper end of the range of correlation coefficients encountered in behavioral science; correlations “about as high as they come.”

7.3.4 RQ1: How do start-ups and mature companies characterize their metrics along with a set of dimensions for the characterization?

This section presents the results for the characterization of the metrics using the six non-differentiating dimensions. It should be noted that dimensions for *"use of the metric"* (D01), *"phase of assessment "*(D02), and *"users of metrics"* (D03) had multiple responses. Thus, these dimensions' total responses may be more than 82 and 50 in start-ups and mature software companies. For example, for dimension *"use of the metric"* (D01), start-ups provided the following responses: Business (52 responses), Technical (52 responses), Specialists (21 responses) and thus a total of 125 responses for this dimension. For this same dimension, mature companies respond; Business (30 responses), Technical (17 responses), Specialists (13 responses), thus a total of 60 responses.

Aggregated common metrics and running examples

Table 7.6 shows aggregated metrics for similar or related metrics in start-ups and mature software companies. The aggregated metrics are used by at least one start-up and one mature software company or metrics used by either multiple start-ups and numerous mature companies. Table 7.6 summarizes the 18 aggregated metrics, the start-ups, and mature software companies that used them, and the companies' categorization of their underlying metrics based on the MeSRAM [1] categories.

The 18 aggregated metrics themes grouped 66 common metrics in software start-ups and mature software companies. We use three metrics as running examples to illustrate how the underlying metrics conform to the trend in each dimension.

The *sales revenue* was the most used aggregated metrics. It was used by 6 mature companies (MC04, MC05, MC06, MC07, MC08, MC09) and 4 start-ups (S08, S11, S14, S18). This metric was directly named "sales revenue" by MC05, MC06, MC08, MC07, S08, S14 and had different names in other companies; MC04("revenue collection"), MC09("energy sales"), S11("sales/invoices"), S18("total monthly revenue"). This metric is categorized as a business metric (by 7 of the companies) and one company as an organization performance metric.

Customer satisfaction was another popular metric amongst mature software companies (MC01, MC02, MC04). This metric was named customer satisfaction by all the mature companies except MC04, called "customer satisfaction and customer problems." MC02 categorized it as a product and design metric. The company also used it for decision making and alerts in both cases and used it for insight as a design metric. It is also used by the technical team, besides its use by the business team in both cases. Two companies (MC04 and MC06) listed the customer satisfaction metric as a product metric.

Project/product quality was categorized as a product, project, and design metric. It is used by four companies (MC01, MC02, MC07, MC10) and two start-ups (S03, S16). The *project/product quality* metric is named variably by companies: MC02("quality and satisfac-

tion"), MC07("product quality"), MC07("code quality"), MC10("quality"), S16("product quality"), S03("quality"). However, most companies see the *project/product quality* as a project metric (4 of the six companies) and on start-up, surprisingly marks it as a business metric.

Product/code stability is the most popular technical metric in start-ups and is used by four start-ups (S03, S08, S10, S16) and one mature company (MC01). All five companies use this metric as a design metric. Another notable technical metric among start-ups is *speed of development*, used by three start-ups (S09, S10, S15) as a product metric. However, the mature company (MC01) uses this metric as a design metric.

In the remainder of the paper, we use one aggregated business metric *sales revenue*, one design metric *product/code stability*, and one product metric *speed of development* as **running examples**. When characterizing the metrics using the dimensions, we present the differentiating dimensions using hypothesis testing and use the running examples to highlight the differences.

7.3.5 Characterising the metrics using the 10 Dimensions

This section presents results using our running examples to characterize the metrics using the ten derived dimensions.

Use of the metric

Figure 7.3 shows responses for the dimension *use of the metric*. This multiple-choice dimension had 125 responses for the 82 metrics in start-ups and 73 responses for the 50 metrics in mature software companies. The majority (78%) of metrics in start-ups are for decision making and a notable (48%) of metrics to gain insight. Mature software companies use metrics in decision making (60%) and gain insight (54%). Both start-ups and mature software companies comparatively use fewer metrics for alerts (start-ups (27%) and mature companies (32%)).

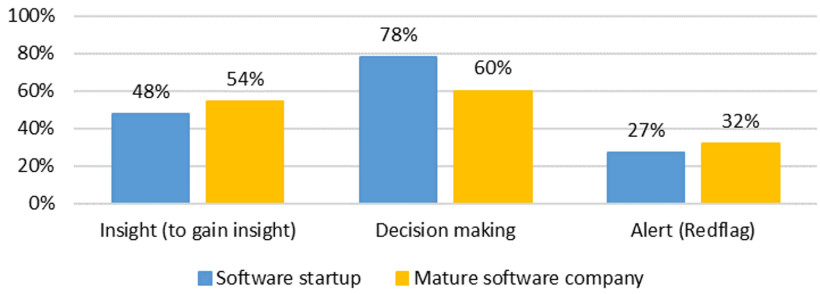

Figure 7.3. Distribution of metrics for the dimension "Use of the metric".

In this dimension, the aggregated metric *sales revenue* is used mainly for decision making by (8/10) metrics and insight in (7/10) metrics. Overall, (6/10) metrics use this metric for both insight and decision making. One metric each used the dimension for decision making and insight alone. The underlying metric *energy sales* by the company (MC09) was marked as "not sure" for this dimension. The metric *total monthly revenue* by start-up (S18) was used for decision making and alert.

The aggregated metric *product/code stability* was variably used. For example, MC01 and S10 use it for alert, S03 for decision making, and S08 and S16 for insight. Finally, the *speed*

of development metric comprises of (3/4) metrics that use it in decision making, with some further use for insight and alerts.

Phases of assessment of a metric

Figure 7.4 shows responses for the dimension *phase of assessment of a metric*. This multiple-choice dimension had 146 responses for the 82 start-up metrics and 84 responses for the 50 mature company metrics.

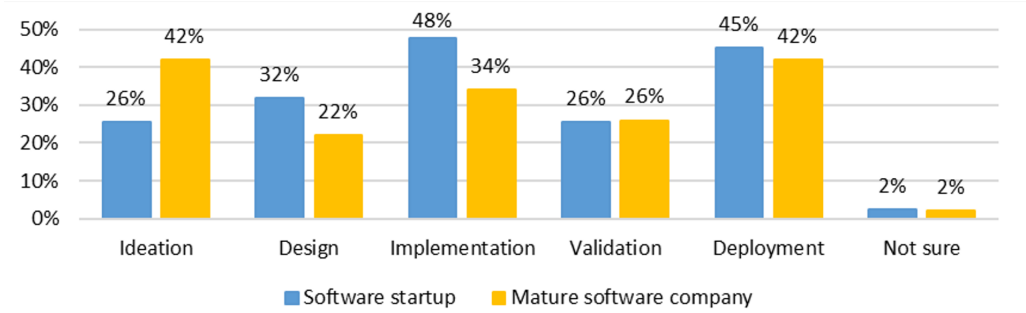

Figure 7.4. Distribution of metrics for the the dimension "Phases of metric assessment".

Both start-ups and mature companies assess most of their metrics in more than one phase (54% of metrics in start-ups and 46% metrics in mature software companies). Software start-ups use most of their metrics in the implementation (48%) and deployment phases (45%). Mature software companies assess most of their metrics in ideation (42%) and deployment (42%) phases and comparatively assess fewer metrics in the implementation phase (34%). The aggregated *sales revenue* metric is primarily assessed in the deployment phase by (7/10) metrics and, to a significant extent, the validation phase (4/10) by the metrics. The *product/code stability* metric is assessed in the design by (4/5) metrics and implementation phase by (4/5) metrics. However, *speed of development* metric is equally assessed across the four phases. These phases include design (3/4), implementation (4/4), deployment (3/4), and validation (3/4).

Users of the metric

Figure 4 shows the multiple-choice dimension *users of the metric* with 125 responses for the 82 start-up metrics and 60 responses for the 50 mature company metrics. Figure 7.5, shows the users of metrics within software companies.

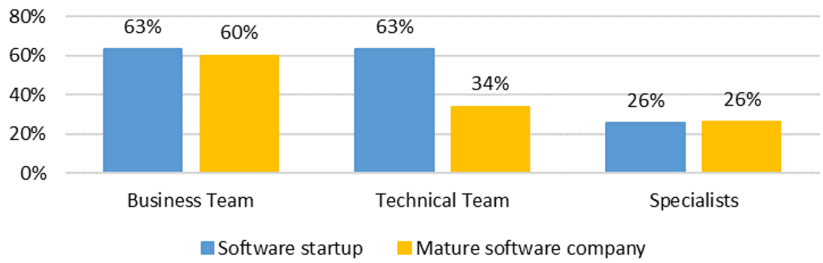

Figure 7.5. Distribution of metrics in start-ups and mature companies with regards to the dimension "Users of the metric within the software company".

43% of the metrics in start-ups and 42% of the metrics in mature companies were used by more than one team. The metrics are evenly used in software start-ups by the business team

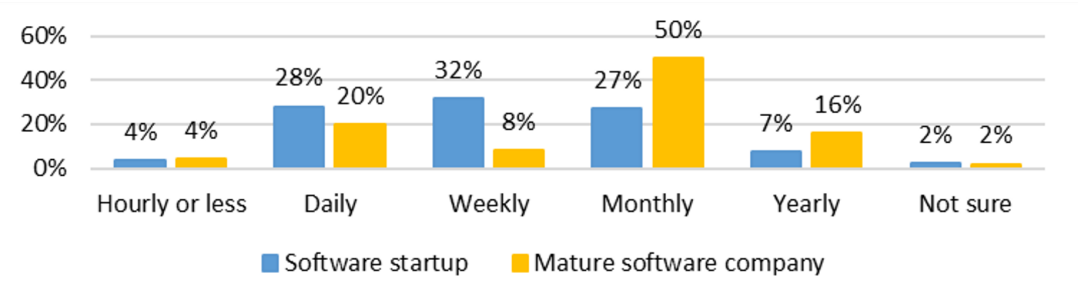

Figure 7.6. Distribution of metrics for the dimension "Meaningful frequency of assessment of the metric".

(63%) and technical team (63%). In contrast, mature software companies used more metrics in the business team (60%) than the technical team (34%). For both types of companies, specialists seldom used metrics. The aggregated *sales revenue* metric was mainly used by the business team (8/10) both types of companies, while *speed of development* is mainly used by the technical team (3/4). The *product/code stability* is mainly used by technical teams (4/5).

Meaningful frequency for assessing the metric

Figure 7.6 shows single-choice dimension *meaningful frequency for assessing the metric* for 82 start-up metrics and 50 mature software companies metrics. Start-ups meaningfully assess their metrics daily (28%), weekly (32%), and monthly (27%), almost evenly. In contrast, mature software companies mostly assessed their metrics monthly (50%), with comparatively fewer metrics set daily and weekly. Mature companies also assess more yearly (16%) metrics compared to start-ups (7%).

For example, most of the companies (start-ups and mature companies) assess the *sales revenue* metric monthly (8/10), with only start-up S14 assessing daily and mature company MC08 assessing it yearly. This dimension seems to vary in software start-ups and mature companies for *product/code stability* metric, ranging from daily to yearly. Start-up S10 was not sure when to assess this metric. For the dimension, the *speed of development* metric is assessed daily to weekly by start-ups, but monthly by the mature company (MC07) using this metric. In this dimension, *volume of the transaction,* a metric used by MC04 and S19, is uniquely assessed hourly or less.

Automation of the measuring process

Figure 7.7 shows the single-choice dimension *automation of the measuring process* with 82 responses from start-ups and 49 responses from mature software companies. In this dimension, start-ups measure most of their metrics manually (43%) and to a considerable extent, automatically (34%), and to a less bit, semi-automatically (23%). In contrast, the majority of metrics in mature software companies are measured semi-automatically (57%), with a comparatively small use of manual (21%) and fully automated (21%) measuring processes.

For the *sales revenue* metric, mature software companies use semi-automated (3/6) to automated (3/6) measuring processes. In contrast, start-ups use manual (2/4) to semi-automated (2/5) measuring methods. For the *product/code stability* metric, start-ups mainly automate the measuring process for this metric, while the mature company (MC03) uses a semi-automated process. Finally, for the *speed of development* metric, the three start-ups

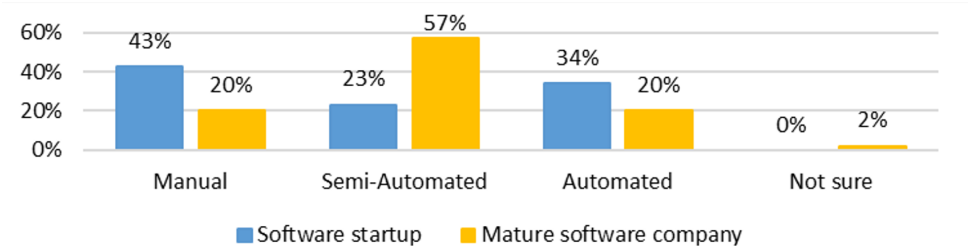

Figure 7.7. Distribution of metrics in start-ups and mature companies with regards to the dimension "Automation of the measuring process".

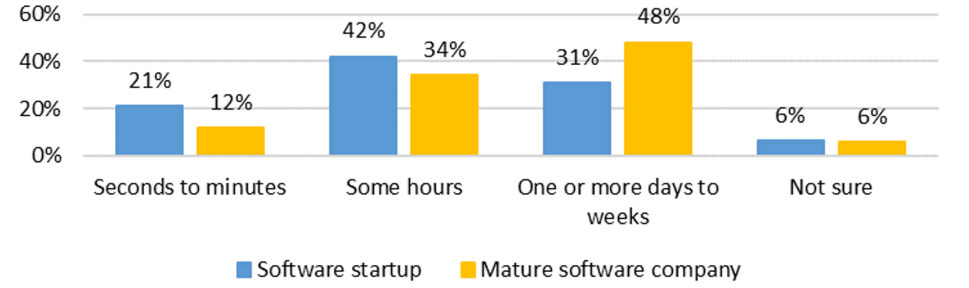

Figure 7.8. Distribution of metrics for the dimension "Time used to assess the metric".

follow manual to semi-automated measuring processes.

Time used to assess the metric

Figure 7.8 shows the single-choice dimension *time used to assess the metric* for 81 start-up responses and 50 mature company responses. The results show that most metrics in start-ups require some hours to be assessed (42%). 31% of the metrics even need days to weeks to be assessed. Most metrics take one or more days (48%) to assess, and 34% of the metrics are assessed within hours (34%). Both start-ups (21%) and mature software companies (12%) comparatively assess a few metrics in seconds to minutes. For example, most companies use some hours or even one or more days to assess the metrics (*sales revenue*, *product/code stability*, and *speed of development*). Only (2/10) metrics in the aggregated *sales revenue* are assessed within seconds to minute.

Time used to set up the measurement tooling/infrastructure

Figure 7.9 shows the single-choice dimension *time used to set up the measurement tooling/infrastructure* for 81 start-up responses and 49 mature company responses.

In this dimension, most metrics in both start-ups and mature companies (51% and 57% respectively) take days to weeks to set up measurement infrastructure. In contrast, shorter set up times of a few minutes or several hours is comparatively less common in the companies. An exception is a *sales revenue* metric for which most metrics (7/10) required only minutes to hours to set up their measurement tooling. For (2/6) metrics, the mature companies (MC08, MC09) are not sure of the time they take to set up the measurement tools. In the *product/code stability* metric, (4/5) metrics required minutes to hours to set up their infrastructure, while the fifth company was not sure how much time it took to set up tooling for its metric. In contrast, (3/4) metrics in the aggregated *speed of development*, metric took days to weeks to set up their measurement tools. Start-up S10 was not sure of

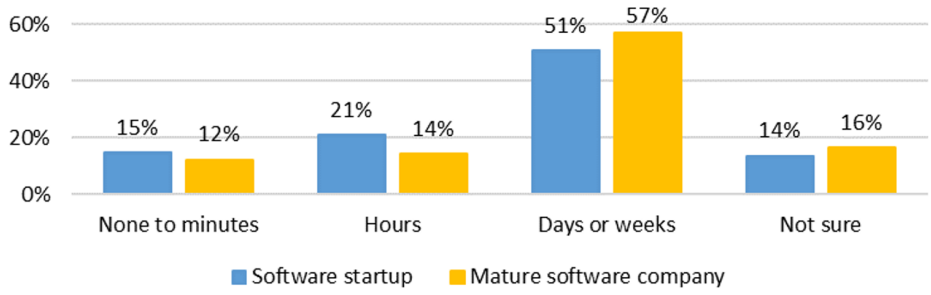

Figure 7.9. Distribution of metrics for the dimension "Time used to set up the measurement tooling/infrastructure".

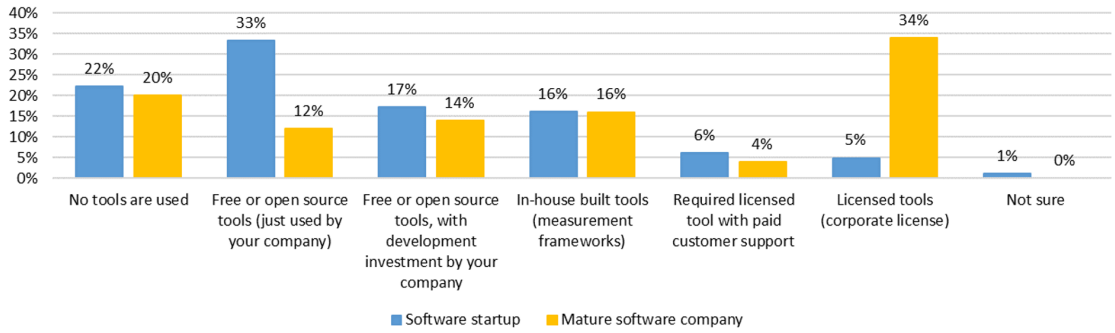

Figure 7.10. Distribution of metrics for the dimension "Type of investment in tools for assessing the metric".

the time it took to set up its tools for this metric.

Type of investment in tools for assessing the metric

Figure 7.10 shows the single-choice dimension *type of investment in tools for assessing the metric* for 81 start-up responses and 50 mature company responses.

For this dimension, start-ups mainly invest in free open source tools without modifications (33%). In contrast, most mature software companies (34%) invested in licensed tools with corporate licenses to assess their metrics. Both start-up and mature companies to a less extent invest in any tools at all (22% for start-ups and 20% for mature companies), invest in free and open-source tools with additional development investment (17% for start-ups and 14% for mature companies) or invest in in-house built tools (16% for start-ups and 16% for mature companies).

