The Impact of Scrum on Customer Satisfaction: An Empirical Study

Bruno Cartaxo[1], Allan Araújo[1,2], Antonio Sá Barreto[1] and Sérgio Soares[1]
Informatics Center - CIn / Federal University of Pernambuco - UFPE[1]
Recife Center for Advanced Studies and Systems - C.E.S.A.R[2]
Recife, Pernambuco - Brazil
Email: {bfsc,arsa,acsbn,scbs}@cin.ufpe.br

***Abstract*—In the beginning of the last decade, agile methodologies emerged as a response to software development processes that were based on rigid approaches. In fact, the flexible characteristics of agile methods are expected to be suitable to the less-defined and uncertain nature of software development. However, many studies in this area lack empirical evaluation in order to provide more confident evidences about which contexts the claims are true. This paper reports an empirical study performed to analyze the impact of Scrum adoption on customer satisfaction as an external success perspective for software development projects in a software intensive organization. The study uses data from real-life projects executed in a major software intensive organization located in a nation wide software ecosystem. The empirical method applied was a cross-sectional survey using a sample of 19 real-life software development projects involving 156 developers. The survey aimed to determine whether there is any impact on customer satisfaction caused by the Scrum adoption. However, considering that sample, our results indicate that it was not possible to establish any evidence that using Scrum may help to achieve customer satisfaction and, consequently, increase the success rates in software projects, in contrary to general claims made by Scrum' advocates.**

I. Introduction

Since the term software engineering emerged in 1968 [**?**] it has motivated a tremendous amount of discussions, works, and research on processes, methods, techniques, and tools for supporting high-quality software development in a wide and industrial scale.

Initially, industrial work — based on manufacturing — introduced several contributions to the software engineering body of knowledge. Many software processes has been supported by industrial work concepts such as functional decomposition and localized labor [**?**]. During the last decades, techniques and tools has been created as an analogy to the production lines. The first generation of software processes family was based on the waterfall life cycle assuming that the software development life cycle was a linear and sequential similar to a production line [**?**]. Then, in the early 90's, other initiatives were responsible for creating iterative and incremental processes such as the Unified Process [**?**].

Despite these efforts and investments, software projects success rate has presented a dramatic situation in which less than 40% of projects achieve success (Figure **??**). Obviously, these numbers may not be compared to other profitable industries [**?**].

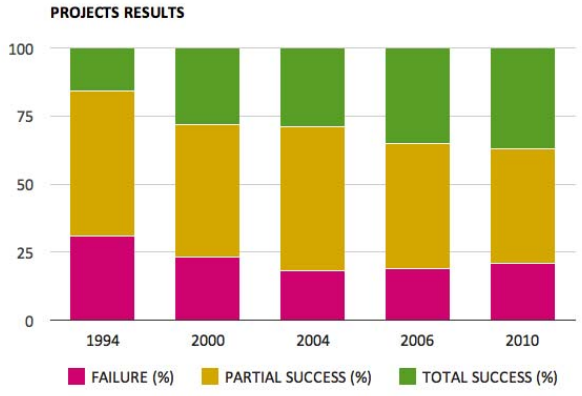

Fig. 1. 2011 Chaos Report - Extracted from [**?**]

Some specialists argue that software development is different from the traditional industrial work in respect to its nature. Software engineering may be described as knowledge work which is "focused on information and collaboration rather than manufacturing placing value on the ownership of knowledge and the ability to use that knowledge to create or improve goods and services" [**?**]. There are several differences between these two kinds of work. While the work is visible and stable in industrial work; it is invisible and changing in knowledge work. Considering that knowledge work (including software development) is more uncertain and less defined than the industrial work that is based on predictability, the application of industrial work techniques on knowledge work may lead to projects with increased failure rates.

Since 2001, agile methods have emerged as a response for overcoming the difficulties related to the software development. Some preliminary results shown that agile methodologies may increase success rates as shown in Figure **??** [**?**]:

Although some results may indicate that agile methodologies help to achieve success in software development, many of these researches fail to present evidence through empirical evaluation. Only through these evaluation it is possible to establish whether and in which context the proposed method or technique is efficient, effective, and can be applied [**?**] [**?**] [**?**]. In particular, for the agile context, a minor part of studies contains an empirical evaluation as shown in Figure **??** [**?**].

