Software start-ups through an empirical lens: are start-ups snowflakes?

Eriks Klotins

Blekinge Institute of Technology, Karlskrona, Sweden
eriks.klotins@bth.se

Abstract. Most of the existing research assume that software start-ups are "unique" and require a special approach to software engineering. The uniqueness of start-ups is often justified by the scarcity of resources, time pressure, little operating history, and focus on innovation. As a consequence, most research on software start-ups concentrate on exploring the start-up context and are overlooking the potential of transferring the best engineering practices from other contexts to start-ups.
In this paper, we examine results from an earlier mapping study reporting frequently used terms in literature used to characterize start-ups. We analyze how much empirical evidence support each characteristic, and how unique each characteristic is in the context of innovative, market-driven, software-intensive product development.
Our findings suggest that many of the terms used to describe start-ups originate from anecdotal evidence and have little empirical backing. Therefore, there is a potential to revise the original start-up characterization.
In conclusion, we identify three potential research avenues for further work: a) considering shareholder perspective in product decisions, b) providing support for software engineering in rapidly growing organizations, and c) focusing on transferring the best engineering practices from other contexts to start-ups.

Keywords: start-ups · software engineering · engineering context

1 Introduction

In recent years, software start-ups have gained attention from the research community. In 2014, a systematic mapping study by Paternoster et al. [39] highlighted the lack of relevant research addressing software engineering in start-ups. Results of this paper are reused by most subsequent studies on software start-ups.

In 2016, Unterkalmsteiner et al. [52] published a research agenda identifying further research directions in the area. These directions explore start-ups from software engineering perspective and only superficially touches upon other, e.g. marketing and business, aspects of start-ups. The underlying idea is that the core of a start-up is development and maintenance of a software-intensive product.

Thus, shortcomings in the product development could hinder any subsequent attempts to build a sustainable business around it [27].

Since 2014, a substantial corpus of empirical data on software start-ups has been collected and analyzed, for example, Giardino et al. [17], Klotins et al. [30,28], and Tripathi et al. [51]. Several models are proposed to explain software engineering in start-ups, for example, Giardino et al. [8,18] and Klotins et al. [31].

Most of the recent research on software start-ups focus on exploring engineering context and used practices. The exploration is motivated by the premise that start-ups are "special" and "unique", thus require a special approach to software engineering, for example, Sutton [49], Blank [5], Gralha et al. [19], and Duc et al. [13]. At the same time, systematic adoption of existing engineering practices for use in start-ups had attracted little attention [29,4].

The empirical data, for instance, Coleman et al. [10], Klotins et al. [28,30] and Giardino et al. [17], show little evidence of anything special, regarding software engineering, in start-ups compared to other market-driven organizations developing innovative software-intensive products. Such results invite to revisit the initial premise.

Understanding to what extent software start-ups are different from established organizations is central to transferring the best engineering practices from other contexts to start-ups. If start-ups are different, the differences need to be explored to develop start-up specific engineering practices. If start-ups are not different, further research needs to emphasize the transfer of the best engineering practices from other contexts to start-ups.

There has been a limited success with formulating a crisp and distinctive definition of a software start-up [44,52]. Ries [43] broadly defines start-ups as human institutions aiming to deliver new products or services under extreme uncertainty. Carmel [6] defines start-ups as new, market-driven companies aiming to launch software product fast with minimal resources. Unterkalmsteiner et al. [52] define software start-ups as newly founded companies developing software-intensive products under time and resource pressures. In our earlier study, we define start-ups as small companies created to develop and to market an innovative and software-intensive product and to aim to benefit from economies of scale [28]. These definitions describe software start-ups, however miss to convey any distinctive features.

Blank [5] argues the key difference between start-ups and established organizations is that established organizations aim to execute their business model, while start-ups are searching for one. To software engineers, this difference translates into a focus on iterative development, frequent product releases, and extensive use of customer feedback. A very similar approach is used for market-driven product development in established organizations [12].

