A SOFTWARE TOOL FOR ASSESSING MULTI-FRAMEWORKS ENVIRONMENTS

Frederico Samuel Felisberto
Frederico.Felisberto@tecnico.ulisboa.pt

Instituto Superior Técnico, Lisboa, Portugal

October 2017

Abstract

Organizations implement and use frameworks to improve organizations competitiveness. Nowadays, approximately 315 standards, guides, handbooks, and other prescriptive documents which were taken as frameworks have been developed over time, adding in that way more instability for process quality, norms and regulations. Taking into account that there is no single reference model that meets all requirements that organizations need to satisfy, organizations currently demand the implementation of practices of several frameworks as support to their needs. However, the assessment of different frameworks implies an unreasonable effort because each framework defines its own scope, structure of process entities, definitions, terminology quality systems and approach. Therefore, in this thesis abstract we present the developed software assessment tool that uses an Enterprise Architecture platform called EAMS to reduce the needed effort to perform assessments in multiframeworks environments.
Keywords: COBIT, Enterprise Architecture,ISO 22001, ISO 33052, Multiframework Environment Assessments, Frameworks, Software tool.

1. Introduction

Taking into account that there is no single framework that meets all requirements which organizations need to satisfy, organizations currently demand the implementation of practices of more than one framework as support to their needs [1]. This situation allows organizations to select and complement their processes from the frameworks which better fit their contexts, e.g. if a framework like the NIST Handbook [2], is not suitable, then an organization can decide to implement ISO 27002. To all intents and purposes, the implementation of multimodel processes from shared goals reduces time and excessive expenditures associated to the adoption of multiple frameworks.

Thus, the integration of several frameworks, e.i, ISO 9001 and CMM [3], CMMI and ITIL [4], CMMI and ISO 12207 and so forth, has became the solution adopted by many organizations to address issues like costs, required human effort, time, human failures and overlapping amongst frameworks.

Bearing in mind all these aspects above, we propose a solution that intends to solve these problems. The proposed solution is divided in two phases: (i) mapping and integration of the different frameworks in a Enterprise Architecture tool called EAMS with collaboration of [Renato Lourinho] and (ii) development of a software tool that will allows multiframeworks assessments.

1.1. Objectives

To achieve the expected results it is important to identify, analyze and define the general objective. Once defined, is refined in smaller objectives with a more constricted but more precise scope. Nevertheless, we have defined the following general objective:

> To develop a WEB tool to assess different frameworks simultaneously. In that way organizations can reduce the resources needed (time, effort, people) to assess multiframeworks environments.

There are five partial objectives which must be fulfilled and are listed below:

(i) Carry out an in-depth study in the structure and composition of each framework as well the existing software's.

(ii) Define unique terminologies and structure, which outline processes, base practices and work products.

(iii) Mapping models such as COBIT 5 and ISO 33052 into EAMS [1], eliminating overlap

[1] EAMS is an innovative solution to keep architectural

through the determining the existing similarities [Renato Lourinho Master Thesis].

(iv) Validate all frameworks and metamodel within EAMS [Renato Lourinho Master Thesis].

(v) Choose the technologies for software developing and in-depth study the EAMS REST Services (GET, POST, PUT, DELETE).

1.2. Thesis Abstract Outline

The aim of this subsection is to describe how this document is structured. In Section 2 the authors present the research methodology used in this thesis abstract. Section 3 describes the research problems. In Section 4 the authors introduce the main approaches and similar software tools that served as inspiration for this research.
Section 5 describes this abstract proposal. In section 6 the authors present the demonstration of the software tool, and the evaluation is presented in Section 7. We finish this thesis abstract in Section 9 with the conclusions, contributions, limitations and future work.

2. Research Methodology

The Design Science Research Methodology (DSRM) was the research methodology adopted in this research. The DSRM is appropriate for researches that seeks to extend the boundaries of human and organizational capabilities by creating new and innovative artifacts [6].

This methodology can also be applied to information technology in order to solve organizational problems. DSRM differentiates from other research paradigms because it attempts to develop and reach artifacts that can be proven effective in real world scenarios [7]. Peffers et al. [7] propose a process model consisting of six iterative activities in a nominal sequence. Table 1 describes each of these activities and how each activity is addressed to this research.

DSRM	Thesis abstract Approach
Problem identification and motivation	Existing tools do not support simultaneously assessments using different frameworks.
Definition of the objectives of a solution	Develop a WEB tool to assess different reference models simultaneously. In that way organizations can reduce the resources needed (time, effort, people) to assess multi references models environments.
Design and development	A WEB Software Tool that uses an Enterprise Architecture tool called EAMS to assess multi reference models.
Demonstration	The demonstration was performed using the following models COBIT 5, ISO 33052 and ISO 270001 in an organization with Enterprise Architecture implemented already
Evaluation	Usability Tests , Interviews and Laboratory experiment [8].
Communication	Dissertation Papers, submission to the 30th International Conference on Advanced Information Systems Engineering.

