FEDERAL UNIVERSITY OF PAMPA

Miguel da Silva Ecar

AutoCosmic: Platform for COSMIC Automated Estimation and Management.

Alegrete
2017

Miguel da Silva Ecar

AutoCosmic: Platform for COSMIC Automated Estimation and Management.

Term Paper presented in Software Engineering Graduation Course of Federal University of Pampa as partial requirement for title getting of Software Engineering Bachelor

Supervisor: Prof. Me. João Pablo Silva da Silva

Alegrete
2017

Miguel da Silva Ecar

AutoCosmic: Platform for COSMIC Automated Estimation and Management.

Term Paper presented in Software Engineering Graduation Course of Federal University of Pampa as partial requirement for title getting of Software Engineering Bachelor

Term Paper defendido e aprovado em 27 de NOVEMBRO de 2017
Banca examinadora:

Prof. Me. João Pablo Silva da Silva
Orientador
UNIPAMPA

Prof. Dr. Fabio Natanael Kepler
UNIPAMPA

Prof. Dr. Maicon Bernardino da Silveira
UNIPAMPA

This work is dedicated primarily to God, because it is essential in my life, the author of my destiny, my guide, help present in the hours of distress, my family and my friends, who keeps me standing.

ACKNOWLEDGEMENTS

I would like to God that give me healthy, strength and wisdom to overcome all difficulties.

A thank to the Institution by the creative environment it is friendly provides.

A special thank to the supervisor, for the dedication dedicated to the elaboration of this work.

A special thank to my family, out of love, I encourage unconditional support.

Last but not least, I would like to thank all who directly or indirectly were part of my training, thank you very much.

There are no facts,
only interpretations.
(Friedrich Nietzsche)

ABSTRACT

Early cost estimation has significant importance in software development. In agile development cost estimation is done using empirical techniques, for example, story points that come from planning poker game. Functional size measurement is a method which considers the functional size based on a manual with specifics guidelines for the functional user point of view. Thus, combining these two techniques, agile development and functional size measurement, it is possible to have more precise cost estimation with less dependence of personal experience. We propose an automated tool which has as input user stories, and as output a precise estimation based on COSMIC function points. In order to have this estimation we propose a mix of techniques. Natural language processing to analyze the textual requirement. A context-free grammar to structure the user story and a dictionary of verbs, that represents functional processes. This work presents a tool where it is possible to manage a software agile project having as bases the automated user story estimated size. Furthermore, a user story parallel corpora annotation tool is embedded inside the tool.

Key-words: COSMIC. Functional Size Measurement. Automated Tool. Agile Development. User Story. Natural Language processing.

Resumo

A estimativa de custo inicial tem bastante importancia no desenvolvimento de software. Em ambientes de desenvolviemnto ágil, a estimativa de custo é feita geralmente baseada em técnicas empíricas. Por exemplo, *story points* que é baseado em *planning poker*. Medição do tamanho functional é um método que considera o tamanho funcional de uma plicação baseada em um manual com um guia específico para encontrar as funcionalidades do ponto de vista do usuário. Então, combinando estas duas técnicas, desenvolvimento ágil e medição do tamanho funcional, é possível ter uma estimativa de custo mais precisa, com uma menor dependência da experiência pessoal do profissional. Foi proposta uma ferramenta automática que recebe como entrada um conjunto de *user stories*, e retorna o tamanho funcional de cada uma delas baseada no método do COSMIC. Para tanto, foi proposto um mix de técnicas, processamento de linguagem natural para analizar o texto do requisito. Foi definida uma gramática livre de contexto para estruturar a *user story* e um dicionário de verbos que representam os processos funcionais. Foi desenvolvida uma plataforma onde é possivel gerenciar projetos de software baseado na estimativa automática das *user stories*. Além disso, foi também desenvolvido dentro desta plataforma um anotador de corpos paralelo para *user stories* e pontos de função COSMIC.

Key-words: COSMIC. Medição do Tamanho Funcional. Ferramente Automática. Desenvolvimento Ágil. User Story, Processamento de Linguagem Natural.

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

BPMN Business Process Model and Notation

CFP COSMIC Function Point

COSMIC Common Software Measurement International Consortium ISO/IEC 19761:2011

CUSS Cosmic User Story Standard

FiSMA Finnish Software Metrics Association ISO/IEC 29881:2010

FP Function Point

FSM Functional Size Measurement

IFPUG International Function Point Users Group ISO/IEC 20926:2009

ISO International Organization for Standardization

Mk II Mark II Function Point Analysis ISO/IEC 20968:2002

MLR Multivocal Literature Review

NESMA Netherlands Software Metrics users Association ISO/IEC 24570:2005

NLP Natural Language Processing

UPROM Unified Business Process Modeling (BPMod) Methodology

US User Story

CONTENTS

1 INTRODUCTION

Functional Size Measurement (FSM) was proposed in order to obtain better unit of sizing. According to International Organization for Standardization (ISO), FSM are designed to overcome the limitations of earlier methods of software sizing by shifting the focus away from measuring how software is implemented to measuring size in terms of the functions required by the user. Akca e Tarhan (2012) explore that FSM aims at measuring the functionality in software. Being independent of technology, platform and individual, the existence of particularly defined procedures for measurement provide standard, objective, consistent, and comparable results.

The method published in 1979 by Allan Albrecht became known as Function Point Analysis. This method has the characteristic of provide the size of software requirements independently of technology applied. The management of the method was taken by International Function Point Users Group ISO/IEC 20926:2009 (IFPUG) and the method has become known as "IFPUG FPA" (COSMIC, 2014).

There are five methods of FSM approved by ISO, they are Mark II Function Point Analysis ISO/IEC 20968:2002 (Mk II), IFPUG, Common Software Measurement International Consortium ISO/IEC 19761:2011 (COSMIC), Netherlands Software Metrics users Association ISO/IEC 24570:2005 (NESMA), Finnish Software Metrics Association ISO/IEC 29881:2010 (FiSMA). According to COSMIC (2014), the IFPUG method is probably the most used for business application software. But it has been criticized for being inapplicable to any kind of software, particularly it is not well accepted by real-time software community. COSMIC method was born by this measurement necessity and it is the method that we choose to develop this work.

COSMIC is a group formed in 1998 which intended to develop the second generation of FSM. The COSMIC group had the objective to develop and gain market acceptance for a method of measuring the functional user requirements for software based on fundamental software engineering principles and conformity to measurement theory, to be applicable for measuring business, real-time and infrastructure software (COSMIC, 2014). COSMIC is a term for both, the group and the method.

COSMIC method has been designed to accept extensions for particular domains (COSMIC MANUAL, 2015). A domain which it has been extended is Agile development. According to COSMIC Agile (2011), the agile guideline has the purpose of providing additional advice beyond the COSMIC Measurement Manual on Agile projects. The COSMIC method even for new software development or changing an existing software measurement is perfectly suited for measuring software evolving through iterations and increments as typically found in Agile development without any adaptation.

We chose COSMIC method due to its huge applicability and guidelines for application to Agile development and also because it is public shared, so, it is not needed to pay taxes or association role.

1.1 Motivation

Software estimation, in general, takes time to be done. The estimator must have a considerable experience, in order to analyse textual requirements and estimate the size. The client generally wants to know as soon as possible what is the cost of the system and depending on the number of requirements, it must be necessary few days for complete analysis and estimation.

Moreover, a precise estimation may be costly due to the estimator need to have advanced training and experience. Diab et al. (2005) say that the repeatability, or capability to reproduce the same estimation result for the same requirement, power may vary according to the experience of the measurer. Junior engineers show poor repeatability compared to experts. They also mention variation between 11% and 30% among several measurements of the same specification, due to the interpretation of the textual requirements.

We hope with this work, achieve significant level of repeatability, so, percentage of variation on the same product backlog measurement close to zero. Furthermore, we hope decrease the cost of estimated measurement once it is done automatically. We choose User Story (US) because this technique is closer to the client vocabulary and abstraction level. According to the COSMIC Agile (2011), on the one hand textual requirements must be enough for measurement in COSMIC Function Point (CFP), on the other hand, if it is difficult, then the is almost certainly due to weaknesses, even ambiguities or omissions, in the US.

1.2 Objectives

The main objective of this work is to develop a tool for automated estimation of US in CFP. The tool must store user stories and count automatically the number of CFP for each US. This objective is divided in small specific objectives:

- **Exploring COSMIC method:** specially the guideline for applying the method in agile projects;
- **Exploring Natural Language Processing (NLP):** this task involves to study techniques to extract the main concepts related to functional size measurement from user story text;
- **Developing a web tool:** the final delivery of this work is a usable web tool, in which it is possible to manage an agile project using the automated estimation feature.

1.3 Methodology

First of all, we performed a systematic mapping to discover related works of proposal to automated the measurement using COSMIC method. Then we performed a multivocal literature review (Multivocal Literature Review (MLR)), to find other proposal to extend the user story writing template.

Based on the result of MLR we proposed our own extension to write user stories. This extension was designed to fit some gaps found in the original template in terms of COSMIC sizing.

After that, we started the platform development. First, we formalized the user story template extension in a context-free grammar, to do this, we used Xtext framework. The platform has also features for automated user story estimation, dictionary of verbs and user story parallel corpora annotator.

Lastly, we perform a validation routine comparing the automated tool estimation results with manual estimation made by experts and junior professionals.

1.4 Contribution

From this work we can detach three contributions. The first one is the procedure for automated user story estimation in a web tool, in which it is possible to follow and manage the textual requirement.

The second contribution is a new template for user story writing. This template is specific for COSMIC based projects and adds important information in terms of a better and precise estimation.

The third contribution is a parallel corpora annotator for COSMIC and user stories. This corpora is helpful for future works adding the possibility to be source for deep learning techniques to improve the automated estimation.

1.5 Organization

This document is organized according to the following:

- **chapter 2:** details of main concepts related to our work, such as, COSMIC, Neural network and technologies;
- **chapter 3:** presents papers which propose tools for automated COSMIC measurement;
- **chapter 4:** presents the proposed new user story writing template;
- **chapter 5:** describe of the developed tool;
- **chapter 6:** presents final remarks about the presented work.

2 BACKGROUND

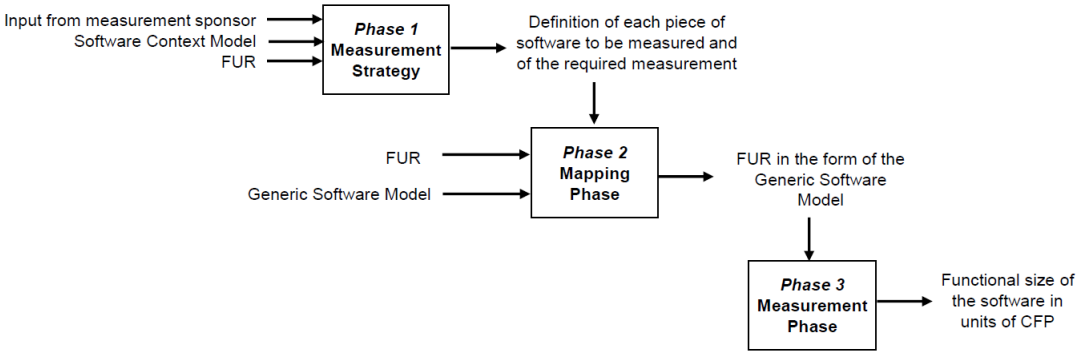
In this chapter, we present theoretical background, such as, COSMIC method, in section 2.1, defined as the second generation of the functional size measurement method. The section 2.2 is about US. We present also the main concepts of natural language processing in section 2.3. The section 2.4 is about context-free grammars. The section 2.5 presents the literature review methods concepts used in this work. Lastly, section 2.6 we present the chapter lessons.

2.1 COSMIC Method

According to COSMIC (2014), the COSMIC method was born from the necessity of measure the functional user requirements of systems such as, business application, real-time and infrastructure software and some types of scientific or engineering software. This necessity comes due to some weaknesses of IFPUG technique. It has become increasingly difficult to map Albrecht's function types to modern ways of modelling software requirements (COSMIC, 2014). The function types that it considers can be given only a very restricted range of sizes (COSMIC, 2014).

The method is divided in three phases, measurement strategy phase, mapping phase and measurement phase. These phases and their definition are shown in the Figure 1.

Figure 1 – The COSMIC measurement process.