Paternoster et al. [39] compile a list of recurring terms describing software start-ups. The terms are, for example, lack of resources and experience, time pressure, small team, high risk of failure among others. This list is often used by later studies, for example, Gralha et al. [19], Giardino et al. [17,8], and Klotins

et al. [30], to define what is a start-up and to justify their uniqueness. However, the list is meant to "illustrate how authors use the term software startup", and does not imply any empirical grounding.

The objective of this study is to examine how much empirical support there is for "unique" characteristics of start-ups. We analyze the list of start-up characteristics proposed by Paternoster et al. [39] and trace the supporting literature. Then, we examine the literature to estimate how much empirical support there is for each characteristic.

The rest of this paper is structured as follows: In Section 2 we examine the terms and the supporting evidence. In Section 3 we discuss our findings. Section 3 concludes the paper.

2 Start-up characteristics

We use the list of recurring terms characterizing software start-ups by Paternoster et al. [39], to drive our analysis. The original list contains the following characteristics:

1. Lack of resources - Economical, human, and physical resources are extremely limited.
2. Highly Reactive - Startups are able to quickly react to changes of the underlying market, technologies, and product (compared to more established companies)
3. Innovation - Given the highly competitive ecosystem, startups need to focus on highly innovative segments of the market.
4. Uncertainty - Startups deal with a highly uncertain ecosystem under different perspectives: market, product features, competition, people and finance.
5. Rapidly Evolving - Successful startups aim to grow and scale rapidly.
6. Time-pressure - The environment often forces startups to release fast and to work under constant pressure (terms sheets, demo days, investors' requests)
7. Third party dependency - Due to lack of resources, to build their product, startups heavily rely on external solutions: External APIs, Open Source Software, outsourcing, COTS, etc.
8. Small Team - Startups start with a small numbers of individuals.
9. One product - Company's activities gravitate around one product/service only.
10. Low-experienced team - A good part of the development team is formed by people with less than 5 years of experience and often recently graduated students.
11. New company - The company has been recently created.
12. Flat organization - Startups are usually founders-centric and everyone in the company has big responsibilities, with no need of high-management.
13. Highly Risky - The failure rate of startups is extremely high.
14. Not self-sustained - Especially in the early stage, startups need external funding to sustain their activities (Venture Capitalist, Angel Investments, Personal Funds, etc.).

15. Little working history - The basis of an organizational culture is not present initially.

For the brevity of our discussion, we group these terms them into 6 categories as some of the terms appear to be related.

In the following subsections, we examine sources of the characteristics. We look into papers, identified by the review [39], to find empirical support each start-up characteristic. In our review, we include both papers listed by the mapping study and any relevant papers referenced by the listed papers. In essence, we attempt to trace the original statement, a piece of data that inspired the formulation of each characteristic. In addition, we discuss to what extent each characteristic is relevant in other types of organizations.

2.1 Lack of resources and dependency on external sponsors

Lack of human, economic, and physical resources to support product engineering is the most frequently used term to describe software start-ups. It is related to dependencies on 3rd parties for funding and having not enough cash-flow to be self sustainable [39].

Following the references, we found 24 papers, of which 17 analyze empirical data. We review these 17 papers to understand what exact empirical data was the basis for claiming that lack of resources and dependency of external sponsors are characteristic to start-ups.

Some of the papers discuss the need or intention to allocate resources to support product engineering, and not the lack of resources as a challenge [55,54,7]. Coleman et al. [11] reference an experience report from a start-up company. The start-up, operating in 1992 was not able to afford then costly Internet connection and had relied on public Internet access elsewhere. May [34] discusses wasted resources in a start-up due to poor work ethics and using sub-optimal technologies. Mudambi et al. [36] and Yoo et al. [56] argue that small organizations have lesser resources at hand than larger organizations and may not yet have a sustainable revenue, thus resource allocation is an ongoing issue. Later studies elaborate on the impact of resource shortages.

Giardino et al. [17] report allocation of resources as one of the Top 10 challenges in start-ups, and elaborates that a studied company was unable to solve some technical problems in the product due to insufficient resources. Lindgren et al. [33] report that start-ups were not able to utilize experimentation to a full extent due to limited resources. Jorgensen [24] report that shortages in human resources caused delays in product development, and a project was canceled due to an insufficient budget.