For the *sales revenue* metric, the four start-ups (S03, S08, S10, S16) invest in free open source tools or no tools at all to assess their four metrics for this aggregated metric. In contrast, most mature companies (5/6) invest in tools that require licenses without or with paid customer support to assess their five metrics. The only mature company (MC10) uses an in-house built tool for assessing this metric. The *product/code stability* metric is assessed by (3/4) start-ups using free open source tools (with or without additional development by the company). Only one S03 start-up uses a licensed tool. The mature company (MC01) does not invest in any tools for assessing *product/code stability* metric. The *speed of development* metric shows a wider variety of investment in assessment tools by the start-ups. This variety is exemplified by the investment in-house built tools by (S09), investment in free or open-source tools (without modifications) by (S15), and investment in free or open-source tools with development investment by the company by (MC01), and

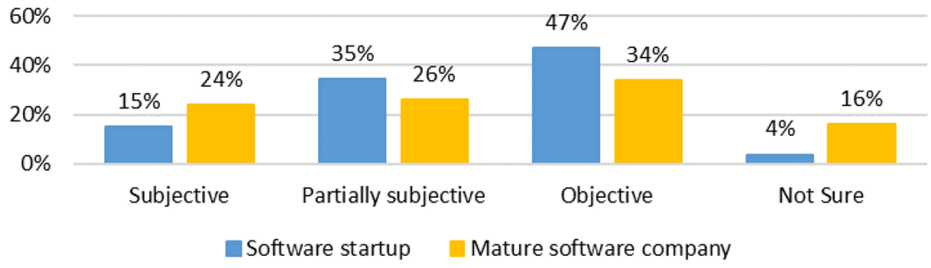

Figure 7.11. Distribution of metrics in start-ups and mature companies with regards to the dimension "Objectivity/subjectivity of the metric".

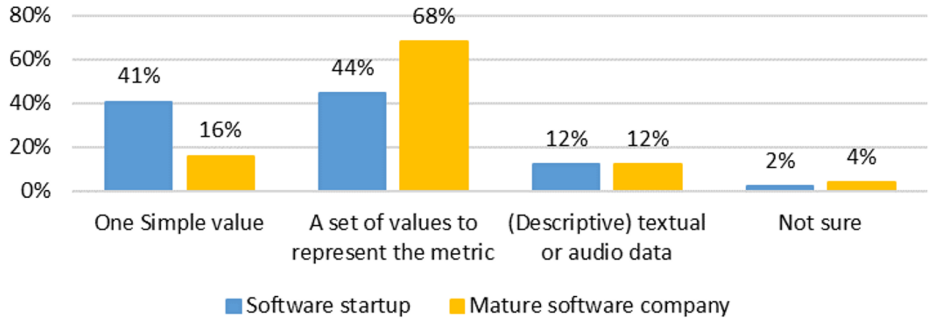

Figure 7.12. Distribution of metrics for the dimension "Form of the metric".

S10 was not sure.

Objectivity/subjectivity of the metric

Figure 7.11 shows the single-choice dimension *objectivity/subjectivity of the metric* for 81 start-up responses and 50 mature company responses.

The software start-ups mainly consider their metrics objective (47%) or partially subjective (35%). In comparison, mature software companies judge their metrics to be subjective (24%), partially subjective (26%), and objectively (34%).

For *sale revenue* metric, the underlying metrics are almost equally considered by companies as subjective (3/10), partially subjective (3/10), and objective (3/10), with only MC05 not sure on how to judge it. The *product/code stability* metric has its underlying metrics judged as partially subjective (3/5) and objective (2/5). The *speed of development* metric is objective and subjective for one company and in each of the other companies (MC05, S09) of this metric.

Form of the metric

Figure 7.12 shows the single-choice dimension *form of the metric* for 81 start-up responses and 50 mature company responses.

In this dimension, software start-ups evenly use metrics of the form that consist of a set of values (44%) and one simple value (41%). In contrast, mature companies majorly use metrics that consist of a set of values (68%). Comparatively few metrics are represented descriptively, i.e., as textual or audio data, in software start-ups (12%) and mature companies (12%).

The *sales revenue* metric in all mature software companies takes the form of one simple value or a set of values for (6/6) metrics. However, the start-ups' metrics take various forms (one simple value, a set of values representing the metric, descriptive (text or audio)). The *product/code stability* metric takes the form of a simple value in the mature company and 2/4 start-ups, while in the other (2/4) start-ups, it takes a set of values. The *speed of development* metric takes its form as a set of values for (3/4) metrics in start-ups and the company, while start-up (S09) is not sure of the form of its metric.

7.3.6 RQ2: Which dimensions show significant differences between metrics used in start-ups and mature companies?

Table 7.7 presents chi-square test results on dimensions with multiple-choice responses. Chi-square tests only apply to independent values and thus not suitable for multiple choice-responses. Therefore, in Table 7.7, we tested each type of response in the multiple-choice response dimensions plus a Bonferroni [153] correction. The Chi-tests did not reject the hypotheses for the single responses except $H03_2$ (rejected). The rejection of $H03_2$ after the Bonferroni correction implied a significant difference between technical users of a metric in software start-ups and mature software companies.

Table 7.8 presents the Chi-tests for metrics in single-choice dimensions. The initial test, including "not sure" as a category, did not rejected five ($H0_4$, $H0_5$, $H0_8$, $H0_9$, and $H0_{10}$) of the ten hypothesis as shown in Table 7.8. We performed a second Chi-test, excluding those data-points to ensure the results were not impacted by differences in the number of "not sure" responses. The initial test rejected $H0_9$ with $p = 0.003005$ yet the second test did not reject it with a $p = 0.2261$. The second test confirmed that the significant result seems to be caused by the high difference of "not sure" responses, which are only 4% in start-ups and 16% for mature companies. For the four rejected hypotheses ($H0_4$, $H0_5$, $H0_8$, $H0_{10}$), the p-value without "not sure" votes was even smaller "not sure" response in these cases did not contribute to the observed difference between start-ups and mature companies.

The radar chart in Figure 7.13 visualises the four dimensions (D04, D05, D08, D10) that were significant (rejected) in Table 7.8. The chart indicates the responses in each of the four dimensions for the start-ups and mature software companies. Each axis represents a dimension with its accompanying scale, as defined in Table 7.2. The lines representing start-ups and mature software companies meet the axes at responses that dominated that axis representing a dimension. For example, in the dimension D08 (investment in tools), start-ups generally invested in free open source tools while the mature companies invested in mainly licensed tools.

Characterization for distinguishing dimensions with running examples

For our four running examples, Figure 7.15 shows how the aggregated *sales revenue metric* manifests itself for the four distinguishing dimensions amongst the four underlying metrics listed by the start-ups. Similarly, Figure 7.14 shows for the same dimension, six underlying metrics in the mature software companies.

The Figure 7.16 shows a combined radar chart for 5 start-ups metrics and one mature software company metric for the *product/code stability metric* in the 4 distinguishing dimensions. Figure 7.17 shows the characterisation of the *speed of development metric* for 3 start-ups metrics and one mature company metric in the 4 distinguishing dimensions.

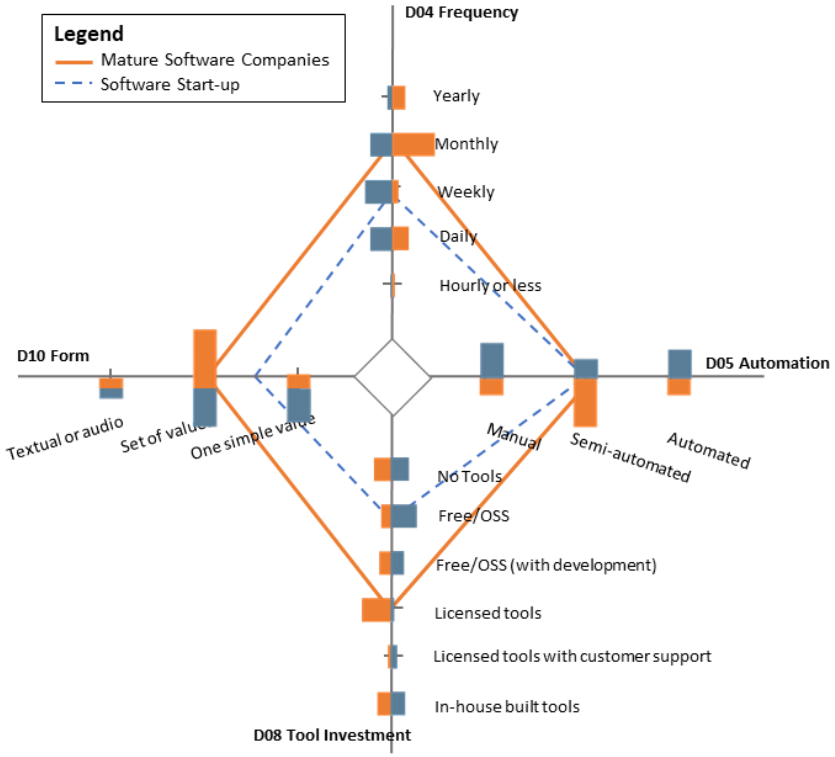

Figure 7.13. The four differentiating dimensions for metrics in start-ups and mature software companies.

7.4 Discussion

In the following sections, we discuss the findings from our results.

7.4.1 Non-distinguishing dimensions

This subsection discusses the dimensions that did not show significant differences between software start-ups and mature software companies.

Use of the metric: The start-ups and mature software companies mainly used metrics for insight and decision making. This finding is consistent with the findings by Staron [152] that highlight the critical role of measurements in decision making in large software organizations. That start-ups, too, using metrics for decision making is conceivable. Given their short operating history, the start-ups may have to make decisions. For example, they make pivoting [156] decisions and gain new insights as they validate their learning [157] to improve their business models and products.

Phases of metric assessment: An interesting observation about this dimension is that a considerable number of metrics are used in multiple phases. This use of metrics across multiple phases may indicate that metrics are mainly picked for purpose rather than its phase. For example, metrics are used for purposes like software quality improvement [158], measuring of product development cycle time [159], knowledge management [160], project performance measurement [161], measuring processes [162], and measuring user experience [163]. The purposeful use can also be seen in the aggregated metrics as well. For example,

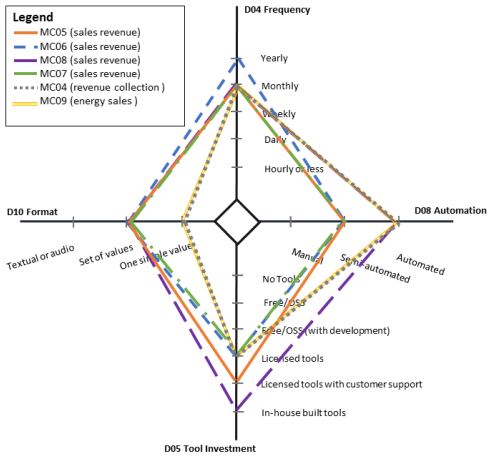

Figure 7.14. Characterisation of mature companies for the *sales revenue* metric for the 4 significant dimensions

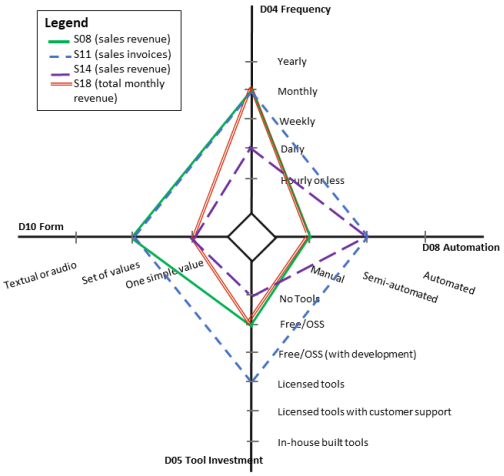

Figure 7.15. Characterisation of start-ups for for the *sales revenue* metric for the 4 significant dimensions

they measure profitability using metrics like gross margin, net profit margin, etc.

Users of the metric: Most of the companies' metrics are used by more than one team with little use by specialists. The use of metrics in multiple roles indicates that the metric might be reused to provide insight and enable technical and business teams' decision-making. The even use of metrics by business and technical teams in start-ups might further be a symptom of a multiple-role play as start-ups are characterized by small and usually inexperienced teams[12]. The slight use of more business team metrics than the technical team in mature software companies potentially points to their broader focus on business aspects than product development in mature companies. The products are generally mature, and mature companies are looking for more customers to use them. Finally, it is interesting to see that start-ups have nearly the same share of specialists' metrics used as mature companies with more expertise. This similar share of metrics in specialists' roles could indicate that metric use is not spread among specialists in both start-ups and mature companies.

Time used to assess the metric: Although not a significant difference, mature software companies tend to spend more time (one or more days to weeks) to assess their metrics compared to software start-ups that mainly spent some hours. Due to their longer operational history, mature companies may be measuring more aggregated and complex metrics that take more time to assess. Due to the smaller teams [9], inexperience in measuring, software start-ups would naturally assess more specific metrics that are easier and take less time to measure. The metrics that take only minutes or seconds to measure might not be of much value to mature companies in the longer term. The lack of interest in short duration metrics could be that companies make decisions after considerably longer reporting windows using metrics assessed over more extended periods.

Time used to set up the measurement tooling/infrastructure: For both start-ups and mature software companies, setting up the initial tools/infrastructure for measuring metrics most often takes days or weeks. This again seems to hint at an inherent character-

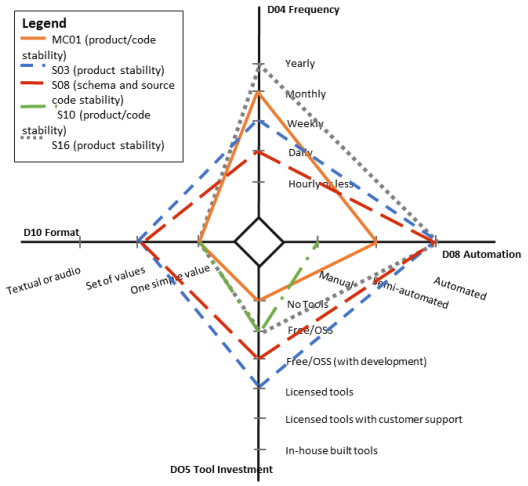

Figure 7.16. Characterisation of start-ups and mature companies for the *product/code stability* metric for the 4 significant dimensions

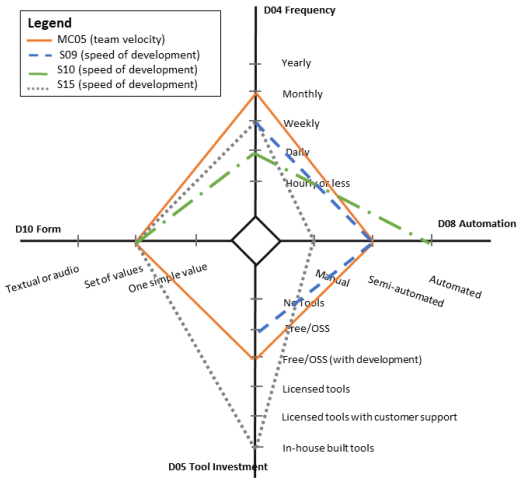

Figure 7.17. Characterisation of start-ups and mature companies for the *speed of development* metric for the 5 significant dimensions

istic/challenge of measurement tooling. Given that start-ups usually are smaller in size [9], focus on building a product before funds run out [2] and inexperienced in measurements (seen by their low adoption of measurement programs[56]). Mature software companies may not necessarily face the same challenges but take equally similar time-frames in setting up measurement tools. The lack of difference in the timeframes used to set-up tools in start-ups and mature software companies could point to companies' inherent time-consuming nature setting up measurement infrastructure.

However, this dimension had many "not sure" responses, indicating a problem with the used scale. In hindsight, it would have been good to introduce a finer granularity in the "days or weeks" category. For example, we should have split up days or weeks into two scales or even introduced a month scale. Doing so might reveal interesting insights for future work.

Observation 1: We could not find significant differences between metrics in software start-ups and mature software companies for six dimensions. The dimensions included, use of the metric, assessment phase for a metric, users of the metric, the time used to assess the metric, and the time used to set up tooling for assessing the metric.

Objectivity/Subjectivity of the metric: Start-ups mainly used objective and partially subjective metrics, whereas mature software companies (even though following the same trend) had a more even distribution of their metrics for this dimension. The companies' stronger emphasis on objective metrics seems logical and desirable as the companies can make more objective decisions that are data-driven with such metrics. However, the presence of subjective and partially subjective metrics indicates that this is not always possible. Thus potentially pointing to the subjective nature of the underlying data used to obtain the metrics.

The considerable number of *"not sure"* responses for mature software companies in this dimension indicates an interesting bias. The mature companies seem to either have more metrics that are difficult to classify along this dimension's scale or are more careful about

doing so. However, the results make it attractive for future studies to investigate the factors influencing companies to use subjective or partially subjective metrics.

Observation 2: The classification of a metric as objective or subjective seems to be non-trivial.

7.4.2 Distinguishing dimensions

This section discusses the four dimensions (D04, D05, D08, D10) with significant differences in start-ups and mature software companies.

Meaningful frequency for assessing the metric: The measurement within short time windows by start-ups hints at the need by start-ups to frequently make decisions based on monitoring their often ad hoc operations. They thus need regular insight to make decisions. These decisions may be driven by the multiple influences that include pressure to deliver a minimal viable product to the customers, limited availability of funds, and as Sutton [9] highlights, pressures from investors, customers, partners, and competitors. Due to their short operating histories, start-ups in early stages are assumed to be learning from their experience as they refine their technical and business models [9].

In contrast, mature companies can probably afford a longer interval of assessing their metrics as they have the benefit of history, potentially routine operations, stable organizational structures, and mature practices [164]. Thus regular monthly reporting may incorporate the use of the metrics.

Automation of the measuring process: Start-ups mainly measure metrics manually or automatically with considerably less use of semi-automated assessments. On the other hand, mature software companies primarily assessed their metrics semi-automatically with few manual and few automatic evaluations.

The higher use of manual measuring processes in software start-ups could be attributed to a lack of funds to invest in buying licenses for tools or a lack of time and skills to measure some of the metrics. The considerable automation by start-ups, on the other hand, could be influenced by the availability of free, open-source tools. The predominant use of semi-automated measuring processes in mature companies is a more surprising observation. One possible explanation could be those mature companies potentially have bigger codebases that are built using varying technologies. Thus, available measurement tools may not fully accommodate the complex workflows in mature companies.

Type of investment in tools for assessing the metric: The majority of metrics in start-ups are assessed using free open source tools. For start-ups, the main reason might again be limited funding. An earlier study [10] of the practices and patterns in start-ups in Uganda highlighted that the bootstrapping model (funding by founders) is expected. Other funding sources include small grants to seed funds.

