

# Using Nokia Test to Evaluating Quality and Productivity on Scrum-CMMI Environments

Célio Santana<sup>1</sup>, Amanda Nunes<sup>1</sup>

<sup>1</sup> Federal University of Pernambuco, Communication and Art Center, Prof Luiz Freire Avenue.  
S/N,50740-540 Recife, Pernambuco, Brazil

{[celio.santana@gmail.com](mailto:celio.santana@gmail.com), [anunes1992@yahoo.com.br](mailto:anunes1992@yahoo.com.br)}

**Abstract.** The Nokia Test is an effective approach to evaluate how teams which use Scrum are following the process proposed by this agile method. The expected result is that Scrum teams presenting higher scores in Nokia Test also present better results in main indicators of quality and productivity defined by companies which are using the traditional maturity model of process improvement software CMMI. This paper describes a case study in three companies located in Recife Brazil which were evaluated in the Nokia Test and also had their quality and productivity indicators measured quantitatively for comparison.

**Keywords:** Scrum, Agile Software Development, Nokia Test.

## 1 Introduction

Software process improvement (SPI) has become a practical tool for companies where the quality of the software is of high value [1]. In a technical report with results from thirteen organizations, and with the number of post-release defect reports used as a measure, Herbsleb and others practitioners [2] showed that due to software process improvement (SPI), the products and business value (especially return on investment – ROI) was improved.

It is generally considered that a well documented and a repeatable process is essential for developing software products of high quality [3]. There is also evidence that the use of standards and process assessment models has a positive impact on the quality of the final software product [4].

The software engineering community has gradually moved from product quality-centered corrective methods to process quality-centered, preventive methods, thus shifting the emphasis from product quality improvement to process quality improvement. Inspections at the end of the production line have long been replaced by design walkthroughs and built-in quality assurance techniques throughout the development life cycle [5].

In traditional SPI methods and approaches [6,7,8], the aspect of organizational improvement has usually been placed in a central role, due to the fact that the planning and control of the SPI initiatives are managed by the organizational stakeholders. The reported positive effects of SPI methods and approaches include reducing indicators such as time to market, risks and costs, and increasing the productivity and quality in software development organizations [9, 10].

However, various negative effects have also been encountered, e.g. regarding the cost-effectiveness of SPI initiatives, their actual effectiveness in improving the software development practices of organizations, the volume of the effort needed to implement SPI initiatives and the low speed at which visible and concrete results are achieved [9,11,12]. In fact, it has been reported that around two-thirds of SPI traditional initiatives fail to achieve their intended goals [13].

From the mid-1990s onwards, agile software development principles and methodologies have been increasingly challenging the traditional view of software development, and therefore,

provides a highly untraditional approach to SPI, in which the process improvement knowledge of software developers and software development teams is acknowledged and valued [14].

We have long known that skilled people are the most crucial resource in software development. Back in the 1990 Bach [15] stated that “Everyone knows the best way to improve software productivity and quality is to focus on people.” However, it took more than 10 years for the agile manifesto and agile methods (Extreme Programming, Scrum, Crystal, and many others) to truly place the emphasis on people and their interaction [3].

In 2006 one method called Nokia Test [16] was proposed to evaluate how teams which using Scrum [17] are following the process proposed by this Agile Method. The goal of the Nokia Test is identifying opportunities of improvement in the team behavior (Agile SPI) when adopting Scrum.

In 2007 Jeff Sutherland proposed a score system [18] for the Nokia Test to measuring the Scrum Process adherence quantitatively and supports Scrum teams to identify where improvements should be carried on by the team.

This paper presents a case study in three Brazilian companies which adopts Scrum and already CMMI level 2 or are towards that certification, but their process are adherent to the CMMI level 2. The Nokia Test was applied in these companies and CMMI indicators were measured. The comparison between the results of different companies and projects is the target of this study.

In this light, the main objective of this paper is to present the result of a case study in three Brazilian companies which adopts Scrum and already CMMI level 2 or are towards that certification, but their process are adherent to the CMMI level 2. To do so, after this introductory section, the section 2 explores Scrum, the section 3 explores the Nokia Test, the section 4 show the relevance of CMMI for this research, in section 5 the goal of the case study and settings of the companies and projects are introduced. Section 6 shows the data retrieved in the companies and their projects, these data will be analyzed using Graphics for better visualization.