Related work on project resource management suggests securing sufficient resources is one of the critical steps in project inception and is linked to project success [47]. In both plan-driven and agile environments, the presence of a committed sponsor is one of the key denominators for project success [9]. The trade-offs between features, resources, and quality, are common in any project [25,14]. In this aspect, start-ups do not look any different.

A study investigating the impact on venture capital to start-ups prospects found that external funding has no significant effect on start-up outcome [48]. Therefore, the focus of further research and practice should be on better methods for engineering resource planning, control, and risk management, to make the best use of any amount of resources. Hadley et al. [20] presents similar findings suggesting an association between venture capital and negative long-term consequences.

A report by Harvard Business Review [38] report that venture capitalists prefer investing in start-ups with younger founders, even though the odds of commercial success are with older and more experienced founders. The report points out that younger founders could be more financially constrained, thus be more willing to cede their business to venture capitalists at a lower price. In other words, young and inexperienced founders could provide higher returns of investment for venture capitalists.

The related work so far does not present any convincing evidence that start-ups would experience the trade-off between resources, scope, and quality differently than other organizations [25]. However, the related work suggests that a potential difference between start-ups and established organizations could be that in an established organization project sponsor and the project team are from the same organization, thus share the same goals to serve customers, improve internal efficiency, and fulfill organization's mission. However, start-ups are often funded by other organizations, for example, venture capitalists. Thus, their goals may not always be aligned [48,38].

As shown by Azoulay et al. [38], venture capitalists could aim to maximize their return on investment. Start-up founders, in turn, could be motivated by an intent to bring their ideas to market, desire for autonomy, and need for accomplishment among other factors [15].

2.2 Time pressure

Time pressure is often used in combination with a lack of resources to describe start-ups [37]. The pressure supposedly originates from investors, external deadlines, and contracts. Following the references, we found 13 supporting papers, of which 6 use empirical data [39].

Examining the papers closer, we found that none of the papers use any data to justify the time pressure in start-ups. However, the papers present a discussion motivating the need for faster delivery time to reduce opportunity cost [5,50,40]. Start-ups aim to spend as little time as possible on activities that have an uncertain contribution to customer value, e.g., building invented features.

Giardino et al. [8] identifies development speed as the core concept in start-ups. It is motivated by the need to keep the team's morale high and to validate the product idea fast. Another study by Giardino et al. [17] links time pressure with available resources and the need to establish a sustainable stream of revenue quickly.

These findings suggest that the time pressure originates from internal considerations and resource limitations, and not from competition or external deadlines. Thus, start-ups may have relative freedom to control the development pace and address the trade-off between quality and speed. Concerning time pressure, established companies face the same opportunity costs. However, they may have more resources at hand to sponsor the product development for longer.

2.3 Innovation

Focus on innovative technologies, products, and market segments is another term used to characterize start-ups [39]. Following the references, we identified in 15 papers, of which 9 uses empirical data, concerning innovation in start-ups.

These studies show that start-ups use innovative offerings primarily to differentiate from other competitors in the market [55,7]. The innovation in start-ups is to a large extent incremental and adds slight improvements to an existing product [32,55]. The innovative aspects can concern product features, quality, packaging, and marketing [26].

Continuous innovation, driven by the innovation strategy, is essential to maintain a competitive edge [32,26]. Heitlager et al. [22] argue, albeit without empirical support, that start-ups start with product innovation to enter the market, followed by process innovation to improve efficiency.

Multi-vocal literature recognizes multiple types of innovation, for example, incremental and process innovation, business model innovation, radical and disruptive innovation [1]. Incremental, process and business model innovation appears to be most suited for small organizations as they focus on improving already known features, activities, and business models [26]. However, disruptive and radical innovation requires substantial investments and time to replace existing products and create entirely new markets with new business models [2]. Thus, these types of innovation could be less suited for resource-strapped start-ups.