The mature software companies mainly invest in licensed tools to assess their metrics. The investment in licensed tools maybe because of the complexity or high granularity that may require premium tools to measure metrics effectively. The open-source tools, too, might be considered risky as their terms of use may require the companies to reveal competitive business practices. It was interesting to see that both start-ups and mature companies rely

to a similar extent on in-house built tools. The companies' reliance on in-house tools could point to developers' availability who have the skills to make the required tools cheaply in-house. Future work could investigate the metrics and motivation for which tools are created in-house for companies.

Form of the metric: The majority of software start-ups use simple value and metrics represented by a set of values. The simple values may meaningfully represent the start-ups' entities and can be easily understood and interpreted by founders in decision-making. In contrast, mature companies represented most of their metrics as a set of values pointing to derived metrics using base metrics. These values may be more representative of the complex processes and entities in mature software companies. Besides, mature companies may need more differentiated assessments. For example, a mature company may not be just interested in the number of customers and related aspects like customer satisfaction, customer's ability to spend, or customer acquisition cost.

Observation 3: Differentiating dimensions in metrics in start-ups and mature companies include:

1. The daily, weekly, and monthly assessment of metrics in start-ups in comparison to the main monthly assessment in mature companies,
2. The start-ups mainly assessing their metrics manually or automatically with minimal use of semi-automated assessments while mature companies mainly assessing semi-automatically,
3. The start-ups investing in free open source tools while the mature companies mainly investing in licensed corporate tools, and finally,
4. The metrics in start-ups are being represented more by one simple value, while mature companies preferred representing their metrics as a set of values.

7.4.3 Lessons learned and future work

This subsection presents lessons learned as well as implications for future research.

No reason not to learn from mature companies: Start-ups and mature software companies use metrics in business and technical teams for insight and decision making. All companies spend some hours to weeks to assess objective and partially subjective metrics across multiple phases. Also, all companies take days to weeks to set up the measurement tools for the metrics. Start-ups usually have short or no operating history and have limited resources, yet this survey indicates that they still use some metrics found in mature software companies. All in all, six of the dimensions do not include characteristics that would prevent start-ups from using a metric. Thus, start-ups can learn from mature companies. For example, metrics used in both start-ups and mature software companies, the start-ups have a chance to learn from the mature company's experience to assess the metric manually until a tool is found and affordable.

Metrics might get more complicated as the start-up grows: More metrics were found in start-ups than mature companies. Start-ups thus seem to benefit from the simplicity of their operations that allow them to assess more metrics in the form of simple values. Mature companies, however, share metrics that consist of sets of values. The use of sets of values in mature software companies may point to the evolution of metric use

as start-ups mature. They may need to adopt and adapt more compound metrics. This evolution could necessitate future work to investigate what motivates start-ups to adopt more complex metrics as they grow.

Consequently, developers and scientists should not make assumptions about the use of simple metrics in mature software companies. The use of complex metrics is in conformance with a study by Kitchenham et al. [89] that reported the need to construct productivity measurements as functions of various measures to capture different aspects of the size of software systems.

Be aware of higher frequency of metric assessment: Start-ups seem to assess their metrics more frequently than mature software companies in Uganda. Metrics require only hours or even days to assess in start-ups. The short duration for assessing metrics is a surprise as intense time pressure to deliver an MVP and the often small teams in start-ups may make metrics that require frequent assessments unattractive. However, the need for closer monitoring and regular decision making using such metrics in start-ups may compel them to use open source tools (if they exist) to automate such metrics. Future work may need to investigate why the start-ups invest time to assess metrics and their value.

Don't assume full automation of metric assessment: While it is easy to imagine that start-ups might not fully automate their measurement process; we observed that even mature companies assessed a number of their metrics semi-automatically. As discussed in Section 7.4.2, the high semi-automation of metric assessment in mature companies might point to the increasing complexity of routines and systems. We should carefully consider the semi-automated assessment of metrics in a company when developing metric-based approaches because it is a seemingly common approach for assessing metrics. For example, the defect prediction approach by Singh et al. [150] relies on automatically collected a large amount of training data like object-oriented metrics. Our results indicate that such an automated data collection might not be a given in all companies.

Directions for future research on metrics and metric uses: Our results point to three additional avenues for future work. Firstly, this study focused on the general characteristics of all metrics in start-ups and mature software companies. The dimensions could differentiate the metrics. However, it would be interesting for future work to explore how the different groups of metrics in the MeSRAM categories differ using the dimensions. Besides, we found that there seem to be further factors in the dimensions that could influence the instantiation of metrics in software start-ups and mature software companies. Future work should focus on identifying these factors. Finally, we found that mature companies used more complex metrics consisting of sets of values than the simple one-valued metrics used by software start-ups. Future work should investigate how mature software companies could fully automate the seemingly complex metrics assessment as they currently assess these metrics using semi-automated to automated means. These running examples also illustrate the versatility of metrics through dimensions. Future research can also investigate how versatile the metrics are within the ten dimensions to highlight the dimensions' influence on start-ups and mature companies' metrics' choice.

7.5 Threats to Validity

We use the following classification scheme by Runeson [116] and Yin [165] to discuss the validity of this pilot study.

Construct validity The main threat to construct validity was the misunderstanding of the metric categories for which we solicited metrics and the use of the ten dimensions to characterize them. The online survey adopted metric types from MeSRAM [1] for respondents to identify and recall the metrics they use. We provided descriptive preambles explaining each MeSRAM metric category. To increase the ten dimensions' understanding, we asked the start-up during the instrument's derivation to give feedback about the dimensions. We incorporated the feedback into the hardcopy questionnaire and the final survey. To increase the chance of getting valid responses to the online survey, we encouraged some start-ups and mature companies that we physically met to clarify the metric groups and MeSRAM categories. Although we cannot guarantee to capture all the feedback in the trial phases in the online survey, we are confident that we captured the most critical feedback and increased the chances of responding to the online tool.

Internal validity: The pilot survey lacked explicit inclusion and exclusion criteria for the respondents. For example, there was no question to ascertain the respondents' level of business or technical experience. The potential inexperience could have compromised the adequate characterization of the metrics using the ten dimensions. We mitigated this by interviewing founders in start-ups and technical managers in each mature company. The mitigation may not have been foolproof; however, it ensured that we did not discount the experience to use metrics and dimensions.

External validity The study used a non-probabilistic quota sampling approach, which is [122]. The small sample size as a concern, too, may not give statistical representation for software start-ups and mature software companies either in Uganda or East Africa. The relative sample size meant we could not make a statistical generalization to all the software start-ups in East Africa. However, we deem the motivation and derivation of the dimensions sufficient to motivate an extensive survey in the emerging East Africa ecosystem. Such a study will go a long way to validate the character of our ten dimensions results.

7.6 Conclusion

This study derived ten dimensions and used them to survey 19 start-ups and ten mature companies in Uganda. Results indicate that 6 of 10 dimensions can similarly characterize metrics in start-ups and mature companies. The similar characterizing dimensions indicated that, in companies, technical and business teams use objective and subjective metrics for decision making in multiple phases. The companies also take a few days to several weeks to set up measurement tools and generally spend hours to weeks to assess their metrics. Four of the ten dimensions can characterize differences in metrics used in start-ups and mature software companies. These dimensions highlighted that start-ups compared to mature companies 1) invest differently in tools, 2) mainly using free open source tools, assess metrics more frequently (daily, weekly, and monthly), 3) use more manual and automatic assessments than semi-automated assessment fro their metrics, and 4) use one-simple value metrics. The derived ten dimensions provide a starting point for start-ups to select their metrics objectively. Understanding the dimensions contributes to our understanding of the metrics used in mature software companies' metrics that can be adopted. Adapted or learned from software start-ups. Metric-based techniques make explicit assumptions about the metric data's granularity, frequency of assessment of metrics, simplicity of the metrics, and automation of the metric evaluation in areas such as cost estimation, process monitoring, or defect prediction may benefit from the results of this study.

Table 7.2. Metric dimensions, definitions and scales

Code	Dimension	Definition	Answer Scale	Answer Type
D01	Use of the metric	The use of the metric within the company	• Insight (to gain insight) • Decision making • Alert (red flag) • Not sure	Multiple choice
D02	Phase of metric assessment	The stage in the life cycle of the product or company where the metric is most likely to be measured	• Ideation/ Conceptualization • Design • Implementation / Prototyping • Validation • Deployment • Not sure	Multiple choice
D03	Users of the metric within the software company	The targeted audience, who interprets and uses the metric in the company	• Technical Team, e.g. architects, developers, software-, test-, and deployment engineers • Business Team, e.g. branding and marketing, business managers • Other specialists, e.g. accountants, lawyers, auditors • Not sure	Multiple choice
D04	Meaningful frequency for assessing the metric	What is a meaningful rhythm to assess the metric, i.e. how long does this metric take to change noticeably?	• Yearly • Monthly • Weekly • Daily • Hour or less • Not sure	Single choice
D05	Automation of the measuring process	The degree of automation of the process of assessing the metric in the company	• Manual • Automated • Semi-Automated • Not Sure	Single choice
D06	Time used to assess the metric	The amount of dedicated time used each time the metric is assessed	• Seconds to minutes • Some hours • One or more days to weeks • Not sure	Single choice
D07	Time used to set up the measurement tooling/ infrastructure	The time needed for the initial set-up of the measurement tooling/ infrastructure for this metric	• None to minutes • Hours • Days or weeks • Not sure	Single choice
D08	Type of investment in tools for assessing the metric	The character of the financial investment in tools made for assessing/ measuring the metric	• No tools are used • Free or open source tools (used as they are) • Free or open source tools (with development investment by the company) • Licensed tools (corporate license) • Licensed tool with paid customer support • In-house built tools (e.g. measurement frameworks) • Not sure	Single choice
D09	Objectivity/ subjectivity of the metric	The degree of subjectivity/objectivity of the metric	• Subjective: human opinions or noise are involved, leading to a degree of uncertainty in the result • Partially subjective: human opinions or noise are involved, but uncertainty is minimized, e.g. by using a high number of data points • Objective: Two or more independent assessments in the same situation lead to the same result • Not Sure	Single choice
D10	Form of the metric	The form of the data assessed when assessing the metric	• One simple value • A set of values to represent the metric • (Descriptive) textual or audio data • Not sure	Single choice

Table 7.3. Characterization of surveyed start-ups

			Type of software products/ services							
Start-up	**# Employees**	**Start-up Age in years**	**Software as a Service (SaaS)**	**Software development as service**	**Mobile Application**	**E-commerce**	**Fintech**	**Two-sided market place**	**User-generated content**	**Other**
S01	>10	>3	x	-	x	-	x	-	-	-
S02	2-5	>3	-	x	-	-	-	-	-	-
S03	6-10	1<=3	x	x	x	-	-	-	-	-
S04	2-5	>3	x	x	x	-	-	-	-	-
S05	2-5	1<=3	-	-	x	-	x	-	-	-
S06	2-5	1<=3	x	x	x	-	-	-	-	-
S07	2-5	1<=3	x	-	x	-	-	x	-	-
S08	2-5	1<=3	x	x	-	-	-	-	-	-
S09	>10	<1	x	x	-	-	x	-	-	-
S10	2-5	1<=3	-	x	x	-	x	-	x	-
S11	2-5	1<=3	x	-	-	-	-	-	-	-
S12	2-5	1<=3	-	-	-	-	x	-	-	-
S13	2-5	1<=3	x	-	-	-	-	-	-	-
S14	2-5	<1	-	-	-	-	-	-	-	x
S15	2-5	>3	-	x	x	-	-	-	-	-
S16	6-10	>3	-	-	-	-	-	-	-	x
S17	2-5	>3	-	x	x	-	x	-	x	
S18	1	>3	x	x	x	-	-	-	-	-
S19	>10	>3	-	-	x	x	x	-	-	-

Table 7.4. Characterization of the mature software companies

Company	# Employees	**C**ompany age (in years)	**T**ype of software services/products
MC01	<50	5-10	Financial Software and Consultancy
MC02	<50	5-10	Financial Solutions
MC03	<50	10-20	Banking and mobile applications
MC04	<50	5-10	Developing banking software, mobile money solutions/payments and e-payments such as utilities, pay TV among others.
MC05	<50	5-10	Develops banking software
MC06	50-100	10-20	Payment solutions, POS solutions, Banking Automation, Mobile Banking, Agency Banking and Internet Banking Solutions, Security Solutions
MC07	<50	5-10	Monitoring and Software development Mobile app Development Website Design and Development Business Support IT Consultancy
MC08	<50	–	–
MC09	<50	5-10	–
MC10	<50	5-10	Web and Mobile Software Solutions/Apps

Table 7.5. Null hypotheses for the ten dimensions.

Hypothesis	**Description**
$H0_1$:	*The use* of a metric does not differ between start-ups and mature software companies.
$H0_2$:	*phase of assessment of a metric* does not differ between start-ups and mature software companies.
$H0_3$:	*The users of a metric* do not differ between start-ups and mature software companies.
$H0_4$:	*The meaningful frequency for assessing a metric* does not differ between start-ups and mature software companies.
$H0_5$:	*The automation of the measuring process* of a metric does not differ between start-ups and mature software companies.
$H0_6$:	*The time used to assess a metric* does not differ between start-ups and mature software companies.
$H0_7$:	*The time used to set up the measurement tools/infrastructure for a metric* does not differ between start-ups and mature software companies.
$H0_8$:	*type of investment in tools for assessing a metric* does not differ between start-ups and mature software companies.
$H0_9$:	*The objectivity/subjectivity of a metric* does not differ between start-ups and mature software companies.
$H0_{10}$:	*Form of a metric* does not differ between start-ups and mature software companies.

Table 7.6. Aggregated metrics in start-ups and mature software companies

				Types of Metrics				
No.	**Aggregated Metrics**	**Mature companies**	**Start-ups**	**Business**	**Product**	**Project**	**Organisation performance**	**Design**
1	**Customer satisfaction**	MC01, MC02, MC04, MC02*	-	**X**	**X**	**-**	**-**	**X**
2	**Volume of Transaction**	MC04,	S19, S19*	**X**	**-**	**-**	**-**	**X**
3	**Customer royalty and Retention**	MC02, MC03	S01, S17, S04	**X**	**X**	**-**	**-**	**-**
4	**Sales revenue**	MC04, MC05, MC06, MC07, MC08, MC09	S08, S11, S14, S18	**X**		**-**	**X**	**-**
5	**Continuous Improvement**	MC01	S03, S16	**-**	**X**	**-**	**-**	**-**
6	**Cost of customer acquisition**	MC06, MC06*	S01, S08, S17	**X**	**X**	**-**	**-**	**-**
7	**Account activation and Usage**	MC08	S01	**-**	**X**	**-**	**-**	**X**
8	**Product maintenance**	MC10	S02	**-**	**X**	**-**	**-**	**-**
9	**Project progress**	MC01	S10, S15	**-**	**-**	**X**	**-**	**-**
10	**Project/product quality**	MC07, MC02, MC07*, MC10	S03, S16	**X**	**X**	**X**	**-**	**-**
11	**No. of executed tasks**	MC03	S04, S13	**-**	**-**	**X**	**-**	**-**
12	**Cost of the project**	MC04	S14	**-**	**-**	**X**	**-**	**-**
13	**Profitability**	MC05	S01, S02, S14	**X**	**-**	**X**	**X**	**-**
14	**Customer and financial perspective**	MC01	S03, S10	**-**	**-**	**-**	**X**	**-**
15	**Turnover rates of employees**	MC02	S01	**-**	**-**	**-**	**X**	**-**
16	**Cost effectiveness**	MC05	S12	**-**	**-**	**-**	**X**	**-**
17	**Product/code stability**	MC01	S03, S08, S10, S16	**-**	**-**	**-**	**-**	**X**
18	**Speed of development**	MC05	S09, S10, S15	**-**	**X**	**-**	**-**	**X**

Note: aggregated metrics might differ in details and are not necessarily exactly the same. We used the metrics' names for identifying similarities. * Company named two related metrics in different metric categories and characterised each with the 10 dimensions.

Table 7.7. Chi-tests for dimensions with multiple-choice responses and their effect size.

Dimension	**Null Hypothesis**	**Multiple choice**	**p-value**	**Bonferroni correction for p-value threshold**	**Result**	**Cramer's V**
D01 Uses of the metric	$H01_1$	Insight (to gain insight)	0.3122	0.016667	$H01_1$ Not Rejected	0.087963
	$H01_2$	Decision making	0.02735	0.016667	$H01_2$ Not Rejected	0.192054
	$H01_3$	Alert (Redflag)	0.5114	0.016667	$H01_3$ Not Rejected	0.057151
D02 Phase of use a metric	$H02_1$	Ideation	0.07697	0.01	$H02_1$ Not Rejected	0.153934
	$H02_2$	Design	0.1096	0.01	$H02_2$ Not Rejected	0.134264
	$H02_3$	Implementation	0.1778	0.01	$H02_3$ Not Rejected	0.117301
	$H02_4$	Validation	0.7302	0.01	$H02_4$ Not Rejected	0.028938
	$H02_5$	Deployment	0.8652	0.01	$H02_5$ Not Rejected	0.01478
D03 Users of the metric	$H03_1$	Business Team	0.8357	0.016667	$H03_1$ Not Rejected	0.018049
	$H03_2$	Technical Team	0.001919	0.016667	$H03_2$ **Rejected**	0.270034
	$H03_3$	Specialists	1	0.016667	$H03_3$ Not Rejected	6.44E-17

Note that we use Bonferroni corrections, since we test every dimension multiple times (once per answer category, except "not sure" answers).

Table 7.8. Dimensions with single choice responses in start-ups and mature software companies and their effect size.

Dimension	**Null Hypoth.**	**Chi-square**	**p-value**	**Result**	**Cramer's V**	**Effect size**
D04 Meaningful frequency for assessing the metric	$H0_4$	$\chi_2 = 15.413$	$p = 0.008735$ ($p = 0.003993$)	$H0_4$ **Rejected**	0.3417129	Large
D05 Automation of the measuring process	$H0_5$	$\chi_2 = 17.966$	$p = 0.0004471$ ($p = 0.0002794$)	$H0_5$ **Rejected**	0.3703279	Large
D06 Time used to assess the metric	$H0_6$	$\chi_2 = 4.356$	$p = 0.2255$ ($p = 0.1135$)	$H0_6$ Not Rejected	0.182351	Medium
D07 Time used to set up the measurement tooling/ infrastructure	$H0_7$	$\chi_2 = 1.2909$	$p = 0.7313$ ($p = 0.5725$)	$H0_7$ Not Rejected	0.09965025	Medium
D08 Type of investment in tools for assessing the metric	$H0_8$	$\chi_2 = 23.107$	$p = 0.0007615$ ($p = 0.0004343$)	$H0_8$ **Rejected**	0.4183925	Large
D09 Objectivity/ subjectivity of the metric	$H0_9$	$\chi_2 = 8.9437$	$p = 0.03005^*$ ($p = 0.2261$)	$H0_9$ Not Rejected*	0.2612897*	Large*
D10 Form of the metric	$H0_{10}$	$\chi_2 = 9.497$	$p = 0.02336$ ($p = 0.009622$)	$H0_{10}$ **Rejected**	0.269251	Large

The respectively second p-values written in brackets indicate the results from the tests excluding "Not sure" answers, where existing. The result marked with * was not confirmed by the second test.

