UPPSALA
UNIVERSITET

Exploring the Perceived Success Factors of Agile Software Projects in Indonesian Startups

Nuruddin Kamil

Master Programme in Industrial Management and Innovation

Masterprogram i industriell ledning och innovation

Abstract

Exploring the Perceived Success Factors of Agile Software Projects in Indonesian Startups

Nuruddin Kamil

UPPSALA
UNIVERSITET

Faculty of Science and Technology

Visiting address:
Ångströmlaboratoriet
Lägerhyddsvägen 1
House 4, Level 0

Postal address:
Box 536
751 21 Uppsala

Telephone:
+46 (0)18 – 471 30 03

Telefax:
+46 (0)18 – 471 30 00

Web page:
http://www.teknik.uu.se/student-en/

A startup is a fledgling business designed to deliver a new product or service under conditions of extreme uncertainty. In the 21st century, startups have been growing significantly both in developed and developing countries. Since 2014, startups are currently a new booming phenomenon in Indonesia. Nowadays, there are 1,716 registered startups that have been rapidly growing in Indonesia which also places the country in 4th place worldwide after the United States, India, and the United Kingdom. Startups in Indonesia were forecasted to grow significantly due to a strong macroeconomic outlook and favorable demographics.

This research is a qualitative study to provide an academic literature for Indonesian startups, especially for agile software development. Regarding the high failure rate for startups, the purpose of this study is to explore the perceived success factors of Indonesian agile development startups. It also covers the adoption of implementing agile project management. This study was conducted through interviews from 20 participants within various IT related startups in Indonesia.

The implementation of agile methodologies has been broadly found in software development projects in Indonesian startups with Scrum as the common agile practice. Most of them have been using agile methods for less than a year. Contrarily, a minority of the participants refused to use agile methodologies and chose a traditional way, such as the waterfall method. It has been investigated that there are six perceived critical success factors that were found in implementing agile methodologies, as follows: project definition process, management commitment, project schedule, team environment, team capability, and delivery strategy. In addition, there is a positive correlation between those critical success factors with previous studies and startup characteristics.

Keywords: agile software development, startups, Indonesia, success factor

Popular Science Summary

Due to newcomer business, a startup has the potential to grow the value of the competitive business through innovation and creativity, but it also needs to struggle with extreme uncertainty conditions. In Indonesia, startups currently become a phenomenon since 2014 with more than thousands of this new businesses were registered in government. Startups are growing economic sector and contribute around 7.44% to the total GDP in 2016 which was better than Russia, Singapore, or even Canada. In addition, the products or services from startups also contributed to the national export as big as USD 20 billion, increased by 3.23% from 2015's USD 19.4 billion. Despite this fact, the survival rate of startups in this country is still disconcerting whereas 90% of startups were failed. Thus, this research is aiming to provide an academic literature for Indonesia startups regarding success factors, so that they have higher chances to survive and grow.

Semi-structured interviews were conducted with 20 participants from various IT-related startups in Indonesia. Based on the interviews, it was found that 16 startups have already implemented agile methodologies while other 4 startups are refused to use agile and chose waterfall method as alternatives. There are 5 main reasons following the implementation of agile methodologies as follows: (1) Based on the mentor's preferences, (2) The agile workflow has helped them develop the product efficiently, (3) startups have a simple organization that supports a quick response to the customer's request, (4) they have sufficient number of team members and most of them are already familiar with agile methodologies, and (5) IT-related startups are dynamic industries which needs a quick development that allows rapid changes during the project. Contrarily, there are 4 non-agile users mentioned limited team members and most of their clients are used waterfall method as the main reasons for choosing waterfall. Even though the agile methodologies have been commonly used, it brings some internal and external challenges for agile users. Internal challenges described as agile techniques, team members, less accurate timelines, and financial funding. External challenge covers the customer involvement if the customer has limited knowledge in the software development process.

Regarding the perceived success factors, there are 6 main critical factors are found to be: (1) Project Definition Process, (2) Management Commitment, (3) Project Schedule, (4) Team Environment, (5) Team Capability, and (6) Delivery Strategy. Those critical success factors were compared to the previous study by Chow and Cao (2008) in which three critical factors were different. Chow and Cao (2008) research stated that agile software techniques, project management process, and customer involvement were critical factors, but in this research, it has been replaced by project definition process, management commitment, and project schedule. Those factors were not importantly appeared due to Indonesian startup characteristics: (1) limited resources lead to adjusting the agile techniques, (2) limited knowledge on the part of the customers, (3) the project management has already given a clear instruction for the workflow and documentation. Another comparison study was conducted by Safwan et al (2013) which focuses on reasons and challenges during agile implementation. As for the reasons, only one reason found to be different from Safwan et al (2013) which is keeping up with the changing environment. Furthermore, there are three differences in challenges: Safwan et al. stated poor quality, insufficient team support, and repetitive work, while this study found agile techniques (daily standup meetings, Scrum Master), communication between team members, and customer involvement are also the challenges.

Acknowledgment

Uppsala, May 2018

Nuruddin Kamil

This thesis is dedicated to my wife Khalda Az Zahra and our upcoming baby,

for without them I would not know how to chase my own dream

and always try to be the best version of myself.

Table of contents

Figures

Tables

Abbreviations

- Bekraf: Badan Ekonomi Kreatif
- Fintech: Financial Technology
- GDP: Growth Domestic Product
- IT: Information Technology
- MVP: Minimum Viable Product
- QA: Quality Assurance
- XP: eXtreme Programming

1. Introduction

1.1. Background

In the 21st century, startups have been growing significantly both in developed and developing countries. In developed countries like the United States – home to information technology (IT) companies such as Google, Facebook and Apple that began as startups – the growing number of startups provides economic impact to the movement of financial resources within the country. Accordingly, there are 298,800 angel investors which lead to 86% startups receiving huge investment value (Moroni, et al., 2015). As an example, Uber – a peer-to-peer ride-sharing startup from California, U.S.A. – has had USD 9.3 billion invested by Softbank, which increased Uber's valuation up to USD 70 billion (Newcomer, 2018). In addition, 60% of startups don't survive the first five years, and 75% of venture capital-funded startups fail (Giardino, et al., 2014). However, in the US, startup failures can be seen as lessons learned, which eventually becomes an asset.

However, the condition of startups in developing countries is slightly different. As an example, 50% of startups in Brazil close within the first four years. There are approximately 6,450 angel investors with only 25% of the number of startups receiving an investment. It is estimated that there are USD 2 billion which represent 0.4% of Brazil's GDP transportation (Moroni, et al., 2015). Thus, in a developing country, startups have provided the social impact that solves local social problems, such as Easy Taxi which provides a better taxi service in a secure and quick way using mobile apps compared to local transportation.

Since 2014, startups are currently a new booming phenomenon in Indonesia. A startup can be defined as a fledgling business that may form as a company, a partnership, or a temporary organization that has a development in capacity, technical, and entrepreneurial management (Badan Ekonomi Kreatif, 2016). It also has the potential to grow the value of the competitive business through innovation and creativity over a period of time. In 2016, *Badan Ekonomi Kreatif* estimated the number of IT related startups operating since 2014 has reached 3,833 startups (Badan Ekonomi Kreatif, 2017). Additionally, Indonesia has 1,716 registered startups, which places it in 4th place worldwide after the United States, India, and the United Kingdom (StartupRanking, 2018). Looking at the future, the Center of Human Genetic Research (CHGR) in Rahayu (2017) projected a threefold increase in the number of startups by 2020.

Based on the research from *Badan Ekonomi Kreatif* (2018), startups are growing the economic sector that contributes around 7.44% to the total GDP in 2016 (Sabdarini, 2018). In detail, Figure 1 shows the 7.44% of GDP contribution represented by USD 65.2 billion. The amount of contribution increases by roughly 10% each year from 2010 to 2016. Furthermore, Figure 2 shows the comparison of Indonesian startups' contribution to the GDP value with other countries in 2016, and Indonesia's was better than Russia, Singapore, or even Canada. In addition, the products or services from startups also contributed to the national total export as big as USD 20 billion, increased by 3.23% from 2015's USD 19.4 billion.
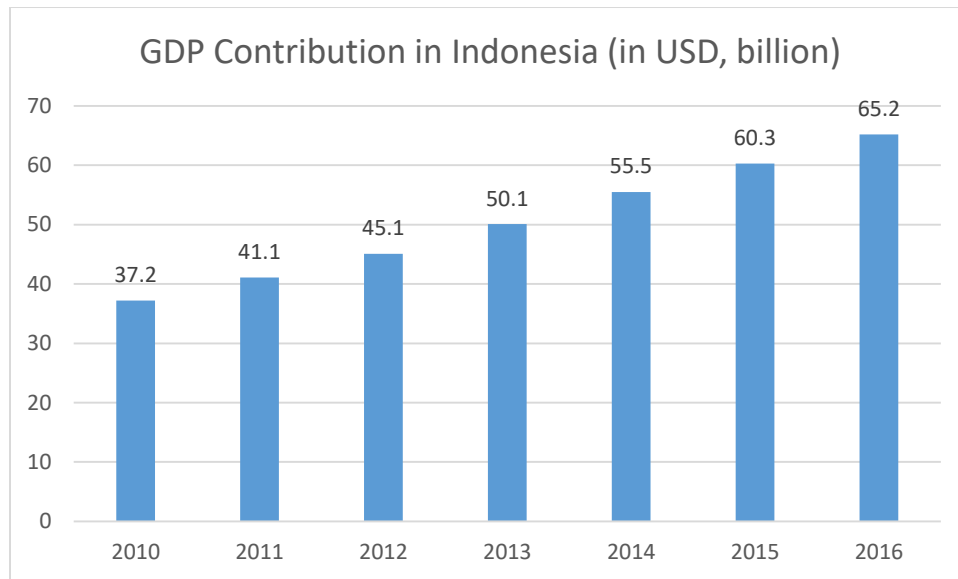
Figure 1. GDP contribution from startups in Indonesia (in billion USD), modified from (Sabdarini, 2018)
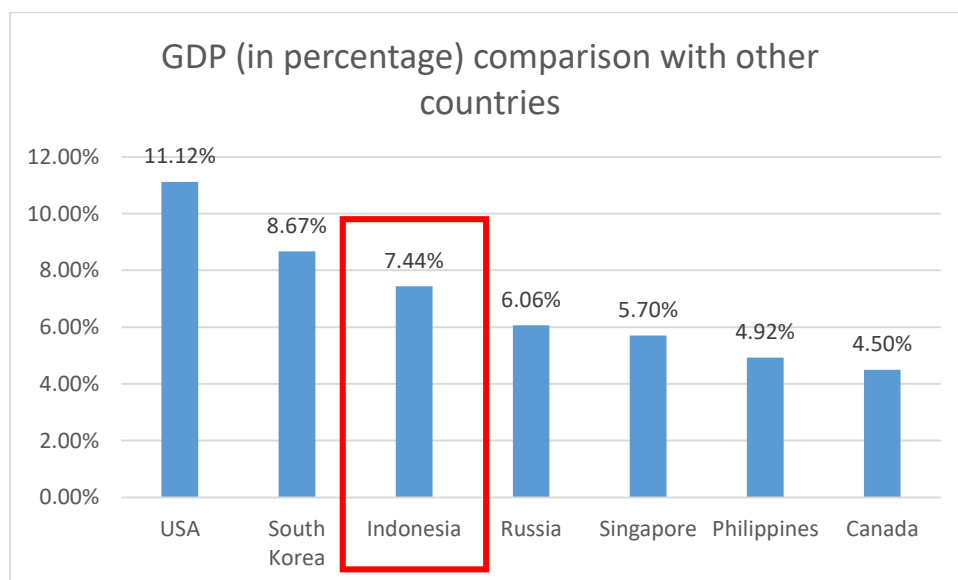
Figure 2. The comparison of GDP contribution from startups in multiple countries, modified from (Sabdarini, 2018)

Furthermore, this growing economic sector attracts a large number of people to work within this sector. There is an increasing number of workforce that is going into the startup business from 15.96 million people in 2015 to 16.91 million people in 2016, an increase of 5.95% (Sabdarini, 2018). Moreover, based on the research from A.T Kearney and Google (2017), startups will grow significantly in Indonesia due to a strong macroeconomic outlook and favorable demographics.
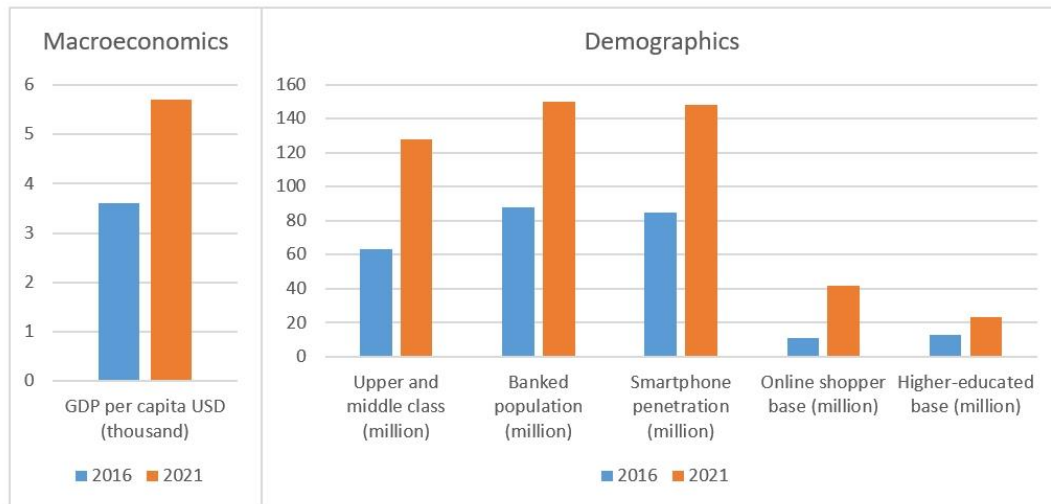
Figure 3. Indonesia consumer trends 2016 vs. 2021, modified from (A.T. Kearney and Google, 2017)

Based on Figure 3, on the macroeconomic side GDP per capita will increase 1.5 times. Indonesia becomes the largest GDP in Southeast Asia with USD 932 trillion (World Bank, 2016). Moreover, on the demographics side, almost all aspects will increase twofold with the exception of online shopping, which will instead increase threefold. This huge potential attracts investment from foreign and local venture capital. The investment number increased twofold from USD 1.4 billion to USD 3.0 billion.

The government plays a big role in supporting the startup ecosystem in Indonesia. Thus in 2015, *Badan Ekonomi Kreatif* (Eng: *The Body of Creative Economy*, officially shortened to Bekraf) was founded by Joko Widodo as the President of Indonesia with the function of helping startups in several aspects, including research and development of products, funding, infrastructure establishment, intellectual property protection, marketing, and branding training (Badan Ekonomi Kreatif, 2018). There is also another organization called Kibar who initiated the *Gerakan Nasional* 1000 Startup Digital (Eng: *National Movement for 1000 Digital Startups*) (Kibar, 2018). It is also supported by the government under the Ministry of Communication and Information Technology in Indonesia.

As the opportunity arises, a startup that operates in software development should be addressing agile approaches to run their projects (Giardino, et al., 2014). As the main characteristics of startups are fast-moving and rapidly-evolving, their final goals often change. The people within the team also play multiple roles to give quick responses to all challenges at any moment. Those conditions may lead startups into chaos, and so they need an approach to keep organized within an unstable environment. Regarding these characteristics, agile approaches offer a solution for startups as they are able to keep up and facilitate their rapid pace of development. It also focuses on planning and achieving short term goals, getting fast results and feedback, and also adapting strategies accordingly (Pentalog, 2017). In summary, agile approaches are able to adapt faster, delivering little and often, testing the environment and remaining customer-focused (Agile Business Consortium, 2018).

Agile methodologies were initially used for software development projects (MacCormack, et al., 2001). This fits well with startups (where there are typically fewer than 20 people) focus on the customer and also the iterative delivery of the product. Agile practices have a wide variety of lightweight approaches *e.g.* scrum, kanban, extreme programming, and fuller approaches e.g. dynamic sys-development method, agile project management, and scaled agile framework.

A well-known startup who applied agile software development in Indonesia is Go-Jek (GOJEK INDONESIA, 2017). This startup was originally made to answer the needs of safe, reliable and cheap two-wheeled transport in 2010, which is now transforming into a complete on-demand service ranging from transportation, electronic payment system and lifestyle in 2016. Go-Jek became the first unicorn[1] startup after securing USD 550 million and its valuation was increased to USD 1.3 Billion (DailySocial, 2016). Go-Jek's main product is an online transportation platform named GO-JEK mobile application. Agile helps Go-Jek define the user's needs and then translate it into user story. Moreover it leads the delivery of new features in short iterations (Ponnappa, 2017). This example indicates that implementing agile software development in developing the GO-JEK mobile application led to the project's success, in terms of quality and time. In regards of project successes, learning the success factors from various software development projects in Indonesian startups can help a new startup to have a higher chance of survival, since 90% of Indonesian startups fail (Pratomo, 2016).

1.2. Aim and purposes

There have been few formal studies on Indonesian startups *per se*, based on recent searches in peer-reviewed academic literature or practitioners related to this topic. There is a similar situation with agile software development projects focusing on Indonesian startups, it is different than other countries such as Sri Lanka, therefore it needs of investigation. The researcher aims to fill these gaps and provide an academic literature for Indonesian startups, especially for agile software development projects.

The purpose of this study is to explore the perceived success factors of agile software development projects in Indonesian startups. To do that, the researcher needs to understand the adoption and the challenges of implementing agile project management. The researcher hopes that by providing academic literature regarding success factors, Indonesian startups can have higher chances to survive and grow.

1.3. Research questions

The research questions are as follows:

1. What is the adoption of agile software development projects in Indonesian startups?

2. What are the challenges of implementing agile software development projects in Indonesian startups?

3. What are the perceived success factors of agile software development projects in Indonesian startups?

---

[1]Unicorns are private companies valued at $1 billion or more. The billion-dollar technology startup was once the stuff of myth (FORTUNE, 2016).

2. Literature Review

2.1. Startup

This section explains the definition of a startup, the startup development phase, the software development startups, and the characteristics of Indonesian software development startups.

2.1.1. Definition

In today's digital era, startups evolve incredibly quickly. Thus, it emphasizes the startup to focus on innovation, integrated business and information technology, and faster agile capability (Aron & McDonald, 2014).Therefore, to provide a better understanding, the definition of startup has been introduced by practitioners and organizations, as follows:

A startup is a temporary organization used to search for a repeatable and scalable business model
(Blank, 2013)

A startup is a human institution designed to deliver a new product or service under conditions of extreme uncertainty

(Ries, 2011)

The startup is a fledgling business that may form as a company, a partnership or a temporary organization that has a development in capacity, technical, and entrepreneur managerial with the potential to grow the value of the competitive business through innovation and creativity over a period of time

(Badan Ekonomi Kreatif, 2016)

The environment for startups is characterized by uncertainty and high risk of mortality in the first year. Hence, startups need to remain creative in its innovative products or services and seek financing for their projects to become profitable and scalable. Consequently, financing becomes the first difference between startups and small-medium enterprises, as the small-medium enterprise is self-financed by generating revenue early on and embarks on a path of slow and steady growth, which focuses on providing a livelihood for the business owner (Lee, 2015).

The processes in startups are all about experimenting and testing the business model. Sometimes, the entrepreneur is uncertain about the target group, so they need to check out the market's niche and learn how to maximize the revenue from the target customer. The startups are then able to grow faster once the business model is settled. By contrast, small-medium enterprises are more permanent and start with a structured organization which focuses on the delivery of value to its already-known customers. It has a stable and successful business model (Entrepreneurinsight, 2016). In addition, top objectives become the biggest difference between startups and small-medium enterprises. Small businesses are driven by profitability and stable long-term value, while startups are focused on top-end revenue and growth potential (Thanedar, 2012). Moreover, being newly founded does not in itself make a startup. There are two main characteristics of high uncertainty and rapid evolution for startups which differentiate them

from established companies (Giardino, et al., 2014). Startups are also different to companies in organizational entity, goals, from employees to culture (Blank, 2010). The differences between startups and company will be further explained in section 2.1.3.

2.1.2. Startup lifecycle

Based on Marmer, who founded the Startup Genome project, a new framework for understanding the startup lifecycle can be defined. Marmer's startup model is a modification of Steve Blank's Customer Developmental Model but differs in that it is product-centric rather than company-centric. This project surveyed over 650 startups as correspondents (Marmer, et al., 2012). Figure 4 shows four stages of the startup lifecycle.
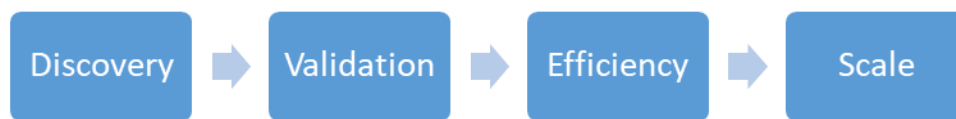
Figure 4. Marmer's startup stages model (Marmer, et al., 2012)

Discovery (5-7 Months)

The first stage is discovery. The goals of this stage are validating their problem definition and checking whether people will be interested in their solution. Meanwhile, the actions that need to be taken in this stage are: forming a founding team, interviewing many customers, finding the value proposition, creating minimally viable products, joining an accelerator or incubator, having a round of friends and family financing, and having the first mentors & advisors come on board.