Regarding innovation, larger organizations may have the leverage to push more ambitious innovations than small start-ups. For example, Apple had created several disruptive innovations by launching its music platform, iPhone, and AppStore. Such innovations were enabled by their experience within the market, human, organizational and economic resources, and their brand name [53]. However, start-ups lack most, if not all, such enablers. Regarding innovation, start-ups may have to be more modest than established organizations [46].

2.4 Rapidly evolving new company

Terms such as rapid evolution, a new company, small and flat team, focus on one product, and little working history are supported by 34 papers, 22 of them analyzing empirical data [39].

There is an agreement among the papers that start-ups are new organizations established by one or a few founders championing the product idea [55,7,34].

More people, resources, and processes are brought in to support product development and customer service. More processes and artifacts are introduced as the organization grows [7,10].

Surprisingly, none of the studies present data illustrating the start-up growth. The growth is extrapolated from interviewee reflections (e.g. Coleman et al. [11]), a generalized model (Carmel. [7]), and plans to grow customer volume and market share rapidly (Yogendra et al. [55]).

Later studies identify evolving engineering practices in start-ups. Gralha et al. [19] and Melegati et al. [35] identify that requirements engineering practices in start-ups develop from informal to more structured as the start-up matures. Giardino et al. [8] identifies a similar pattern in the adoption of agile practices. Early on, start-ups opt for an ad-hoc approach to engineering and introduce new practices as needed. Introduction of new practices and processes impair development speed, however improve coordination and product quality [28,8].

Established organizations, compared to start-ups, are per definition more stable. Although organizational changes occur in established organizations, they are supported by processes, infrastructure, and concern one or few aspects of the organization at the time [58]. Therefore, rapid evolution in multiple aspects at once could be the most substantial difference between start-ups and other types of organizations.

2.5 Lack of experience

Inexperienced start-up teams are reported as a common theme in literature [39]. This term is supported by 7 papers. However, by looking at the papers closer, we found that none of them present any empirical data supporting the statement.

By analyzing the papers, we found several studies presenting data and analysis providing a strong link between the experience of the teams and prospects of start-up success [26,56,7]. More experienced people require less management [10], and are an essential resource for rapid product development [7,3]. However, May [34] and Giardino et al. [17] note that it is not always easy to find skilled and motivated individuals.

A report by Harvard Business Review [38] analyzing a large sample of founders from the US show that most start-up founders are 30 - 50 years old. The average age of commercially successful start-up founder is 45. Authors of the report emphasize the importance of previous experience and acumen to start a new business that comes with older age. Such findings refute the idea of young and inexperienced start-up founders as a typical case.

Other studies add further support for the importance of technical and business experience to start-up success [57,41]. Giardino et al. [8] emphasizes the importance of a small and motivated team of skilled individuals. However, we could not find any evidence that start-ups would have disproportionally more inexperienced engineers than any other type of organization.

Established organizations put substantial effort into on-boarding new software engineers. For example, by providing on-the-job training, mentoring, employee guides, and so on [23]. It could take several months until a recruit reaches

full productivity [16]. A small start-up may lack the capacity to provide such resources to new engineers. As a consequence, start-ups may aim to hire engineers with relevant technical and domain knowledge to compensate for the lack of on-the-job training.

2.6 Highly risky

High risk of failure and uncertainty is identified as characteristic to start-ups is supported 12 studies, of which 8 uses empirical data [39].

Examining the studies further, we found that none of them present any data on start-up failure rate. Blank [5] estimates a 75% failure rate among start-ups and motivates it by a report from Harvard Business School. However, we were not able to find the original report.

Looking further, we found a study reporting small business survival rate of 66% after the first year, and 40% after six years or more [21]. The sample includes all types of recently established small businesses. While exact numbers from different sources vary, they agree that most new companies do not survive past the first few years. That said, we were not able to find any credible source estimating a general failure rate among start-ups.

Carmel [6] emphasizes that launching a new venture is inherently associated with the risk of failure. However, estimating success and failure rate of start-ups is difficult. Likely, many start-up initiatives are closed down before they appear on any records. After closure, there is no evidence left behind to be studied. Part of the difficulty to estimate start-up failure rate is lack of a clear definition of what is a start-up, and what are their success and failure conditions.

