ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN

"ANÁLISIS DE MÉTODOS, TÉCNICAS Y HERRAMIENTAS DE VERIFICACIÓN Y VALIDACIÓN DE SOFTWARE USADOS POR EMPRESAS ECUATORIANAS DESARROLLADORAS DE SOFTWARE"

TESIS DE GRADO

Previa a la obtención del Título de:

INGENIERO EN COMPUTACIÓN
Especialización: Sistemas de Información

Presentado por:

FREDDY GONZALO TITUANA VERA

Directora:

ING. VERÓNICA MACÍAS MENDOZA

GUAYAQUIL – ECUADOR

2009

DEDICATORIA

Primeramente a Dios por permitirme lograr este anhelo y por siempre darme la fuerza para no desmayar en mis metas, a mis padres que desde pequeño me supieron inculcar valores y que a pesar de la distancia siempre conté con su apoyo incondicional, a mi hermano quien siempre estuvo a mi lado y sin su ayuda no hubiera sido posible realizar esta tesis.

Gracias…

FREDDY TITUANA VERA

AGRADECIMIENTO

A la Escuela Superior Politécnica del Litoral por haberme formado como profesional y brindado la suficiente base teórica para poder incursionar en un medio altamente competitivo.

A la Ing. Verónica Macías, Directora de Tesis por toda su ayuda, paciencia y entrega en la realización de este proyecto, además agradecer por el apoyo y libertad que me ha dado para desarrollar este estudio.

Al componente 8 de Ingeniería de Software que nos facilito sus instalaciones para la realización de este trabajo.

Al Centro de Investigación Científica y Tecnológica de la ESPOL por haber financiado este estudio.

A los docentes, Ingeniera Carmen Vaca e Ingeniera Mónica Villavicencio, quienes revisaron esta tesis y aportaron con sus conocimientos de ingeniería de software, brindando invalorables recomendaciones.

A los especialistas de REVVIS que contribuyeron con la revisión de la encuesta proporcionado valiosas observaciones

A todos mis amigos y compañeros que a lo largo de mi carrera universitaria me brindaron su ayuda.

TRIBUNAL DE GRADUACIÓN

....	....
Ing. Jorge Aragundi	Ing. Verónica Macías
SUB DECANO DE LA FIEC	DIRECTOR DE TESIS
....	....
Ing. Carmen Vaca	Ing. Mónica Villavicencio
MIEMBRO DEL TRIBUNAL	MIEMBRO DEL TRIBUNAL

DECLARACIÓN EXPRESA

"La responsabilidad por los hechos, ideas y doctrinas expuestos en esta tesis, me corresponden exclusivamente; y el patrimonio intelectual de la misma, a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL"

(Reglamento de Exámenes y Títulos profesionales de la ESPOL)

....

FREDDY GONZALO TITUANA VERA

RESUMEN EJECUTIVO

El presente trabajo de tesis fue orientado a estudiar como se encuentran actualmente las empresas ecuatorianas desarrolladoras referente al conocimiento y uso de Metodologías, técnicas y Herramientas de V&V del Software. Como resultado del estudio se proporciona un listado de Métodos, Técnicas y Herramientas de V&V de software para que las empresas cuenten con una base de conocimiento que puedan aplicarlo en sus respectivos proyectos. Para el efecto se han desarrollado cuatro capítulos en donde se explica el proceso para cumplir lo propuesto.

Capítulo I

El capítulo uno refiere a la recopilación de antecedentes y justificación de la Tesis, tiene como propósito describir la situación actual de la industria del software ecuatoriano, además de estudios realizados anteriormente tanto en Ecuador como en países tales como: Argentina y Brasil. Finalmente se expone la importancia del estudio realizado.

Capítulo II

El capítulo dos denominado Marco Teórico hace una recopilación de la literatura obtenida de muchos textos revisada para la elaboración del presente

estudio, como temas de Manipulación de datos para realizar análisis estadístico y los principales métodos, técnicas y herramientas de V&V de software.

Capítulo III

En el capítulo tres, también denominado diseño del estudio, se formulan las hipótesis, se plantean los objetivos, se explica la metodología utilizada para recopilar información, se presenta la población objetivo y las razones de selección de la muestra. Finalmente se detalla y explica el plan seguido para el levantamiento de información.

Capítulo IV

El capítulo cuatro denominado análisis y presentación de resultados como su nombre lo indica se encarga de mostrar los resultados obtenidos luego de la aplicación de la encuesta y su análisis; posteriormente se verificará las hipótesis planteadas en capítulo tres; y, finalmente se realiza un listado de las Herramientas, Métodos y Técnicas mas idóneas para las empresas Ecuatorianas en base a los resultados obtenidos.

ÍNDICE GENERAL

RESUMEN EJECUTIVO I

ÍNDICE GENERAL.. III

ABREVIATURAS.. V

INDICE DE FIGURAS Y GRÁFICOS VII

INDICE DE TABLAS...IX

INTRODUCCIÓN X

CAPÍTULO I 1
1. ANTECEDENTES Y JUSTIFICACIÓN DE LA TESIS 1
1.1. Industria del Software en América Latina 1
1.1.1. La Industria del Software en Brasil 1
1.1.2. La Industria del Software en Argentina....................................... 5
1.1.3. La Industria del Software en Ecuador.. 11
1.2. Objetivos.. 14
1.3. Justificación de la Tesis 15

CAPÍTULO II 16
2. MARCO TEORICO.. 16
2.1. Proceso de Desarrollo de Software 16

2.2 Verificación de Software 22
2.2.1 Conceptos básicos de Verificación... 22
2.2.2 Verificación Estática 22
2.2.3 Verificación Dinámica 22

2.3 Validación de Software 23
2.3.1 Conceptos básicos de Validación.. 23
2.3.2 Validación simple.. 23
2.3.3 Validación cruzada 23

2.4 Herramientas de Distribución Libre para Verificación y Validación de Software. 23
2.5 Herramientas de Distribución Comercial para Verificación y Validación de Software. 30
2.6 Técnicas de Verificación y Validación de Software......................... 35
2.7 Métodos de Verificación y Validación de Software. 40
2.8 Proceso de Pruebas 43
2.8.1 Niveles de Pruebas 43
2.8.2 Métodos y Tipos de Pruebas... 44

2.9 Herramientas de interpretación de datos... 45

2.9.1 Diagramas de Dispersión. 45
2.9.2 Diagramas causa-efecto.. 46
2.9.3 Histogramas. 47
2.10 Análisis Univariado. 48
2.11 Análisis Multivariado... 48
CAPÍTULO III 49
3. DISEÑO DEL ESTUDIO.. 49
3.1 Hipótesis.. 49
3.2 Metodología para la recolección de la información 49
3.3 Población objetivo 50
3.4 Instrumento de medición 53
3.5 Plan de levantamiento de información.. 55
CAPÍTULO IV... 57
4 ANÁLISIS Y PRESENTACIÓN DE RESULTADOS DEL ESTUDIO ... 57
4.1 Presentación de estadísticas.. 57
4.2 Pruebas de verificación de hipótesis 74
4.3 Análisis de indicadores obtenidos 80
4.4 Listado de herramientas, métodos y técnicas de Verificación y Validación de Software adecuadas para el uso de las empresas ecuatorianas desarrolladoras de software.. 84
CONCLUSIONES 89
RECOMENDACIONES 91
BIBLIOGRAFÍA 93
ANEXO 1: Encuesta Piloto aplicada al Estudio 99
ANEXO 2: Encuesta Final aplicada al Estudio....................................... 100
ANEXO 3: Software SPSS.. 101
ANEXO 4: Análisis de la Encuesta.. 102
ANEXO 5: Carta para las Empresas.. 103

ABREVIATURAS

Las abreviaturas presentadas en la siguiente tesis son las siguientes:

V&V: Verificación y Validación de Software

AESOFT: Asociación Ecuatoriana de Software.

VLIR: Vlaamse Interuniversitaire Raad

REVVIS: Reunión de especialistas en Verificación y Validación de Software

CMM: Capability Maturity Model (Modelo de Capacidad y Madurez).

CMMI: Capability Maturity Model Integration (Modelo de Capacidad y Madurez Integrado).

CORPEI: Corporación de Promoción de Exportación e Inversiones del Ecuador.