Source: COSMIC (2014)

The Measurement Strategy phase is about to define what will be measured. As the method is not specific for human users, it must be defined the view point or who is the functional user, such as, humans, hardware devices or other software which will interact with the software measured. According to COSMIC (2014) there is the need first to define the purpose of measurement which will leads to defining the scope and the functional users. Thus, there are five key strategy parameters to determine in this phase, they are:

- **Purpose:** the purpose helps determine all following parameters;

- **Scope:** defines what is included in the functionality and what is excluded, what are the restrictions;
- **Level of composition:** pieces of software to be measured, for example, the whole application (level 0), or a primary components of distributed system (level 1), or a re-usable component in service-oriented software (level 2);
- **Functional users:** must be defined for each piece of software. They are humans or things which will be send or receive data to or from the software;
- **Layers:** the software architecture, the piece of software must be confined in one layer.

The Mapping phase is about to create the COSMIC model of the functional user requirement, it starts from whatever the artifacs are available, in our case, these artifacts are US. The model is created applying the COSMIC Generic Software Model, which is defined for four principles (COSMIC, 2014).

The first principle is that software functionalities are consisted of functional processes, each functional process respond to an event in the world of functional users.

The second one is, functional processes are consisted of subprocesses, they do only two things, they move and manipulate data. Data movements which move data from functional users into functional processes or vice-versa are called **Entries** and **Exits**. Data movements which move data from persistence storage and vice-versa are called **Writes** and **Reads**. It may be seen in Figure 2.

The third principle says that each data movement moves a single Data Group. Data group is defined as a single object of interest of the functional user.

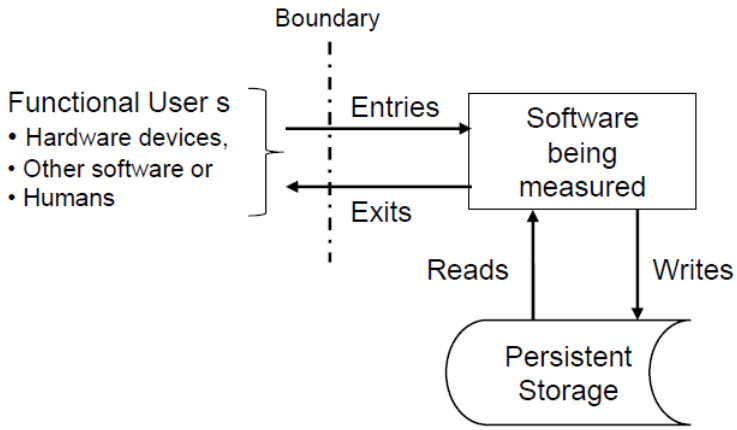
The last principle defines that data manipulation are assumed to be accounted by the data movement with which they are associated. Data manipulation are not measured separately.

The Measurement Phase is about to take account of data movements, each data movement is a unit of CFP. In this phase, they are counted and sum over all functional processes. A single functional processes must be at least two data movements and there is no upper limit to the size. When measuring an enhancement of existing software, it must be identified all data movements added, changed and deleted and sum them over all its functional processes. The minimum of any modification is 1 CFP (COSMIC, 2014).

Sizing software in Agile development requires exactly the same knowledge, principles and rules of COSMIC method when used in any other project management method (COSMIC AGILE, 2011).

When sizing US, Message Sequence Diagram may be used as shown in Figure 3, The vertical line represents a functional process and horizontal arrows represent data movements. Entries and Reads are shown as arrows incoming to functional process and

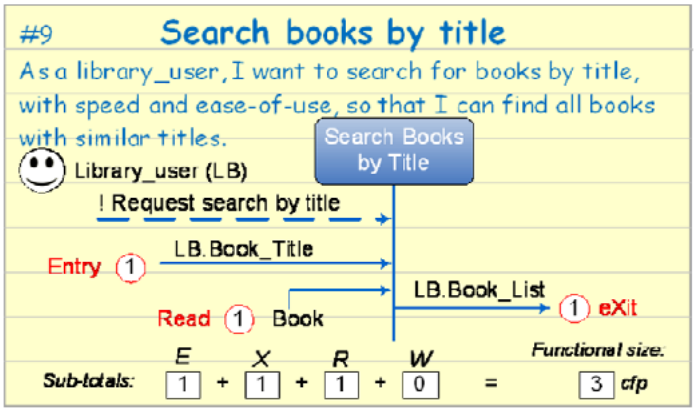
Figure 2 – Data Movements.



Source: COSMIC (2014)

Exit and Writes as outgoing arrows, appearing in the required sequence as top-down order (COSMIC AGILE, 2011).

Figure 3 – User Story and Message Sequence Diagram.



Source: COSMIC Agile (2011)

2.2 User Story

User Story (US) is a technique widely used in Agile development. They are characterized as short and high level description of required functionality written in customer language. The general form of US is (ABRAN et al., 2004):

"As a <role>, I want to <goal/desire>, so that <benefit>".

US is used in very early of requirement gathering. It intend to contain just enough information in order to be able to produce estimate effort for implementation. A procedure it indicated to be written by the customer before the implementation for appropriated acceptance.

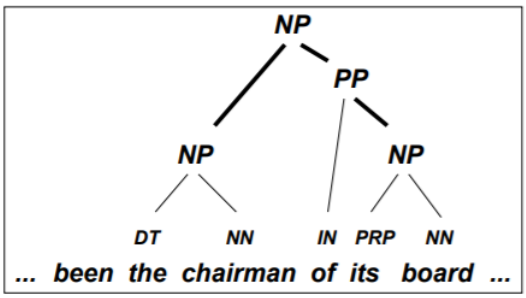
2.3 Natural Language Processing

NLP is an area of research and application that explores how computers can be used to understand and manipulate natural language text or speech to do useful things (CHOWDHURY, 2003).

Based on this research area, it is possible to extract concepts from natural language text and manipulate it. It is also possible to identify semantics, syntax, grammatical class among others.

Inside the NLP area, we used Syntactic Parse Tree to extract the main concepts for our work. Syntactic parse tree is an structure that represents a context free language. The Figure 4 shows an example of syntactic parse tree where the elements in the figure represent `DT - Determiner`, `NP - Noun Phrase`, `PP - Prepositional Phrase`, `NN - Noun, singular or mass`, `IN - Preposition or subordinating conjunction` and `PRP - Personal pronoun`.

Figure 4 – Syntactic Parse Tree.



Source: Kambhatla (2004)

2.4 Context-Free Grammar

A context-free grammar is a collection of context-free phrase structure rules. Each such rule names a constituent type and specifies a possible expansion thereof (BUNDY; WALLEN, 1984).

A context-free grammar is defined by 4-tuple `G=(N, E, P, S)`, where `N` is the set of non-terminal symbols, `E` is the set of terminals, `P` is the finite list of production rules and `S` is a non-terminal called initial symbol (RODRIGUES; LOPES, 2007).

Based on a context-free grammar it is possible to determinate if a sentence belongs to a context-free language or not (RODRIGUES; LOPES, 2007).

2.5 Literature Review Methods

In this section we present the two methods used to make literature review. The subsection 2.5.1 presents how systematic mapping is performed and MLR is presented in

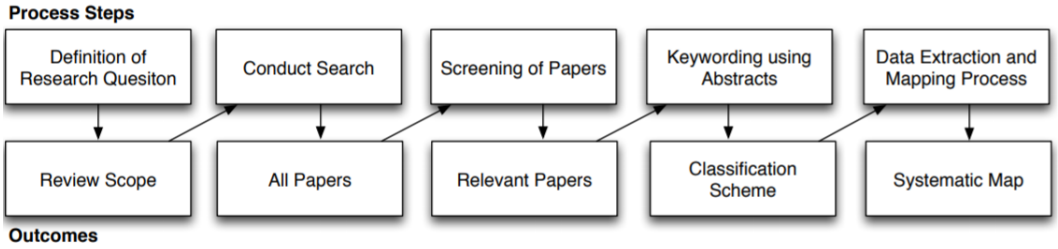
subsection 2.5.2.

2.5.1 Systematic Mapping Study

According to Petersen et al. (2008) "A systematic mapping study provides a structure of the type of research reports and results that have been published by categorizing them and often gives a visual summary, the map, of its results."

The Figure 5 shows a common systematic mapping process. This king of study has vantages of being well defined and reproducible. Thus, if anyone else follows the same protocol it tends to find the same result. First of all, it is need to define the research question, the question that the final set of papers should answer. Then, the search is conducted defining the search string and search engines. Next step is to select the papers according to defined including and excluding criteria. The last step is extracting the data which answer the research question.

Figure 5 – Systematic Mapping Process.



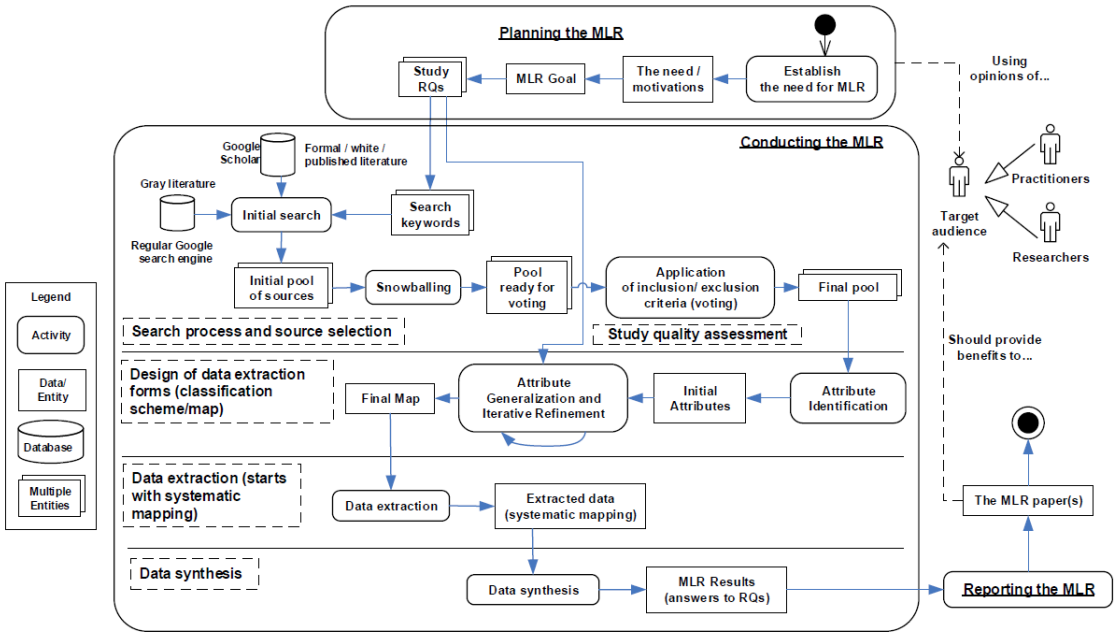
Source: Petersen et al. (2008)

2.5.2 Multivocal Literature Review

Multivocal Literature Review MLR (GAROUSI; FELDERER; MÄNTYLÄ, 2017) is a form of a Systematic Literature Review (SLR) which includes not just formal literature (e.g., journal and conference papers), but also grey literature, such as, blog posts and white papers (GAROUSI; FELDERER; MÄNTYLÄ, 2017). MLRs are useful for researchers and practitioners since they provide summaries the state-of-the-art and practice in a given area (GAROUSI; FELDERER; MÄNTYLÄ, 2017).

The general MLR process is presented in Figure 6. Basically it follows the same step of a systematic mapping, but after that, the search for gray literature should be conducted. The search for gray literature should follows the same process as a common systematic review, including and excluding criteria are applied taking in account the characteristics of gray literature, such as, informal writing, weak of references among others.

Figure 6 – Multivocal Literature Review Process.



Source: Garousi, Felderer e Mäntylä (2017)

2.6 Chapter Lessons

In this chapter we explored main concepts of our work. The COSMIC method is a good choice, once it fits very well in Agile development and has guidelines for its application in this context. We also presented the natural language processing techniques used to extract elements from plain text.

3 RELATED WORK

In this chapter, we present a systematic mapping of related literature. Systematic mapping has been used in order to investigate the proposal of automated tools for COSMIC function point measurement. In section 3.1 is presented the systematic mapping protocol. In section 3.2 is presented an analysis of selected papers according to the objective. In section 3.4 is presented chapter lessons.

3.1 Systematic Mapping Protocol

We performed a systematic mapping to investigate proposals of automated tools for COSMIC function point measurement. Three research questions have been proposed in order to filter the result.

- What are entry data for automated measurement?
- What methodology or technique is used?
- What is the objective, estimate or counting?

The research engine chosen was Google Scholar. The research has been done by the following key-words, "COSMIC function point" AND automated AND tool. In order to filter the large amount of resulting papers, we determined including and excluding criteria.