Table 1: DSRM activities' descriptions. Adapted from [6, 7, 8]

3. Problem Statement

Approximately 315 standards, guides, handbooks, and other prescriptive documents which were taken as frameworks have been developed over time [9]. Taking into account that there is no single framework that meets all the requirements that organizations need to satisfy, organizations currently demand the implementation of practices of several frameworks as support to their needs.

The key reason why organizations use these frameworks is to ensure that they meet performance and compliance requirements. It means that organizations should to continually improve Information Technology(IT) performance with regard to its economic efficiency.

In order to undertake continual improvement activities many organizations perform process assessments. These assessments involve the systematic measurement, analysis and reporting of the performance of core processes. The results are then used to evaluate the capabilities of these processes and drive process improvement activities.

In that way, continual improvement requires the company's positioning with regard to its IT capabilities and the quality of its goods and services. As a rule, this positioning involves a comparison with the company's goals, external requirements (e.g. customer demands, laws or guidelines), or benchmarks.

For each aspect of the company's IT under investigation, the questions arise what needs to be measured, how, and what to compare it with, in order to assess the current situation of a company and to assign it a specific quality or degree of maturity [10].

Moreover, organizations struggle with the per-

representations up-to-date with minimum effort. EAMS is able to generate on-the-fly organization wide architectural blueprints (models, maps or diagrams) based on information gathered from catalogs and other information sources within the organization.This allows managers, architects and technical leaders to explore and navigate over a full graphical view of the organizations business, systems and infrastructure [5].

ceived complexity and difficulty of adopting and assessing several frameworks at the same time [9], because each practice defines its own scope, definitions, terminologies and structure. Therefore, one can argue that several of these Enterprise Governance of IT Frameworks overlap each other as one which can be seen in Figure 1.

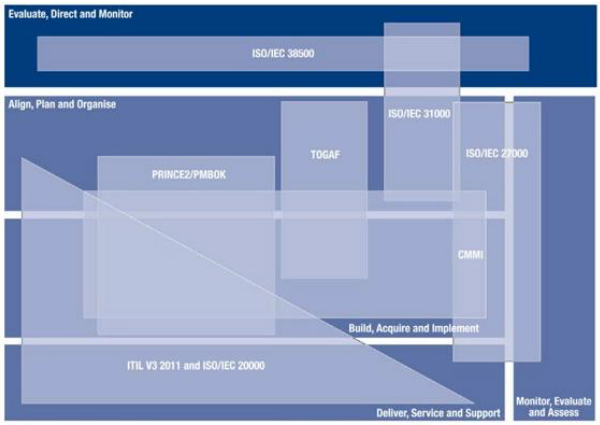


Figure 1: Overlapping amongst different frameworks.

At a time when organizations strive to be efficient and effective, it seems counter intuitive to be wasting resources by having different organizational departments handling both approaches independently [11].

In that way, the thesis where this abstract has been created intends to facilitate the assessment of different frameworks in order to avoid and help stakeholders to reduce the effort needed to assess several frameworks (COBIT, ITIL, ISO 27000 and so fourth) simultaneously. The authors intend to achieve this goal by developing a Web based software tool with EAMS as foundation in order to be integrated with Enterprise Archicteture. So, with both integrated, is possible to transform a legacy of fragmented applications, organizational structures and processes (both manual and automated) into an integrated environment with optimised processes that are responsive to change and the delivery of the business strategy [12]. The proposal, is a WEB based software tool, developed with REST API 2 technology, composed by REST services and developed under EAMS WEB tool by remote invocations of services. Furthermore, the tool was developed to support multiple assessments simultaneously as long as the different structures are mapped into EAMS allowing the stakeholders/assessors to perform desired assessments using different frameworks at same time.

4. Related Work

The aim of this section is to describe and compare similar approaches that served as inspiration for the solution. In that way, we intend to prove that the existing software tools do not solve this thesis problem. Therefore, amongst all assessment tools reported, we will describe three of them based on following order: (i) Appraisal Assistant, (ii) Appraisal Wizard and (iii) SPiCE-Lite. Table 2 shows the existing software tools.

4.1. Appraisal Assistant

Appraisal Assistant is an stand-alone software which supports assessment of organizational maturity and process capability. It closely follows approaches consistent with the requirements of ISO/IEC 15504 - Information technology: Process assessment and the Assessment Requirements for CMMI.