/13 $31.00 © 2013 IEEE
DOI 10.1109/SBES.2013.10

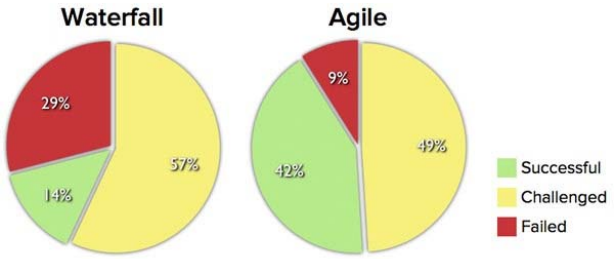

Fig. 2. Waterfall vs. Agile - Extracted from [**?**]

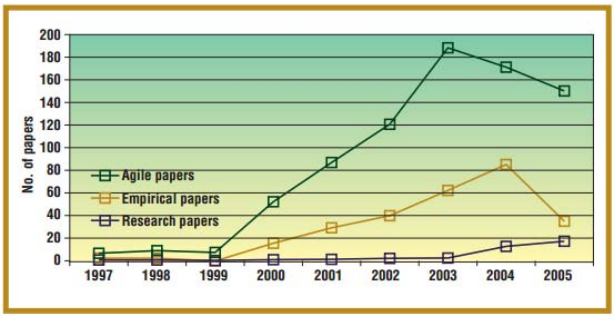

Fig. 3. Agile empirical evaluation rate - Extracted from [**?**]

Thus, the scope for this work was defined intending to provide a comparison between agile methods and traditional software development approaches. First, it necessary to point out that there are several agile methodologies such as Scrum, Extreme Programming (XP), Feature-Driven Development, Dynamic Systems Development Method (DSDM), Lean Software Development that are intended to support knowledge work (less defined and more uncertain) [**?**]. In parallel, it also exist many traditional approaches that are intended to support industrial work (more defined and less uncertain). These methods and processes are usually based on the remarkable frameworks such as PMBoK (Project Management Body of Knowledge) [**?**] and Unified Process [**?**]. These methods may include several perspectives such software engineering, project management, design and so on. For an objective analysis, it was chosen the project management perspective. On one hand, for agile methods, it was selected Scrum (project management based); on the other hand, it was chosen any traditional approach that include a perspective for the project management.

In this context, a survey was executed in the C.E.S.A.R (Recife Center for Advanced Studies and Systems) using a random sample containing 19 different projects adopting Scrum or any other traditional approach for managing the initiative involving 156 developers. The main expected contributions by this study are listed below:

- Increase the body of knowledge about Scrum and agile methods using a systematic approach through evidences within an industrial environment. In particular it is intended to reduce the lack of empirical evaluation in software development discussions.
- Help the organization to understand how to increase internal success rates by analyzing and discussing the results obtained from this research.

Hence, this paper is organized as following. Sections 2 and 3 present the definition for this study, including the conceptual model and the research method used for the survey, respectively. Section 4 is aimed to find out the results obtained from the survey execution. Limitations of this study as well as possible future studies are discussed at Section 5. Section 6 introduces some related studies and, finally, Section 7 presents the conclusion. Additionally, we present the applied questionnaire as well as the used likert anchoring scheme at the appendix.

II. Conceptual Model of Customer Satisfaction

The research model presented by this study verifies the impact of an independent variable (software development approach) on the project's success indexes considering the customer point of view. This independent variable may be assigned with two different values: Scrum and not Scrum (traditional approaches for software project management).

In particular, it is necessary to recognize that customers probably have different definitions for "success" within a software project. In order to establish an external perspective, the model assumes seven critical factors for customer satisfaction (dependent variables), and consequently, for project success: time, goals, quality, communication and transparency, agility, innovation and benchmark. The next subsections provide more details for each one.

A. Time

In general, "time to market" is a critical variable within a software project. Thus, we define a project as successful if agreed and negotiated deadlines are met. Since Scrum is based on small iterations, it is expected anticipated delivery of valuable software [**?**] and also short time-to-market. Hence, we argue that a software project in which Scrum is adopted is able to provide higher customer satisfaction rates regarding to the time constraints by meeting the agreed and negotiated deadlines. **Hypotheses 1:** *Scrum-based projects provide increased customer satisfaction from the time perspective.*

B. Goals

Software projects are launched for strategic purposes, such as costs reduction, legal compliance, market-share increase, etc. Thus, we define a project as successful if the goals that motivated the endeavor are met. Since Scrum considers a deeper and frequent stakeholder participation and collaboration, it is expected a continuous goals adjustment [**?**]. Hence, we argue that a software project in which Scrum is adopted is able to provide higher customer satisfaction rates by addressing the customer needs regarding to the defined goals within a project. **Hypotheses 2:** *Scrum-based projects provide increased customer satisfaction from the goals perspective.*