IEEE: The Institute of Electrical and Electronics Engineers, (Instituto de Ingenieros Eléctricos y Electrónicos).

ISO: International Organization for Standardization (Organización Internacional para la Estandarización).

MPYMES: Micro, pequeñas y medianas empresas.

MSF: Microsoft Solution Framework.

PIB: Producto Interno Bruto.

CASE: Computer Aided Software Engineering (Ingeniería de Software Asistida por Ordenador)

PROSOFT: Programa para el Desarrollo de la Industria del Software.

PSP: Personal Software Process (Proceso personal del software)

SEI: Software Engineering Institute.

TI: Tecnologías de la Información.

CVS: Ciclo de Vida del Software

INDICE DE FIGURAS Y GRÁFICOS

Figura 1.1.2.1 Distribución de empresas según principal servicio provisto [6] 9

Figura 1.1.2.2 Distribución de empresas según su ubicación geografica[6] 10

Figura 1.1.3.2 Distribución de empresas por ciudades [8] 12

Figura 2.1.1 Proceso de desarrollo de software. [9] 17

Figura 2.1.2 Modelo de desarrollo en Cascada 18

Figura 2.1.3 Modelo de desarrollo Evolutivo 19

Figura 2.1.4 Paradigma de programación Automática 19

Figura 2.1.5 Desarrollo basado en reutilización de componentes 20

Figura 2.1.6 Modelo de desarrollo iterativo incremental [10] 20

Figura 2.1.7 Modelo de desarrollo en Espiral [11] 21

Figura 2.9.1. Ejemplo de un diagrama de dispersión [56] 46

Figura 2.9.2 Ejemplo de un diagrama causa-efecto 46

Figura 2.9.3. Ejemplo de un histograma 47

Figura 3.3.3 Porcentaje de empresas encuestadas por ciudad 53

Figura 4.1.1. Estadísticas de la pregunta 1 de la encuesta 58

Figura 4.1.2. Estadísticas de la pregunta 2 sección III de la encuesta 59

Figura 4.1.3. Estadísticas de la pregunta 3 sección III de la encuesta 60

Figura 4.1.4. Estadísticas de la pregunta 4 sección III de la encuesta 61

Figura 4.1.5. Estadísticas de la pregunta 1 sección IV de la encuesta 62

Figura 4.1.6. Estadísticas de la pregunta 2 sección IV de la encuesta 63

Figura 4.1.7 Estadísticas de la pregunta 3 sección IV de la encuesta 63

Figura 4.1.8. Estadísticas de la pregunta 4 sección IV de la encuesta 64

Figura 4.1.9 Estadísticas de la pregunta 5 sección IV de la encuesta 65

Figura 4.1.10. Estadísticas de la pregunta 6 sección IV de la encuesta 66

Figura 4.1.11. Estadísticas de la pregunta 1 sección V de la encuesta..... 67

Figura 4.1.12. Estadísticas de la pregunta 2 sección V de la encuesta..... 68

Figura 4.1.13. Estadísticas de la pregunta 3 sección V de la encuesta..... 69

Figura 4.1.14. Estadísticas de la pregunta 4 sección V de la encuesta..... 70

Figura 4.1.15. Estadísticas de la pregunta 5 sección V de la encuesta..... 72

Figura 4.1.16. Estadísticas de la pregunta 6 sección V de la encuesta..... 73

Figura 4.4.1. Porcentaje de Técnicas de V&V que aportan mas calidad a empresas Ecuatorianas 85

Figura 4.4.2 Porcentaje de Métodos de V&V que aportan más calidad a empresas Ecuatorianas 86

Figura 4.4.3 Porcentaje de Herramientas de Distribución Libre de V&V que aportan más calidad a empresas Ecuatorianas... 87

Figura 4.4.3 Porcentaje de Herramientas Propietarias de V&V que aportan más calidad a empresas Ecuatorianas.. 88

INDICE DE TABLAS

Tabla 1.1.1.1 Mercado de la Tecnología de la Información en los países de América Latina [5] 2

Tabla 1.1.1.2. Ranking de las mayores empresas de Tecnología Brasileras [5] 3

Tabla 3.3.1 Detalles de la indagación en Guayaquil 51

Tabla 3.3.2 Detalles de la indagación en Quito 52

Tabla 4.2.1 Tabla de resultados de Hipótesis 1 75

Tabla 4.2.2 Tabla de resultados de Hipótesis 2 76

Tabla 4.2.3 Tabla de resultados de Hipótesis 3 77

Tabla 4.2.4 Tabla de resultados de Hipótesis 4 78

Tabla 4.2.5 Tabla de resultados de Hipótesis 5 80

INTRODUCCIÓN

Con el crecimiento de los sistemas de software, que han evolucionado hasta alcanzar tamaños y complejidades extraordinariamente altos, su inestabilidad se está convirtiendo, lamentablemente, en algo cotidiano y asumido con resignación por los usuarios en una sociedad de la información cada vez más exigente [2]. Para garantizar esta fiabilidad son necesarias técnicas, métodos y herramientas que soporten de forma adecuada el proceso en todas las etapas del desarrollo del software.

La Ingeniería de Software es el área de la ciencia encargada de todos los aspectos de la producción de aplicaciones de software; incluyendo estos la definición y aplicación de técnicas, métodos y herramientas en cada una de las fases de producción. El objetivo principal de un Ingeniero de software es generar un producto de calidad, dentro del tiempo y del presupuesto establecido. Existen muchos factores dentro de la elaboración de un sistema que pueden afectar a la calidad del producto; y uno de estos factores es la verificación y validación del software (V&V por sus siglas en inglés).

La V&V del software es un proceso de chequeo y análisis para asegurarse de que la aplicación cumple con las especificaciones funcionales establecidas y que estas son en efecto las deseadas por los clientes del sistema. Existen varios métodos, técnicas y herramientas que nos permiten llevar a cabo el proceso de V&V.

El aplicar adecuadamente las técnicas, métodos y herramientas de V&V tiene un alcance mucho mayor que el área inmediata de desarrollo de los proyectos, porque su efecto se repercute al resto de la empresa, áreas de negocio y cliente final. Así, mientras el departamento de Sistemas cumple su compromiso de plazo, coste y calidad, se facilita la transición de Desarrollo a Producción, ya que los proyectos pasan al entorno de Producción con estabilidad, de esta manera se elimina la "etapa" de post-implantación, evitando el efecto cascada en el correctivo, y ofreciendo una mejor respuesta a los aspectos técnicos relevantes del ciclo de vida en Producción. Todo ello hace que se incremente la confianza y el nivel de satisfacción de los usuarios finales [3].

Actualmente en nuestro país se ha hecho necesario que conozcamos y comprendamos los factores que afectan la calidad del software (tal como la V&V del software) ya que en los últimos años se ha registrado el nacimiento de un nuevo producto de exportación no tradicional en nuestro país que es el software [1]; y es importante que, para asegurar el crecimiento de su comercialización, las empresas ecuatorianas desarrolladoras de software entreguen un producto con buena calidad.

Antes de iniciar este trabajo no existía un estudio que nos indique en que porcentaje las empresas desarrolladoras de software conocen y emplean herramientas, técnicas o métodos de V&V para el desarrollo de un producto de Software; y dado que no se puede mejorar lo que no se conoce, quisimos aportar con este estudio para que sirva como base para futuros proyectos.

CAPÍTULO I

1. ANTECEDENTES Y JUSTIFICACIÓN DE LA TESIS

1.1. Industria del Software en América Latina

Esta sección está enfocada en hacer una comparación de la Industria de Software entre países como Brasil, Argentina y Ecuador.

1.1.1. La Industria del Software en Brasil

Estudios realizados en la industria de software en Brasil en los últimos años se basan en análisis de las empresas dedicadas al desarrollo de Software. Nos enfocaremos en los estudios realizados en el año 2004 por Luis Alberto Herrera titulado “Marco Normativo en el Sector Software” y el estudio realizado en el año 2006 por Paulo Bastos Tigre denominado La industria del Software en Brasil. [59]

La industria de software factura en Brasil alrededor de U$S 4.2 billones por año, representando 0.7% del PBI, lo que convierte a Brasil en el 7º mercado del mundo. Sin embargo, sus exportaciones apenas sobrepasan los 100 millones de dólares. [5]

El actual gobierno, a pesar de no tener una política global de promoción del sector, se ha planteado como meta para el final de su mandato, alcanzar los 2 billones de dólares anuales en exportaciones de software.