We considered as including criteria, full research papers which present tools for automating counting or estimate COSMIC function points and papers published between 2010 and 2015, this interval was set to get only the last and more relevant papers. As excluding criteria, we disregarded systematic mapping, review or survey, book chapters, master thesis and papers not written in English.

3.2 Systematic Mapping Result

The resulting amount of papers returned by this research was 60. After that, the selection was done considering papers which present some type of automated tool for counting or estimate based on COSMIC function points. Thus, we selected 7 papers, presented in Table 1, which have a proposal for automated tools. The result list of papers were organized in two groups, subsection 3.2.1 presents automated tools for estimation papers and subsection 3.2.2 presents automated tools for counting papers.

3.2.1 Automated Tools for Estimation

Marín e Quinteros (2014) approach an automated tool for estimate CFP having as bases Business Process Model and Notation (BPMN) diagrams. They present a procedure called E-FSMP, meaning Early Functional Size Measurement Procedure, and an

Table 1 – List of Selected Papers.

Title	**Author**
A COSMIC Measurement Procedure for BPMN Diagrams	Marín e Quinteros (2014)
Analytical Convertibility of Functional Size Measures: a Tool-based Approach	Lavazza, Bianco e Liu (2012)
Applying Reverse Engineering Techniques to Verify the Estimation of Software Code Size using COSMIC Full Function Point	Yeh Yi-Hong Chen (2015)
Approximation of COSMIC functional size to support early effort estimation in Agile	Hussain, Kosseim e Ormandjieva (2013)
Automated Functional Size Estimation using Business Process Models with UPROM Method	Aysolmaz e Demirors (2014)
Measuring COSMIC Software Size from Functional Execution Traces of Java Business Applications	Sag e Tarhan (2014)
Run-time measurement of COSMIC functional size for Java business applications: Initial results	Akca e Tarhan (2012)

Source: Author.

automated tool which implements this procedure. As COSMIC has three phases, the procedure has three phases also, they are, Measurement strategy phase, Mapping phase and Measurement phase. In Measurement strategy it is applied two rules in order to defined the strategy of effort by the definition of the purpose, scope level of granularity and functional users. After, in Mapping phase, is it applied three rules where functional processes and data groups are identified. In Measurement phase, it is applied six rules for data movement and each functional identification. Thus, they implement a web based tool, which allows measurement by BPMN or use case diagrams. The tool works submitting a XML file with the specification of BPMN diagram.

Hussain, Kosseim e Ormandjieva (2013) approach an attempt to automatically estimate the functional size measured by COSMIC method by textual requirements. They argue that requirements in agile development are written less formally in comparison with Use Cases. This lack of formalism restricts functional size measurement methods. Thus, in their work they present an alternative solution to estimate COSMIC functional size by informally written textual requirements. They classify functional processes in four classes: small, medium, large and complex. Then select a set of textual requirements and training the text mining algorithm to classify the requirement in a class. This training is done by syntactic features, keyword features and extraction and classification feature. They conclude based on statistics that their work shows plausible classification from textual requirement using text mining and confirm that their methodology is effective in approximating the size of functional processes.

Aysolmaz e Demirors (2014) propose an automated method based on UPROM notation. In the paper, they present the UPROM FSE (Function Size Estimation) method

and describe mapping rules and conversion to estimate software size in CFP. In UPROM there are seven operation types based on CRUDL to express user requirements. These operation types, related base CRUDL operation and data movements conversion are shown in Table 2.

Table 2 – UPROM conversion from operation type to Data Movement.

Operation Type	**Base CRUDL Operation**	**Data Movements**
Crete	Create	E, W
Change	Update, List	E, R, W, X
Delete	Delete	E, W
View	List	E, R, X
List	List	E, R, X
Read	Read	R
Use	Read	R

Source: Aysolmaz e Demirors (2014).

3.2.2 Automated Tools for Implemented Artifact

Yeh Yi-Hong Chen (2015) present a tool that reverse engineers the Java source code to detailed sequence diagrams and calculate the function points. They use rules for function points calculation proposed by Jenner (2001), where arrows directing from actors to interface objects correspond to CFP entries. Arrows emitting from interface objects to actors correspond to CFP exits. Arrows between objects in the system correspond to CFP reads or writes, arrows representing returning values are not considered. Then, a regression analysis is performed on the values calculated and the corresponding lines of code. Thus, it would be possible to know how many numbers of LoC are approximately equal one unit of CFP. The result shows that depending on the kind of application, the accuracy of COSMIC method is variable. The model can be applied to systems without high level of complex decisions and computations.

Akca e Tarhan (2012) present a run-time application for COSMIC function size measurement. They developed a Measurement Library which has specific methods for functional measurement. This approach was applied in a three-tier application which the Measurement Library is imported. Specific methods of this library are called, in the main application methods, such as, action methods of the buttons, methods which execute queries for database communication, etc. In their application example the automatic and manual measurement were 92% convergent.

Sag e Tarhan (2014) approach the possibility of measuring functional size from source or binary code automatically. In their method, they convert the code to UML

sequence diagrams derived from functional execution traces of the software at runtime. The method is divided in four steps, they are: Prepare Pointcuts, Perform Dynamic Analysis, both steps, using aspect oriented framework, Generate structured text version of sequences, to get functional process sequence, and Apply COSMIC Measurement, running COSMIC measurement rules to calculate the functional size. They also developed a tool prototype in order to demonstrate their automated method. After execution of the prototype in a sample application, they conclude that automatic counting is convergent with manual counting by specialist.

Lavazza, Bianco e Liu (2012) present a method to convert CFP into Function Point (FP) (IFPUG unadjusted Function Points) and vice-versa. They also, present a supporting tool which implements this conversion method automatically. The method consists in analysing the common aspect between both functional measurement approaches ans establishing the correspondence between both. For example, RET's and Logical Data File in IFPUG Function points can correspond to Data Group in COSMIC, or set a DET crossing the application boundary in a transaction or FTR access within a transaction correspond to Data Movements. The case study applied shows that the approach is effective, however the identification of data groups and functional processes is more immediate than identification of the type of data movements. For this they propose double measurement or to explore all details of recorded transactions as solution strategies.

3.3 Threats to Validity

Based on the mentioned protocol we can detach some threats to validity. The first one is that the systematic mapping was performed using only google scholar. The reason is that scholar index other research bases, so, it was chosen to minimize the time effort.

To mitigate the comprehensiveness problem caused by search only in one secondary base, we performed a snowballing in each result. This was made to guarantee that the biggest number of related works were found.

There is also a threat that the systematic mapping was performed only by one researcher. This threat was accepted, once, this work is part of a term paper and there were no other researcher related to this.

The last threat is the research range, this range was set due to get only the latest approaches for automated FSM sizing. Taking in account that older approaches might be useless and out of date.

3.4 Chapter Lessons

Considering the mapping protocol applied, we could realize that just one paper is about automated measurement of COSMIC from textual requirements. This one uses

basically Use Cases as inputs, there is no approach for User Stories. Other papers which have approaches for estimation, use diagrams such as, BPMN or UPROM.

We also observed that papers interested in automated measurement are about use source code as input. Even adding a counting library in the developing application or converting the source code to sequence diagrams then measuring them. There is also an approach which intend to automatically convert CFP to FP and vice-versa and according to them this conversion is sufficient accurate.

Analysing these papers, we could understand basic concepts about automated tools and its importance. Moreover, the relevance of early size estimation in the software development life cycle. Furthermore, the advantages FSM as estimation technique instead of empirical ones.

4 COSMIC USER STORY STANDARD

This chapter relates the why and how we found out that a new user story writing template was needed. We conducted a MLR shown in section 4.1 to find other approaches to write user stories. In section 4.2 we propose a new one the fits the gaps found in other templates. The section 4.3 show how the new template was evaluated. Finally, in section 5.8 we present chapter lessons.

4.1 Current User Story Writing Templates

Traditional User Story template, even expressing basic information that is enough for COSMIC sizing, it does not support some key points for a more precise estimation. For example, connections between data groups in a functional process and feedback to functional user clearly.

Considering that beyond traditional template, there are several other ones for US writing, we conducted a MLR to find the biggest number of proposed templates then we analyzed all of them in terms of COSMIC sizing.

The MLR protocol was designed based on one research objective: "Finding the current user story models and/or templates in traditional and gray literature." So, we defined one research question: "Does the paper propose a template or model for user story writing?". The research enginees were Scopus, IEEE Xplore, SpringerLinkl, ACM DL, ScienceDirect and Compendex. The generic search string was: `TITLE((user story) OR (user stories)) AND KEY((user story) OR (user stories))`.

We defined two including criteria: "Paper must show a user story template/model." and "Paper must propose a user story template/model.". and two excluding criteria: "Paper not written in English." and "Paper must propose a user story template/model.".

Moreover, we defined one quality assessment: "Does the paper propose or cite a user story template?" and four possible answers "Paper which propose an US template" with weight 1.0; "Paper which cite an US template" with weight 0.5; "Paper does not propose or cite an US template" with weight -0.5; and "Full paper is Inaccessible" with weight -1.0"

Lastly we extracted two data types "Proposed user story template/model." and "User story template/model reference.".

The first step is the search in each search engine. Second step is detection and elimination of duplicated papers, the third step is application of including and excluding criteria. The fourth step is the quality assessment and the last step is consideration of papers with weight higher than 0.5, in other words, papers which propose an US template. Table 3 presents the number of result for each search base in the table on the left and detailed number of paper after each step is in the Table on the right. The final list of paper is presented in Table 4. The Table 5 shows with ID "FL" templates found.

After finding these result, we conducted the second search which is considering

Table 3 – Papers returned for each search base and in each step.

Search Base	Papers
ACM Digital Library	14
El Compendex	89
IEEE Digital Library	15
Science@Direct	4
Scopus	71
Springer Link	0
Total	**193**

Source: Author.

All results -	193
After duplicated detection -	100
Accepted -	74
Higher than 0 -	20
Higher than .5 -	3

Source: Author.

Table 4 – Papers from First result.

ID	Paper Name
FL01	Agile user stories enriched with usability (MORENO; YAGÜE, 2012)
FL02	User stories template for object-oriented applications (ZEAARAOUI et al., 2013)
FL03	UserX story: Incorporating UX aspects into user stories elaboration (CHOMA; ZAINA; BERALDO, 2016)

Source: Author.

the gray literature. The protocol is basic the same, but with changing in search engine, that was replace by Google. The including and excluding criteria, quality assessment and data extraction were kept the same.

Unlike common Systematic Literature Review, different stopping criteria for gray literature searches are needed in MLR (GAROUSI; FELDERER; MÄNTYLÄ, 2017). We defined stopping criteria the first ten pages in Google search, where each page shows 10 results, since there were evidence of exhaustion or saturation.

The search string used in Google was *“"user story" template model standard example”*. This string returned 99.600 results. The resulting pages which a different US template was found were investigated until the root reference of this template (Snowballing), as happened in Wikipedia. The result of this search is presented in Table 5 with ID “GL”.

4.1.1 MLR Result

The final results are presented in Table 5. User story templates found in SLR are with ID FL, templates found in Google search are with ID GL and previous known template is with ID KL.

The template FL01 (MORENO; YAGÜE, 2012) does not present so much contribution, just introduce and specific use of the user stories to express usability requirements, so in comparison with traditional template there is no relevant difference.

As mentioned before, the template FL02 (ZEAARAOUI et al., 2013) does not also present significant difference, just explicit that the object must be present in the user story, which is a natural practice if you are using the traditional template idea.

Template FL03 (CHOMA; ZAINA; BERALDO, 2016) present contribution in terms of usability engineering. The last part which is responsible for express the feedback or expectation from the user point of view in traditional template, in FL03 this part is specific for Nielsen's heuristic values, which should be met in the user story.

Template GL01 (COHN, 2008), found in gray literature, was introduced setting the last part of traditional template as optional. Considering this, the user story is shorter and does not provide information about feedback or user expectation.

Template GL02 (MATTS, 2011) takes the received benefit, which is present as last part in traditional template, and put it as first part, before the user, then it follows the traditional approach.

Template GL03 (PUPEK, 2008) is also known as "Five Ws", it is added with more relevant information from the user perspective. The added information are exactly, "when" and "where", the other "Ws" have already been present in traditional template.

Template KL01 was found in COSMIC Agile Guide (COSMIC AGILE, 2011), the added information is precisely about non-functional requirement found in user story functional requirement. The guide goal was not propose this template but, the root reference for that was not found. Considering the counting example presented in the guide, this added information is not relevant for sizing estimation.

Table 5 – List of Templates.