## **2 Scrum**

Scrum [17] has been in use for a relatively longer period than other agile methodologies. Scrum, along with Extreme Programming [19], is one of the more widely used agile methodologies. The focus of Scrum is based on the fact that “defined and repeatable processes only work for tackling defined and repeatable problems with defined and repeatable people in defined and repeatable environments” [20], which is obviously not possible.

To solve the problems caused by defined and repeatable processes, Scrum split the project into short iterations, which are called Sprints, from 15 until 30 days. Before a sprint begins, the priority features are chosen, the amount of functionality that the team could deliver is “packed” to that Sprint, and the team must deliver all of these features in the defined timebox (15 or 30 days).

The point is to stabilize the requirements during the Sprint. Scrum emphasizes project management concepts [21] though some may argue that Scrum is as technical as XP. The term Scrum is borrowed from Rugby: A Scrum occurs when players from each team clump closely together in an attempt to advance down the playing field.

Ken Schwaber first described Scrum in 1996 as a process that accepts that the development process is unpredictable, formalizing the “do what it takes” mentality, and has found success with numerous independent software vendors. Scrum starts with the premise that software development is too complex and unpredictable to be planned exactly in advance. Instead, empirical process control must be applied to ensure visibility, inspection, and adaptation.

The different environmental and technical variables (such as time frame, quality, requirements, resources, implementation technologies and tools, and even development methods) must be controlled constantly in order to be able to adapt to changes flexibly. This is achieved through an iterative and incremental development process [17].

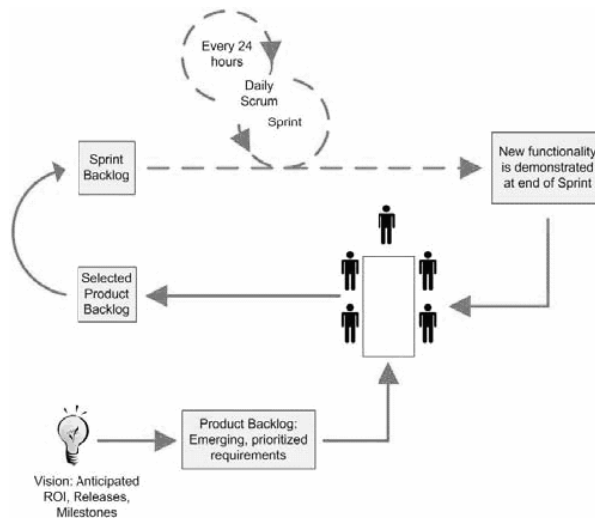
Scrum implements an iterative, incremental skeleton through three roles: the Product Owner, the Team, and the ScrumMaster [17], as shown in Table 1

**Table 1.** Scrum's roles and responsibilities

Role	Responsibilities
Product Owner	Represents the interests of everyone with a stake in the project and its resulting system. Maintains the Product Backlog, i.e., a prioritized list of project requirements with estimated times to turn them into completed product functionality.
Team	Responsible for developing functionality. Teams are self-managing, self-organizing, and cross-functional, and they are responsible for figuring out how to turn Product Backlog into an increment of functionality within an iteration and managing their own work to do so. Team members are collectively responsible for the success of each iteration and of the project as a whole.
ScrumMaster	Responsible for managing the Scrum process, i.e., for teaching Scrum to everyone involved in the project, for implementing Scrum so that it fits within an organization's culture and still delivers the expected benefits, and for ensuring that everyone follows Scrum rules and practices.

The Scrum flow is shown in Fig 1. According to Schwaber [17], a SCRUM-based project starts from a high level vision of the system will be developed. After that, Product Backlog is created containing a list of known requirements. So, the items of the Product Backlog are prioritized and divided in small time-boxed iterations called sprints.

A Sprint is a 30-day period of development time which is initiated with a Sprint planning meeting, where the Product Owner and Team get together to collaborate about what will be done for the next Sprint. Selecting from the highest priority Product Backlog, the Product Owner tells the Team what is desired, and the Team tells the Product Owner how much of what is desired it believes it can turn into functionality over the next Sprint. In the first sprints, most of architecture and infra-structure work is done, so less functionality are released [17].