Dicha política establece que, para lograr una mayor inserción internacional, se debe estimular sectores en los cuales Brasil tiene mayor capacidad de desarrollar ventajas competitivas. De esta forma se establecen cuatro sectores estratégicos, siendo uno de ellos el sector software.

Tabla 1.1.1.1 Mercado de la Tecnología de la Información en los países de América Latina [5]

País	Hardware	Software	Servicios	Gastos internos	Total
Brasil	49%	52%	51%	40%	45%
México	18%	17%	18%	26%	21%
Argentina	10%	11%	10%	7%	10%
Colombia	4%	5%	4%	9%	5%
Venezuela	4%	5%	6%	9%	5%
Chile	3%	3%	4%	5%	4%
Resto	12%	7%	6%	4%	9%

El principal instrumento que plantea esta política para el desarrollo del sector es el **Nuevo Prosoft** (Programa para el Desarrollo de la Industria Nacional de Software y Servicios Relacionados). La meta de este programa es promover el crecimiento de las empresas brasileñas y aumentar su participación en el mercado interno y externo.

Prosoft se subdivide en 3 subprogramas: Prosoft- Empresa (financiamiento para inversiones y planes de negocios de empresas brasileñas productoras de software y servicios relacionados), Prosoft-Comercialización (financiamiento para adquisición, en el mercado interno, de software y servicios relacionados desarrollados en Brasil) y Prosoft- Exportación (financiamiento a la exportación de software y servicios relacionados desarrollados en Brasil).

.

Por otra parte Brasil tiene un grupo de empresas muy importantes y con una significativa participación de capital extranjero, convirtiéndolo en uno de los países mas desarrollados en lo referente a la industria de Software.

Tabla 1.1.1.2. Ranking de las mayores empresas de Tecnología Brasileras [5]

Empresas	Ventas (millones de dólares)	Origen
Microsoft	362	Estados Unidos
SERPRO	372	Brasil
Computer Associates	260	Estados Unidos
EDS	240	Estados Unidos
CPM	204	Brasil
Accenture	194	Estados Unidos
Oracle Brazil	182	Estados Unidos
SAP Brazil	124	Alemania
Politec	104	Brasil
Consist	77	Estados Unidos
Microsiga	72	Brasil
CPqD	64	Brasil
DBA	62	Brasil
CITS	57	Brasil
Proceda	52	Brasil

Además en Brasil, el desarrollo del sector de tecnologías de la información y comunicación (TIC) fue estimulado tanto por la demanda interna como por la puesta en marcha de políticas públicas dirigidas al desarrollo de las actividades de tecnologías de información y comunicación. La demanda interna por TI permitió el desarrollo de la capacitación en actividades de alto valor agregado, sobre todo en aplicaciones verticales en las áreas de servicios financieros, gobierno, telecomunicaciones y automatización comercial. La articulación de la relación usuario-proveedor permitió avances en segmentos de mayor complejidad tecnológica y que exigen conocimientos tácitos de la cadena productiva de software.

Tal capacitación por lo general es identificada como un potencial por explotar en el proceso de internacionalización de la industria brasileña. Los segmentos que se destacan como mayores demandantes de software y servicios de información (SSI) en Brasil son: Consumidores residenciales, sector público, servicios financieros y empresariales e industria de la transformación.

Los consumidores residenciales fueron responsables, en 2005, de casi un cuarto del gasto total de TIC. El consumo doméstico esta siendo estimulado por programas de inclusión digital del gobierno federal, en especial el de PC Conectada, que estableció exención de impuestos y abrió líneas de crédito para la compra de computadoras. Otro aliado de los consumidores fue la fuerte devaluación del dólar frente al real observada en los últimos años, que volvió los precios del hardware más accesibles al nivel de ingresos de la población.

En la industria de la transformación también se observa un crecimiento acelerado en el uso de TIC, sobre todo en función del desarrollo de programas de computadora de apoyo gerencial (ERP) para pequeñas y medianas empresas. Los otros dos usuarios importantes, bancos y gobierno, se analizarán con más detalle debido a su contribución estructural al crecimiento de la economía de la información en el país.

En el sector Público, gobierno federal y algunas instancias administrativas constituyen un gran y complejo mercado para programas de computadora y servicios.

Algunos ejemplos de éxito en el desarrollo local son: el sistema de Pagos Brasileño (SPB), que permite la liquidación en tiempo real de transacciones financieras; la automatización de la declaración del impuesto sobre la renta; el sistema de urnas electrónicas; que permite obtener el resultado de la elecciones en menos de 24 horas, y sistemas de compras gubernamentales, como el Comprasnet del gobierno federal [59].

1.1.2. La Industria del Software en Argentina

Se han realizado muchos estudios sobre la industria de software en Argentina en los últimos años, muchos de estos se basan en encuestas dirigidas hacia el conjunto del sector de Software y Servicios Informáticos. Nos enfocaremos en el estudio realizado en el año 2005 por la Cámara de Empresas de Software & Servicios Informáticos, titulado "Situacion Actual y desafíos futuros de las PYMES de Software y Servicios Informáticos". [6]

Refiriéndose en éste estudio, señala que las principales características de este sector en Argentina son: Recursos humanos muy bien calificados, Innovación y capacidad creativa, Infraestructura de telecomunicaciones e informática adecuada. [6]

Se trata de una industria que está fuertemente concentrada en la Ciudad de Buenos Aires, pero existen también importantes conglomerados de empresas en el Conurbano Bonaerense, Rosario, Córdoba, Mendoza, Tandil, Mar del Plata y Bahía Blanca, entre otros. Estos polos tienen en común la disponibilidad de recursos humanos calificados, la relevancia de la Universidad como "semillero" de emprendedores en el área tecnológica y una fuerte vocación de cooperación empresarial. [6]

Este sector está compuesto básicamente por:

- Un número reducido de grandes empresas, la mayoría de capitales extranjeros, dedicadas principalmente a la comercialización de productos de otros países y a la prestación de servicios informáticos. En este grupo, que incluye prácticamente a la totalidad de las empresas más relevantes a nivel mundial, pueden distinguirse tres tipos diferenciados: las que son representantes de sus casas matrices con escasa generación de valor agregado doméstico, aquellas que elaboran localmente parte de sus productos o proveen distintas clases de servicios, tanto para nuestro país como para el exterior, y aquellas que son proveedoras de servicios de consultoría u outsourcing para el Estado y grandes clientes.

- Un reducido grupo de empresas de capital nacional, proveedoras de servicios, Integradoras y/o desarrolladoras de soluciones específicas, con un volumen de operaciones por encima de los $20.000.000. Estas empresas emplean corrientemente entre 100 y 500 personas.
- Un numeroso y heterogéneo conjunto de pequeñas y medianas empresas, de capital nacional o mixto, distribuidas aproximadamente en tercios entre empresas de más de 15 años en el mercado, de entre 15 y 5 años, o menor de esta antigüedad, dedicadas al desarrollo local de productos de software, ya sea para el mercado doméstico o externo, y a la provisión de servicios informáticos variados. Estas empresas emplean aproximadamente entre 5 y 100 personas.