ID	**User Story Template**
FL01	"As a <role>, I want <usability requirement>" (MORENO; YAGÜE, 2012)
FL02	"As a <role>, I want to <action> <object>, so that <business value>" (ZEAARAOUI et al., 2013)
FL03	"As a <persona>, I want/need <goal> so that <Nielsen's heuristic> will be met" (CHOMA; ZAINA; BERALDO, 2016)
GL01	"As a <role>, I want <goal/desire>" (COHN, 2008)
GL02	"In order to <receive benefit> as a <role>, I want <goal/desire>" (MATTS, 2011)
GL03	"As <who> <when> <where>, I <what> because <why>." (PUPEK, 2008)
KL01	"As a <role>, I want to <goal/desire>, <non-functional requirement>, so that <benefit>" (COSMIC AGILE, 2011)

Source: Author.

4.1.2 Result Analysis in Terms of COSMIC

In terms of COSMIC sizing, the biggest part of found templates which add, in fact, modifications to traditional template, does not have significant contribution.

Templates which add usability information, provide valuable information for usability engineers, but it is more related to non-functional requirements, while COSMIC is a functional measurement method.

The "Five Ws" template, add few relevant information in terms of COSMIC sizing. Information of when the requirement should happen or be available and the information of where the user should run the requirement present in the user story, are in fact valuable for COSMIC estimators.

Despite this, there is still a gap of information for COSMIC sizing. The movements identification in a functional process are direct connected with the linked data groups or entities, present in the system.

4.1.3 Threats to Validity

Considering that this work presented in the paper is based on multivocal literature review, there are threats to validity mapped.

We performed a snowballing, in each gray literature that presented a new template and in the papers found in SLR, to find the root proposal of each US template.

Threats to validity the best result in gray literature. As mentioned in (GAROUSI; FELDERER; MÄNTYLÄ, 2017), stop rules should be stated among the huge result number in a Google search. We stated as considering the first 10 pages, with 10 links each page. In case of we perceive no saturation at this point, more pages would be considered, but in fact, results were found in only in the first page, so the following 9 pages did not presented any new result.

Another threat to validity, is that study was performed by only one researcher. In order to minimize this threat, we decided to perform a multivocal literature review instead of a systematic literature review. By this, any other result not considered in SLR could be found in gray literature step, moreover, a snowballing was performed in each resulting paper, book, or web page.

In order to avoid Google shows result based on user experience, the search was performed in an incognito window of chrome browser, therefore, the search result tend to be replicable.

4.2 Proposed Template

Based on user story templates shown in Table 5, we can observe some weaknesses and limitation.

First of all, none of existent templates provide information about connections between data groups or entities. This information is important in terms of COSMIC estimation, once data movements are detected also considering connections or links present in the functional process.

Furthermore, current templates, besides providing a place for result or user expectation, it is not specific for feedback. Thus, it may be used to express user expectation, which might not be related to the functional process, for example, *"As a user, I want to receive by email daily news, so that I am always up to date to the news."*

We propose a new template called COSMIC User Story Standard (CUSS). The main improvement is adding information about connections of the data group manipulated in the functional process with other data groups retrieved from other layers. Another improvement is related to user feedback. Some functional processes provide feedback to user, while others may not provide, so it is clearly presented in US.

The CUSS template is the following:

"As a <who/role>, I want to <what>, linked to <connections>; so/then be notified about operation status."

where:

- <who/role> is the Functional User;
- <what> is the verb representing the action or the functional process;
- <connections> represents other data groups involved in this functional process; and
- "so/then be notified about operation status" is optional and represents the feedback to user.

The corresponding "context-free grammar" is presented in Figure 7. The grammar is defined as `G = (V, E, R, S)`

`V` is the set of non-terminal symbols, that are `US`, `I`, `L`, `AS`, `IWANT`, `LINK`, `FEED`, `FBACK`, `PLINK`, `SLINK`, `TLINK`, `DOT`, `COMMA`, `SEMICOLON`, `USER`, `METHOD`, `DG`.

`E` is the set of terminal symbols, that are `As a`, `As an`, `As the`, `I want to`, `I can`, `connected to`, `so`, `then`, `.`, `,`, `;`, `and`, `be informed about operation status`, plus any words representing a role, a verb, or a data group.

`R` is the finite relation where $(V \cup E)$* where members of R are the grammar productions.

`S` is the initial symbol, that is `US`.

Examples are presented bellow.

- `As a` Manager, `I want to` remove a book.
- `As an` user, `I can` update books; `so be notified about operation status.`

Figure 7 – Context-free Grammar.

```
US → AS USER I DOT | AS USER I L DOT |
      AS USER I FEED DOT |
      AS USER I L FEED DOT
I → IWANT METHOD DG
L → LINK PLINK |
     LINK PLINK SLINK TLINK |
     LINK PLINK TLINK

AS → As a | As an | As the
IWANT → COMMA i want to | COMMA i can
LINK → COMMA connected to

FEED → SEMICOLON so | SEMICOLON then
FBACK → FEED be informed about operation status

PLINK → DG
SLINK → COMMA PLINK | COMMA PLINK SLINK
TLINK → and DG

DOT → .
COMMA → ,
SEMICOLON → ;

USER → <list of roles>
METHOD → <list of verbs/actions>
DG → <list of data groups>
```

Source: Author.

- `As a` Manager, `I want to` add a new book, `connected to` author.
- `As the` Manager, `I want to` save books, `connected to` author `and` publishing company.
- `As a` Manager, `I want to` create books, `connected to` author `and` publishing company; `then be notified about operation status.`

4.3 CUSS Evaluation

The survey was created to validate the US template proposal. The survey is divided in 3 parts. The first part is composed by questions to identify the respondent profile. The second part is composed by open questions about current templates, the proposed one and the impact in the agile aspect. The last part are closed questions in likert scale (LIKERT, 1932), where 0 is “Disagree Strongly” and 4 is “Agree Strongly” about the same questions from second part.

The survey had 22 responses and we starts the answer analysis presenting the respondent profile. The Table 6 shows the respondents relation of certified holders and years of experience with COSMIC.

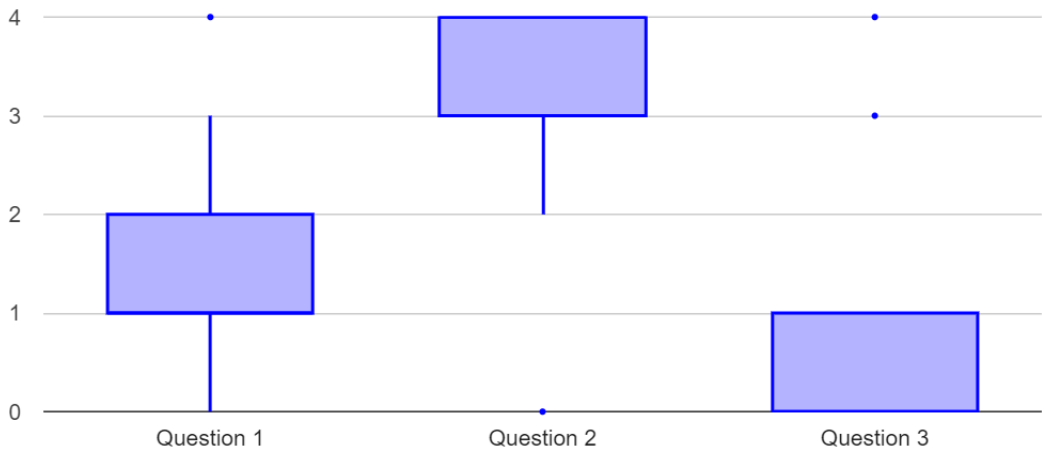
Table 6 – Respondents distribution into groups.

	Certified	Non Certified
1 Year	0	1
2 Years	1	1
3 Years	4	0
4 Years	5	0
5 Years	5	0
6 Years	1	0
8 Years	2	0
10 Years	1	0
15 Years	1	0
	20	**2**

Source: Author.

Skipping to the survey second part, Figure 8 show the boxplot in likert scale for the three statements, Question 01 is "Current Templates provide enough information in term of COSMIC Sizing.". Question 02 is "COSMIC User Story Standard provides greater expressiveness in term of COSMIC Sizing.". Question 03 is "COSMIC User Story Standard Template compromises the agility aspect in a process.".

Figure 8 – First Statement



Source: Author.

Based on the chart in Figure 8, we can observe that concerning to current US templates expressiveness, the concentration is around disagree moderately with an outlyer in agree strongly, in other words, we can infer that current US templates do not have good expressiveness in terms of COSMIC sizing.

Moreover, it is observable that regarding to expressiveness increase of CUSS the concentration is in agree strongly, based on this, we can conclude that CUSS, is a step forward to have a better US COSMIC size estimation.

Further more, in respect to CUSS compromising the agile aspect, the chart shows that concentration is around disagree strongly, in other words, we can conclude that there is no heaviness, from agile point of view, in the information added in COSMIC User Story Standard.

Returning to the second part of the survey, the open questions analysis, was based on two perspectives. We conducted a content analysis in order to analyze the answers with a deep interpretation. The second perspective is a sentiment analysis.

Sentiment analysis is the task of identifying positive and negative opinions, emotions, and evaluations (WILSON; WIEBE; HOFFMANN, 2005). Sentiment analysis systems are being applied in almost every business and social domain (LIU, 2012). The opinion is fundamental to almost all human activities and is key influencer of our behaviors (LIU, 2012). It has proven useful for companies, recommender systems, and editorial sites to create opinions that consist of subjective expressions extracted from reviews (PANG; LEE, 2004).

The sentiment analysis was performed based on a software called Depechemood, which is a totally automated emotion lexicon by harvesting crowd-sourced affective annotation from a social news network (STAIANO; GUERINI, 2014).

The first open question is “Do you think that there is a lack of information in current User Story Templates? Please discourse about your answer.”, the content analysis is divided in two groups, “Yes” and “No” answers. The “No” answers had not further discourse, so there were no classes in it. The “Yes” answers which had further discourse, were divided in four classes. Table 7 present the content analysis for the first open question. The percentage is over all answers for this question.

Based on answers related to lack of information in current US templates, we can observe that almost 80% of respondents agree that there is gaps in these templates. From this group, around a half respondents see no problem in this information lack, so, around a half from those ones that agree that there is gaps in current US template see that it could provide more information for COSMIC size estimation.

The second open question is “In your opinion, Cosmic User Story Standard helps to identify more movements than other templates? Please discourse about your answer.”. The content analysis, was also, based on “Yes” and “No” answers end their further discourses. The “No” answers were divided in two classes and “Yes” answers were divided also in two

Table 7 – Content Analysis for First Question Responses.

Group	**Class**	**Percent.**
No	-	22.72%
Yes	"Yes, but it is inevitable, expected"	13.63%
Yes	"Yes, it is designed to provide basic information"	27.27%
Yes	"Yes, it misses information"	13.63
Yes	Just Yes	22.72%

Source: Author.

classes as shown in Table 8.

Table 8 – Content Analysis for Second Question Responses.

Group	**Class**	**Percent.**
No	"User Story is useless, there is no relevance in it"	13.63%
No	"I prefer other template"	4.54%
Yes	"Yes, it is a major step forward"	18.18%
Yes	"Yes, it is good, great, I like it"	63.63%

Source: Author.

Based on content analysis of second question, we can conclude that over 80% agree that CUSS helps to identify more movements. From the percentage, around three-quarter classified it as "good" or "great" and one fourth classified it as "a step forward for US COSMIC sizing". According to this, we can conclude that CUSS may be well received by COSMIC community, and in fact, may help in early COSMIC size estimation.

The third questions is "Do you think that this addition of information compromises the agile aspect? Please discourse about your answer." Likewise, we considered "Yes" and "No" answers, and classified its discussion. Table 9 presents the content analysis for third question.

Based on Table 9, about 95% of the respondents believe that information added in proposed template does not compromise the agile aspect of user story, in other words, the information added is light weight and the agile aspect is not corrupted.

The sentiment analysis was performed considering positive and negative sentiments. So, The sentiments of "afraid", "angry", "annoyed", "dont care" and "sad", were classified as negatives and sentiments of "amused", "happy" and "inspired" were classified as positive.