Chapter 8

Paper 5: Requirements for Measurement Dashboards and their Benefits: A Study of Start-ups in an Emerging Ecosystem

Chapter Overview

Metrics, often visualized with dashboards, are considered crucial to help software start-ups focus on the right aspects during the first years. However, earlier research indicates that metric choices in emerging ecosystems are not necessarily the same as in literature, mostly focusing on developed countries. More knowledge is required to provide dashboards that suit East African software start-ups. This study aims to identify key requirements for measurement dashboards in early software start-ups for monitoring the daily health of a start-up and how start-ups expect to benefit from the measurement dashboards. We performed semi-structured interviews with 36 software start-ups in Uganda and Kenya to identify and categorize requirements for measurement dashboards and hopes associated with the use of such dashboards. Our results show that most start-ups want measurement dashboards to visualize performance.

8.1 Introduction

Measurement is an activity that exists in the business and technical domains. Being an important activity that helps inform technical, business, and administrative decisions, domains such as manufacturing, health, and marketing have devised ways to standardize and simplify it by adopting tools. One of the most popular approaches to simplifying measurement is to employ dashboards. Several descriptions of dashboards exist, but generally, they are tools that provide reporting on data through visualizations and aid in informed decision making and monitoring organizations. Dashboards have a wide range of uses in companies that include quality monitoring [27], performance management [166] and measurements for design of large-scale systems [167]. Dashboards have been widely adopted in mature businesses to report sales, performance, and financial management. In software businesses, they are used for quality monitoring of processes and deployed systems. convener Our earlier study [48] in East Africa confirmed that early software start-ups are measuring aspects of their business and technical operations. We define software start-ups

in this study as small, newly created businesses that develop software-intensive products and services [4], within limited time frames [31] and are challenged by limited resources, multiple influences, dynamic markets, and technologies [9]. Many software start-ups are not using existing dashboards like Google Analytics, Facebook analytics to measure their progress in as much as they believe that measurements will benefit them. However, the few that use dashboards focus mainly on customer and sales-related measurements. Consequently, given their unique nature as young businesses with widely-known challenges [29, 18], existing dashboards may not be able to support many software start-ups with measurements that can help them gauge their progress.

Failure of software companies is attributed to an engineering perspective and a business perspective considering available resources, capabilities, and business activities. These two perspectives make start-ups unique because we cannot look at them with just the engineering and the business lens. Measurement standards and tools must consider the unique nature of start-ups. Measurement will enable the start-ups to realize the benefits of measuring experienced in mature engineering and big software companies. This study's primary goal is to collect and evaluate the requirements for dashboards that software start-ups can use to measure their progress and the perceived benefits of using these dashboards. This paper is organized as follows: In Section 2.10.5 we discuss general uses of dashboards in various domains, in Section 8.2 we describe the methodology we use, in Section 8.3 we present the results, in Section 8.4, we discuss the key findings of the study, in Section 8.5 we highlight the significant threats to validity of the study and finally, in Section 8.6, we present our conclusions and future work.

8.2 Research Methodology

This section introduces the key research questions for the study and describes the research approach we used to answer them. The following research questions guided us in this study.

- RQ1: What are the requirements for measurement dashboards in software start-ups?
- RQ2: How do start-ups expect to benefit from using measurement dashboards?

8.2.1 Interview instrument

To answer the research questions, we performed semi-structured interviews with the founders of 38 East Africa software start-ups. These interviews lasted 35-45 minutes each. The research presented in this paper builds on our previous work reported in [48]. In this paper, we focus on interview questions that were previously asked but not analyzed and reported in our earlier work, namely: (1) If you have a dashboard to show you daily progress on your business' different aspects, how would you want it to work? (2) What do you think will be the advantages/benefits of using dashboards to measure your start-up progress? We extended the earlier study with a second set of interviews whose aim was to identify more requirements for dashboards measuring progress in start-ups using the previous questions. We also aimed to further motivate the development of start-up dashboards for software start-ups in East Africa. The second set of interviews also had role-related questions to capture more information about start-ups to characterize them for future studies. Given that the start-ups were young and small, the start-up founders were still playing multiple roles, and they could answer these role-based questions. The role-related questions were based on the typical organizational roles found in Ugandan start-ups, for example, start-up

Table 8.1. Characterisation of Interviewed Start-ups

No.	Startup	Specialisations	Age (Years)	Hosting Space
1	**SHU**	Software as a service, software development	1<=3	Independent
2	**SCM**	Software development	1<=3	Independent
3	**S25**	Software development	1<=3	Co-working space
4	**SLM**	Software as a Service	6<=9	Co-working space
5	**SES**	Software development, training, consultancy	1<=3	Independent
6	**SBT**	ICT Services and training	1<=3	Independent
7	**SSO**	Software development, Software as a service, digital marketing	1<=3	Co-working space
8	**SJA**	Mobile app	<1	Co-working space
9	**SVI**	Software as service, Software development	3<=6	Co-working space
10	**SET**	Using software in their workflow	1<=3	Co-working space
11	**STG**	E-Commerce, Tech solutions	1<=3	Co-working space
12	**SG5**	Software development, ICT Services	1<=3	Incubation hub
13	**SWI**	Software development	6<=9	Incubation hub
14	**SYT**	ICT Services	3<=6	Hybrid
15	**SST**	Software Development	3<=6	Hybrid
16	**SBS**	Software solution	3<=6	Hybrid
17	**SDA**	Software solution	1<=3	Hybrid
18	**SSE**	Using software in their workflow	1<=3	Hybrid
19	**SCB**	Software as a service	1<=3	Incubation hub
20	**SCZ**	Software as a service	<1	Incubation hub
21	**SES**	Software as a service	1<=3	Incubation hub
22	**SCE**	Mobile app	<1	Incubation hub
23	**SWE**	Two-sided marketplace	1<=3	Co-working space
24	**SWT**	Two-sided marketplace	<1	Co-working space
25	**SNF**	Software as a service	3<=6	Co-working space
26	**SOI**	Using software in their workflow	3<=6	Incubation hub
27	**SOC**	Fintech	1<=3	Incubation hub
28	**SID**	Media site	1<=3	Hybrid
29	**SIH**	e-commerce	<1	Hybrid
30	**SIA**	Software as a service	<1	Hybrid
31	**SIR**	Mobile app	<1	Hybrid
32	**SIP**	Two-sided marketplace	1<=3	Hybrid
33	**SIN**	Fintech	1<=3	Hybrid
34	**SIS**	Software as a service	3<=6	Hybrid
35	**SRI**	e-commerce	3<=6	Incubation hub
36	**SIT**	Software Development	1<=3	Hybrid

founder, technical and business roles. For this study, we report and analyze the results for the two questions we asked in both sets of interviews.

Moreover, in the second set of interviews, one of our respondents preferred to fill in responses to the interview questions rather than give an interview. To address the open-ended character of questions in this particular case, we used free-text responses.

8.2.2 Start-up selection

In the interviews carried out in Uganda and Kenya, we used convenience sampling [140] and interviewed respondents from start-ups in incubation hubs, co-working spaces, and hybrid spaces have a close working relationship with Makerere University as well as known contacts. Of all the start-ups we interviewed, Only four are operating outside managed spaces.

We discarded two start-up interviews at the transcription stage due to the low quality of audio and duplication, leaving us with interview data from 36 start-ups. Table 8.1, characterizes the 36 start-ups highlighting what they do, where they are hosted, and their age. A reasonable number (26/36) of the interviewed start-ups were early-stage start-ups with an operational age of 3 years and below.

8.2.3 Interview execution

We carried out the first set of interviews in March and April 2018, where we interviewed respondents from 18 software start-ups in Uganda and Kenya. In Kenya, we used known contacts that led us to other start-ups. We then conducted the second set of interviews in Kampala, Uganda, during October 2018. We accessed some of the start-ups through their founders and others through snowball references. We used mobile phones to record each of the 18 interviews and used pen and paper to write down some extra but vital information.

8.2.4 Coding and analysis

We transcribed the interviews and coded them using Microsoft Excel. We created a matrix that had the start-ups as rows and the questions as columns. We used code analysis and derived the emerging codes for the two main interview questions. We aggregated the common requirements and identified start-ups that expressed them. We highlight the requirements and perceived benefits of using dashboards for measuring progress for the discussion of the results. We also compared the dashboard requirements for start-ups in East Africa to user requirements for performance management dashboards found in Finnish companies.

8.3 Results

In this section, we present results from the analysis of the 36 start-up interviews. Characterization of the start-ups can be found in Table 8.1. Note that we do not provide the full names of the start-ups to maintain their anonymity.

8.3.1 Dashboard requirements for measuring progress

To answer the first research question (RQ1), we coded the 36 transcripts. We found that 6 of the start-ups (SHU, SWT, SET, SJA, SRI, and S25) had various uses for their dashboards. SHU for presentation and analysis of staff and project performance, SWT for tracking retailer performance, S25 for showing daily progress, SET for customer feedback and analytics, SJA for analyzing sales and monitoring customer stock, and SRI to monitor revenue and customer acquisition.

Start-up S25 was content with the dashboard they were using. However, SWT planned to upgrade its dashboard to visualize performance and point out areas to improve. SRI desired some extra functions, and SHU used a dashboard for performance tracking and monitoring of staff and projects. Below are some verbatim quotes from respondents of start-ups SHU, S25, SWT, and SRI to question how they would use dashboards for measuring progress.

> "We have a dashboard. It shows expenditure, hours worked per individual, how long this project is taking and is it overdue. [...] We have a calendar that tracks what is to be done each day, " [Startup SHU]

> "We use Asana to show daily progress. We find it reliable" [Start-up S25].

> "Our dashboards are usually to track the performances of our retailers and wholesalers. [...] We do not have a dashboard that visualizes progress. We only have an input-based dashboard and are switching to a new one, Salesforce. With this one, we shall be able to visualize performance and areas where to improve and also give us alerts on what is not going correctly" [Startup SWT].

Table 8.2. Benefits of using a measuring dashboard

Benefits	**Start-ups**
p5cm**Resources**	
Enables planning	SCE
Resource usage monitoring	SCE
Resource saving	SCB
Prioritisation of resource allocation/Optimising limited resources	SOI, SIY
Shows staff performance	SES, SCB, SWT
Team	
Reminding of unfinished tasks	SCZ
Collection of team feedback	SWT
Team awareness of KPIs/bench-marking on KPIs	SIT, SIY
Provide motivation for team, founders	SWE, SOI
Performance	
Insight/Reporting for (quick accurate decision making, strategic direction)	SWT, SOI, SID,SIS, SIN, SIR, SRI
Project monitoring/track project status/progress	SCE, SIT, SCZ, SIY, SLM, SES, SOC, SIN, SBS
Helps in measuring growth	SIS
Know current status (project/start-up)	SHU, SCM, S25, SLM, SSN, SBT, SVI, SG5, SBS
Use progress as a reference to build trust	SSE
In process tuning/Improvement	SIH
Product	
Monitor service/product usage	SIA
Shows extent of solution coverage	SIA
Helps in product improvement	SJA
Identify areas for improvement	STG, SIY, SBS,
Investment	
Shows startup profile to investors and potential partners	SOC
Judge Market potential for Return on Investment	SCB
Provides statistical numbers to use for investment	SJA
Access more funding	SSE

> "I have a dashboard, and I use it in the morning, at night, all the time. I see revenue, I see the cloud, I customer acquisition. However, what I still do not have is how much money we have kept and known if the money correlates with what is in the bank. I would like to measure that. So, be honest with what the KPIs tell you and what they are measuring." [Start-up SRI]

In this section, we present requirements for all start-ups to describe how they use or would like to use a dashboard to measure their progress. We categorize the identified dashboard requirements into user, functional, and non-functional requirements [168]. To gain more insight into the discussion, we further grouped the requirements into focus areas for the dashboard features, including task, resource, performance, analytic, and presentation. These areas are: ***Task:*** features that aid a user to know, start, or complete an activity in the start-up. ***Resource:*** features that enable a user to monitor, measure, start-up resources like finances, infrastructure, staff. ***Performance:*** features that will enable the start-up to monitor, measure, and convey information about productivity and efficiencies in the start-up. ***Analytics:*** features that support users by presenting aggregated/derived information that can be used to make informed decisions in the start-up. ***Presentation:*** channels through which the dashboard will convey information to the users. Table 8.3 summarizes the identified dashboard requirements for the start-ups grouped into areas of focus.

The table shows that more than half of the start-ups (21/36) expressed at least one performance requirement for the dashboard. The next popular areas of focus for the requirements were presentation (13/36), task (11/36), analytics (12/36), and the least popular was resource (6/36).

Figure 8.1 shows the number of start-ups that expressed requirements in the different areas of focus for dashboard features. We found mostly functional and user requirements for the dashboard. Very few (4/36) expressed non-functional requirements, and these were mainly

Table 8.3. Requirements and their categorization

	Requirements Focus	Requirements	Startups Expressing the Requirement
User Requirements	**Task**	Use dashboard to schedule tasks/work to do	SCZ, SHU, SCM, SBT
		Setting time-frames/task completion times	SCZ, SWE, SWI, SBT
		Dashboard should show task progress status	SSO, SWI
		Monitor developer execution of tasks/goals	SIT
		List accomplished tasks/goals/work	SIT,SSO, SG5
	Performance	Use the dashboard to set up teams	SCZ
		Use the dashboard to set timelines and targets	SWE, SSO
		Reviewing performance by comparing achievement to set goals	SET
	Presentation	Reminders on what to focus/Show areas of focus	SOI, SBT
		Dashboard should visualize individual staff productivity	SNF
		Dashboard should visualize information in (graphs, charts)	SIP, SIN, SIS, SIY
		Visualize/Redflag points of failure/show areas of stagnation	SIP, SIS, SDA
		Colour coding progress indicators	SLM
Functional Requirements	**Resource**	Use the dashboard to monitor resource usage for example finances	SCE
		Monitor finances, Revenue management(income, expenditure, cost centers)	SIA, SIN, SBT, SHU
		Dashboard should have staff user accounts	SIT
	Performance	Measure product development progress	SES
		Measure revenue growth	SES, SWE, SIN, SIS, SRI
		Monitoring of different areas of business for efficiencies like product manufacturing and delivery time	SIH
		Performance tracking/monitoring for staff (workload, support needed, hours worked)	SNF, SIH, SBS, SBT, SHU
		Show and track work/project/company progress status	SOC, SIH, SHU,SCM, S25, SG5, SWI, SDA
		Reconcile KPIs with actual data	SRI
	Analytics	Track service/product usage (user/customer numbers, active users, number of product installations)	SCE, SWE, SOI, SIS, SJA
		What-if analysis (market, cost implications, ROI)	SIR, SST, SBT, SCB
		Show rate of customer acquisition	SIS, SRI
		Dashboard should show start-up areas with shortfalls and good performance	SES, SBT
		Use the dashboard to monitor status of the system and its user activities	SV1
	Presentation	Should have a filter to enter data manually without affecting the visual part of the work	SIP
Non-Functional Requirements	**Presentation**	Multiple platform access (desktop, web, mobile)	SCZ
		Dashboard should be Viewed by all team members	SOC
		Use the dashboard to present in startup pitches	SWE
		24/7 availability and accessibility to the dashboard	SRI

focusing on presentation.

The user requirements of the dashboard focused mainly on dashboard features for tasks (11/36) and presentation (7/36) and a few on performance (4/36). The dashboard focuses on tasks that included setting, scheduling, monitoring, and listing of tasks. The start-ups also expressed using the dashboard for the presentation of measurements. They wanted the dashboards to visualize the measurements using aids like graphs, charts, colors, and abstractions. However, the start-ups did not express any user requirements that focus on resources and analytics. The start-ups expressed most of their functional requirements with a focus on performance (17/36). They wanted to use the dashboard in measuring product

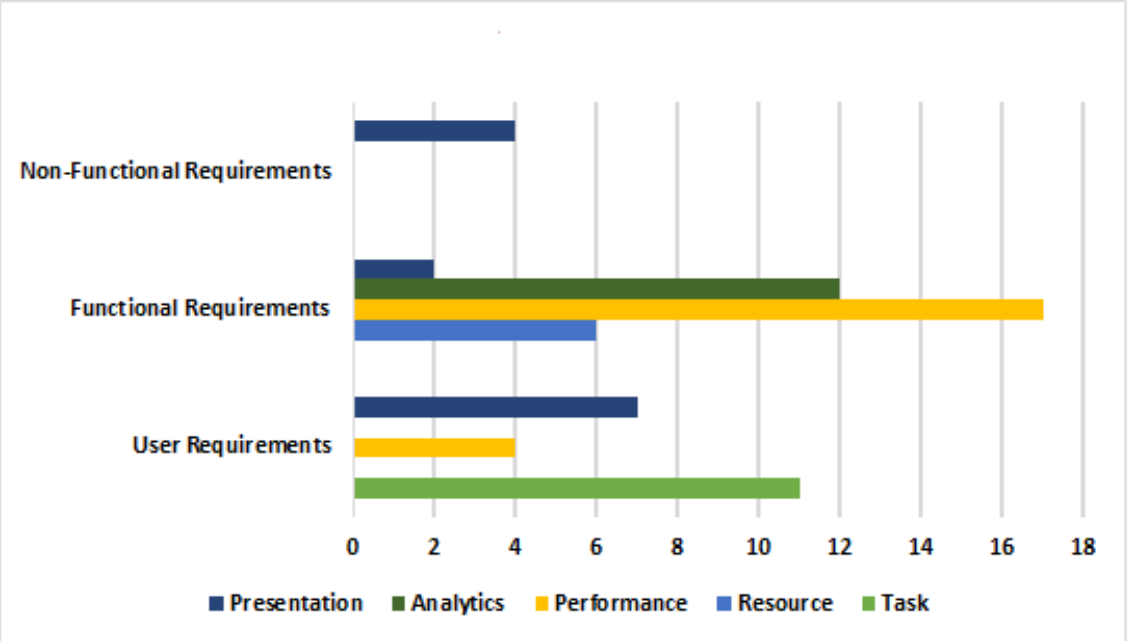

Figure 8.1. Focus of the Dashboard Requirements in the start-ups

development, revenue growth, monitoring efficiencies, tracking staff productivity, showing progress, and bench-marking progress with KPIs.