C. Quality

By definition, "quality is the degree to which a set of inherent characteristics fulfill requirements" [**?**]. Product and process quality depend on the software project criticality demanded by the customers. Thus, we define a project as a

successful if the required quality standards for that specific situation are met. So, regular inspections (one of the Scrum pillars) are one of most effective quality tools within a software development project [**?**]. Hence, we argue that a software project in which Scrum is adopted is able to provide higher customer satisfaction rates by addressing the customer needs regarding to the defined quality standards within a project. **Hypotheses 3:** *Scrum-based projects provide increased customer satisfaction from the quality perspective.*

D. Communication and Transparency

Software projects are expected to create intangible products under a dynamic and uncertain environment. Therefore, frequent and continuous communication is required in order to provide confidence to the stakeholders regarding to the work progress. One of the Scrum pillars is transparency [**?**]. Thus, we define a project as successful if the customers feel themselves confident as a result of the communication and transparency. Hence, we argue that a software project in which Scrum is adopted is able to provide higher customer satisfaction rates by addressing the customer needs regarding to the expected level of communication and transparency within a project. **Hypotheses 4:** *Scrum-based projects provide increased customer satisfaction from the communication and transparency perspective.*

E. Agility

Some projects occurring in a fast-moving or time-constrained environments, call for an agile approach [**?**]. The main characteristics of an agile software project are the "early and continuous delivery of valuable software" and "ability to provide fast response to changes". Thus, we define a project as successful if the agility expected by the customers is met. Hence, we argue that a software project in which Scrum is adopted is able to provide higher customer satisfaction rates by addressing the agility demanded by the customer. **Hypotheses 5:** *Scrum-based projects provide increased customer satisfaction from the agility perspective.*

F. Innovation

Software projects are expected to deliver new software-based products and services for users/customers existing and emerging needs. Therefore, the innovation comes through new ways of work, study, entertainment, healthcare, etc. supported by software. Since Scrum also supports the principle of "early and continuous delivery of valuable software" it is expected that Scrum software development might help to create innovative products and services for the customer business. Thus, we define a project as successful if the innovation expected by the customer is met. Hence, we argue that a software project in which Scrum is adopted is able to provide higher customer satisfaction rates by addressing the customer expectation through innovative products and services generated by the project. **Hypotheses 6:** *Scrum-based projects provide increased customer satisfaction from the innovation perspective.*

G. Benchmark

Usually, software projects are launched as a procurement initiative in which an organization (buyer) hires a development organization (seller) to create a product or service that may be developed by several companies. It is natural that seller organizations do comparison between their suppliers. In this sense, we consider "benchmark" as a comparison between organizations that develop software. Thus, we define a project as successful if customers may recommend a development organization when comparing its project results to other organizations project results. Hence, we argue that a software project in which Scrum is adopted is able to provide higher customer satisfaction rates by comparing a project executed by a specific organization with other ones. **Hypotheses 7:** *Scrum-based projects provide increased customer satisfaction from the benchmark perspective.*

III. Research Method

In order to define a methodology to guide this study, we have chosen an approach based on surveys; and selected five of six recommended steps by Kitchenham [**?**], as below:

- **Setting the objectives:** This study investigates the relationship between the Scrum adoption (as a software development approach) and the customer satisfaction;
- **Survey design:** Cross-sectional, since the survey instrument was applied only once at a fixed point in time. It is not intended to promote a forward-looking to provide information about changes in the specific population through time;
- **Developing the survey instrument:** It was based on a questionnaire designed to identify the customer satisfaction within a particular project which determines its success degree from the external point of view;
- **Obtaining valid data:** The questionnaire was sent through e-mail for each customer business representatives (e.g. sponsor, product or project managers);
- **Analyzing the data:** Finally, the data analysis was executed using techniques from descriptive and inferential statistics.

The following subsections present discussions related to the population, sample, variables, data collection procedure, and data analysis techniques used for this study.

A. Population

The population for this study is targeted on software intensive organizations, including companies of different sizes, developing several software-based solutions for a wide variety of markets.

B. Sample

It was selected a random sample of projects executed by C.E.S.A.R - Recife Center for Advanced Studies and Systems[1] which belongs to the target population. C.E.S.A.R is an innovation institute which has more than 500 employees working

[1] http://www.cesar.org.br/site/

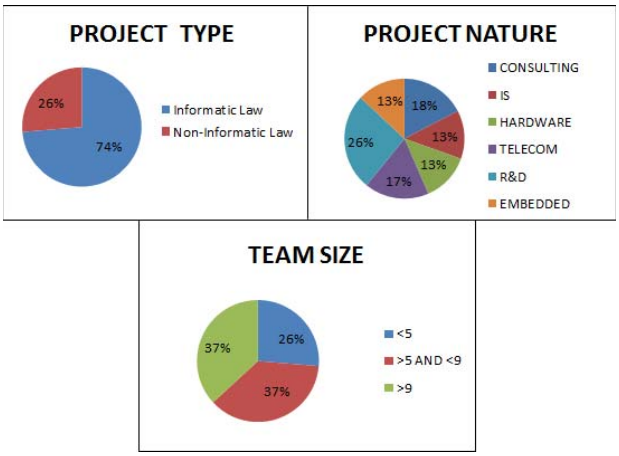

Fig. 4. Contextual variables

on projects from different business domains (e.g. finance, third-sector, manufacture, service, energy, government, telecommunication, etc.), creating solutions for several platforms (mobile, embedded, web, etc.). The number of projects may vary from 70 to 100 in a year.