The Figure 9 shows sentiments boxplot for first open question. In this chart, is

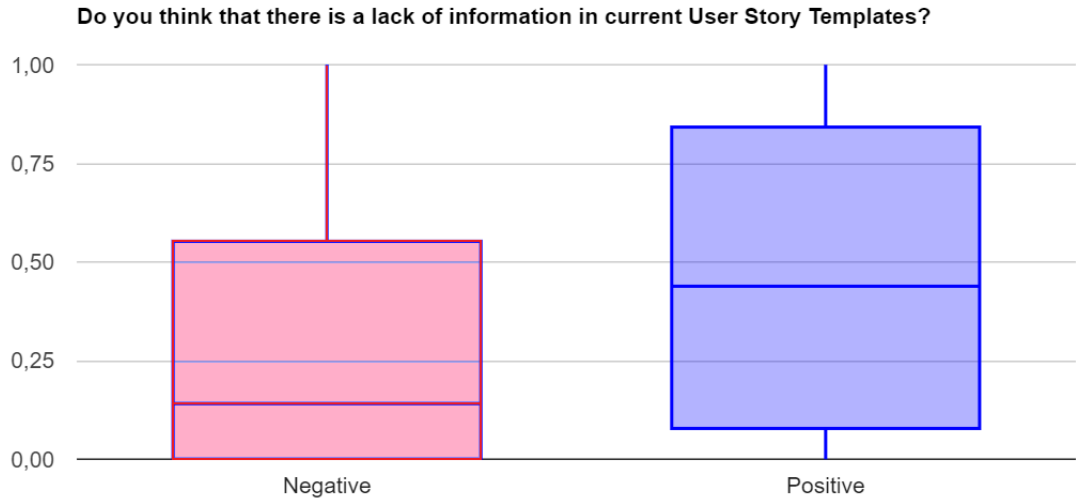
Table 9 – Content Analysis for Third Question Responses.

Group	Class	Percent.
No	-	31.81 %
No	“somehow”, “Why?”	27.27%
No	“there is no relation”, “agile keeps agile”	27.27%
No	“the addition of information is light weight”	9.09%
Yes	“Certainly”	4.54%

Source: Author.

possible to observe high variability to almost all sentiments. Negative sentiments have median under 0.25. Positive sentiments, in which the top box is around 0.75. Moreover, both sentiments have huge variability, negative and positive sentiments had answers from 0 to 1. While in negative sentiments the range of variation (IQR) is between 0 and slightly above 0.5, positive sentiments IQR is between little above 0 and over 0.75.

Figure 9 – First Open Question Sentiment Analysis

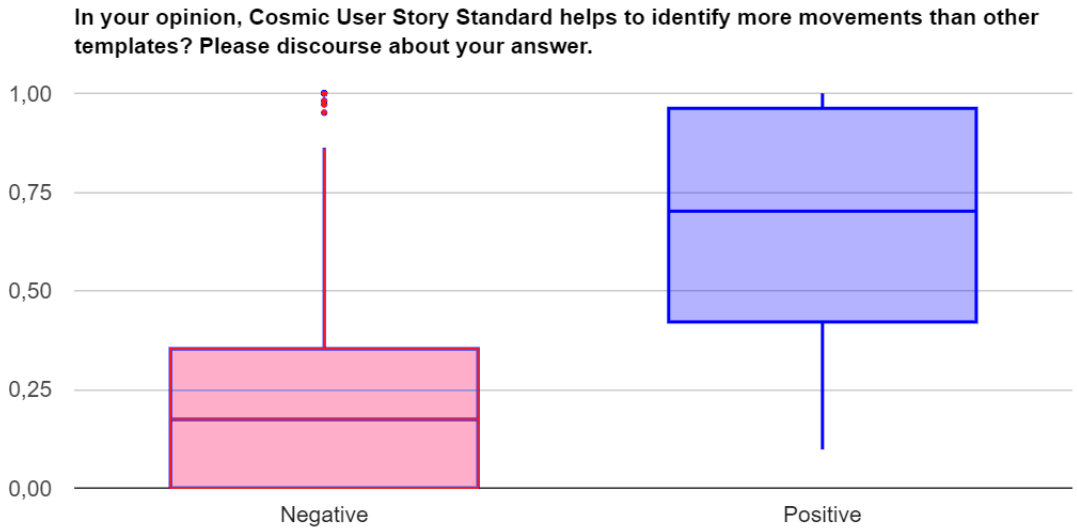


Source: Author.

The sentiment analysis for second open question is shown in Figure 10. This chart shows a major difference between sentiments. We can observe that positive sentiments are tending to top of the chart, while negative sentiments are tending to bottom. Negative sentiments median is below 0.25, while positive sentiments median is slight under 0.75. The IQR has short variability, in negative sentiments IQR is between 0 and slightly over 0.25, while in positive sentiments IQR is slight under 0.5 and close to 1.

The Figure 11 presents boxplot for sentiment analysis of third question. Based on this, we can observe that negative sentiments have significant variability and median is

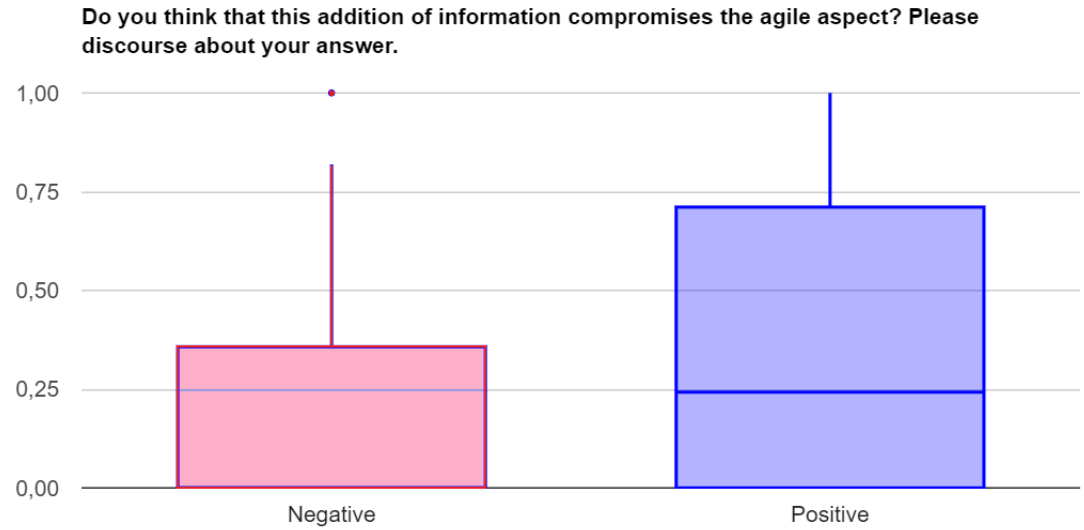
Figure 10 – Second Open Question Sentiment Analysis



Source: Author.

in zero, while positive sentiments has huge variability and median is in 0.25. The IQR for negative sentiments has small variability being between 0 and little above 0.25, while IQR for positive sentiments is between 0 and slightly under 0.75.

Figure 11 – Third Open Question Sentiment Analysis



Source: Author.

The result of sentiment analysis corroborate the content analysis result. Considering the first question, we can observe in content analysis that among 65% of the respondents considered that there is information lack in current user story templates, but it is not a problem, user story is designed to express basic information. Taking this into account, the sentiment analysis shows significant variability between both sentiments.

There is no substantial significance in the difference between positive and negative sentiments (Figure 9).

The content analysis for the second question, among 80% consider that the proposed template is a good improvement in terms of COSMIC sizing, almost 20% considered it a major step forward. In sentiment analysis it can be confirmed looking to positive sentiments, which are more condensed and tending to 1. Negative sentiments have median bellow 0.25 and are, also condensed and tending to 0 (Figure 10).

For the third question, based on the content analysis among 95% of the respondents considered that the information added in proposed user story template does not compromise the agile aspect. Considering this answers, almost 60% sad just "No" or were not clear about what they think, while among 37% considered that it is not a problem. Based on this, the sentiment analysis, shows a huge variability in positive sentiments and significant variability in negative sentiments. Negative sentiments box are tending to 0, while positive sentiments show no tendency. This may denote no interest in the question, thus, if there were commitment in agile aspect the sentiment analysis would show more tendentious result (Figure 11).

4.4 Chapter Lessons

In this chapter we talked about the propose template for user story writing. We performed a multivocal literature review, in order to find proposed templates for user story writing. Furthermore, a specific analysis was made to find the relevance of these templates for COSMIC sizing purposes. The result is that there are several approaches to extend the traditional template for US writing. Huge part of result found in this study are concerned with usability aspects in requirements. The information added or rewritten is concerned to usability information instead of other worries. There was only one result that provide relevant information in terms of COSMIC sizing, The "Five Ws" template. Despite of this, the added information tend to solve two of the information gaps in traditional US template. The more important gap in traditional US template, that are entities linked to requirement expressed in the US are not solved.

Based on this, we presented a new approach for User Story writing. This approach is called COSMIC User Story Standard and has purpose is improve user story expressiveness in terms of COSMIC size estimation. The information added in this template is related to connection between data groups in the system being estimated, and also there is an improvement related to user feedback. The added information is lightweight and do not makes the user story saturated of information.

In order to validate the Cosmic User Story Standard (CUSS), we developed a survey with COSMIC community to explore its possibilities and get feedback from potential users. The survey answers were analyzed under two perspectives, content analysis and sentiment analysis. The result of both is enthusiastic, the template was well received by

COSMIC community.

Content analysis allow us to realize that the information lack in user story is perceived by COSMIC users, but not necessarily it is seen as a huge problem. The sentiment analysis corroborate this result showing that sentiments have high variability and there is no tendency. Moreover, content analysis also allow us to conclude that the proposed template is valuable for COSMIC community being classified as great and a step forward in terms of COSMIC user story estimation. The sentiment analysis certify this conclusion when positive sentiments has tendency to be higher and close to the top while negative ones tend to be lower and close to the bottom.

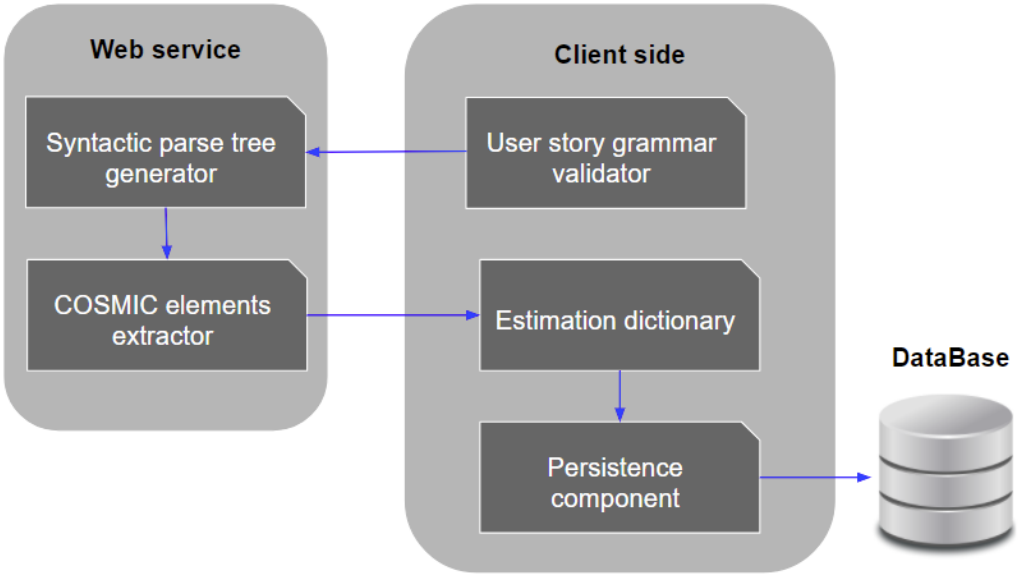
5 AUTOCOSMIC

AutoCosmic is a web tool implemented in Java with PostgreSQL Data-Base. In this chapter we detail the tool overview in section 5.1. The section 5.2, we detail the grammar validator component. In section 5.5 we present the parallel corpora building feature. In section 5.7 we show how the web tool was evaluated. Lastly, in section 5.8 we present chapter lesson.

5.1 Overview

The general architecture of AutoCosmic is shown in Figure 12. It is composed of two applications, a web service and a client. The web service component is responsible for receiving a list of user stories and return this user stories with important elements extracted, these elements are: functional user, data group, connections between data groups, functional process and user feedback. The client side is responsible for user story grammar validation and estimation of the functional process against a dictionary, moreover, the tracking and persistence of user stories. Furthermore, in the client side, the user is able to see the estimation history of each user story.

Figure 12 – AutoCosmic Architecture.