4.1.1 Tool description and Features

Unlike other existing tools, Appraisal Assistant takes an explicitly evidence-driven approach to recording the information generated in an assessment [13]. The evidences are arranged by instances and collected using spreadsheets. The evidences can be imported from older projects using evidence registry.

Appraisal Assistant provide innumerable features and some of them are listed below:

- Support for multiple process models which includes ISO/IEC 15504-5, ISO/IEC 15504-6, SPICE, and CMMI staged/continuous representation [13].
- Support for user defined appraisal models [13].
- Support for multiple methods including SCAMPISM A (v.1.2) and B Class, and generic ISO/IEC 15504 conformant assessment methods.
- Support for user defined assessment methods [13].
- Support for conversion of results from one framework to another; i.e. expression of CMMI Appraisal results as ISO 15504 standard process profiles,and definition of coverage of ISO 9001 requirements by CMMI or ISO 15504 assessments / appraisals [13].
- Facility to split and consolidate evidence capture activities within the team through the establishment of mini-teams [13].
- Automatic generation of a variety of reports, including Appraisal Disclosure Statement, PIID, Assessment Record, Appraisal / Assessment Findings, Strength / Weakness summaries, Rating Profiles, and workload summaries [13].

- Automatic reporting of model coverage by collected evidence, to meet the requirements of the SCAMPISM A v.1.2 method [13].

4.2. Appraisal Wizard

Appraisal Wizard is a comprehensive solution to assist management and execution of planning, preparation, data collection and consolidation, reporting of appraisals, audits, compliance and process checks [14].

Its functionality helps businesses organize and monitor compliance against frameworks and governance activities using any internal or external model, method. Appraisal Wizard facilitates improved audit readiness, more effective planning, and efficient preparation, on-site, and post-appraisal activities [14, 15].

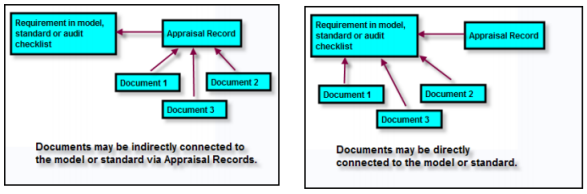
4.2.1 Description and Features

Appraisal Wizard is characterized by variety of functionalities when compared to other existing tools [16]. Likewise Appraisal Assistant, the Appraisal Wizard takes an explicitly evidence-driven approach and below is demonstrated how they can be configured:

Documents Recorded in the database and then connected to model elements or appraisal records and act as supporting evidence of compliance when connected [15].

Appraisal records Recorded in the database and usually connected to model elements to provide information about model compliance, noncompliance, expected evidence, interview questions to be asked, corrective actions needed and any other type of information the user would care to associate or record [15].

Figure 2 shows documents logical of how they can be connected to frameworks or appropriate part of it. There are many available methods and operations that can be used to created or updated documents [15].



(a) Connected to the model element via an appraisal record

(b) Directly connected to the model element

Figure 2: Evidences collection structure.

With Appraisal Wizard, audits and practitioners can perform following actions [15]:

- Conduct high quality appraisals using a variety of methods.
- Work with industry standard models such as CMMI, ISO 9001, ITIL, P-CMM, eSCM, ISO 15288, DMM, and ISDs ISF.
- Reuse appraisal assets, evidence, or appraisal results.
- Use the historical database to compare results across business units or monitor progress over time.

4.3. SPiCE-Lite

SPiCE-Lite is the self assessment tool for software development organizations, projects management and to evaluate and analyze the software development processes, quickly and efficiently. This tool allows to perform an assessment and to determine strengths and weaknesses concerning the way of developing software [17]. The correspondent process model can be seen in Figure 3.

Furthmore, SPiCE-Lite is useful to rate the most important process categories for software development within a group of organization (Self-Assessment) or guided using the help of a consultant (Guided Self-Assessment) [17].

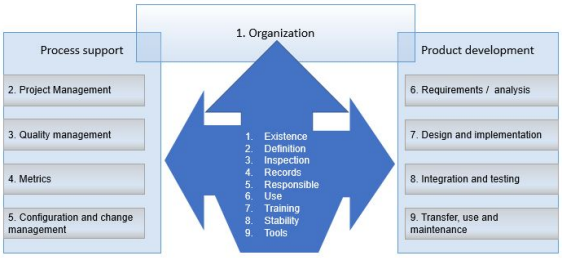


Figure 3: The SPiCE-Lite Process Model: [17].

4.3.1 Features

SPiCE-Lite poses various features and three solutions (*Personal*, *Company* and *Corporate*) and they can be enabled according to the agreements between the client and owner.