Validation (3-5 Months)

The second stage is validation. The goals of this stage are validating that people are interested in their product. Meanwhile, the actions that need to be taken in this stage are: a refinement of core features, implementing initial user growth, metrics, and analytics, seed funding, the first key hires, pivots (if necessary), acquiring the first paying customers, and product market fitting.

Efficiency (5-6) Months)

The third stage is efficiency. The goals of this stage are refining the business model and improving the efficiency of the customer acquisition process. Meanwhile, the actions that need to be taken at this stage are: refining the value proposition, overhauling the user experience, optimizing the conversion funnel, achieving viral growth, and finding repeatable sales process and/or scalable customer acquisition channels.

Scale (7-9 Months)

The fourth stage is scale. The goals of this stage are driving growth aggressively. Meanwhile, the actions that need to be taken at this stage are a massive customer acquisition, back-end scalability improvements, the first executive hires, process implementation, and the establishment of departments.

2.1.3. Software development startup

A software development startup is a newly created company which develops a piece of software under highly uncertain conditions and limited resources to tackle fast-growing markets. Software development startups have a distinct combination of characteristics which embody several challenges of software development activities.

Startups are not a smaller version of a large company and that startups are a different organizational entity than a large established company. While startups explore and test business models (Entrepreneurinsight, 2016), the company is known and has proven profitability and is cash-flow positive (Blank, 2010). It is possible for startups to transform into established companies through a transition phase, though this phase can only be achieved when the startups are already scalable. The scalable startups mean all their activities cover customer development, agile development, pivots, search for repeatability, scale up, and stable business model. When startups enter the transition phase and are preparing to move further into being a company, they need to have new senior management, cash-flow breakeven, profitable, rapid scale, and having at least 150 employees (Blank, 2010).

Startups seek to generate revenue and obtain funding to continue the development, which means that software quality is not their critical concern because they will update their product through incremental updates. Engineering activities must allow flexibility and reactivity in development workflows. On the other hand, flexible and reactive methods are designed to stimulate customer feedback which increases the number of perspectives and solutions available to the decision makers. There are common practices to support the rapidly-evolving atmosphere (Giardino, et al., 2014):

- Use of well-known frameworks to quickly change the product following the market needs
- Use of evolutionary prototyping and experimentations via existing components
- Ongoing customer acceptance through early adopter focus groups
- Continuous value delivery, focusing on core functionalities that engage paying customers
- Empowerment of teams to influence final outcomes
- Use of metrics to quickly learn from consumer's feedback and demand
- Use of easy-to-implement tools to facilitate product development and handle fast-paced, changing information.

In addition, there are fifteen characteristics representing software development startups that are ranked from the highest to the lowest impact within these startups, as follows (Paternoster, et al., 2014):

- **Lack of resources**: Extremely limited in economic, human, and physical resources.
- **Highly reactive**: Startups have a quick reaction to changes in the underlying market, technologies, and product compared to established companies.
- **Innovation**: As startups grow in the highly competitive atmosphere, they need to focus on highly innovative segments.
- **Uncertainty**: Startups deal with highly uncertain ecosystems under different perspectives such as market, product features, competition, people, and finance.
- **Rapidly evolving**: Successful startups aim to grow and scale rapidly.
- **Time pressure**: The environment often forces startups to release fast and to work under constant pressure such as terms sheets, demo days, and investor requests.
- **Third-party dependency**: Since they lack resources to build and develop their product, startups heavily rely on external solutions.
- **Small team**: Startups begin with only a few people in the team.

- **One product**: Startups focus on one product or service only.
- **Low-experienced team**: The team consists of people with less than 5 years of experience and often recently graduated students.
- **New company**: The company was recently created.
- **Flat organization**: The structural organization is founder-centric and everyone has big responsibilities, with high management being unnecessary.
- **Highly risky**: The failure rate is extremely high.
- **Not self-sustained**: Startups need external funding to sustain their activities, especially in the early stage (Venture Capitalist, Angel Investments, Personal Funds, *etc.*).
- **Little working history**: The basis of an organizational culture is not present initially.

2.1.4. Indonesian software development startup characteristics

Bekraf (2017) has studied the characteristics of Indonesian software development startups. This includes not only startups who develop software as it is, but also mobile applications and game developers as well. There are several software development startup characteristics, see table 1:

Table 1. Indonesian startup characteristics (Badan Ekonomi Kreatif, 2017)

Characteristics	Explanation
Gender	75% man 25% woman
Startup size	87% ~ 1-4 labor 9% ~ 5-19 labor 3% ~ 20-99 labor 1% ~ >100 labor.
Business entity	83.3% temporary organization 9.8% limited liability company 3.5% Commanditaire Vennootschap 3% special permission business 0.4% other business entities
Revenue	87.5% ~ revenue ≤ USD 21,214 8.3% ~ USD 21,214 < revenue ≤ USD 176,790 3.5% ~ USD 176,790 < revenue ≤ USD 3.5 Million 0.7% ~ revenue > USD 3.5 Million

The most popular Indonesian startups are on-demand services, financial technology (fintech), and e-commerce, software house (Daily Social, 2016; Eka, 2017).

The on-demand service marketplace is significantly growing. On demand service is defined as activity created by technology startup that fulfill consumer demand via the immediate provisioning of goods and services. One of the most successful startups is Go-Jek, which continues to build a platform to facilitate "anything on demand" with a total of 14 services. Go-Jek is a leading service that covers transportation on demand such as Go-Jek, Go-Car,Go-Food, *etc*. As for Go-Food, it is established as

the market leader for food delivery services, leaving FoodPanda and Klik-Eat behind. FoodPanda Indonesia closed its service in October 2016.

Since the establishment of the Indonesia Fintech Association in September 2015, the growth of fintech players in the following year has reached 78%, with around 140 fintech player registered by the end of 2016. Fintech is a type of innovation in financial services. The example of fintech is Amartha who gives peer to peer lending to their customer. On the regulator side, both Bank Indonesia (BI) and the Financial Services Authorities (OJK) launched many programs to secure fintech ecosystems, such as the regulation of the financial services authority (POJK) and BI Fintech Office. Based on the Financial Services Authorities (OJK), the most popular fintech sector is peer-to-peer lending and payment.

Indonesia's e-commerce market has also boomed within the last few years. E-commerce itself is the online version of commerce - buying and selling goods using online platforms or websites. There are already established e-commerce startups such as Tokopedia, Bukalapak, Olx, *etc.* Indonesian e-commerce projection in 2016 is USD157 billion, which is bigger than the original projection of USD88 billion estimated for all six South East Asia markets combined. The entry barrier to joining the general marketplace is too high (in terms of investment amount), and thus many new players focus on niche segments.

As for the software house, it is defined as a startup that focuses on software development. Software houses make software for another company/startup. On the other hand, game developers are startups that build specific software, which are games.

On the other hand, the Indonesian government under the command of Joko Widodo as the President of Indonesia launched a vision for Indonesia to become "The Digital Energy of Asia". In order to support that vision, the Indonesian government under the Ministry of Communication and Information Technology in Indonesia and Kibar founded *Gerakan Nasional 1000 Digital Startup* (Eng: *National Movement for 1000 Digital Startup*). This movement has 5 stages, which are ignition, workshop, hackathon, bootcamp, and incubation. Ignition consist of initial seminar about entrepreneurship for Indonesian and targeted 4000 participants which 2000 participants will be selected for the next stage. The second stage is workshop which consists materials needed to develop their startup such as Design Thinking, Market Validation, and Business Model, then 1000 participants will be selected for Hackathon stage. In the hackathon stage, participants (young entrepreneur) were expected to build a prototype from their idea, then half of them will be selected for the bootcamp stage. Bootcamp stage consists of mentoring session for young entrepreneur, also within this stage, an agile management was recommended by mentors to be applied in the startups as it suitable for startups which allows rapid changing due to customer needs. In this stage, 200 young entrepreneurs will be selected for the final stage, incubation. Incubation provides training for the startups owner and prepared them to get sufficient funding from the investors (Pratama, 2016).

In addition, *Badan Ekonomi Kreatif* was founded by the Indonesian government with the function of helping startups in several aspects, including research and development of products, funding, infrastructure establishment, intellectual property protection, marketing, and branding training (Badan Ekonomi Kreatif, 2018). BEKUP (BEKRAF for Pre-Startup) program was initiated in 2017 by BEKRAF supported by MIKTI (Masyarakat Industri Kreatif TIK/Digital Indonesia, *Eng: Creative Industrial Community for Information and Communication Technology/Digital Indonesia*) and PT. Telekomunikasi Indonesia. This program aimed to give support and advice to young entrepreneur who interested in digital startup in Indonesia through assistantship program to reduce the failure risk that

startup commonly faced. BEKUP which also involving 10 incubators, accelerator, and investor targeted to develop 450 young entrepreneurs with 150 newly digital products (Majalah CSR, 2017).

There are 105 mentors were assigned and trained to accompany young entrepreneurs developing their startups. In mentor' training, the agile project management was introduced as a suitable project management for startup especially in Indonesia. BEKUP has 3 programs, 3 curriculums, and 4 stages within 4 months. Three programs including BEKUP BASIC, BEKUP START, and BEKUP JOURNEY. BEKUP BASIC aimed for young entrepreneurs who need some advices to develop startup in technical part through programming curriculum and creative design in UI/UX. BEKUP START aimed for young entrepreneur who already have team and idea to build a startup through extended curriculum such as technical (android programming and backend php), creative design in UI/UX, and business. Also within this curriculum, there are four sustainable stages as follows: workshop, talent development, founder preparation, pre-incubation. BEKUP JOURNEY aimed for startup that already develop a product which will be advised through extend assistantship program and link-and-match with potential partner (Majalah CSR, 2017).

Minister of Communications and Information Technology explained that government role is currently moved from regulator to become a facilitator and accelerator for digital startups. This statement was mentioned in Minister's speech at National Seminar which was held by Indonesian Telecommunication company in 2018. Also in his speech, the government also promise to give an easy access and simplified the regulation to create a higher opportunity for newly startups. In addition, the government released affirmative regulation for IT developer with accelerates the launching of 4G technology, build High Throughput Satellite, and stimulate the development of newly startups including Over the Top services (Kominfo, 2018). Also, the government launched regulation No. 11 in 2008 about information and electronic transaction (UU ITE No.11 Tahun 2008), Regulation No.7 in 2014 about Trading (UU No. 7 Tahun 2014), and President's Commands No. 74 in 2017 about Road Map e-Commerce for 2017-2019 (Perpres No.74 Tahun 2017) (Okezone, 2017). There are also private sectors such as Google has a free Android course for Indonesian startups called Indonesia Android Kejar.

2.2. Agile project management

Agile Project Management is an approach based on a set of principles, whose goal is to render the process of project management simpler, more flexible and iterative in order to achieve better performance (cost, time and quality), with less management effort and higher levels of innovation and added value for the customer (Conforto, et al., 2014). In order to get a better understanding of agile project management, the researcher will explain the framework, practices, agile software development as well as the success factors of agile software development.

There has been a limited number of previous studies on the implementation of agile project management in Indonesia. However, there are several studies that have been conducted to evaluate the implementation of agile methodologies in software development projects, which will be further explain in section 2.2.4.

The main characteristics of startups are fast-moving and rapidly-evolving, and thus their final goals often change. The agile project management is suitable for startups, especially software development startups, since this project management is able to keep up and facilitate the rapid pace of development process (Pentalog, 2017). The implementation of agile project management in software development projects could lead to project success. When there is a project that succeeds because of implementing

agile methodologies, it means that agile could be used in other projects. Furthermore, when several projects have been successful, then startup success must be achieved (Chow & Cao, 2008).

2.2.1. Agile framework

The agile framework consists of 'being' agile and 'doing' agile. 'Being' agile implies having an agile mindset and following agile values complete with the principles. On the other hand, 'doing' agile implies implementing the agile practices that can be seen as an activity done by both the management and the team (Ranganath, 2011). Figure 5 shows agile project management framework
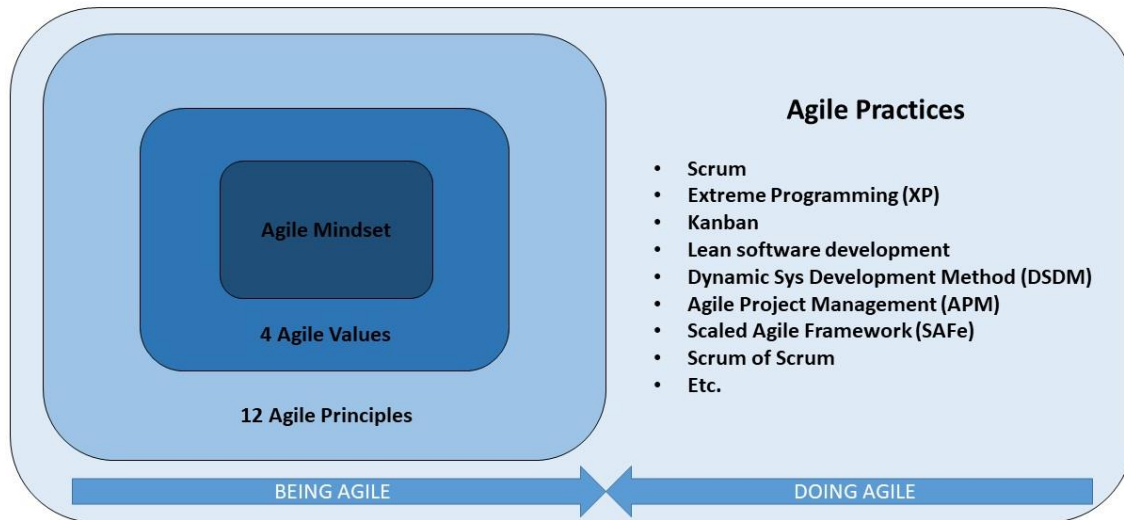
Figure 5. Agile project management framework

To standardize the agile processes, the agile manifesto is developed by the Agile Alliance consisting of four values and fifteen principles (Beck, et al., 2001). The agile values are as follows:

> *"We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:*
>
> *Individuals and interactions over processes and tools.*
> *Working software over comprehensive documentation.*
> *Customer collaboration over contract negotiation.*
> *Responding to change over following a plan.*
>
> *That is, while there is value in the items on the right, we value the items on the left more."*

In addition, there has been a latest version of Agile Manifesto that was introduced by Beck (2010) at Startup Lessons Learned Conference 2010. This agile manifesto is more suitable with the current condition of startup (Denning, 2015). The latest Agile Manifesto as follows:

> ***Team vision and discipline****/over individuals and interactions/over processes and tools.*
> ***Validated learning****/over working software /over comprehensive documentation.*
> ***Customer discovery****/over customer collaboration/over contract negotiation.*
> ***Initiating change****/over responding to change/over following a plan*

Along with the manifesto, 12 principles were also established, which are as follows:

(1)   Our highest priority is to satisfy the customer through the early and continuous delivery of valuable software.

(2)   Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

(3)   Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

(4)   Business people and developers must work together daily throughout the project.

(5)   Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

(6)   The most efficient and effective method of conveying information to and within a development team is a face-to-face conversation.

(7)   Working software is the primary measure of progress.

(8)   Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

(9)   Continuous attention to technical excellence and good design enhances agility.

(10) Simplicity--the art of maximizing the amount of work not done--is essential.

(11) The best architectures, requirements, and designs emerge from self-organizing teams.

(12) At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

2.2.2.   Agile project management practices

The agile manifesto has become the starting point of agile project management practices (Abrahamsson, et al., 2002). Agile project management practices can be divided into 2 main categories: lightweight approaches and fuller approaches. The lightweight approaches imply a simple method that is easy to follow due to having few rules and being inherently flexible enough to be tailored to each project, depending on the characteristics of each (Carroll, 2002; Wells, 2009). By contrast, the fuller approaches imply incorporating the agile values and principles into a traditional organization, so that the top management within the organization can be more agile in running their projects without forgetting the hierarchy in that organization. Hence, Table 2 will list some agile project management practices.

AGILE PRACTICES	
Lightweight approaches	**Fuller approaches**
Scrum	Dynamic Sys Development Method (DSDM)
Lean software development	Agile Project Management (AgilePM)
Kanban (process + method)	Agile Unified Process (AUP)
Extreme Programming (XP)	Open Unified Process (OpenUP)
Continuous Integration (CI)	Scaled Agile Framework (SAFe)
Feature Driven Development (FDD)	Disciplined Agile Delivery (DAD)
Crystal Family	Scrum-of-Scrums
…	…

In this study where startups are the object, the researcher only emphasizes the lightweight approaches due to startup characteristics such as flat organization, small team and rapidly evolving. Those characteristics are fit with the lightweight approaches as an example the scrum team is a small team consists of 5-11 people. Thus, the researcher has decided to provide a comparison of general features from several known and common lightweight agile practices *i.e.* extreme programming, scrum, crystal family, and feature-driven development in Table 3.

Table 3. General features of agile project management practices, modified from (Sommerville, 2011)

Practices	Team size	Iteration length	Key points	Special features
Extreme programming	2-10	2 weeks	Customer-driven development, small teams, daily builds.	Refactoring – the ongoing redesign of the system to improve its performance and responsiveness to change.
Scrum	5-11	4 weeks	Independent, small, self-organizing development teams, 30-day release cycles.	Enforce a paradigm shift from the "defined and repeatable" to the "new product development view of Scrum."
Crystal	variable	< 4 months	The family of Methods. Each has the same underlying core values and principles, techniques, roles, tools, and standards vary.	Method design principle. Ability to select the most suitable method based on project size and criticality.
Feature Driven Development	variable	< 2 weeks	Five step process, object-oriented component (*i.e.* feature) based development. Very short iterations	Method simplicity, design and implement the system by features, object modeling

Extreme Programming (XP) is a lightweight process focusing on addressing constraints in software development (Beck & Andres, 2005). There is "a set of rules to follow that guarantee success" in which teams work in a strict priority order. The features that will be developed are prioritized by the customer and the team is required to work on that order. XP allows any changes within their iterations before the developer has not started to develop a particular feature, thus a new feature with equivalent size can be swapped into the XP team's iteration in exchange for the un-started feature (Cohn, 2007)

Scrum teams typically work in iterations called sprints for two to four weeks, and XP teams work in shorter periods of one to two weeks long (Cohn, 2007). At the beginning, the development teams meet with the client or product owner to discuss new requirements. The scrum master is the facilitator for an agile development team who manages the process for how information is exchanged. In the sprint planning, the new requirements will be divided into several tasks which will be done in the upcoming sprint. During the sprint, the team will update their progress through daily meetings. When the sprint ends, the team will deliver a potentially shippable product increment. Scrum is suited for projects with rapidly changing or highly emergent requirements such as Web projects or product development for new markets (Cohn & Ford, 2003).

Crystal was developed by Alistair Cockburn in the mid-1990s. The practice introduced color codes to classify the risk of human life. For example, the Crystal Sapphire is a project that involves risk to human life, while Crystal Clear is used when projects do not have such risks. Crystal focuses on six primary aspects: people, interaction, community, communication, skills, and talents. This method is flexible and avoids rigid processes as it focuses on human-powered or people-centric (Santos, 2017).

Feature-Driven Development (FDD) was introduced by Jeff De Luca in 1997. It aims to deliver a tangible, working software repeatedly in a timely manner. This practice was built around software engineering's best practices such as domain object modeling, developing by feature, and code ownership. The process consists of five basic activities: the development of an overall model, the building of a feature list, the planning by feature, the designing by feature, and the building by feature. Furthermore, the last process includes short iterative processes to develop the feature which takes no longer than two weeks (Santos, 2017).

2.2.3. Agile software development

Identifying success factors for projects in industries like software development can be defined as agile if it follows some aspects, *i.e.* incremental (small software releases, with rapid cycles), cooperative (customer and developers working constantly together with close communication), straightforward (the method itself is easy to learn and to modify, well documented) and adaptive (able to make last moment changes) (Abrahamsson, et al., 2002). Jurca (2014) also emphasized that putting the customer needs at the center of the project with a focus on involvement and satisfying the customer is mandatory for agile software development (Jurca, et al., 2014). Hence, agile software development methods are suitable for small teams developing small- or medium-sized products or services (Dybå & Dingsøyr, 2008; Sommerville, 2011). Furthermore, agile can be used to solve complex software problem (Mishra & Mishra, 2011). As for the startups, agile is needed in order to scale up the business (Kelly, 2017)

The process management in software development startups represents all the engineering activities used to manage product development. The agile approaches have been considered the most viable process because startups need the flexibility to accommodate frequent changes. The agile methodologies embrace changes, allowing development to adapt the business strategy (Coleman & O'Connor, 2008).

