TEST Engineering & Management

March - April 2020
ISSN: 0193-4120 Page No. 13428 - 13433

Software Testing of Information Systems in Peruvian Public Organizations

Elizabeth Mendoza[1], Consuelo Espino[2], Ciro Rodriguez[3], Alex Pacheco[4]

[1,2,3,4] Postgraduate School of the National University Federico Villarreal

Article Info
Volume 83
Page Number: 13428 - 13433
Publication Issue:
March - April 2020

Article History
***Article Received**: 24 July 2019*
***Revised**: 12 September 2019*
***Accepted**: 15 February 2020*
***Publication**: 20 April 2020*

Abstract

To have a well-defined process for software tests on computer systems is necessary to ensure the implementation of each system requirement as tested for conformity, which implies a double effort by the staff of development and an additional cost in system maintenance. The objective of this paper is to describe the importance of the software testing process and establishes guidelines to regulate activities and tasks during the development and maintenance of IT systems in the IT areas to confirm that the integrated software product meets the requirements defined, the Peruvian Technical Standard NTP-ISO: 122017: 2016 “Software and Systems Engineering. Software life cycle processes” and the NTP-ISO / IEC 27001 Information TechnologySecurity techniques.Information security management systems, which cover the Life cycle support process, system quality tests, and security in the development and support processes. The key in the proposed model is the definition of roles and responsibilities, the activities of each sub-process, and the use of control templates, which has allowed to reduce failures in systems already in production.

Keywords: *software testing, information systems, technical standard ISO, software maintenance*

I. INTRODUCTION

The vast majority of organizations in the Peruvian state do not have a well-defined process for software tests performed on information systems, before and after putting them into production.

[17] The absence of the process of testing the information systems software results in instability and insecurity of the product, which implies a double effort by the development staff and an additional cost in system maintenance.

Having problems during the process of the implementation of information systems, procedures, methodologies, and guides have been found to minimize the findings detected; this is where the Software Testing Process of Computer Systems takes a significant role, it is one of the key processes for all organizations that perform software engineering, this enables better quality during the software development and implementation process.

According to [1], The software qualification testing process, its purpose is to confirm that the integrated software product meets your defined requirements. This standard contributes to the results of the verification process of [2].

In [3]. It indicates that the tests are an integral part of the general development of the software, the tests should intentionally try to make things go wrong to determine if things happen when they should not or should not happen when they should. [4], a glossary of software engineering Technology, defines the tests as "The process of operating a system or component under specified conditions, observing or recording the results, and making an evaluation of some aspects of the system or component."

In [5], the control A.14.2.8 Tests of system security functionality must be carried out during development. In control A.14.2.9, the acceptance testing programs and related criteria must be established for new information systems, updates, and new versions.

In [6]. It indicates that the tests must be aligned to the development process; It is key then to affirm that it is important to carry out a test process that includes: "Review of the requirements, carrying out documentary analyzes, identification of defects, functional and non-functional tests, dynamic and static tests, integration tests, reports of confidence in the quality level, information for decision making, continuous improvement plans "[7]. The mentions that the testing process impacts the risks of the software product; therefore, software companies need to formulate and adapt software quality testing process

II. MATERIALS AND METHODS

The following were taken into account in the materials and method:

Research techniques and instruments:

Direct observation: the main activities in the Software Testing process of the information systems were monitored to detect the current status.

Data analysis.

Logical content techniques or qualitative analysis to obtain conclusions.

Unstructured Interviews

A series of interviews were conducted with the staff that develops, tests, and implements the software to obtain detailed information on this activity.

Referential Review:

The study of the Peruvian Technical Standard NTP-ISO / IEC 12207: 2016 Software and Systems Engineering, software life cycle process; NTP-ISO / IEC 27001; ISO/IEC/IEEE 15288:2015 Systems and software engineering — System life cycle processes;Standard Glossary of Software Engineering Terminology IEEE 610.

Instruments:

- Peruvian Technical Standard NTP-ISO / IEC 12207: 2016.
- NTP-ISO / IEC 27001.
- ISO/IEC/IEEE 15288:2015 Systems and software engineering — System life cycle processes;
- Standard Glossary of Software Engineering Terminology IEEE 610.
- Sonarqube
- Fortify Static / OWASP
- Jmeter
- Bizagi Modeler tool.
- Computer requirements.
- Meeting Minutes.
- Development methodology.
- Microsoft Office package, where Word, Excel, and PowerPoint programs will be used.
- Laptop with internet access.
- Office resources.

