EVALUACIÓN DE TÉCNICAS DE TESTING A TRAVÉS DE UN EXPERIMENTO CONTROLADO EJECUTADO EN UN AMBIENTE UNIVERSITARIO

Sergio Zapata[1], Ariadna Soto[2], Pablo Sánchez[3], Cintia Ferrarini[4], Myriam Herrera[5], Laura Aballay[6]

Abstract — *This paper presents the definition, planification, operation and analysis of results obtained from a controlled experiment. Two testing techniques such as: Equivalence Partitioning and Table of Decision where compared in this experiment regarding their effectiveness and efficiency. Advanced students of degree programs in computer science from the Universidad Nacional de San Juan (Argentina) participated of this experiment. Significant evidences about both techniques of functional testing were found. Moreover, characteristics such as easy to learn and usability was identified in each techniques. Also, student achieved high level of satisfaction for their active participation in such experiment of the Empirical Software Engineering.*

Index Terms — *Software Engineering, Education in Software Engineering, Empirical Software Engineering.*

INTRODUCCIÓN

El software, se ha convertido en un bien que día a día extiende su aplicación a distintos aspectos de la vida humana, abordando problemas cada vez más complejos y críticos. Por ende, requiere de mayor calidad en su desarrollo [8]. Lamentablemente, la industria del software no está acompañando esa expansión con una mejora en la calidad de los productos que libera al mercado. La falta de calidad de los productos de software, es reconocida tanto por la academia, como por la industria.

La calidad del software se mide en función del grado con el que un sistema cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario [10] . Existen distintas herramientas que ayudan a los ingenieros de software a proveer calidad a sus productos siendo una de las más tradicionales el testing de software [13]. Otras herramientas complementarias que aportan a la calidad son las inspecciones de software[6] y los modelos de procesos de software centrados en la calidad (como CMMI-SEI). [5]

El testing es un proceso que tiene por objetivo encontrar defectos en el código fuente del software que se está evaluado. [11] Es considerado como *"... el proceso de operar un sistema o componente bajo ciertas condiciones específicas, observando y almacenando los resultados, y haciendo una evaluación de un aspecto del sistema…"* [13]

Existe una gran diversidad de técnicas de testing de software, clasificadas en técnicas funcionales y técnicas estructurales; lo que hace que, al momento de seleccionar una de ellas se deba tener ciertas consideraciones, como por ejemplo: eficiencia, eficacia, facilidad de aplicación, intención de uso, entre otras. Por lo que es importante que ingeniero de software, cuente con información que le permita tomar una decisión acertada, al momento de optar por una técnica de testing,.

En este trabajo se presenta una comparación de dos técnicas de testing funcionales, *Partición de equivalencia* y *Tabla de decisiones*, mediante un experimento controlado ejecutado en un escenario educativo, con la participación de estudiantes de grado, a fin de evaluar cuál de estas técnicas es más efectiva para la detección de errores, desde el punto de vista del tester.

Las técnicas de testing [1] se pueden clasificar en técnicas de caja blanca y técnicas de caja negra. En las primeras el proceso se basa en un análisis de la estructura interna (algoritmo, código fuente) del módulo de software; mientras que en las técnicas de caja negra o funcionales se analizan los resultados o salidas de esos módulos, las que surgen en función de los datos de entrada provistos a dichos módulos. Entre las técnicas de prueba[15],[16] funcionales más difundidas se encuentran *Clase de Equivalencia* y *Tabla de Decisión*, motivo por el cual fueron seleccionadas como objeto de estudio en este trabajo.

