

Article

# Initial Coin Offerings and Agile Practices

Simona Ibba <sup>1,\*</sup>, Andrea Pinna <sup>1</sup> , Maria Iliaria Lunesu <sup>1</sup> , Michele Marchesi <sup>2</sup> and Roberto Tonelli <sup>2</sup>

<sup>1</sup> Department of Electric and Electronic Engineering, University of Cagliari, 09123 Cagliari, Italy; a.pinna@diee.unica.it (A.P.); ilaria.lunesu@diee.unica.it (M.I.L.)

<sup>2</sup> Department of Mathematics and Computer Science, University of Cagliari, 09124 Cagliari, Italy; marchesi@unica.it (M.M.); roberto.tonelli@dsf.unica.it (R.T.)

\* Correspondence: simona.ibba@diee.unica.it

Received: 18 September 2018; Accepted: 19 October 2018; Published: 23 October 2018



**Abstract:** An ICO (Initial Coin Offering) is an innovative way to fund projects based on blockchain. The funding is based on the selling of tokens by means of decentralized applications called smart contracts written in Solidity, a programming language specific for Ethereum blockchain. The ICOs work in a volatile context and it is crucial that the team is capable of handling constant changes. The Agile methods, proven practices enabling to develop software in presence of changing requirements, could be a means for managing uncertainty. The main goals of this work are to understand software engineering activities related to ICOs, recognize the ICOs developed using Agile methods, and make a comparison between ICOs and Agile ICOs. In addition, we perform a deeper analysis of Agile ICOs concerning project planning, software development, and code features. Our work shows that the roles of the people involved in an ICO can be compared to the typical roles of the SCRUM methodology. The majority of Agile ICOs use tool of testing before storing smart contract on blockchain. Finally, the application of volumetric and complexity software metrics shows that the files of Agile ICOs is on average shorter and less complex than in other smart contracts.

**Keywords:** Initial Coin Offerings; Agile methodology; smart contracts; blockchain

## 1. Introduction

An ICO (Initial Coin Offering) is a new way to perform crowdfunding campaigns, based on blockchain technology like Bitcoin and Ethereum [1–3]. It allows to finance startups working on blockchain technology or applications on a global scale, directly and without intermediaries. With the aim of involving as many investors as possible, a startup will create and distribute its tokens. The token of the ICO can be developed through a smart contract, a computer program running on a public blockchain [4]. Smart contracts are the base for the development of decentralized applications (DApps). Most smart contracts, used for ICOs, run on Ethereum blockchain, and are usually written in a programming language called Solidity. Although startups performing ICOs share many characteristics with software startups not based on blockchain technology, several factors make the software development context of ICOs unique. Consequently, the research presented in this paper aims primarily at understanding the main software development characteristics of ICO startups, from the planning phase to the testing phase, also taking into consideration the quality of code. To this purpose, we analyzed the whole set of ICOs gathered from ICObench (available at <http://icobench.com>) and registered between August 2015 and 20 February 2018. We focused on their main features, such as: The team size and roles in order to understand if its composition is consistent with the best known software development methodologies; the rating provided by ICObench in order to investigate the projects quality; the use of social media in order to discover if ICOs teams believe that the

communication with investors and their feedback are important; the financial aspects related to motivating the people involved in an ICO.

The market of ICOs is extremely volatile and complex. ICO teams and investors should pay particular attention to the speed of changes and to the technological risks that characterize its evolution.

Furthermore, software startups in general, and even more so, startups founded through an ICO, operate in conditions of great uncertainty and the team's ability to manage changes is crucial. Software startups, therefore, need to follow the market trends, and to adapt to ever-changing business activities and risks [5]. For this reason, in this work we investigated the use of Agile methodologies in ICOs, as a mean of managing uncertainty.

According to prior studies [6–8], Agile methodologies are optimal for projects that, like ICOs, demonstrate high variability in the development process, in the team or stakeholders abilities, and in the technology being used. In particular, Agile development is especially appropriate for products or services providing high value for the customers, and for all involved stakeholders [7]. Considering the decentralized nature of the blockchain, an ICO can have investors, customers, and other stakeholders from all over the world. In our work we aim to understand if and how Agile practices are used in ICOs in response to such high technological, process, and market variability. To do this, we take into account the principles of Agile Software Development and we study if and how Agile methodologies and practices are used in the ICOs development.

First, for each ICO registered on ICObench until 20 February 2018, we carried out a textual analysis of its online available documentation, in order to detect any typical Agile keywords. In this way, we found a subset of ICOs developed with an Agile approach (for the sake of simplicity named Agile ICOs in the following). We also made a comparison between the characteristics of Agile ICOs and the properties of the whole dataset in terms of team composition, rating, financial aspects, and use of social media.

In addition, we analyzed more in depth the set of Agile ICOs, specifically focusing on their roadmap and their project development. We conducted an analysis of smart contracts [9] source codes of the Agile ICOs in terms of code metrics and use of test tools.

To summarize, the main contributions of this paper are:

- Understanding the main characteristics of ICOs, investigating software engineering activities related to ICOs, recognizing the ICOs developed using Agile methods, and making a comparison between the characteristics of Agile and not Agile ICOs.
- Providing a deep analysis of the Agile ICOs in terms of their project planning, software development, and source codes.

The remainder of the paper is organized as follows. Section 2 describes the most relevant related work on ICOs and on Agile methodologies. Section 3 presents the research method that we followed in our study. This section describes the steps which allowed us to define our dataset, to find the Agile ICOs, and to perform our analysis. Section 4 describes the results of the analysis of the ICOs dataset, including the comparison between the results obtained for the ICOs in general, and the results obtained for the Agile ICOs. These results include the team composition, the rating, the financial aspects. Section 5 analyzes in depth Agile ICOs. This section includes the results of the analysis of the ICO roadmaps, of the software projects, and of the smart contracts. Section 6 provides a discussion about the outcomes of our analysis. Finally, Section 7 concludes the paper.

## 2. Related Work

Research literature on blockchain in general, and on ICOs in particular, is limited to the last few years [10]. The principles described in the Agile Manifesto [11] are the basis for most Agile methodologies. Agile methodologies were used in Sabrix, Inc. (San Ramon, CA, USA) [12], a startup software company that, to accommodate the pressing demands of users, exploited the urgency as the main engine for the product development, and allowed the startup to switch from a initial chaotic

management to the correct implementation of a software product–team and product progressed simultaneously. In order to verify if Agile practices are applied in software startups, [13] performed a survey involving 1526 software startups, with questions related to five Agile practices, including regular refactoring, agile planning, frequent release, and daily standup meeting. The survey was focused on the relationship between velocity and quality in Agile practices, and they discovered that the software startups favored velocity over quality. Ref. [14] claims that startups run the risk of failure and of being quickly out of business, if some engineering practices are not used. They studied the software development strategies employed by startups, and found that it is necessary to reduce time-to-market, speeding up the development of product using users' feedback. Ref. [15] provides an exploration of the state-of-art on software startup research and specifies which software engineering practices must be chosen to increase the ability of startups to survive under highly uncertain conditions. Recently, Ref. [16] proposes a unified and multidimensional framework to represent together the role, active or passive, of digital startups with respect to change, and to the level of dynamism of the environment. According to the results of this exploratory study, Lean Startup Approaches (LSA) are strongly related to the use of Agile development methodologies. In addition, startups oriented to have an active role in determining changes use the approach called Business model innovation. Ref. [17] shows that the Agile and Lean Startup methodologies can be compatible and complementary. Agile methodologies drive software development, whereas the Lean Startup methodology is more oriented towards the development and management of the business and of the product. The blockchain technology is an invention that led to a high dynamism in several business areas [2]. It gave rise to the definition of a new branch of software engineering called "Blockchain-Oriented Software Engineering" [18]. According to [19], the blockchain technology and in particular the invention of digital tokens, has the potential to create a *new entrepreneurial landscape*, representing the opportunity to invest in early-stage projects, and, on the other hand, the opportunity for startups to fund their projects in a more democratic way. These opportunities represent the core of the Initial Coin Offerings (ICOs) phenomenon, subject of this study.

Ref. [20] presents a model that rationalize the use of an ICO for the launch of a peer-to-peer platform that still needs to be built. This work highlights two strategies that can generate value: A coordination model among many subjects involved in a peer to peer network and the use of "popular wisdom", that is, the analysis of information available on the web and posted by users or by other stakeholders that describe the quality of the platform.

The possibility of using an ICO as a fundraising tool to finance business and technology initiatives directly and without intermediaries was analyzed in [21]. In this work, the Lean Startup methodology as a tool to face the main critical aspects of a startup and examined some ICOs based on lean startup approach, is analyzed.

A first overview about ICOs was made in [22]. They examined all ICOs, published on 2017 on icobench.com website in order to evaluate their quality and software development management and to discover the features that can influence the ICO success. A similar analysis is described in [23]. According to this work, the success factors of an ICO originated in the process behind the organization of the ICO. Another success factor is the quality of the services provided to the investors who buy the tokens.