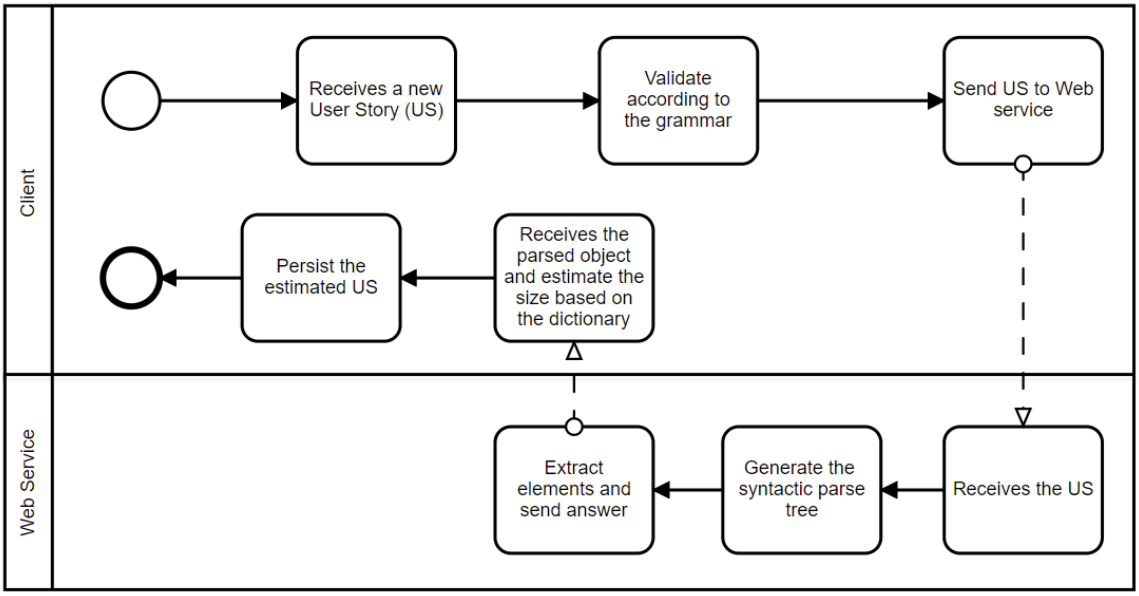


Source: Author.

The tool works as presented in Figure 13. The client component receives a new user story, the component validates if the informed text is valid according to a grammar, then it is send to the web service component. In the web service receives the user story, then it generates the syntactic parse tree and extract the elements, so the user story is returned to the client component. The client component receives the parsed object and

estimate the size of the user story according to provided dictionary. finally, the user story is persisted in the database.

Figure 13 – AutoCosmic Workflow.



Source: Author.

5.2 User Story Grammar Validator

AutoCosmic is powered by a grammar validator, which automatically verify if the user story correspond to a specific grammar. This validation is performed by the Xtext framework (BEHRENS et al., 2008). Xtext is an Eclipse based framework for Domain Specific Language (DSL) building. The Figure 14 shows the xtext grammar based on Figure 7 in section 4.2.

5.3 Parser Component

The parser component is a web service which receives a string in JSON format containing a list of user stories. The component parse the US and extract its functional user, data group and functional process. Figure 15 shows how the component answer is.

5.4 Dictionary

After receive the answer from the parser component, the tool search in the dictionary for the corresponding functional process weight. After that it is stored in the data-base and shown to the user. When a US is updated its estimation history is also stored and shown to the user. Detailed estimation can be seen in Figure 16.

Figure 14 – User Story Grammar built in Xtext framework.

```
Model:
    (stories += UserStory '.')
;

UserStory:
    ('As a' | 'As an' | 'As the') user=NL
    (', I want to' | ', I can') action=NL
    ((', connected to') link=NL ((', ' slink=NL)* ('and' tlink=NL))?)?
    (('; so' | '; then') 'be informed about operation status')?
;

NL hidden(WS):
    (ID|INT|SPECIAL)+
;

SPECIAL:
    ('-' | '(' | ')')
;
```

Source: Author.

Figure 15 – Example of response in JSON format.

```
{
"0":"As a librarian, I want to save a new book, connected to author; then be notified about operation status.",
"0_user":"librarian",
"0_links":"author",
"0_message":"true",
"0_data":"author",
"0_method":"save"
}
```

Source: Author.

Figure 16 – Example of Estimation History of a US.

Source: Author.

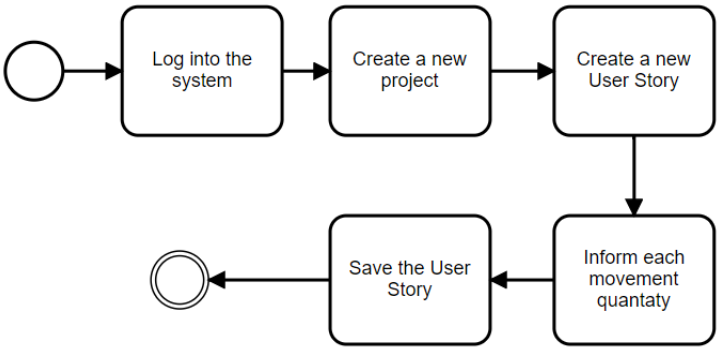
5.5 Corpora Annotator for COSMIC and User Stories

Corpora Annotator for COSMIC and User Stories (CACUS) is a tool where the user can create agile projects and store user stories and its respective size estimation in CFP units. The user should inform the number of each movement, being Entry, Exit,

Write and Read.

The Figure 17 shows the process of accessing the system and creating a project and user stories.

Figure 17 – CACUS Process.

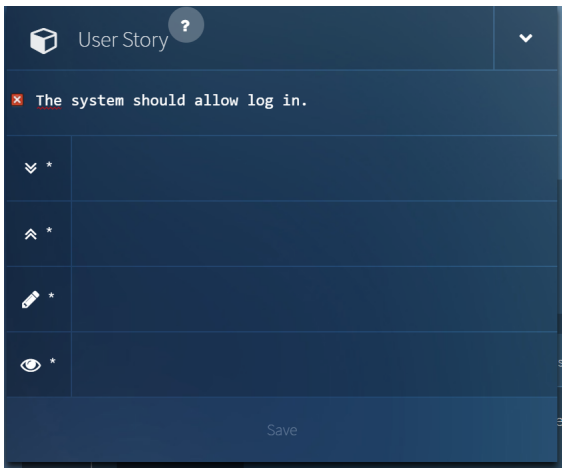


Source: Author.

The process starts when the user access and "log into the system". So, after being logged, the user is able to "create a new project" and inform its name, privacy and short description. Once the project is created, the user can "create a different number of user stories" inside the project and "inform each movement quantity" related to the US, then the user can "save" it.

Figure 18 shows an example where the text is refused, so it cannot be saved. In Figure 19 the grammar is validated as correct, so it can be saved with the respective size in CFP units.

Figure 18 – Refused User Story.



Source: Author.

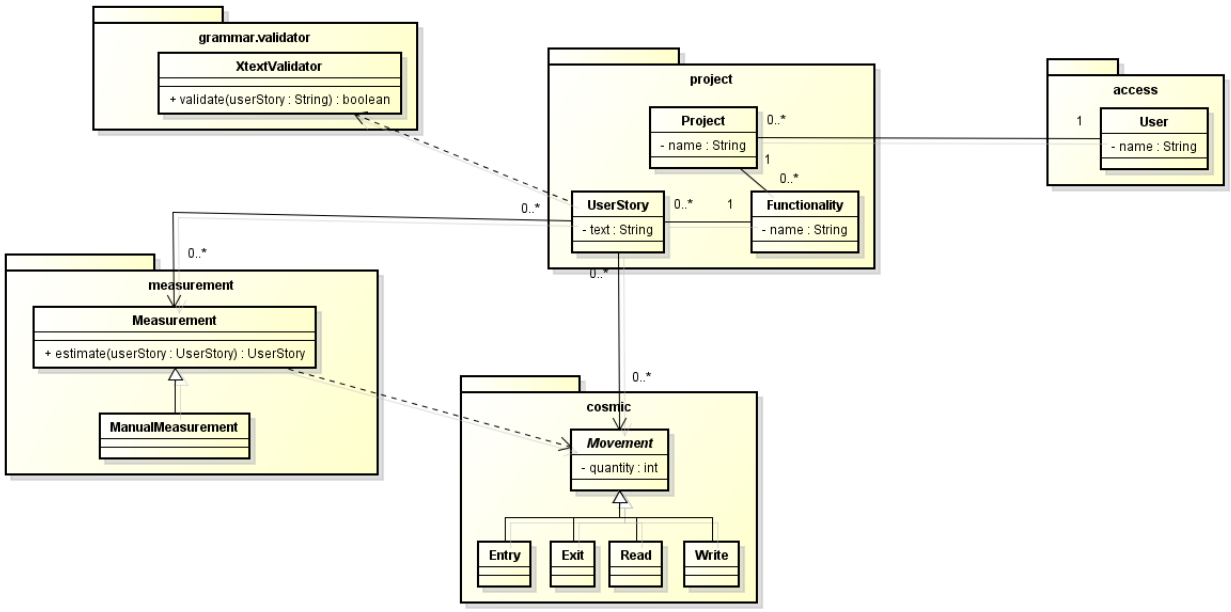
The class diagram of CACUS is shown in Figure 20. It is possible to observe the connection between packages and the main classes related to the parallel corpora. Package

Figure 19 – Correct User Story.

Source: Author.

cosmic stores classes related to the COSMIC method. Package *project* stores the *UserStory* class. Package *access* stores the class related to system user. Package *measurement* stores the classes responsible for executing the manual measurement. Finally, package *grammar.validator* is responsible for storing classes related to the Xtext framework.

Figure 20 – CACUS class diagram

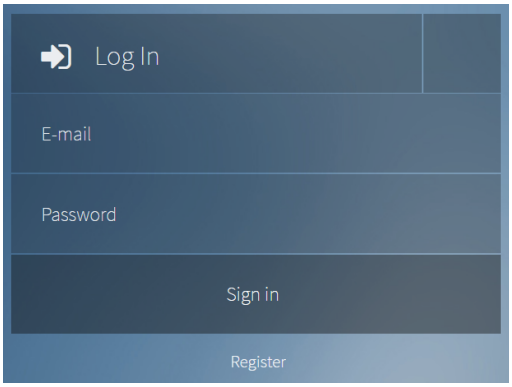


Source: Author.

5.6 Tool Demonstration

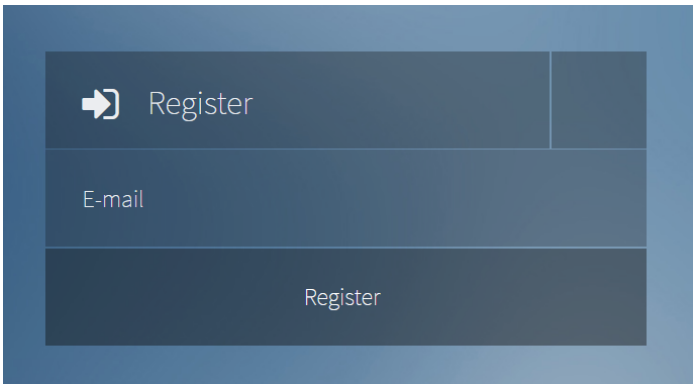
First of all you should log into the system by providing your email and password as shown in Figure 21. In case you do not have an access you should go into register page and to provide your email as shown in Figure 22, then the password will be sent to you.

Figure 21 – Example of Log In Page.



Source: Author.

Figure 22 – Example of Register Page.



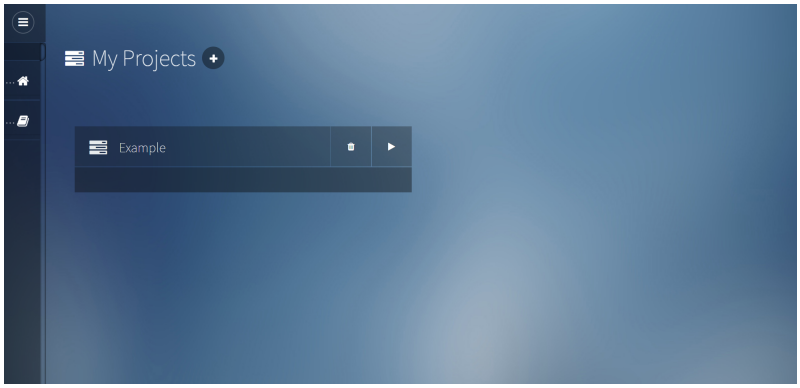
Source: Author.

After you login, you are taken to the home page (Figure 23), then you should click in the `Plus` button to create a new project, as shown in Figure 24. After save, you return to home page, and should click in the button `Play` in your created project.

So, you are taken to `Functionality` page (Figure 25). In this page you should create a new functionality, a functionality is an epic, or a group of functional processes. To create a new functionality you should click on the `Plus` button, and provide its name, for example “CRUD Books”.