They also expressed a reasonable number of functional requirements that focused on analytics (12/36). The start-ups wanted to analyze product usage (customer numbers and activity, acquisitions, downloads), predictions (markets, return on investments (RoI)]) and show shortfalls and status of performance. However, few start-ups (6/36) expressed functional requirements focused on resources like their usage and finance (incomes, expenditure, and cost centers) and users. There were no task-focused functional requirements and very few presentation-focused functional requirements.

8.3.2 Expected benefits of using dashboards

This section presents the expected or realized benefits of using dashboards for measuring progress in start-ups. Start-ups are measuring, and some are experiencing the benefits of measuring as presented in an earlier study of the East African ecosystem [48]. However, very few start-ups indicated that they used dashboards, let alone other tools to measure their progress. Nevertheless, most of the start-ups were optimistic that measuring would benefit them [48]. In this research question, we asked what they expected to benefit from using dashboards to measure their progress. The results from this research question (RQ2) provide a sound basis for implementing dashboards to support measuring in early-stage software start-ups in East Africa. Table 8.2 presents the benefits expressed by start-ups expected/experienced using a measurement dashboard.

The results showed that most start-ups (24/36) expect/experienced performance-related benefits of using dashboards. The most popular performance benefits for start-ups included project monitoring/tracking (9/24), knowing of current project/start-up status (9/24), and insight (6/24). The popularity of the metric is in high contrast to the benefits related to resources (6/36), the team (6/36), product (5/36), investment (4/36), where start-ups averagely expressed different benefits. Different start-ups expressed the performance-related benefits in several ways. The performance-related benefits included insight/reporting for

decision making, measuring growth, knowing progress status, and process improvement. For example, the following start-ups had this to say;

> "It will also give us a better idea of how we are doing and then subsequently better insights on how we can do better" [Start-up SWT].

> "Analytical analysis: especially in decision making. You can look at the dashboard, and it guides you on what decisions to make and how soon, depending on past occurrences. Another is the strategy. [...] So, you notice that such a client comes once in eight months, so do not lose focus on the small client because those come every day. [..] " [Startup SOI].

> "Knowing whether you are going in the right direction and if you need to change. For example, when I look at our data, I know there are things we need to work on to increase our output. Second is our retention. [...] " [Startup SRI].

The following start-ups had this to say about the benefit of project monitoring/tracking for status/progress

> "It helps you track where the project is and then see where you are" [Startup SCZ].

> "[..] it will show the investors progress, for the time you have been in place and where you intend to go against your projections. So, you could have done side projections, and these are like your pathway. Now the dashboard will be showing how far you have come [..]" [Startup SOC].

> "Monitoring is the first benefit and evaluation would be done" [Start-up SIT].

> "It helps you monitor progress, not to relax, not to be reluctant. It helps you do more, aim higher" [Start-up SBS].

Others had this to say about the dashboard benefit of helping the start-up know its status of projects or as a company,

> "If you get up to go on a journey and you do not know to keep track of how far you have gone on that journey, how will you know when you have reached the destination. [...]

> "When you are everything, certain things will get lost somewhere between the lines.[...] I will never know how best I am moving forward" [Start-up SG5].

> "You know whether to go on or not, you know whether you are making money or not, you can know the status of your project, how much is your revenue for a certain period" [Startup SVI].

8.4 Discussion of Results

In this section, we discuss the study results regarding (RQ1) about the requirements for dashboards measuring progress in start-ups, and (RQ2) the expected/experienced benefits of using the dashboards.

8.4.1 RQ1: What are the requirements for measurement dashboards in software start-ups?

Our results indicated that most start-ups require measurement dashboards to support user and functional requirements for performance. They also wanted the dashboards to support the measurements' presentation, hinting that start-ups may want to get insight into their performance. They would also like the dashboards to present measures in ways that may give them easy interpretation and potentially help them in informed and fast decision-making. These are expressed by the fact that the popular presentation requirements included reminders on what to focus/showing areas of stagnation **(SOI, SBT)** and visualization/red-flagging failures and areas of stagnation by **(SIP, SIS, SDA)**. The popular requirements for start-up performance included measuring revenue growth **(SES, SWE, SIN, SIS, SRI)**, performance tracking/monitoring for staff **(SNF, SIH, SBS, SBT, SHU)**, and showing/tracking work/project/company progress **(SOC, SIH, SHU, SCM, S25, SG5, SWI, SDA)**.

Start-ups did not express non-functional requirements in the other start-up focus areas except for a few presentation requirements expressed by only (4/36) start-ups. The start-ups possibly focused on how the dashboard will measure performance, tasks and present them easily to interpret form instead of constraints on the dashboard's operation or operating environment. The start-ups only expressed user requirements for tasks with no task-related functional and non-functional requirements pointing to the fact that the majority of the start-ups were young and used task-based approaches to meet their start-up goals. They may be focusing on tasks/goals/work activities that lead to the development of a working product or minimal viable product (MVP) [143] [144]. Evidence of this could be that the majority of the start-ups (8/10) that expressed these requirements are young start-ups three years and below **(SCZ, SHU, SCM, SBT, SWE, SWI, SSO, SIT, SIY, SG5)**.

User requirements for presentations in dashboards included visualizations (color coding, graphs, charts), visualizing specific measures (staff productivity, areas of failure), reminders, and showing start-up focus areas. The preference for visualizations maybe because start-ups want to understand and interpret measurements and other areas of their operations to gain insight and make decisions. This potential use for visualization relates to one of the dashboards' purposes by [3] about decision-making and decision rationalization. It enables managers to extract relevant information on which they base their decisions justifying these decisions to themselves or their superiors.

Most of the start-ups expressed their functional requirements that focused on performance. The focus on performance hints at start-ups working under time pressure to deliver a Minimal Viable Product (MVP), with limited resources, and wanting early feedback from their activities to pivot. Thus, they may be eager to see growth, and dashboards can provide indicators of this. The dashboard will also allow them to interpret the measurements based on the fact that presentation-focused requirements were the second most popular requirement.

8.4.2 Mapping of start-up dashboard requirements to known performance management requirements

Velcu-Laitinen and Yigitbasioglu [3] mention that dashboards incorporate visual and functional features and are performance management tools. In their work, they point out that dashboards in performance management have evolved to provide features that include (1) real-time notifications and alerts, (2) scenario analysis, (3) drill-down capabilities, (4) presentation format flexibility, and (5) external bench-marking. They also highlight the four

primary purposes for the use of dashboards by businesses as (1) communication and consistency, (2) monitoring, (3) rationalizing of decisions, (4) problem-solving. Our results showed that most of the start-ups expressed performance and presentation requirements. The earlier work by Velcu-Laitinen and Yigitbasioglu [3] explored the use of dashboards for performance management. Their work slightly differs from ours in that it surveys Finnish companies and is not focused on software businesses or start-ups. Nevertheless, the performance dashboard's purposes and features, namely performance, and presentation have similarities with those from East African start-ups.

In this discussion, we identified the dashboard requirements and mapped them to these known features and purposes presented by [3]. Table 8.4, illustrates the comparison matrix for dashboard requirements in East Africa start-ups to those presented by Velcu-Laitinen and Yigitbasioglu [3] for Finnish companies. The mapping of the start-up requirements to available dashboard features and purposes on the broader business domain enables us to gauge the extent to which start-ups measure-up in terms of their requirements for dashboards. The mapping is important because it aids in comparing start-ups in a different ecosystem and geographical region (East Africa) to existing businesses in an established and more developed ecosystem in Finland. The mapping of dashboard requirements for start-ups to other businesses' dashboard requirements helps understand the generality and potential re-usability of dashboard requirements across businesses. The mapping could have positive implication for the design and development of start-up dashboards Examining the purpose and features of dashboards used for performance management in Finnish companies compared to software start-ups in East Africa has shown that communication consistency had the most (15/33) matching requirements for the dashboard. This result indicates that the dashboard's primary use is to communicate no matter the type of business. This assertion is reinforced by the study by Velcu-Laitinen and Yigitbasioglu [3] that indicates communication consistency ranked the highest amongst the purposes that Finnish companies had to use dashboards. Within communication consistency, presentation flexibility was the dominant feature that East African start-ups wanted, and it is the second with Finnish companies. This could be because most start-ups were not using dashboards, yet the Finnish companies were already using them, hinting to a better appreciation of the drill-down feature than the East African start-ups.

The communication consistency purpose also had the most number of requirements that we could not accurately map to the known features found in the Finnish companies. The popularity of this purpose could point to the fact that the comparative study focused on performance management, yet the start-ups had other interests for measurement dashboards besides performance.

Monitoring was the second most popular purpose (9/33) for the start-ups. It was also the second most popular purpose to use performance dashboards in the Finnish study. The most popular feature with the start-ups expressed was real-time notification and alert with almost all the start-up requirements for the dashboard falling here. No other known feature in the Finnish study had any matching requirements for this purpose in the start-ups. The popularity of this one feature could still be because the start-ups face challenges like lack of enough resources, need to acquire customers, short time to market, dynamic teams. They may thus be anxious to know and want quick feedback from the business.

Rationalization of decision making was the least popular purpose for start-ups, and the drill-down feature was the most expressed by the few start-ups that expressed this requirement. The lack of popularity for this purpose in start-ups could be because they are still small and have flat management structures where the founders make decisions. The start-ups are also still 'transparent' with simple activities compared to established businesses, creating little need to rationalize other decisions besides predicting where the start-up is

going by using actual data. However, there were two requirements in this category about measuring product development and measuring revenue growth, that we could not map to the known features of the performance management dashboards. Moreover, we did not find any requirements to map the performance management dashboard features to the problem-solving purpose. Similarly, we could not map some non-functional requirements for purposes and features given by Velcu-Laitinen and Yigitbasioglu's [3] work. For example, requirements like 24/7 availability and accessibility to the dashboard, and using the dashboard to present in start-up pitches,

Table 8.4. Comparison of start-up dashboard requirements with Key Functional Features of Performance Management Dashboards by Velcu-Laitinen and Yigitbasioglu [3]

Dashboard Features	Requirements identified by our start-ups
Communication and Consistency	
Drill down	Should have a filter to enter data manually without affecting the visual part of the work
Presentation flexibility	Dashboard should visualize information in (graphs, charts)
	Visualize/Redflag points of failure/show areas of stagnation
	Dashboard should be viewed by all team members
	Multiple platform access (desktop, web, mobile)
	Colour coding progress indicators
Scenario analysis	List accomplished tasks/goals/work
Real-time notification and alerts	Dashboard should show start-up areas with shortfalls and good performance
	Reminders on what to focus/Show areas of focus
Unclassified requirements	Use dashboard to schedule tasks/work to do
	Setting time-frames/task Completion times
	Monitoring of different areas of business for efficiencies like product manufacturing and delivery time
	Use the dashboard to set timelines and targets
	Dashboard should visualize individual staff productivity
	Use the dashboard to set up teams
Monitoring	
Real-time notification and alerts	Monitor developer execution of tasks/goals
	Monitor finances, Revenue management(income, expenditure, cost centers)
	Performance tracking/monitoring for staff (workload, support needed, hours worked)
	Show/Track work/project/company progress status
	Track service/product usage (user/customer numbers, active users, number of product installations)
	Use the dashboard to monitor status of the system and its user activities
	Dashboard should show task progress status
Unclassified requirements	Use the dashboard to monitor resource usage for example finances
	Dashboard should have staff user accounts
Rationalization of Decision	
Drill down	What-if analysis (market, cost implications, ROI)
External benchmarking	Reviewing performance by comparing achievement to set goals
	Reconcile KPIs with actual data
Unclassified requirements	Measure product development progress
	Measure revenue growth
	Show rate of customer acquisition

8.4.3 RQ2: How do start-ups expect to benefit from using measurement dashboards?

Start-ups measure various aspects of their businesses. A study by [48] indicated the benefits of measuring in start-ups. However, given the known challenges the start-ups face [18] [12] [29], use of measurement tools like dashboards may benefit them. To evaluate the benefits of using measurement dashboards, we grouped them into five broader themes: resources, team, performance, product, and investment. We use the themes as a basis for this discussion. Our results indicated the requirement for performance-related feautures, thus pointing to the need to create a Minimal Viable Product [169] and validating it. Thus, start-ups may desire to see the first results of their early effort by monitoring and tracking them using dashboards to know their progress status and improve or pivot.

Besides the performance benefits that several start-ups expected from measurement dashboards, there was a wide variety of benefits expected by start-ups concerning resources, teams, products, and investment. The variation in benefits could point to either lack of known measurement standards or common measurement systems for those aspects of the start-ups, thus the variation in knowledge about their performance measures.

8.5 Threats to Validity

To present the threats to the validity of our results, we follow guidelines from Yin [115].

Construct validity The main threat was asking the wrong or vague questions in the interview. We iterated the interview guide between the three authors and tested the guide with four start-ups in the first round of interviews to mitigate this. We then proceeded to ask the questions to the first 18 start-ups. We asked the same study questions about measurement dashboards and their benefits to the 18 start-ups for the second set of interviews.

Internal validity In the second set of interviews, one start-up (SCM) opted to answer the question on paper instead of participating in an interview due to lack of time. The results from this were captured in the coding sheet. Given that only one start-up coding, the script may not have had a significant bearing on the analysis.

External validity Of course, a bigger sample size would increase confidence in the results. It is also not clear to what extent the results can be generalized to the broader ecosystem outside East Africa. However, we are confident that the sample represents a reasonable geographical scope given that we carried out interviews in Uganda and Kenya and on different start-ups at different periods. The comparison of dashboard requirements for performance management from businesses in Finland by Velcu-Laitinen and Yigitbasioglu [3] with dashboard requirements from software start-ups in East Africa may have made some assumptions about the two contexts and thus provide a gap inconsistency of analysis. However, this was mitigated by closely mapping the start-up dashboard requirements using two dimensions of purpose and feature as provided in [3]. We believe that these two dimensions helped us to map the found dashboard requirements closely.

8.6 Conclusion and Future Work

In this study, we interviewed 36 start-ups. Most of them were measuring but not using dashboards to show their progress using the measurements. The start-ups show that they would like to use measurement dashboards for mainly performance aspects and show the status of their progress by visualizing these measures. They expect to mainly gain benefits related to performance when using the measurement dashboards, which is in line with other businesses studied elsewhere.

Performance and its visualization are the main priorities for the start-ups. To this effect, some start-ups have adopted existing tools, but using these tools by start-ups is still very low. The results also showed that current dashboard use for performance management in Finnish companies [166] can still be realized by the measurement dashboards in software start-ups. However, some of the dashboard requirements expressed by start-ups could not

be sufficiently captured by the features of the dashboards used for performance management, as seen in Finnish companies [166]. These different requirements for dashboards in East Africa start-ups highlight a need for a more integrated dashboard to capture performance and other aspects unique to software start-ups. The expression of varied requirements and expected benefits coupled with the current low use of measurement dashboards points to a gap in the tooling for measurements in the start-ups.

Dashboards have been applied in mature companies but are not commonly used in start-ups as they perceive them beneficial. This paper demonstrates that, despite this, there are needs for dashboards for various reasons, and the start-ups do perceive the benefits of using them. We are currently working on a prototype for integrated progress measuring dashboard that considers the unique character of early start-ups and contextual influences in the East African ecosystem. This prototype was to be validated in future longitudinal studies with early-stage start-ups in East Africa.

Chapter 9

General Discussion

9.1 Chapter Overview

This chapter presents a wrap up discussion of the results from the five studies published in this thesis. We highlight the main research gap, position the study and discuss the emerging results as we highlight their implications in relation to software engineering, software start-ups, and the emerging East Africa ecosystems.

9.2 Positioning the Research

Start-ups as businesses based on innovative software products and services are an important component of national economies as they create jobs through addressing socio-economic challenges in their ecosystems. Through the development of innovative products and services, software start-ups can support the transformation of industrial sectors in terms of new technologies and ways of working, and they can promote sustainable growth in a region. Given that about one in ten start-ups will be successful in developed ecosystems, the question is how make many of the start-ups emerge and grow into mature firms within a given region. The question is therefore how to generate as many start-ups as possible in a region in order to achieve sustainable growth or sustainable transformation.

Start-up hubs are a mechanism to promote early-stage software start-ups and generate new ones through provision of incentives, business skills mentoring, and linkages to funding networks and markets. The concept of start-up hubs is internationally established and there are numerous examples of success. However, there is still not much documented experience with emerging ecosystem start-up hubs. Especially for countries like Uganda and other East Africa economies, where a startup ecosystem has to be established first, such experiences are important. The question here is whether established practices will be successful with start-up hubs and start-ups in these emerging ecosystems. There is also the question of appropriate metrics to assess the progress and success of the software start-ups. The results of this research promise practical suggestions for benchmarking best practices in start-ups and hubs in relation to successful start-ups and hubs within the emerging East Africa ecosystems but also on operation of start-ups and design and operation of software start-up hubs.

9.3 The Journey

The thesis systematically deals with the necessary concepts of a software start-up itself, but also the concepts of start-up hubs as key players in nurturing early-stage software start-ups. It further delves into what the start-ups and hubs measure to track their progress and the requirements for tools that can help them in measuring their success. The policy and regulatory aspects of the start-ups and other start-up actors like angel investors, markets are out of scope of this study in as much as we were cognisant of the fact that, those aspects play an important role in shaping the ecosystems.

9.4 Discussion of results and their implications

The goal of this study was to explore how software start-ups operate in emerging East African ecosystems. The problem statement essentially focuses on better understanding the challenges of start-ups in emerging ecosystems and examines the challenges and problems that exist and the opportunities to build ecosystems in such a way that they are successful in terms of fostering software start-ups. Since the problem is quite broad, we focused on practices in particularly metrics. Metrics play a role especially when it comes to understanding and assessing the state of internal and external development of a software start-up. From an investor's point of view, metrics have a somewhat different role. Here, it is a matter of assessing the value of a start-up, identifying risks and determining the amount of an investment, among other things. For start-up hubs, metrics can be interesting to identify better measures to support software start-ups and thus optimize their operations to be more profitable and also nurture more successful start-ups.

The East Africa software ecosystem can be characterised as emerging and two studies map practices and patterns in software start-ups and hubs in the two emerging East Africa ecosystems of Kenya and Uganda. The studies also examine measurement and metrics as essential aspects of both start-ups and the hubs that nurture them. Metrics can play an important role in many aspects of start-ups and hub operations as they provide data-driven insight and help in objective decision making. Given that start-ups are still testing and validating both technical and business aspects, pivoting is a key aspect of their evolution and the use of metrics can play a fundamental role. However, the type of metrics and what motivates the start-ups and hubs to use the metrics in emerging ecosystems is a subject of study that this thesis partly addresses in Chapter 7.