**Fig. 1.** Scrum process overview [17].

After deciding what has to be done in the next Sprint, the Team develops the Sprint Backlog, i.e., a list of tasks that must be performed to deliver a completed increment of potentially shippable product functionality by the end of the Sprint. The tasks in the list emerge as the Sprint evolves and should be divided so that each takes roughly 4 to 16 hours to finish [17].

During sprints execution, the team meets daily in 15- minute meetings to track work progress and schedule other meetings, if necessary. At the Daily Scrum, each Team member answers three questions:

- What have you done on this project since the last Daily Scrum Meeting?
- What will you do before the next meeting?
- Do you have any obstacles?

At the end of each sprint, the team presents its results to the stakeholders. This presentation will guide the inspection of developed functionalities and possible adjustments in the project [17]. At the end of the Sprint, a Sprint review meeting is held at which the Team presents what was developed during the Sprint to the Product Owner and any other stakeholders who want to attend.

After the Sprint review and prior to the next Sprint planning meeting, the Scrum Master also holds a Sprint retrospective meeting in order to encourage the Team to revise, within the Scrum process framework, its development process to make it more effective and enjoyable for the next Sprint [17].

The Scrum team must be cross-functional, self organized and empowered, there is no project managers, the ideal size of a Scrum team is 7 (seven) members or following the rule  $[N*[N-1]/2]$  [17].

### 3 Nokia Test

In 2005, Bas Vodde was coaching teams at Nokia Networks in Finland and developed the first Nokia Test [16] focused on Agile practices. He had hundreds of teams and wanted a simple way to determine if each team was doing the basics.

It does not provide the secret sauce for hyper performing teams. However, it is the first line of the recipe for high performance. These tests were applied in Scrum Teams at OpenView Venture Partners and to their portfolio companies as the venture group does not expect good performance from Scrum teams without passing the Nokia test. They are also very interested in predictability of release dates which is impossible without passing grades on the test [18].

The test consists in eight questions about how the teams are adopting Scrum. The Nokia Test is in two parts. The first part consists of three questions asking if the team is doing Iterative Development. The questions are:

- Iterations must be timeboxed to less than 4 weeks?
- Software features must be tested and working at the end of each iteration?
- The Iteration must start before specification is complete?

The next part of the test checks whether the team is doing Scrum, in view of the Nokia, understanding, the remaining five questions are:

- You know who the product owner is?
- There is a product backlog prioritized by business value?
- The product backlog has estimates created by the team?
- The team generates burndown charts and knows their velocity?
- There are no project managers (or anyone else) disrupting the work of the team?

In 2008 Jeff Sutherland developed a Nokia Test scoring system. Each person on the team takes a sheet of paper and prepares to score eight questions on a scale of 1-10 and teams must average their score [18].

In 2009, a team question was added to the Nokia test about teams. Sutherland created four categories for Nokia Tests results and linked these averages with following revenues [18]:

- **ScrumButt** (Average < 8,0) - revenue up 0-35%
  - Yahoo - average 35% productivity improvement
  - IDC 2008 study of Agile companies 16% improvement
- **Pretty Good Scrum** (8,0 <= Average < 9,0) – revenue up 150% - 200%
  - Systematic Software Engineering- 200%
  - Google - 160%
- **Good Scrum** (9,0 <= Average < 10,0) - revenue up 300%
  - Companies in Scandinavia, he can't talk about
- **Great Scrum** (Average = 10) - annual revenue up 400%
  - PatientKeeper

Sutherland [12] stated that in the initial evaluations the teams are average 4.0, but in six months their average up to 7,0 and the velocity is about 300% of initial velocity.

The Return of Investment (ROI) is about 11000% on first year since that is relatively easy and cheap to change scrum process for better. Answering the Nokia Test provide to the team some idea about how much it is really doing Scrum. The final result may not reveal how much good are but where each team can improve in Scrum [9].

## 4 CMMI

The Capability Maturity Model Integration (CMMI) [8] is defined as a process improvement maturity model for the development of products and services. It consists of best practices that address development activities that cover the product life cycle from conception through delivery and maintenance. There are 22 Process Areas (PAs) where each area is a cluster of related practices that, when implemented collectively, satisfies a set of goals which satisfies software business needs and considered important for promote improvement in that area [8].

One of these process areas in CMMI is called Measurement and Analysis, this PA is related to all kind of measurement in the company process. In measurement and analysis process area is suggested that the company must define which indicators will be watched by the team, by the project managers and by the top managers. And how these indicators will be collected and analyzed.