Initially, the sample contained 27 projects, but it was reduced to 19 projects because incomplete questionnaires responses were eliminated from the sample. Even though, it represents an effective response rate of 70.3%, which is above the minimum norm of 40% suggested by [**?**] for academic studies.

Furthermore, it was collected additional information related to each project, including project type, team size as below (Figure **??**):

- **Project type:** 5 private and 14 public/brazilian tax incentives law.
- **Team size:** From 4 to 21.
- **Project nature:** Consulting: 4; Information Systems: 3; Telecommunications: 4; Maintenance: 1; Research & Development (R&D): 6; Embedded Systems: 4.

Notice that one project may have different natures. Due to this reason, the number may be slightly different from the sample size.

C. Variables

This study contains several variables as following:

- **Independent Variable:** The software process is the independent variable and may assume two different values: Scrum (agile method) and Non-Scrum (any traditional approach).
- **Dependent Variables:** The success of a software project is the result of customer satisfaction from an external point of view considering several aspects: time, goals, quality, communication and transparency, agility, innovation and benchmark. In order to measure the customer satisfaction the Likert scale was used assuming values from 1 (poor) to 5 (excellent) values.
- **Contextual Variables:** Project type, team size, and project nature were identified as variables that may potentially influence the results. Project type and nature categorization was previously defined. The team size was the number of people involved during the development, including engineers, designers and manager.

TABLE I. STUDY HYPOTESES

Null Hypotheses (NH)	Alternative Hypotheses (AH)
(NH1) Ts = Tns	**(AH1)** Ts $\neq$ Tns
(NH2) Gs = Gns	**(AH2)** Gs $\neq$ Gns
(NH3) Qs = Qns	**(AH3)** Qs $\neq$ Qns
(NH4) CTs = CTns	**(AH4)** CTs $\neq$ CTns
(NH5) As = Ans	**(AH5)** As $\neq$ Ans
(NH6) Is = Ins	**(AH6)** Is $\neq$ Ins
(NH7) Bs = Bns	**(AH7)** Bs $\neq$ Bns

D. Data Collection Procedure

First, the questionnaires were sent to customer business representatives through e-mail in a Microsoft Excel spreadsheet format. Each document contained the project categorization regarding to the contextual variables (project type, nature, and team size) and to the independent variable (Scrum/Non-Scrum).

Thus, the customer business representatives were responsible for answering the questionnaire and then sending it back to the C.E.S.A.R project management office (PMO).

E. Data Analysis Techniques

The data analysis considered two different techniques. First, it was executed an exploratory data analysis (descriptive statistics) using tools such as barplots and boxplots in order to identify the preliminary insights about the data characteristics regarding to measures such as mean, position and variation.

Then, hypotheses tests (inferential statistics) were conducted to provide more robust information for the data analysis process as shown in Table **??**. After the exploratory data analysis, it was not found apparent relevant difference in the obtained results. Thus, the alternative hypotheses were modified to verify the inequality, instead of the superiority.

IV. RESULTS

A. Descriptive Statistics - Exploratory Data Analysis

Initially, the final sample - the one containing 19 projects - was divided into two groups (Scrum and Non-Scrum). Then, some exploratory data analysis techniques (descriptive statistics) were applied in order to find out central tendency, position and dispersion related to the data set. On one hand, barplots (Figure **??**) helped to identify the means (central tendency) for each variable representing different aspects of customer satisfaction. On the other hand, boxplots (Figure **??**) helped to reveal the data dispersion and position [**?**].