The results of the first study are presented in Chapter 4 provide a rough overview of the extent to which practices and strategies packaged as patterns from developed and seemingly more successful start-up ecosystems can be transferred or are applied in software start-ups in emerging ecosystems. This study samples five selected patterns for their broadness of coverage of start-up aspects and their relatedness to known start-up challenges. The results showed transfer-ability of strategies to some common challenge and a variation in solutions to some, thus highlighting some ecosystem influence on the practices. This result can motivate emerging ecosystem start-ups to reenact proven strategies from start-ups in developed ecosystems to solve some of the challenges they may encounter for the first time as they grow. On the other hand the existing regulations and policies in emerging ecosystems can be updated to support start-ups in fully realising practices that are partially or not wholly practiced in their start-ups yet they are seen to be widely used in developed and more successful ecosystem start-ups.

The results of the second study provide a structured overview of hub activities. Furthermore, the results provide an overview of metrics that the hubs use to track the progress of

their software start-ups. This study, however does not provide a distinction between software start-up phases which would have mapped the metrics and resources and incentives to the different phases of the start-ups. This is because software start-ups in early stages typically have different success metrics than start-ups in later stages such as the growth phase. However, given that hubs in the emerging ecosystems typically host early-stage software start-ups, the set of metrics and the identified resources and incentives from these hubs map onto that early-stage phase of the software start-ups and are thus relevant to many hubs in the ecosystem. A distinction or more precise classification of the metrics would be helpful in future work. The metrics could be standardised and benchmarked with more successful hubs in the emerging ecosystem so that hubs that host software start-ups can use them in successfully tracking the progress of their start-ups. The use of metrics in hubs and software start-ups reinforces the importance of measurement as an age-old practice that originated in classic engineering domains but still has much use in the emerging fields like software engineering.

In the third study, presented in Chapter 6 the metrics used in start-ups in emerging East Africa ecosystems are compared with metrics from the Lean startup domain like Lean Analytics. Such a comparison can be considered helpful, as it can also be used to assess the maturity of software start-ups in terms of measurement methods. The software start-ups in emerging ecosystems also see many benefits in using measurements. This aspect can be considered as a particularly valuable result from our study and can motivate research into measurement methods and tools to facilitate this activity within software start-ups. An examination of the metrics used in the start-ups also revealed that some metrics used in the software start-ups could not be clearly mapped to those published in Lean Analytics, an indication that a broader and more inclusive categorisation of the metrics and assessment methods thereof can benefit the software start-ups that deem the outlying metrics useful.

The results of the fourth study presented in Chapter 7 analyze meta aspects of metrics with a goal of discovering the most important aspects that motivate the choice and use of particular metrics. The meta aspect such as the use of tools in collecting measurement data for a given metric could be a hindrance or a motivator for a start-up to adopt a given metric. The results of this study show that software start-ups may be motivated to chose and use certain metrics from established companies in regard to four aspects, the collection of measurement data being one of them. By comparing the software start-ups and mature software companies, the study reveals that metrics may be used or dropped as start-up companies mature since some of these differentiating aspects for the choice of use of the metrics may vary as start-ups get more experience in measurement practices, get more funding to afford tools and have established business and technical processes that they can report using metrics. The aspects that can influence metrics use are generally related to start-up practices that were examined and revealed in the previous studies. Future work can examine and map out the relationship between the metric aspects and the start-up practices that influence these aspects.

The last study presented in Chapter 8 focuses on dashboards and examines tools as practical channels to realise measurement and direct benefits of measuring to the start-ups. Start-ups indicate that measuring dashboards as tools will benefit them in practice and these tools are particularly important to software start-ups to identify and monitor the right metrics in order to take the right actions at the right time. This study motivates the development of usable tools that leverage the fact that metrics are used in software start-ups as already used in large software development organisations. The measurement tools can thus help in inculcating standardised measurement practices in start-up in emerging ecosystems.

9.5 Summary

In this chapter, we position our study in the wider context of software start-ups and the emerging start-up ecosystem, highlight the study gap that we addressed and discuss the results from the five studies compiled in this thesis while pointing out the implication of our results.

Chapter 10

Conclusion and Recommendations

This chapter presents an overview of each of the five studies, highlighting significant findings in answering the main research questions (MRQ1, MRQ2, MRQ3). We also point out the contributions to various start-up and software engineering areas and make recommendations for future research areas to build on this study.

10.1 MRQ1: How do software start-ups and hubs operate in the emerging East Africa start-up ecosystem?

To answer this question, we set out to explore and understand the common practices in software start-ups in Uganda, as reported in Chapter 4. Software start-ups as young firms face several challenges as they grow. They mitigate these challenges using various common solutions in some contexts and vary some solutions. The varied solutions could hint at ecosystem influences. These reoccurring problems and solutions strategies are captured by Dande et al. [14] as patterns across the Swiss-Finish software start-up ecosystems. However, to the best of our knowledge, the extent to which these patterns apply in start-ups in emerging ecosystem contexts is unknown. Emerging ecosystems have limited infrastructure, technology skill gaps, lack of enabling policies, low funding potential, few entrepreneurial networks, and low-literacy technology markets.

Taking an example of Uganda as an emerging start-up context, we investigated the applicability of software start-up patterns found in Switzerland and Finland ecosystems to the software start-ups in Uganda. Using the modified Coplien start-up pattern format, we considered three sections context, problem, and solution. We defined and applied pattern matching criteria the comprised 1) complete pattern matching, 2) partial pattern matching or variant, and 3) no pattern matching. We used the criteria on five patterns representing significant challenges in early-stage software start-ups.

Results showed that all five patterns have matches in the seven interviewed start-ups in Uganda. The matching of the patterns confirmed their generalizability across cultural and ecosystem differences. For some given practices, the start-ups provide similar solutions but ignore some solutions across other patterns altogether, potentially hinting on contextual/ecosystem influences that may make specific solutions easier to apply in Uganda while not deeming the others not feasible. The findings' applicability is limited to start-ups hosted in hubs because it did not explore the patterns in start-ups outside the hubs.

10.2 MRQ2: What do software start-ups and hubs in emerging ecosystems measure?

Studies have pointed to a high failure rate of the software start-ups in the early stages of their growth. To this, privately and publicly owned hubs have come to the fore in nurturing the early-stage software start-ups to improve their success. The hubs thus provide incentives and resources as the start-ups evolve into mature businesses. Studies elsewhere have generally found hubs as nurturing spaces that provide incubation, acceleration, and co-working space as services for start-ups. However, understanding how the hubs operate regarding software start-ups in the Emerging East Africa ecosystem is unknown.

Given the potential contextual influences of the ecosystem on the software start-ups, we undertook a study to characterize the operations and progress metrics of hubs concerning software start-ups in the emerging East Africa ecosystem. This study reported in Chapter 5, involved ten hub interviews in Uganda and Kenya. We analyzed the data using thematic analysis. The study revealed that hubs organized, ran networking and team building events, added value to their start-ups, and mainly focused on incubation as one service to software start-ups. All hubs provided some incentives to their alumni start-ups and some recognized value from their alumni. Many hubs had varying internal activities and provided mainly business growth incentives, noticed mostly business and some organizational effects, and shared no standard criteria for selecting software start-ups. The hubs, too, primarily tracked business metrics and scalability metrics. The study reported in Chapter 5 makes three significant contributions. Firstly it provides researchers a better understanding of the workings of hubs in the East African ecosystem. Secondly, the study enables hubs to 1) identify and explore value-adding activities, resources, and incentives that other hubs use and 2) to track and measure the progress metrics that can provide a basis to benchmark with the more successful hubs in the ecosystem.

The study reported in Chapter 6 is a multi-case in 19 software start-ups hosted in hubs in Uganda and Kenya to investigate metrics in software start-ups in the emerging ecosystem. It aimed at understanding the use and perceived benefits of measurement in software start-ups. Software measurement is essential for software companies to make decisions that can help them stay competitive and deliver quality software products. Existing literature highlights that measurement programs are the vehicles for large software companies' metrics; however, there is limited research on measurement in software start-ups [31]. We did not find studies about measurement in start-ups in emerging software ecosystems to the best of our knowledge.

This study revealed that software start-ups preferably used business and product metrics and, to a lesser extent, organizational performance metrics. There was no evidence of use or wish for design metrics in start-ups. The older software start-ups also wished for project-oriented metrics. The start-ups also showed considerable expectations in the benefits of measuring, although a number of them perceived their measuring activities to be insufficient. This study contributes to software start-ups' characterization by providing empirical evidence of common and unique metrics software start-ups in the emerging East Africa ecosystem use in their operations.

Early studies point to large software companies using metrics, organizing, and evaluating them in formal measurement programs. Previous studies on metrics use highlighted that software start-ups did not formally measure but still perceived the metrics as beneficial. What influences the use and choice of metrics in software start-ups is unknown. Existing studies focus on the characterization of metrics for processes, products, and projects. The software engineering-focus on metrics' characterization is still essential and popular for un-

derstanding, monitoring, and quantifying aspects of the products, processes, and projects in the software start-ups as software-intensive businesses. However, start-ups and even mature and large software companies are interested in business and organizational metrics that are quantitative and qualitative. These metrics are generally not expressed as quantities with units derived using a measurement process that meets metrology requirements presented in the Measurement Information Model specified as ISO 15939[86]. Chapter 7 we derived ten dimensions to characterize metrics and surveyed 19 start-ups and ten mature companies. The dimensions were primarily motivated by mapping metrics to targeted users [26] and known challenges start-ups face. Results indicated that four dimensions showed differences between start-ups and mature software companies: 1) frequency of assessment, 2) automation of the measuring processes, 3) type of investment in the measuring tools, and 4) the final form of metrics. The metrics used by software start-ups and mature companies are similar in 6 of the ten dimensions that include use of a metric, phase of assessment, target users of the metric, time spent on assessing a metric, time used to set up tools, and objectivity/subjectivity of the metric. This study highlighted that start-ups and mature companies could choose a metric differently based on four characterizing dimensions.

10.3 MRQ3: How can tools support measurements in early-stage software start-ups?

An earlier study reported in Chapter 5 [48] identified the metrics in early-stage software East African. The study highlighted that some metrics differ from those used in software start-ups in developed ecosystems start-ups. However, this study did not investigate how start-ups utilized the metrics to monitor their progress. In Chapter 8, we investigated the requirements for measurement dashboards and how the start-ups expect to benefit from them. In this study, we conducted 38 semi-structured interviews with start-ups in Kenya and Uganda. We categorized requirements into user requirements, functional requirements, and non-functional requirements. We further grouped the metrics into start-up focus areas that included task, resource, performance, analytic, and presentation. Results indicated that start-ups require the dashboards to support user and functional requirements for performance measurement. They also wanted the dashboards to support the display for the metrics. Most start-ups expect/experienced performance-related benefits with relatively fewer and varying benefits related to resources, teams, products, and investment.

We compared software start-up dashboard requirements to the requirements presented by Velcu-Laitinen and Yigitbasioglu [3]. Our results were similar in that communication consistency was the purpose and features of performance management dashboards. This result highlighted the primary use of dashboards as communication. Presentation flexibility was an essential feature in the software start-ups and was ranked second popular in Finnish companies. Monitoring was the second most popular purpose in the start-ups, and rationalization of decision making was the least popular purpose in start-ups (pointing to the unilateral and ad hoc nature of most decisions made in early-stage start-ups). The study in Chapter 8 contributes requirements and motivation for developing metric-based measurement tools for early-stage software start-ups, as evidenced in our prototype tool contribution hosted at StartupHealth[28]. This study was important as it forms a basis for developing a software tool for measurement dashboards suited for early-stage software start-ups in the East Africa ecosystem by capturing and visualizing the metrics used by start-ups.

10.4 Recommendations

Based on the results from this study, in this section, we present our recommendations to researchers, software start-ups, hubs and governments in the emerging ecosystems where the studies were carried out.

10.4.1 Researchers

Emerging ecosystems software start-ups were seen to largely adopt practices and strategies seen elsewhere in developed start-up ecosystems with slight variation in some solutions to common challenges. Based on these results, we recommend researchers to interest themselves in understanding the extent to which the software start-ups apply, adhere and are affected by these practices as they mature in the ecosystems.

10.4.2 Software start-ups and Hubs

Given that symbiotic relationship between start-ups and hubs as seen by the fact that the hubs provide resources and incentives and both the hubs and start-ups recognise each others' value, we recommend the strengthening of efforts by the government to subsidise hubs in an effort keep them sustainable. This subsidies are required because start-ups as young companies are severely underfunded and can barely afford to pay fees that keep the hubs running yet on one hand, the start-ups have potential to grow into fully fledged businesses after developing innovative software products and services. The start-ups' early-stages of development are nurtured by the burgeoning hubs as the hubs have been proven to play a critical role in the growth and maturity cycle of the start-ups in vibrant ecosystems in the developed world.

10.4.3 Government

The published frameworks for classifying start-ups and measurements used by software start-ups do not account for some start-up types that are prevelant in the emerging East Africa ecosystems. For example start-ups that provide software development as a service and Fintechs that are prevalent in the East Africa emerging ecosystems are hardly classified and their peculiar metrics hardly captured by the Lean Analytics framework. Of interest would be how policies can be formulated to strengthen these types of start-ups as they seem to be popular in the ecosystems. For example, the current effort by Financial Technology Service Providers Association of Uganda (FITSPA) in drafting a regulatory framework for Fintechs is a welcome effort that recognises the value that Fintech start-ups have in moving the Ugandan economy from cash-based to cashless. We also recommend development a related guiding framework inline with National Development Goals (NDGs) that can be used to guide activities, protect contracts and harness software-development as a service start-ups as these employ a number of young skilled developers who do offshore software development for larger companies and are paid way and above the regular salaries that Ugandan companies can offer.

10.5 Future Work

The study presented in Chapter 4 is an exploratory study of practices and patterns in software start-ups in Uganda. Future work may need to investigate the general applicability

of the practices found in start-ups in developed ecosystems but not used in start-ups in emerging ecosystems. Future studies should include a wider qualitative and quantitative scope that explores both software start-ups in and outside incubation hubs at different stages of growth in related emerging start-up ecosystems.

The study presented in Chapter 5 offers a future opportunity for software start-ups to corroborate our reported findings. Future work should also compare the software start-ups hosted by the hubs with software start-ups that mostly bootstrapped themselves to understand the hubs' real effects. Other studies could also investigate hub archetypes and identify their most significant operations in nurturing successful start-ups in emerging start-up ecosystems.

Future work for the study in Chapter 6 should explore the metrics' suitability in measuring certain aspects of start-ups for the metrics identified in the East African software start-ups. To further understand hubs' operations concerning start-ups, future work will need to investigate how the hubs use different metrics to track the progress and day-to-day decision-making. The study presented in Chapter 7 characterized metrics using ten dimensions. Therefore, future work will need to investigate what factors play a role in instantiating these metrics. The exciting finding that software start-ups used more metrics than mature software companies presents an opportunity for future research to investigate 1) why some metrics are not used anymore by start-ups, 2) which new metrics are adopted, and 3) why they are adopted start-ups grow.

The study in Chapter 8 presented requirements for measurement dashboards perceived benefits that formed a basis to prototype a tool [28] for start-ups in the emerging ecosystem context. Future work will involve deploying the prototype measurement dashboard in software start-ups for longitudinal studies to evaluate its use and aid in decision-making.

In general, future work can extend the study to examine the legal and policy frameworks that support ICT innovations and software start-ups in the emerging ecosystems like Uganda as these frameworks or lack of them, in addition to the known challenges the start-ups themselves face still affect the the growth and maturity of start-up ecosystems.

10.6 summary

This chapter has presented a summary of the overall study, highlighting the key results emerging from answering the three research questions, the contributions of these results to the wider areas of software start-up practice and knowledge and the future direction for this research.

Bibliography

[1] Miroslaw Staron and Wilhelm Meding. Mesram–a method for assessing robustness of measurement programs in large software development organizations and its industrial evaluation. *Journal of Systems and Software*, 113:76–100, 2016.

[2] Alistair Croll and Benjamin Yoskovitz. *Lean analytics: Use data to build a better startup faster.* " O'Reilly Media, Inc.", 2013.

[3] Oana Velcu-Laitinen and Ogan M Yigitbasioglu. The use of dashboards in performance management: Evidence from sales managers. *International Journal of Digital Accounting Research*, 12, 2012.

[4] Yunfei Shi, Dongming Xu, and Iris Vessey. Early-stage software start-up survival: the effects of managerial actions on firm performance. In *International Conference on HCI in Business*, pages 761–771. Springer, 2015.

[5] Eric Ries. *The lean startup: How today's entrepreneurs use continuous innovation to create radically successful businesses.* Crown Books, 2011.

[6] Steve Blank. *The startup owner's manual: The step-by-step guide for building a great company.* BookBaby, 2012.

[7] M Marmer, BL Herrmann, E Dogrultan, R Berman, C Eesley, and S Blank. The startup ecosystem report 2012. *Startup Genome*, 2012.

[8] Nicolò Paternoster, Carmine Giardino, Michael Unterkalmsteiner, Tony Gorschek, and Pekka Abrahamsson. Software development in startup companies: A systematic mapping study. *Information and Software Technology*, 56(10):1200–1218, 2014.

[9] Stanley M Sutton. The role of process in software start-up. *IEEE Software*, 17(4):33–39, 2000.

[10] Grace Kamulegeya, Regina Hebig, Imed Hammouda, Michel Chaudron, and Raymond Mugwanya. Exploring the applicability of software startup patterns in the ugandan context. In *2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, pages 116–124. IEEE, 2017.

[11] Nirnaya Tripathi, Pertti Seppänen, Ganesh Boominathan, Markku Oivo, and Kari Liukkunen. Insights into startup ecosystems through exploration of multi-vocal literature. *Information and Software Technology*, 105:56–77, 2019.

[12] Carmine Giardino, Xiaofeng Wang, and Pekka Abrahamsson. Why early-stage software startups fail: a behavioral framework. In *International Conference of Software Business*, pages 27–41. Springer, 2014.

[13] Carmine Giardino, Sohaib Shahid Bajwa, Xiaofeng Wang, and Pekka Abrahamsson. Key challenges in early-stage software startups. In *International Conference on Agile Software Development*, pages 52–63. Springer, 2015.

[14] Anuradha Dande, Veli-Pekka Eloranta, Antti-Jussi Kovalainen, Timo Lehtonen, Marko Leppänen, Taru Salmimaa, Mahbubul Sayeed, Matti Vuori, Claude Rubattel, Wolfgang Weck, et al. Software startup patterns-an empirical study. *Tampereen teknillinen yliopisto. Tietotekniikan laitos. Raportti-Tampere University of Technology. Department of Pervasive Computing. Report; 4*, 2014.

[15] Romy Voss and Christoph Müller. How are the conditions for high-tech start-ups in germany? *International Journal of Entrepreneurship and Small Business*, 7(3):284–311, 2009.