The main problem of an ICO is the capability of investors to make a distinction between a genuine fundraising activity and a scam. Ref. [24] analyzed this phenomenon, collected information from specific websites, and categorized the ICOs in order to identify their key success factors. Other issues are related to the legal aspects of an ICO. Another aspect to be managed, is the possibility of changes in ICO legal regulation. The rapid explosion of the ICOs phenomenon has generated some legislative loopholes. In [25] it is pointed out the lack of clear rules for the accounting of ICO funds in the company balance sheet. In [26] it is described how ICOs are regulated in different countries. At the moment, there are no uniform ICO regulation standards.

### 3. Research Method

The phenomenon of crowdfunding based on cryptocurrency is very recent. This study is the first in literature about this subject, and has an exploratory nature. We analyzed all ICOs records on ICObench (<https://icobench.com>) until 20 February 2018. This is a free ICO rating platform that collects data about thousands of ICO. It provides the ICObench Data API (Application Programming Interface described in <https://github.com/ICObench/data-api>), allowing developers to get the information stored in the platform, including ICO listings, ratings, and stats. The research approach is composed of two main parts: Data collection and analysis of all ICOs registered on ICObench until 20 February 2018, and the choice, through appropriate keywords, of the ICOs to be specifically analyzed, because they are managed with an Agile approach.

#### 3.1. Data Collection

We divided the data acquisition process into different steps, as described below. To collect the data related to ICOs we used ICObench Data API, introduced on 12 December 2017 (according to <https://medium.com/@ICObench/icobench-2017-in-numbers-d987b0a280d0>), to study the list of all ICOs, the list of ICOs by search parameters and filters, the list of all ICOs ratings, all information in the ICO profile and other statistics. ICObench Data API has already been used in other major studies [27–29]. We called the extraction of this data “Step 1”.

To identify which ICOs exhibit an Agile approach, we established a search string with pre-defined keywords which we will introduce in Step 2.

##### 3.1.1. Step 1

We collected the ICOs’ data from the specialized website called *icobench.com*. To date, this website provides an useful API for developers and data scientists who want to create new applications or to analyze the ICO phenomenon. For the API specification see <https://icobench.com/developers>, accessed on 16th July 2018.

In particular, this API needs user authentication and uses the HMAC method with SHA384 algorithm to authenticate the query. The data provided are in JSON format. In order to acquire the full available data, we used the POST request called “ICO-Profile”, which we sent for all ICOs’ “id”. The “ICO-Profile” request has URL <https://icobench.com/api/v1/ico/id|url>.

We implemented in R [30] the procedure to automatize the connection to the API, to call the request, to organize and save the ICO data in the R list data type. We collected the data of the first 1.952 ICOs recorded in the *icobench.com* database until 20 February 2018. We discovered that 115 of them had no available data. The data of the acquired ICOs occupy about 50 MB of memory.

Each list item describes an ICO with up to 25 named sublists, that groups hundreds of named values. We focused on five sublists: Team, rating, finance, dates, links. The team includes name, country, title, link to socials media, and so on for each team member. The rating provides the *icobench* ICO evaluation vote. The dates includes the date related to the timing of the ICO (opening, closing). Links contains the URLs of the ICO official website and the link to the ICO white paper. Because of the importance of the ICOs’ white papers, we wrote a R script to download all the available documents. We collected a total of 1.144 readable PDF files. The ICOs white papers pdf files occupy 4.3 GB of memory.

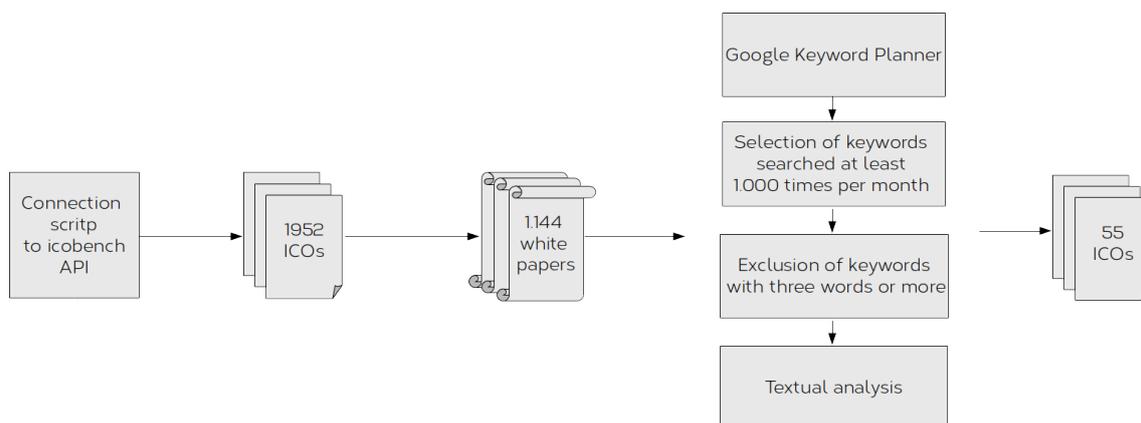
##### 3.1.2. Step 2

We identified the ICOs that apply Agile practices by searching for keywords in their white papers. A white paper is a comprehensive technical report describing the ICO product or service. We chose the keywords to look for in the following way.

- We used the Google Keyword Planner tool and we obtained all the keywords associated with the main keyword: “*Agile methodology*”. In this way we obtained a total of 700 keywords.
- We selected all keywords that have an average monthly number of searches on Google above 1000.

- We included keywords consisting of single words (for example “*scrum*”) and their specifications covering at most another word. For example, we selected the keyword “*scrum programming*” and we excluded the keyword “*scrum programming development*” because it is implicitly included in the previous one.

Eventually, we obtained 90 keywords. We performed a textual analysis, by means of a R script, to verify in which white papers at least one of the 90 selected keywords was present. We analyzed all 1.144 readable white papers. The script, converts the pdf content in a text string. Subsequently, for each white paper it counts the number of occurrences of each keyword in a case-insensitive mode. As a result, the script returns the table of keywords occurrences per ICO white paper. We identified 55 ICOs in which at least one of the 90 selected keywords is present. We, then, manually verified that each of these 55 white papers actually referred to an Agile software development mode. The obtained subset is about the 5% of the total analyzed ICOs. For simplicity, in the rest of the document, we will call this subset *Agile ICOs*. It must be pointed out that in our classification we focused only in the ICOs development approach, that can be Agile or not Agile. To be clear, in these two categories of approaches, the use of the blockchain technology is the same, and the differences between Agile and not Agile ICOs regard other aspects we will discuss in the following. Figure 1 schematically describes the process that allowed us to identify the 55 ICOs.



**Figure 1.** Flow diagram describing the selection process of the Agile ICOs.

### 3.2. Analysis Setup

After the creation of the set of Agile ICOs, we compared them with the overall set of ICOs. We implemented specific R scripts to collect and analyze the information available in the ICO dataset. We collected data about the size of the team, and its composition in terms of roles and gender. We obtained the gender of team members by means of a names database realized by Mark Kantrowitz (available from <http://www.cs.cmu.edu/afs/cs/project/ai-repository/ai/areas/nlp/corpora/names/>). We used R to classify the team members by gender, to perform statistical and comparative analysis, and to collect and analyze some of the business and financial information available in the ICO dataset, also including the rating and the use of social media.

## 4. Data Analysis

In this section, we report the results of the analysis. As described in section Research Method, we organized the analysis in steps. In the followings we present the numbers, the distributions and the statistical values that characterize the acquired ICOs, and in particular the Agile ICOs.

#### 4.1. Analysis of ICOs Teams

According to [31], one of the main factors that impact the sustained usage of Agile methods is the team composition, that should have a right balance in terms of technical skills, domain knowledge, team size, and gender, race, and culture.

##### Team Size and Composition

We analyzed a total of 18,699 people involved in ICO founding teams. We firstly analyzed the size of the team that develops each ICO. We consider 1646 ICOs that declare at least one team member. We found that the mean size of the ICO team amounts to 11.4 people. According to the results of the Anderson-Darling normality test, the team size distribution does not follow a normal distribution. The Anderson-Darling normality test, computed in R using the library *nortest*, produces a  $p$ -value  $< 2.2 \times 10^{-16}$  (the distribution is normal if  $p$ -value  $> 0.05$  [32]). The maximum team size includes 67 people. These results are higher than the results reported in [21], where the average team size was 10.87, and the maximum team size was 57.

When we computed these statistics on the subset of 55 Agile ICOs, we found that ICOs in this typology have larger teams. The average size is 14.9 people, despite the fact that the larger team includes 43 people. Also in this case, the distribution is not normal. The Anderson-Darling normality test provide a  $p$ -value = 0.0055, lower than the threshold. We, then, performed the Wilcoxon test [33] which is not based on distributional assumptions and therefore provides reliable results even when data do not have a Gaussian distribution. Through the test carried out using the implementation provided by R, we can see that the differences are significant. The  $p$ -value =  $5.003 \times 10^{-6}$  is lower than the threshold value  $\alpha = 0.05$ . Empirical evidence is strongly contrary to the null hypothesis and the observed data are statistically significant.