The *Company* and *Corporate* versions includes an integrated software called *SynEval* which allows the assessors to calculate ratings for two or more evaluation based on frequency distribution [17]. The *Corporate solution* includes a software tool called *SynEdit* which is used to customize the *SPiCE-Lite* assessment tool for individual needs [17].

4.4. Software Process Assessment Tools Comparison

There are several software process assessment tools reported in current research, and Gazel et al. provide a list of these tools [18]. For this research, we

described three of them, providing existing similarities as well as differences.

Reviewing the latest versions of each tool, we compared their attributes in Table 2 based on: *assessment method supported*, *type of the application (stand-alone or web-based)*, and *support for self-assessment*.

Name	**Assessment method supported**	**Stand-alone or Web-based**	**Support for self-assessm.**
Appraisal Assistant [13]	SCAMPI, SPICE	Stand-alone	No
Appraisal Wizard [14]	SCAMPI, SCE, CPA IPI	Stand-alone	Yes
SPiCE-Lite Tool [17]	ISO/IEC 15504	Stand-alone	Yes
COBIT 5 Self-Assessment Tool [19]	COBIT - PAM	Web-based	Yes
HProcess [20]	Models, Methodologies and Standards	Web-based	Yes
CMMiPal v1.0 [21]	Closer to SCAMPI	Stand-alone	No
CMM-Quest v1.3 [22]	ISO/IEC 15504	Stand-alone	Yes
SPICE 1-2-1 [23]	ISO/IEC 15504	Stand-alone	Yes
Model Wizard [24]	Not applicable	Stand-alone	No

Table 2: List of Software Process Assessment Tools based on different reference models, methodologies and standards. Adapted from [25].

4.5. Critical Analysis

Three Software Process Assessment Tools were described in this section. So, the reported tools in this section, were developed to evaluate and analyze software development processes, which means that, there are able to measure the maturity level achieved by assessed processes withing those organizations.

Considering the *Appraisal Assistant* and *Appraisal Wizard*. Both tools are quite similar (same goals, interfaces and features) and although they were designed for different contexts they share same the purposes. The only difference between both is concerning to the standards and supported methods by each of them.

So, it should also be said that, organizations are not looking for this kind of software, i.e, software's which support singular framework but, they need an unique software which is capable to support any framework.

The software integrity, overlapping amongst frameworks and others issues are extended to the rest of the tools reported in this research, except to the *Hprocess Tool* which define an ontology amongst frameworks without identify existing relations, similarities or differences amongst processes, bases practices and work products. This tool brings a new insight into this industry by defining a unique ontology for these frameworks and terminologies. Nevertheless, all of these tools poses static structure and they have been developed and addressed specifically to certain frameworks which make all of them cumbersome for integrated environment.

Our solution has advantages over all others tools reported in this section in two important aspects. (I) Assess several frameworks simultaneously and (II) by having Enterprise Architecture as foundations [12].

5. Research Proposal

This section describe the proposed solution stated and detailed in Section 3. However, was used the Unified Modeling Languages - *UML* notation to describe all required artifacts in both phases (*design* and *implementation*). Furthermore, is included and demonstrated, the proposed low fidelity prototypes (sketches) and high fidelity prototypes (interfaces).

5.1. Design and Implementation

This section presents in detail the *Design* and *Implementation* phases proposed in the DSRM. The business processes description is done in the analysis phase which usually includes models and diagrams as stated in the *Rational Unified Process* development methodology [26]. The proposed solution comprises the following Use Cases: *Create Assessment, Delete Assessment, Update Assessment, Validate Assessment, Create Account, Delete Account, Check Account, Update Account, Recover Password, Login* and *Logout*. Due to the number of existing Use Cases and the complexity we will not describe all of them, i.e, only the main one which corresponds to *Create Assessment*. Thus, the business description is made by UML diagrams and will comprise the following diagrams or models *Use Case, Domain, Activity, Sequence, Component, Class* and *Deployment* but for this thesis abstract we will detail only *Use Case* and *Class*.

5.2. Use Case Model

The Use Case models are used to help identify user roles, and usually are focused on user goals. Their representation is based on in graphs, stereotypes which present actors of system, workers, Use Cases

and relations. The Use Case model corresponds to an external view of a system. Additionally, Use Case models can be seen as high-level representation, i.e, their can be seen as a high-level of abstraction about the interactions amongst external elements and system features. The graphical representation for our solution can be seen in Figure 4.