1 Sergio G. Zapata, Director del Instituto de Informática de la Facultad de Ciencias Exactas Físicas y Naturales de la Universidad Nacional de San Juan – Argentina, Meglioli Rivadavia CP 5400, San Juan, Argentina, szapata@iinfo.unsj.edu.ar.
2 Ariadna Soto, Licenciada en Sistemas de la Información, recientemente egresada de la Facultad de Ciencias Exactas, Físicas y Naturales de la Universidad Nacional de San Juan. – Argentina, Meglioli Rivadavia CP 5400, San Juan, Argentina, ariadnasoto@yahoo.com.ar
3 Pablo, Sánchez, Licenciado en Sistemas de la Información, recientemente egresada de la Facultad de Ciencias Exactas, Físicas y Naturales de la Universidad Nacional de San Juan – Argentina, Meglioli Rivadavia CP 5400, San Juan, Argentina,. pablito_listas@yahoo.com.ar.
4 Cintia Ferrarini Oliver, Profesora Auxiliar de 1 era categoría del Instituto de Informática de la Facultad de Ciencias Exactas Físicas y Naturales de la Universidad Nacional de San Juan – Argentina, Meglioli Rivadavia CP 5400, San Juan, Argentina, ferrarini@iinfo.unsj.edu.ar / ferrarinicintia@yahoo.com.ar.
5 Myriam Herrera., Profesora Adjunta Exclusiva del departamento de Informática de la Facultad de Ciencias Exactas Físicas y Naturales de la Universidad Nacional de San Juan – Argentina, Meglioli Rivadavia CP 5400, San Juan, Argentina, ,mherrera@iinfo.unsj.edu.ar
6 Laura N. Aballay, Jefe de Trabajos Prácticos del Instituto de Informática de la Facultad de Ciencias Exactas Físicas y Naturales de la Universidad Nacional de San Juan – Argentina, Meglioli Rivadavia CP 5400, San Juan, Argentina, laballay@iinfo.unsj.edu.ar

Técnicas de Testing: Tabla de Decisión y Clases de Equivalencia

Una Tabla de Decisión [12] es un mecanismo que permite sistematizar la asociación de condiciones de entrada con acciones o funcionalidades de salida del modulo de software, de esta manera se obtienen un conjunto de reglas o casos de test no redundantes y consistentes. Sólo por usar esta técnica durante la preparación de una prueba ya se pueden detectar algunos de los mayores defectos en las especificaciones funcionales. [7]

En la técnica Clases de Equivalencia [12],[13] el conjunto universo de datos de entradas se particiona en subconjuntos disjuntos o clases de equivalencias. El criterio básico de partición es que el módulo de software se comporte de igual forma para cualquier valor de dato que pertenezca a la misma partición o clase de equivalencia. De esta manera se obtiene un número administrable de casos de pruebas que luego serán aplicados en el proceso de testing propiamente dicho.

Las técnicas de Ingeniería de Software requieren ser estudiadas experimentalmente, si se pretende que la Ingeniería de Software no sea solo una disciplina teórica. La Ingeniería de Software es una "gran ciencia"; y la experimentación es un ingrediente necesario de cualquier gran ciencia [3].

Mediante los métodos empíricos es posible evaluar nuevos aportes de la Ingeniería de Software antes de que sean incorporados en los procesos de desarrollo de software de las empresas [9].

Los estudios empíricos más utilizados en ambientes educativos son los experimentos controlados [2] en donde la participación del alumno toma relevancia tanto por ser sujeto de los mismos como destinatario de una acción pedagógica.

Este artículo se presenta cuatro secciones: Diseño Experimental, donde se plantea el objetivo y la planificación del experimento; Ejecución del experimento, en el que se describe como se llevó a cabo, con el fin de recabar los datos para ser analizados (operación de un experimento); Análisis e interpretación de los resultados, en la cual se muestran los resultados obtenidos del procesamiento de los datos, teniendo en cuenta las variables eficiencia, eficacia, intención de uso y facilidad de aplicación y, por último, Conclusiones y trabajos futuros.