This data has its intrinsic importance. In an ICO, a very small and anonymous team increases the investment risk and may be a scam. It is therefore important for investors to be able to consult the information relating to each team member, both on the ICO website and on LinkedIn or Twitter. The greater number of people involved, with a detailed description of each person supported by the link to the related social pages, the lower the probability that the ICO may be a scam.

An advisor is a domain expert (for instance an academic), or an investor, or a consultant. We found that 4.461 people involved in ICOs are advisors. As shown in Table 1, advisor represents 18.9% of team composition, or in other terms there is nearly an advisor every five people.

**Table 1.** Summary of statistics on ICO team. The average percentage of advisors and females per team are computed on teams of at least one person.

ICO Stats	Avg. Size	Max Size	% Advisors	% Women
All	11.4	67	18.90	16.30
Agile	14.9	43	23.04	15.00

In Agile ICOs, there are more advisors. On average, they represent over the 23% of the team. According to [22], in a very large team sometimes there are many advisors who contribute suggestions, but are not really involved in the ICO operations. We can consider advisors as a group of individuals with experience and able to provide credibility and value to the project, as long as they specifically work in the ICO and create added value. We can see that some advisors have their names in over 100 projects, each of which spans overlapping periods of time. Such an advisor cannot physically allocate enough time to a project to provide true value to their customers. We can compare the roles inside an ICO team with the typical roles described in the SCRUM methodology. In particular, in the SCRUM methodology the world is divided into “pigs” and “chickens” [34]. The former, during the project development, bring into play “the skin”. All other stakeholders are spectators (chickens). The chickens may also be strongly interested in the project, but they do not work in a strict and direct

way like the pigs. By analogy, in an ICO we can define team members as pigs, while advisors can be considered as chickens. The advisors are consultants who can support the team if it is needed; often they are selected for marketing reasons, and do not work directly to the development of the project.

#### 4.2. Gender Heterogeneity

We investigated the gender heterogeneity in ICO teams. It is well known that a global gender gap exists in entrepreneurship, and in particular in the technological startup founding teams. Female presence in startup teams is typically below 30%, which is the highest percentage recorded in the Chicago startup ecosystem [35].

For example, in a recent survey [13] that examined about 1526 software startups, they discovered that in their teams a very small percentage (8%) are females, in comparison to the percentage of males (76%). Note that 16% of team members did not reveal gender information.

In addition, ICO teams consist of predominantly males. Our algorithm classified 9776 people; we found that the female presence is equal to 16.3%. Considering the average of the number of men and women per ICO founding team, we found that the number of men is about 5 times larger than the number of women. Considering the Agile ICO set, we noticed that the female presence is slightly lower than the presence computed in the total dataset. These teams have 15% of women, being on average composed by 6.5 men per 1 women. There is no case of team composed only by women.

In the ICOs, however, the presence of women is twice the presence detected by [13] in software startups not based on an ICO. Ref. [36] shows that gender diversity plays a significant role when considering productivity and collaboration within a software development team. These results could also be applied to ICO teams. In addition, also in relation to investors, according to [37], the number of women interested in investments in cryptocurrencies represents currently the 13% (one in eight women) of the total. This research also suggests that women invest very differently than men: The former take a much more strategic approach, and suffer less the “Fear of Missing Out” [38] than their male counterpart. The study also shows that women investors tend to collaborate much more than men, consulting family and friends about their investment before proceeding.

#### 4.3. ICO Rating

The statistical analysis of the ICO Rating (a score parameter provided by ICObench that summarize the ICO reliability) allow us to compare the overall results with the subset of Agile ICO. The Rating is a real number which ranges from zero to five computed by ICObench, using a weighted average of two distinct evaluations. The first is computed by a proprietary assessment algorithm which considers the team composition, ICO information, product presentation, the marketing campaign and the presence on & social media. The second is assigned by experts who evaluate from 1 to 5 the ICO for team, vision, and product (as described in <https://icobench.com/ratings>). Actually, we found that ICOs Ratings vary from 0.4 to 4.9.

On average, ICOs have a Rating equal to 3.1.

Agile ICOs have, on average, a better Rating score. The average Rating value of the 55 ICOs is 3.6. Considering the diversity of samples analyzed—the first with 1646 elements, the second with 55 elements—we performed a statistical analysis on the significance of the differences. We first verified that the values of the ICO rating do not follow a normal distribution.

This distribution is not normal. The Anderson-Darling normality produces  $p$ -value  $< 2.2 \times 10^{-16}$  lower than  $\alpha = 0.05$ . We then performed the Wilcoxon test and by its result we can see that the differences are significant. The  $p$ -value  $= 2.476 \times 10^{-5}$  is lower than the threshold value  $\alpha = 0.05$ . This result is strongly contrary to the null hypothesis and then the observed data are statistically significant. The minimum value of Agile ICOs is 2.1.

#### 4.4. Social Media and Tools

We analyzed the social media use in order to understand how ICOs use this channel to communicate with investors and customers. 1810 ICOs out of 1952 use at least one social media. 1769 ICOs have at least one Twitter account, 1528 have a Facebook page. Telegram is used by 1231 ICOs, Youtube by 1112, Medium by 1069, Reddit by 812, GitHub by 796, Slack by 555, and Discord by 46 ICOs. All Agile ICOs communicate with investors and customers by means of social media. All Agile ICOs in fact have a Twitter account and 51 out of 55 have at least one Facebook page. Telegram is used by 42 Agile ICOs, Youtube by 39, Medium by 38, Reddit by 32, GitHub by 32, Slack by 14, and Discord by 3 ICOs. Given the decentralized nature of ICOs, the developers have to create a strong and active virtual community to support projects. As reported in [39], social media became part of the standard communication tools in recent years. Telegram groups and Slack channels are used as tools to which interested parties can ask questions about the ICO. On the other hand, social networks are used by the team to share information and news, and to raise awareness on their cryptocurrency and ICO. According to [40], the communities of cryptocurrency users require transparent and reactive communication. According to Agile Methods, a software product is a constantly evolving project in which the initial idea could be modified and adapted, using the feedback provided continuously by users. External feedback at each stage, allows the development of true competitive value of the product and services offered, promoting quality, efficiency, and trust. Given the decentralized nature of the blockchain, social media are the main channels, for an ICO, to communicate with users and investors. It is therefore not surprising that the use of social media is greater in Agile ICOs than in non-Agile ones because they can enhance contacts and interactions between customers and the ICO's team, simulating the *customer on site* prescription.

#### 4.5. Financial Aspects

A token is a digital asset that, in addition to having an exchange value, has an intrinsic value that derives from its use. ICObench describes the way in which the tokens are sold. The parameters reported on the website are: The number of tokens for sale, the percentage of token to be sold during the ICO, the hard and the soft capitalization, namely the goal of the ICO offer expressed in a reference currency, and the minimum selling target to be reached to develop the product. The number of ICOs which provide this parameter is 1053. We discovered that over 60% of such ICOs provides more than 50% of their tokens to investors. The remaining tokens are managed by the team. In particular, about 44% of ICOs choose to distribute more than 60% of its tokens to investors during the crowdfunding. This set is not characterized by a normal distribution of values. Applying the Anderson-Darling normality test we obtained a  $p$ -value equal to  $6.404 \times 10^{-12}$ , lower than the threshold  $\alpha$  equal to 0.05.

Agile ICOs differ with respect to the general statistics. The 38 out of 55 Agile ICOs which provide this parameter, on average distribute a lower percentage of tokens to the investors during the crowdfunding. In particular, only 50% of ICOs distribute more than 50% of tokens during the crowdfunding, and only 23% of Agile ICOs assign more than 60% of tokens to investors. The remaining tokens are managed by the ICO team. According to the results of the Anderson-Darling normality test, we can consider these values as sample of a normal distribution of values ( $p$ -value = 0.06406 > 0.05). The result of the Wilcoxon test allows us to consider significant the differences between the two sets of data. The  $p$ -value is equal to 0.0262 and is lower than 0.05. So the null hypothesis is rejected, and the two samples can be considered as taken from different populations. Figures 2 and 3 show the histograms of the number of ICOs per percentage of Token to be sold during the ICO for All ICOs and for Agile ICOs.

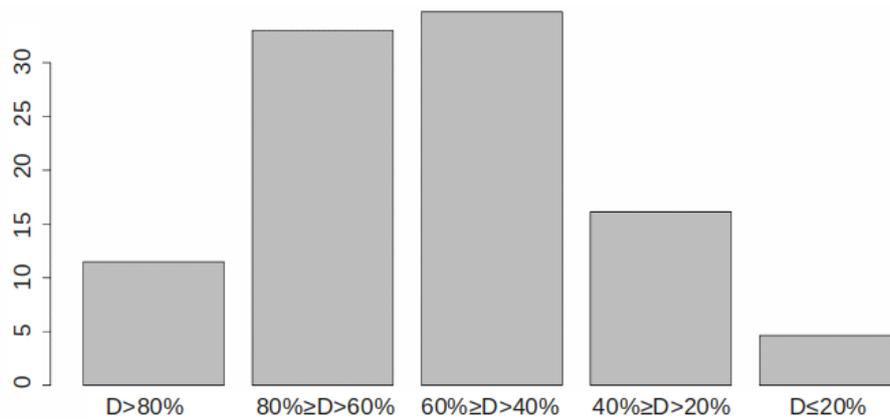


Figure 2. Histogram of the number of ICOs per percentage of Token to be sold during the ICO.