Proposed model:

The activities of a test process must comply with the following: planning and debugging: debugging is a systematic process to find and reduce the number of errors or defects in a computer program. Control; select test conditions; design and execute test cases; check the results; evaluate the results criteria; prepare reports on the process and application under test, including experience logs [8].

Figure 1 indicates the processes and activities of software testing of computer systems: roles, responsibilities, and environments of the testing process.

Testing process of the development team.

The project manager and the development team must prepare a set of tests that are most likely to discover software defects. Documentation of test cases must be made. This Test must be carried out in the development environment, the newly designed and developed tests, as well as the existing ones, if required.

 In the Execution of the Designed and Developed Tests, the Programmer will execute the tests based on the test cases that were created.

- ✓ In the Execution of Regression Tests, the Programmer will re-execute the tests that previously produced errors or the functional tests that have undergone modifications. This way the corrections are verified and errors in the application caused by corrections or modifications to the code can be detected.
- ✓ Execution of the system tests, the Analyst or the Programmer will execute the system tests defined in the design and in which they are responsible for their execution.

When all defects have been corrected and are in compliance, the Project Manager must request the corresponding review from the quality team.

Testing process of the Software Quality team

It is important to develop a test plan is, in this plan, you should have the following information: the scope of the tests, supporting documents, identify the technological environment (on type of server, database software, communications, security hardware, etc.), list of materials, test data, tools, number the characteristics that will be tested and not tested identify the types and techniques of test used in this process, develop test cases, identify risks and contingency. The immediate boss must approve this plan.

The quality manager will prepare the test plan; the quality team will execute the plan, if any observations are presented at the stage of testing, the quality manager will request the project manager to correct the observations found through a logbook of observations. The quality team must verify that the observations have been made and that the date and name of the person responsible for the correction of each observation have been recorded. When verifying and validating the computer system will coordinate with the team responsible for the production pass.

Testing process in the production environment

When the computer system is in production, you must perform postproduction tests called the white march; with this test, you can verify and validate that the computer system that meets the user's satisfaction. At the end of this test, a final report will be made; if there is no observation, the computer system will be released.

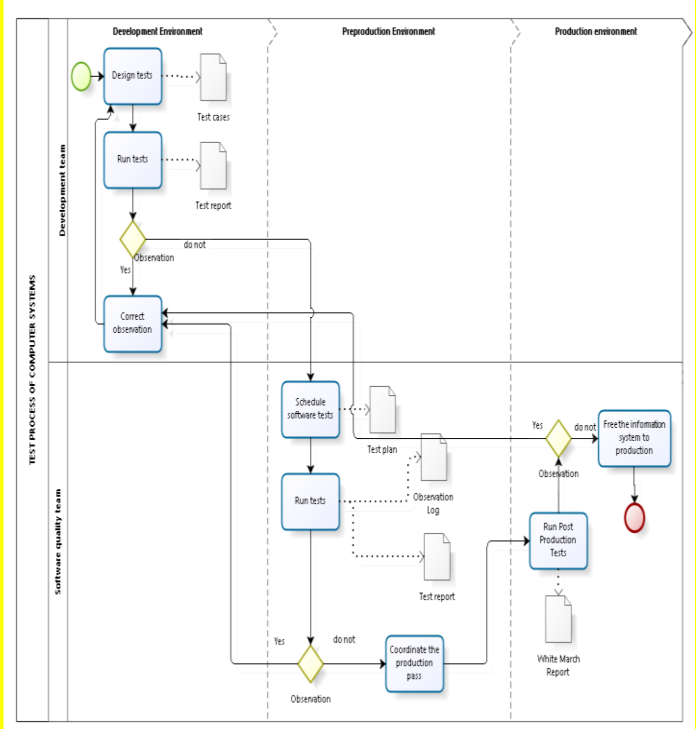

Figure 1: Computer systems testing process

During the life cycle of the software projects, the developers carry out the tests as an activity that seeks to guarantee that the product meets the requirements of the users [9], ensuring its quality.

When we refer to mobile technology tests, the quality team must always keep in mind the mental map in Figure 2.