Startups encourage fast-release products with an iterative and incremental approach which shortens the lead time from idea conception to production with fast deployment (Giardino, et al., 2014).

Startups are fast-moving and rapidly-evolving, and thus their final goals often change. The team members also have multiple roles to give quick responses to all challenges at the moment. Those conditions may lead startups under unstable environment and might end up with chaos, so they need an approach to keep organized. In connection with these characteristics, agile methodologies offer a solution for startups as agile has been designed to keep up and facilitate their rapid pace of development. It also focuses on planning and achieving short term goals, getting fast results and feedback, and also adapting strategies accordingly (Pentalog, 2017). Agile methodologies are also able to adapt faster, delivering little and often, testing the environment and remaining-focused (Agile Business Consortium, 2018). Moreover, Agile can become the key to startup success because it can deliver faster with a good prioritization. In example, Spotify broke their employees into squads and each squad has an agile mentor that decides the agile practices that will be used. Implementing agile in Spotify led them to stay competitive with a Google Play Music and Apple Music (Gontovnikas, 2016)

2.2.4. The implementation of agile software development project in Indonesia

There are limited studies looking at the implementation of agile methodologies in software development project in Indonesia. The earliest journal was recognized from Kusumasari et al. (2011) that introduced a collaboration model adopting agile methodology and combined a few adjustments to the conditions that may occur in environment business. The collaboration suggested a combination of face-to-face and collaboration software tools, with the selection of software tools depending on the needs of software development and mystery of the tool. The combination occurs in every phase of the development process between customers, developers, and stakeholders. But this study lacks agile implementation and principles within the real case.

Al-Kautsar et al. (2013) investigated the adoption of agile methodology in Indonesia based on perceptions among software practitioners. It shows that 28 out of 32 respondents have heard of agile methodologies, while 21 of them had actually used agile at least once. Furthermore, a few number of new software practitioners were aware of agile software development methods. It supports the first challenge, (1) lack of exposure to agile methods in Indonesian academic institution being a possible reason. On the other hand, (2) the awareness of agile methods is still in its infancy due to a lack of variety agile practices around Scrum and XP. In addition, (3) a lack of awareness also happened among customers with limited knowledge in the agile process.

Another study from Salleh, et al. (2014) introduced the challenges in implementing agile methodologies could be classified into internal and external challenges. The internal challenges mean the challenges arise from the teams, such as (1) sharing the same vision and mission between team members towards the project, (2) the programmers tend to wait for assigned work rather than self-assigning, (3) a lack of trust and confidentiality to implement agile methodology as they previously used traditional method, (3) team members's limited knowledge of agile methodologies. The involvement of customers and stakeholders in the agile process may become some external challenges. It has been found that the external challenges are as follows: (1) limited cooperation and collaboration from the customer, (2) the customers having limited knowledge in agile methodologies, (3) extensive resources due to a large population in Indonesia.

2.2.5. Agile software development project success factors model

Nowadays, software development has not been consistently successful, often resulting in delayed, failed, abandoned, and rejected software projects. Even software projects that are already implemented

may need expensive on-going maintenance and corrective releases or service packs. Considering this, the biggest challenge for software development projects is an improvement to avoid problems of waste and inefficiency. There has been a recent emergence of a new class of software development process called 'Agile' methods, which operate rather differently than traditional methods.

The agile software development project success factors were studied by Chow and Cao (2008) which conducted a survey study using a quantitative approach based on 109 agile software development projects from 25 countries around the world. The research began with a preliminary list of potential critical success factors of agile projects that were identified from the literature, then a reliability analysis and factor analysis were used to consolidate the preliminary list into a final set of 12 possible critical success factors for each of the four project success attributes, including Quality, Scope, Time, and Cost.

The literature review has been conducted on factors impacting the success in agile software development, covering the failure factors, success factors, and success attributes from previous journals and publications. Table 4 shows literature used in success factors study.

Table 4. Literature used in success factors study

Indicator	Literature
Failure factors	Reel (1999), Cohn and Ford (2003), Larman (2004), Boehm and Turner (2005), Nerur et al. (2005), Chow and Cao (2008)
Success factors	Highsmith (2002), Boehm and Turner (2003), Reifer (2003), Lindvall et al. (2004), Schatz and Abdelshafi (2005), Karlstrom and Runerson (2005), Augustine et al. (2005), Ceschi et al. (2005), Ambler (2006), Chow and Cao (2008)
Success attributes	Chow and Cao (2008)

2.2.5.1 Failure factors

Studying the failure factors can contribute to the understanding of how to avoid certain serious pitfalls that are critical to the success of a project. Failure or problem research is typically based on "lessons learned" from certain types of projects, but they are mostly similar and can be generalized. The failure consists of 19 factors which were grouped into four dimensions: Organizational, People, Process, and Technical. The table 5 shows failure factors based on Chow and Cao (2008).

Table 5. Failure factors based on Chow and Cao (2008)

Dimensions	Factor
Organizational	Lack of executive sponsorship
	Lack of management commitment
	Organizational culture too traditional
	Organizational culture too practical
	Organizational size too large
	Lack of agile logistical arrangements
People	Lack of necessary skill-set
	Lack of project management competence
	Lack of teamwork
	Resistance from groups or individuals
	Bad customer relationship
Process	Ill-defined project scope
	Ill-defined requirements
	Ill-defined planning
	Lack of agile progress tracking mechanism
	Lack of customer presence
	Ill-defined customer role

Technical	Lack of complete set of correct agile practices
	The inappropriateness of technology and tools

Additionally, the failure factors that have been summarized from Chow & Cao (2008), there is also another research that introduced 10 signs in software development project failures, with 7 of them fully determined before a design is developed or a line of code is written. The 10 signs of project failures are as follows: (1) project managers do not understand the user's needs, (2) the project is not clearly defined, (3) the project changes poorly, (4) the chosen technology changes, (5) the business needs change, (6) the deadlines are unrealistic, (7) the users are resistant, (8) losing sponsorship, (9) the project has limited people with appropriate skills, (10) managers ignore the best practices and lessons learned (Field, 1997).

2.2.5.2 Success factors

The literature reviews mostly discussed case studies or meta-data or compilations and observations of agile project and practices. They also focused on agile implementations in large organizations or scaling of agile methods to large projects using a wide range of success factors. The success factors have been classified into five categories: Organizational, People, Process, Technical, and Project (see table 6).

Table 6. Success factors in agile software development project based on Chow and Cao (2008)

Dimensions	Factor
Organizational	Strong executive support
	Committed sponsor or manager
	Cooperative organizational culture instead of hierarchal
	Oral culture placing high-value face-to-face communication
	Organizations where the agile methodology is universally accepted
	Collocation of the whole team
	Facility with the proper agile-style work environment
	Reward system appropriate for agile
People	Team members with high competence and expertise
	Team members with great motivation
	Managers knowledgeable in the agile process
	Managers who have light-touch or adaptive management style
	Coherent, self-organizing teamwork
	Good customer relationship
Process	Following agile-oriented requirement management process
	Following agile-oriented project management process
	Following agile-oriented configuration management process
	Strong communication focus with daily face-to-face meetings
	Customer having full authority
	Honoring regular working schedule, no overtime
	Strong customer commitment and presence
Technical	Well-defined coding standards up front
	Pursuing simple design
	Rigorous refactoring activities
	The right amount of documentation
	Regular delivery of software
	Delivering most important feature first
	Correct integrating testing
	Appropriate technical training to team
Project	Project nature being non-life-critical
	Project type being of variable scope with the emergent requirement

	Projects with dynamic, accelerated schedule
	Projects with a small team
	Projects with no multiple independent teams
	Projects with up-front cost evaluation done
	Projects with up-front risk analysis done

Chow and Cao (2008) defined the overall perception of success in the particular project, suggesting Quality includes delivering a good working product, Scope covers meeting all the requirements from customers, Timelines in delivering on time, and Cost within estimated budget and effort.

The quantitative methods with coefficient alpha have been used for calculating internal consistency reliability. It is a direct function of both number of items and their magnitude of inter-correlation, and is the lower boundary to the test variance attributable to common factors among the items within each variable. From this analysis, Chow and Cao (2008) introduced a research model that can be used as a framework for identifying the success factors and success attributes in the agile software development project. Using Google Scholar, we can see that the model has been used in 717 publications around the world since its publication in 2008. The success factors can be divided into 5 main factors with 12 detailed factors. The detailed success factors along with the research model can be seen in figure 6.
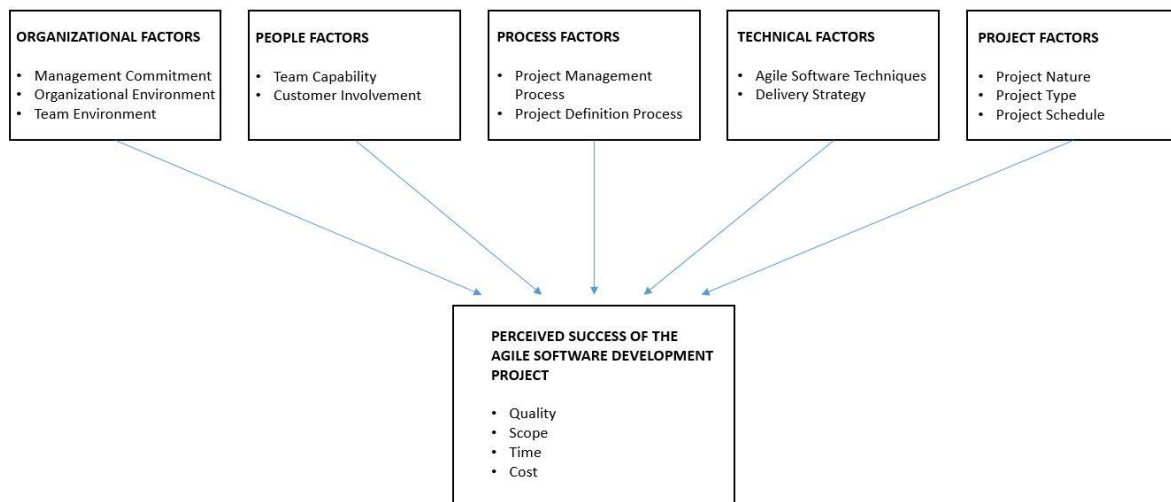
Figure 6. Research model introduced by Chow and Cao (2008)

The 5 main success factors are Organizational, People, Process, Technical, and Project factors. Furthermore, the 12 detailed success factors are (1) strong management commitment, (2) agile-friendly organizational environment, (3) agile-friendly team environment, (4) high-caliber team capability, (5) strong customer involvement, (6) agile-style project management process, (7) methodical project definition process, (8) agile-style software engineering techniques, (9) correct delivery strategy, (10) non-life-critical project nature, (11) variable-scope project type, and (12) dynamic, accelerated project schedule.

Ahimbisibwe et al (2017) proposed Top Management Support (TMS) to be the primary success factor for software developments, which assumes that TMS drives and influences the organizational factors. The top-level management commitment significantly and positively influences project performances (Jung, et al., 2008). Another study also mentions that TMS highly contributes to the improvement in agile software development process (Wan & Wang, 2010). Aside from commitment in top-level management, project team commitment (PTC) also affects the project success. PTC is the willingness of a team to devote energy and loyalty to a project in effective, continuance, and normative forms

(Meyer and Allen, 1997). Furthermore, the team members with great motivation positively influence the success of the agile software development projects (Chow and Cao, 2008).

The people factors in terms of team members are related to an internal project communication (IPC). IPC is defined as the practices that increase information exchange and cohesion among development team members. It also enhances the collaboration and sharing information between team members that will reduce the amount of team conflict and keeps the team stable (Ahimbisibwe et al, 2017). Another people factor is customer involvement, which becomes a characteristic of agile software development. The customer is encouraged to be involved in feedback and suggestion for the incremental changes of the product's development (Jun et al, 2011).

Technical complexity (TC) and project uncertainty are frequently regarded as independent, but the use of unfamiliar technologies can lead to software problems that reduce the performance of the software product and delay the project (Nidumolu's, 1996; Jun et al, 2011). Agile methodologies have a quick response to change and uncertainty in requirements, to reduce the cost of change throughout the project (Cockburn and Highsmith, 2001).

Based on Chow and Cao (2008), the critical success factors are found to be (a) a correct delivery strategy, (b) a proper practice of agile software engineering techniques, and (c) a high-caliber team. The other three factors, (a) a good agile project management process, (b) an agile-friendly team environment, and (c) a strong customer involvement, are factors that could be critical in certain success dimensions. They also mentioned that *"It may be worthwhile to repeat such a study again in five and ten years to see whether any new factors may emerge or current key success factors become no longer critical"*. Therefore, it becomes a reason for a later study that was conducted by Stankovic et al. (2013) in agile software development in former Yugoslavia IT companies. Another study was also conducted by Safwan et al. (2013) to provide a testing method in the same model for agile software development in Sri Lanka.

Furthermore, the study conducted by Stankovic et al (2013) introduced three more factors that could potentially be considered as critical success factors in terms of timeliness and cost, compared to the previous study from Chow & Cao (2008). Those three factors are: project definition process, project nature, and project schedule. However, all factors from the previous study cannot be considered as critical success factors in the Yugoslavia IT companies. They suggested that strong executive support and project type have no influence on the success of agile projects, while organization environment is not a significant factor to agile project success.

Safwan et al (2013) identified a low perception from agile users towards methodologies and that there is a difficulty of getting everyone in the team to take responsibilities. The people need to see the proven success stories from agile implementation to increase their confidence in trying agile methods. The challenges mostly occur from organizations having a hierarchical approach, where the management is expected to set the datelines and control the process. Other challenges found include lack of documentation, organizational aspects, involvement, knowledge and culture are based in or related to the people factor, where a mindset change is needed to overcome those challenges.

In Indonesia, there has not been a formal research investigating the success factors in agile software development projects within Indonesian startups. This study is thus important as Indonesian startups have contributed 7.44% to the total GDP in 2016 which then places Indonesia as the third ranked country worldwide whose economy receives significant contributions from startups (Badan Ekonomi Kreatif, 2018). Moreover, Pratomo (2016) stated that 90% of Indonesian startups failed, leaving a small number of startups that have survived today. It emphasizes that reducing the failure rate of Indonesian startups is essential to gain a higher value to GDP contribution. This study may lay down some foundation and give indicators on how they can improve their performance through understanding the perceived success factors in agile software development projects in Indonesia. This study also aims to

examine whether the critical success factors are different or similar to what was found by Chow and Cao (2008).

2.2.6. Previous case studies in agile software development in Sri Lankan companies

An empirical study has been conducted by Safwan et al. (2013) investigating the perceptions of using agile methodologies by software practitioners in Sri Lanka. The study departs from an assumption that agile software development perceptions in Sri Lankan software industries are still low due to its recent introduction in Asia, while it has been used broadly in Western countries.

Nowadays, software industries have become the fastest growing in the business sector that leads them to move industries from high wage countries to lower wage countries, especially in Asia (Cockburn and Highsmith, 2001). Supporting this, Sri Lanka has the potential to improve enabling factors and make the software industry the main foreign exchange earner of the country within a relatively short period (Senanyake, 2011). Unfortunately, in reality Sri Lanka lacks the usage of software processes and problems in delivering a proper quality product, as well as lacking a clear methodology that shows how requirements can be obtained.

The research used semi-structured interviews with Sri Lankan IT companies called Zone24x7 and Sri Lankan Airlines IT Department. There were 61 software practitioners interviewed, including developers, testers, system analysts, business analysts, and project management involved in the agile project. This interview as part of a qualitative study helps in understanding the respondent's behavior and identifying the unknown and never-before explored scenarios. It will also help describe the implementation of agile methodologies.

In summary, the perceptions were identified from the respondent's awareness of the way they introduced the method and the challenges they are facing. It was found that a low perception from agile users has led to the difficulty of getting everyone on the team to take responsibilities. Therefore, people need to see the proven success stories from agile methodologies to increase their confidence to apply agile methods in their team. The biggest challenges were found from organizations with traditional approaches where the management is expected to set datelines and control processes. Another challenge is a lack of documentation, organizational aspects, involvement, knowledge, and culture, all based in or related to the people factor. But in reality, software development companies show an increasing interest towards agile methodologies. Their interest will be higher as software engineering and software process provide knowledge and references on agile methods to the country and nearby regions.

The agile software development study in Sri Lanka will be used as a comparison with the current study, which investigates the similarities and differences in the implementation of agile methodologies in software development project between two Asian developing countries, Sri Lanka and Indonesia.

3. Methodology

3.1. Introduction and overview

In order to answer the research questions that focus on exploring the perceived success factors within the agile software development project in Indonesian startups, a qualitative approach is chosen. A qualitative approach is needed to study the actors, which are practitioners of software development in Indonesian startups. According to Bryman (2011), a qualitative approach is used in order to get a better understanding of a phenomenon that heavily focuses on actor interactions. The choice of a qualitative approach was made because the researcher expects the unexpected result and flexible approach to solving the research questions.

Also, the exploratory study is selected for this study. An exploratory study aims to examine an issue that has not been studied more clearly. In this regard, agile software development in Indonesian startups have rarely been researched. Exploratory studies often rely on reviewing available literature and formal qualitative research through in-depth interviews.

3.2. Research design

The research design for this qualitative research is a case study. There is more than one startup that becomes the object, so the research does case study for each 20 startups. The case study is needed in order to explore a real life, multiple bounded system over time, through detailed data collection for a specific topic (Cresswell, 2013). Figure 7 shows research design

Figure 7. Research design

The whole process of conducting this research is shown in Figure 5. This research starts with a problem definition process. The researcher attempts to find the gap in previous research and also explore a phenomenon that happens in a specific topic. The second step is building a theoretical framework. The theoretical framework is written in the literature review part consisting of two main resources, which are startup and agile project development. Following the next step is building the research methodology. Every aspect of the research is constructed and completed with a guide in order to achieve accurate, correct, trustworthy and objective research. The fourth step is data collection. The data collection method for this research is a semi-structured interview. All the data gathered from the interview are analyzed later on. The last step is to make a conclusion that answers the research questions.

3.3. Literature review

The literature review was conducted to gain an initial impression of a specific topic that is intended to be better understood through research. Mongan-Rallis (2014) emphasizes that a literature review is not a summary of the professional literature, but rather is only focused on a specific topic that the current research aims to answer, including a critical analysis of the relationship among different works and structuring it into the construction of the literature review. There are several steps required in the literature review. The first step is identifying the literature that needs to be reviewed. Regarding this first step, the researcher identified two main topics, startups and agile project management, as relevant literature. All of the literature is provided by Uppsala University Library, Google Scholar, and trusted websites. Some of the resources, especially from the Indonesian government, are written in Indonesian, so the researcher self-translate in a proper way.

After all the relevant literature has been acquired, the researcher needs to do an analysis which consists providing an overview of the articles (key terms, concepts, research methods, and findings), and grouping those articles into categories. The next step is summarizing the literature based on the research needs. After summarizing, the researcher needs to synthesize the literature and put it into a coherent essay.

3.4. Research sample

There are two main criteria that need to be followed to choose the participants for this study *i.e.* startups and startup members. The first criterion is startup characteristics. The startups chosen for this study should have the characteristics of: (1) operating in Indonesia; (2) focusing on software development as their main product; (3) have had an 'efficiency' phase of startup lifecycle, and have been in business for approximately 8 months or more. The 'efficiency' phase is chosen because startup has a validated market and have core members, in 'efficiency' phase, startup refine its business process. If all the characteristics are met, then the second criterion is considered. The second criterion is startup members. The startup members should have the characteristic of being a part of software development projects on the technical side. All the characteristics are defined as such in order to give results applicable to answering the research questions of this study.

The total population of software development projects based on Bekraf is 3,833 startups. The researcher decided 20 samples are sufficient for this research to reach saturated samples. This notion is supported by Mason (2010), who argued that 20 sample sizes are most common for qualitative research. Moreover, Bertaux (1981) claimed that fifteen is the smallest acceptable sample. The sampling technique is nonrandom sampling, namely convenience sampling. All the participants are selected because of their convenient accessibility and proximity to the researcher.