DISEÑO EXPERIMENTAL

En esta sección se presentan las etapas del experimento. Las mismas se han elaborado siguiendo las pautas de Wohlin [19], que propone para el proceso experimental, las siguientes etapas principales:

- La "definición" de los objetivos del experimento, el porqué del mismo
- La "planificación" del experimento
- La "operación" o ejecución
- Y el "Análisis e interpretación de los resultados"

Definición

En esta etapa se determina el fundamento del experimento. Tomando la plantilla GQM (Goal-Question-Metrics) [4] se resumió la definición y el objetivo del experimento de la siguiente forma: "Analizar las técnicas funcionales de Prueba de Software *Partición Equivalente* y *Tablas de Decisión*, con el propósito de evaluarlas, respecto a la eficiencia, eficacia, facilidad de aplicación e intención de uso; desde el punto de vista del Testeador/Tester; por estudiantes de 5to año de la carrera Licenciatura en Sistemas de Información de la Facultad de Ciencias Exactas, Físicas y Naturales de la Universidad Nacional de San Juan, Argentina."

Planificación

Es la etapa donde se determina el plan del experimento, se prepara la instrumentación y se evalúan las amenazas al experimento. La etapa de planificación se puede dividir en varios pasos, que pueden ejecutarse en forma iterativa hasta alcanzar el resultado final de esta etapa:

Selección del contexto. Desde este punto de vista se tiene en cuenta cuatro dimensiones que definen la generalidad del mismo:

- El experimento se llevó a cabo "off-line", es decir, en un entorno de laboratorio.
- Los sujetos fueron estudiantes o egresados recientes.
- El experimento abordó un problema real.
- El experimento fue válido para un contexto específico.

Formulación de las hipótesis. Este paso consiste en transformar la idea concebida para el experimento en una frase formal, es decir en hipótesis.

A continuación se enuncian hipótesis para cada variable dependiente.

a. Hipótesis de la Eficiencia

<u>En cuanto a la técnica</u>:

- H0,a: La Eficiencia de la técnica PE es igual a la Eficiencia de la técnica de TD
- H1,a: La Eficiencia de la técnica PE no es igual a la Eficiencia de la técnica de TD

<u>En cuanto al problema</u>:

- H0,a: La Eficiencia de la técnica aplicando el problema del triángulo es igual a la Eficiencia de la técnica aplicando el problema de nextdate
- H1,a: La Eficiencia de la técnica aplicando el problema del triángulo no es igual a la Eficiencia de la técnica aplicando el problema de nextdate

b. Hipótesis de la Eficacia

<u>En cuanto a la técnica</u>:

- H0,b: La Eficacia de la técnica PE es igual a la Eficacia de la técnica de TD
- H1,b: La Eficacia de la técnica PE no es igual a la Eficacia de la técnica de TD

En cuanto al problema:

- H0,b: La Eficacia de la técnica aplicando el problema del triángulo es igual a la Eficacia de la técnica aplicando el problema de nextdate
- H1,b: La Eficacia de la técnica aplicando el problema del triángulo no es igual a la Eficacia de la técnica aplicando el problema de nextdate

c. Hipótesis de la facilidad de aplicación

- H0,e: La Técnica "Partición Equivalente" es mas fácil de aplicar que la Técnica "Tablas de Decisión"
- H1,e: La Técnica "Partición Equivalente" no es mas fácil de aplicar que la Técnica "Tablas de Decisión"

d. Hipótesis de la intención de uso

- H0,d: La Técnica PE logra una mayor intención de uso por parte de los alumnos que la TD
- H1,d: La Técnica PE no logra una mayor intención de uso por parte de los alumnos que la TD

Selección de las variables. se dividieron en independientes, es decir que no dependen del fenómeno estudiado y pueden tener un impacto en el resultado; y dependientes, dependen del fenómeno estudiado. Descriptas en la tabla I:

TABLA I: VARIABLES SELECCIONADAS PARA EL EXPERIMENTO

Independientes	Dependientes
Las técnicas de pruebas funcionales	Eficiencia
Los problemas de codificación	Eficacia
	Intención de uso
	Facilidad de aplicación

Las variables independientes serán los factores a considerar. Las técnicas de pruebas funcionales, toman dos valores posibles (tratamientos), "Partición Equivalente" y "Tablas de Decisión". Para los problemas de codificación, los valores posibles (tratamientos) serán, "Triángulo" y "NextDate". Por otro lado, se distinguen las siguientes variables Variables Dependientes:

- Eficiencia: definida como la capacidad para llevar a cabo un trabajo o una tarea con el mínimo consumo de tiempo. Están relacionados con las variables que nos indican el tiempo invertido en la obtención de la cantidad de caso de test aplicada a una técnica a un problema determinado.
- Eficacia: definida como la capacidad de detectar errores aplicando una técnica a un problema determinado.
- Intención de uso: definida como el grado de preferencia que tiene una persona para utilizar una técnica de testing en el futuro.
- Facilidad de Aplicación: definida como el grado en el que una persona asimila y utiliza una técnica de testing.

Selección de sujetos. Se seleccionaron 19 estudiantes avanzados de la carrera Licenciatura en Sistemas de Información, de la Facultad de Ciencias Exactas, Físicas y Naturales, dependiente de la Universidad Nacional de San Juan (Argentina).

Diseño del experimento. El tipo de diseño que se utilizó para realizar el experimento fue el de "Dos factores con dos tratamientos" y dentro de este el "Anidado".

El primer factor, A, representan las técnicas de pruebas de software y el segundo factor, B, constituyen los problemas de codificación. La tabla II muestra el diseño experimental:

TABLA II: DISEÑO DEL EXPERIMENTO

Factor A = Técnicas de Prueba de Software			
Tratamiento A1 = Partición Equivalente		*Tratamiento A2 = Tablas de Decisión*	
Factor B = Problemas de Codificación		***Factor B = Problemas de Codificación***	
Tratamiento B1=Triángulo	Tratamiento B2 =NextDate	Tratamiento B1=Triángulo	Tratamiento B2= NextDate
Sujeto 1, 5	Sujeto 2, 6	Sujeto 3, 7	Sujeto 4, 8

La asignación de cada sujeto a cada grupo para aplicar una técnica de prueba de software a un problema específico, a fin de obtener los casos de prueba, fue realizada en forma aleatoria mediante sorteo, a fin de asegurar la validez de los resultados.

EJECUCIÓN DEL EXPERIMENTO

En esta sección se abordará la operación del experimento. Se llevó a cabo bajo la modalidad de taller, con una carga horaria de 7 horas, distribuidas en dos días.

El taller, se dividió en dos partes, una teórica y otra práctica. En la primera, se les explicó a los participantes en qué consistían las dos técnicas de testing: *Partición Equivalente* y *Tabla de Decisión*, con algunos ejemplos de aplicación. Luego se distribuyó en forma aleatoria a los sujetos en cada grupo y se les proporcionó documentación teórica sobre las técnicas de testing, enunciados y planillas, para que los distintos grupos realizaran la práctica. La que consistía en analizar y registrar en planillas prediseñadas, los casos de test de las especificaciones funcionales asignadas [18]. Una vez terminado este proceso, los estudiantes asistieron al laboratorio de computación para validar los casos de test hallados y comprobar redundancias o la ausencia de casos de test, en su análisis.

Teniendo en cuenta la cantidad total de sujetos, y el diseño del experimento elegido (división en cuatro grupos). Se distribuyeron los sujetos como se indica en la Tabla III:

TABLA III

DISTRIBUCIÓN DE SUJETOS POR GRUPO

Grupos	Tratamientos	Cantidad de Sujetos
I	A1B1	6
II	A1B2	4
III	A2B1	5
IV	A2B2	4

Al finalizar la validación de los casos de test, se les proporcionó un cuestionario, con el objetivo de conocer su

nivel de satisfacción en cuanto a las variables: facilidad de aplicación e intención de uso respecto a la técnica empleada. La recolección de datos fue manual. Se realizó a través de planillas, cuestionarios; y mediante el registro de las observaciones presenciales, llevadas a cabo por los profesores durante el taller.

ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS

En esta sección se analizó como las variables independientes influyen en las dependientes (ver tabla I). Respecto al análisis de los datos se necesitó realizar un estudio de la distribución de los datos y la docimasia de hipótesis utilizando el software estadístico Infostat / Profesional versión 2007p©. En la tabla IV se muestran las medidas obtenidas para las variables Eficiencia y Eficacia:

TABLA IV

ESTADÍSTICA DESCRIPTIVA DE LAS VARIABLES EFICIENCIA Y EFICACIA

Variable	Técnica	n	Media	D.E.	Var	Mín	Máx	Q1	Q3
Eficiencia	PE	10	0,34	0,16	0,03	0,08	0,67	0,23	0,43
	TD	9	0,27	0,13	0,02	0,11	0,44	0,14	0,36
Eficacia	PE	10	0,72	0,26	0,07	0,17	1,00	0,67	1,00
	TD	9	0,54	0,18	0,03	0,17	0,67	0,50	0,67

Se puede deducir que la técnica TD tiene mayor concentración de datos por debajo de 0,38, siendo la técnica PE más eficiente.

En cuanto a la posibilidad de aplicar un test paramétrico. Se analizaron los supuestos del modelo y se determinó que era posible aplicar ANOVA, por que la distribución de las variables Eficiencia y Eficacia correspondía a una distribución normal (de acuerdo al test de Shapiro –Wilks). Se aplicó el análisis de la Varianza (ANOVA) para determinar si las variaciones o fluctuaciones que se presentan respecto a las variables Eficiencia y Eficacia (variables dependientes) se deben a la técnica Partición Equivalente o Tabla de Decisión (PE o TD) y/o al problema tratado (Triángulo o NextDate) (uno o ambos factores). En la tabla V se presentan los valores obtenidos:

TABLA V

CUADRO DE ANÁLISIS DE LA VARIANZA (SC TIPO III) PARA LAS VARIABLES: EFICIENCIA Y EFICACIA

	Eficiencia					Eficacia				
F.V.	**SC**	**gl**	**CM**	**F**	**P valor**	**SC**	**gl**	**CM**	**F**	**p-Valor**
Modelo	0,20	2	0,10	8,20	0,0035	0,38	2	0,19	4,61	0,0263
Técnica	0,02	1	0,02	1,32	0,2679	0,14	1	0,14	3,32	0,0871
Problema	0,18	1	0,18	14,64	0,0015	0,22	1	0,22	5,49	0,0324
Error	0,20	16	0,01			0,65	16	0,04		
Total	0,40	18				1,03	18			

En cuanto a la técnica, el p-valor es mucho mayor al nivel de significancia que consideramos (α=0,10 – un nivel del 90% de confianza), no hay evidencia de diferencias significativas de la Eficiencia según las técnicas empleadas, por lo que no existe evidencia para rechazar la hipótesis nula H0a (μ PE = μ TD). En cuanto al problema, el p-valor es menor al nivel de significancia que consideramos (α=0,10 un nivel del 90% de confianza), se rechaza H0a (μ NextDate = μ Triángulo). Se puede afirmar que, la eficiencia no es significativamente diferente respecto a las técnicas usadas, mientras que la eficiencia sí es significativamente diferente respecto de los problemas (enunciados) utilizados.

De la gráfica de caja y bigotes (figura 1), se observa que la eficiencia en la técnica PE se conserva mayor a la técnica TD, además la mediana de la técnica PE es mayor a la mediana de la técnica TD.

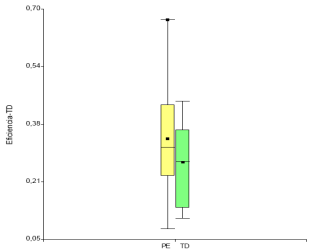

FIGURA 1

EFICIENCIA DE LAS TÉCNICAS PE Y TD

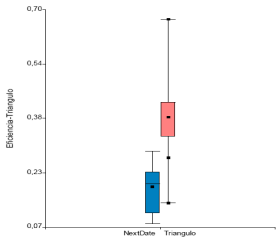

FIGURA 2

EFICICIENCIA DE LOS ENUNCIADOS NEXTDATE Y TRIÁNGULO

De la gráfica de caja y bigotes (figura 2), se puede ver que la eficiencia con el problema del triángulo se conserva mayor que con el problema de NextDate, además la mediana del problema del triángulo es mayor a la mediana del problema de NextDate.