Es este último subuniverso de PYMES de SSI, que ofrecen bienes y servicios intangibles el que será objeto de análisis en el presente estudio, tenemos que las ventas de estas PYMES de SSI promedian $1.800.000 por año. Estas empresas se encuentran jugando un papel de creciente importancia en la dinámica de la economía argentina, ya que su expansión está estrechamente ligada a la masiva introducción de las nuevas tecnologías de la información que van redefiniendo los mecanismos de producción, venta y competitividad de diferentes sectores productivos del país. [6]

En líneas generales, se proyecta a nivel nacional un crecimiento para todo el sector de un 15% durante 2006, lo cual duplicaría las expectativas de crecimiento de la economía argentina (7% según el presupuesto nacional). La inversión de estas empresas (orientada a nueva infraestructura, apertura de nuevos mercados, renovación o ampliación de los productos existentes, y la creación de nuevos centros de desarrollo, entre otros rubros) posiblemente supere los $500.000.000, llevando a que la reinversión total en el sector supere el 15% de los ingresos. [6]

En este contexto, es importante comprender que, si bien la industria ha crecido en los últimos años en un promedio anual superior al 20%, sólo podrá continuar desarrollándose en la medida en que se otorguen soluciones a los nuevos desafíos que amenazan su crecimiento y que necesitan ser afrontados en un futuro inmediato. [6]

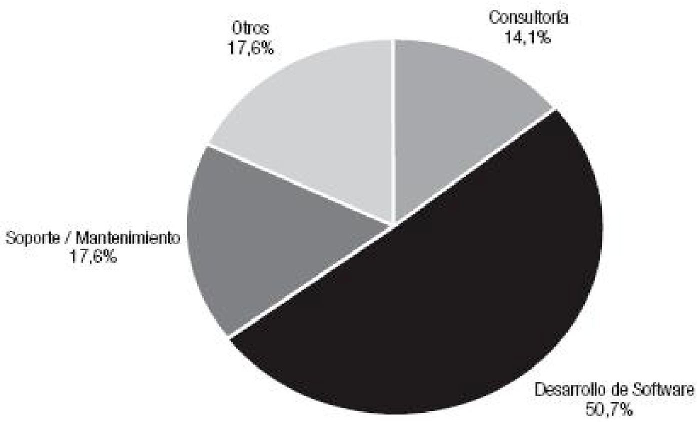
Entre los principales retos del sector, sobre todo en lo que concierne a la situación de las empresas pequeñas y medianas, se encuentran algunos de los temas mencionados como principales problemas por los empresarios encuestados , y que serán desarrollados en este estudio: la falta de acceso al financiamiento y la disponibilidad de recursos humanos entrenados.

La superación de éstos y otros obstáculos supone llevar a cabo diversas transformaciones tanto a nivel público y privado como académico. Sólo así se le podrá garantizar a las empresas argentinas de SSI las condiciones necesarias para poder aprovechar al máximo las oportunidades que hoy tienen al alcance de sus manos. [6]

Debido al que el software es intangible, existe un rango amplio y heterogéneo de servicios informáticos prestados por las pequeñas y medianas empresas. Para intentar comprender en mayor profundidad el comportamiento de las empresas encuestadas por el Observatorio PyME en función de las características del panel, comenzaremos describiendo estas firmas según su principal servicio provisto: la mitad está dedicada al desarrollo de software, alrededor de un 14% a la prestación de servicios de consultoría, casi un 18% brinda soporte o mantenimiento a otras empresas, y otro porcentaje idéntico presta otros tipos de servicios informáticos, entre los que se encuentran capacitación, implementación de software, licencias, utsourcing, procesamiento de datos y selección de recursos humanos.

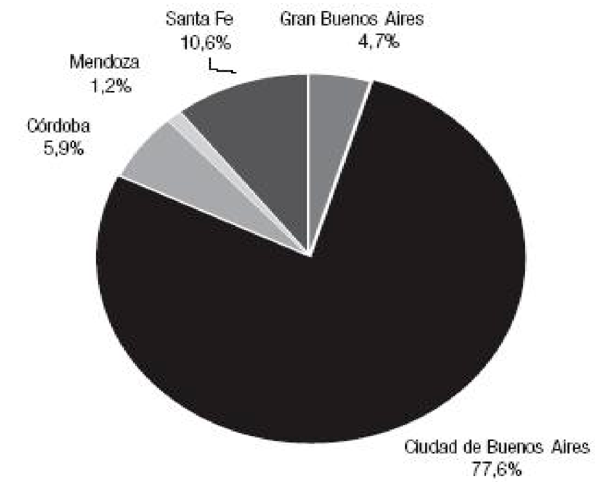
Se estima que a nivel nacional las prestadoras de servicios informáticos absorben prácticamente la mitad del empleo y de la facturación del sector como se observa en la figura 1.1.2.1

Figura 1.1.2.1 Distribución de empresas según principal servicio provisto [6]



En cuanto a la distribución espacial de las PyME de este sector, las mismas se encuentran fuertemente concentradas en la Ciudad Autónoma de Buenos Aires, en donde junto al Gran Buenos Aires se ubica más del 80% de las sedes centrales. El resto de las empresas encuestadas se encuentran en grandes centros urbanos como Rosario, Córdoba y Mendoza. Asimismo, es posible observar algunas ciudades o áreas geográficas que se destacan por el creciente asociativismo entre Pymes, alentadas por universidades o autoridades locales, o propia iniciativa, tales como Tandil, Mar del Plata, Bahía Blanca, Santa Fe y Bariloche, entre otros, como se observa en la figura 1.1.2.2

Figura 1.1.2.2 Distribución de empresas según su ubicación geografica[6]



En particular, Rosario Córdoba y Mendoza conforman algunos de los "polos o clusters tecnológicos" del interior del país, donde se conglomeran muchas empresas del sector debido a la confluencia de varios factores como, por ejemplo, la disponibilidad de recursos humanos calificados, la vinculación con las universidades locales y una cooperación tanto entre las empresas como con los gobiernos municipales y provinciales. [6]

1.1.3. La Industria del Software en Ecuador

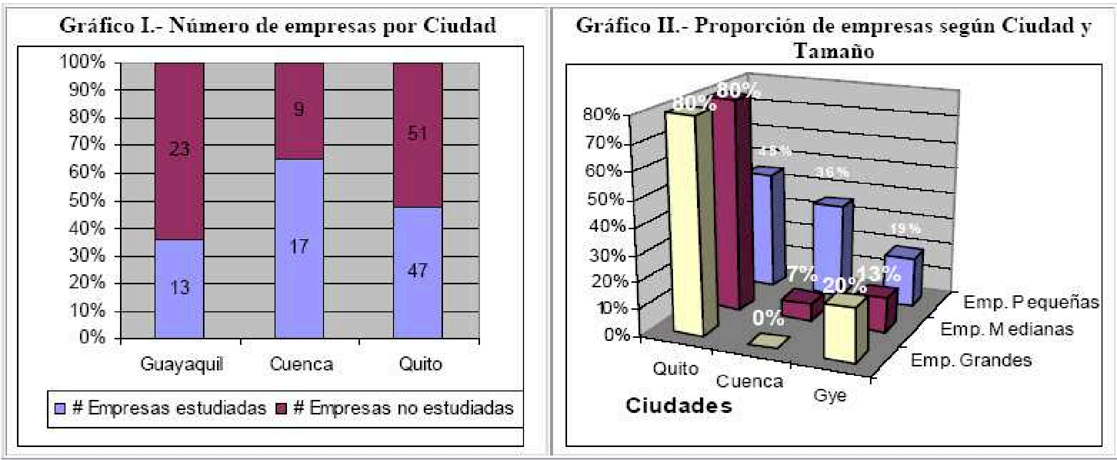
En los últimos años en Ecuador ha surgido como uno de los productos de exportación no tradicional el software; esto gracias a que la demanda de este producto a nivel global se ha incrementado a medida que las computadoras y el Internet han penetrado casi todos los aspectos de la sociedad [1]. Debido a esta demanda, se hace cada vez más necesario para las empresas desarrolladoras de software ecuatorianas la utilización de métodos, técnicas y herramientas que permitan garantizar la calidad de los productos; y así mejorar la competitividad de este nicho de mercado.

Hasta ahora, la industria ecuatoriana ha venido realizando esfuerzos para adquirir y aplicar conocimiento que permita mejorar la calidad de sus productos. Entre ellos podemos mencionar el estudio de métricas del sector del Software realizado por la AESOFT (Asociación Ecuatoriana de Software) en el año del 2005 llamado “Primer Estudio de la Industria de Software del Ecuador” [7]; y los diferentes proyectos que esta comunidad lleva a cabo con el objetivo de capacitar a las empresas socias y ayudar a la certificación de las mismas en el estándar de calidad del software CMM. También la ESPOL se ha preocupado de esta temática, y por medio del proyecto VLIR-ESPOL ha llevado a cabo varios estudios [8].

El Software del Ecuador cuenta con un potencial exportador real demostrado en múltiples casos de éxito de la empresa privada que han sido reconocidos en los mercados internacionales especialmente el Software Bancario [6].