[16] Subhash V Kasturi and MH Bala Subrahmanya. Start-ups and small scale industry growth in india: do institutional credit and start-ups make a difference? *International Journal of Entrepreneurial Venturing*, 6(3):277–298, 2014.

[17] Daniel Cukier, Fabio Kon, and Norris Krueger. Designing a maturity model for software startup ecosystems. In *International Conference on Product-Focused Software Process Improvement*, pages 600–606. Springer, 2015.

[18] Mark Crowne. Why software product startups fail and what to do about it. evolution of software product development in startup companies. In *Engineering Management Conference, 2002. IEMC'02. 2002 IEEE International*, volume 1, pages 338–343. IEEE, 2002.

[19] Eriks Klotins, Michael Unterkalmsteiner, Panagiota Chatzipetrou, Tony Gorschek, Rafael Prikladnicki, Nirnaya Tripathi, and Leandro Pompermaier. A progression model of software engineering goals, challenges, and practices in start-ups. *IEEE Transactions on Software Engineering*, 2019.

[20] Natalie Chirchietti. The role of innovation hubs taking start-ups from idea to business. *IZNE Working Paper Series*, 2017.

[21] Victor Du Boucher. A few things we learned about tech hubs in africa and asia. *GSMA. http://www. gsma. com/mobilefordevelopment/programme/ecosystem-accelerator/things-learned-tech-hubs-africa-asia*, 2016.

[22] Duncan Gathege and Hilda Moraa. Ict hubs model: Understanding the key factors of the ihub model, nairobi kenya. *Source: www. research. ihub. co. ke/uploads/2013/may/1367834943_ _374. pdf (23.02. 2017)*, 2013.

[23] Fabio Kon, Daniel Cukier, Claudia Melo, Orit Hazzan, and Harry Yuklea. A panorama of the israeli software startup ecosystem. *Available at SSRN 2441157*, 2014.

[24] Daniel Cukier and Fabio Kon. Early-stage software startup patterns: Strategies to building high-tech software companies from scratch. In *Proceedings of the 22nd Pattern Languages of Programs Conference (PLoP)*, page 11. ACM, 2015.

[25] Daniel Cukier and Fabio Kon. A maturity model for software startup ecosystems. *Journal of Innovation and Entrepreneurship*, 7(1):14, 2018.

[26] Michael K Daskalantonakis. A practical view of software measurement and implementation experiences within motorola. *IEEE Transactions on Software Engineering*, 18(11):998, 1992.

[27] Miroslaw Staron, Wilhelm Meding, Jörgen Hansson, Christoffer Höglund, Kent Niesel, and Vilhelm Bergmann. Dashboards for continuous monitoring of quality for software product under development. In *Relating System Quality and Software Architecture*, pages 209–229. Elsevier, 2014.

[28] Grace Kamulegeya. Startuphealth, https://makinnovation.com/project/26, 2020.

[29] Carmine Giardino, Sohaib Shahid Bajwa, Xiaofeng Wang, and Pekka Abrahamsson. Key challenges in early-stage software startups. In *International Conference on Agile Software Development*, pages 52–63. Springer, 2015.

[30] Gerry Coleman and Rory V O'Connor. An investigation into software development process formation in software start-ups. *Journal of Enterprise Information Management*, 21(6):633–648, 2008.

[31] Michael Unterkalmsteiner, Pekka Abrahamsson, XiaoFeng Wang, Anh Nguyen-Duc, Syed Shah, Sohaib Shahid Bajwa, Guido H Baltes, Kieran Conboy, Eoin Cullina, Denis Dennehy, et al. Software startups–a research agenda. *e-Informatica Software Engineering Journal*, 10(1), 2016.

[32] Zenas Block and Ian C MacMillan. Milestones for successful venture planning. *Harvard Business Review*, 63(5):184, 1985.

[33] Schaul Chorev and Alistair R Anderson. Success in israeli high-tech start-ups; critical factors and process. *Technovation*, 26(2):162–174, 2006.

[34] Eriks Klotins, Michael Unterkalmsteiner, and Tony Gorschek. Software engineering knowledge areas in startup companies: a mapping study. In *International Conference of Software Business*, pages 245–257. Springer, 2015.

[35] IEEE Computer Society. *Guide to the Software Engineering Body of Knowledge Version 3.0 (SWEBOK Guide V3.0).*

[36] Marc Salas Martinez et al. Good practices of the lean startup methodology: Benefits, challenges and recommendations. 2016.

[37] Markus Schumacher, Eduardo Fernandez-Buglioni, Duane Hybertson, Frank Buschmann, and Peter Sommerlad. *Security Patterns: Integrating security and systems engineering.* John Wiley & Sons, 2013.

[38] James O Coplien and A Word On Alexander. Software patterns. 1996.

[39] Marko Leppänen. Patterns for starting up a software startup company. In *Proceedings of the 19th European Conference on Pattern Languages of Programs*, page 25. ACM, 2014.

[40] Veli-Pekka Eloranta. Organizational patterns: Creating an on-boarding experience. In *Proceedings of the 10th Travelling Conference on Pattern Languages of Programs*, page 2. ACM, 2016.

[41] Veli-Pekka Eloranta. Towards a pattern language for software start-ups. In *Proceedings of the 19th European Conference on Pattern Languages of Programs*, page 24. ACM, 2014.

[42] Jorge Melegati and Alfredo Goldman. Patterns for software development startups. In *Proceedings of the 22nd Pattern Languages of Programs Conference (PLoP)*, page 11. ACM, 2015.

[43] Touseef Tahir, Ghulam Rasool, and Cigdem Gencel. A systematic literature review on software measurement programs. *Information and Software Technology*, 73:101–121, 2016.

[44] Milind Kumar Sharma, Rajat Bhagwat, and Govind Sharan Dangayach. Practice of performance measurement: experience from indian smes. *International Journal of Globalisation and Small Business*, 1(2):183–213, 2005.

[45] Valtierra Claudia, Muñoz Mirna, and Mejia Jezreel. Characterization of software processes improvement needs in smes. In *2013 International Conference on Mechatronics, Electronics and Automotive Engineering*, pages 223–228. IEEE, 2013.

[46] Ansar Malook Bhatti, Hafiz Muhammad Abdullah, and Cigdem Gencel. A model for selecting an optimum set of measures in software organizations. In *European Conference on Software Process Improvement*, pages 44–56. Springer, 2009.

[47] Ita Richardson and Christiane Gresse Von Wangenheim. Guest editors' introduction: Why are small software organizations different? *IEEE software*, 24(1):18–22, 2007.

[48] Grace Kamulegeya, Raymond Mugwanya, and Regina Hebig. Measurements in the early stage software start-ups: A multiple case study in a nascent ecosystem. *Foundations of Computing and Decision Sciences*, 43(4):251–280, 2018.

[49] Oswaldo Gómez, Hanna Oktaba, Mario Piattini, and Félix García. A systematic review measurement in software engineering: State-of-the-art in measures. In *International Conference on Software and Data Technologies*, pages 165–176. Springer, 2006.

[50] María Díaz-Ley, Félix García, and Mario Piattini. Mis-pyme software measurement maturity model-supporting the definition of software measurement programs. In *International Conference on Product Focused Software Process Improvement*, pages 19–33. Springer, 2008.

[51] Danilo Caivano. Continuous software process improvement through statistical process control. In *Ninth European Conference on Software Maintenance and Reengineering*, pages 288–293. IEEE, 2005.

[52] Luigi Lavazza and Giancarlo Barresi. Automated support for process-aware definition and execution of measurement plans. In *Proceedings of the 27th international conference on Software engineering*, pages 234–243, 2005.

[53] Francisco J Pino, Félix García, and Mario Piattini. Software process improvement in small and medium software enterprises: a systematic review. *Software Quality Journal*, 16(2):237–261, 2008.

[54] Pat Allen, Muthu Ramachandran, and Hisham Abushama. Prisms: an approach to software process improvement for small to medium enterprises. In *Third International Conference on Quality Software, 2003. Proceedings.*, pages 211–214. IEEE, 2003.

[55] Ayse Tosun, Ayse Bener, and Burak Turhan. Implementation of a software quality improvement project in an sme: a before and after comparison. In *2009 35th Euromicro Conference on Software Engineering and Advanced Applications*, pages 203–209. IEEE, 2009.

[56] Touseef Tahir, Ghulam Rasool, and Muhammad Noman. A systematic mapping study on software measurement programs in smes. *e-Informatica Software Engineering Journal*, 12(1), 2018.

[57] Victor R Basili. Gianluigi caldiera ja h. dieter rombach. goal question metric paradigm. kirjassa john j. marciniak, toim., encyclopedia of software engineering, ss. 528–532, 1994.

[58] María Díaz-Ley, Félix García, and Mario Piattini. Mis-pyme software measurement capability maturity model–supporting the definition of software measurement programs and capability determination. *Advances in Engineering Software*, 41(10-11):1223–1237, 2010.

[59] Cigdem Gencel, Kai Petersen, Aftab Ahmad Mughal, and Muhammad Imran Iqbal. A decision support framework for metrics selection in goal-based measurement programs: Gqm-dsfms. *Journal of Systems and Software*, 86(12):3091–3108, 2013.

[60] Romy Voss and Christoph Müller. How are the conditions for high-tech start-ups in germany? *International Journal for Entrepreneurship and Small Business*, 7(3):284–311, September 2009.

[61] Eric Vaz, Teresa de Noronha Vaz, Purificacion Vicente Galindo, and Peter Nijkamp. Modelling innovation support systems for regional development–analysis of cluster structures in innovation in portugal. *Entrepreneurship & Regional Development*, 26(1-2):23–46, 2014.

[62] F Kon, D Cukier, C Melo, O Hazzan, and H Yuklea. A conceptual framework for software startup ecosystems: the case of israel. *Department of Computer Science, University of São Paulo, Tech. Rep. RT-MAC-2015-01*, 2015.

[63] LN Safiullin, AM Fatkhiev, and KA Grigorian. The triple helix model of innovation. *Mediterranean Journal of Social Sciences*, 5(18):203–203, 2014.

[64] Erik Stam. Measuring entrepreneurial ecosystems. In *Entrepreneurial ecosystems*, pages 173–197. Springer, 2018.

[65] Daniel J Isenberg. How to start an entrepreneurial revolution. *Harvard business review*, 88(6):40–50, 2010.

[66] Jeremy De Beer, Paula Millar, Jacquelene Mwangi, Victor Nzomo, and Isaac Rutenberg. A framework for assessing technology hubs in africa. *NYU J. Intell. Prop. & Ent. L.*, 6:237, 2016.

[67] Leandro Pompermaier and Rafael Prikladnicki. Brazilian startups and the current software engineering challenges: The case of tecnopuc. In *Fundamentals of Software Startups*, pages 331–345. Springer, 2020.

[68] C Scott Dempwolf, Jennifer Auer, and Michelle D'Ippolito. Innovation accelerators: Defining characteristics among startup assistance organizations. *Small Business Administration*, pages 1–44, 2014.

[69] Songphon Munkongsujarit. Business incubation model for startup company and sme in developing economy: a case of thailand. In *2016 Portland International Conference on Management of Engineering and Technology (PICMET)*, pages 74–81. IEEE, 2016.

[70] Femke Van Noorloos, Diky Avianto, and Romanus Otieno Opiyo. New master-planned cities and local land rights: the case of konza techno city, kenya. *Built Environment*, 44(4):420–437, 2019.

[71] Tom Standage. Why does kenya lead the world in mobile money. *The Economist*, 2013.

[72] Regional Advantage. Culture and competition in silicon valley and route 128, 1994.

[73] Banji Oyelaran-Oyeyinka and Dorothy McCormick. *Industrial clusters and innovation systems in Africa*. 2007.

[74] Anh Nguyen-Duc, Pertti Seppänen, and Pekka Abrahamsson. Hunter-gatherer cycle: a conceptual model of the evolution of software startups. In *Proceedings of the 2015 International Conference on Software and System Process*, pages 199–203, 2015.

[75] J Rikkila, P Abrahamsson, and X Wang. The implications of a complexity perspective for software engineering practice and research. 2012.

[76] Martin Steinert and Larry J Leifer. 'finding one's way': Re-discovering a hunter-gatherer model based on wayfaring. *International Journal of Engineering Education*, 28(2):251, 2012.

[77] David J Snowden and Mary E Boone. A leader's framework for decision making. *Harvard business review*, 85(11):68, 2007.

[78] Sohaib Shahid Bajwa, Xiaofeng Wang, Anh Nguven Duc, and Pekka Abrahamsson. How do software startups pivot? empirical results from a multiple case study. In *International Conference of Software Business*, pages 169–176. Springer, 2016.

[79] Saras D Sarasvathy. Causation and effectuation: Toward a theoretical shift from economic inevitability to entrepreneurial contingency. *Academy of management Review*, 26(2):243–263, 2001.

[80] Paul Williams. The competent boundary spanner. *Public administration*, 80(1):103–124, 2002.

[81] Veli-Pekka Eloranta. Patterns for controlling chaos in a startup. In *Proceedings of the 8th Nordic Conference on Pattern Languages of Programs (VikingPLoP)*, page 1. ACM, 2014.

[82] Anna Bergek and Charlotte Norrman. Incubator best practice: A framework. *Technovation*, 28(1-2):20–28, 2008.

[83] Donald F Kuratko and William R LaFollette. Small business incubators for local economic development. *Economic Development Review*, 5(2):49, 1987.

[84] Rosa Grimaldi and Alessandro Grandi. Business incubators and new venture creation: an assessment of incubating models. *Technovation*, 25(2):111–121, 2005.

[85] Danny P Soetanto and Sarah L Jack. Business incubators and the networks of technology-based firms. *The Journal of Technology Transfer*, 38(4):432–453, 2013.

[86] Alain Abran. *Software metrics and software metrology*. John Wiley & Sons, 2010.

[87] Robert E Park. Software size measurement: A framework for counting source statements. Technical report, CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST, 1992.

[88] Allan J. Albrecht and John E Gaffney. Software function, source lines of code, and development effort prediction: a software science validation. *IEEE transactions on software engineering*, SE-9(6):639–648, 1983.

[89] Barbara Kitchenham and Emilia Mendes. Software productivity measurement using multiple size measures. *IEEE Transactions on Software Engineering*, 30(12):1023–1035, 2004.

[90] Norman Fenton and James Bieman. *Software metrics: a rigorous and practical approach*. CRC press, 2014.

[91] Sohaib Shahid Bajwa, Cigdem Gencel, and Pekka Abrahamsson. Software product size measurement methods. In *Proceedings of International Workshop on Software Measurement and the International Conference on Software Process and Product Measurement, IWSM-Mensura*, volume 14, pages 176–190, 2014.

[92] Peter Comer and Jonathan Chard. A measurement maturity model. *Software Quality Journal*, 2(4):277–289, 1993.

[93] Michael K Daskalantonakis, Robert H Yacobellis, and Victor R Basili. A method for assessing software measurement technology. *Quality Engineering*, 3(1):27–40, 1990.

[94] Nopadol Rompho. Operational performance measures for startups. *Measuring Business Excellence*, 22(1):31–41, 2018.

[95] Bhagyashree Paranjape, Margaret Rossiter, and Victor Pantano. Performance measurement systems: successes, failures and future–a review. *Measuring Business Excellence*, 10(3):4–14, 2006.

[96] IEEE. Ieee standard for a software quality metrics methodology. *IEEE Std 1061-1998*, pages i–, Dec 1998.

[97] Jonathan E Cook and Alexander L Wolf. Software process validation: quantitatively measuring the correspondence of a process to a model. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 8(2):147–176, 1999.

[98] Kai-Kristian Kemell, Xiaofeng Wang, Anh Nguyen-Duc, Jason Grendus, Tuure Tuunanen, and Pekka Abrahamsson. 100+ metrics for software startups-a multi-vocal literature review. *arXiv preprint arXiv:1901.04819*, 2019.

[99] Saurabh Tiwari and Santosh Singh Rathore. Coupling and cohesion metrics for object-oriented software: A systematic mapping study. In *Proceedings of the 11th Innovations in Software Engineering Conference*, page 8. ACM, 2018.

[100] Paolo Taticchi, Paola Cocca, and Marco Alberti. A framework to assess performance measurement systems in smes. *International Journal of Productivity and Performance Management*, 2010.

[101] Michael Unterkalmsteiner, Tony Gorschek, AKM Moinul Islam, Chow Kian Cheng, Rahadian Bayu Permadi, and Robert Feldt. Evaluation and measurement of software process improvement—a systematic literature review. *IEEE Transactions on Software Engineering*, 38(2):398–424, 2011.

[102] NE Schneidewind. Body of knowledge for software quality measurement. *Computer*, 35(2):77–83, 2002.

[103] Yingxu Wang. The measurement theory for software engineering. In *CCECE 2003-Canadian Conference on Electrical and Computer Engineering. Toward a Caring and Humane Technology (Cat. No. 03CH37436)*, volume 2, pages 1321–1324. IEEE, 2003.

[104] Alain Abran, Jean-Marc Desharnais, and Juan Jose Cuadrado-Gallego. Metrology and quantitative analysis in iso 15939. In *Software Engineering Research and Practice*, pages 592–598, 2009.

[105] Amy Miller and Jennifer Cioffi. Measuring marketing effectiveness and value: the unisys marketing dashboard. *Journal of Advertising Research*, 44(3):237–243, 2004.

[106] Wayne W Eckerson. *Performance dashboards: measuring, monitoring, and managing your business*. John Wiley & Sons, 2010.

[107] Koen Pauwels, Tim Ambler, Bruce H Clark, Pat LaPointe, David Reibstein, Bernd Skiera, Berend Wierenga, and Thorsten Wiesel. Dashboards as a service: why, what, how, and what research is needed? *Journal of service research*, 12(2):175–189, 2009.

[108] Nils H Rasmussen, Manish Bansal, and Claire Y Chen. *Business dashboards: a visual catalog for design and deployment*. John Wiley & Sons, 2009.

[109] Pasi Tyrväinen, Matti Saarikallio, Timo Aho, Timo Lehtonen, and Rauno Paukeri. Metrics framework for cycle-time reduction in software value creation. In *ICSEA 2015: The Tenth International Conference on Software Engineering Advances, ISBN 978-1-61208-438-1*. IARIA, 2015.

[110] Victor R Basili1 Gianluigi Caldiera and H Dieter Rombach. The goal question metric approach. *Encyclopedia of software engineering*, pages 528–532, 1994.

[111] Robert S Kaplan and David P Norton. Putting the balanced scorecard to work. *The economic impact of knowledge*, 71(5):315–24, 1998.

[112] David N Card. Practical software measurement. In *25th International Conference on Software Engineering, 2003. Proceedings.*, pages 738–739. IEEE, 2003.