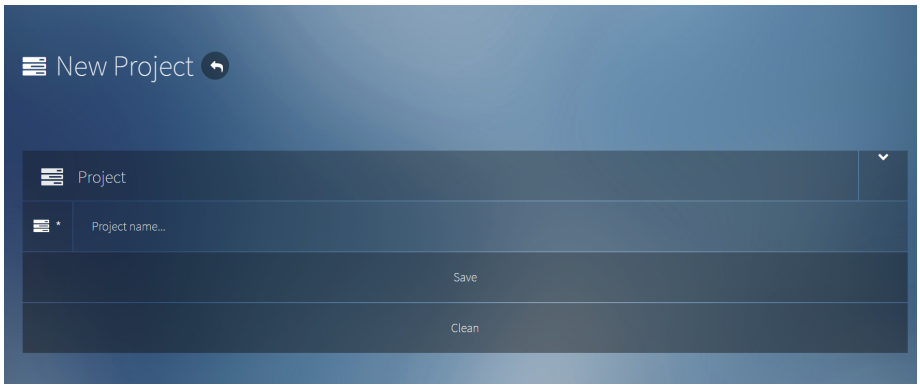
After create your first functionality, you should click on the `Play` button under your functionality name to create user stories inside it.

Figure 23 – Example of Home Page.



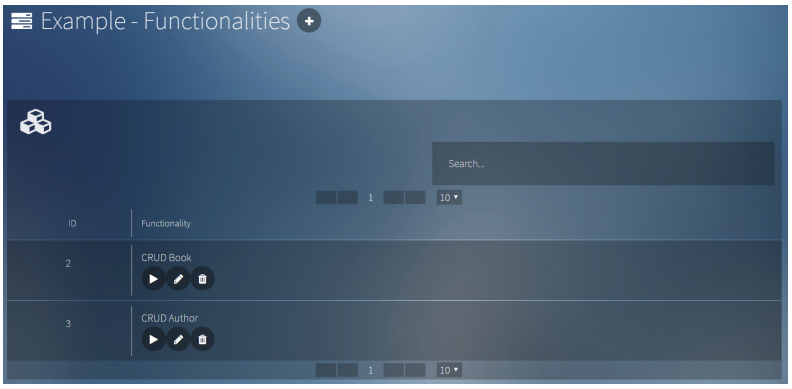
Source: Author.

Figure 24 – Example of New Project Page.



Source: Author.

Figure 25 – Example of New Functionality Page.

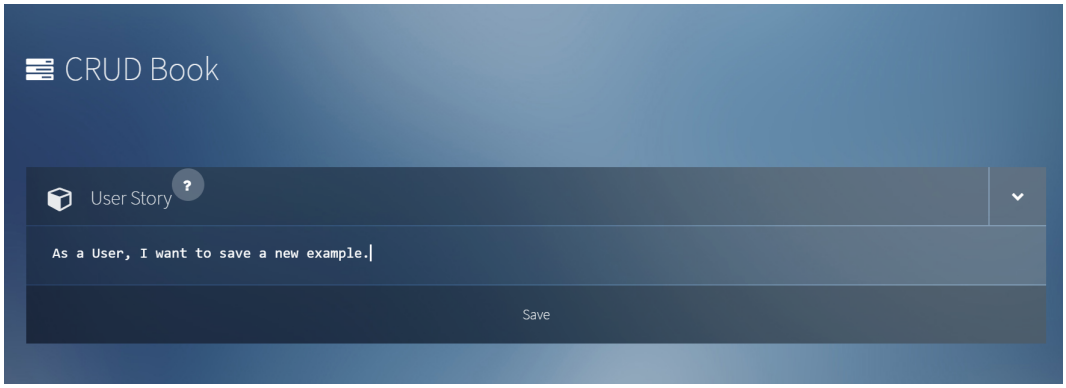


Source: Author.

Now, you are in the main page of the system. In this page you can create your user stories, Figure 26 shows an example of valid user story. The Figure 27 shows example of a text which did not match with the user story grammar.

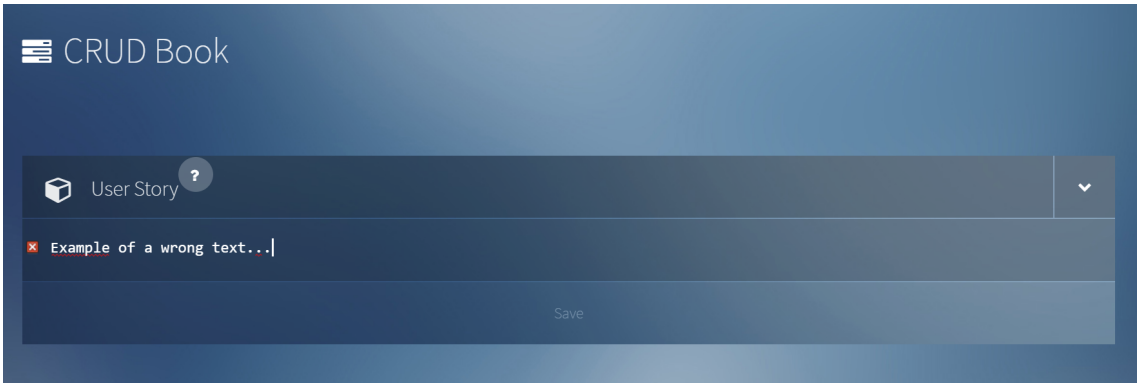
Bellow the user story input you have a table with your created user stories and

Figure 26 – Example of New User Story Page.



Source: Author.

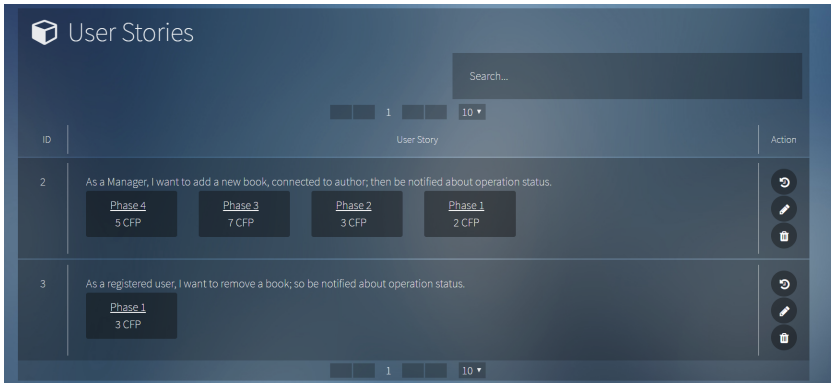
Figure 27 – Example of Wrong User Story Page.



Source: Author.

the respective estimations for each user story as shown in Figure 28. If you click on the button history, in the column on the right of the table, you can see the detailed estimation and the changing history of your user story, as shown in Figure 29.

Figure 28 – Example of User Story Table Page.



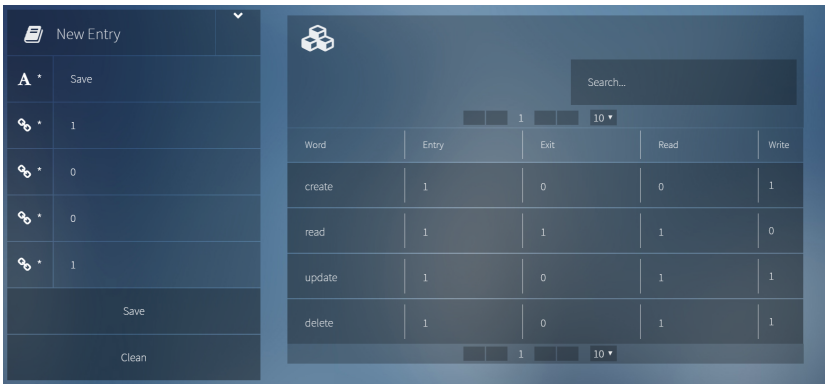
Source: Author.

Figure 29 – Example of User Story History Page.

Source: Author.

The Dictionary page can be accessed by clicking in the button on the left side menu. In the page shown in Figure 30 you are able to inform a new functional process name and its respective movements quantity.

Figure 30 – Example of Dictionary Page.



Source: Author.

5.7 AutoCosmic Evaluation

We performed the automated tool evaluation, using a sample of 40 user stories divided in 4 scenarios. Each User Story was estimated by two professionals, an expert certified one and a junior measurer, and also by the automated tool.

The Table 10 shows the US from the first scenario and its respective estimation sizes in CFP units. The first scenario is a system for Library management.

Based on the first scenario on Table 10 we can observe only one estimation among expert, junior and tool estimation is different. The US12 was estimated as one less CFP. This is cause by a miss interpretation from the automated tool.

The second scenario is a sensor monitoring system. The Table 11 presents the US and the respective estimates in CFP units.

Table 10 – First scenario: Library.

ID	US	Expert	Junior	Tool
US10	As a Librarian, I want to register a new book.	3	2	2
US11	As a Manager, I want to list all books.	3	3	3
US12	As a Librarian, I want to register a new book, connected to author and publisher; then be informed about operation status.	7	7	6
US13	As a Manager, I want to disable a book; then be informed about operation status.	5	5	5
US14	As a Manager, I want to list all booked books, connected to clients.	5	3	5
US15	As a Client, I can book a book; then be informed about operation status.	-	-	-
US16	As a client, I can rent books.	-	-	-
US17	As a Manager, I want to list clients with pending loan, connected to book.	5	5	5
US18	As a Manager, I want to analyze all loans.	-	-	-
US19	As a Librarian, I want to send emails about new books arrival.	2	2	2

Source: Author.

Table 11 – Second scenario: Monitoring.

ID	US	Expert	Junior	Tool
US20	The sensor, can register when the tank fluid level is under 1 meter.	2	-	2
US21	The sensor, should inform the opening the tap when the tank fluid level is under 50 cm.	2	2	2*
US22	The sensor, can monitor the fluid.	-	-	-
US23	The sensor, can register when the tank is dry.	2	2	2
US24	The sensor, can trigger the alarm if the fluid color changes.	2	3	2
US25	The sensor, can trigger the alarm when the tank fluid level is under 10 cm.	2	-	2
US26	The sensor, can verify the fluid color.	-	-	-
US27	The sensor, should inform the closing the tap if the fluid level is above 5 meters.	2	2	2*
US28	The sensor, can monitor the fluid temperature.	-	-	-
US29	The sensor, can inform the draining out the tank when the temperature is above 30 °C.	2	2	2*

Source: Author.

Based on the second scenario on Table 11 we can observe that the tool estimation is corresponding to the Expert Professional estimation. It is observable that the lines US21, US27 and US29 has an asterisk. This is caused by a change made in the user story text. The first tool estimation of these samples accused ambiguity, so contacting the professional it was accorded to change the text so, it was clear and the tool could estimate it.

The Table 12 shows the third scenario, that is a web service.

Table 12 – Third scenario: Web Service.

ID	US	Expert	Junior	Tool
US30	The web service, can remain available 24 / 7.	-	-	-
US31	The web service, should reply code 110 if there is an error.	2	2	2
US32	The web service, can receive values then to return the mean.	2	-	?
US33	The web service, can receive a word then to return its synonyms.	3	3	?
US34	The web service, can verify if there is connection.	-	-	-
US35	The web service, can receive a number then return if it is even or odd.	2	2	?
US36	The web service, can register the time of all query.	2	2	2
US37	The web service, can calculate the median.	-	-	-
US38	The web service, can receive values then return them in crescent order.	2	2	?
US39	The web service, can accept xml or json.	-	-	-

Source: Author.

Based on the third scenario on Table 12, we can observe that the tool estimated US32, US33, US34, and US38 as immeasurable. This failure is cause by a miss interpretation from the automated tool. This failure must be corrected in subsequent versions.

The Table 13 shows the fourth scenario that is a system to create questionnaires.

Based on the fourth scenario on Table 13, we can also observe that the automated tool result also correspond to to expert professional estimation. There is only one result that diverges that in US48. This bad result was also caused by a miss interpretation from the tool.

5.8 Chapter Lessons

In this chapter, we showed the proposed tool for automated user story size estimation using COSMIC method, based on Natural Language Processing.

Our proposed tool accepts a user story according to a well defined grammar, detects the important entities in terms of COSMIC inside it (Functional User, Data

Table 13 – Fourth scenario: Questionnaire maker.