In this research, the importance of the companies being adherence to the CMMI level 2 is the obligation and the discipline to realize measurement activities in regular intervals and always following the same “defined process” for collect and analyze the data avoiding BIAS in this process.

Measurement and Analysis is a process area from level 2 of the CMMI, which means that all companies which are CMMI level 2 must perform some measurement activities. Normally the companies choose as indicators, basic information about software development such as productivity, cost, delays, effort and size of the tasks. Other indicators could be adopted by companies according their internal policies.

## 5 Case Study

The case study is related to three companies located in Recife – Brazil. The goal of this case study is collect quality and productivity indicators, required by CMMI measurement and analysis process area. These collect will be done for each team and compare these indicators to the Nokia Test Score obtained by that team.

After that, these data will be analyzed quantitatively to find some relationship between quality and productivity indicators and the Nokia Test Score obtained in each team. Thus, it is

possible to verify some coincident behavior in Nokia Test and the quality and productivity indicators.

The Nokia Test considered to this study case was the eight question version. The ninety question showed confuse to some teams and to equalize the answers this question, added later in original Nokia Test, was not considered in the test.

These indicators will be analyzed considering 3 Sprints of each company being a first evaluation of the improvements in the quality and productivity indicators when using Scrum. These evaluations will consider the Nokia Test since that the Nokia Test not changed in the period, and the Nokia Test evaluate “objectively” the team Adherence to Scrum, and this adherence is target of this Study.

The information about each company is given below, for privacy reasons just one of them will use it real name while the other two companies will use mnemonics A and B.

### **5.1 State Bureau of IT (ATI)**

This company is a civil public agency, located in Recife-Brazil, which is responsible for providing all IT services to citizens. This bureau of state jurisdiction adopted the policy for outsourcing local software factories for coding their solutions. Because of this legal restriction the ATI board decided to follow the Capability Maturity Model Integration for Acquisition (CMMI-ACQ) [27].

Considering this, the indicators chosen by ATI for controlling their contracted was product quality and team performance. Also, ATI adopts Scrum as method for conduct their contract management. The ATI acts as Product Owner and the contracted factory acts as Team & Scrum Master. ATI uses a 15 days time box.

ATI started its experience collecting information from 2 different projects. The first project involves the development of an integrated system for government management (GRP) and the team is composed by 18 programmers and 1 Scrum Master. The second projects involves the building of all government services web-based, the team is composed by 12 programmers and 1 Scrum Master. Both teams are using Scrum for 10 Months and they are adopting Scrum practices step by step, the last addition to the Scrum was the Retrospective Meeting.

The indicators used in both projects are:

- Amount of Story Points Delivered
- Amount of Effort (in hours) Realized
- Amount of Stories
- Number of Bugs Found
- Amount of Effort Spent in Rework
- Amount of Stories not Delivered
- Amount of System Evolution Stories
- Amount of Urgencies Requests which emerges during Sprint
- Amount of Stories which leaves the Sprint Backlog During Sprint
- Budget

Since that the projects present different settings, it was necessary create derivate indicators to compare, without judgment, their performance based in related aspects which become possible the comparison. These new indicators are:

- Points delivered per Person per Sprint (PDS)
- Point Size - Hours per Point (PS)
- Bugs per 100 Points (BHP);
- Percentage of Rework in Hours (PRH)
- Percentage of points spent in Urgencies Requests (PPR)
- Percentage of points non delivered (PPD)

The PDS indicator shows how much points, in average, one person of the team delivered in

that Sprint. If the team delivers more or less points is a relative way for evaluating productivity because the point could be enlarged or shrunk. Thus, this indicator is evaluated considering the PS indicator.

The PS indicator represents how much time, in hours, was spent to finish one point. This average cannot reflect the real productivity of the team, but it is a tool for recognize abnormal growth in PDS indicator.

The BHP indicator represents how much bugs were found in products for 100 points amount. This indicator is related to quality control of the product. It is related to PRH indicator which represents the percentage of rework done in current Sprint.

The PPR indicator shows the percentage of points used in Urgent requests. These urgent requests are stories which enter after the sprint start. This kind of requests are the volatile scope of the Sprint, because to these requests enter in Sprint Backlog, the same amount of points must leave.