En lo que respecta a la calidad en la industria ecuatoriana, se afirma que el 94.8% de las empresas conocen acerca de la norma ISO 9001, el 48% acerca de MSF y el 29.8% acerca de CMM. En el mismo estudio se determino que el 36.3% de las empresas encuestadas utilizaban estándares de calidad en el desarrollo de software, de los cuales, sólo el 24.6% correspondía a estándares internacionalmente reconocidos. [8]

Figura 1.1.3.2 Distribución de empresas por ciudades [8]



Observamos que en la ciudad de Guayaquil y Cuenca la mayor parte de las empresas son pequeñas, mientras que en la ciudad de Quito la mayoría de las empresas son medianas. De toda la población, la mayoría de las empresas grandes están asentadas en Quito como se muestra en la figura 1.1.3.2 [8].

Refiriéndose al estudio realizado en el año 2006 por la ESPOL [60]; se notó que en cuanto a la dirección de proyectos en Ecuador, solo el 40% de las empresas de software siempre documentan los procesos, además según las respuestas de los entrevistados creen que las empresas no llevan documentación debido principalmente a la urgencia de terminar un proyecto. Este es un aspecto de vital importancia considerando que teniendo una buena documentación facilita el mantenimiento y la renovación de un producto de software; es más esta documentación se convierte en un lenguaje común que permite a diferentes personas trabajar en proyectos similares. Por otro lado, aunque una gran parte de los entrevistados demostraron una falta de conocimiento sobre el concepto de métricas de software, 68% de las compañías llevan a cabo algún tipo de medida en sus proyectos [60].

En otro artículo publicado en el año 2007 por la ESPOL [61], mediante el cual se presentó un estudio del porcentaje de error en la estimación del esfuerzo, el cual indica el grado de acercamiento de los proyectos a sus estimaciones de tiempo; el promedio general de todos los proyectos estimo el 5.5% menos del tiempo necesario para desarrollarlos. Un caso particular a considerar es que un proyecto se estimo un 12% menos del tiempo total requerido, situación que rectifica la tendencia en el sentido de que la gran mayoría de los proyectos informáticos desarrollados por las PYMES ecuatorianas carecen de una adecuada estimación.

Además se obtuvieron porcentajes sobre las inspecciones por cada fase, en donde se observó que la mayor parte de las inspecciones se realizan en la etapa de Especificaciones, y que las fases siguientes tuvieron niveles de inspección porcentualmente decrecientes. Por lo cual las empresas deben de inspeccionar también su plan de trabajo para prever de manera temprana complicaciones en el desarrollo del proyecto [61].

A pesar de todos estos esfuerzos, el campo de la Ingeniería de Software es tan amplio que aún queda mucho por estudiar. Así por ejemplo tenemos que no existen estudios respecto al uso de métodos, técnicas y herramientas de V&V de software que se estén usando actualmente por parte de la industria de software ecuatoriana, ni se conoce tampoco cuál ha sido el impacto que el uso de los mismos puede haber tenido para las empresas. Es debido a la falta de estos estudios, que no se tiene ahora una base local de conocimiento que permita a las empresas dedicadas a la elaboración de sistemas de software, la selección apropiada de métodos, técnicas o herramientas de V&V que garanticen un impacto positivo en la calidad de los productos de software generados.

1.2. Objetivos

Los Objetivos planteados para el siguiente estudio son los siguientes:

1. Identificar que porcentaje de las empresas ecuatorianas usan métodos, técnicas y herramientas de verificación y validación en el proceso de desarrollo de software.
2. Identificar los métodos, técnicas y herramientas de validación y verificación usadas actualmente por las empresas desarrolladoras de Software del Ecuador.
3. Identificar en qué etapas del proceso de desarrollo de software las empresas ecuatorianas utilizan métodos, técnicas y herramientas para la validación y verificación de Software.

4. Determinar los factores que inciden en el proceso de validación y verificación en las empresas ecuatorianas desarrolladoras de Software.
5. Elaborar un listado de herramientas, métodos y técnicas existentes de verificación y validación del software apropiado para las necesidades de las empresas ecuatorianas desarrolladoras de software.

1.3. Justificación de la Tesis

En la actualidad las empresas ecuatorianas desarrolladoras de Software en su gran mayoría no poseen un proceso de desarrollo debidamente documentado en el cuál se indique cuales son los métodos, técnicas o herramientas más adecuadas que se pueden aplicar para la elaboración de un sistema de Software.

Basándonos en esta realidad latente por parte de las empresas y tomando en cuenta que hoy en día existen a disposición una amplia gama de métodos, técnicas y herramientas cuyo uso adecuado nos puede servir para ayudar a la tarea de verificación y validación en el proceso de desarrollo de Software, hemos creído importante llevar a cabo esta investigación para que pueda ser utilizada por las empresas desarrolladoras de Software del país; y así coadyuvar a mejorar sus procesos y a su vez elaborar un producto de calidad.

CAPÍTULO II

2. MARCO TEORICO

2.1. Proceso de Desarrollo de Software

Un proceso de desarrollo de software tiene como propósito la producción eficaz y eficiente de un producto software que reúna los requisitos del cliente. Dicho proceso, en términos globales se muestra en la Figura 2.1.1. Este proceso es intensamente intelectual, afectado por la creatividad y juicio de las personas involucradas [4]. Aunque un proyecto de desarrollo de software es equiparable en muchos aspectos a cualquier otro proyecto de ingeniería, en el desarrollo de software hay una serie de desafíos adicionales, relativos esencialmente a la naturaleza del producto obtenido. A continuación se explican algunas particularidades asociadas al desarrollo de software y que influyen en su proceso de construcción.

Un producto software en sí es complejo, es prácticamente inviable conseguir un 100% de confiabilidad de un programa por pequeño que sea. Existe una inmensa combinación de factores que impiden una verificación exhaustiva de todas las posibles situaciones de ejecución que se puedan presentar (entradas, valores de variables, datos almacenados, software del sistema, otras aplicaciones que intervienen, el hardware sobre el cual se ejecuta, entre otras.).

Un producto software es intangible y por lo general muy abstracto, esto dificulta la definición del producto y sus requisitos, sobre todo cuando no se tiene precedentes en productos de software similares. Esto hace que los requisitos sean difíciles de consolidar tempranamente. Así, los cambios en los requisitos son inevitables, no sólo después de entregado el producto sino también durante el proceso de desarrollo.

Además, de lo anterior, siempre puede señalarse la inmadurez de la ingeniería del software como disciplina, justificada por su corta vida comparada con otras disciplinas de la ingeniería.

Figura 1.1.1 Proceso de desarrollo de software. [9]

El proceso de desarrollo de software no es único. No existe un proceso de software universal que sea efectivo para todos los contextos de proyectos de desarrollo. Debido a esta diversidad, es difícil automatizar todo un proceso de desarrollo de software.

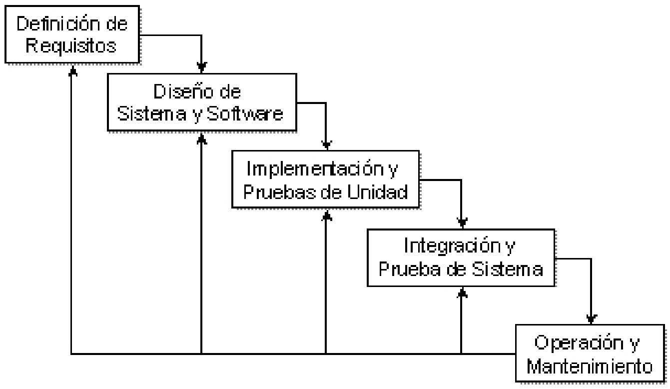
Dentro del Proceso de Desarrollo de Software Sommerville [4] define un modelo de proceso de software como "Una representación simplificada de un proceso de software, representada desde una perspectiva específica. Por su naturaleza los modelos son simplificados, por lo tanto un modelo de procesos del software es una abstracción de un proceso real."

Los modelos genéricos no son descripciones definitivas de procesos de software; sin embargo, son abstracciones útiles que pueden ser utilizadas para explicar diferentes enfoques del desarrollo de software.