Traditional project management literature considers a project successful if it is delivered within budget, time, and scope [47]. The economic perspective on start-ups identifies return of investment as the accurate measure of success [42]. Customer-centric view proposes to use customer satisfaction to assess the project success [45]. Carmel [6] argue that speed is the essential success metric in start-ups.

So far, the related work does not present any evidence that start-ups would have substantially different survival rate than other types of recently established ventures. However, as we have discussed earlier, start-ups may have stakeholders with different interpretations of success. For example, the investors could be looking for specific return of investment ratio. The odds of attaining such specific objectives could be much lower than of general survival of the company.

3 Discussion

We perform this inquiry to understand if there is enough evidence to claim that start-ups are different from established companies and need a different approach to software engineering. We examine 15 start-up characteristics that are often used to define and differentiate start-ups from established organizations.

By reviewing the literature, we identify several common shortcomings. Firstly, many studies present an anecdotal characterization of start-ups. Such characterization of start-ups is often placed in the introduction, motivating the study. Meanwhile, the research itself focuses on different aspects that neither add or remove support for the characteristics. Such anecdotes propagate, are generalized by further studies, and cause misconceptions about engineering start-ups.

Secondly, studies investigating start-ups rarely, if at all, discuss their findings in a broader context. As a consequence, some challenges, for example, lack of resources and innovation, are presented as unique to start-ups. Such narrow focus takes away the opportunity to transfer the best engineering practices from other contexts to start-ups, and vice versa.

By evaluating the actual empirical evidence, we find little support for most of the characteristics. For example, we could not find any empirical evidence showing that start-up teams are inexperienced. Quite the opposite, empirical studies show that start-ups are often founded by middle-aged entrepreneurs with substantial experience and business acumen. Furthermore, some of the characteristics that are presented as "unique" to start-ups are common in other types of organizations. For example, the challenge of balancing project scope with available resources is hardly unique to start-ups. In other words, by examining the literature, we could not find convincing empirical evidence that start-ups would be in any way "unique" regarding software engineering. Such results suggest that the focus of further research should be on transferring the best engineering practices from established organizations to start-ups.

We identify several limitations concerning our study. The start-up characteristics discussed in this paper are based on work by Paternoster et al. [39]. There could be other studies more accurately describing start-ups and emphasizing their distinctive characteristics. However, to our best knowledge, the terms identified by Paternoster et al. are the most commonly used, thus serve as a good enough basis to raise the discussion on what is so special about software engineering in start-ups.

The literature analyzed in this paper is identified by following traceability information provided by Paternoster et al. [39]. There is a threat that this information is incomplete and we may have overlooked some important studies. To address this treat, and explore a concept in a broader context, we perform independent searches for relevant literature.

Our discussion is limited only to software engineering perspective of start-ups. Other perspectives, for example, business, finances, and marketing could present more distinct differences between start-ups and established organizations. Such other perspectives are left out from our discussion.

4 Conclusions and further work

In this paper, we examine the commonly used characteristics to distinguish between start-ups and established organizations. We found that most of the frequently used start-up characteristics have little empirical support, and some

of the characteristics are present in larger organizations as well. We conclude that the terms characterizing software start-ups, and the definition of software start-ups from software engineering perspective need to be revised.

Such finding has implications to our main question whether or not start-ups are special, and should use different engineering practices than small-medium enterprises and other types of organizations. We could not find convincing evidence that start-ups need a different approach to engineering than other types of organizations. We found that rapid evolution and conflicting stakeholders objectives could be adding extra complexity to software engineering. Such additional complexity suggests that start-ups should be more, not less, structured in following the best engineering practices.

From our analysis, we identify three potential research directions concerning software start-ups.

1. Rapid evolution: Growing an organization from a few people to multiple teams working together in a short time requires an evolution of communication and coordination practices as well. Practices that work with few engineers, customers, and a small product, will not suffice in a larger team, thousands of customers and a complex product. There are plenty of engineering practices aimed at dynamic environments, e.g., agile. However, realizing the need for, selection, and continuous adoption of new practices is a major engineering challenge.