The PPD indicates the amount of points not delivered by the team. This is the most important indicator of both projects because it is considered the efficacy “thermometer” of the team.

## **5.2 Company A**

This company is four years experienced in software development and also CMMI level 2. This company does not develop software for ATI, so this is a distinct environment. The company uses Scrum in many projects, but only three will be evaluated in the research.

These three projects: Alfa, Beta, Gama composed by 5 (five), 9 (nine) & 7 (seven) programmers respectively. Each team is still composed by one Scrum Master, one Product Owner, one Quality Assurance and one Designer. The timebox defined to these projects are 30 Days. The company adopts Scrum since 2008 and there is no information team experience in Scrum.

The indicators used in these projects are:

- Profit Value Planed to the month (PVP)
- Profit Value Obtained in the month (PVO)
- Estimated Profit Value at the End of the Project (EAC)
- Budget (US\$)
- Critically of the Project (CRI)

These indicators are relative indicators already, except budget, thus there is no need for adjustment in these indicators. The budget will be not considered because its value is variable depending the size of the project.

The PVP indicator shows the expectations about the profit of the project to the current month while the PVO was the real profit obtained in that month.

The CRI show how critical is the project to the company, this indicator is used to allocate budget and resources.

## **5.3 Company B**

The case study is related to a 50 (fifth) person sized company located in Recife -Brazil with 15 years of experience market and will be evaluated CMMI level 2 at the mid of 2010 using Scrum in 6 teams. Three projects Alfa, Beta & Gamma with 8 (four), 7 (three), 6 (two) members respectively.

Each team is composed of one Scrum Master, one Product Owner, one Quality Assurance, one Designer and the remaining members are programmers (4, 3 & 2 respectively). The Scrum

Master, Product Owner, Quality Assurance and Designer are shared by all teams projects, with a total of fourteen people studied.

The Scrum Master, Product Owner, Quality Assurance and Designer, which work in all projects, did not answered the Nokia Test, because they are not full time in evaluated projects which could bring bias to the collected data.

The company adopts Scrum for 2 years, but the average time of Scrum experience of the team is about 18 months. The indicators used in these projects are:

- Effort Deviation - Planned x Realized (ED)
- Points Size Deviation - Planned Size X Realized Size (PSD)
- Size of the Point - (SOP)
- Percentage of Rework (RE)
- Process Issues – Non Compliance (PI)
- Number of Bugs (BUG)

Since that the projects present different settings, it was necessary adjust one of these indicators, since the first five indicators are relative. The new indicator needed is:

- Bugs per 100 Points (BHP);

The ED and PSD indicators show the difference between planned and realized. The ED is related to hours (effort) deviation while PSD is related to points deviation.

The size of the point (SOP) represents how much average time, in hours, one programmer spent to finish one point.

The percentage of rework (RE) shows how many points in current sprint was spent in work already done.

The PI process indicates how much time the team not follow the process, and the BHP represents how much bugs were found in the products for each 100 points.