According to the barplots in Figure **??**, we can notice that the projects using Scrum presented better results considering

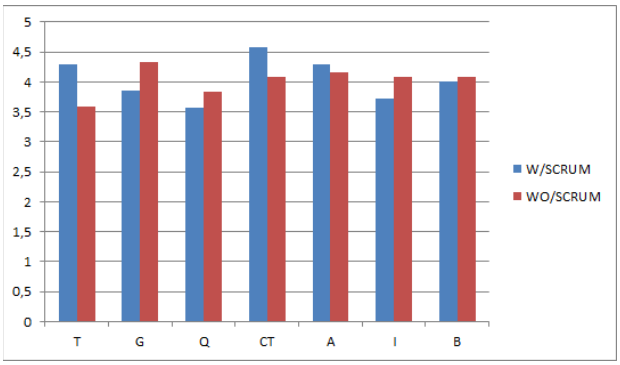

Fig. 5. Dependent variables means

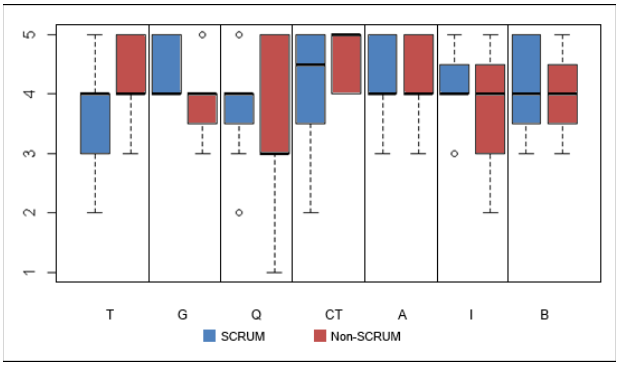

Fig. 6. Dependent variables boxplots

the following aspects: time, communication and transparency and agility. The projects that did not use Scrum presented better results for quality, goals, innovation and benchmark aspects. Despite these results, it not possible to assume that any group (Scrum and Non-Scrum) has an absolute advantage.

According to the boxplot in Figure **??**, it is possible to make some comments about each aspect of customer satisfaction considering the grades obtained from the sample observations:

- **Time (T):** For the Scrum projects groups, the grades presented a dispersion from two to five; and the second and third quartiles are coincident, showing that many grades four were given by the customers. For the Non-Scrum projects, the grades presented a more concentrated behavior with a dispersion from three to five; and a the first and second quartiles are coincident.
- **Goals (G):** For both groups, it was possible to identify a more concentrated data dispersion: from four to five in the Scrum projects; and three to four in the Non-Scrum projects. Besides, there are many occurrences of grades four in both groups. In particular, for the Non-Scrum group, it may be seen an outlier (the grade five).
- **Quality (Q):** For the Scrum group, the variation (dispersion) was from three to four and the mode (most frequent value) was four with two outliers (the grades two and five). For the Non-Scrum group, there was a lot of data dispersion from grade one to five; and three was the mode.
- **Communication and Transparency (CT):** For the Scrum group, there was a variation (data dispersion) from grade two to five without a predominance of any value. For the Non-Scrum group, the grades were more concentrated from grade four to five and the mode was five.
- **Agility (A):** Both boxplots (Scrum and Non-Scrum groups) for the agility variable were extremely similar presenting a variation from grade three to five and the mode was the grade four.
- **Innovation (I):** For the Scrum group the variation was from grade four to five with an outlier (the grade three). For the Non-Scrum group the grades presented a dispersion from grade two to five.
- **Benchmark (B):** For both groups, the variation was the same: from grade three to five without any additional information.

Finally, it is not possible to determine a relevant difference between the results from the groups considering the seven dependent variables as aspects of customer satisfaction. Therefore, there is no evidence about an advantage for the projects in which Scrum was applied.

B. Inferential Statistics - Hypotheses Tests

Since the exploratory data analysis (descriptive statistics) was not able to provide any conclusion within this study, it was decided to go ahead through another method. Hypotheses tests (inferential statistics) was then used intending to establish a systematic basis for a decision about the data set behavior.

First, the same previous segmentation was handled separating the sample into two groups: Scrum (seven elements) and Non-Scrum (12 elements) projects. Thus, we assumed both as independent samples containing ordinal data. In this case, it is recommended using nonparametric test for ordinal variables. In particular, it was chosen the Mann-Whitney's U test [**?**]. When performing nonparametric (or distribution free), there is no need to perform any kind of normality test (goodness of fit).

The choice of U Mann-Whitney test did not bring harm to problem analysis, as in situations where the data are normal, the loss of efficiency compared to using the Student's t test is only 5%; in other situations where the data distribution has a "heavier" tail than normal, the U test will be more efficient [**?**].

Thus, hypotheses tests were performed (using the U test) through R language[2] to determine equality or inequality considering the samples means for each group (Scrum and Non-Scrum) from the perspective of each aspect (dependent variable).