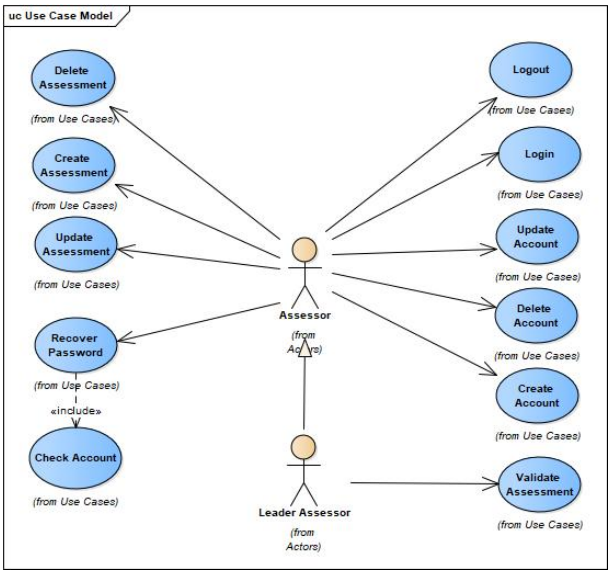


Figure 4: Use Case Model

5.3. Proposed Framework, their Structure and Mapping

Before we start the development, we defined the common terminologies, metamodel modeling and their validation within the EAMS which allowed us to know how the REST Service, i.e, POST, GET, DELETE and PUT work. Additionally, we had to validate the metamodel according the description of EAMS Services provided by owner.

Figure 5 is a sample of how different models are related to each other and how they are mapped. As can be seen in Figure 5, each framework define own terminology, structure and so foth. For instance, it can be seen in figure below that the work product name exist in COBIT and in ISO 33053 is named *Related Information Items* even having a similar meaning.

5.4. Proposal Description and Prototypes

The prototyping phase includes different activities which should be taken in consideration such as brainstorming, low-fidelity interfaces design also named sketches and high-fidelity interfaces design. To all of these activities is also included two type of meetings: *formative* and *summative* which was performed prior to the tool development and for some reason during development process. However, before the development process start, many steps was

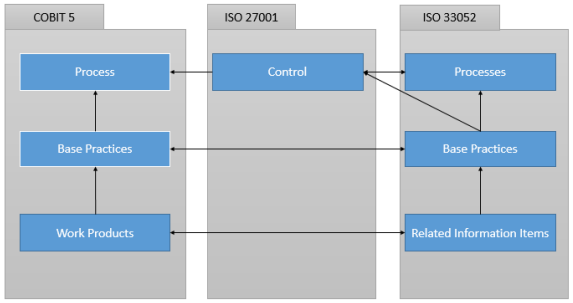


Figure 5: Different approaches and structures for assessment model

considered, e.g, task definition, meeting scheduling, list of requirements specifically the *functional requirements* and *non-functional requirements*.

Several interfaces were designed for high-fidelity prototypes, but for this purpose we will present only one of them which correspondent for assessment creation and their relations due their importance. Figure 6 presents the most important features of proposed tool. In Figure 6 is possible to see the relation amongst different processes and from different frameworks, a list of base practices and work products. Inside each base practice, there are a list of related work products and for each work product is possible to upload files at same time for all related instances and for singular as well.

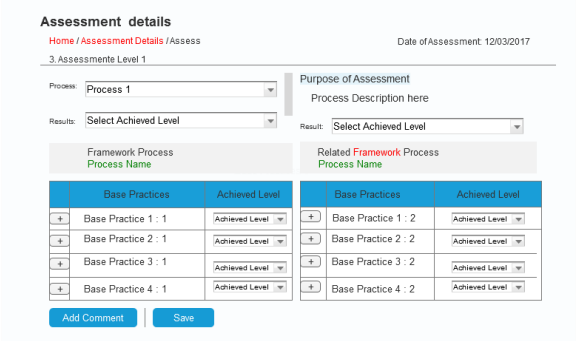


Figure 6: User Interface Prototype: Assessing interface

6. Demonstration

The tool demonstration was based on in one of the largest banks located in Portugal. The assessed process is *DSS02 - Manage Service Requests and Incidents* which belong the COBIT 5 framework and that falls in the management for enterprise IT area of COBIT 5 with the purpose of *achieving increased productivity and minimise disruptions through quick resolution of user queries and incidents*. However, to demonstrate how the software tool is addressed

for environments which work with one or more frameworks was added the following process *COM-08 Operational planning* from ISO 33052.

Furthermore, the demonstration comprised two phases, which in the first phase is showing how the assessments are performed taking in account one framework or without relations, i.e, when the selected processes belongs to the same framework or if there are not available relations amongst the selected processes. In second phase is showing how those assessments are performed when two or more frameworks are related, i.e, if is there some relations amongst the selected processes. Usually, this happen when the selected processes belongs to different frameworks (COBIT, ISO, ITIL and so on).