2. Thinner margins of error: Given their small size and dependency on external sponsors, start-ups have little margin for errors. The errors may concern both product decisions, e.g., what features and quality to build, and process decisions, e.g., determining the most efficient way of delivering the features. Larger organizations could cover losses of one product with profits from another. And, compensate for inefficient practices with more resources. However, in start-ups failure to deliver customer value quickly usually means the closure of the company. To software engineers, this translates into the need for proven engineering methods, continuous process improvement, stricter control over resource utilization, and better risk management.

3. Misaligned stakeholder objectives: When project sponsors and the project team are from the same organization, they share the same high-level goals, e.g., to serve their customers, and fulfill the company's mission. However, in start-ups project sponsors could be from a different organization, thus may have very different goals. For instance, venture capitalists may aim to maximize the returns of investment, while a start-up could aim to pioneer an innovative technology. To software engineers, this implies the need to balance the interests of different stakeholder groups, namely, customers, shareholders, and the start-up itself.

5 Acknowledgments

Ideas presented in this paper arise from discussions with Dr. M. Unterkalmsteiner, Prof. Dr. T. Gorschek, and members of Software Start-up Research Network[1].

The author would like to thank all the reviewers for their time and valuable input in shaping this paper.

References

1. 15 types of innovation. `https://thegentleartofsmartstealing.wordpress.com/types-of-innovation/`, accessed: 2018-10-05
2. Radical innovation. `https://www.innovation-creativity.com/radical-innovation.html`, accessed: 2018-10-05
3. Ambler, S.: Lessons in agility from Internet-based development. Software, IEEE **19**(2), 66 – 73 (2002)
4. Berg, V., Birkeland, J., Nguyen-Duc, A., Pappas, I., Jaccheri, L.: Software startup engineering: A systematic mapping study. Journal of Systems and Software (2018)
5. Blank, S.: Why the Lean Start Up Changes Everything. Harvard Business Review **91**(5), 64 (2013)
6. Carmel, E.: Rapid development in software package startups. In: Proc. 27th Hawaii Int'l Conf. System Sciences (1994)
7. Carmel, E.: Time-to-completion in software package startups. In: 1994 Proceedings of the Twenty-Seventh Hawaii International Conference on System Sciences (1994)
8. Carmine, G., Paternoster, N., Unterkalmsteiner, M., Gorschek, T., Abrahamsson, P.: Software Development in Startup Companies: The Greenfield Startup Model. IEEE Transactions on Software Engineering **X**(September), 233 (2016)
9. Chow, T., Cao, D.B.: A survey study of critical success factors in agile software projects. Journal of Systems and Software **81**(6), 961–971 (2008)
10. Coleman, G., O'Connor, R.V.: An investigation into software development process formation in software start-ups. Journal of Enterprise Information Management **21**(6), 633–648 (2008)
11. Coleman, G., O'Connor, R.V.: An investigation into software development process formation in software start-ups. Journal of Enterprise Information Management **21**(6), 633–648 (2008)
12. Dahlstedt, Å.G., Karlsson, L., Persson, A., Natt och Dag, J., Regnell, B.: Market-Driven Requirements Engineering Processes for Software Products - a Report on Current Practices. In: International Workshop on COTS and Product Software, RECOTS 2003 (2003)
13. Duc, A.N., Dahle, Y., Steinert, M., Abrahamsson, P.: Towards understanding startup product development as effectual entrepreneurial behaviors. In: International Conference of Software Business. pp. 199–204. Springer (2017)
14. Elonen, S., Artto, K.A.: Problems in managing internal development projects in multi-project environments. International journal of project management **21**(6), 395–402 (2003)
15. Estay, C., Durrieu, F., Akhter, M.: Entrepreneurship: From motivation to start-up. Journal of International Entrepreneurship **11**(3), 243–267 (2013)