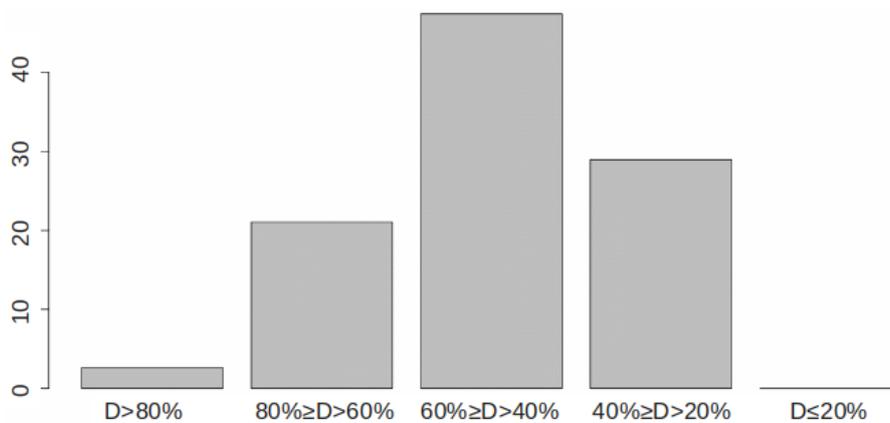


Figure 3. Histogram of the number of Agile ICOs per percentage of Token to be sold during the ICO.

### Ico Market Capitalization

The market cap is the value of tokens expressed in a reference currency. An ICO, through its *hard cap* (hard capitalization), sets the limit of how much money will be accepted to finance the product development. The excess money received is returned to the investors. The *soft cap* (soft capitalization) is the minimum amount required by the project in order to continue its development. If this amount is not reached, investors can withdraw their contribution. A crowdsale that reaches the soft cap is considered successful.

We found that 469 ICOs provide both the hard cap and the soft cap. Among these, the soft cap is on average 19% of the hard cap, so a crowdsale that reaches 19% of the hard cap is considered successful. Analyzing in particular Agile ICOs, 25 out 55 provide both hard cap and soft cap. In this case, on average, the soft cap is 25% of the hard cap, so in Agile ICOs only a crowdsale that reaches 25% of the hard cap is considered successful. Therefore, Agile ICOs needs a initial capital proportionally higher than the non-Agile ICOs to develop the project. No Agile IC reached 100% of the hard cap (the maximum percentage is 85%).

In this case, the differences found between Agile ICOs and other ICOs are not significant. The results of the Wilcoxon test suggest that the two set of data can be considered as elements of the same population. The resulting *p*-value is equal to 0.05878, greater than the threshold value,  $\alpha$ , equal to 0.05, and therefore the null hypothesis can be considered valid.

The distribution of the ratio (in percentage values) between *softcap* and *hardcap* is shown in Figures 4 and 5.

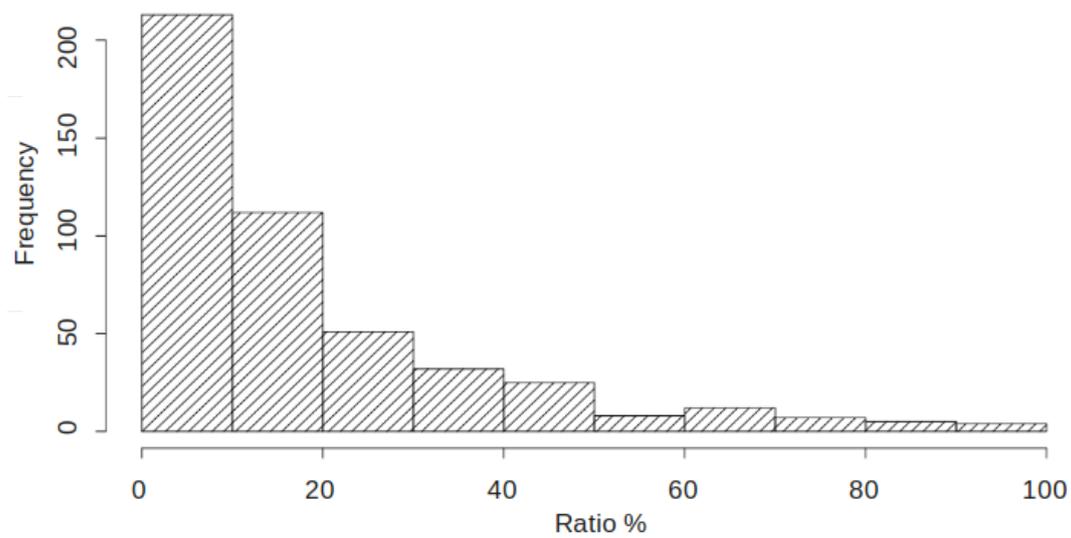


Figure 4. Histogram of the percentage of the Hard Cap per ICO to be reached to consider the ICO as a success.

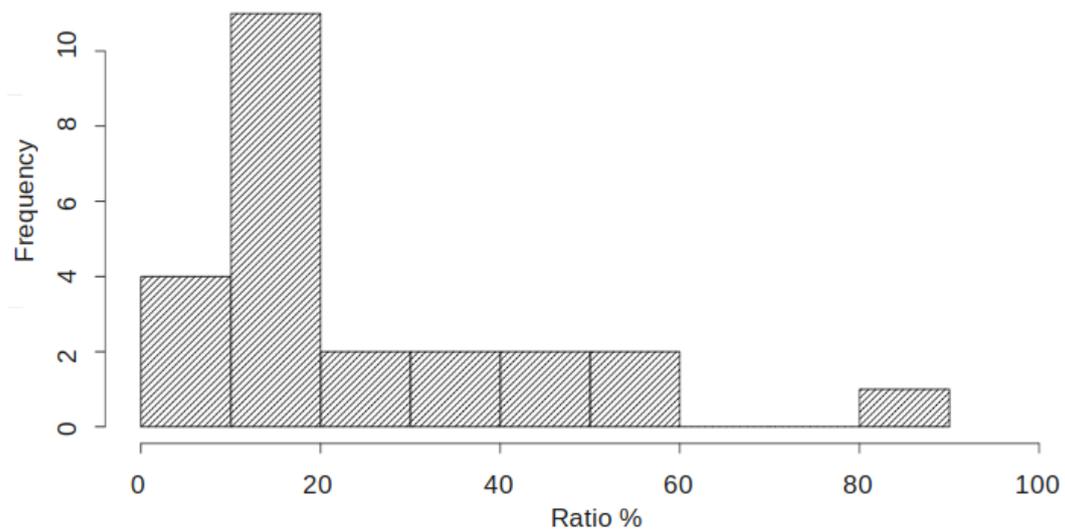


Figure 5. Histogram of the percentage of the Hard Cap per Agile ICO to be reached to consider the Agile ICO as a success.

### 5. Analysis of Agile ICOs Projects

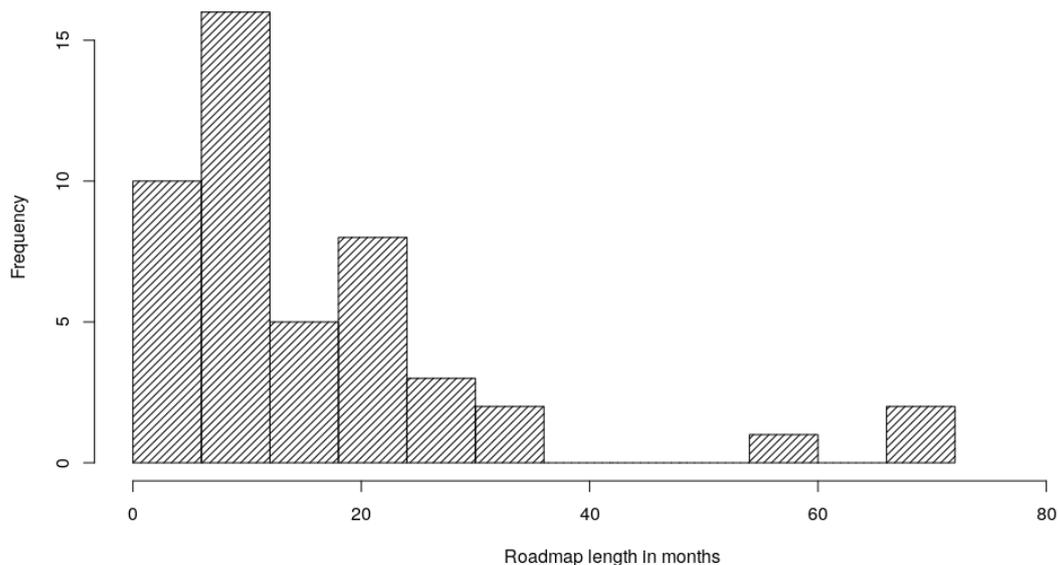
In this section we focus on the analysis of software projects of Agile ICOs. We studied in particular two aspects: The first is the *ICO roadmap*, the typical step by step description with which the ICO proposers declare their development program of the proposed product or service. The second is the software development project, that is the available repository of the ICO source files. This second aspect includes the analysis of development tools and testing practices used by the developers of the ICO.