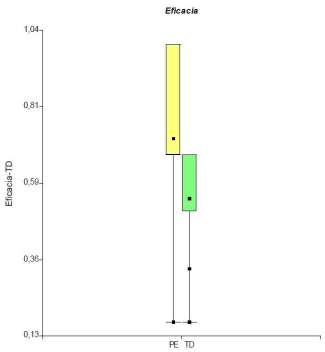

FIGURA 3

EFICACIA DE LAS TÉCNICAS PE Y TD

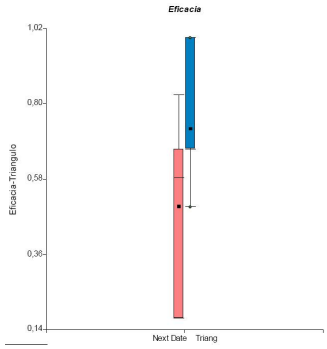

FIGURA 4

EFICACIA DE LOS ENUNCIADOS NEXTDATE Y TRIÁNGULO

De la gráfica de caja y bigotes (figura 3) se puedes ver que la eficacia en la técnica PE se conserva mayor a la técnica TD, además la mediana de la técnica PE es mayor a la mediana de la técnica TD. Y de la figura 4 se observa que la eficacia con el problema del triángulo se conserva mayor que con el problema de NextDate, además la mediana del problema del triángulo es mayor a la mediana del problema de NextDate

Variables Facilidad de aplicación e Intención de uso

Se administró un cuestionario a los alumnos participantes, a fin de conocer sus opiniones sobre las variables cualitativas "Facilidad de aplicación" e "Intención de uso". Los resultados obtenidos se presentan graficamente en figuras 5 y 6.

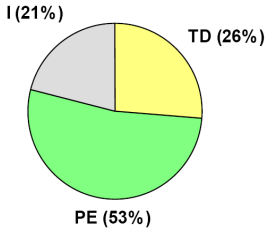

FIGURA 5

GRÁFICO DE TORTA RESPECTO A LA VARIABLE INTENCIÓN DE USO

Se puede observar en la Figura 5, que más del 50% de los sujetos elegirían la técnica de Partición Equivalente para uso futuro como profesionales informáticos, por su facilidad de comprensión y su forma práctica de detectar los casos de prueba.

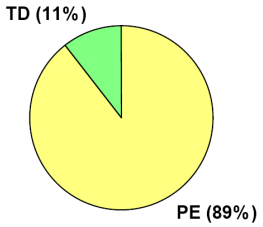

FIGURA 6

GRÁFICO DE TORTA RESPECTO A LA VARIABLE FACILIDAD DE APLICACIÓN

Se puede observar en figura 6, que un alto porcentaje, cercano al 90%, opina que le resultó más fácil de aplicar la técnica de Partición Equivalente que la técnica de Tablas de Decisión.

CONCLUSIONES Y TRABAJOS FUTUROS

En este trabajo se presentó la descripción del proceso para diseñar e implementar un experimento controlado que tiene como objetivo comparar dos técnicas de Testing de Software.

Mediante la observación de los datos estadísticos, se puede afirmar que, la eficiencia no es significativamente diferente respecto a las técnicas usadas, mientras que sí lo es significativamente diferente respecto de los problemas (enunciados) utilizados. En cuanto a la eficacia, La técnica de Partición Equivalente fue mayor a la técnica de Tablas de Decisión.

Se puede observar que, la mayoría de los sujetos elegirían la técnica de Partición Equivalente para uso futuro como profesionales informáticos, por su facilidad de comprensión y su forma práctica de detectar los casos de prueba, lo que implica un menor costo en tiempo en la obtención en los casos de prueba.