7. Evaluation

This section describe the obtained results during the demonstration of the tool. Offermann. [8] proposes three approaches for software evaluation, (i) *experty survey*, (ii) *Laboratory experiments* and (iii)*Case Study / action research*. In both approaches, it requires the hypothesis refining once the solution reaches a stable state, i.e, defining the general research hypothesis and the smaller hypotheses.

Prat at all [28] define a list of hierarchy of criteria for information system artifact evaluation. Amongst all of them, the widely used [28] is *goal/efficacy* which is characterized by following criteria: *efficacy, efficacy, validity* and *generality*.

However,the proposed methods to perform evaluation, will help us to collect the necessary evidences to evaluate the proposed criteria and also to know if the main objectives has been achieved or not. Table 3 show the tasks list to be evaluated and validate.

Task Names	Description
Task:1	Create Account
Task:2	Login
Task:3	Create Assessment
Task:4	Assessment details
Task:5	Edit Assessment
Task:6	Delete Assessment

Table 3: Task List

7.1. Metrics and Estimation

This process will be centered in proposed criteria. The efficiency define the needed time to complete tasks as the number of clicks needed to perform tasks, so in this point we will compare the obtained results after evaluation with our expectation. Table 4 illustrates the expected results after performing each of task.

While the efficiency compare the needed time to complete tasks successful and number of clicks, the effectiveness is focused on numbers of errors and task completion rate. So, likewise the efficiency, in effectiveness we need to define the estimated number of errors for each task. Table 5 show these list for each of task.

Task	Estimated execution time and number of clicks
Task:1	On average, users should spend less than 80 seconds to perform this task at the first attempt and 60 seconds next times and than less 2 clicks..
Task:2	On average, users should spend less than 45 seconds to perform this task with 1 click.
Task:3	On average, users should spend less than 5 minutes to perform this task at the first attempt and 4 minutes next times than less 30 clicks.
Task:4	On average, users should spend less than 2 minutes to perform this task at the first attempt and 90 seconds next times and than less 6 clicks.
Task:5	On average, users should spend less than 3 minutes to perform this task at the first attempt and 2 minutes next times and with 15 clicks.
Task:6	On average, users should spend less than 15 seconds to perform this task with 3 clicks.

Table 4: Efficiency: Estimated execution time and number of clicks

Task	Estimated number of errors
Task:1	2
Task:2	1
Task:3	2
Task:4	1
Task:5	2
Task:6	0

Table 5: Effectiveness: Estimated number of errors

The satisfaction, validaty and generality metric are based on System Usability Scale questions [29]. The system usability scale questions have 10 question related the user satisfaction where the final users only must rank each question from 1 to 5 based on how much they agree with the statement. The number 5 means that they agree completely and 1 means that they disagree vehemently.

This questionnaire will be provided to the final user after performing the indicated tasks. Table 7 illustrates the system usability scale question list. Lets denote each system usability scale question in *Q:n* where *n* denote the question number. So, Table

6 illustrated the expected answers which based on final results we will compare the achieved results with expected.

Q:1	Q:2	Q:3	Q:4	Q:5	Q:6	Q:7	Q:8	Q:9	Q:10
5	1	1	1	1	1	5	1	5	1

Table 6: Satisfaction: Estimated SUS answers

Question	Description
Q:1	I think that I would like to use this system frequently
Q:2	I found the system unnecessarily complex
Q:3	I thought the system was easy to use
Q:4	I think that I would need the support of a technical person to be able to use this system
Q:5	I found the various functions in this system were well integrated
Q:6	I thought there was too much inconsistency in this system
Q:7	I would imagine that most people would learn to use this system very quickly
Q:8	I found the system very cumbersome to use
Q:9	I felt very confident using the system
Q:10	I needed to learn a lot of things before I could get going with this system

Table 7: List of SUS Questions

8. Data Collection and Analyses

In order to validated the obtained data has been used a inferential statistics named *Confidence Interval*, with following descriptive statistics values *Mean, Median* and *Standard Deviation*. Once calculated these values, for each task we compute the confidence interval, i.e, the minimums and maximums values. If the estimated value for each task is within these confidence intervals mean that, our hypotheses are not true. In other words, our estimation are false and there is a need to review certain functionalities.

To get the confidence interval, we have to consider the values of mean, standard deviation, the probability to exceed the estimated time ($\alpha = 0.05$) for the sample data and the level of confidence (in this case will be 95% of confidence interval).