[113] M. Saunders, P. Lewis, and A. Thornhill. *Research Methods for Business Students*. Always learning. Financial Times Prentice Hall, 2009.

[114] Colin Robson and Kieran McCartan. *Real world research*. John Wiley & Sons, 2016.

[115] Robert K Yin. *Case study research and applications: Design and methods*. Sage publications, 2017.

[116] Per Runeson and Martin Höst. Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering*, 14(2):131, Dec 2008.

[117] Bent Flyvbjerg. Five misunderstandings about case-study research. *Qualitative inquiry*, 12(2):219–245, 2006.

[118] Robert Yin. K.(2003). case study research: Design and methods, 2003.

[119] A. Bryman. *Quantity and Quality in Social Research*. Taylor & Francis, 2003.

[120] Per Runeson and Martin Höst. Guidelines for conducting and reporting case study research in software engineering. *Empirical software engineering*, 14(2):131, 2009.

[121] Robert E Stake. *The art of case study research*. sage, 1995.

[122] Barbara Kitchenham and Shari Lawrence Pfleeger. Principles of survey research: part 5: populations and samples. *ACM SIGSOFT Software Engineering Notes*, 27(5):17–20, 2002.

[123] Ilker Etikan, Sulaiman Abubakar Musa, and Rukayya Sunusi Alkassim. Comparison of convenience sampling and purposive sampling. *American journal of theoretical and applied statistics*, 5(1):1–4, 2016.

[124] John W Creswell. *Educational research: Planning, conducting, and evaluating quantitative*. Prentice Hall Upper Saddle River, NJ, 2002.

[125] Colin Robson. *Real world research: A resource for social scientists and practitioner-researchers*, volume 2. Blackwell Oxford, 2002.

[126] Erich Gamma. *Design patterns: elements of reusable object-oriented software*. Pearson Education India, 1995.

[127] Daniela S Cruzes, Tore Dybå, Per Runeson, and Martin Höst. Case studies synthesis: a thematic, cross-case, and narrative synthesis worked example. *Empirical Software Engineering*, 20(6):1634–1665, 2015.

[128] Simone Sperindé and Anh Nguyen-Duc. Fostering open innovation in coworking spaces: A study of norwegian startups. In *Fundamentals of Software Startups*, pages 161–178. Springer, 2020.

[129] Vebjørn Berg, Jørgen Birkeland, Anh Nguyen-Duc, Ilias O Pappas, and Letizia Jaccheri. Software startup engineering: A systematic mapping study. *Journal of Systems and Software*, 144:255–274, 2018.

[130] Anh Bui. Startup ecosystem in finland. 2016.

[131] Maria Teresa Cometto and Alessandro Piol. *Tech and the City: The Making of New York's Startup Community*. Mirandola Press, 2013.

[132] UNICeF et al. Unicef innovation labs: A do-it-yourself guide, 2012.

[133] Virginia Braun and Victoria Clarke. Using thematic analysis in psychology. *Qualitative research in psychology*, 3(2):77–101, 2006.

[134] Uwe Flick. *An introduction to qualitative research*. Sage, 2014.

[135] Alan Bryman. *Social research methods*. Oxford university press, 2016.

[136] Bowman Fishback, Christine A Gulbranson, Robert E Litan, Lesa Mitchell, and Marisa A Porzig. Finding business' idols': A new model to accelerate start-ups. *Available at SSRN 1001926*, 2007.

[137] Aziz Nanthaamornphong and Rattana Wetprasit. Thailand's software startup ecosystem. In *Fundamentals of Software Startups*, pages 195–213. Springer, 2020.

[138] Igor Tasic, A Montoro-Sánchez, and MD Cano. Startup accelerators: an overview of the current state of the acceleration phenomenon. In *XVIII Congresso AECA. Cartagena*, 2015.

[139] Ulla H Graneheim and Berit Lundman. Qualitative content analysis in nursing research: concepts, procedures and measures to achieve trustworthiness. *Nurse education today*, 24(2):105–112, 2004.

[140] Michael Quinn Patton. Qualitative research and evaluation methods, 2002.

[141] Robert K Yin. *Applications of case study research*. Sage, 2011.

[142] Gouri K Bhattacharyya and Richard A Johnson. Statistical methods and concepts, 1977.

[143] Eric Ries. Minimum viable product: a guide. *Startup lessons learned*, 2009.

[144] Anh Nguyen Duc and Pekka Abrahamsson. Minimum viable product or multiple facet product? the role of mvp in software startups. In *International Conference on Agile Software Development*, pages 118–130. Springer, 2016.

[145] Thomas J McCabe. A complexity measure. *IEEE Transactions on software Engineering*, SE-2(4):308–320, 1976.

[146] Maurice Howard Halstead et al. *Elements of software science*, volume 7. Elsevier New York, 1977.

[147] Hareton Leung and Zhang Fan. Software cost estimation. In *Handbook of Software Engineering and Knowledge Engineering: Volume II: Emerging Technologies*, pages 307–324. World Scientific, 2002.

[148] Sohaib Shahid Bajwa, Cigdem Gencel, and Pekka Abrahamsson. Software product size measurement methods: A systematic mapping study. In *2014 Joint Conference of the International Workshop on Software Measurement and the International Conference on Software Process and Product Measurement*, pages 176–190. IEEE, 2014.

[149] Shyam R Chidamber and Chris F Kemerer. A metrics suite for object oriented design. *IEEE Transactions on software engineering*, 20(6):476–493, 1994.

[150] Satwinder Singh and Rozy Singla. Object-oriented metrics for defect prediction. In *Software Engineering*, pages 305–318. Springer, 2019.

[151] Krishna Sai Bitla and Sairam Sagar Veesamsetty. Measuring process flow using metrics in agile software development, 2019.

[152] Miroslaw Staron. Critical role of measures in decision processes: Managerial and technical measures in the context of large software development organizations. *Information and Software Technology*, 54(8):887–899, 2012.

[153] Eric W Weisstein. Bonferroni correction. 2004.

[154] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2017.

[155] Jacob Cohen. *Statistical Power Analysis for the Behavioral Sciences*. Academic Press, 2013.

[156] Sohaib Shahid Bajwa, Xiaofeng Wang, Anh Nguyen Duc, and Pekka Abrahamsson. "failures" to be celebrated: an analysis of major pivots of software startups. *Empirical Software Engineering*, 22(5):2373–2408, 2017.

[157] Thomas R Eisenmann, Eric Ries, and Sarah Dillard. Hypothesis-driven entrepreneurship: The lean startup. *Harvard Business School Entrepreneurial Management Case*, (812-095), 2012.

[158] Alan R Hevner. Phase containment metrics for software quality improvement. *Information and Software Technology*, 39(13):867–877, 1997.

[159] Abbie Griffin. Metrics for measuring product development cycle time. *Journal of Product Innovation Management: An International Publication of the Product Development & Management Association*, 10(2):112–125, 1993.

[160] Vanessa Goldoni and Mírian Oliveira. Knowledge management metrics in software development companies in brazil. *Journal of Knowledge Management*, 2010.

[161] Sungmin Yun, Jiyong Choi, Daniel P de Oliveira, and Stephen P Mulva. Development of performance metrics for phase-based capital project benchmarking. *International Journal of Project Management*, 34(3):389–402, 2016.

[162] Noriko Hanakawa and Masaki Obana. Process metrics for system quality with specifications' shifts from a bid phase to an operation phase (s). In *SEKE*, pages 687–686, 2018.

[163] Kerry Rodden, Hilary Hutchinson, and Xin Fu. Measuring the user experience on a large scale: user-centered metrics for web applications. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 2395–2398, 2010.

[164] Aftab Ahmad Mughal. *A Framework for Implementing Software Measurement Programs in Small and Medium Enterprises.* PhD thesis, University of Otago, 2017.

[165] R.K. Yin. *Case Study Research: Design and Methods.* Applied Social Research Methods. SAGE Publications, 2003.

[166] Ogan M Yigitbasioglu and Oana Velcu. A review of dashboards in performance management: Implications for design and research. *International Journal of Accounting Information Systems*, 13(1):41–59, 2012.

[167] Richard W Selby. Measurement-driven dashboards enable leading indicators for requirements and design of large-scale systems. In *11th IEEE International Software Metrics Symposium*, pages 10–pp. IEEE, 2005.

[168] Martin Glinz. On non-functional requirements. In *15th IEEE International Requirements Engineering Conference (RE)*, pages 21–26. IEEE, 2007.

[169] Ash Maurya. *Running lean: iterate from plan A to a plan that works.* " O'Reilly Media, Inc.", 2012.

Appendices

Appendix A

Interview Protocol

Welcome and thank you for your participation today. My name is Kamulegeya Grace B. I am a PhD research student at Makerere University conducting a Study in partial fulfillment of the requirements for the degree of Doctorate of Philosophy. This interview will take about _____ minutes and will include___ questions regarding your start-up on the challenges, growth milestones and measurements.

I request for your permission to audio record this interview, so I may accurately document the information you convey. If at any time during the interview you wish to discontinue the use of the recorder or the interview itself, please feel free to let me know. All of your responses are confidential. Your responses will remain confidential and used to develop a better understanding of start-ups and how to measure their progress. The purpose of this study is to characterize software start-ups within their ecosystem and measure their progress as they grow.

I would like to remind you of your written consent to participate in this study. I am the responsible investigator, specifying your participation in the research study: Characterization and Measurement of Capabilities and Processes of Software Start-ups in Incubation Hubs. You and I have both signed and dated each copy, certifying that we agree to continue this interview. You will receive one copy and I will keep the other under lock and key, separate from your transcribed audio-recordings. Your participation in this interview is voluntary. If at any time you need to stop or take a break, please let me know. You may also withdraw your participation at any time without consequence. Do you have any questions or concerns before we begin? Then with your permission, we will begin the interview.

Consent Form

Your Signature _____ Date _____

Your Name (printed) _____

In addition to agreeing to participate, I also consent to having the interview audio-recorded.

Your Signature _____ Date _____

Signature of person obtaining consent _____ Date

_____

Printed name of person obtaining consent _____ Date

_____

The researcher will keep this consent form for at least three years beyond the end of the study.

Interview #_____

Date_____/_____/_____

Appendix B

Your Signature _____ Date _____

Your Name (printed) _____

In addition to agreeing to participate, I also consent to having the interview audio-recorded.

Your Signature _____ Date _____

Signature of person obtaining consent _____ Date _____

Printed name of person obtaining consent _____ Date _____

This consent form will be kept by the researcher for at least three years beyond the end of the study.

The title of the study should appear at the top of every page.

Interview #_____

Date_____/_____/_____

Interview Protocol

Welcome and thank you for your participation today. My name is Kamulegeya Grace B and I am a graduate student at Makerere University conducting a Study in partial fulfillment of the requirements for the degree of Doctorate of Philosophy. This is an interview that will take about 45 minutes and will include 19 questions regarding your work in start-ups about the challenges, growth milestones and measurements. I would like your permission to audio record this interview, so I may accurately document the information you convey. If at any time during the interview you wish to discontinue the use of the recorder or the interview itself, please feel free to let me know. All of your responses are confidential. Your responses will remain confidential and will be used to develop a better understanding of start-ups and how to measure their progress. The purpose of this study is to characterize software start-ups within their ecosystem and measure their progress as they grow.

I would like to remind you of your written consent to participate in this study. I am the responsible investigator, specifying your participation in the research study: **Characterization and Measurement of Capabilities and Processes of Software Start-ups in Incubation Hubs**. You and I have both signed and dated each copy, certifying that we agree to continue this interview. You will receive one copy and I will keep the other under lock and key, separate from your transcribed audio-recordings.

Your participation in this interview is completely voluntary. If at any time you need to stop or take a break, please let me know. You may also withdraw your participation at any time without consequence. Do you have any questions or concerns before we begin? Then with your permission we will begin the interview.

Start-up Challenges

1. What are main challenges you face as start-up?
2. How do you address or intend to address these challenges?
3. What are the technology, process and business practices in your start-up?
4. Why do you choose to use these practices?

Hub operation and practices

5. Describe the technical and business operations in your incubation hub?
6. What is the objective of carrying out these operations?
7. How do your operations influence your start-ups as they mature through the hub?
8. When do you normally carry these activities?
9. What is your observed outcome of each these activities on the start-ups within your hub?
10. How does your hub measure and /or benchmark progress of its start-ups?

Start-up growth

11. When did you found this start-up?
12. What are your milestones and when are you hoping to reach them?
13. Of these milestones what are the milestones that indicate progress in your start-up?
14. What are you currently doing and hope to do to reach these milestones in your start-up?

Measurement in start-ups

15. What are the technical and business capabilities that your start-up team has?
16. Describe the technical processes in your start-up?
17. Describe your current business processes?
18. How does your start-up measure and /or benchmark its processes and capabilities as it grows?
19. What metrics can be used to measure capabilities and processes in your startup?
20. What do you think will be the advantages of measuring the progress of your start-up?

Thank you for your participation in this interview

Appendix C

Software start-ups Interview Guide

Welcome and thank you for your participation today. My name is Kamulegeya Grace B. I am PhD researcher at Makerere University, School of Computing and IT. I am conducting a Study in partial fulfillment of the requirements for the degree of Doctorate of Philosophy.

Start-up Challenges

1. What is the name of your startup/innovation and what does it do (business activities)?
2. What are the main challenges you face as a start-up?
3. How does your startup address or intend to address these challenges?

Start-up growth milestones

4. When did you start this startup?
5. What major changes have you implemented/observed in the startup since you started and when/how did these start?
6. Which of these changes indicate positive progress in your startup and how?
7. Which of these changes have affected your startup negatively and how?
8. What are you currently doing and or intend to do to grow your startup?

Measurement in start-ups

9. What capabilities (experience, technical, business, mentoring skills) does your startup team have?
10. Where do you currently use these different capabilities in your startup?
11. What new capabilities does you need and in which areas/activities of the startup?
12. When do you intend to acquire these capabilities?
13. How does your startup assess/document its capabilities?
14. What incentives and/or capabilities does this hub offer to your startup?
15. How do you use these incentives and capabilities in your startup?
16. Describe the technical processes in your startup?
17. How do you assess/document the progress of your startup?
18. Do you think you are measuring enough of the progress in the startup and What activities/processes would you like to measure progress for?
19. What do you think will be the benefits of measuring these activities or processes in your startup?
20. If you would have a dashboard (visualization continuously documenting the progress of your startup) how would you use it?
21. What benefits will this dashboard give you as a startup?

Thank you for your participation in this interview

Appendix D

Hubs Interview Guide

Hub Activities/practices

1. What is the name of your hub and when did it start?
2. What does your hub do?
3. How many startups do you host at any one time in your hub?
4. How many of them still operate 2 years after their time in your hub?
5. Describe mechanism/process you have to work with the startups that have left your hub?
6. What is the criteria for accepting a startup in your hub?
7. When and how do startups leave your hub?

Activities offered to the start-ups:

8. Describe the incentives (technical and business activities/practices) that your hub offers your startups?
9. What are your main objectives of carrying out these activities/practices and /or giving these incentives?
10. When do you carry out these activities in your hub?
11. What criteria do you to select startups to participate in the offered hub activities or use the offered incentives?
12. What are your observed differences between startups that do and that do not participate in these activities?
13. How do these activities/practices and /or incentives affect the startups as they mature in your hub?
14. What are your observed effects of these activities/Incentives on the past and present startups the hub?

Measurements:

15. (How) do you measure and /or benchmark progress of startups in your hub?
16. What will you want to measure about the progress of each startup in the hub?

Appendix E

Welcome and thank you for your participation in this survey. My name is Kamulegeya Grace B. I am PhD researcher at Makerere University, School of Computing and IT. I am conducting a Study in partial fulfillment of the requirements for the degree of Doctorate of Philosophy. The main objective of this survey is to collect data about the extent to which software companies measure within their processes, services and products.

How old is this company? (Tick one that applies)	<1 Year	1 <=3 Years	> 3 Years	Mature company			
What type of software company are you? (Tick all that apply to your company)	Software as a Service (SaaS)	Mobile App	E-commerce	Fintech/Payment service/solution	Two-sided market place (Linking buyers and Sellers of products and services)	User-generated content	Software Development as service
Choose the measurements that you use in your company?(Tick all that apply)	Product maintenance/support	Ability to build a complete product	Ability to close a business deal	Comparing product versions (added features)	Documenting and reviewing activities for progress	Feedback from friends about product features (peer endorsement)	Product awareness/customer interest	Production process time estimation
	Reaching key business milestones (patents, tax registration, incorporation)	Set and evaluate Key Performance Indicators (KPIs)	Set and evaluate tasks	Set and review business targets	Time-based project performance appraisal	Using a telemetry tool	Using market as a benchmark	Customer analytics (Number of people using platform, customer behaviour)

Please use the numbers corresponding to your answer to fill out the matrix for each measurement that you mention.

No	Name any other measurements that you carry out in your start-up	What type of tools do you use to carry out this measurement? [(1) Manual, (2) Free Open Source Software Tools, (3) Licenced software, (4)Standard licensed tool with in-house additions]	What do you use the measurement for in your company? [(1) Insight only, (2) Decision making, (3) Alert(Redflag), (4) Both Insight and Decision making, (5) Not sure]	In which company phase do you measure and/or apply the measurement? [(1) Ideation/Conceptualization, (2) Design, (3) Implementation/prototyping, (4) Validation, (5) Deployment, (6) Business, (7) Scaling] You can choose more than one phase	Who uses/applies the measurement in your company [(1) Technical team, (2) Business Team, (3) Specialist use]	What is the source of information used in this measurement? [(1) Other companies, (2) Team, (3) Business partners, (4) External friends/peers]	How long does this measurement take to change its value? [(1) Yearly, (2) Monthly, (3) Weekly, (4) A day and less]	How much dedicated time does the team use to measure this? [(1) Lot of time, (2) Moderate time, (3) Little Time]	How much time is needed to set up the procedure for this measurement? [(1) Low, (2) Medium, (3) High]	Do use any tools or know any tools that can automate this measurement [(1) Yes, (2) No, (3) Not sure]	How much investment in tools by your company may you require or currently use for this measurement [(1) Low, (2) Moderate (3) High]	How consistently do you measure this in your company [(1) Repeatedly (2) Not Repeated]	How do you finally get the value of the measurements[(1) One Simple value (2) Use a number of values to obtain the measurement]	Where do you focus this measurement [(1) Company, (2) Project, (3) Product (4) Process (5) Team]	Can you assign a number of or a standard unit to this measurement? [(1) Yes (2) No (3) Not Sure	How automated is the process of getting this measurement [(1) Manual, (2) Can be Automated, (3) Automated (4) Semi-Automated]
1																
2																
3																
4																
5																
6																
7																
8																

Your Compay Name	_____	Are based in a hub?	Yes	No	