## **6 Data Gathering and Analysis**

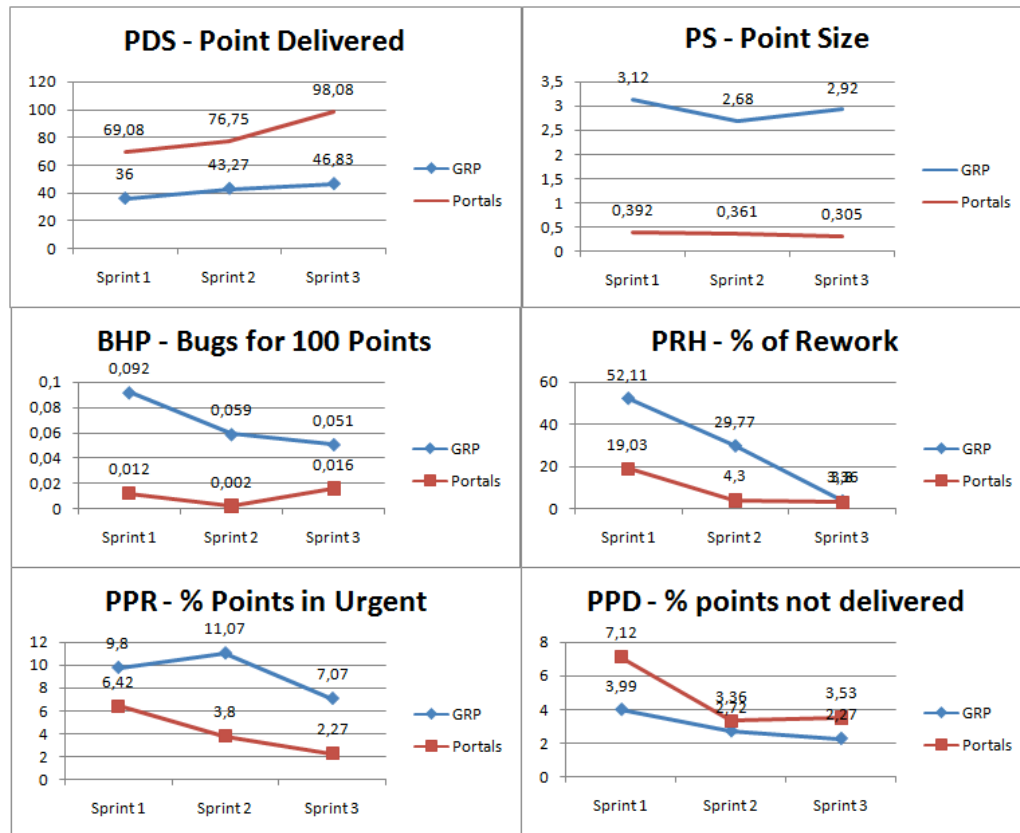
This section aims to present the data retrieved in the three companies, their related projects and to discuss the previous results.

### **6.1 State Bureau of IT (ATI)**

In projects related to this company, the Nokia Test was realized considering Team, Scrum Master and Product Owner opinions. For the GRP project the Nokia Test is 5,3847 and in the web projects 4,4876. The Figure 2 shows the Graphics with the data collected on ATI.

The data collected indicates that both projects are presenting significant improvement in the CMMI indicators even their Nokia Test being lower than the companies A and B presented below. This scenario shows that is possible to improving the quality and productivity results even using a ScrumButt presenting low grades





**Fig. 2.** CMMI indicators in ATI.

## 6.2 Company A

In projects related to this company, the Nokia Test was realized considering only the opinions programmers's. For the Alfa project the Nokia Test is 6,9. In the Beta project the Nokia Test Score was 7,53. The Nokia Test Score in Gama project was 6,61. The Figure 3 shows the Graphics with the data collected on Company A.

The data collected indicates that all projects are decreasing their results looking the CMMI indicators. The Company A assigned the responsibility of these bad results to the World Crisis once all of these projects are related to multi-national companies. The influence of this external factor was crucial to reaching these results.

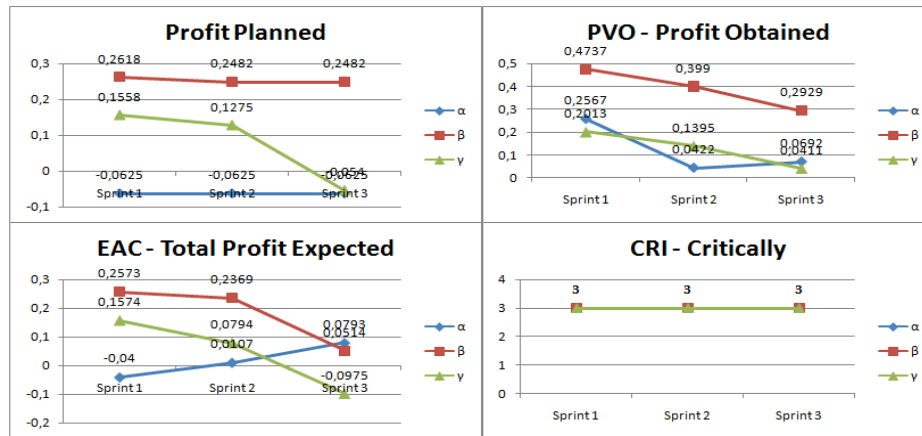


Fig. 3. CMMI indicators in Company A.

### 6.3 Company B

In projects related to this company, the Nokia Test was realized considering Team members opinions. For the Alfa project the Nokia Test is 8,78. In the Beta project the Nokia Test Score was 8,17. And the Nokia Test Score in Gama project was 7,25. The Figure 4 shows the Graphics with the data collected on Company B.