#### 5.1. Roadmap and ICO State

As we said in section Data Analysis, a team describes, usually in the white paper, the roadmap of activities after the crowdfunding, and outlines the actions which they aim to achieve during the product development. Generally, this description is a simple graphical overview of the project’s goals and deliverables, and of the related timeline. Investors look the roadmap to know when and how the business idea will be operative and profitable, to understand the development phases of the product. We chose to identify the starting-point of a roadmap as the time when the ICO crowdfunding closes, and

the team gathers the money needed to develop the product. In the subset of 55 Agile ICOs, only 9 ICOs don't provide a roadmap. In the remaining 46 ICOs the roadmap is also called milestone, timeline or highlights. In these, the time period that roadmaps cover, ranges from few month to over five years, as shown in Figure 6 and summarized in the following:

- only 18 ICOs present a roadmap longer than one year;
- the longest roadmap extends 72 months after the end of the ICO;
- the average duration of roadmaps is 16 months after the ICO end;
- in 2 cases, the roadmap is concluded with the closing of the ICO.



**Figure 6.** Histogram of the number of months of development described in the Agile ICO roadmaps.

We define the time of post-ico state as the number of months passed after the ICO period. Figure 7 shows the distribution of the state of ICOs in terms of number of months passed after the end of the ICO selling phase.

The roadmaps therefore concern the future of the ICO and provide a realistic plan on the use of the funds in view of the objectives set. The content of the roadmap helps investors to understand when and how they are involved in the project, and provides them a view on possible changes. A too detailed roadmap is typical of the plan-driven methodologies and contradicts one of the fundamental principles of Agile methodologies, that aim to respond to change more than to follow a predetermined plan [41]. In a roadmap developed with an Agile approach, the possible difficulties that the team can meet and the way to deal within possible obstacles should be taken into consideration. A roadmap developed with an Agile approach should therefore be compared not only to the future features of a project, that is difficult to accurately detail and predict, but should also show the daily work and progress of the team, with a special focus on feedback and opinions of the people involved. The roadmap is therefore an useful tool to promote the transparency of development, also in order to manage customer expectations. Focusing on specific features diverts attention from the general vision of the project. An Agile roadmap is therefore able to embrace the inevitable changes, to communicate a short-term plan, but it must also include a flexibility that allows this plan to be adjusted to the customer's value or changes imposed by the market. According to [42,43] we define below some guidelines that characterize a roadmap designed with an Agile approach.

- The roadmap must be oriented towards objectives much more than towards the features be developed, so that everyone involved can understand the evolution of the product.

- The creation of Agile roadmaps requires continuous communication within the team, and with investors. It also needs to harness the effort of all the parts involved.
- In an ICO, it is essential to respond adequately to the needs of investors. Responding promptly to customers' needs through continuous dialogue is one of the essential characteristics of the Agile approach. The roadmap should therefore take into account all investors' feedback for possible improvements. The new ideas, evaluated through a score, must be included in a future release backlog. The ideas of investors and customers should therefore guide the definition of future priorities.
- The roadmap should be changed quite frequently (from one to three months) in order to adapt the plans with the obtained feedback. Updating the roadmap can help a project to face changes without diverting attention from long-term goals.

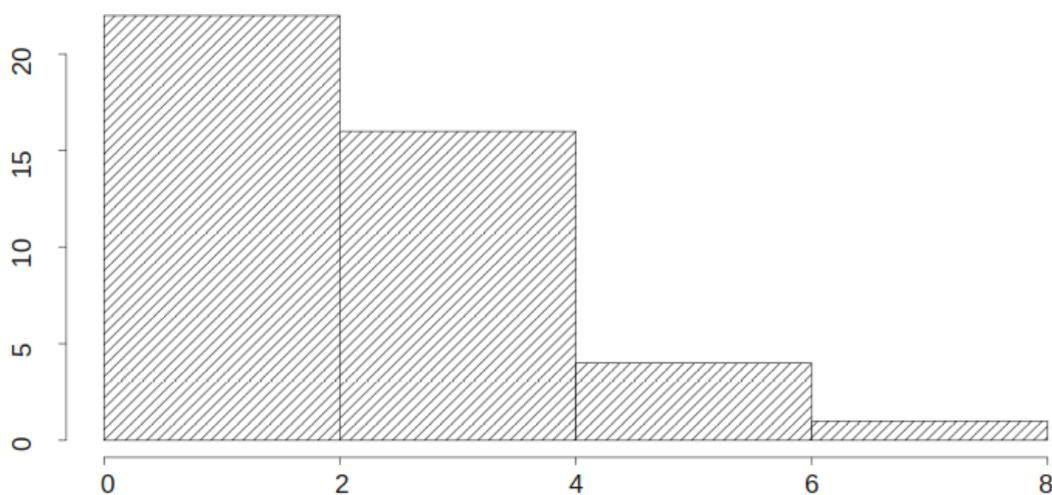


Figure 7. Histogram of the number of months passed after the end of the ICO.

## 5.2. Software Development

In order to better understand the process of software development of ICOs, for each of the 55 ICO projects selected in step 2 we examined the documentation to find the availability of software project repositories in which the development team stores and manages the software under development. We found that:

- 36 ICOs have a software project publicly available on the Github platform.
- 12 of the remaining ICOs published in the Ethereum blockchain explorer *Etherscan* (<https://etherscan.io/>) the solidity code of the smart contract used to implement the token selling.
- 7 ICOs do not have a publicly available software project nor provide any smart contract solidity file.

Summarizing, 48 out of 55 Agile ICOs provide at least the smart contracts used to develop the token of the ICO. These smart contracts are written in Solidity [44], which is the most popular high-level language for implementing smart contracts.

We analyzed the 32 Agile ICOs software projects available on Github. In particular, we counted the number of *repositories* (the folders of the project), the typology of files, and the number of solidity files.

In summary, the 32 Github projects contain a total of 14,199 files. The total number of folders is about 2800. On average, each project contains 5.8 *repositories*, and the maximum number of *repositories* per project is 38.

Regarding the contents, source code files represents the most of the files present in ICO software projects. In particular, *js* files (javascript) dominate the scene with 5,015 files, equal to 35.32% of the total. The projects contain 786 smart contracts written in solidity (.sol), equal to 5.54% of the total.

Graphics file formats (like png and svg formats) represent the second most frequent kind of files. Table 2 summarizes the number of occurrences of the ten most common file types found in Agile ICO software projects.

To better understand the process underlying the development of Agile ICOs, we have verified for each project the use of specific development frameworks and in particular the use of *Truffle*. Truffle is a popular development framework for Ethereum that includes built-in smart contract compilation, linking, deployment, binary management, and automated testing. Truffle is available at <https://truffleframework.com>. We found that Truffle is commonly used in Agile ICOs. Nineteen out of 36 projects include the typical Truffle elements.

**Table 2.** The ten most common file extensions in Agile ICO projects.

Extension	# Files	Percentage
js	5015	35.32%
png	1043	7.35%
sol	786	5.54%
cs	680	4.79%
md	594	4.18%
hpp	533	3.75%
json	485	3.42%
cpp	467	3.29%
go	436	3.07%
svg	418	2.94%
others	3742	26.35%

### 5.2.1. Smart Contract Code Metrics

In this section, we report the results of the analysis of the smart contracts used to implement the Agile ICOs. In particular, in order to characterize the content of these solidity files (with extension *.sol*), we applied a selection of code metrics. We also report the comparison between our results with the results provided by [45], related to more than 12,000 smart contracts published on the blockchain explorer Etherscan and deployed on the Ethereum blockchain until January 2018.

In our analysis, we examined 502 solidity files found in Agile ICOs github projects, which could be referred directly to the ICO developers. We excluded from our analysis the files copied (or *forked*) from other projects (including templates taken from development framework like *Openzeppelin* (<https://openzeppelin.org/>)).

We added to the analysis the 12 smart contracts published only in Etherscan, as described before. In total, we examined 514 smart contract files. For each of them we applied the software metrics defined in Table 3, that include volume metrics and complexity metrics. In the following, we will use the term *contract* to refer to a specific type of object of the Solidity language [44]. The declaration of an object *contract* is similar to the declaration of a *class* of object oriented languages. In a contract definition, it is possible declare functions and variables, that can be modified to be private, public, or internal to the contract. A solidity contract can also inherit from other contracts. We considered “function” the declaration, through the keyword *function*, of the executable units of code within a contract. We did not considered as functions the definition of *function modifiers*. In solidity, a modifier is a short portions of code defined through the keyword *modifier* that can be called and incorporated in functions. They can be used to easily change the behavior of functions.

Table 4 reports the summary of the statistics of the computed volume source code metrics. In the table, we also provide, for each metric, the value of the First Quartile that separates the lowest 25% of values from the highest 75%, and the Third Quartile that separates the lowest 75% of values from the highest 25%. All the reported statistics represent an overview of the distribution of metrics values. All statistical analysis were performed using R. Results show that smart contracts in Agile ICO projects have on average 65.59 lines of code. The maximum number of LOC is 808 and the minimum is 2.