According to the previous hypothesis definitions, the equality was supposed to be accepted if the null hypothesis could not be rejected. Instead (in case of null hypothesis rejection)

[2]http://www.r-project.org/

TABLE II. HYPOTHESES TEST RESULTS

Criterion	NH	AH	P-Value	W
T	Ts = Tns	Ts $\neq$ Tns	0.09736	60
G	Gs = Gns	Gs $\neq$ Gns	0.1137	26
Q	Qs = Qns	Qs $\neq$ Qns	0.7911	39
CT	CTs = CTns	CTs $\neq$ CTns	0.4849	49.5
A	As = Ans	As $\neq$ Ans	0.7126	46
I	Is = Ins	Is $\neq$ Ins	0.4681	34
B	Bs = Bns	Bs $\neq$ Bns	0.8216	39.5

we were supposed to recognize a difference related to the means for each group and assume the alternative hypothesis. The results obtained are presented in Table **??**.

The reference parameter to allow inference about the acceptance or rejection of the null hypothesis was the p-value, which is the test significance level. In addition, the p-value obtained in each test would be compared to the Fisher's scale [**?**] which states that any p-value less than 0.05 should cause the rejection of the null hypothesis. Hence, the obtained results of p-values were absolutely superior compared 0.05. In this case, no null hypothesis should be rejected.

Therefore, there is no evidence that the Scrum group results were higher than Non-Scrum group results. Thus, it is impossible to infer that the adoption of Scrum may increase the customer satisfaction (and the project success as well) within the scope of this research work.

V. LIMITATIONS AND FUTURE WORK

In this research work the internal validity was reduced at the expense of external validity. On one side, the data was collected from real-life industrial software development projects and helped to increase the study external validity. On the other hand, there are several contextual variable (e.g. organization culture and environment factors) that were not controlled and may influence on results, harming the internal validity. So, for studies with reduced internal validity, it is not possible to determine causality and generalization to other contexts.

Furthermore, it is important to point out that it was not executed any test to evaluate the questionnaire psychometric properties what may jeopardize construct validity related to this research.

Spite of these limitations, the study is expected to contribute to agile methodologies body of knowledge and to Scrum discussion (in particular) once it supported by real-life experiences and an empirical evaluation. Thus, some refinements are listed as possible future works as below:

- Increase the sample size for obtaining more robust results. The larger the sample, the stronger will be the inferences about the data population behavior.
- Investigation through other perspectives than customer satisfaction such as team and satisfaction related to the definition of success within a software project.
- Execution of a different empirical evaluation technique. It would be promoted an experiment in order to determine causality relationships. Additionally, it would be included a case study intended to figure out behavior and phenomenon that may lead to increased customer satisfaction.

VI. RELATED WORK

França et al [**?**] promoted a survey aimed to investigate the relationship between agile practices usage and success of projects using Scrum. The context for that research was similar to the one considered for this work: software development companies located in the Porto Digital initiative, Recife, Pernambuco, Brazil. Among the 25 attributes of agile methodologies, only 8 (32%) correlated with the success of the projects. Thus, as in our study, the agile methodologies practices do not seem to show evidence of being decisive for projects success.

Otherwise, a longitudinal case study conducted for 2 (two) years by Mann [**?**] obtained quantitative indications that Scrum adoption may lead to increased customer satisfaction and overtime reduction.

Begel [**?**] also presented an industrial survey with Microsoft employees about the use of agile practices. In this context, it was reported improved communication, quick releases, and flexibility/rapid response to changes as the main benefits. On the other side, it was also reported disadvantages including excessive number of meetings, difficulty to scale-up for large projects; and buy-in decisions management.

VII. CONCLUSION

This paper has described an empirical evaluation designed to provide insights for the question: "What is the impact of Scrum on the Customer Satisfaction"? In general, people who are enthusiastic of agile methods (including Scrum) argue that these approaches are more suitable for software development that is uncertain and requires flexibility to accommodate changes. In this context, we aimed to investigate the relationship between the adoption of agile methodologies and increased success rates in software development projects.

In order to provide an accurate comparison, we defined the scope as considering an external perspective for success based on the customer satisfaction according to several aspects including time, goals, quality, communication and transparency, agility, innovation and benchmark (dependent variables). Thus, other perspectives and aspects were considered out of scope for this research. Additionally, for a proper comparison the study focused on the project management property for software development approaches.

We chose a cross-sectional survey using a real-life project sample as our empirical evaluation method. The sample was separated into two groups, Scrum and Non-Scrum (independent variable). This segmentation was intended to allow a comparison between projects using Scrum and those using other traditional approach for managing software development approaches. In particular, the comparison was performed for each dependent variable, intending to promote a detailed analysis instead of an overall comparison.