ID	US	Expert	Junior	Tool
US40	As a User, I want to create a questionnaire.	2	2	2
US41	As a User, I want to send a questionnaire by email.	2	3	2
US42	The system, can notify the questionnaire owner when it is answered.	-	3	-
US43	As a User, I want to map all answers.	-	-	-
US44	The system, can update to blocked the questionnaire when it reaches the deadline.	3	3	3*
US45	As a User, I want to verify the questionnaire answer frequency.	-	-	-
US46	As a User, I want to copy all questionnaire content.	3	4	3
US47	The system, can show the numeric answers in graphic format.	2	2	2
US48	As a User, I want to receive notification via email when a questionnaire is blocked.	2	3	7
US49	As a User, I want to correct the answers.	-	-	-

Source: Author.

Groups, Functional Process) and estimate the size in CFP units, based on a mapping between functional processes and movements.

Moreover, the tool was validated comparing its estimation results with results from a certified professional with more than 15 years of experience in COSMIC sizing. The resultant is that AutoCosmic has a performance at least as accurate as an expert professional and it is faster and misinterpretation proof. Furthermore, we plan to conduct an experimental study to validate it empirically.

6 FINAL REMARKS

The main objective of this work is to create an automated tool called AutoCosmic for COSMIC estimation. It intends to minimize the measurement cost and helps software size estimation in agile development that uses US.

We first proposed a new template for user story writing. This new template extends the original one and add some expressiveness in terms of COSMIC sizing. This template was validated based on a survey with COSMIC certified professionals. In this survey we could observe an high tendency to consider that the new template is valuable not only in terms of COSMIC sizing, but to obtain a better understanding of the requirement.

This new template is the basis for the web tool, in which it is possible to store user stories and obtain is estimation based on natural language processing techniques. Moreover, this web tool has also a feature for parallel corpora building, that stores user stories and its COSMIC size estimation.

Both approaches were validated along COSMIC certified professionals. The proposed template were well received by them and classified as a great step forward in terms of user story writing for COSMIC sizing. The web tool for automated COSMIC estimation, has a good precision when contrasted with professional manual result, but it still need and more effective empirical evaluation.

Based on this, we can consider that the objectives of this work were well achieved. We developed a functional web based tool, where professionals can obtain the estimated size of user stories and based on this manage software projects. Additionally, a new user story writing template was released, contributing to better expressive user stories. Furthermore, this web tool has also a parallel corpora building feature, that open the possibility to create a corpora to be source for deep learning technique which can be used to improve the estimation accuracy and add more intelligence to the web tool itself.

BIBLIOGRAPHY

ABRAN, A. et al. **Guide to the software engineering body of knowledge: 2004 version SWEBOK**. [S.l.]: IEEE Computer Society, 2004. Cited in page 25.

AKCA, A.; TARHAN, A. Run-time measurement of cosmic functional size for java business applications: Initial results. In: **Software Measurement and the 2012 Seventh International Conference on Software Process and Product Measurement (IWSM-MENSURA), 2012 Joint Conference of the 22nd International Workshop on**. [S.l.: s.n.], 2012. p. 226–231. Cited 3 times in pages 19, 30, and 31.

AYSOLMAZ, B.; DEMIRORS, O. Automated functional size estimation using business process models with uprom method. In: **Software Measurement and the International Conference on Software Process and Product Measurement (IWSM-MENSURA), 2014 Joint Conference of the International Workshop on**. [S.l.: s.n.], 2014. p. 114–124. Cited 2 times in pages 30 and 31.

BEHRENS, H. et al. Xtext user guide. **Dostupné z WWW: http://www. eclipse. org/Xtext/documentation/1_0_1/xtext. html**, p. 7, 2008. Cited in page 50.

BUNDY, A.; WALLEN, L. Context-free grammar. In: **Catalogue of Artificial Intelligence Tools**. [S.l.]: Springer, 1984. p. 22–23. Cited in page 26.

CHOMA, J.; ZAINA, L. A.; BERALDO, D. Userx story: incorporating ux aspects into user stories elaboration. In: SPRINGER. **International Conference on Human-Computer Interaction**. [S.l.], 2016. p. 131–140. Cited 2 times in pages 36 and 37.

CHOWDHURY, G. G. Natural language processing. **Annual review of information science and technology**, Wiley Online Library, v. 37, n. 1, p. 51–89, 2003. Cited in page 26.

COHN, M. **Advantages of the "As a user, I want" user story template**. 2008. <https://www.mountaingoatsoftware.com/blog/advantages-of-the-as-a-user-i-want-user-story-template>. Accessed: 2017-09-09. Cited in page 37.

COSMIC. **Introduction to the COSMIC Method of Measuring Software**. [S.l.], 2014. Cited 4 times in pages 19, 23, 24, and 25.

COSMIC AGILE. **Guideline for the user of COSMIC FSM to manage Agile projects**. [S.l.], 2011. Cited 5 times in pages 19, 20, 24, 25, and 37.

COSMIC MANUAL. **COSMIC Measurement Manual**. [S.l.], 2015. Cited in page 19.

DIAB, H. et al. μ c rose: automated measurement of cosmic-ffp for rational rose realtime. **Information and Software Technology**, Elsevier, v. 47, n. 3, p. 151–166, 2005. Cited in page 20.

GAROUSI, V.; FELDERER, M.; MÄNTYLÄ, M. V. Guidelines for including the grey literature and conducting multivocal literature reviews in software engineering. **arXiv preprint arXiv:1707.02553**, 2017. Cited 4 times in pages 27, 28, 36, and 38.

HUSSAIN, I.; KOSSEIM, L.; ORMANDJIEVA, O. Approximation of cosmic functional size to support early effort estimation in agile. v. 85, p. 2–14, 2013. ISSN 0169-023X. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0169023X1200064X>. Cited in page 30.

JENNER, M. S. Cosmic-ffp and uml: Estimation of the size of a system specified in uml–problems of granularity. In: **The Fourth European Conference on Software Measurement and ICT Control**. [S.l.: s.n.], 2001. Cited in page 31.

KAMBHATLA, N. Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. **Proceedings of the ACL 2004 on Interactive poster and demonstration sessions**. [S.l.], 2004. p. 22. Cited in page 26.

LAVAZZA, L.; BIANCO, V. del; LIU, G. Analytical convertibility of functional size measures: A tool-based approach. In: **Software Measurement and the 2012 Seventh International Conference on Software Process and Product Measurement (IWSM-MENSURA), 2012 Joint Conference of the 22nd International Workshop on**. [S.l.: s.n.], 2012. p. 160–169. Cited 2 times in pages 30 and 32.

LIKERT, R. A technique for the measurement of attitudes. **Archives of psychology**, 1932. Cited in page 40.

LIU, B. Sentiment analysis and opinion mining. **Synthesis lectures on human language technologies**, Morgan & Claypool Publishers, v. 5, n. 1, p. 1–167, 2012. Cited in page 42.

MARíN, B.; QUINTEROS, J. A cosmic measurement procedure for bpmn diagrams. In: . [S.l.: s.n.], 2014. Cited 2 times in pages 29 and 30.

MATTS, C. **Feature Injection: three steps to success**. 2011. <https://www.infoq.com/articles/feature-injection-success>. Accessed: 2017-09-09. Cited in page 37.

MORENO, A. M.; YAGÜE, A. Agile user stories enriched with usability. In: SPRINGER. **International Conference on Agile Software Development**. [S.l.], 2012. p. 168–176. Cited 2 times in pages 36 and 37.

PANG, B.; LEE, L. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. **Proceedings of the 42nd annual meeting on Association for Computational Linguistics**. [S.l.], 2004. p. 271. Cited in page 42.

PETERSEN, K. et al. Systematic mapping studies in software engineering. In: SN. **12th International Conference on Evaluation and Assessment in Software Engineering**. [S.l.], 2008. v. 17, n. 1. Cited in page 27.

PUPEK, D. **Writing User Stories the 5 Ws Way**. 2008. <http://blog.agilejedi.com/2008/03/writing-user-stories-5-ws-way-writing.html>. Accessed: 2017-09-09. Cited in page 37.

RODRIGUES, E.; LOPES, H. S. Inferência de gramáticas livres de contexto usando programação genética. **I Simpósio Brasileiro de Inteligência Computacional**, 2007. Cited in page 26.

SAG, M.; TARHAN, A. Measuring cosmic software size from functional execution traces of java business applications. In: **Software Measurement and the International Conference on Software Process and Product Measurement (IWSM-MENSURA), 2014 Joint Conference of the International Workshop on**. [S.l.: s.n.], 2014. p. 272–281. Cited 2 times in pages 30 and 31.

STAIANO, J.; GUERINI, M. Depechemood: A lexicon for emotion analysis from crowd-annotated news. **arXiv preprint arXiv:1405.1605**, 2014. Cited in page 42.

WILSON, T.; WIEBE, J.; HOFFMANN, P. Recognizing contextual polarity in phrase-level sentiment analysis. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. **Proceedings of the conference on human language technology and empirical methods in natural language processing**. [S.l.], 2005. p. 347–354. Cited in page 42.

YEH YI-HONG CHEN, C.-Y. Y. D. Applying reverse engineering techniques to verify the estimation of software code size using cosmic full function point. In: **6th International Conference on Applied Informatics and Computing Theory**. [S.l.: s.n.], 2015. Cited 2 times in pages 30 and 31.

ZEAARAOUI, A. et al. User stories template for object-oriented applications. In: IEEE. **Innovative Computing Technology (INTECH), 2013 Third International Conference on**. [S.l.], 2013. p. 407–410. Cited 2 times in pages 36 and 37.

Appendix

APPENDIX A – COSMIC USER STORY STANDARD SURVEY

Cosmic User Story Survey

This survey is part of a research which intends to improve Requirements Engineering in Agile Environments which use COSMIC FSM as measurement Method.
Please, answer these questions with your opinion and give suggestions about this theme.

Thank you in advance. =)

There are several templates for writing an User Story (US) some of them are presented bellow.

1 - "As a <role>, I want <goal/desire> so that <benefit>" (Connextra)

2 - As a <role>, I want <goal/desire>" (Mike Cohn)

3 - "As <who> <when> <where>, I <what> because <why>." (Five Ws)

According to "Guideline for the use of COSMIC FSM to manage Agile projects", If a Message Sequence Diagram can be drawn for a functional process, which is the minimum needed to understand the user requirement, then measuring the COSMIC functional size is a trivial task.

But in other point of view, these US templates may not provide all information needed to express the whole functional process in terms of movements identification. For example. connections or links with other Data Groups, such as, functional processes which demands selecting a value from a dropdown (eg. Selecting a City in a User Register).

Considering this, a new US template is presented providing greater expressiveness in terms of COSMIC Sizing. Remenbering that the objective is not express the whole requirement life cycle but, be more expressive in the first step of requirement gathering.

This standard has being called COSMIC User Story Standard(CUSS).

As a <who/role>, I want to <what>, linked to <connections>; so be notified about operation status.

Where:
<who/role> is the Functional User.
<what> is the verb which represent the action or functional process.
<connections> represents other datagroups involved in this functional process.
"so be notified about operation status" is optional and represents the feedback to functional user.

Ex.:
1 - As a Manager, I want to persist a new book.

2 - As a Registered User, I can add a new book; so be notified about operation status.

3 - As the Manager, I want to add a new book, linked to author.

4 - As a Librarian, I want to save a new book, connected to author and publishing company; then be notified about operation status.

Considering this explanation, please, answer the questions bellow.

*Required

Profile

1. **Years of experience with COSMIC** *

2. **Are you COSMIC certified?** *
   Mark only one oval.

   - ◯ Yes
   - ◯ No

Open Questions

In this section your are open to give your opinion and feedback, so discourse a little about your answers.

3. **Do you think that there is a lack of information in current User Story Templates?** *

4. **In your opinion, Cosmic User Story Standard helps to identify more movements than other templates? Please discourse about your answer.** *

5. **Do you think that this addition of information compromises the agile aspect? Please discourse about your answer.** *

Closed Questions

Scale according to your agreement. being 0 (zero) disagree strongly to 4 (agree strongly)

6. **Current Templates provide enough information in term of COSMIC Sizing.** *
   Mark only one oval.

	0	1	2	3	4	
Disagree strongly	◯	◯	◯	◯	◯	Agree strongly

7. **COSMIC User Story Standard provides greater expressiveness in term of COSMIC Sizing.** *
Mark only one oval.

	0	1	2	3	4	
Disagree strongly	◯	◯	◯	◯	◯	Agree strongly

8. **CUSS Template compromises the agility aspect in a process.** *
Mark only one oval.

	0	1	2	3	4	
Disagree strongly	◯	◯	◯	◯	◯	Agree strongly

FeedBack

Please, give me your opinion about this research.

9. **Do you have any question or suggestions to improve this research?**

Thank you Again =D

INDEX