[1] Software Start-up Research Network: `http://softwarestartups.org`

16. Ganzel, R.: Putting out the welcome mat. Training **35**(3), 54–60 (1998)
17. Giardino, C., Bajwa, S.S., Wang, X.: Key Challenges in Early-Stage Software Startups. In: Agile Processes, in Software Engineering, and Extreme Programming. vol. 212, pp. 52–63 (2015)
18. Giardino, C., Wang, X., Abrahamsson, P.: Why Early-Stage Software Startups Fail : A Behavioral Framework pp. 27–41 (2014)
19. Gralha, C., Damian, D., Wasserman, A.I.T., Goulão, M., Araújo, J.: The evolution of requirements practices in software startups pp. 823–833 (2018)
20. Hadley, B., Gloor, P.A., Woerner, S.L., Zhou, Y.: Analyzing vc influence on startup success: They might not be good for you
21. Headd, B.: Redefining business success: Distinguishing between closure and failure. Small business economics **21**(1), 51–61 (2003)
22. Heitlager, I., Helms, R., Brinkkemper, S.: A tentative technique for the study and planning of co-evolution in product. In: Software Evolvability, 2007 Third International IEEE Workshop on. pp. 42–47. IEEE (2007)
23. Johnson, M., Senges, M.: Learning to be a programmer in a complex organization: A case study on practice-based learning during the onboarding process at google. Journal of Workplace Learning **22**(3), 180–194 (2010)
24. Jørgensen, K.: Newcomers in a global industry: Challenges of a norwegian game company. Games and Culture p. 1555412017723265 (2017)
25. Junk, W.S.: The Dynamic Balance Between Cost, Schedule, Features, and Quality in Software Development Projects. Computer Science Dept., University of Idaho **SEPM-001** (2000)
26. Kakati, M.: Success criteria in high-tech new ventures. Technovation **23**(5), 447–457 (may 2003)
27. Klotins, E., Unterkalmsteiner, M., Gorschek, T.: Software Engineering Anti-patterns in start-ups. In review by IEEE Software (2017)
28. Klotins, E., Unterkalmsteiner, M., Catzipetrou, P., Gorschek, T., Prikladnicki, R., Tripathi, N., Pompermaier, L.B.: A progression model of software engineering goals , challenges , and practices in start-ups. Transactions of Software Engineering **xxx**(9), 1–22 (2018)
29. Klotins, E., Unterkalmsteiner, M., Gorschek, T.: Software Engineering Knowledge Areas in Startup Companies : a mapping study. In: Lecture Notes in Business Information Processing. pp. 245–257. Springer (2015)
30. Klotins, E., Unterkalmsteiner, M., Gorschek, T.: Software engineering in start-up companies: An analysis of 88 experience reports. Empirical Software Engineering pp. 1–35 (2018)
31. Klotins, E., Unterkalmsteiner, M., Gorschek, T.: Software-intensive product engineering in start-ups: a taxonomy. IEEE Software **35**(4) (2018)
32. Linda, S.l.L.: Chinese entrepreneurship in the internet age: Lessons from alibaba. com. International Science Index, Economics and Management Engineering **4**(12) (2010)
33. Lindgren, E., Münch, J.: Raising the odds of success: the current state of experimentation in product development. Information and Software Technology **77**, 80–91 (2016)
34. May, B.: Applying Lean Startup: An Experience Report – Lean & Lean UX by a UX Veteran: Lessons Learned in Creating & Launching a Complex Consumer App. In: Agile Conference. pp. 141–147. Ieee (aug 2012)
35. Melegati, J., Goldman, A., Paulo, S.: Requirements Engineering in Software Startups : a Grounded Theory approach. In: 2nd International Workshop on Software Startups, Trondheim, Norway (2016)