3.5. Data collection

The data collection method used in this study is a semi-structured interview. A semi-structured interview utilizes a list of questions on fairly specific topics to be covered, which is further referred to as an interview guide. This type of interview leads the interviewee to have enough room to answer each question with their style and makes the interview more casual (Bryman & Bell, 2011, p. 205). While interviewing, the following questions (see Appendix 1) refers to the interviewee's answers, so it may not follow the interview guide exactly. This flexibility is needed in order to check the consistency of the answers from the interviewee. Additionally, by adding follow-up questions, the researcher can also

check whether the interviewee really knows what he/she understand. There are two versions of interview guides, based on whether the interviewee adopts agile software development or not. If the interviewee adopts agile software development, the interview guide used is shown in Table 7. The interview questions for research question 3 follow Chow and Cao (2008) because it provides a set of guideline of which critical success factors that is relevant with Agile software development project and to examine whether the critical success factors are different or similar to what was found. On the other hand, if the interviewee does not adopt agile software development, the interview guide used is shown in Table 8.

Table 7. Interview guide for agile software development

RQ 1	*What is the adoption of agile software development like within a project in Indonesian startups?*
	Do you use agile software development project?
	How long have you used agile software development?
	Why do you use agile software development?
	What are the agile software development practices that you use?
	How does agile software development work in your project?
RQ 2	*What are the challenges of implementing agile software development within a project in Indonesian Startup?*
	What are the challenges that you face when implementing an agile software development project?
	How do you overcome the challenges?
RQ 3	*What are the perceived success factors of agile software development projects in Indonesian startups?*
	a. Do you think **Management Commitment** is a success factor? Why? b. Do you think **Organizational Environment** is a success factor? Why? c. Do you think **Team Environment** is a success factor? Why? d. Do you think **Team Capability** is a success factor? Why? e. Do you think **Customer Involvement** is a success factor? Why? f. Do you think **Project Management Process** is a success factor? Why? g. Do you think **Project Definition Process** is a success factor? Why? h. Do you think **Agile Software Techniques** is a success factor? Why? i. Do you think **Delivery Strategy** is a success factor? Why? j. Do you think **Project Nature** is a success factor? Why? k. Do you think **Project Type** is a success factor? Why? l. Do you think **Project Schedule** is a success factor? Why?
	Are there any other factors that you think have become a success factor of agile software development project?

Table 8. Interview guide for other software development

RQ 1	*What is the adoption of agile software development like within a project in Indonesian startups?*
	What kind of project management in software development project do you use?
	Why do you not use agile software development?
	Why do you use that particular project management?
	Are you interested in using agile software development in the future?

The participants for the interview were contacted using email and online messaging. First of all, the researcher spread information of the needs of the interviewee in several relevant groups, such as academic group (LPDP), startup group (NextDev Telkomsel), tech enthusiast group (Google Student Ambassador, Top Contributor Group) *etc*. to gather people that fit the criteria. Then, people that are deemed to fit the criteria and interested to be interviewed answers on that group. After that, the researcher checks if the interviewee candidate fit the criteria. The researcher personally contacts them by online messaging and email to inform them that they have been chosen. Researcher only needs 20 people of the interviewee.

Following that, the researcher sends detailed information regarding this study and inform them of the ethical conduct by email. If they consent, the researcher setups an interview scheduled within a week based on the availability of the interviewee. The researcher also notes that there is a time difference between Sweden and Indonesia, so the schedule is based on an agreement between the interviewee and the researcher.

The interview process takes around 30-45 minutes for each interview. Because of the geographical difference, the interview uses video calls like Skype or Google Hangout. If the internet connection is not supported, especially on the interviewee's side, the backup plan is an interview by phone. The researcher takes a voice record for every interview. This voice record then be transcribed, and then sent back to the interviewee to check whether the researcher's interpretation matches what he/she said.

3.6. Data analysis

After all of the interview data has been transcribed and accepted by the interviewee, the data is analyzed. The method that is used for data analysis is thematic content analysis. It aims to find the common pattern across a data set of interview transcript (Anderson, 2007). There are several steps in this method, which are: (1) getting familiar with the data; the researcher should read and re-read all of the interview transcripts to get a whole understanding of all available data, (2) searching for the themes based on the research model presented in the literature review; the researcher needs to structure and categorize all the information into different themes. The structuring and categorizing of information mainly come from keywords as well as the meaning of the sentence in the interview transcripts. See Appendix 3. (3) Fitting the data into themes. After categorizing, the data is fitted into themes. (4) Interpretation. The researcher should seek a deep understanding from the categorized data based on themes into interpretation. This interpretation is compared with the literature review and used to form a conclusion in the end.

3.7. Ethical consideration

Ethics was taken into consideration because the involvement of people may result into a conflict in the data collection process. The researcher should keep his integrity by following a set of moral principles and rules. There are several major principles associated with ethical conduct, such as do no harm, privacy and anonymity, confidentiality, informed consent, inappropriate behavior, data interpretation, and data ownership (Bryman & Bell, 2011). All principles of ethical conduct used in this study are listed in the following: (1) **Privacy and anonymity**. The researcher should guarantee privacy and anonymity by removing all identifying information from the interview process. (2) **Confidentiality**. The researcher has a responsibility to keep the information confidential. All the collected data is saved in a secure place within the researcher's computer and backed up in a cloud system that only the researcher that has the password to access. (3) **Informed consent**. It is the researcher's responsibility to make sure the interviewees are informed with complete information. The participation in this study is a voluntary act. The interviewee is also able to withdraw from the study at any time without feeling penalized. All data regarding the withdrawing interviewee is immediately removed (4) **Data interpretation**. The researcher should be expected to analyze data in a manner that avoids misinterpretation or fraudulent analysis.

3.8. Bias

The qualitative approach that is used in this research affects subjectivity in a large way. The researcher's subjectivity comes from personal interaction in the interview process and the researcher becomes the instrument of both data collection and data analysis (Rajendran, 2001). The researcher and the research situation are never neutral, and thus bias is inevitable.

Even though it is easy to label the potential sources of bias, the researcher needs to make a guideline of research in order to minimize biases. Guidelines also keep this study honest and enhance its fairness. The researcher formulates some guidelines in order to deal with bias, as follows:

1. Selection of participants
    Even though the convenience sampling method is chosen in this study, the researcher set up a guideline to choose the participants in the interview process. This guideline is explained in subsection 3.4: Research Sample.
2. Data collection
    In the interview process, the interview result can be biased due to a different interpretation between what the interviewer perceived and what interviewee said. In order to tackle that bias, the interview result is sent to the interviewee and be checked by them before analyzed in the data analysis phase. Additionally, the researcher also makes field notes in the interview process. The field notes involve aspects of what happens (where, what, who) and the researcher's own reactions to what happens (feelings, interpretations, and associations). These field notes are used to reflect on the interview result and to reduce the element of bias.
3. Data analysis
    After ending up with a qualitative data from the interview, the next thing the researcher does is following the guideline for data collection in subsection 3.6: Data Analysis. Worth mentioning is that the interview data should not be taken at face value. The literature review is constructed to provide a framework for analyzing the results of the interviews. The literature review leads to objectivity, rather than subjectivity, of the researcher's opinion.

Moreover, all the research strategy that has been defined requires the researcher's stance of neutrality with regards to pursuing the accurate, correct, trustworthy and objective research.

3.9. Assumptions

Making assumptions is needed in this study in order to make this study relevant and for controlling uncontrolled situations. Thus, Simon (2011) emphasizes that reasoning for the assumptions is needed in order to justify that the assumption is "probably" true. The assumptions, the reasons for making them and their justifications are shown in Table 9.

Table 9. List of assumptions made in this study

Assumptions	Reason	Justification
Information from an interviewee is the truth	The data is mainly based on an interview	The names of the participant are kept anonymous and researcher has already received his/her consent to keep the trustworthiness
The participant understands the interview questions	The data is mainly based on an interview	The researcher provides the question list beforehand and is also present in the interview process
The selected startups represent other startups within the same field	As a multiple case study, the result should gather 20 startups in order to represent the software development startup population	The startup's characteristics are predetermined in order to get similar cases respectfully
Data gathered from the interview is not influenced by the researcher's presence	The researcher is not intimidate and direct the interviewee strictly	The interview process is a semi-structured interview. The researcher is open to new and related questions that are deemed necessary to better understand the context
The chosen participants have the same level of experience	The researcher only chooses participants that are directly involved in software development projects on the technical side	To fit the purpose of understanding the perceived success factor in agile software development project within Indonesian startups, choosing the right participant is a must
Cultural background is not interfering	Both the interviewer and interviewee are Indonesian	The same nationality means a similar cultural background without considering ethnicity

3.10. Limitations and delimitations

The researcher is aware that there are limitations and delimitations in this study. Limitations can be seen as potential weaknesses and are out of the researcher's control, and needs to be dealt with in order to not affect the outcome of the study (Simon, 2012). The limitations of this study are listed below:

1. The overall time of conducting the study, approximately 5 months
   The researcher made a timetable in which to conduct this study with specific milestone dates, to ensure that this study can be finished in a specific period of time.
2. The researcher uses a convenience sampling method to choose20 samples, all of them located in Indonesia.
   The result of this study can only generalized to a specific population, which is software development startups in Indonesia. It cannot be generalized to all startups in Indonesia.
3. There is a language barrier between some of the participants and the researcher because the native language for the Indonesian people is Indonesian (*Bahasa Indonesia*) and the interview process is conducted in English
   The researcher already sent out the question list beforehand in English, so participants can ask questions if there are unclear questions. If the participants insist on using *Bahasa Indonesia*, the result of the interview process is translated by capable people and sent back to the participants to check whether they agree with the results of the interview.

On the other hand, the delimitations of this study define the limit of the scope and boundaries of this study (Simon, 2012). The delimitations are listed as follows:

- The chosen problem statement, which is exploring the perceived success factors of agile software projects.
- The geographic region chosen in this study is Indonesia.
- The chosen participants for the interviews come from software development startups, and are also part of software development projects.
- Only lightweight approaches of agile project management practices are studied.

4. Result and Analysis

4.1. Demographics

The selected sample was 58 individuals, of which 23 accepted to participate. After conducting the interviews, 3 participants were disqualified because one was not related, and the other two did not give meaningful feedback. At the end, there are 20 participants for interviews who are working in IT related Indonesian startups (see detailed interviewee profile in Appendix 2). Figure 8 below shows how the interviews were distributed by startup company type. It can be seen that there is a higher number of participants from e-commerce and software houses. Comparing this two with the data in the literature review, it can be seen that it is in line with startup trends from Daily Social (2016) and Eka (2017).
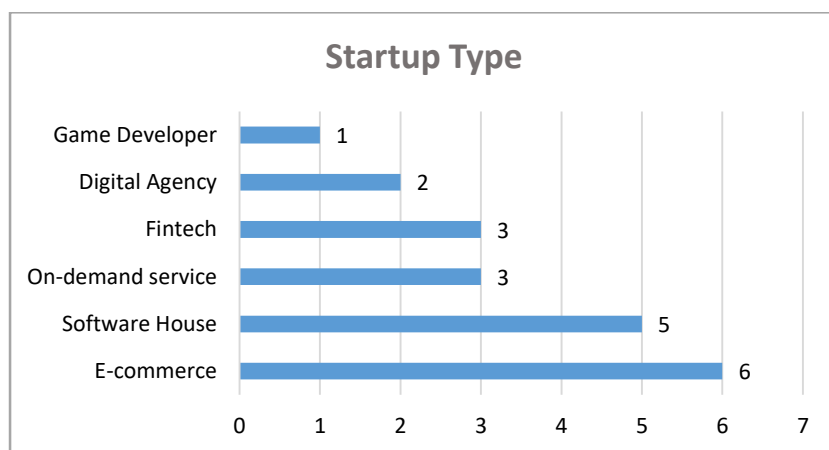
Figure 8. Startup types

Figure 9 shows the gender representativeness from interviewees, women representatives make up less than a quarter. This condition is slightly different with the Bekraf data in 2017, which mentioned that the proportion between women and men in startups is 1:3. Thus, it happened due to convenience sampling which lead to coincidence in the gender proportion. Therefore, the author believes there are no differences in their interview responses.
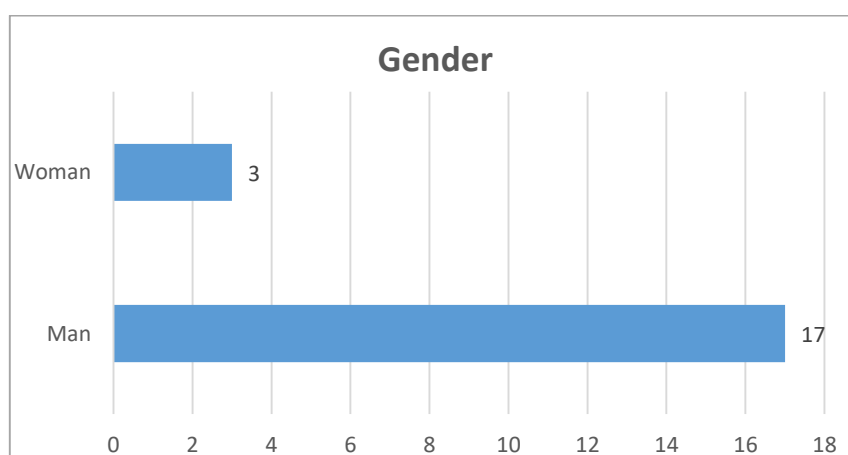
Figure 9. Gender profile
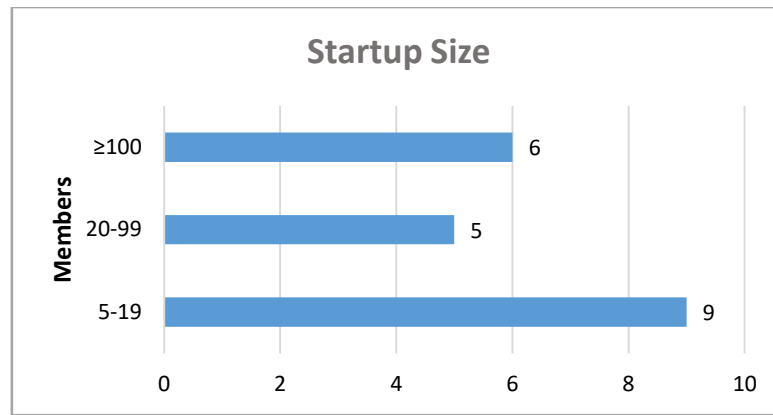
Figure 10. Startup sizes

Based on Figure 10, the largest startup size is 5-19 members, with 9 startups. On the other hand, startups that have ≥ 100 members and 20-99 members come respectively with 6 startups and 5 startups. This trend is slightly difference with the Bekraf data (2017) which showed the proportion of startup size in Indonesia as: 5-19 (70%), 20-99 (25%) and ≥100 (5%). The difference comes from the lack of startups over size of 5-19, which should be 14, and an excess of startup sizes of ≥ 100, which should be 1. Hence, this condition happened because of convenience sampling and sampling criteria. The convenience sampling prevented the researcher from fully controlling the startup size of interviewees, especially since the startup size was only known after the interview had been conducted. On the other hand, the sampling criteria which focused on startup criteria for interviewees are focused on startups that are at least in the "efficiency" phase, so there will be a big possibility from startups in the 'scale' phase that has members of at least 100. It is shown as an indication that not only startups in the 'efficiency' phase participated in this research, but startups in the 'scale' phase also participated.
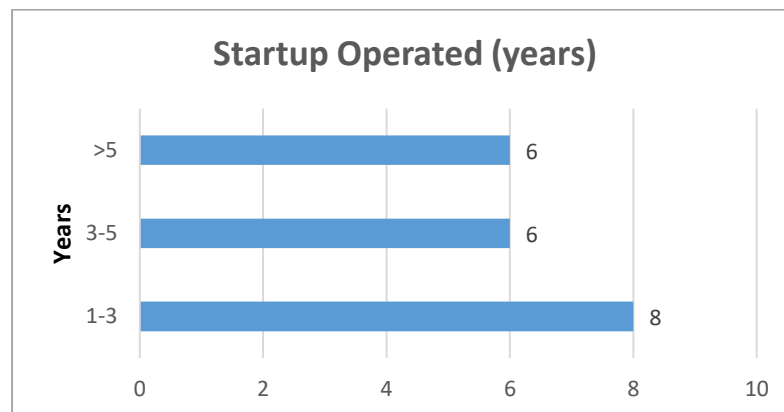
Figure 11. Startups operated (years)

The length of startup operations can be seen in Figure 11. There are 8 startups that have been operating within 1-3 years, followed by startups with 3-5 years, and then over 5 years, both categories at 6 startups. In accordance to Marmer's startup stage model (2012), the efficiency stage can be achieved within at least 8 months of the startup being established, and this result shows that most of the startups have already reached the efficiency stage, and some of them have already scaled up, especially startups that have been operating for more than 5 years. This finding strengthens the previous indication from startup sizes.

4.2. Adoption of agile software development in Indonesian startups

This subsection will address the first research question: *"What is the adoption of agile software development projects in Indonesian startups?"*

4.2.1. The implementation of agile methodologies

Regarding the implementation of agile methodologies in software development projects, most of the participants have sufficient knowledge of agile methodologies as they have been using it in their projects. Based on Figure 12, a majority of startups agreed that they are implementing agile methodologies, while the rest only used agile partially (5 startups) or not at all (4 startups). They also show that Scrum is a common agile practice that has been used widely in software development project. This finding is different to Al Kautsar et al. (2013), which found that 65% of software practitioners implemented agile software development methods. The difference can be explained due to an increasing exposure of agile methodology between 2013 and 2018, which increases the adoption of agile methodology, whether full or partial, to 80%.

Additionally, it was found that most of the interviewees who fully implement agile are developing their own products. They chose agile because of the high complexity of the product due to rapidly evolving requirements from the product's users. This discovery is supported by Mishra and Mishra (2011), as agile startups can be used for a complex project, and moreover, agile projects can be adaptive rather than predictive.
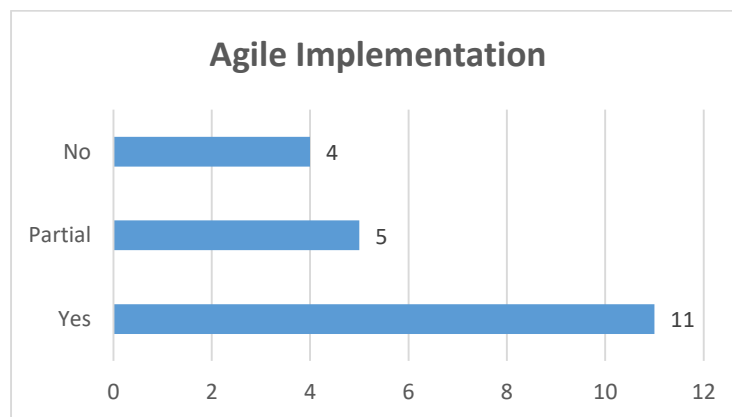
Figure 12. Agile Implementation

Based on Figure 13, there are 7 startups that have started using agile methodologies in their project for less than a year, while 6 startups have been using agile for 1-3 years and 3 startups have been using agile for more than 3 years. Most of the startups that have practiced agile for less than a year are startups that have been operating for less than three years. It means that those startups are still in the efficiency stage, which is a stage away from scaling up. In order to scale up their startups, they decided to use scrum. This finding is in line with Kelly (2017) about the reason why agile is needed by startups.
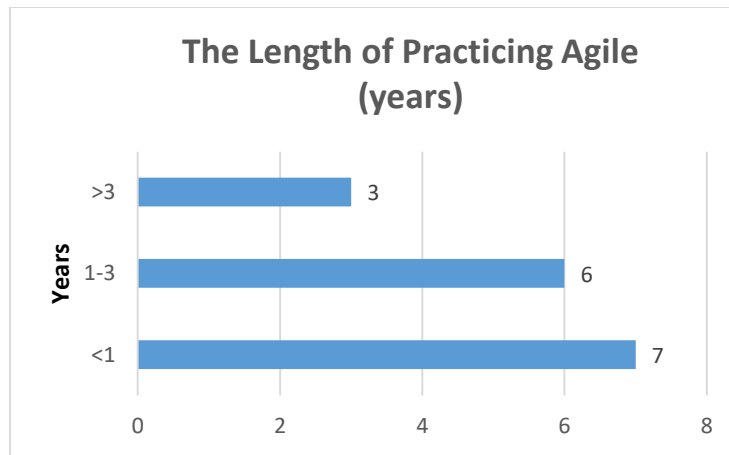
Figure 13. The length of practicing agile (years)

The application of agile methods within their startups is related to 'being' and 'doing' agile. According to the interviews, all of the agile users are already 'being' agile as they have an agile mindset due to their understanding of the values, principles, and practices. However, in terms of 'doing' agile, some of the participants are using several agile techniques which fit the working environment. These users are partially doing agile as they mostly skip the retrospective technique and finishing with a sprint review. One participant explained that they also skip daily standup meetings, which are replaced by weekly meetings.

> *"We have a sprint which runs for 2 weeks and we do not have any daily stand up meetings, there are only weekly meetings at least once a week to update the progress, issues, and to-do list"* -Participant 20.