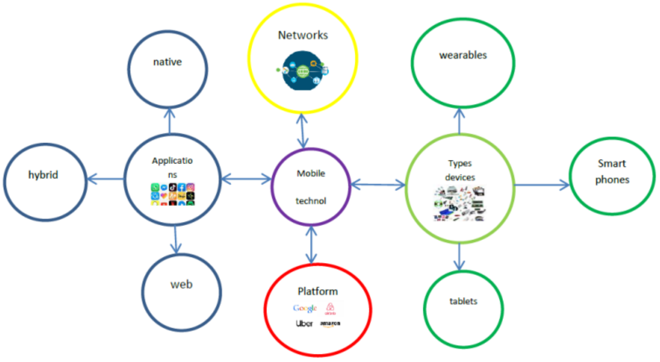

Figure 2: Mind map of mobile technology

The quality team, according to the test strategy designed, can evaluate the following characteristics for an application:

- ✓ The installation and uninstallation process
- ✓ The error control you have
- ✓ Network changes
- ✓ Behavior in multitasking scenarios
- ✓ Visualizations in horizontal and vertical mode
- ✓ Tests on different types of resolutions and screen sizes.
- ✓ If integration with the server is presented, keep in mind the different information exchange scenarios.
- ✓ Check if the application affects or is affected by other applications.
- ✓ The behavior against the use of the device's controls.
- ✓ Add scenarios associated with the characteristics of these devices, such as turning, shaking, performing different types of touch gestures on the screen.
- ✓ Testing interruptions for messages, calls, connection, and disconnection of cables

About test types

Unit tests [10] They seek to test the smallest units of the software, the software component or module, to validate each one, taking into account that the software development methodologies separate the complexity of the requirements between the different programmers functional over non-functional.

Component tests are aimed at locating defects and checking the operation of software modules, programs, objects, classes, etc. That can be tested separately; that is, they can be performed independently of the rest of the system depending on the context. Design of test cases: "Requirements of the components, design of detail in the use cases, code in the module or component" [11].

Integration tests are responsible for testing the interfaces between the components or modules; for example, the user validation component with the operating system, the file system in integration with the hardware, etc. Design of test cases: "Software design, architecture, workflows, use cases, typical test objects must be taken into account: 1. Subsystem database, 2. Infrastructure, 3. Interfaces, 4. System configuration, 5. Configuration data" [12].

System tests [13] They seek to test the system as a whole and with other systems with which it relates, to verify that the specifications, both technical and functional, are met.

Acceptance tests [13, 14] They are performed by users who verify that the system or application is ready for production.

Regression tests [13] Execution of test cases previously performed when changes in the software are implemented.

III. RESULTS

With the software testing process of the information systems, I was able to reduce failures in the systems put into production; it is essential in the quality of the products and services provided by the state companies that develop software, they also reduce the cost of maintenance. In carrying out this process, meetings were held with the IT Department as a result of these meetings, the baseline of the elements generated by the process was obtained, as well as the criteria for the integrated software that demonstrates compliance with the software requirements being developed; test results are recorded; Regression strategies to repeat the integrated software test when a change is made to the software elements are developed and applied; test strategies for mobile developments; generate collaborative methodologies workflow creation; Modification of the old process.

For the implementation of the start-up of the new process: Five (05) computer requirements were carried out:

- ✓ The code quality improves since we can detect errors at an earlier stage of development and faster.
- ✓ You can work in a more agile way, as it facilitates changes and favors integration.
- ✓ Reduce the cost of maintaining the project.
- ✓ Through test-driven development, it helps us improve the design of our software.
- ✓ Increase satisfaction and inspire user confidence
- ✓ Promotes organization, productivity, and efficiency.

Table 1: Software testing process comparison

Description	Before the Process	After the Process
Design test findings	15	05
Unit Test Findings	120	15
Component Test Findings	80	10
Integration test findings	40	02
System Test Findings	250	10
Regression Test Findings	35	05
Acceptance test findings	25	05

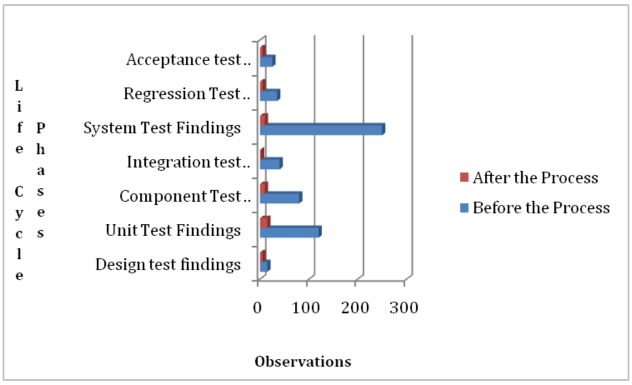

Figure 3: Software testing process comparison

Once the software testing process of the computer systems is completed, a considerable improvement of the five (05) computer requirements can be evidenced, minimizing design, component, integration, system, regression and acceptance errors, higher product quality in information systems.