36. Mudambi, R., Treichel, M.Z.: Cash crisis in newly public internet-based firms: An empirical analysis. Journal of Business Venturing **20**(4), 543–571 (2005)
37. Nguyen-Duc, A., Wang, X., Abrahamsson, P.: What influences the speed of prototyping? an empirical investigation of twenty software startups. In: International Conference on Agile Software Development. pp. 20–36. Springer (2017)
38. P. Azoulay, B. Jones, J.D.K.J.M.: Research: The average age of a successful startup founder is 45. Harvard Business Review (Jul 2018)
39. Paternoster, N., Giardino, C., Unterkalmsteiner, M., Gorschek, T., Abrahamsson, P.: Software development in startup companies: A systematic mapping study. Information and Software Technology **56**(10), 1200–1218 (oct 2014)
40. Payne, J.W., Bettman, J.R., Luce, M.F.: When time is money: Decision behavior under opportunity-cost time pressure. Organizational behavior and human decision processes **66**(2), 131–152 (1996)
41. Politis, D.: Does prior start-up experience matter for entrepreneurs' learning? a comparison between novice and habitual entrepreneurs. Journal of small business and Enterprise Development **15**(3), 472–489 (2008)
42. Reid, G.C., Smith, J.A.: What makes a new business start-up successful? Small Business Economics **14**(3), 165–182 (2000)
43. Ries, E.: The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses. Crown busi edn. (2011)
44. Sánchez-Gordón, M.L., Colomo-Palacios, R., de Amescua Seco, A., O'Connor, R.V.: The route to software process improvement in small-and medium-sized enterprises. In: Managing Software Process Evolution, pp. 109–136. Springer (2016)
45. Sauvola, T., Lwakatare, L.E., Karvonen, T., Kuvaja, P., Olsson, H.H., Bosch, J., Oivo, M.: Towards customer-centric software development: a multiple-case study. In: Software Engineering and Advanced Applications (SEAA), 2015 41st Euromicro Conference on. pp. 9–17. IEEE (2015)
46. Slinger Jansen, Sjaak Brinkkemper, Ivo Hunink, C.D.: Pragmatic and Opportunistic Reuse in Innovative Start-up Companies. IEEE Software pp. 42–49 (2008)
47. Snyder, C.S.: A guide to the project management body of knowledge: Pmbok (®) guide. Project Management Institute: Newtown Square, PA, USA (2014)
48. Suominen, A., Hyrynsalmi, S., Still, K., Aarikka-stenroos, L.: Software Start-up failure An exploratory study on the impact of investment pp. 55–64 (2017)
49. Sutton, S.M., Cubed, E.C., Andretti, M.: The Role of Process in a Software Start-up. IEEE Software **17**(4), 33–39 (2000)
50. Tingling, P., Saeed, A.: Extreme programming in action: a longitudinal case study. In: HCI International. pp. 242–251 (2007)
51. Tripathi, N., Klotins, E., Prikladnicki, R., Oivo, M., Pompermaier, L.B., Kudakacheril, A.S., Unterkalmsteiner, M., Liukkunen, K., Gorschek, T.: An anatomy of requirements engineering in software startups using multi-vocal literature and case survey. Journal of Systems and Software (2018)
52. Unterkalmsteiner, M., Abrahamsson, P., Wang, X., Nguyen-Duc, A., Shah, S., Bajwa, S.S., Baltes, G.H., Conboy, K., Cullina, E., Dennehy, D., et al.: Software startups–a research agenda. e-Informatica Software Engineering Journal **10**(1) (2016)
53. West, J., Mace, M.: Entering a mature industry through innovation: Apple's iphone strategy. In: DRUID Summer Conference. pp. 18–20 (2007)
54. Yoffie, D.B., Cusumano, M.A.: Building a company on internet time: Lessons from netscape. California Management Review **41**(3), 8–28 (1999)

55. Yogendra, S., Sengupta, S.: Aligning business and technology strategies: a comparison of established and start-up business contexts. In: Engineering Management Conference, 2002. IEMC'02. 2002 IEEE International. vol. 1, pp. 2–7. IEEE (2002)
56. Yoo, C., Yang, D., Kim, H., Heo, E.: Key value drivers of startup companies in the new media industry - the case of online games in korea. Journal of Media Economics **25**(4), 244–260 (2012)
57. Zhang, J.: The advantage of experienced start-up founders in venture capital acquisition: evidence from serial entrepreneurs. Small Business Economics **36**(2), 187–208 (2011)
58. Zhou, K.Z., David, K.T., Li, J.J.: Organizational changes in emerging economies: Drivers and consequences. Journal of International Business Studies **37**(2), 248–263 (2006)