A continuación describimos algunos modelos:

- Codificar y corregir: Este es el modelo básico utilizado en los inicios del desarrollo de Software. Se compone de 2 pasos: Escribir el código y Corregir problemas en el código.
- Modelo en cascada: Este toma las actividades fundamentales del proceso de especificación, desarrollo, validación y evolución y las representa como fases separadas del proceso. Se compone de las siguientes fases: Definición de los requisitos, Diseño de software, Implementación y pruebas unitarias, Integración y pruebas del sistema y Operación y mantenimiento

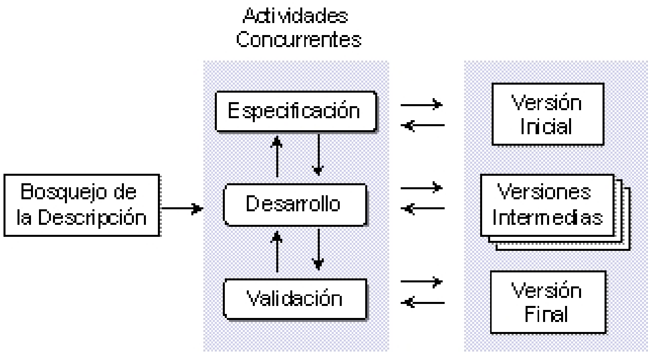
Figura 2.1.2 Modelo de desarrollo en Cascada



- Desarrollo evolutivo: La idea de este modelo es el desarrollo de una implantación del sistema inicial, exponerla a los comentarios del usuario, refinarla en N versiones hasta que se desarrolle el sistema adecuado. En la Figura 2.1.3 se observa cómo las actividades concurrentes: especificación, desarrollo y validación, se realizan durante el desarrollo de las versiones hasta llegar al producto final.

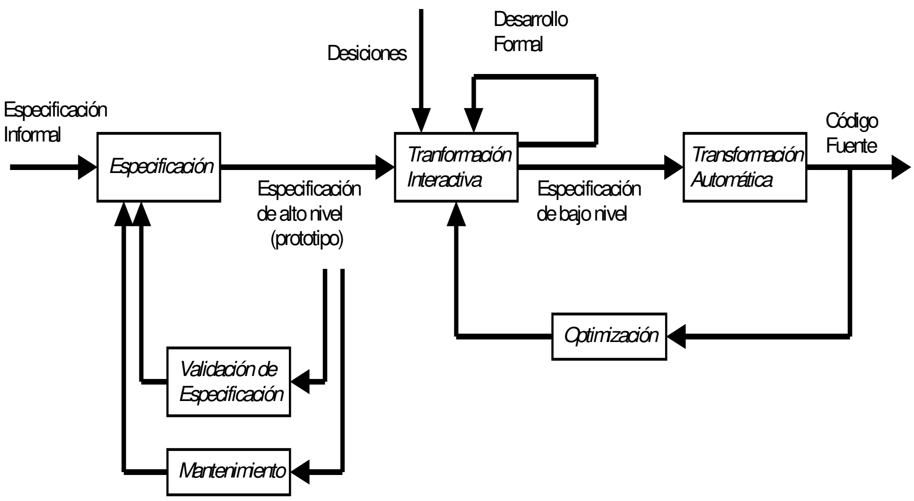
  Una ventaja de este modelo es que se obtiene una rápida realimentación del usuario, ya que las actividades de especificación, desarrollo y pruebas se ejecutan en cada iteración [4].

Figura 2.1.3 Modelo de desarrollo Evolutivo



- Desarrollo formal de sistemas: Este modelo se basa en transformaciones formales de los requisitos hasta llegar a un programa ejecutable, se ilustra en la figura 2.1.4.

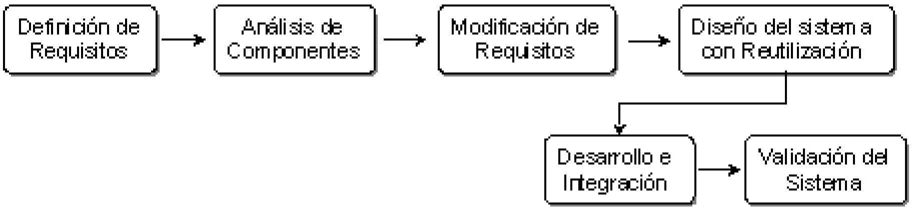
Figura 2.1.4 Paradigma de programación Automática



- Desarrollo basado en reutilización: Como su nombre lo indica, es un modelo fuertemente orientado a la reutilización. Este modelo consta de 4 fases ilustradas en la Figura 2.1.5. A continuación se listan las fases: Definición de requerimientos, Análisis de Componentes, Modificación de Requerimientos, y Diseño del

Sistema con Reutilización que a su vez se subdivide en Desarrollo e integración y Validación del sistema.

Figura 2.1.5 Desarrollo basado en reutilización de componentes

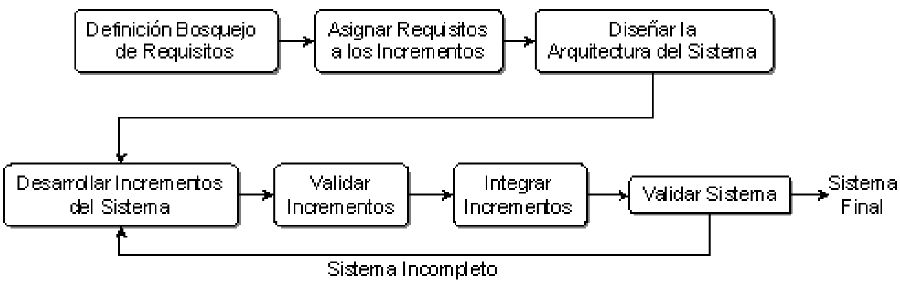


- Desarrollo incremental: Mills sugirió el enfoque incremental de desarrollo como una forma de reducir la repetición de trabajo en el proceso de desarrollo y dar oportunidad de rastrear la toma de decisiones en los requisitos hasta adquirir experiencia con el sistema (ver Figura 2.1.6). Es una combinación del Modelo de Cascada y Modelo Evolutivo. **[10]**

  Reduce el rehacer trabajo durante el proceso de desarrollo y da oportunidad para retrasar las decisiones hasta tener experiencia en el sistema.

  Durante el desarrollo de cada incremento se puede utilizar el modelo de cascada o evolutivo, dependiendo del conocimiento que se tenga sobre los requisitos a implementar. Si se tiene un buen conocimiento, se puede optar por cascada, si es dudoso, evolutivo. **[10]**

Figura 2.1.6 Modelo de desarrollo iterativo incremental [10]

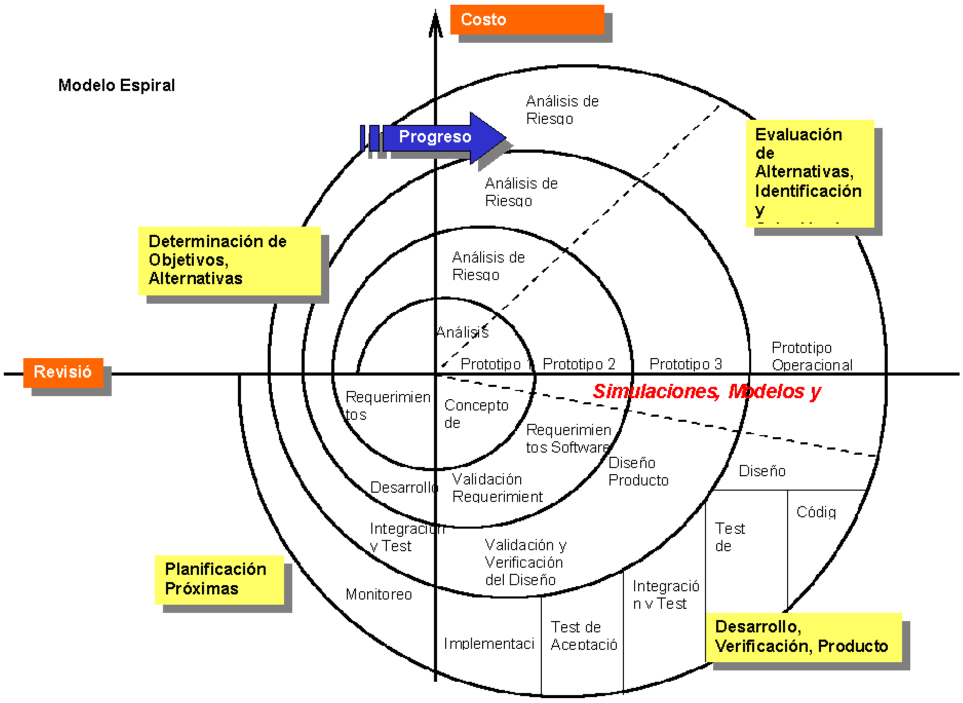


- Desarrollo en espiral: El modelo de desarrollo en espiral (ver Figura 2.1.7) es actualmente uno de los más conocidos y fue propuesto por Boehm. El ciclo de desarrollo se representa como una espiral, en lugar de una serie de actividades sucesivas con retrospectiva de una actividad a otra.