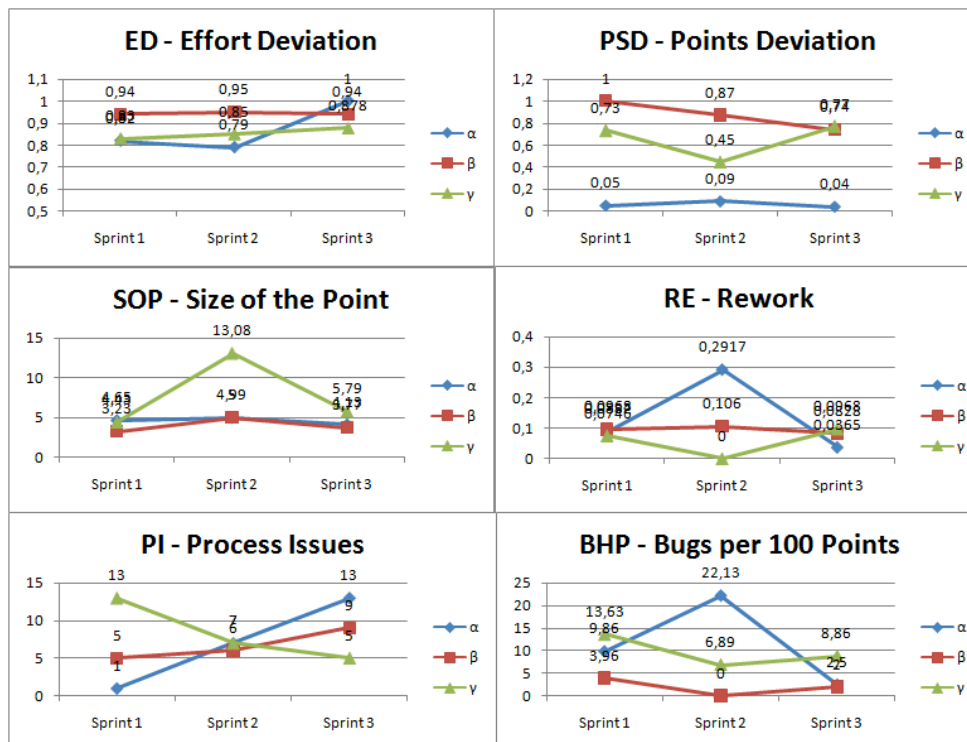


Fig. 4. CMMI indicators in Company B.

The data collected indicates that this company presents results more stable than ATI and Company A. In general this company shows a slightly improvement in their CMMI indicators,

but, the Nokia Test Score is the best of the three companies studied, thus, it would be expected a better result to this company.

## **7 Conclusion**

These data could not provide the full understanding about using Nokia Test itself as indicator in CMMI environments which use Scrum. The first reason is the tiny sample, just three companies, which provided few data for a deep investigation about that subject. The second reason is the difference between the expected and real results in these companies.

The expected result was that teams presenting higher Nokia Test score, should present better results of quality and productivity indicators, or/and present a exponential growth in some of these indicators.

These expected results were based in the Sutherland works [18,22] which states that: the higher the score in Nokia Test provides better results on investment (ROI). However this statement cannot be surely repeated when related to quality and productivity indicators.

### **7.1 Contributions**

Sutherland [18,22] stated that the ROI for adopting Scrum above the ScrumButt level, score 8.0 or higher in Nokia Test. But, the question was what happens with other quantitative indicators in these companies when teams are using Scrum?

The first impression, although not conclusive, is those indicators are not related to the Scrum adoption. The behavior of the indicators during the sprints was irregular and it is not possible to identify one tendency, considering the size of the sample.

The authors of this research agree with Bas Vodde [16] statement which is: The Nokia Test must not be used to compare different teams or different projects. The data of this research shows unexpected behavior.

Companies which want to use CMMI and Scrum cannot use the Nokia Test as indicator for evaluations in it CMMI + Scrum process. At least when it comes to a short period of time, in this case were analyzed just Sprints.

### **7.2 Future Works**

This research is related to a MSc Thesis about Software Process Improvement in CMMI + Agile Environments. The idea of using the Nokia Test as CMMI indicators in CMMI + Scrum projects could be useful and quick way for appraisal this kind of process.

The first future work is watching what happens with these same quality and productivity indicators when they are observed for a long period. The possibility of stabilization of the process can provide consistent data for new analysis.