Por otra parte, si bien, los participantes seleccionados fueron alumnos avanzados, en lugar de profesionales en el área de Ingeniería de Software, existen pruebas, de que los resultados obtenidos, son de gran utilidad para obtener evidencias importantes que luego puedan ser trasladadas a la industria de desarrollo de software. Además, se cree necesario replicar este experimento, en empresas del medio, o en otras universidades, a fin de mejorar los resultados precisando las conclusiones e incrementando de esta manera el conocimiento científico en el área de la ingeniería de software. Así como también, generar herramientas computacionales que permitan agilizar la implementación de las técnicas analizadas.

REFERENCIAS

[1] Ahlowalia, Naresh. 2002. Testing From Use Cases Using Path Analysis Technique. International Conference On Software Testing Analysis & Review

[2] Baresi, L., Morasca, S. and Paolini, P. 2003.Estimating the Design Effort of web Applications. Proceedings of the 9th International Software Metrics Symposium (METRICS'03), pp. 62-71

[3] Basili V. 2006.The Past, Present, and Future of Experimental Software Engineering. Journal of the Brazilian Computer Society (JBCS).

[4] Basili, V., Rombach, H. 1988. The TAME Project: Towards Improvement-Oriented

[5] CMMI Product Team, 2002. CMISM for Systems Engineering/Software Engineering, Version 1.1, Staged Representation. Technical report, (SEI)CMU/SEI-2002-TR-002, Software Engineering Institute. http://www.sei.cmu.edu/cmmi/

[6] Fagan, M. 1986."Advances in Software Inspections." IEEE Transactions on Software Engineering, SE-12, 7, (Julio 1986).

[7] Ferriday, C., 2007. A Review Paper on Decision Table-Based Testing. http://www.cs.swan.ac.uk/~csmarkus/CS339/dissertations/FerridayC.pdf

[8] García Romero, C.I., 2001. El Modelo de Capacidad de Madurez y su Aplicación en Empresas Mexicana de Software. Tesis Licenciatura. Ingeniería en Sistemas Computacionales. Departamento de Ingeniería en Sistemas Computacionales, escuela de Ingeniería, universidad de las Américas- Puebla. Mayo.

[9] Höst. M..2002. Introducing Empirical Software Engineering Methods in Education. Proceedings of the 15th Conference in Software Education and Training (CSEET'02), pp. 170-179.

[10] IEEE Standard Glossary of Software Engineering Terminology, IEEE Std. 610-1990 , IEEE Standards Software Engineering, Volume 1; The Institute of Electrical and Electronics Engineers, 1999; ISBN 0-7381-1559-2.

[11] Jorgensen, Paul, 2002. Software testing: A Craftsman's Approach, CRC Press, Second Edition..

[12] Mosley D., 1993. The Handbook of MIS Application Software Testing. Methods, Techniques, and Tools for Assuring Quality Through Testing. New Jersey: Prentice Hall. 352 P.

[13] Myers G., 2004. The Art of Software Testing. New Jersey: Wiley John and Sons. 256 P.

[14] Myers, Glenford J. 2004.The art of Software Testing: Second Edition, John Wiley & Sons, Inc.

[15] Pressman, Roger, 2004. Ingeniería del Software. Un Enfoque Práctico. Sexta edición Prentice-Hall

[16] Software Environments. IEEE Transactions on Software Engineering 14 , 758-773.

[17] Sommerville, I.2005. Ingeniería del Software. Séptima edición.. Pearson Educación, S.A., Madrid. ISBN:84-7829-074-5. 712 P

[18] Soto, A. A. y Sánchez, P. E..2009. Ingeniería de Software Experimental: Técnicas de Pruebas de Software (Testing) Funcional. Tesis de Licenciatura en Sistemas de Información. Facultad de Ciencias Exactas, Físicas y Naturales, Universidad Nacional de San Juan.

[19] Wohlin C.; Runeson P.; Höst M.; Ohlsson M.; Regnell B. y Wesslén A, 2000. experimentation in Software Engineering: An Introduction. Springer. 228 P.