For comparison, results of [45] report that the mean number of LOC is 183.8. We found that smart contracts present in Agile projects are characterized by a lower number of LOC.

**Table 3.** Definition of computed source code metrics.

LOC	Lines of Code: Number of lines containing executable code
CPL	Comment per Line: Number of lines containing comments divided by the number of lines of code
NDC	Number of Declared Contracts in the solidity code file
NDF	Number of Declared Functions in the solidity code file
FPC	Functions Per Contract (number of functions in the source code file divided by the number of contracts)
AFL	Average function length in a solidity code file, in terms of lines of code
Acyclo	Average McCabe Cyclomatic Complexity of the functions among a solidity file
Mcyclo	The maximum McCabe Cyclomatic Complexity computed among all functions of the solidity file
SumCyclo	Sum of the single McCabe Cyclomatic for all functions complexity of the solidity file.

**Table 4.** Volume metrics of smart contracts belonging to Agile ICO projects.

Metric	LOC	CPL	NDC	NDF	FPC	AFL
Mean	65.59	0.453	1.961	6.661	4.291	7.785
Median	28	0.317	1	3	3	5.667
St. Dev	88.47	0.48	4.87	8.96	4.50	12.19
Max.	808	2.727	66	71	28	172.5
Min.	2	0	1	0	0	1
1st Qu.	16	0.067	1	2	2	3
3rd Qu.	81.75	0.717	1	8	5	9

The mean and the median value of CPL allows us to state that examined files are well commented. As reported, there is expected about one line of comments every two lines of code.

The examined solidity files declare, on average, only two contracts. This number can be considered low, in relation to the value of 9.2 reported by [45]. The maximum number of contract declarations is 66 and the minimum is 1.

In total, the mean number of functions (NDF) declared in a contract is about 6.6, the maximum value is 71, and the minimum is zero. Also in this case, the mean number is lower than the results of [45] (25.9 functions per file). The mean values of NDC and NDF can be considered related to the value of the LOC metric. Solidity files are shorter and consequently less functions and contracts are declared. Contracts having no function declarations can be used to define variables and data structures to be inherited by other contracts. In general, if there is no constructor, the contract will assume the default constructor.

The metric Function Per Contract (FPC) represents the equivalent of the number of *method per class* in object oriented languages. Considering each contract in the files, we found that, on average, each of them declare 4.3 functions. The maximum number of declared functions per contract is 28 and the minimum is zero. On average, each contract has functions that are long 7.8 lines (AFL). The maximum average length is 172.5. The related distribution is characterized by a high standard deviation.

From these results, we can deduce that the smart contracts of Agile ICOs tend to be short programs, with a limited number of elements (contracts and functions) and with short functions. This favors easier reuse and maintenance of the code, according to [46].

In general, the development of the ICO token pass through the implementation of a standard interface called ERC20 (specifications described in [https://theethereum.wiki/w/index.php/ERC20\\_](https://theethereum.wiki/w/index.php/ERC20_)

[Token\\_Standard](#)). Given the availability of already implemented ERC20 tokens, the reuse of code is commonly adopted during the creation of new tokens. The high values of standard deviation show that smart contracts are very different from each other; these results are typical of long-tail distributions, whose tails collect the highest values.

For each solidity file, we computed the McCabe cyclomatic complexity [47], of all the functions implemented in it. The cyclomatic complexity measures the number of linearly independent paths through a function. We used a commercial software. In particular, we computed the cyclomatic metrics using *Understand*, by *scitools*. Cyclomatic metrics are described in <https://scitools.com/support/cyclomatic-complexity/>. Table 5 summarizes the results related to the average, the maximum and the sum of the cyclomatic complexity of all the functions defined in each solidity file belonging to Agile ICO projects. The minimum values of these metrics are equal to zero due to the presence of contracts that do not implement any function.

**Table 5.** Cyclomatic metrics computed in the solidity files belonging to Agile ICO projects.

Metric	ACyclo	MaxCyclo	SumCyclo
Mean	1.21	1.833	7.969
Median	1	1	4
St. Dev	0.59	1.70	12.27
Min.	0.00	0	0
Max.	7	17	134
1st Qu.	1	1	2
3rd Qu.	1	2	9

We found that the average cyclomatic complexity (*ACyclo*) has a value equal to 1.2. The maximum value of the average cyclomatic complexity is 7.

The maximum cyclomatic complexity (*MaxCyclo*) is, for each contract, the the maximum value of McCabe cyclomatic complexity among the functions of the contract. Its mean value is 1.83, and its highest value is equal to 17.

Contracts are characterized by a limited sum of the cyclomatic complexity (*SumCyclo*) computed for each function in their solidity files. The mean value of the sum is 7.97, lower than the value reported by [45], due to the fact that contracts of Agile ICO projects are shorter in terms of LOC. Values of this metric are characterized by a standard deviation equal to 12.27, and a maximum value equal to 134. These value are typical of long tail distributions.

### 5.2.2. Testing

In the Ethereum platform, each smart contract deployed in the blockchain both the data related to the transactions, and the code that implements the logic to allow the sending of transactions between two or more actors. Therefore, data and logic that compose a smart contract are stored irreversibly. Given the principle of the immutability of the blockchain, once a smart contract gets deployed, its code cannot be changed.

If a developer finds a bug or wants to correct an error, s/he has to develop a new smart contract, deploy it on the blockchain and transfer all the existing data to the new contract. The deployment of a smart contract includes an Ethereum transaction that requires the payment of a fee, which depends on the size of the smart contract. For this reason, the test phase before the deployment is very important and it should be managed appropriately also through the adoption of best practices and specific tools for continuous testing, typical of Agile methodologies.

We, therefore, analyzed the use of development tools and practices for smart contracts. In particular, we look for the use of the Truffle suite for testing practices. We found that Truffle is commonly used in Agile ICOs. Nineteen out of 36 projects include the typical Truffle elements (the file *truffle.js* and the directories *build*, *contracts*, and *migration*). Using Truffle, developers can take advantage of several development tools included in this suite. One of the most relevant is the

possibility to create a Test suite. Tests can be written both in solidity and in javascript. We found that 16 of the 19 Agile ICO projects that use Truffle include a test suite. In addition, we found the presence of other kinds of testing code developed to test other components of the project. The use of Truffle, which provides an automated testing framework to test smart contract before the deployment on blockchain, is fully consistent with the application of Agile methodologies.

## 6. Discussion

The results of this study provide an overview of the world of ICOs, which can be described as a new blockchain based fundraising mechanism in which a startup sells its tokens in exchange for Bitcoins or Ethers. The ICOs, therefore, offer important new possibilities related to the intrinsic properties of blockchain technology. In order to make the most of this potential, in such recent and innovative context, it is necessary to employ appropriate software development methodologies. We believe that the Agile methodologies, which are suitable for the management of innovative startups because they allow to easily face the changes, can also be useful when applied to the ICOs. Agile methods are suited to develop system whose requirements are not completely understood, or tend to change. These characteristics are present in ICOs because they are typically very innovative applications and often there is a run to launch an ICO to be the first on the market. An Agile approach, is based on iterative and incremental development with short iterations, and is suited to deliver quickly and to deliver often. In addition, Agile methodologies are suited for small, self-organizing teams working together, as is the case for many ICO teams.

Table 6 summarizes the results of the comparison between the full set of ICOs and the Agile ICOs performed in our analysis.

**Table 6.** Comparison between ICOs and Agile ICOs.

Parameter	ICOs	Agile ICOs
Number of examined ICOs	1952	55
Average team size	11.4	14.9
Max team size	67	43
Average percentage of advisors in teams	18.90%	23.04%
Average percentage of women in teams	16.30%	15%
Average ICObench Rating	3.1	3.6
Minimum ICObench Rating	0.4	2.7
Percentage of ICOs present in social media	92.7%	100%
Percentage of ICOs with a Twitter profile	90.1%	100%
Percentage of ICOs with a Facebook page	81.6%	92.7%
Percentage of ICOs using an GitHub account	40.8%	58.1%
Average percentage of token used for crowdfunding	60%	50%
Average ICOs success threshold (in terms of percentage of the hard capitalization)	19% of HC	25% of HC