  Cada ciclo de desarrollo se divide en cuatro fases [11]:

  i. Definición de objetivos: Se definen los objetivos. Se definen las restricciones del proceso y del producto. Se realiza un diseño detallado del plan administrativo. Se identifican los riesgos y se elaboran estrategias alternativas dependiendo de estos.
  ii. Evaluación y reducción de riesgos: Se realiza un análisis detallado de cada riesgo identificado. Pueden desarrollarse prototipos para disminuir el riesgo de requisitos dudosos. Se llevan a cabo los pasos para reducir los riesgos.
  iii. Desarrollo y validación: Se escoge el modelo de desarrollo después de la evaluación del riesgo. El modelo que se utilizará (cascada, sistemas formales, evolutivo, entre otros.) depende del riesgo identificado para esa fase.
  iv. Planificación: Se determina si continuar con otro ciclo. Se planea la siguiente fase del proyecto.

Figura 2.1.7 Modelo de desarrollo en Espiral [11]



2.2 Verificación de Software

2.2.1 Conceptos básicos de Verificación

La verificación es el proceso de evaluar un sistema o componente para determinar si el producto en una fase dada de desarrollo satisface las condiciones impuestas al inicio de la fase **[12]**.

Otra manera de ver la tarea de Verificación es respondiéndose a la pregunta “¿Estamos construyendo el producto correctamente?”

2.2.2 Verificación Estática

Se ocupa del análisis de representaciones estáticas del sistema para describir problemas.

Pueden ser complementadas por documentos basados en herramientas y análisis del código **[12].**

2.2.3 Verificación Dinámica

Se ocupa de la ejercitación y la observación del comportamiento del producto. A demás permite que la detección del defecto sea combinada con otros chequeos de la calidad [4].

El sistema se ejecuta con datos de pruebas y se observa su comportamiento operativo.

2.3 Validación de Software

2.3.1 Conceptos básicos de Validación

La validación es el proceso de evaluación de un sistema o componente durante o al final del proceso de desarrollo para determinar si esto satisface los requerimientos especificados [12].

Otra manera de ver la tarea de Validación es respondiéndose a la pregunta "¿Estamos construyendo el producto correcto?"

2.3.2 Validación simple

El proceso de validación simple se basa en la inspección del cumplimiento de los requerimientos mediante el empleo de un proceso establecido para el desarrollo [4] [13].

2.3.3 Validación cruzada

La validación cruzada, es la práctica estadística de partir una muestra de datos en subconjuntos de tal modo que el análisis es inicialmente realizado en uno de ellos, mientras los otros subconjuntos son retenidos para su uso posterior en la confirmación y validación del análisis inicial [14].

2.4 Herramientas de Distribución Libre para Verificación y Validación de Software.

Existen en la actualidad muchos aportes en la web de personas involucradas en el proceso de Verificación y Validación de software que basados en su experiencia obtenida con el pasar de los años han podido brindar un aporte significativo; especialmente la comunidad de código abierto. A continuación listamos algunas de las herramientas cuyo adecuado uso nos puede brindar excelentes resultados.

- UISpec4J (Framework)
  Es una librería funcional de código abierto y/o prueba de unidad para las aplicaciones de java basadas en Swing.

  Los APIs de UISpec4J son diseñados para esconder lo más que pueda la complejidad de Swing, resultando fácil de escribir y leer las pruebas de scripts.

  Su meta principal es de proveer un soporte o ayuda para las pruebas de los casos de escritura de unidad y de funcionalidad basado en la aplicación de la interfaz del usuario. UISpec4J quiere permitir que sus pruebas escritas se conviertan en las actuales especificaciones detalladas de una aplicación [15].

- JUnit
  Es un conjunto de librerías que nos permiten realizar la ejecución de clases Java de una manera controlada para poder evaluar si el funcionamiento de cada uno de los métodos de la clase se comporta como se espera. Es decir, en función de algún valor de entrada se evalúa el valor de retorno esperado; si la clase cumple con la especificación, entonces JUnit devolverá que el método de la clase pasó exitosamente la prueba; en caso de que el valor

esperado sea diferente al que regresó el método durante la ejecución, JUnit devolverá un fallo en el método correspondiente **[16].**

- TestManager
  Es una herramienta de plataforma cruzada que se utiliza para el manejo de casos de prueba. Esta herramienta utiliza un diseño cliente-servidor donde un administrador manda a correr un servidor, por medio del cual los usuarios acceden a través de una página web estándar [17].

- Java Path Explorer
  Esta herramienta nos permite monitorear la ejecución de programas Java. Este ambiente facilita las pruebas de ejecución de trazas a diferencia de los niveles de especificación altos, incluida la lógica temporal de formulas. Además esta herramienta contiene algoritmos para detectar errores en programas concurrentes como los Interbloqueos (Dead-Locks) [18].

- JMock 2.0
  Es una librería que soporta desarrollo de prueba-conducida de código de Java con objetos ficticios. Los objetos ficticios ayudan a diseñar y probar las interacciones entre los objetos en los programas [19].

  La librería jMock:

    - Hace más fácil y rápido definir los objetos ficticios.

- Le permite especificar las interacciones entre los objetos, reduciendo así la quebranto de las pruebas.
- Trabaja bien con el auto completado y re fabricación de elementos en tu IDE.
- Se conecta en tu prueba favorita de esquema.
- Es fácil de extender.

- Selenium
  Selenium es una herramienta de prueba para aplicaciones web. Se ejecuta directamente en el navegador de nuestra aplicación, esto es lo que justamente un usuario real hace **[20].** Este se puede ejecutar en Internet Explorer, Mozilla y Firefox en Windows y Macintosh, Safari en la Mac.

- JsTester
  Permite la validación de código javascript dentro de java. Provee un grupo de métodos de afirmaciones, que permite que el usuario use sus propias validaciones [21].

  El proyecto prueba dos maneras de crear pruebas: por herencia (JsTestCase) y por composición (JsTester). Puede utilizar JsTestCase así como utilizaría cualquier extensión de TestCase.

- JSystem(Framework)
  Es un esquema para escritura y corrida de pruebas automatizadas, basado en JUnit [22]. Su propósito principal es dar soporte al automatizado de pruebas funcionales o del sistema. Se entiende que la automatización del proyecto es un proyecto de

software, y para esto se necesita del lenguaje más fuerte y de ambiente de desarrollo. En JSystem, las pruebas son escritas en Java [23].

- DDTUnit (Framework)
  Sirve para crear pruebas de módulo o componente durante el ciclo de desarrollo, es extremadamente útil y productivo para obtener un enfoque de información céntrica para cubrir múltiples casos de prueba en métodos de prueba codificados. La idea básica de DDTUnit es proveer una descripción XML de casos de prueba y combinarlos con la simplicidad de JUnit. Todo el flujo del programa es codificado en Java [24].

- Jameleon
  Es un esquema automatizado de pruebas que puede ser fácilmente usado ya sea por usuarios técnicos y no técnicos. Uno de los conceptos principales detrás de Jameleon es de crear un grupo de palabras claves o etiquetas que representen diferentes pantallas de una aplicación. Toda la lógica requerida para automatizar cada pantalla particular puede ser definida en Java y mapeada a esas palabras claves. Las palabras claves pueden ser organizadas con distintos conjunto de data para formar pruebas de scripts sin requerir un conocimiento profundo de cómo la aplicación funciona. Las pruebas de scripts son usadas para automatizar las pruebas y generar la documentación de casos de estudio manuales [25].

- T2
  T2 es totalmente automático, una herramienta de prueba aleatoria basada en la trazabilidad, ofreciendo especificaciones in-code y pruebas reflexivas. También es casi Iterativo; dependiendo de la complejidad de sus clases este puede responder en menos de un segundo.