Even though they chose to skip those two techniques, all of the agile users were found to use other techniques such as sprint, sprint planning, and sprint review. The partial agile users apply only parts of agile practices and techniques due to two main reasons that were mentioned by the participants. Firstly, they are new users who have only just started using agile a couple of months ago and are trying to implement all the techniques gradually, but they still need more time. The second is that it depends on the client's request; several clients have not used agile so they prefer a traditional way, such as the waterfall method.

4.2.2. The reason in implementing agile methodologies

Several reasons following the implementation of agile methodologies were mentioned during the interviews. There are 5 main reasons that have been explored, as follows: (1) Based on the mentor's preferences, from 4 interviewees, (2) The agile workflow has helped them develop the product efficiently, from 3 interviewees, (3) Startups have a simple organization that supports a quick response to the customer's request, from 2 interviewees, (4) They have sufficient number of team members and most of them are already familiar with agile methodologies, from 2 interviewees, (5) IT-related startups are a dynamic industry which needs a quick development that allows rapid changes during the project, from 2 interviewees.

Today, startups in Indonesia are rapidly growing due to a big support from the government. Based on Bekraf (2018), the government through Bekraf has several programs to support the startup ecosystem in the country. One of these programs is Startup Incubation Academy that provides mentoring and workshops for new startups, and each startup has their own mentor who is an expert in certain fields and business to discuss technology and business management that is suitable for them. Regarding the first reason, most of the mentors from that program suggested agile methodologies to be implemented in the startups. In addition to the government, there are also other organizations who have coaching

clinics for startups, such as *Gerakan Nasional 1000 Startup Digital* (Eng: *National Movement for 1000 Digital Startups*) which was initiated by Kibar (Kibar, 2018).

Abrahamsson, et al (2002) introduced incremental, cooperative, straightforward and adaptive as the four main aspects in agile methodologies. Those aspects were translated into agile workflow, which starts with sprint planning, sprint, daily standup meeting, sprint review, and retrospective. Most of the participants agreed that this method has significantly helped them in working efficiently.

> *"Methods in agile such as sprint planning, daily standup meeting, and breaking down the tasks into short periods has helped us develop the product."* –Participant 10

> *"We read a book called Management Modern Software Development Project that introduced us to agile methodology, which supports the faster development of technology. We learned that framework and methods in agile which has been described in the book is suitable for our software development project."* –Participant 11

One of the startup characteristics is created from a simple structure organization (Paternoster et al, 2014). It also became the reason in implementing agile methodologies as a simple organization will create quick decisions in a short period. The team members have rights to give their opinion in order to give a quick response to the customer's request changes. It also corresponds to Jurca (2014), who emphasized putting the customer needs at the center of the project, with a focus on involvement and satisfying the customer being mandatory for agile software development. It will be different with a company which has a complex bureaucracy which needs lots of permissions and takes longer time to respond to changes.

> *"Because our startup has a lean organization, it is possible for us to manage our team with the SCRUM method, and it also focuses on the business process and it has a clear objective and goals."*- Participant 17

Most of the participants mentioned that their startups consist of several small teams; team members are also already familiar with agile methodologies. Furthermore, the agile software development methods are suitable for small teams developing small- or medium-sized products or services (Dybå & Dingsøyr, 2008; Sommerville, 2011).

> *"At that time, all the early team members were familiar with SCRUM, and it is the most commonly used right now if you are a software-developing team in a startup. So if we recruit programmers, they are most likely familiar with SCRUM, so we do not need to teach them again about SCRUM."* –Participant 1

A large number of the participants came from startups who developed the software for their own purposes, such as e-commerce, on-demand service, fintech, and game developer. Those startups are developing their software incrementally through monthly updates, so they need a quick development to launch a new feature within a short term. Moreover, they explained that agile methodologies support quick development and allow rapid changes in the middle of the project.

> *"We use SCRUM because we need quick development as a startup. So, we think that we cannot use the waterfall method, which takes a longer time to create a product."* – Participant 2

4.2.3. The process of agile methodologies

In real practice, the agile workflow starts with a new requirement from the client, who is also the Product Owner. The new requirement will be discussed and prioritized through internal discussion between the development team and the Scrum Master. This discussion will produce a clear definition of the project

and a product backlog. After that, it continues to create a sprint planning in the beginning of the week in which the project will be broken down into several tasks or assignments and distributed to the team members. The timeline and deadline are also created in this stage, and the results of this discussion are also documented through a sprint backlog. After they have been assigned a certain task, they will work on it during the sprint. Each sprint mostly runs for 2 weeks depending on the complexity of the project (Sommerville, 2011). Every morning, there is a daily standup meeting to discuss the progress, issues, and to-do list for each day. The progress is also sometimes recorded using Scrum tools such as JIRA, Trello, and Phabricator software, while burndown charts and progress boards are also possible.

After the development has finished, the feature will be internally tested before it is delivered to the QA team to be tested further. All the bugs found during the testing phase will be recorded in the Scrum tools, so the developer will work on fixing bugs based on the report. After the bugs have been fixed, the feature is ready to be launched to the client. They will first test the feature and give some feedback directly, which helps the development team to give a quick response to the client's feedback. If the feature is going to be launched to the public, some of the participants said that they first launched it to selected users and wait for their feedback. If the users give positive feedback, then it will be launched to all users gradually.

During the process, the client could interrupt the process and request some changes. If it happens, the development team will decide the possibility of the changes to be finished in the current sprint, or if it should be moved to the next sprint. Several participants explained that their customer has been involved since the development teams created a mock-up version which the client has access to, to test the product mock-up. After all the stages are finished, the product owner, development team, and Scrum Master will evaluate the sprint through a sprint preview stage to make some improvements for the next sprint. The last stage in Scrum is retrospective, which involves deep analysis in evaluating the previous sprint, especially examining the strengths and weaknesses of the product and team itself.

> *"Yes, I am interested especially in SCRUM and I'm willing to learn about it first. As we can see, nowadays it becomes the trend in Indonesian startup industries. Also it is supported by lots of training in SCRUM itself for the Project Manager."* –Participant 21

4.2.4. Non-agile user

It was found that only four startups refused to use agile methodologies. All of these startups are applying traditional ways in the form of waterfall methodologies. They also described a few reasons why they still keep this method. Two participants said that they do not have enough team members to run agile methodologies, with their development team mostly consisting of fewer than four members, with even one of the participants is working alone without other members. This limited team membership negatively impacts the capability of the team, particularly when they need to handle lots of changes during the project which will lead to overpressure and workload for the members.

> *"I prefer to use waterfall to prevent any changes during the project, and also to prevent the software developer from getting bored and exhausted to fulfill the rapid changes. Because it is dangerous when the web developer is getting bored of their task, as it will make them create a non-proper product"* -Participant 4.

In some cases, it is dangerous as they will not deliver a product of proper quality. They are also working with a client-based project and most of their clients are used to the waterfall method in their company. Consequently, it has supported them to use the same methodologies in order to make it easier for them to fulfill the client's request. One of the participants also mentioned that the waterfall method allows customization of the process and makes it flexible without encouragement to stick with some tools or stages. Another reason that has been explored is a financial concern to getting the licenses for agile tools such as JIRA, Trello, and Phabricator software. This concern has brought them to the current condition in which they feel appropriate for waterfall practices.

The non-agile users are eventually interested to use agile methodologies in the future. But in order to make them ready to implement this method, they argue that a capable team – including member's

capabilities in agile and a sufficient number of members – becomes a critical aspect that must be first fulfilled. Also, they are willing to learn it more as agile has been a trending method, particularly in Indonesian startups. Many workshops and training were held to educate startup members about the importance of using agile methods within a software development project. By contrast, one participant still denied applying agile methodologies because the waterfall method could avoid any changes which lead to a higher efficiency of the project.

4.3. Challenges during the implementation of agile software development in Indonesian startups

This subsection will address the second research question *"What are the challenges of implementing agile software development projects in Indonesian startups?"* The solutions to overcome the challenges will also be explained in this section.

The majority of the participants show that agile methodologies have been commonly used within software development projects in Indonesian startups. Nevertheless, the implementation of this method also brings some challenges for the agile users. In this study, every case is unique with its own challenges and the way they to deal with each issue. Those challenges were examined and differentiated into two main categories: internal and external challenges.

Some of the challenges internally appeared from the startups, which cover techniques, team members, a less accurate timeline, and financial funding. The characteristic of Scrum that involves a Scrum master to manage the project becomes one of the internal challenges experienced by new startups. The absence of this role requires the agile users to be self-aware to keep their progress up to date and able to manage themselves based on the timeline. For a small startup, a team member who is working on the technical aspects can also be assigned as a Scrum master, but it further leads to inefficient work.

> *"Scrum master has become our greatest challenge so far as we don't have a person that has been assigned to that role. In consequence, the Head of Technical also play a role as a Scrum master, but sometimes this person gets too busy with the technical work to manage the team. Meanwhile, I'm also busy in business strategy so we actually need a SCRUM master."*-Participant 20.

The possibility of hiring a Scrum Master is an extra cost which further relates to a limited financial funding that a fledgling business has. In addition, budget concerns also happen when they want to use licensed agile tools such as JIRA and Confluence software. Therefore, the participants who experienced this particular challenge are treating these paid tools as an investment. Some of them are also choosing free tools that can be easily found online as alternatives.

Ineffective daily standup meetings also become an internal challenge which has been noticed by the agile users. These meetings should be a quick discussion about the progress, issues, and a to-do list of the day, but some of the team members discuss the technical issues in much greater detail which takes up a lot of time for the meeting.

> *"We always have a daily standup meeting, but it seems to not be effective because it's supposed to be a short meeting no longer than 5 minutes, but sometimes team members will discuss the details which will make the meeting last longer. I think all the detailed issues must be discussed separately."* –Participant 23.

In order to make the daily standup meetings more effective, the technical and details issues should be reduced and separated in a different discussion. When the team members realized that another member starts getting into the deeper issues, they must be warned so that certain topics could be discussed later.

Communication within the team is another internal challenge that has been demonstrated by two agile users. This challenge frequently appears when the team is working remotely, but they try to maintain communication through intensive online meetings. Another internal challenge is the definition of the task or assignment at the beginning of the project. The team leader needs to be concise and detailed

when explaining the task, because an unclear direction will lead to some delays in delivering the product, which also impacts the quality of the product.

> *"The challenge is how to describe a task in words so every team member understands and have the same mission and vision with the task. Sometimes I make a video which explains a certain task because it's really important to make sure that they really understand what they are doing."* –Participant 17.

All of the participants acknowledge timeline as the important part of the agile process. But some delays often happen as a result of internal and external factors. A less accurate timeline occurs when the team has failed to estimate the complexity of the project from the start. In some startups, it commonly happens when the team consists of less-experienced members, so the senior members should take part in creating a more accurate work schedule.

> *"Sometimes, there was a less accurate estimation in scheduling the project. It could be either an under-estimation or an over-estimation of time, especially from the junior engineers. But as the time passes, these junior engineers are able to make a more accurate timeline."* –Participant 8.

Customer involvement is an external challenge which also affects the timeline that was mentioned by the participants. Agile methodologies allow the customer to give their feedback on the development process in almost every stage. But their feedback often comes with some changes that might interrupt the whole process and delays the deadline. Therefore, they need to prioritize those request changes and decide which change is urgent and is possible to be finished before a sprint ends.

> *"Sometimes we have a lot of request changes from the customer and other teams. For any light changes we can finish it directly, but it will become a dilemma when the changes are critical because it needs approval from other teams."* –Participant 10.

However, the customer feedback still becomes critical as they are working on the customer-based project, but delays in giving the feedback sometimes happen and again lead to missing the deadline. Thus, the development team should regularly approach and communicate with the customer. Many agile users also experienced the challenge with customers who have limited knowledge in the software development process, as they sometimes do not have any clear standardization due to their request. There is a lack of awareness among customers that has been found in Al-Kautsar, et al (2013). Consequently, suggestions from the developer are needed to drive the customer's need.

4.4. Perceived success factors of agile software development projects in Indonesian startups

This subsection will address the third research question *"What are the perceived success factors of agile software development projects in Indonesian startups?"* Other success factors that come up from the data collection will also be explained in this section. Additionally, there were only 16 responses from agile participants to address this research question.

4.4.1. Management commitment

All the participants in this study consider management commitment as a success factor, though, there are 8 participants that emphasize this as a critical factor. Participants described that management commitment can be translated into several things, such as clear direction for the project, resource support, and incentive. Thus, this result can be related to Chow and Cao's (2008) theory. Based on Chow and Cao, a strong management commitment is a success factor which will be later described as a clear prioritization and adequate resources.

In a startup, the product owner is sometimes the project manager (or management) itself. A clear direction from the project manager means providing a clear prioritization of which features need to take precedence. Additionally, the feature should be completely defined based on the user story. All feature swill be compiled in the product backlog. Thus, a clear direction is needed in order to reduce misinterpretation among the development team members on the product backlog.

Meanwhile, resource support can be shaped as financial, manpower or tools. As much as agile software development projects need financial support in terms of adequate funding, the project needs to have diverse manpower that has a different set of skills which complement each other, such as front end, back end, design, and server. As for tools, the management should provide Agile software *i.e.* Redmine, JIRA or Trello, which can be quite expensive. Not just Agile software, but also sometimes it can be new devices. There is one statement from a participant that is interesting to mention here, as follows:

> *"Based on my experience, when we are developing new software, sometimes we need new devices for testing for bugs, and fortunately the management will always give that to us."* –Participant 8

4.4.2. Organizational environment

There are 8 participants who see organizational environment as a success factor. In addition, five participants highlighted this as a critical factor, while the rest said that they are unsure whether this factor is a success factor or not. Participants illustrate that organizational environment can be seen in terms of a simple structure organization. Hence, this result can be related to Chow and Cao's (2008) theory. Based on Chow and Cao, an agile-friendly organizational environment is a success factor which will be later translated into a cooperative organizational culture instead of hierarchical.

Most startups that are still in the 'efficiency' phase have a simple organization that emphasizes fluidity within the project members. Startup members from different divisions will be assembled onto a software development project in order to bring insight from different points of view. However, a few participants mentioned that there is no fixed team for every project, as the project members will be rotated based on their skills and the product requirements. Additionally, having a simple organization structure can provide another benefit in the decision-making process. A startup that has fewer members compared to a big settled company can make a decision easier and relay that decision to their members quickly and comprehensively.

There is another unique finding from 2 participants that have a client from a state-owned enterprise. That state-owned enterprise had a convoluted hierarchical organization which also has complex bureaucracy. This condition can increase the risk of failure of the project. The following statement from Participant 12 points out the risk of failure if working with a state-owned enterprise.

> *"A state-owned enterprise is strictly bound by government regulation and sometimes there is a sudden change in regulation that can lead in-progress projects to fail."*

4.4.3. Team environment

There are 8 participants that agree on team environment as a success factor, with 6 participants highlighting this as a critical factor and the other 2 are unsure whether this factor is a success factor or not. The team environment is translated into the workspace based on responses from interviewees. Thus, this finding correlates with Chow and Cao's theory (2008) of success factors in an agile project, which is an agile-friendly team environment. That explanation fits with an agile-friendly team environment that can be translated into collocation of the whole team, a facility that supports agile-style work environment, and placing a high value of face-to-face communication.

Most of the startups have a collaborative workspace which has wide open spaces in their offices. The team members are encouraged to be in the same place in order to attain face-to-face communication. Some participants raised concerns about the workspace based on their actual conditions, as not all of

the project members are present in the same place because they do their tasks remotely. Nevertheless, they already setup communication channels using teleconferences, but sometimes due to huge different time zones, it hinders the communication between members. The following is a statement from a participant emphasizes this concern:

> *"if you want to create a product on time, I prefer to have a team in the same location."*–
> Participant 2

4.4.4. Team capability

6 interviewees agree team capability is a success factor. Meanwhile, 6 interviewees underline this as a critical success factor while 3 participants are unsure and 1 participant disagree on this as a success factor. As the participants mentioned, team capability can be translated into the high technical skills of the project members. In accordance with the success factor theory from Chow and Cao (2008), high-caliber team capability means team members with high competence and expertise.

Furthermore, high technical skill has a positive correlation with time scope. Consequently, the timeline will be managed based on their capability, which also gives some assurance that they can deliver on the task that is given to them. Some participants stated that project members with low capability usually cannot keep up with the timeline. Hence, it makes the progress of projects slower.

On the other hand, having project members that have great capabilities can also bring a challenge in terms of egoism. There should be a project leader that can lead and manage them properly. Worth mentioning is a statement from Participant 18 about the leader's role:

> *"there is no superman in a team, but the most important thing is how to build a super team"*

Based on participant explanations, there is one unique value that can be found in a startup, which is a willingness to learn a new skill on their own. Keeping up-to-date with the latest technology and innovation is needed for startup members to be able to survive in the market. As they keep learning, they will increase their expertise and pile up new skills.

4.4.5. Customer involvement

Most of the participants consider customer involvement as a success factor, with 4 participants emphasizing it as a critical success factor and one participant stating this as a non-success factor. The participants that agree towards this factor stated that customer involvement in software development projects start from the beginning until the end of the project. Hence, this finding correlates with Chow and Cao (2008) theory of success factor, which is strong customer involvement in the form of good customer relationship and feedback throughout the project.

At the beginning, the customer should be actively involved in the process of making a product backlog, as the development team and product owner will ask the customer directly of their needs. Their needs will later be translated into the product backlog. Moreover, every feature that has been listed and sorted by priority in the product backlog will be done in a sprint. In the middle of the progress, when each sprint is done, the product owner or development team should report it to the client in order to get their feedback. Hence, the development team usually uses AB testing in order to bring more insights, such as bug reports and customer reactions towards their work-in-progress product before it is released to the public. There are a lot of channels for the customer to give their feedback to the development team, such as email, review in PlayStore or AppStore, or some mechanism within the software that automatically records bugs and send it to the development team.

At the end of the project, the customer will also do some evaluation on whether the end product is satisfactory or not. Based on the participant's experience, sometimes there are still minor revisions, but nothing that is major nor takes a lot of time to finish. Emphasizing strong customer involvement is needed, and there is a statement that shows the importance of customer involvement.

4.4.6. Project management process

There are 10 participants that favor project management process as a success factor, with 5 participants agreeing that it is a main factor, and 1 participant unsure whether this factor is a success factor or not. Agile project management process gives detailed instructions for project management, such as agile workflow, internal and external coordination, and documentation. Corresponding to Chow and Cao's (2008) theory of success factor, this findings fits with Agile-style project management process.

In terms of agile workflow, the project team will automatically follow several practices such as sprint, set up a meeting, sprint planning, *etc.*

"Project management process will help arrange the timeline, stages, and dividing the tasks to the team members, so it is really important." –Participant 3

As for internal coordination, the project leader – sometimes represented by a senior developer – takes a role to coordinate the tasks and responsibilities among project members. This is needed in order to match the project member's capabilities and their tasks. Moreover, external coordination means the product owner should always communicate with the customer to update their progress. If the startup has several divisions with multiple projects, it can help them to work in a collaborative way, especially if there is project dependency.

In addition, an agile project management process can simplify the documentation process. Project members can utilize agile software such as JIRA to help them check the progress and report it to the client. And not just project members, but startups can even give access for the agile software to the client in order to streamline the documentation. Obviously, this will reduce time for software development projects due to less meeting and increase the quality of the product with reactive feedback from each phase.

4.4.7. Project definition process

There are 6 participants that favor project definition process as success factor. In fact, 9 participants highlighted this factor as a critical factor, whereas 1 participant said this is not a success factor. As for participants that favor this factor, they stated that project definition process is needed in order to know what the project members are going to do based on well-defined requirements. Furthermore, the well-defined requirements will lead team members to have the same perception and vision of the project. Hence, above explanation embodies Chow and Cao's (2008) success factor, which is project definition process. Moreover, it can be further explained by following agile-oriented requirement management process.

In an agile software development project, definition does not have to be very detailed, but the objective and the general direction of the project have to be agreed on and clear. Meanwhile, the technical details of this project can be flexible, such as what technology or database that will be used for the project. Sometimes, as a software developer you do not even know what the end product would look like, because the software development project for a startup that uses the software as their product/service is a never-ending project. The software developer always seeks to answer the dynamic customer needs. The following is a statement from a participant emphasizing this condition:

"Most of the time you don't know what the end-goal of your product is, you only know what you are going to make and what the goal of the project you'll be working on is. So, the product definition is more likely to define the goal of the product, such as the target customer, what we will do, but not the actual definition of the product itself, such as are we going to use website or apps." –Participant 9

There is one experience from a participant when they handle a customer from a state-owned enterprise. The client is used to working in waterfall methods, and as he knows, waterfall methods give detailed requirements from all aspects of the project at the upfront, which brings a dilemma for the startup because they want to stick with Agile methodologies. Thus, the participant does a trade-off by satisfying both parties. The participant translated product requirements from the client into product backlog and make agreements with the client about how they will deliver the project based on sprint, but still report based on the client template, which is similar with what waterfall methods look like. The client at the beginning was quite hesitant because it seems complicated and gives more tasks to them, but after a few iterations they start to accept the process and experience the results of agile methodology.