We found that in Agile ICOs the team is on average made up of 14.9 people, of which 23.04% on average are advisors. This means that the development team is on average made up of 10 people. Among them, we include 6 or 7 developers, but also other professionals, like testers or UI designers. In SCRUM methodology, for example, a development team consists of 3-9 people. According to [48], for larger projects, Scrum provides a mechanism called Scrum of Scrums that assign the large project to several teams. ICOs are developed using technologies like the ERC20 Token Standard, a particular smart contract interface, written in the new programming language named Solidity, which abstracts many development process useful to set up a new cryptographic asset. This abstraction can be related to the Agile practice named Coding Standards, as described by [11]. Moreover, the practice of Collective code ownership is also employed in ICOs, due to the transparency of smart contracts in the blockchain. Most of the ICOs' smart contracts are public and readable by everyone. On the other hand, we need to take into account some important factors inherent in decentralized technologies

that need to be carefully controlled in applying Agile methodologies. An ICO is based on the use of smart contracts that record data transactions and manage the specific functionalities of the project. The data stored on the blockchain are immutable, so, in order to modify a smart contract we have to deploy a new smart contract. Agile development, on the contrary, insists on continuous feedback from users to foster innovation and it also strongly highlights the necessity to continually iterate on the product. This is the part of the process that can create issues for blockchain products. As previously mentioned, the data stored on the blockchain are immutable. The smart contracts, once deployed on the blockchain, cannot be updated except by uploading a new smart contract. Developers therefore must be sure that the smart contract will always work. It must be right the first time it is deployed, and this challenges the present Agile wisdom. Before storing a smart contract on the blockchain, it is therefore important to perform a very thorough testing, auditing and verification phase, also through the use of tools such as Truffle, a tool to build smart contracts that provides an automated test framework and allows writing tests in Javascript and Solidity. The practice of software development called Test Driven Development (TDD) described by [49], used in Agile methodologies and especially in Extreme Programming according to [50], which consists of writing the tests before the actual implementation of the functionalities, could therefore be of great help to develop smart contracts of the highest quality. These contracts will be able to be deployed in the blockchain, guaranteeing that the need to change them will not arise. Another aspect that must be taken into great consideration is the one related to project planning. The planning of an agile project is based on the split of functionalities in user stories, which are fragments of functionalities that give value to the user. The analysis of the requirements leads to the definition of a certain number of stories, each having an intrinsic value to the project. A user story must be measurable, and must be developed in a limited time. A roadmap developed with an Agile approach should therefore pay particular attention to the feedback and wishes of all involved stakeholders. The roadmap is therefore a useful tool to promote the transparency of the development, also in order to manage customer expectations. It must not be too detailed, and must allow the implementation of the project as a set of user stories or features. The implementation flow could take place in Sprints, and each user story should be provided with one or more Acceptance Tests, according to [51].

Regarding the quality of the code, the results obtained by us are not consistent with those found by [52], that made a comparison between software metrics in a software system (written in Python and in Java) and developed using Agile methodologies and in systems developed using plan-driven methodologies. They assert that metrics distribution generated from Agile methodologies are not related to better quality of software. Unlike ours for example they found that the LOC distribution does not demonstrate major differences. In our analysis smart contracts present in Agile projects are characterized by a lower number of LOC in comparison with ICOs that not use Agile methodologies. In general all smart contracts tend to be short with a limited number of contracts and functions. It is also connected with the fact that to deploy a file on the blockchain it is necessary to pay a fee proportional to the size of the file. The simplicity of the code is perfectly consistent with one of the five fundamental values of the XP programming, reported by [50]. Moreover in order to improve our study for the future work, we summarize below the main limitations of our study. The first issue is related to the novelty of our work. The typology of analysis we made is therefore pioneering. We focused part of our study on the collection of Agile Keywords inside the white paper, in order to recognize which kind of ICOs funded project used Agile practices. We therefore did not analyze ICOs in which no agile keywords are officially stated in the white paper, but the development could be anyway Agile based. We are indeed aware that not all ICOs they may have declared within its own documentation Agile keywords even if they use agile Methods for developing projects.

We collected 55 Agile ICOs and we calculated the software metrics only on the smart contracts of these ICOs (for a total of 514 smart contracts). We made a comparison between this subset and the code metrics related to all smart contracts stored on Ethereum blockchain in 2017 (around 12,000 smart

contracts). Our subset may be small but the results obtained can serve for a future generalization considering a consistent dataset.

## 7. Conclusions

The present work proposed an analysis of Initial Coin Offerings, a new phenomenon that has recently become a relevant topic of study within the blockchain community. In order to better understand this phenomenon, operating in an uncertain and constantly changing context, we investigated all engineering activities related to ICOs, from the planning phase to the testing phase. We analyzed the whole set of ICOs registered in ICObench until the 20 February 2018 in order to discover the team composition, the communication channels with investors distributed around the world and the financial aspects. We therefore studied the use of Agile practices in ICOs as a method to cope with change. We have selected and analyzed in detail a subset of ICOs specifically developed with Agile methodologies relatively to their roadmap, project development, and source code quality. Overall, about the 5% of the examined ICOs apply Agile practices. In addition we conducted an analysis of smart contracts of Agile ICOs in terms of code metrics, language versions, and use of test tools. We discovered that the Agile methodologies are suited to develop ICOs because these are highly innovative projects, whose requirements are not completely understood or tend to change. The Agile approach is iterative and incremental with short iterations and is suited to deliver quickly and often. This property is very useful in the context of ICOs. On the other hand the necessity to continually iterate on the product, typical of Agile methodologies, can create issues for ICOs. The immutability of the blockchain must be taken into consideration, smart contracts can not be updated once they have been loaded. To face this difficulty the Test Driven Development is very useful. The practice of Collective code ownership is also guaranteed in ICOs by the transparency of smart contracts in the blockchain. Another practice of Agile development that are applied in ICOs is the use of Coding Standards. The too detailed roadmaps are instead typical of the plan-driven methodologies. Finally, we can say that the smart contracts of Agile ICOs have good metrics of the software because their files are very short and simple.

*Future Work.* In the next iteration of this empirical study we want examine more aspects of ICOs to recognize the use of Agile practices in a more comprehensive way taking into consideration an analysis related to each single Agile methods. In the future will be possible use the results of this work as a snapshot dated February 2018 of ICOs characteristics, and use it for a time-based comparison focusing mainly on the rate of adoption of Agile practices. In addition, a future work should provide a correlation analysis between the usage of Agile practices and the ICOs success, both in terms of capitalization and in terms of the development of their projects.

**Author Contributions:** S.I. and A.P. conceived and designed the experiments, prepared the figures and tables, and drafted the work; all authors analyzed the data, revised the manuscript draft and approved the final draft; S.I. conceived the concept search.

**Funding:** This research is partially supported by the Research Project Fondazione di Sardegna "Algorithms for Approximation with Applications [Acube]", annuity 2017; and by the Research Project "AIND—Amministrazioni e Imprese Native Digitali"—Programmazione Unitaria 2007–2013—P.O. FESR 2007/2013—Interventi a sostegno della competitività e dell'innovazione, annuity 2013.

**Acknowledgments:** The authors would like to thank the team of [icobench.io](https://icobench.io) that provide the API from which we extracted our ICOs dataset.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. Available online: <https://bitcoin.org/en/bitcoin-paper> (accessed on 20 October 2018).
2. Swan, M. *Blockchain: Blueprint for a New Economy*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2015.