The second future work is study the behavior of one company, whose processes is stabilized, when their Nokia Test are changed for higher or lower grades. How the variation of Scrum process, and therefore the Nokia Test, could influence the quality and productivity indicators in a CMMI + Scrum environment.

Other future work is to identify differences or particularities in CMMI + Scrum environments which can cause interference in Nokia Test score or even significant differences in the CMMI indicators which are not ROI.

## **References**

1. Järvinen, J.: On comparing process assessment results: BOOTSTRAP and CMM. In Software Quality Management, SQM94, 247--261), Edinburgh (1994)

2. Herbsleb, J., Carleton, A., Rozum, J., Siegel, J., Zubrow, D.: Benefits of CMM-based software process improvement: Initial results. Technical report, CMU/SEI- 94-TR-13 (1994)
3. Stamelos, I., Sfetos, P.: Agile Software Development Quality Assurance. Information Science Reference, (2007)
4. Kitchenham, B., Pfleeger, S. L.: Software quality: The elusive target. In *IEEE Software*, 13(1), 12--21. (1996)
5. Goergiadou E.: Software Process and Product Quality Assurance: A historical Perspective. *Cybernetics and Systems Analysis*. 11(4), 19--27 (2004)
6. Basili, V. R.: The Goal Question Metric Approach. In: *Encyclopedia of Software Engineering*. John Wiley & Sons, Inc. 13(6), 528--532. (1994)
7. Basili, V. R., Caldiera, G.: Experience Factory. In: *Encyclopedia of Software Engineering*. Marciniak, J. J. (ed.). John Wiley & Sons, Inc. 469--476 (1994)
8. SEI.: Capability Maturity Model® Integration (CMMISM), Version 1.2. Carnegie Mellon Software Engineering Institute. (2006)
9. Krasner, H.: The Payoff for Software Process Improvement: What it is and How to Get it. In: *Elements of Software Process Assessment & Improvement*. El Emam, K. & Madhavji, N. H. (ed.). IEEE Computer Society: Los Alamitos, California. 113-130 (1999)
10. Van Solingen, R., Berghout, E.: The Goal/Question/Metric Method: A Practical Guide for Quality Improvement of Software Development. The McGraw-Hill Companies. (1999)
11. Dybå, T.: An Instrument for Measuring the Key Factors of Success in Software Process Improvement. *Empirical Software Engineering*, Vol. 5, 357-390 (2000)
12. Goldensen, D. R., Herbsleb, J. D.: After the Appraisal: A Systematic Survey of Process Improvement, Its Benefits, and Factors that Influence Success. CMU/SEI-95-TR-009. Software Engineering Institute. Pittsburgh. (1995)
13. Debou, C.: Goal-Based Software Process Improvement Planning. In: *Better software practice for business benefit: Principles and experience*. Messnarz, R. & Tully, C. (ed.) IEEE Computer Society: Los Alamitos, CA. 107--150 (1999)
14. Salo, O.: Enabling Software Process Improvement in Agile Software Development Teams and Organisations, PHD Thesis, Oulu, (2007)
15. Bach, J.: The Immaturity of CMM, *American Programmer Magazine* Vol 3, 7--8 (1994)
16. Vodde, B.: Nokia Networks and Agile Development. In *Proceedings of EuroMicro Conference*. (2006)
17. Schwaber, K., Beedle, M.: *Agile Software Development with Scrum*, Prentice Hall (2000)
18. Sutherland, J.: Money for nothing: And your change for free – Agile Contracts. In *Proceedings of Agile Development Conference* (2008)
19. Beck, K.: *Extreme Programming Explained: Embrace Change*, Addison Wesley, (2000)
20. Fowler, M.: Put your process on a diet. In *Software Development*. 8(12), 32--36 (2000)
21. Mnkandla, E., Dwolatzky, B.: A survey of agile methodologies. In *Transactions of the South Africa Institute of Electrical Engineers*. 95(4), 236—247 (2004)
22. J. Sutherland, Nokia Test: aka the scrumbutt test. Available in: <http://jeffsutherland.com/scrum/nokiatest.pdf>
23. SEI.: Capability Maturity Model® Integration for Acquisition (CMMI ACQ), Version 1.2. Carnegie Mellon Software Engineering Institute. (2007)