4.4.8. Agile software techniques

Based on the interviews, most participants agree that agile software techniques are a success factor, while 4 participants do not agree with this as success factor. All of the participants that agree use SCRUM methodologies. The SCRUM methodology creates a new culture within their startup because everyone in their startup follows the scrum workflow such as sprint, daily standup meetings, *etc.* which helps team members finish the project on time. Moreover, daily standup meetings help project members to understand the condition of the on-going project. This explanation corresponds to Chow and Cao's (2008) theory of success factor, which is agile-style software engineering technique.

> *"Because with the agile method, we can create a product within 2 weeks and that's the great thing about this method"* –Participant 2

> *"I mean you can start the startup using non-agile methodologies such as waterfall and it still works, it still may lead to success, but agile methodologies in today's environment are a better option for a new startup as you cannot define the end-goal from the beginning, so the traditional project management style such as waterfall is not suitable."* –Participant 9

In reality, some participants adjust agile techniques to fit with their startup's condition. They mentioned that as a startup, especially new startup, they have limited resources. Hence, they only implemented the core scrum techniques.

> *"Not all scrum practices need to be implemented. I only implemented practices that I think help our project to achieve goals."* –Participant 12

> *"Even though we implement scrum, there are so many events that it needs to be adjusted. I do not need to follow scrum point by point (100%) in order to be successful. Each project and situation is unique. In my startup, if the project is not that complex and critical, I prefer to not use scrum; using waterfall is enough."* –Participant 1

In a contrast to people that agree with this factor as a success factor, they mentioned that the techniques only help them to manage the process, but it depends on the team itself on how to implement this technique. Rather than emphasizing this factor as a success factor, other factors such as team capability are more critical because, in the end, they are the ones that execute the project.

4.4.9. Delivery strategy

According to the interviews, most of the participants showed a positive response to delivery strategy as one of the perceived success factors in software development project. There are 6 participants who mentioned that delivery strategy as a main factor while there is 1 participant who is not sure whether or not this is a success factor, and another participant does not agree at all. They agree that the correct delivery strategy could help them create a good image and maintain the relationship with the customers. It is related to prioritizing the list of features based on their impact to the users, which critical features should become a priority, while the remaining features could be postponed into several sprints ahead.

It also corresponds to Chow and Cao's (2008) research, which explained delivery strategy as a regular delivery of the software and delivering the most important features first.

> *"I think the delivery strategy is the core success factor in the project, as with agile we can produce a new MVP (minimum viable product) in a short time, so yes, we have to manage a strategy in delivering it to the customer."* –Participant 9

There is one participant also used a spotlight place in the mobile app store to introduce their new feature to the public, which has been useful in gaining the customers' attention. Thus, they will develop the latest features to achieve that spotlight within a certain period. Another delivery strategy also described as releasing a new product or feature to the customers gradually, so the product will first launch to a small group of customers. As the developer gets their feedback, it will then be launched to a larger group until the product is finally ready to be published widely. By contrast, only one participant believe that delivery strategy gives more impact directly to the success of the company but not the project itself, as it has been created to achieve the target of the company.

4.4.10. Project nature

It has been examined that participants have a proportional argument due to project nature as a perceived success factor. Half of the participants believe that project nature should involve the project success, which is detailed by 2 participants mentioning this as a critical success factor, 5 participants mentioning this as a success factor, and a participant that partially agrees. The nature of the project can be translated into the impact of each project to the customer. It is also related to Chow and Cao (2008) which described project nature as a nonlife-critical project. Having a connection with delivery strategy, all of them agreed that projects predicted to give a huge impact to the customer should become prioritized more than projects with less impact.

> *The nature in terms of the project impact is important. As an example: fixing critical bugs is better than creating new features, and also fixing a new bug is actually a different project with its own backlog that needs to be defined. So yes, it is the main success factor based on the impact to the actual customer.* –Participant 9

The other half of the participants expressed different attitudes. Their reason was mostly that the project nature is not related to the user experience because in some cases, major features do not fulfill customer satisfaction, while small features could give a huge impact to the customer experience. In addition, whether the project has a significant impact or not, it still needs effort to work on.

> *"The project success doesn't lie on the scale of the project. Every project, whether they have a big or small impact, all of them deserve to be a success as long as we start with good planning and manage the timeline accurately."* –Participant 19

4.4.11. Project type

Most of the participants expressed positive feedback for the type of a project as one of the perceived success factors in their project. 4 participants mentioned this factor as critical, 6 participants agree this factor is a success factor and 1 participant partially agrees. Based on Chow and Cao (2008), project type in agile methodologies has been defined as being of variable scope with the emergent requirement. In accordance with this definition, all of the agreeing participants give the same perception that projects within their startups should be flexible. The flexibility allows them to adapt to the changes that commonly happen during the process, and is also typical of agile practices due to high customer involvement. There are two participants who gave interesting statements supporting project type:

> *"A flexible project will adapt more easily to changes during the project. As an example, if we found a critical bug, we have to work on this bug first before continuing with another assignment."* –Participant 5

> *"The startup cannot be rigid: you have to be flexible and you will always need to adapt to customer needs and complaints."* –Participant 9

On the other hand, there are five participants who disagree with the previous statement. They assume that typical dynamic projects with lots of changes in the middle of the project will become an obstacle, which leads to some delays. They also argue that all types of projects deserve to be a success as long as the project was managed properly.

> *"I think a flexible project will become an obstacle for the project itself. The changes that happen during the project will lead to a delay in finishing the project."* –Participant 19

4.4.12. Project schedule

It has been examined that 6 participants accepted project schedule as one of the perceived success factors. More than half of them (8 participants) also agreed that it is a critical factor for their project success. In developing software, the multiple teams consisting of the development team, design team, and QA team should work together until the product is ready to be published to the end-users. In this case, a proper schedule of the project plays a significant role to maintain the work path between teams as it reduces some idle process that might occur during the project. In connection with Chow and Cao (2008), a project should be dynamic with schedules accelerated based on their need and level of complexity. In addition, in practice even though those teams work independently, the developer still needs a final mockup design from the designer to develop the product before it is delivered to the QA team to be tested. Those sequence processes that support each project should be scheduled properly at the beginning.

> *"The different teams who work on a project need a clear schedule to prevent a lag work which will mainly affect the final stage."* –Participant 8

Another participant also experienced working with the state-owned enterprise as their client, which usually has a strict timeline. The state-owned enterprise usually has set the schedule in three months without considering the complexity of the project. Some of their projects sometimes cannot be finished within the available time slot, so they often extend the timeline without getting paid from their client. It has been encouraging the development team to create a strategy for scheduling the projects, so it can be delivered soon.

There are two participants who responded differently to the majority; they believe as agile methodologies allow some request changes from the client during the project, then the non-ideal condition commonly happens and the development team needs to adapt to changes in the schedule as well. Also, they have defined a sprint for specific working days so the way the project has been scheduled, it cannot be translated into success.

4.4.13. Other Factors

In addition to the 12 factors that have been analyzed previously, some of the participants explained that there are others factors that can be a success factor for agile software development project in a startup. The other factors are as follows: whole team commitment, communication, post-project refreshment, and skill upgrading.

Previously, management factor is a success factor for project success, but two participants mentioned that it is not only the management that needs to be committed, but also the whole team, such as clients as represented by product owners, the development team itself, and the scrum master. If the whole team already agreed to do the project in an Agile way, they need to follow all the processes such as attending daily standup meetings every morning and update their progress in Agile software such as JIRA.

As for the communication factors, this factor is almost integrated into most of the success factors and needs to be highlighted. Furthermore, good communication skills can decrease conflict and increase

positive collaboration within the team. The following is an interesting comment from a participant that mentions the communication factor:

> *"Even though you do not have the advanced skills to finish the task, by having good communication, you can ask your colleagues to teach you."* –Participant 12

There is another interesting success factor, which is post-project refreshment. Working in a startup with the same routines every day and a huge pressure due to high risk of failure in developing their service/product can bring boredom among the project members. Post-project refreshment such as dinner or spending time together after the project is finished are potential solutions. The following statement emphasizes the need for post-project refreshment:

> *".... because we are working with people, we need to manage their passion and mood so they will give the best performance."* –Participant 17

The last other factor is skill upgrading. As participants described their startups, there is always a skill gap between the junior members and the senior ones. Thus, this condition brings more disadvantages than benefit. In order to minimize the skill gap, the management needs to support their project members to upgrade their skills by giving them a workshop or training.

5. Discussions

5.1. Comparison with previous studies

6 critical success factors have been found among 12 perceived success factors. The author defines critical success factor based on the data collection process, where the participants have commonly mentioned some factors as critical factors. The critical success factors are as follows: (1) Project Definition Process; (2) Management Commitment; (3) Project Schedule; (4) Team Environment; (5) Team Capability; (6) Delivery Strategy. Furthermore, the other perceived success factors are less common or not even mentioned once as a success factor. Additionally, it can be interpreted that specific perceived success factors only become critical in their startup environment. Thus, the author cannot generalize those specific perceived success factors to other startups and make it as a critical success factor.

In accordance to Chow and Cao's studies in 2008, they mentioned that *"It may be worthwhile to repeat such a study again in five and ten years to see whether any new factors may emerge or current key success factors become no longer critical"*. Thus, the researcher aims to fill this gap by comparing Chow and Cao's (2008) and Safwan et al.'s (2013) studies with this study in 2018. Table 10 shows comparison between the different studies.

Table 10. Comparison with Previous Studies

Parameter	Chaw and Cao (2008)	Safwan et al. (2013)	Author (2018)
Methods	Quantitative	Qualitative	Qualitative
Sample	109 Project with 406 respondent	60 respondent - 41 Agile - 19 Non-Agile	20 project represented by 20 respondent - 16 Agile - 4 Non-Agile
Sample Scope	Worldwide	Sri Lanka	Indonesia
Agile Methods	XP, SCRUM, other	SCRUM, XP, other	SCRUM
Findings:			
- Critical Success Factor	- Delivery Strategy - Team Capability - Team Environment - Project Management Process - Agile Software Techniques - Customer Involvement		- Delivery Strategy - Team Capability - Team Environment - Project Definition Process - Management Commitment - Project Schedule
- Reason		- Enforced by superiors - Motivated by its benefits - Keeping up with the changing environment	- Enforced by superiors (mentors) - Motivated by its benefits - Fit with startup's simple organization structure
- Challenges		- Costly - Time wastage - Insufficient team support	- Financial funding (Cost) - Time wastage (inefficient timeline)

		- Poor quality - Repetition of work	- Communication between team member - Techniques (daily standup meeting, scrum master) - Customer involvement

Based on Table 10, the author will give deeper interpretations by comparing (1) Chow and Cao with the author's research, and (2) Safwan et al. with the author's research.

5.1.1. Research from Chow and Cao (2008)

The author's research was made 10 years after Chow and Cao's research. At a glance, the findings in critical success factors found in both research are different. But these differences can only indicate that new critical factors have emerged. This happens because the author did not fully replicate the research methodology and research questions in this research. Instead of using quantitative research, the author decided to use qualitative research in order to gain more insight from the agile practitioners. The author also minimized the research scope by choosing Indonesian software development startups as the research scope. Another thing that needs to be mentioned is agile methodologies that have been used in software development projects show that in Indonesia, agile practitioners commonly use SCRUM as their methods.

In this research (2018), the agile software techniques, project management process, and customer involvement are not identified as critical success factor for agile software development projects. While the critical success factors are as follows: critical success factors such as project definition process, management commitment, project schedule, team environment, team capability, and delivery strategy. Hence, the last three factors are similar with critical success factors from Chow and Cao (2008).

Agile software techniques become not critical in this study because most of the participants are still new in implementing the Agile methodology. With limited resources, they need to adjust agile techniques in order to fit the startup's condition. As for project management process, it is less important because agile project management has already given a clear instruction such as agile workflow and documentation, thus they only need to follow it. The last factor is customer involvement. Even though in normal conditions customer involvement is a critical success factor and it is a reflection of Agile values, in Indonesian startups not every customer understands or is even aware of Agile methodologies. One example is the startup that has a client from a state-owned enterprise: instead of becoming a success factor, it can become a challenge for a startup to educate the client.

As for new emerging success factors, instead of project management process, the management commitment is needed in order to guarantee the project has sufficient resources such as funding, manpower, and devices. Additionally, project definition process becomes the key success factor due to the nature of Indonesian startups that are still in the 'efficiency' phase, the startup needs to refine their value proposition in order to scale their business. Value proposition refinement is obtained by well defining requirements from the customer. As for the last factor, the project schedule is needed because when developing a product, it is broken down into multiple projects that are handled by a number of different teams such as the development team, the design team, and the quality assurance team. Hence, it can maintain the smooth transition between the team and reduce idle processes that might be occurred.

5.1.2. Research from Safwan et al. (2013)

Author's research is comparable with Safwan et al. as both studies use qualitative methods with similar sample scopes due to Indonesia and Sri Lanka both being developing countries. Moreover, there is only a 5-year gap between these two.

The similarities in findings can be found both in reasons and challenges of agile software development implementation. In terms of reasons, two are the same and only one that is different. In term of a simple organization that startups have, it supports the implementation of agile methodologies. Moreover, a simple organization leads to quick decision-making without complex bureaucracy, since bureaucracy is the main problem for the customer from the state-owned enterprise. While the reason from Safwan et al. that keeping up with the changing environment becomes the main reason due to their sample coming from a big company, Sri Lankan Airlines IT Department, that previously implemented the waterfall method and is eager to transition into agile methodologies.

As for the challenges, there are two similarities in terms of cost and time wastage. In addition, costliness in Safwan et al. means a high investment for implementing Agile, while financial funding (cost) means limited funds from investors to hire a scrum master or buying agile software such as JIRA. On the other hand, communication between team members is also part of insufficient team support from Safwan et al. If there is poor communication within the team, it will lead to a less supportive team environment. The participants did not see repetitive work and poor quality as challenges because they are more capable in managing the task delegation to make sure that every team member is responsible for their own task, which leads to deliver a good quality of the product. Indonesian startups expressed customer involvement and agile-oriented techniques as challenges due to the limited knowledge both from customers and startups regarding agile methodology.

5.2. Correlation with startup characteristics

The perceived critical success factors found in software development projects in Indonesian startups correlates to the startup characteristics that have been defined by Paternoster, et al (2014). These fifteen characteristics have built a certain environment for a startup to evolve and sustain under remarkably uncertain conditions in order to achieve their target. This section will describe a comprehensive analysis of six critical success factors in regard to the startup characteristics. Table 11 shows a correlation of critical factors with startup characteristics.

Table 11. Correlation of critical success factor with startup characteristics

Critical Success Factors	Startup Characteristics
Project Definition Process	• Low-experienced team • Third party dependencies • New company
Management Commitment	• Uncertainty • Lack of resources • Not self-sustained
Project Schedule	• Highly reactive • Time-pressure • Low-experienced team
Team Environment	• Small team • Flat organization • Lack of resources
Team Capability	• Small team • Low-experienced team • New company

Delivery Strategy	• One product
	• Innovation
	• Rapidly evolving

The project definition process becomes the first factor that has been commonly mentioned as the main success factor in a project. Chow and Cao (2008) categorized process factors which impact the quality, scope, time, and cost. In this research, this factor has been explained as a well-defined requirement in the early stage including instructions, assignments, and goals. It becomes important since clear directions will lead team members to have the same perception and vision about the project. However, as agile methodologies allow sudden changes during the project, the project definition does not have to be very detailed, but still needs to be objective and flexible in the technical side. Some of the participants said they often do not know what the final project would be like because the software development project itself uses the product or services as a never-ending project. There will be an incremental innovation in developing the product following the customer's demands.

This factor is associated with the startup as an amateur business which has started within less than five years, while they began using agile in the last two years. It also mostly created by the young generation from a millennial age that also leads them to recruit their friends who likewise have similar experiences to them. As the young generation who starts a business or professionally work in a team, they still lack experiences. So, a clear definition of the project will actually help them understand their assignments, and the more familiar they are with certain projects will lead to more efficient work. Most of the startup's project is based on the client's demands. In this case, the client also acts as the third party that keeps the startup growing and surviving. Moreover, a client should clearly explain their requests about the product, and if there are any changes while the project runs, those changes should also be clearly defined as it helps the development team to create a high-quality product to satisfy the client.

Half of the agile users noticed that management commitment is also a critical factor in the success of the project. The management commitment is one of the organizational factors in the research model introduced by Chow and Cao (2008), and it also covers all four success dimensions. In association, an active involvement and support from management has significant impact on the implementation of agile methods (Al Kautsar et al., 2013). This factor has been expressed by several parameters such as a clear direction for the project, resource support, and incentive. This factor has to be achieved before running a project due to a clear definition of the product backlog being important in reducing misinterpretation for the development team.

Paternoster et al. (2014) classified that startups often deal with a highly uncertain ecosystem under different perspectives: market, product features, competition, people, and finance. It exposes the startup to an extremely high failure risk when the management refuses to commit to the projects. In startups that have a simple organization, the management level consists of few people with experience in business, so every decision made were considered from those perspectives, but mainly they are concerned with the financial and technical as critical. The management commitment also needs to understand the process in the agile framework in order to give an appropriate support for the resources, as a startup is typically limited in economic, human, and physical resources. It also correlates with the sustainability of the startup, as a lack of resources has brought a startup to not self-sustain, especially in the early stage.

The project schedule becomes the most popular success factor that has been defined by the participants. Similar to the management commitment, the project schedule has been mentioned as a critical factor by half of the agile users. Based on Chow and Cao (2008), the project schedule is classified in project factors grouped together with project nature and project type. This factor was described as plotting the project within the right timeline, especially when a startup is handling several projects simultaneously. It is also related to multiple teams who are working together in developing a product. The teams are

composed of designer teams, development teams, and QA teams. The teams work consecutively, so a proper project schedule will reduce idle process between teams and avoid any heavy workload for the QA team in the final stage.

Furthermore, startups are able to give a quick response to any changes that happen during the project. Since they are dealing with a customer-based project, the client has a right to interrupt in the form of requesting changes in the middle of the process. So, in particular cases, the schedule also needs to be flexible in adapting to sudden changes. On the other side, startups are required to deliver their program within a tight period which forces them to get used to constant pressure. The customer becomes an important part of the startups in keeping their sustainability, so they need to achieve the customer's satisfaction and gain another customer soon. The team capability in terms of the limited experience of the members also affect the process of scheduling the project, and a less accurate timeline can come from junior members which leads the team in missing the deadline.

Another critical factor is the team environment which all the participants agreed on as a success factor, while some of them emphasized that it is also a critical factor. This factor is located within the same group as management commitment in the organizational factors on Chow and Cao's (2008) research model. An ideal team environment is created by the team members that have a similar perception of the agile mindset, goal, and passion. Also, every member needs to contribute to the team with their own capability. Some of the participants highlighted communication as the main part of building a proper team environment. Thus, face-to-face communication is a high priority, which was also emphasized by Chow and Cao (2008).

Startups commonly lack resources, especially the employees, meaning startups work with a small team. The small team has supported a higher value in face-to-face communication when the team members collocate. It also leads to a pair programming that some of the participants have as an alternative way. The pair programming has effectively helped the members in bridging their skill gap and also strengthened the bond between the senior and junior members. In another case of communication, it has been found that some of the startups also have team members working remotely in different areas or even different countries. In addition to a success factor, the communication also becomes a challenge, but as they have intensive communication through online meetings or social media, then the challenge turns into a huge success. Flat organization within the startup also helps to build an agile-friendly environment because all the decisions can be made quickly without any complex bureaucracy.

All participants agreed that team capability is a success factor for a project, and half of them also highlighted this factor as a critical factor. Based on Chow and Cao (2008), the team capability is categorized in the people factors group that affects the quality, scope, time, and cost of the startup. It has been translated as high technical skill from each member because the timeline will be managed based on their capability. Some of the participants said that team capability highly contributes to the successful project. Thus, having team members with great capabilities will lead to a highly efficient project. But in some cases it might bring a sense of egoism between the members as a highly capable person will be looking at a more proper team or project.

Regarding the startup characteristics, startups that were recently created commonly recruit a fresh graduate to reduce high-cost spending in salary. Consequently, the fresh graduate employees have limited experience in technical skills towards several projects. In connection with the team environment factor, pair programming also helps them assure that every member has similar levels of capability. In addition, pair programming fits better in the small team environment.