The preliminary results from the exploratory analysis showed no differences regarding to the data behavior fo both

groups (Scrum and Non-Scrum), considering several properties such as central tendency, position, and dispersion. Then, quantitative analysis using a Mann-Whitney hypothesis test (U test) also showed no relevant difference between both groups results. Therefore, it was not possible to establish any superiority associated with the use of Scrum in software development projects.

We recognize some limitations for this study. First, the internal validity might be threatened since we did not control any contextual variable. Then, the construct validity might be harmed because we were not able to verify: the psychometric properties related to the questionnaire and also the standard application of Scrum practices and guidelines. In spite of these limitations, we expect this research can help industry and academia in developing the software development body of knowledge by combining scientific rigor with industry experience. We also expect to contribute to the organization (C.E.S.A.R) to understand how to increase success rates internally.

In the future, we intend to execute another survey with an increased size sample (containing projects from several organizations) considering the contextual variables as criteria for data categorization. By promoting these refinements, we aimed to figure out patterns of data behavior for specific groups. In addition, other empirical evaluation techniques (experiments, case study) might be applied in order to overcome the limitations mentioned previously.

APPENDIX - QUESTIONNAIRE

This appendix describes the questionnaire used as instrument to measure each specific aspect of customer satisfaction, as well as the likert scale anchoring approach as recommend by Uebersax [**?**]. It is aimed to provide a common understanding about the concept model and their qualitative values.

A. Time: What is the customer feeling regarding to the project deadlines?

5. Excellent: All deadlines defined or negotiated with customer have been achieved. Deadlines adjusted due to external dependencies to the customer must be considered here.
4. Good: All deadlines, including the ones were negotiated due to internal technical problem within executing organization were met. In this classification each deadline may not have been rescheduled more than once.
3. Fair: Existence of negotiated deadlines more than once, due to problems with the executing organization, but were met.
2. Unsatisfactory: Existence of some deadlines that were not met and the deliveries occurred late.
1. Poor: Much of time constraints were not met, or there is delay (s) that seriously impacted the customer.

B. Goals: Does the customer think the project objectives were met?

5. Excellent: All agreed objectives were met.
4. Good: Nearly all agreed objective were met. Goals not met are less priority.
3. Fair: Some important goals were not met according to the customer expectations.
2. Unsatisfactory: Not meeting several important goals.
1. Poor: The executing organization staff showed lack of ability to identify customer needs and care of the goals was very unsatisfactory.

C. Quality: What is the perception about the quality of the project and its products and services?

5. Excellent: It was found no defect or only a few minor ones.
4. Good: Some low severity defects were found and they were resolved in a satisfactory manner and within the agreed time.
3. Fair: Few moderate severity defects were detected and they have been resolved in a satisfactory manner and within the agreed time.
2. Unsatisfactory: Various defects of low severity were identified. Or defects in general were not resolved within the time agreed with the client.
1. Poor: Critical severity defects were identified at the stage of acceptance tests.

D. Communication and Transparency: Does the customer feel comfortable due to the information provided on the progress of the project?

5. Excellent: Very effective communication between the executing organization and the customer is performed proactively, without the client request, providing the proper level of information.
4. Good: Continuous transparency to the project through its execution. Communication is established when requested by the customer.
3. Fair: Existence of some problems related to one of them: information display; lack of information, form of presentation and data confusion.
2: Unsatisfactory: At various times it was not easy to see the actual project progress. Information was not available when they should be.
1. Poor: Transparency about the project progress was nonexistent throughout its execution.

E. Agility: What is the customer perception about the organization agility within a specific project?

5. Excellent: Expectations exceeded and high level of professionalism.
4. Good: There was a satisfactory flexibility.
3. Fair: There was flexibility but sometimes expectations were not met.
2. Unsatisfactory: There was some problem that not impacted the project execution. However, there are several improvement areas.
1. Poor: Existence of major problems that introduced impacts the project execution, including unresolved and controversial issues.

F. Innovation: What is the customer perception about the team capacity of bring innovation and innovative solutions?

5. Excellent: Team with excellent ability to present innovative and efficient solutions, beyond expectations.
4. Good: Team presented satisfactory innovative solutions, meeting expectations.
3. Fair: Team presented some innovative solutions, but not all expectations were met.
2. Unsatisfactory: Team with low capacity to present innovative solutions to the tasks. Several problems were faced when trying to resolve more complex requirements.
1. Poor: Lack of ideas / innovative solutions, not meeting the expectations.