3. Kosba, A.; Miller, A.; Shi, E.; Wen, Z.; Papamanthou, C. Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. In Proceedings of the 2016 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 22–26 May 2016; pp. 839–858.
4. Buterin, V. A nExt-Generation Smart Contract and Decentralized Application Platform. Available online: [http://blockchainlab.com/pdf/Ethereum\\_white\\_paper-a\\_next\\_generation\\_smart\\_contract\\_and\\_decentralized\\_application\\_platform-vitalik-buterin.pdf](http://blockchainlab.com/pdf/Ethereum_white_paper-a_next_generation_smart_contract_and_decentralized_application_platform-vitalik-buterin.pdf) (accessed on 20 October 2018).
5. Coleman, G.; O'Connor, R.V. An investigation into software development process formation in software start-ups. *J. Enterp. Inf. Manag.* **2008**, *21*, 633–648. [CrossRef]
6. Nerur, S.; Mahapatra, R.; Mangalaraj, G. Challenges of migrating to agile methodologies. *Commun. ACM* **2005**, *48*, 72–78. [CrossRef]
7. Highsmith, J.A.; Highsmith, J. *Agile Software Development Ecosystems*; Addison-Wesley Professional: Boston, MA, USA, 2002; Volume 13.
8. Highsmith, J.; Cockburn, A. Agile software development: The business of innovation. *Computer* **2001**, *34*, 120–127. [CrossRef]
9. Luu, L.; Chu, D.H.; Olickel, H.; Saxena, P.; Hobor, A. Making smart contracts smarter. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 24–28 October 2016; pp. 254–269.
10. Yli-Huumo, J.; Ko, D.; Choi, S.; Park, S.; Smolander, K. Where is current research on blockchain technology? A systematic review. *PLoS ONE* **2016**, *11*, e0163477. [CrossRef] [PubMed]
11. Beck, K.; Beedle, M.; Van Bennekum, A.; Cockburn, A.; Cunningham, W.; Fowler, M.; Grenning, J.; Highsmith, J.; Hunt, A.; Jeffries, R.; et al. Manifesto for Agile Software Development. Available online: <http://agilemanifesto.org/> (accessed on 20 October 2018).
12. Blotner, J.A. Agile techniques to avoid firefighting at a start-up. In *OOPSLA 2002 Practitioners Reports*; ACM: New York, NY, USA, 2002; pp. 1–ff.
13. Pantuichina, J.; Mondini, M.; Khanna, D.; Wang, X.; Abrahamsson, P. Are software startups applying agile practices? The state of the practice from a large survey. In Proceedings of the International Conference on Agile Software Development, Cologne, Germany, 22–26 May 2017; Springer: Cham, Switzerland, 2017; pp. 167–183.
14. Giardino, C.; Paternoster, N.; Unterkalmsteiner, M.; Gorschek, T.; Abrahamsson, P. Software development in startup companies: the greenfield startup model. *IEEE Trans. Softw. Eng.* **2016**, *42*, 585–604. [CrossRef]
15. Paternoster, N.; Giardino, C.; Unterkalmsteiner, M.; Gorschek, T.; Abrahamsson, P. Software development in startup companies: A systematic mapping study. *Inf. Softw. Technol.* **2014**, *56*, 1200–1218. [CrossRef]
16. Ghezzi, A.; Cavallo, A. Agile Business Model Innovation in Digital Entrepreneurship: Lean Startup Approaches. *J. Bus. Res.* **2018**. [CrossRef]
17. Miski, A. Development of a mobile application using the lean startup methodology. *Int. J. Sci. Eng. Res.* **2014**, *5*, 1743–1748.
18. Porru, S.; Pinna, A.; Marchesi, M.; Tonelli, R. Blockchain-oriented Software Engineering: Challenges and New Directions. In Proceedings of the 39th International Conference on Software Engineering Companion (ICSE-C '17), Buenos Aires, Argentina, 20–28 May 2017; IEEE Press: Piscataway, NJ, USA, 2017; pp. 169–171.
19. Chen, Y. Blockchain tokens and the potential democratization of entrepreneurship and innovation. *Bus. Horiz.* **2018**, *61*, 567–575. [CrossRef]
20. Li, J.; Mann, W. Initial Coin Offering and Platform Building. Available online: <https://ssrn.com/abstract=3088726> (accessed on 20 October 2018).
21. Ibba, S.; Pinna, A.; Baralla, G.; Marchesi, M. ICOs Overview: Should Investors Choose an ICO Developed with the Lean Startup Methodology? In Proceedings of the International Conference on Agile Software Development, Porto, Portugal, 21–25 May 2018; Springer: Cham, Switzerland, 2018; pp. 293–308.
22. Fenu, G.; Marchesi, L.; Marchesi, M.; Tonelli, R. The ICO phenomenon and its relationships with ethereum smart contract environment. In Proceedings of the 2018 International Workshop on Blockchain Oriented Software Engineering (IWBOSE), Campobasso, Italy, 20 March 2018; pp. 26–32.
23. Adhami, S.; Giudici, G.; Martinazzi, S. Why do businesses go crypto? An empirical analysis of initial coin offerings. *J. Econ. Bus.* **2018**. [CrossRef]

24. Hartmann, F.; Wang, X.; Lunesu, M.I. Evaluation of initial cryptoasset offerings: the state of the practice. In Proceedings of the 2018 International Workshop on Blockchain Oriented Software Engineering (IWBOSE), Campobasso, Italy, 20 March 2018; pp. 33–39.
25. Demidenko, D.S.; Malevskaia-Malevich, E.D.; Dubolazova, Y.A. ISO as a real source of funding. Pricing issues. In Proceedings of the 2018 International Conference on Information Networking (ICOIN), Chiang Mai, Thailand, 10–12 January 2018; pp. 622–625.
26. Emtseva, S.; Morozov, N. Comparative Analysis of Legal Regulation of ICO in Selected Countries. *KnE Soc. Sci.* **2018**, *3*, 77–84.
27. Momtaz, P.P. Initial Coin Offerings; Available online: [https://www.hhfrc.de/fileadmin/user\\_upload/Website/Forschung/Working\\_Paper/HHFRC\\_No31.pdf](https://www.hhfrc.de/fileadmin/user_upload/Website/Forschung/Working_Paper/HHFRC_No31.pdf) (accessed on 20 October 2018).
28. Hahn, C.; Wons, A.; Hahn, C.; Wons, A. *Umsetzung des ICOs*; Springer Gabler: Wiesbaden, Germany, 2018.
29. Chod, J.; Lyandres, E. A Theory of ICOs: Diversification, Agency, and Information Asymmetry. Available online: <https://ssrn.com/abstract=3159528> (accessed on 20 October 2018).
30. R Core Team. *R: A Language and Environment for Statistical Computing*; R Foundation for Statistical Computing: Vienna, Austria, 2018.
31. Senapathi, M.; Srinivasan, A. Sustained agile usage: A systematic literature review. In Proceedings of the 17th International Conference on Evaluation and Assessment in Software Engineering, Porto de Galinhas, Brazil, 14–16 April 2013; ACM: New York, NY, USA, 2013; pp. 119–124.
32. Thode, H. *Testing For Normality*; Statistics, Textbooks and Monographs; Taylor & Francis: Oxford, UK, 2002.
33. Gehan, E.A. A generalized Wilcoxon test for comparing arbitrarily singly-censored samples. *Biometrika* **1965**, *52*, 203–224. [[CrossRef](#)] [[PubMed](#)]
34. Schwaber, K. *Agile Project Management with Scrum*; Microsoft Press: Redmond, WA, USA, 2004.
35. Berger, E.S.; Kuckertz, A. Female entrepreneurship in startup ecosystems worldwide. *J. Bus. Res.* **2016**, *69*, 5163–5168. [[CrossRef](#)]
36. Ortu, M.; Destefanis, G.; Counsell, S.; Swift, S.; Marchesi, M.; Tonelli, R. How Diverse Is Your Team? Investigating Gender and Nationality Diversity in GitHub Teams. Available online: <https://link.springer.com/article/10.1186/s40411-017-0044-y#citeas> (accessed on 20 October 2018).
37. William, M. Number of Women Eyeing Crypto Investing Doubled Since Start of Year. Available online: <https://cointelegraph.com/news/number-of-women-eyeing-crypto-investing-doubled-since-start-of-year> (accessed on 20 October 2018).
38. Przybylski, A.K.; Murayama, K.; DeHaan, C.R.; Gladwell, V. Motivational, emotional, and behavioral correlates of fear of missing out. *Comput. Hum. Behav.* **2013**, *29*, 1841–1848. [[CrossRef](#)]
39. Storey, M.A.; Singer, L.; Cleary, B.; Figueira Filho, F.; Zagalsky, A. The (r) evolution of social media in software engineering. In Proceedings of the on Future of Software Engineering, Hyderabad, India, 31 May–7 June 2014; ACM: New York, NY, USA, 2014; pp. 100–116.
40. Davis, J. Do You Have a Communications Strategy for Your Initial Coin Offering (ICO)? Available online: <http://www.coincatalyst.com/communications-strategy-for-initial-coin-offering-ico/> (accessed on 20 October 2018).
41. Cockburn, A.; Highsmith, J. Agile software development, the people factor. *Computer* **2001**, *34*, 131–133. [[CrossRef](#)]
42. Schuurman, R. Tips for Agile Product Roadmaps and Product Roadmap Examples. Available online: <https://www.scrum.org/resources/blog/tips-agile-product-roadmaps-product-roadmap-examples> (accessed on 20 October 2018).
43. Jury.Online Tech Development: Agile Roadmap. Available online: <https://medium.com/jury-online/jury-online-tech-development-agile-roadmap-30ecd9d93ab0> (accessed on 20 October 2018).
44. Solidity Documentation, Release 0.4.25, Ethereum. Available online: <https://github.com/ethereum/solidity/releases> (accessed on 20 October 2018).
45. Tonelli, R.; Destefanis, G.; Marchesi, M.; Ortu, M. Smart contracts software metrics: A first study. *arXiv* **2018**, arXiv:1802.01517.
46. Sant’Anna, C.; Garcia, A.; Chavez, C.; Lucena, C.; Von Staa, A. On the reuse and maintenance of aspect-oriented software: An assessment framework. In Proceedings of the Brazilian Symposium on Software Engineering, Manaus, Brazil, 1–4 October 2003; pp. 19–34.
47. McCabe, T.J. A complexity measure. *IEEE Trans. Softw. Eng.* **1976**, 308–320. [[CrossRef](#)]

48. Mundra, A.; Misra, S.; Dhawale, C.A. Practical scrum-scrum team: Way to produce successful and quality software. In Proceedings of the 2013 13th International Conference on Computational Science and Its Applications (ICCSA), Ho Chi Minh City, Vietnam, 24–27 June 2013; pp. 119–123.
49. Beck, K. *Test-Driven Development: By Example*; Addison-Wesley Professional: Boston, MA, USA, 2003.
50. Beck, K.; Gamma, E. *Extreme Programming Explained: Embrace Change*; Addison-Wesley: Boston, MA, USA, 2000.
51. Martin, R.C. *Agile Software Development: Principles, Patterns, and Practices*; Prentice Hall: Upper Saddle River, NJ, USA, 2002.
52. Destefanis, G.; Counsell, S.; Concas, G.; Tonelli, R. Software metrics in agile software: An empirical study. In Proceedings of the International Conference on Agile Software Development, Rome, Italy, 26–30 May 2014; Springer: Cham, Switzerland, 2014; pp. 157–170.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).