The last critical factor is the delivery strategy that has been frequently noticed by most of the participant. There are six agile users among the participants who mentioned this factor as the main factor in project success. The delivery strategy is a part of technical factors shown in Chow and Cao's (2008) research model. This factor is translated into the strategy in prioritizing the features based on their impact to the

customer, deciding which features should become a priority which needs to be finished first and launched to the costumer, while other, lesser priority features will be launched after several sprints ahead. In addition to a priority list of features, the strategy also includes launching the product to the customer gradually, which is defined as releasing the product to a small number of customers, and if those customers give good feedback, then it will be continuously launched to other customers until it is ready to be published broadly.

The correct delivery strategy also helps the startups in creating a good image and maintaining a relationship with the customers because most startups focus on developing one product, so the customer which represents a target market has become important parts to work with. An incremental innovation also becomes one of the startup characteristics. Instead of launching a major innovation within a long period, the startup is rapidly evolving and adapting to the changes in customer behavior and technology. Thus, the startup will gradually update their product frequently.

To sum up, the startup characteristics that were introduced by Paternoster, et al (2014) showed a positive correlation with all the critical success factors found in this research. Some of the characteristics are significantly associated with certain critical success factors. However, the whole startup characteristics are linked to each other in creating a unique ecosystem of the startups.

5.3. The government support to increase startup survival rate

Since Joko Widodo, the Indonesian President stated that the country needs to be "The Digital Energy of Asia", the government starts to create programs and regulations to support fledgling business. It was beginning with regulation No. 11 in 2008 about information and electronic transaction (UU ITE No.11 Tahun 2008), Regulation No.7 in 2014 about Trading (UU No. 7 Tahun 2014), and President's Commands No. 74 in 2017 about Road Map e-Commerce for 2017-2019 (Perpres No.74 Tahun 2017). Those regulations have helped young entrepreneur to get their legal protection in running their startup business.

Besides regulation, the government also put attention in stimulating young generation to start their business and help new startups to develop with some innovative program such as incubator academy and accelerator. As an example, Bekraf has BEKUP (Bekraf for Pre-Startup) to give guidance and advices for startups within different stages. In addition, the government through Ministry of Communications and Information Technology collaborate with Kibar also creates Gerakan Nasional 1000 Startups Digital (*Eng: National Movement for 1000 Digital Startups*) which act as incubator for startups in each cycles.

Both of those programs has five roles as follows: (1) give access to the community and business network, (2) support from experienced mentor, (3) idea validation, (4) infrastructure, (5) give access to the investor. The first role helped young entrepreneur to introduce their product to the public through festival and exhibition. Startups ideas are commonly born to solve community's problem, thus introducing their solution in the form of product to the public will give them some insights. Mentors also become a critical part or incubation program, they were assigned and trained by the professional to build an interactive mentoring session. They also trained with agile professional to implement agile project management in startups. The agile management indeed will not give any direct indications to the success factor of startups, but looking at general startup characteristics which have lean organizations, limited member and experience, and customer dependent, agile management provide an alternative way to fulfill the young entrepreneur needs. This alternative way gives their startup opportunity to increase productivity which will lead to higher survival rate. Furthermore, the young entrepreneur as a new player in market needs advices and guidance about developing their product, thus idea validation is important to give them suggestions from different point of view. Regarding infrastructure, the government provides cheaper or even free co-working spaces for startups to build

their product properly. The last role is government as stake holder also give an access for developing startups to get their funding from the investor. Incubation programs usually have some potential investor which is ready to give their funding to the most valuable startups.

6. Conclusions

6.1. What is the adoption of agile software development projects in Indonesian startups?

The implementation of agile methodologies has been broadly found in software development projects in IT related Indonesian startups, with Scrum as the most common agile practice. Even though the participating startups come from different backgrounds, all of them have their own software development projects, which can be further extended into developing their own software or developing the client-based software. Agile methodologies have been commonly used for months until years as it becomes a trend in Indonesia, since workshops and training in importance of this method has helped educate startups. It is related to the mentor or the project manager's preference as a main reason in applying agile methodologies that lead the team members to become familiar and eager to learn about the whole agile process. Startups also have a simple organization that enables them to create quick decisions which support quick responses to the customer's request. In addition, this agile methodology allows customers to be frequently involved in almost all the stages, giving their feedback or even request changes during the project.

All of the agile users were examined as 'being' agile as they already have an agile mindset due to their understanding of the values, principles, and practices. Nevertheless, in terms of 'doing' agile, not all of them are fully agile users where they use all the agile software techniques; some of the participants are partially agile who use several agile techniques that fit their working environment. Most of the partially agile users skip the retrospective phase.

Instead of applying agile methodologies, a minority of the participants choose to implement a traditional way, such as waterfall methodologies. Limited members within a team has been raised as the main reason they are still keeping up the current method, as they explained that having only a few members negatively impacts the capability of the team, especially in handling lots of changes occurring during the project. On the other hand, as they work in client-based projects, most of their clients are commonly familiar with waterfall rather than agile. In addition, there is a financial concern to get licenses of agile tools software. Despite all the reasons, they are eventually interested to use agile methodologies in the future when they have a high caliber team with sufficient financial support.

6.2. What are the challenges of implementing agile software development projects in Indonesian startups?

The implementation of agile methodologies also brings some challenges for agile users. The challenges are categorized into two main groups: internal and external challenges. The internal challenges have been described as agile techniques, team members, less accurate timelines, and financial funding. Agile techniques cover ineffective daily standup meetings and the Scrum master. The inefficient daily standup meetings were found where team members discuss a deep technical issue, which takes a longer time for the meetings. Meanwhile, a Scrum master's role has been absent in several startups due to limited team members and funding to hire a capable person. The limited financial funding was also raised as a challenge to use licensed agile tools such as JIRA and Confluence software, which leads them to use free agile tools as alternatives. The communication within the team also becomes a challenge, particularly when they have several teams working remotely. In addition, a challenge in creating a timeline commonly happens as the team failed to estimate the complexity of the project.

The external challenge includes the customer involvement if the customer has limited knowledge in the software development process, leading to unclear standardization from the project's start. Furthermore, they also mentioned the customer feedback within the project usually come with some changes that might interrupt the whole process and lead to delays. Some delays also happen with the customer feedback in which the development team should regularly approach and remind the customer.

6.3. What are the perceived success factors of agile software development projects in Indonesian startups?

There are 12 perceived success factors that have been asked from the participants through interviews. In general, all of the factors were accepted to contribute to project success. Those perceived success factors are described as follows: (1) Management Commitment, (2) Organizational Environment, (3) Team Environment, (4) Team Capability, (5) Customer Involvement, (6) Project Management Process, (7) Project Definition Process, (8) Agile Software Techniques, (9) Delivery Strategy, (10) Project Nature, (11) Project Type, and (12) Project Schedule. According to the interviews, the participants responded differently to each factor, which helped in examining the critical factors. There are six factors that have been commonly mentioned by agile users as critical factors to their success of the project.

The main six critical factors are found to be: (1) Project Definition Process, (2) Management Commitment, (3) Project Schedule, (4) Team Environment, (5) Team Capability, and (6) Delivery Strategy. Those factors have been obtained from the most common factors mentioned by the participants, as most of the participants also clearly emphasized those factors as critical factors for their project success. Moreover, the other perceived success factors were less commonly mentioned by the agile users.

The critical success factors were compared to the previous study by Chow and Cao (2008). It has been examined that there are three critical factors from Chow and Cao (2008) that are not found in this research. Based on the previous study, the critical factors agile software techniques, project management process, and customer involvement were not commonly noticed as a critical factor, but they have been replaced by project definition process, management commitment, and project schedule. Those three factors from Chow and Cao (2008) that were not important appeared due to Indonesian startups characteristics: (1) limited resources lead to adjusting the agile techniques, (2) limited knowledge on the part of the customers in agile methodologies and the software development process made customer involvement a challenge, (3) the project management has already given a clear instruction for the workflow and documentation.

Due to limited sources of critical factors from Safwan et al. (2013), the comparison study only focuses on the reasons in using agile methodologies and challenges that happen during the implementation of this method. As for the reason, there is only one reason found to be different from Safwan et al. (2013) which is keeping up with the changing environment. In this research, the participants come from software development projects within new startups where most of them have been using agile methodologies since the beginning, while Safwan et al. (2013) took samples from software development projects within a big company, Sri Lankan Airlines IT Department. This Sri Lankan company had previously implemented the waterfall method and then migrated to agile methodologies. Furthermore, there are three differences in challenges: Safwan et al. stated poor quality, insufficient team support, and repetitive work, while this study found agile techniques (daily standup meetings, Scrum Master), communication between team members, and customer involvement are also the challenges.

The critical factors also positively correlate with the general startup characteristics introduced by Paternoster, et al (2014). Fifteen characteristics are significantly matched to certain critical factors. However, the whole startup characteristics link to each other in creating a unique ecosystem of the startups. The correlation between critical success factors with the startup characteristics are described as follows: (1) Project Definition Process: low-experienced team, third party dependencies, new company; (2) Management Commitment: uncertainty, lack of resources, not self-sustained; (3) Project Schedule: highly reactive, time-pressure, low-experienced team; (4) Team Environment: small team, flat organization, lack of resources; (5) Team Capability: small team, low-experienced team, new company; and (6) Delivery Strategy: one product, innovation, rapidly evolving.

7.   Contributions and Recommendations for Future Research

7.1. Academic contributions

Most of the academic contributions come from empirical contribution. Previously in Subsection 1.2 Aims and Purposes, it is explained that the aims of this research was to fill the gaps and provide an academic literature for Indonesian startups, especially for agile software development projects. The author hopes the findings will help to reduce the difficulties encountered when they start implementing agile methodologies. Additionally, this study will also help them to focus on several critical success factors in order to achieve project success, which will eventually lead to the sustainability of the startup.

As for theoretical contributions, the author noticed that there are very few theories discussing startups, especially Indonesian startups. This study has contributed in compiling all of the scattered information on startups and Indonesian startups. In addition, the researcher provided a comprehensive analysis in perceived success factors in agile software development projects.

The last academic contributions are the methodological contributions. This study provides a research design which can be used in similar studies, especially in a way to tackle ethical considerations. Moreover, putting an interviewee as anonymous from the beginning helps the author to get a better understanding of their situation. Thus, interviewees are willing to speak the truth and in more detail.

7.2. Recommendations for Further research

Due to the scope and time limitations, this study only focused on agile software development projects in Indonesian startups. The author only used interviews as a data collection method to answer the research questions, and chose to make a general conclusion for perceived success factors including the adoption and challenges for agile software development projects. However, the conclusion given here does not provide an absolute comprehensive picture. Thus, the author suggests it would be interesting if other researchers replicate this research, and add observations and focus group discussion methods with more samples in order to accept or reject the conclusions that have been made here. In addition, observations and focus group discussions are needed to get a deeper understanding on a target group.

On the other hand, the author suggests that it may be possible to replace the target group with companies, especially state-owned enterprises that have already been established for a long time. It would be interesting to see their perceived success factors due to different characteristics compared to startups. Such studies would help understand the agile phenomena from a different point of view. Furthermore, it will also lead to more comprehensive academic literature for Indonesian academicians and government.

References

A.T. Kearney and Google, 2017. *Indonesia Venture Capital Outlook 2017,* s.l.: A.T. Kearney - Google.

Abrahamsson, P., Salo, O., Ronkainen, J. & Warsta, J., 2002. *Agile Software Development Methods: Review and Analysis.* Espoo: VTT publication 478.

Agile Business Consortium, 2018. *Business Agility.* [Online]
Available at: https://www.agilebusiness.org/business-agility
[Accessed 1 March 2018].

Ahimbisibwe, A., Daellenbach, U. & Cavana, R. Y., 2017. Empirical comparison of traditional plan-based and agile methodologies: Critical success factors for outsourced software developmentprojects from vendors' perspective. *Enterprise Information Management,* 30(3), pp. 400-453.

Al-Kautsar, E., Salleh, N., Hoda, R. & Asnawi, A. L., 2013. *Challenges in adopting Agile practices: Perceptions of Software Practitioners in Indonesia.* Pattaya, the 5th International Conference on Internet (ICONI 2013).

Ambler, S., 2006. Supersize me. *Software Development,* 14(3), pp. 46-48.

Anderson, R., 2007. *Thematic Content Analysis: Descriptive Presentation of Qualitative Data Using Mocrosoft Word,* s.l.: s.n.

Aron, D. & McDonald, M., 2014. *Taming the Digital Dragon: The 2014 CIO Agenda,* Stamford: Gartner.

Augustine, S., Payne, B., Sencindiver, F. & Woodcock, S., 2005. Agile project management: steering from the edges. *Communications of the ACM,* 48(12), pp. 85-89.

Badan Ekonomi Kreatif, 2016. *Regulation of The Head of The Creative Economy Agency of The Republic of Indonesia number 10 of 2016 on The provision of Government Assistance for Initial Funding Startup Business,* Jakarta: Badan Ekonomi Kreatif.

Badan Ekonomi Kreatif, 2017. *Profil Usaha/Perusahaan 16 Subsektor Ekraf Berdasarkan Sensus Ekonomi 2016 (SE2016).* Jakarta: Badan Pusat Statistik.

Badan Ekonomi Kreatif, 2018. *Profil BEKRAF.* [Online]
Available at: http://www.bekraf.go.id/profil
[Accessed 1 March 2018].

Beck, K., 2014. *Kent Beck talks beyond Agile Programming @ Startup Lessons Learned Conference 2010.* [Online]
Available at: https://www.youtube.com/watch?v=d4qldY0g_dI
[Accessed 13 June 2015].

Beck, K. & Andres, C., 2005. *Extreme Programming Explained, Embrace Change.* 2nd ed. Boston: Pearson Education, Inc..

Beck, K. et al., 2001. *Manifesto for Agile Software Development.* [Online]
Available at: http://agilemanifesto.org/
[Accessed 1 March 2018].

Bertaux, D., 1981. From the life-history approach to the transformation of sociological practice. In Daniel Bertaux (Ed.). *Biography and society: The life history approach in the social sciences,* pp. 29-45.

Blank, S., 2010. *A Startup is Not a Smaller Version of a Large Company.* [Online]
Available at: https://steveblank.com/2010/01/14/a-startup-is-not-a-smaller-version-of-a-large-company/
[Accessed 20 May 2018].

Blank, S., 2013. *The Four Steps to The Epiphany.* 2nd ed. s.l.:K&S Ranch.

Boehm, B. & Turner, R., 2003. Using risk to balance agile and plan-driven methods. *Computer,* 36(6), pp. 57-66.

Boehm, B. & Turner, R., 2005. Management challenges to implement agile processes in traditional development organizations. *IEEE Software,* 22(5), pp. 30-39.

Bryman, A. & Bell, E., 2011. *Business Research Methods.* 3rd ed. New York: Oxford University Press.

Carroll, E., 2002. *LIGHTWEIGHT AND AGILE SOFTWARE DEVELOPMENT.* [Online]
Available at: https://ubiquity-acm-org.ezproxy.its.uu.se/article.cfm?id=512514
[Accessed 6 April 2018].

Ceschi, M., Sillitti, A., Succi, G. & Panfilis, S., 2005. Project management in plan-based and agile companies. *IEEE Software,* 22(3), pp. 21-27.

Chow, T. & Cao, D.-B., 2008. A survey study of critical success factors in agile software projects. *The Journal of Systems and Software,* Issue 81, pp. 961-971.

Cockburn & Highsmith, J., 2001. Agile software development: The people factor. *Computer,* Volume 34, pp. 131-133.

Cohn, M., 2007. *Differences Between Scrum and Extreme Programming.* [Online]
Available at: https://www.mountaingoatsoftware.com/blog/differences-between-scrum-and-extreme-programming
[Accessed 20 May 2018].

Cohn, M. & Ford, D., 2003. Introducing an agile process to an organization. *Computer,* 36(6), pp. 74-78.

Coleman, G. & O'Connor, R. V., 2008. An Investigation into Software Development Process Formation in. *Journal of Systems and Software,* 81(5), pp. 772-784.

Conforto, E. C. et al., 2014. Can Agile Project Management Be Adopted by Industries Other than Software Management?. *Project Management Journal,* 45(3).

Cresswell, J. W., 2013. *Qualitative inquiry and research design: Choosing among five approaches.* Sage ed. California: Thousand Oaks.

Dabrowski, M., 2017. *Scrum study guide mind map.* [Online]
Available at: https://www.mindmeister.com/443624561/scrum-study-guide-mind-map
[Accessed 6 April 2018].

DailySocial, 2016. *Indonesia'sTech Startup Report 2016,* s.l.: DailySocial ID.

Denning, S., 2015. Updating the Agile Manifesto. *Strategy & Leadership,* 43(3).

Dybå, T. & Dingsøyr, T., 2008. Empirical studies of agile software development: A systematic review. *Information and Software Technology,* Volume 50, pp. 833-859.

Eka, R., 2017. *Temuan Menarik tentang Startup di Yogyakarta Tahun 2017.* [Online]
Available at: https://dailysocial.id/post/temuan-menarik-tentang-startup-di-yogyakarta-tahun-2017/
[Accessed 16 May 2018].

Entrepreneurinsight, 2016. *Start-Up versus SME – Know The Difference.* [Online]
Available at: http://entrepreneurinsight.com.my/start-up-versus-sme-know-the-difference/
[Accessed 20 May 2018].

Field, T., 1997. When BAD Things Happen to GOOD Projects. *CIO Magazine,* pp. 55-62.

Giardino, C. et al., 2014. What Do We Know about Software Development in Startups?. *IEEE Software,* pp. 740-759.

Glaser, B. G. & Strauss, A. L., 1967. *The Discovery of Grounded Theory: Strategies for Qualitative Research.* Chicago: Aldine.

GOJEK INDONESIA, 2017. *Go-Jek Engineering - Guiding Principles 101.* [Online]
Available at: https://www.go-jek.com/blog/go-jek-engineering-guiding-principles-101/
[Accessed 19 May 2018].

Gontovnikas, N., 2016. *Faster Than the Rest: Why Staying Agile Is the Key to Startup Success.* [Online]
Available at: https://dzone.com/articles/faster-than-the-rest-why-staying-agile-is-the-key
[Accessed 15 June 2018].

Highsmith, J., 2002. *Agile Software Development Ecosystems.* Boston, Massachusetts: Addison-Wesley.

Jung, Y., Wang, J. & Wu, S., 2008. Competitive strategy, TQM practice, and continuous improvement of international project management: a contingency study. *Quality and Reliability Management,* 26(2), pp. 161-183.

Jun, L., Qiuzhen, W. & Qingguo, M., 2011. The effects of project uncertainty and risk managementon IS development project performance: a vendor perspective. *Project Management,* 29(7), pp. 923-933.

Jurca, G., Hellman, T. D. & Frank, M., 2014. *Integrating Agile and User-Centered Design: A Systematic Mapping and Review of Evaluation and Validation Studies of Agile-UX.* Kissimmee, IEEE.

Karlstrom, D. & Runeson, P., 2005. Combining agile methods with Star-. *IEEE Software,* 22(3), pp. 43-49.

Kelly, A., 2017. *Agile for Startups: Getting the most from your software development team,* s.l.: Allan Kelly Associates.

Kibar, 2018. *Profil Gerakan Nasional 1000 Startup Digital.* [Online]
Available at: https://1000startupdigital.id
[Accessed 17 May 2018].

Kominfo, 2018. *Menkominfo: Peran Pemerintah Jadi Fasilitator dan Akselerator.* [Online]
Available at: https://www.kominfo.go.id/content/detail/12798/menkominfo-peran-pemerintah-jadi-fasilitator-dan-akselerator/0/berita_satker
[Accessed 15 June 2018].

Kusumasari, T. F., Supriana, I., Surendro, K. & Sastramihardja, H., 2011. Collaboration Model of Software Development. *International Conference on Electrical Engineering and Informatics.*

Larman, C., 2004. *Agile & Iterative Development.* Boston, Massachusetts: Addison-Wesley.

Lee, T., 2015. *What is a startup?.* [Online]
Available at: https://www.techinasia.com/what-is-startup-definition
[Accessed 20 May 2018].

Lindvall, M. et al., 2004. Agile software development in large organizations. *Computer,* 37(12), pp. 26-34.

MacCormack, A., Verganti, R. & Iansiti, M., 2001. Developing Products on "Internet Time": The Anatomy of a Flexible Development Process. *Management Science,* 47(1), pp. 133-150.

Majalah CSR, 2017. *Hadir, Sekolah Startup di Indonesia.* [Online]
Available at: http://majalahcsr.id/hadir-sekolah-startup-di-indonesia/
[Accessed 15 June 2018].

Marmer, M., Herrmann, B. L., Dogrultan, E. & Berman, R., 2012. *Startup Genome Report,* Oakland: Startup Genome.

Mason, M., 2010. Sample Size and Saturation in PhD Studies Using Qualitative Interviews. *Forum: Qualitative Social Research,* 11(3).

Meyer, J. & Allen, N., 1997. *Commitment in the Workplace: Theory, Research, and Application.* Thousand Oaks, CA: Sage Publications.