G. Benchmark. What is the organization performance compared to other suppliers considering the project execution?

5. Excellent
4. Good
3. Fair
2. Unsatisfactory
1. Poor

ACKNOWLEDGMENT

The authors would like to thank C.E.S.A.R (Recife Center for Advanced Studies and Systems) for kindly providing real-life projects data obtained from its PMO (Project Management Office); We would also like to thank Federal University of Pernambuco (UFPE) and Informatics Center (CIn) for supporting this research work. This work was partially supported by the National Institute of Science and Technology for Software Engineering (INES), grants 5739642008-4 (CNPq) and APQ-1037-1.03/08 (FACEPE). Bruno Cartaxo and Antonio Sa Barreto is supported by FACEPE, Sérgio Soares is partially supported by CNPq grant 3050852010-7.

REFERENCES

[1] P. Naur and B. Randell, Eds., *Software Engineering: Report of a conference sponsored by the NATO Science Committee, Garmisch, Germany, 7-11 Oct. 1968, Brussels, Scientific Affairs Division, NATO*, 1969.

[2] M. Griffiths, *PMI-ACP Exam Prep: Rapid Learning to Pass the Pmi Agile Certified Practitioner (Pmi-acp) Exam - on Your First Try!: Premier Edition*. Rmc Publications Incorporated, 2012. [Online]. Available: http://books.google.com.ar/books?id=mM6rtgAACAAJ

[3] I. Sommerville, *Software Engineering*, 9th ed. Harlow, England: Addison-Wesley, 2010.

[4] P. Kruchten, *The Rational Unified Process: An Introduction*, 3rd ed. Boston: Addison-Wesley, 2003.

[5] "2001 chaos report," Tech. Rep.

[6] B. Cartaxo, I. Costa, D. Abrantes, A. Santos, S. Soares, and V. Garcia, "Eseml: empirical software engineering modeling language," in *Proceedings of the 2012 workshop on Domain-specific modeling*, ser. DSM '12. New York, NY, USA: ACM, 2012, pp. 55–60. [Online]. Available: http://doi.acm.org/10.1145/2420918.2420933

[7] D. I. K. Sjoberg, T. Dyba, and M. Jorgensen, "The future of empirical methods in software engineering research," in *2007 Future of Software Engineering*, ser. FOSE '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 358–378. [Online]. Available: http://dx.doi.org/10.1109/FOSE.2007.30

[8] W. F. Tichy, "Should computer scientists experiment more?" *Computer*, vol. 31, no. 5, pp. 32–40, May 1998. [Online]. Available: http://dx.doi.org/10.1109/2.675631

[9] T. Dyba and T. Dingsoyr, "What do we know about agile software development?" *Software, IEEE*, vol. 26, no. 5, pp. 6–9, 2009.

[10] PMI, Ed., *A Guide to the Project Management Body of Knowledge (PMBOK Guide): An American National Standard ANSI/PMI 99-001-2008*, 4th ed. Newtown Square, PA: Project Management Institute, 2008.

[11] K. Schwaber, *Agile Project Management With Scrum*. Redmond, WA, USA: Microsoft Press, 2004.

[12] B. A. Kitchenham and S. L. Pfleeger, "Personal Opinion Surveys," in *Guide to Advanced Empirical Software Engineering*, F. Shull, J. Singer, and D. I. K. Sjøberg, Eds. Springer, 2008, pp. 63–92+.

[13] Y. Baruch, "Response Rate in Academic Studies-A Comparative Analysis," *Human Relations*, vol. 52, no. 4, pp. 421–438, Apr. 1999. [Online]. Available: http://dx.doi.org/10.1177/001872679905200401

[14] W. O. Bussab and P. A. Morettin, *Estatística Básica*, 6th ed. Saraiva, 2010.

[15] S. Siegel and N. Castellan, *Nonparametric statistics for the behavioral sciences*, 2nd ed. McGraw–Hill, Inc., 1988.

[16] A. C. C. França, F. Q. B. da Silva, and L. M. R. de Sousa Mariz, "An empirical study on the relationship between the use of agile practices and the success of scrum projects," in *Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, ser. ESEM '10. New York, NY, USA: ACM, 2010, pp. 37:1–37:4. [Online]. Available: http://doi.acm.org/10.1145/1852786.1852835

[17] C. Mann and F. Maurer, "A case study on the impact of scrum on overtime and customer satisfaction," in *Proceedings of the Agile Development Conference*, ser. ADC '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 70–79. [Online]. Available: http://dx.doi.org/10.1109/ADC.2005.1

[18] A. Begel and N. Nagappan, "Usage and perceptions of agile software development in an industrial context: An exploratory study," in *Proceedings of the First International Symposium on Empirical Software Engineering and Measurement*, ser. ESEM '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 255–264. [Online]. Available: http://dx.doi.org/10.1109/ESEM.2007.85

[19] J. S. Uebersax, "Likert scales: Dispelling the confusion," april 2013. [Online]. Available: http: //www.john-uebersax.com/stat/likert.htm