8.1. Interpreting Results based on time

So, for *Task:1, Task:2, Task:3, Task:4 and Task:5*, we conclude with 95% of confidence that the estimate goals was achieved successful since that the medium time spent by the users is less than estimated times. Unfortunately we can not say the same for *Task:6*. The estimated time is less than medium time that users spent performing this task, and beside that, the estimated time is not within the obtained *Xmin* and *Xmax* values which means the expected goal was not achieved as expected.

8.1.1 Interpreting Results based on number of clicks

Likewise, in previous analyses based on time, for *Task:1, Task:2, Task:3 and Task:4*, we conclude with 95% of confidence that the estimate goals was achieved successful with medium number of clicks less then than estimated number of clicks. For *Task:5* the defined goal was not successful achieved but archived as well. The medium number of clicks obtained is sixteen and the estimated number of clicks is fifteen, but the obtained values of *Xmin* and *Xmax* are *14,7* and *17.2*. The value *15* is within the range of minimum means and maximum and we can say that, the fifteen value is acceptable and the goal was achieved.

Unfortunately for *Task: 6* the goal was not achieved with confidence of 95%, since the estimated value is less than the medium number of clicks made by the users to perform the task. It means that some users made more clicks than expected.

8.1.2 Interpreting Results based on number of errors

For *Task:1, Task:2, Task:3, Task:4 and Task:5*, with 95% of confidence we conclude that the goal was achieved, since that the mean number of errors practiced by the users for these tasks are above than the obtained values and also above the *Xmax* values. The *Task:6* once again it is not satisfied. The estimated number of errors for *Task: 6* is zero, but many errors were collected, which means that estimates fail for some reasons.

8.1.3 Interpreting Results based on SUS answers

The analysis of the results of these questionnaires will be done differently from the previous one due to some reasons. The questions are ranked from 1 to 5, which means that, based on the values obtained for each question, we have to determine which average values are closer to our expected value. So, the values of the means will be rounded up in excesses or

in defects, where afterwards we verify which values are more closer to the expected values defined.

Question 1 : user satisfaction level is lower than expected. We said with 95% of confidence that users found the application little complex, since that the expected value is larger than the mean of user satisfaction level.

Questions 2, 5, 7, 8 and 10 : we consider that our expectations are in agreement with the users;

Question 3 : the complexity of manipulated information has made users feel cumbersome. As can be seen the obtained values is closer to the expected, but we can not say that we achieved our goal successfully because is around 20% below to the expected.

Question 4 : in 95% of confidence, the user perspective is also not good, they considered that application is complex and will require the support of a technical person.

Question 6 : the interface elements such as buttons, text fields and so on justify the obtained results which was not as expected.

Question 9 : the manipulated information and the environment where the application is addressed was the only reason which the users gave us. Fortunately for some users which know how to deal with these frameworks the answers was different and satisfactory as well.

9. Conclusions

A continuous process evaluation activity is a very important step that should be carried out by organizations in order to improve the quality of their processes, products and services, but the huge number of frameworks and their complexity, brings new insights and challenges in this field of study. Some of these challenges are related to (i) *multiframework* environments and (ii) development of a software tool which implement these ontology.

Bearing in mind all these challenges and their importance, is proposed a Web-Based software tool with EAMS foundation, which implements the defined ontology (COBIT, ISO's, ITIL, etc). The solution is detailed in Section 6 with a real world problem, in that case a *Banking Sector*. The solution description was extended in Section 5 where has some prototypes, diagrams and models.

Once performed the evaluation, was possible to know which task were achieved and which of them should be improved. As such, we did some improvements and the other complexes tasks such as *Features optimization, Application accessibility, Entity Relationships, Reporting* and *Dashboard* were addressed for future works.

Acknowledgements

I would like to express my sincere gratitude to God for blessing me more than I deserve as well as for my wonderful life and good health. Likewise, I would like to thank my mother Teresa Rafael Felisberto, my Father Samuel Felisberto and my brothers, because without their continuous support and encouragement I would never ever been able to achieve my goal.

My most sincere thanks go to my advisor and mentor, Professor Miguel Leitão Bignolas Mira da Silva. I thank him for introducing me to the wonders and frustrations of scientific research and also guiding and supporting me during the development of this thesis. I also want to thank Rafael Almeida for numerous helpful advices and inspiring discussions that were indispensable to reach my goal.

Finally, I would like to express my gratitude to Nelson Teodoro, the Country Manager of Novabase Mozambique and other staff members of Novabase Mozambique, and Portugal who helped me during this period at Instituto Superior Técnico.

References

[1] David Tuffley. A process-driven socio-technical approach to engineering high-performance organisations. *Advances in social and organizational factors*, pages 551–561, 2012.