Mishra, D. & Mishra, A., 2011. Complex software project development: Agile methods adoption. *Software Maintenance and Evolution Research and Practice,* 23(8), pp. 549-564.

Moroni, I., Arruda, A. & Araujo, K., 2015. The design and technological innovation: how to understand the growth of startups companies in competitive business environment. *Procedia Manufacturing,* Volume 3, pp. 2199-2204.

Nerur, S., Mahapatra, R. & Mangalaraj, G., 2005. Challenges of migrating to agile methodologies. *Communications of the ACM,* 48(5), pp. 72-78.

Newcomer, E., 2018. *Uber Completes Sale of $9.3 Billion in Stock Led by SoftBank.* [Online]
Available at: https://www.bloomberg.com/news/articles/2018-01-18/uber-completes-sale-of-9-3-billion-in-stock-led-by-softbank
[Accessed 19 May 2018].

Nidumolu, S., 1996. A comparison of the structural contingency and risk-based perspectives oncoordination in software-development projects. *Management Information System,* 13(2), pp. 77-113.

Nobel, C., 2011. *Why Companies Fail—and How Their Founders Can Bounce Back.* [Online]
Available at: https://hbswk.hbs.edu/item/why-companies-failand-how-their-founders-can-bounce-back
[Accessed 20 May 2018].

Okezone, 2017. *Didukung Regulasi Pemerintah, Pelaku Bisnis Start Up dan Masyarakat Merasa Terbantu.* [Online]
Available at: https://news.okezone.com/read/2017/11/23/542/1819249/didukung-regulasi-pemerintah-pelaku-bisnis-start-up-dan-masyarakat-merasa-terbantu
[Accessed 15 June 2018].

Paternoster, N. et al., 2014. Software development in startup companies: A systematic mapping. *Information and Software Technology,* Volume 56, pp. 1200-1218.

Pentalog, 2017. *Agile Methodologies: The Perfect Match for Startups.* [Online]
Available at: https://www.pentalog.com/blog/agile-methodology-startups/
[Accessed 20 May 2018].

Ponnappa, S., 2017. *The point of story points.* [Online]
Available at: https://blog.gojekengineering.com/the-point-of-story-points-456e8e483653
[Accessed 19 May 2018].

Pratama, A. H., 2016. *5 Strategi Pemerintah untuk "Melahirkan" 1.000 Startup Digital di Indonesia.* [Online]
Available at: https://id.techinasia.com/pemerintah-luncurkan-gerakan-nasional-1000-startup-digital
[Accessed 20 May 2018].

Pratomo, 2016. *Tingkat Kegagalan Startup Indonesia Masih Tinggi.* [Online]
Available at: https://techno.okezone.com/read/2016/10/03/207/1505060/tingkat-kegagalan-startup-indonesia-masih-tinggi
[Accessed 31 March 2018].

Rahayu, N., 2017. *Amvesindo Demo Day 2017 Dukung Ekosistem Startup Indonesia.* [Online]
Available at: https://www.wartaekonomi.co.id/read149826/amvesindo-demo-day-2017-dukung-ekosistem-startup-indonesia.html
[Accessed 1 March 2018].

Rajendran, N., 2001. *Dealing with Biases in Qualitative Research: A Balancing Act for researchers.* Kuala Lumpur, Qualitative Research Convention.

Ranganath, P., 2011. *Elevating Teams from 'Doing' Agile to 'Being' Agile and 'Living' Agile.* Salt Lake City, IEEE.

Reel, J., 1999. Critical success factors in software projects. *IEEE Software,* 16(3), pp. 18-23.

Reifer, D., Maurer, F. & Erdogmus, H., 2003. Scaling agile methods. *IEEE Software,* 20(4), pp. 12-14.

Ries, E., 2011. *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses.* New York: Currency.

Sabdarini, I. T., 2018. *Infografis Ringkasan Data Statistik Ekonomi Kreatif Indonesia,* Jakarta: Badan Ekonomi Kreatif.

Safwan, M. et al., 2013. An Empirical Stduy of Agile Software Development Methodologies: A Sri Lankan Perspective. *International Journal of Computer Applications,* 84(8).

Salleh, N., Al-Kautsar, E., Hoda, R. & Asmawi, A. L., 2014. A Window into the Emergence of Agile Software Development Landscape in Indonesia. *Int. J. Advance Soft Compu. Appl,* 16(1), pp. 1-17.

Santos, J. M. D., 2017. *XP, FDD, DSDM, and Crystal Methods of Agile Development.* [Online]
Available at: https://project-management.com/xp-fdd-dsdm-and-crystal-methods-of-agile-development/
[Accessed 20 May 2018].

Schatz, B. & Abdelshafi, I., 2005. Primavera gets agile: A successful transition to agile development. *IEEE Software,* 22(3), pp. 36-42.

Senanyake, G., 2011. *The Effectiveness of Scrum Project Management in Sri Lankan Context.*
[Online]
Available at: www.ccs.neu.edu/home/rukmal/.../AshaSenanayake_MB A_Thesis.pdf

Simon, M., 2012. *Dissertation and scholarly research: Recipes for success.* 2013 edition ed. Seattle:
Dissertation Success, LLC.

Sommerville, I., 2011. *Software engineering.* 9th ed. s.l.:Pearson.

Stankovic, D., Nikolic, V., Djordjevicc, M. & Cao, D., 2013. A survey study of critical success
factors in agile software projects in former Yugoslavia IT companies. *The Journal of Systems and
Software,* Volume 86, pp. 1663-1678.

StartupRanking, 2018. *Countries.* [Online]
Available at: https://www.startupranking.com/countries
[Accessed 1 March 2018].

Thanedar, N., 2012. *Are You Building A Small Business - Or A Startup?.* [Online]
Available at: https://www.forbes.com/sites/theyec/2012/08/15/are-you-building-a-small-business-or-
a-startup/#2dfd461ca528
[Accessed 20 May 2018].

Wan, J. & Wang, R., 2010. Empirical research on critical success factors of agile software process
improvement. *Software Engineering and Applications,* 3(2), pp. 1131-1140.

Wells, D., 2009. *Extreme Programming: A gentle introduction.* [Online]
Available at: http://www.extremeprogramming.org/
[Accessed 6 April 2018].

World Bank, 2016. *GDP (current US$).* [Online]
Available at: https://data.worldbank.org/indicator/NY.GDP.MKTP.CD?year_high_desc=true
[Accessed 28 February 2018].

Appendix

Appendix 1 - Interview Template

Information

This study is being carried out by Nuruddin Kamil; an MSc student at Uppsala University taking Industrial Management and Innovation. This is being conducted under the supervision of Ibrahim Alaff and Professor Marcus Lindahl of Department of Engineering Sciences.

It's important you read the following information before consenting to take part in this study. If there are any issues you wish to raise or any questions you may have, please email the address below before participating.

Due to the nature of this research, you should be a part of a software development project on the technical side belonging to an Indonesian software development startup that is in the efficiency phase of startup lifecycle and has already been running for approximately 8 months or more.

You will be asked a series of interview questions, which will take an estimated 30-45 minutes to answer.

The aim of the research

The aim of this study is to explore the perceived success factors of agile software development projects in Indonesian startups.

Do I have to take part?

You are under no obligation to take part.
If you decide to participate, all the data collected is completely anonymous. As such you can withdraw your response if you feel inconvenienced with the interview results.

Will my information be shared?

All data collected is completely anonymous. In every interview, the voice will be recorded and used only for research purposes. Thus, only the researcher and the supervisor will have access to this data.

Any problems?

If you have any concerns or questions, please contact:
Nuruddin Kamil (Nuruddin.Kamil.8095@student.uu.se)

Interview Question

Section 1

1. Do you use agile software development in your project? If yes, continue to section 2. If no, continue to section 4.
2. How long have you used that agile software development?
3. Why do you use agile software development?
4. What are the agile software development practices that you use?
5. How does agile software development go in your project?

Section 2

6. What are the challenges that you face when implementing agile software development projects?
7. How do you overcome the challenges?

Section 3

8. Perceived success factors for project success
   a. Do you think **Management Commitment** is a success factor? Why?
   b. Do you think **Organizational Environment** is a success factor? Why?
   c. Do you think **Team Environment** is a success factor? Why?
   d. Do you think **Team Capability** is a success factor? Why?
   e. Do you think **Customer Involvement** is a success factor? Why?
   f. Do you think **Project Management Process** is a success factor? Why?
   g. Do you think **Project Definition Process** is a success factor? Why?
   h. Do you think **Agile Software Techniques** is a success factor? Why?
   i. Do you think **Delivery Strategy** is a success factor? Why?
   j. Do you think **Project Nature** is a success factor? Why?
   k. Do you think **Project Type** is a success factor? Why?
   l. Do you think **Project Schedule** is a success factor? Why?
9. Are there any other factors that you think are success factors of agile software development projects?

Section 4

Do you use agile software development project? IF NOT:
1. What kind of project management in software development project do you use?
2. Why do you not use agile software development?
3. Why do you use that particular project management?
4. Are you interested in using agile software development in the future?

Appendix 2 – Interviewee Profiles

No	Name	Gender	Startup Type	Startup Member	Startup Operated (year)	Agile	Method	Agile Practicing (year)
1	Participant 1	Man	Fintech	>100	>5	Yes	SCRUM	1-3
2	Participant 2	Man	E-commerce	>100	1-3	Yes	SCRUM	<1
3	Participant 3	Man	On-demand	>100	3-5	Yes	SCRUM	1-3
4	Participant 4	Man	Software House	20-99	1-3	No	Waterfall	
5	Participant 5	Man	On-demand	5-19	1-3	Yes	SCRUM	1-3
6	Participant 6	Man	Fintech	20-99	3-5	Partial	SCRUM, Waterfall	<1
7	Participant 8	Man	Game Developer	5-19	3-5	Yes	SCRUM	>3
8	Participant 9	Man	E-commerce	5-19	1-3	Yes	SCRUM	1-3
9	Participant 10	Woman	E-commerce	>100	>5	Partial	SCRUM, Waterfall	>3
10	Participant 11	Man	Software House	5-19	3-5	Yes	SCRUM	1-3
11	Participant 12	Man	Software House	5-19	1-3	Yes	SCRUM	<1
12	Participant 13	Man	Digital Agency	20-99	>5	No	Waterfall	
13	Participant 15	Man	E-commerce	5-19	1-3	No	Waterfall	
14	Participant 17	Man	Software House	5-19	>5	Partial	SCRUM, Waterfall	<1
15	Participant 18	Man	E-commerce	>100	>5	Yes	SCRUM	>3
16	Participant 19	Man	E-commerce	20-99	1-3	Yes	SCRUM	<1
17	Participant 20	Man	On-demand	5-19	1-3	Yes	SCRUM	<1
18	Participant 21	Woman	Digital Agency	5-19	3-5	No	Waterfall	
19	Participant 22	Man	Software House	20-99	>5	Partial	SCRUM	<1
20	Participant 23	Woman	Fintech	>100	3-5	Partial	SCRUM	1-3

Participant 7, 14 and 16 were disqualified because one was not related, and the other two did not give meaningful feedback

Appendix 3 – Keywords for Theme

Reason for implementing Agile Software Development project

1	Superior preference	Request, boss, mentor, preference, scrum master,
2	Motivated by its benefits	Fit, product, goals, objective, business, process, breakdown
3	Fit with startup's simple organization structure	Structure, task, assignment, effective, efficient, divide
4	Familiarity	Familiarity, capable
5	Rapid Development	Quick, new, feature, development, rapid, changing

Challenges for implementing Agile Software Development project

1	Techniques	Scrum master, daily stand up
2	Communication between team member	Ineffective, dependent, member, communication
3	Time wastage	Complexity, sprint, definition, delay, unfinished, user, demand, change,
4	Financial funding (Cost)	Tools, costly, foreign, coordination
5	Customer involvement	Feedback, customer, awareness, timeline, education, unclear, goal

Appendix 4 – Transcript Interviewee Example

Participant 1

Time   : Sunday, 15th April 2018, 06.00 AM/11.00 WIB

Name   :

Startup   :

Position   :

Team Member : 200 people as whole startup, with 3 teams, (strategy teams 1 person), (software development team 4 persons),

Condition   : Postponed by one hour, disturbed by family, first interview

This interview has been conducted in English using voice call.

Section 1

10. Do you use Agile software development project?

    Yes

11. How long have you used that agile software development?

    2 years

12. Why do you use agile software development?

    At that time, all the early team members were familiar with SCRUM, it is the most commonly used right now if you are software developing team in startup. So if we recruit programmers, they are most likely be familiar with SCRUM, so we do not need to teach them again about SCRUM. It's convenience, at that time, it is what we know best and team members are familiar with SCRUM. We just go with the convenience and popularity.

13. What is the agile software development practice that you use?

    SCRUM

14. How does agile software development go in your project?

    Basically, every software development cycle is divided into sprint. Before each sprint, we did a sprint planning, we were also using a proper planning to determine the workload and the resource for each feature. We use Kanban board to track the progress of each sprint. We just implement several best perspective of sprint. 1 sprint runs for ten working days (2 weeks).

Section 2

15. What is the challenge that you face when implementing Agile software development project?

    They do not have any specific challenge for implementing Agile because from the beginning they already use SCRUM. As for daily use of SCRUM they also do not have any meaningful challenge. The process of SCRUM itself is quite straightforward. Everybody doesn't have any problem. In terms of tools, it is not a challenge, but especially for tools like JIRA and Confluence are expensive for early startup. The dilemma comes when your startup has few

members (3-5 members) but you need to pay around 50 USD/month to use Agile tools. Is it worth it or not? After the startup members increase into 50 people, the tools are become crucial and needed for collaboration.

16. How do you overcome the challenges?
At that time, we just decided that we used JIRA and confluence. We treated it as investment. In terms of scrum tools, actually you do not need to use any specific tools, you can also use free agile tools. As for my understanding, we choose the paid version such as JIRA and confluence because those are the standard. If you are recruiting programmer, usually they are familiar with JIRA.
In his personal perspective, He suggest that if it is still early startup, better to use free software for agile tools such as redmine.
JIRA and Confluence are made by Atlasian. Those tools are integrated each other. The flow is PRD (Product Requirement Document) is made in confluence then you can export it directly into a task or sprint in JIRA. It is really convenience. So if your startup has minimum 10 members, it will be worth it.
If your startup member is less than 10 people, you can use Google Docs to make PRD.

Section 3

17. Perceived success factors for project success
    a. Do you think **Management Commitment** is the perceived success factors? Explain why?
    Yes. The management commitment can be translated into two parts. The first one is resources. You will get the allocated person and budget for your project. Clearly this is an important factor for your project. The second part is incentive. Some people can get some incentive in a form of recognition, money or etc. if they can finish their job in a project. It will drive the project member to work diligently.

    b. Do you think **Organizational Environment** is the perceived success factors? Explain why?
    Partial, in a startup, organizational environment is not a problem because position within the project is a fluid thing. We work based on a project. Based on the organizational structure, I only responsible to strategy and operation, but in many cases i also manage software development part. And also for the scrum, basically we assemble different project members from different department for several sprint. So I do think in early startup, organizational environment is less matter because in each sprint, you will exclusively work with same member in short time period.

    c. Do you think **Team Environment** is the perceived success factors? Explain why?
    Yes, in term of scrum, you measure your progress daily by doing daily standup meeting. In that sense, the scrum won't succeed if the team have a lot of conflict, do not talk to each other, do not support each other. In a sprint within the scrum, the collaboration between project member is going to be very intense so conducive team environment is needed.

d. Do you think **Team Capability** is the perceived success factors? Explain why?
    Yes, but you do not necessarily need to fully understand scrum, but you need to understand what is your duty based on your position. The project will success if you do your duty. In each sprint, the scrum master will make sure that every member understands and also manage the scrum process.
    The team does not necessarily in the same page of the scrum but as long as they have the capability to finish the job, the project will be success.

e. Do you think **Customer Involvement** is the perceived success factors? Explain why?
    Yes, if you are in a sprint, all the project member need to be collaborative mode. The business user (customer) has to be heavily collaborated with project member. A sprint is needed to develop a feature of a software.
    In day 1 of the sprint, the developer can ask directly to the customer whether their assumption is correct or not without waiting until the end of sprint (ten working days). Especially in this startup, if you are business user, you have to be involve and attend daily standup meeting. Those people will sit in the same table and discuss in order to get direct feedback based on the progress.
    As for the end customer, the end customer will be represented by product manager. It is important that product manager knows what end customer wants. Before the sprint started, the product manager need to ask directly to the customer by doing a UX testing. Once the assumption about the feature is validated, then they can start to do the project. The end customers do not need to involve daily to the project member which is different with business user.

f. Do you think **Project Management Process** is the perceived success factors? Explain why?
    Yes, project management is one aspect of scrum, so if you are implementing scrum, you will automatically follow several methodologies such as sprint, stand up meeting, sprint planning that can ensure project success,
    Project manager is needed especially if it is a big project, but every member is expected to manage their own deadline. In software developer project, the deadline manages by product manager and also lead software engineer.
    Implementing scrum and having dedicated person such as project manager is help the startup to make a project success, in a sense of meeting all the requirement, deadline, quality of the product, etc.

g. Do you think **Project Definition Process** is the perceived success factors? Explain why?
    Yes, you have to be a clear and everyone have to understand the project itself. They are expected to be understand the goal (deliveries) of this project by the end of this sprint. If people doesn't understand the project scope, everything will be chaos. Based on my experience, you do not need to be super detail, it is different with Waterfall method which everything need to be detailed and agreed upon members. In Scrum, you only need to have high level of objective i.e. increase engagement in website. Engagement can be translated into the number of visitor of the website. They want to increase the engagement X%. The way scrum solve this objective can be flexible. Take for the

example, in sprint A, the team want to make 10 features, but then, in the middle of the sprint they realize that some of the feature is not working as they expected, then they decide to change it.

So project definition does not have to be very detail, but I think the objective and the general direction of this project have to be agreed in the beginning. Meanwhile, the technical details of this project can be flexible.

The question of WHAT & WHY should be detailed, in the other hand, the question HOW can be less detailed.

h. Do you think **Agile Software Techniques** is the perceived success factors? Explain why?

Not really important, even though they implement scrum, there are so many event that they need to be adjusted. I do not need to follow the scrum point by point (100%) in order to be successful. Each project and situation are unique. In ▮▮▮▮▮, they implement SCRUM but they neglect some practices within the scrum. SCRUM techniques is not really important. If the project is not that complex and critical, I prefer to not use scrum, using waterfall is enough. Majority of our project are using scrum, but some minor project such as building minor dashboard for management, you can use waterfall. In ▮▮▮▮▮, you can't be rigid, you just choose whatever fits in the project. Just do it, start coding!

i. Do you think **Delivery Strategy** is the perceived success factors? Explain why?

Yes, every year we have a strategy meeting to define the big objective, so it basically can be 2-3 strategic objective and we break it down into quarter. For each quarter, they will define the feature that need to be released. It called top down planning, it is written in the roadmap made by product manager. In the other hand, there is also bottom up planning, this bottom up planning is written in backlog that consist of bug, things to be improve, customer feedback. Our delivery strategy is combination between strategical objective and daily bug fixing from the customer.

If the delivery strategy is translated into prioritization of released feature, then it becomes important factor but it is the domain of product manager because so many things that to be done, a lot of pressure and managing daily bug fixing. Prioritization means solving the conflict of interest between parties.

j. Do you think **Project Nature** is the perceived success factors? Explain why?

No, I don't think it is relevant. Because SCRUM is a set of guidelines and very generic, you can implement it whatever you want, as long as you understand the benefit, and the objective of agile methodologies. Whatever the software project, it will be the same.

k. Do you think **Project Type** is the perceived success factors? Explain why?

No, the project type is also not relevant due to SCRUM is only a set of guidelines, SCRUM is not a tools, is not a people.

l. Do you think **Project Schedule** is the perceived success factors? Explain why?

No, I do not think this is important, they already define each sprint is 10 working days. They do not know what is the reason why they use 10 days. If it small project, you can

change it into half sprint which is 5 days. The way you schedule the project doesn't translate into the success of the project.

18. Is there any other factor that you think become the perceived success factors of agile software development project?

    The whole team commitment, if they already agree to do this project in a SCRUM way, they need to be committed in a sense of available in standing meeting, upload the progress in JIRA. Everyone have to be agree on this and they need to follow all the regulations.

    If it is implemented in a Big Company, the problem occurs because the business user is not attending the standup meeting. It can be fatal if after a sprint finished, the business user does not accept the result. It will waste all the effort and resource.

    Do not use scrum because you just follow the trend, but use scrum because it's benefit.

Section 4

Do you use Agile software development project? IF NO

5. What kind of project management in software development project that you used?
6. Why do you not use agile software development?
7. Why do you use that particular project management?
8. Do you interest to use agile software development in the future?