IV. DISCUSSION

As [1] the standard indicates that the process ensures that the implementation of each system requirement is tested for conformity and that the computer system is ready for delivery; For companies in the Peruvian state, it is applicable as long as the organization is committed, respecting the procedures efficiently. Verification must be performed on each deliverable, and validation is performed with the user.

The software testing process of the information systems has allowed more objective tests to be carried out on the solutions obtained in the projects. The documentation of the process is done in this most effective way; the findings that are detected are monitored and only closed when it has really been verified that they were solved.

All the results collected from the realization of the testing process in the different projects analyzed for this work allow us to prepare the quality laboratory test report.

ACKNOWLEDGMENTS

This research paper acknowledges Dr. Ciro Rodriguez for his unconditional contribution.

REFERENCES

1. Peruvian Technical Standard NTP-ISO / IEC 12207: 2016 Software and Systems Engineering, software life cycle process,
2. ISO/IEC/IEEE 15288:2015 Systems and software engineering — System life cycle processes.
3. Abu Sayed Mahfuz. Software Quality Assurance Integrating Testing, Security, and Audit. Version: 20160408, 2016.
4. Standard Glossary of Software Engineering Terminology IEEE 610.
5. The Peruvian Technical Standard NTP-ISO / IEC 27001 INFORMATION TECHNOLOGY. Security techniques, Information security management systems.
6. J. A. Mera Paz, “Análisis del proceso de pruebas de calidad de software”, Ingeniería Solidaria, vol. 12, no. 20, pp. xx-xx, oct. 2016. doi: http://dx.doi.org/10.16925/in.v12i20.1482.
7. Müller Thomas, Libro Programa de Estudio de Nivel Básico Para Probador Certificado, istqb, Version 2013 Pág. 14. Disponible en: http://www.istqb.org/ downloads/syllabi/foundation-level-syllabus.html.
8. International Software Testing Qualifications Board [istqb], “Certified Tester Foundation Level Syllabus. Released version 2013”, 2013 [en línea]. Disponible en: http://www.istqb.org/downloads/ send/2-foundation-level-documents/3-foundationlevel-syllabus-2011.html4
9. D. E. Soto Durán, A. X. Reyes Gamboa, y J. Jiménez Builes, “Aplicación de la Gestión de Conocimiento al proceso de pruebas de software”, Ingenierías USBMed, vol. 8, no. 2, pp. 6-13, 2017. DOI: https:// doi.org/10.21500/20275846.2836
10. B. Bruegge, A. H. Dutoit. Ingeniería de software orientado a objetos. México: Pearson educación, 2016.
11. Müller Thomas, Libro Programa de Estudio de Nivel Básico Para Probador Certificado, istqb, Version 2017 Pág. 14. Disponible en: http://www.istqb.org/ downloads/syllabi/foundation-level-syllabus.html
12. S. M. Velásquez, D. E. Monsalve Sossa, M. E. Zapata, M. E. Gómez Adasme, y J. P. Ríos “Pruebas a aplicaciones móviles: avances y retos, Lámpsakos, No. 21, pp. 39-50. (enero-junio, 2019). DOI: https://doi.org/10.21501/21454086.2983.
13. P. Bourque, and R. E. Fairley (Eds.). Guide to the software engineering body of knowledge (SWEBOK (R)): Version 3.0. IEEE Computer Society Press.
14. H. A. Parada Gélvez. Contribución a la gestión de los procesos de pruebas de software y servicios. Madrid: Universidad Politécnica de Madrid, 2016.
15. V. Esterkin y C. Pons “Quality evaluation in software development model driven by models”, Inter-American Open University, Argentina, 2017.
16. R. S. Pressman y B. R. Maxim, Software Engineering: A practitioner's approach, 8th ed. New York, NY, USA: McGraw-Hill Education, 2015.
17. E.Mendoza, C.Rodriguez, D.Esenarro “Configuration Management of informationsystems in Peruvian Government organizations”. International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8 Issue-12S2, October 2019.DOI: 10.35940/ijitee.l1007.10812s219