[2] Mell P and Grance T. The nist definition of cloud computing - recommendations of the national institute of standards and technology. *National Institute of Standards and Technology*, September 2011.

[3] M.C Paulk. Comparing iso 9001 and the capability maturity model for software. *Software Quality Journal*, (2):245256, 1993.

[4] B Mutafelija and H Stromber. Iso 9001:2000 cmmi v1.1 mappings. software engineering institute, 2013.

[5] Link Consulting. Learn about eams, url: http://www.linkconsulting.com/eams/ :online; acceessed 02-september-2017.

[6] Alan R. Hevner, Salvatore T. March, Jinsoo Park, and Sudha Ram. Design science in information systems research. *MIS Q.*, 28(1):75–105, March 2004.

[7] Ken Peffers, Tuure Tuunanen, Marcus Rothenberger, and Samir Chatterjee. A design science research methodology for information systems research. *J. Manage. Inf. Syst.*, 24(3):45–77, December 2007.

[8] Philipp Offermann, Olga Levina, Marten Schönherr, and Udo Bub. Outline of a design science research process. In *Proceedings*

of the 4th International Conference on Design Science Research in Information Systems and Technology, DESRIST '09, pages 7:1–7:11, New York, NY, USA, 2009. ACM.

[9] Mario A. Pardo, Norman Silverberg, Diane Gendron, Emilio Beier, and Daniel Palacios. pardo et al 2013, 09 2013.

[10] Jörg Becker, Ralf Knackstedt, and Jens Pöppelbuß. Developing maturity models for it management. *Business & Information Systems Engineering*, 1(3):213–222, 2009.

[11] Marco Vicente, Nelson Gama, and Miguel Mira da Silva. Using archimate to represent itil metamodel. In *Business Informatics (CBI), 2013 IEEE 15th Conference on*, pages 270–275. IEEE, 2013.

[12] Stuart Macgregor. *Enterprise Architecture as competitive advantage*. Johannesburgl - The Open Group, March 2011.

[13] F Liang and A Rout, T. Tuffley. Appraisal assistant beta, 2007. URL: https://www.sqi.griffith.edu.au /AppraisalAssistant /about.html :Online; acceessed 02-September-2017.

[14] Appraisal wizard and wizard lite, url: http://isd-inc.com/tools.appraisalwizard/ :online; acceessed 26-september-2017.

[15] Appraisal wizard 7 user guide, url: http://members.isd-inc.com/files/awv7userguide.pdf :online; acceessed 06-september-2017.

[16] V7 product suite features, url: http://members.isd-inc.com/files/v7 20product 20features.pdf :online; acceessed 06-september-2017.

[17] Spice-lite assessment tool, url: http://www.spicelite.com/cms/en/ :online; acceessed 26-september-2017.

[18] Sema Gazel, Ebru Akçapinar Sezer, and Ayca Tarhan. An ontology based infrastructure to support cmmi-based software process assessment. *Gazi University Journal of Science*, 25(1):155–164, 2012.

[19] Cobit 5 self-assessment tooll, url:https://cobitonline.isaca.org/landing-sat/ :online; acceessed 26-september-2017.

[20] Cesar Jesus Pardo-Calvache, Felix Oscar Garcia-Rubio, Mario Piattini Velthuis, Francisco Jose Pino-Correa, and Maria Teresa Baldassarre. A reference ontology for harmonizing processreference models. *Fac. Ing. Univ. Antioquia*, (73):29–42, December 2014.

[21] CMMI Institute Partner. Cmmipal v1.0, url: http://www.chemuturi.com/cmmipaldtls.html :online; acceessed 26-september-2017.

[22] Cmm quest v1.3, url:http://www.cmm-quest.com/cms/en/ :online; acceessed 26-september-2017.

[23] Spice 1-2-1 forinternational standards, url: http://www.spice121.com/cms/en/ :online; acceessed 06-september-2017.

[24] Model wizard and model mapper, url: http://isd-inc.com/tools.modelwizard/ :online; acceessed 26-september-2017.

[25] Ayca Tarhan, Oktay Turetken, and Fadime Ilisulu. Business process maturity assessment: State of the art and key characteristics. 2015.

[26] Philippe Kruchten. *The rational unified process: an introduction*. Addison-Wesley Professional, 2004.

[27] Wikipedia. Class diagram — wikipedia, the free encyclopedia, 2017.

[28] Prat N, Comyn-Wattiau I, and Akoka J. Artifact evaluation in information systems design-science research - a holistic view. *PACIS 2014 Proceedings. Paper 23*, 2014.

[29] John Brooke et al. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7, 1996.