  T2 chequea errores internos, errores en tiempo de ejecución (run time exception), y invariantes de clase. Al contrario de otras herramientas de pruebas, T2 lee especificaciones escritas en Java puro. Ellas se localizan en la clase que nosotros especificamos. El esfuerzo de mantenimiento también es mínimo [26].

- Kronos
  Es una herramienta desarrollada para verificar sistemas de tiempo real complejos. Los sistemas de tiempo real deben desarrollar un trabajo sin estrictos límites de tiempo. Controladores embebidos, circuitos o protocolos de comunicación son ejemplos de sistemas dependientes del tiempo. Estos sistemas son usualmente parte de aplicaciones complejas que son difíciles de desarrollar y analizar, pero que deben tener un correcto comportamiento, puesto que una falla tendría consecuencias severas.

  En Kronos, los componentes de sistemas en tiempo real son modelados por autómata temporalizado y los requerimientos correctos son expresados en lógica temporal de tiempo real

TCTL. Kronos es distribuido de manera gratuita a instituciones académicas [27].

- HyTech (The Hybrid Technology Tool)
  HyTech es una herramienta automática utilizada para el análisis de sistemas embebidos. HyTech computa las condiciones bajo las cuales un sistema lineal híbrido satisface un requerimiento temporal. Los sistemas híbridos son denominados como colecciones de autómata con componentes discretos y continuos, y los requerimientos temporales son verificados por chequeos de modelos simbólicos. Si la verificación falla, Hytech genera un rastro de diagnostico de error [28].

- SMV (Symbolic Model Verifier)
  Esta herramienta nos permite manejar volúmenes grandes de casos de prueba para verificar sistemas cuyo número de escenarios crece exponencialmente con los de datos de entrada.

  El método usa fórmulas de Booleanas para representar conjuntos y relaciones para evitar la construcción de un grafo de estados de transición explícito que ocupa una gran cantidad de memoria del computador [29].

- Spin
  Es una herramienta de software abierto popular, utilizado por miles de personas en el mundo entero, que pueden ser usados para la verificación formal de sistemas de software distribuidos.

Una de las metas de Spin es la verificación de software eficiente, no de hardware. La herramienta soporta lenguajes de alto nivel para especificar las descripciones de los sistemas, llamada PROMELA.

Spin puede ser usado como, simulador, verificador exhaustivo, un driver para verificación de swarm (enjambre) o como un sistema de aproximación de prueba [30].

- UPPAAL

  Es una herramienta integrada de ambiente para modelar, verificar y validar modelados de los sistemas de tiempo-real como redes de autómata temporalizados, extendidos con tipos de datos (arreglos, etc). La herramienta ha sido desarrollada por el Departamento de Información Tecnológica en la Universidad de Uppsala de Sweden y el Departamento de Ciencias Computacionales en la Universidad de Aalborg en Dinamarca [31].

2.5 Herramientas de Distribución Comercial para Verificación y Validación de Software.

- TeamSystem

  Team System 2008 es una aplicación integrada de administración (ALM) abarcando herramientas, procesos y la guía para ayudar a cualquiera del equipo a mejorar sus habilidades y a trabajar en grupo de una manera más efectiva. Visual Studio Team System les permite a los miembros de su grupo [32]:

- Colaborar y comunicarse de una manera más efectiva con otros miembros del grupo.
- Asegurar la calidad del software utilizando herramientas de calidad avanzada en cada paso del ciclo de vida de la aplicación.
- Ganar visión en las actividades y prioridades de un proyecto para hacer decisiones informadas basados en datos en tiempo real.

- ChecKing de ALS
  checKing es la herramienta de monitorización del proceso de desarrollo software y sus resultados, que cubre las necesidades de organizaciones que desean controlar la calidad del software antes de su puesta en producción.

  Para ello la herramienta sigue un modelo de métricas en el que se integran medidas obtenidas automáticamente del proceso de desarrollo (actividad, requisitos, defectos y cambios) y de elementos analizables del software: código fuente, documentación del proyecto, scripts de construcción, y scripts de pruebas.

  checKing añade transparencia al ciclo de desarrollo, capturando, integrando y presentando de forma automática dichas métricas, obtenidas automáticamente través de diversas herramientas y sistemas externos: analizadores, gestores de defectos, sistemas de control de versiones, herramientas de pruebas, IDEs y otras herramientas usadas por los desarrolladores[33].

- Open Load Tester
  Es la solución de optimización de desarrollo mas fácil de usar en la industria, utilizada para pruebas funcionales y de load, así como también para el monitoreo de desarrollo de páginas dinámicas y aplicaciones web. Open Load Tester V5.0 provee una facilidad, precisión y escalabilidad sin precedentes para el desarrollo integrado y las pruebas de ambiente [34].

- QACenter de Compuware
  QACenter nos brinda una prueba de productos automatizados y soluciones diseñadas para validar aplicaciones en curso, en el completo espectro de los entornos, aislamientos y corrección de problemas. Asegurar que los sistemas pueden manejar la carga anticipada, antes que la aplicación empiece a funcionar [35].

- Security Tester de Fortify
  Fortify Security Tester para Visual Studio 2005, proporciona seguridad de pruebas efectiva para el desarrollo y a los equipos de calidad, permitiéndoles verificar la lectura de la seguridad de las aplicaciones antes de ser desplegadas. Fortify Security Tester es un producto de próxima generación de pruebas de seguridad que aprovecha el poder de las pruebas de caja blanca dentro del equipo de sistemas de Visual Studio, dando visibilidad al origen de las causas de imperfecciones, no solo de los síntomas. Las vulnerabilidades de la seguridad se arreglan rápidamente y las aplicaciones pueden ser desplegadas con confianza [36].

- SOATest de Parasoft

Parasoft SOAtest es una prueba colaborativa y es una herramienta de análisis diseñada especialmente para las pruebas y validaciones de servicios de arquitecturas orientadas [37].

Provee de los siguientes beneficios:

- Entrega un proceso de calidad que no sea dependiente de QA.
- Maneja consistencia SOA.
- Ahorra tiempo y se asegura del éxito del proyecto.

- Software Security

  Fortify Software Security Manager proporciona a todos los usuarios, en tiempo real, la información de seguridad del software que necesitan. Es el almacén para un número de componentes del ciclo de vida de desarrollo seguro. Estos componentes deben incluir proyectos, resultados de audiencias, conjuntos de reglas de códigos de seguridad, diferentes tipos de políticas, identidades de usuarios y mucho más. Fortify Software Security Manager proporciona también relaciones entre diferentes componentes y permite a la organización seguirle la pista a dichas relaciones. Una gran cantidad de información secundaria se genera desde las métricas basadas en las relaciones de varios componentes de seguridad [38].

- .TEST de Parasoft

Test parasoft es una solución integrada para automatizar el amplio rango de las mejores prácticas probadas para aumentar la productividad del equipo de desarrollo del software y la calidad del software. Las pruebas de Parasoft les garantizan a los desarrolladores que su código en .NET trabaje como ellos lo esperan permitiéndoles codificar políticas forzadas, análisis estáticos y unidades de prueba. También guarda el tiempo de desarrollo de los equipos por medio de un proceso de revisión de código manual. Puede ser usado en escritorio como un plugin de Microsoft Visual Studio y en procesos agrupados vía la interfaz de línea de comando para pruebas de regresión [39].

Parasoft Test trabaja con lenguajes que pertenecen a Microsoft.NET, incluyendo C#, VB.NET y MC++. Prueba cualquier archivo que ha sido construido para tomar ventaja de .NET CLR.

- Webking de Parasoft
  Es una solución avanzada automatizada de verificación para la creación, ejecución y administración. Ahora está disponible con el apoyo para los desarrollos de AJAX. AJAX es una técnica web de desarrollo para crear aplicaciones web altamente interactivas. Parasoft WebKing ha sido diseñado especialmente para manejar aplicaciones AJAX [40].

  Tiene las siguientes capacidades:

  - El código de JavaScript puede tener muchos errores que son difíciles de encontrar. Parasoft WebKing los encuentra por ti.

- Evalúa aplicaciones AJAX tales como los Google Maps.
- Elimina la molestia de reescribir scripts cuando tu aplicación web cambia. Usuarios que son menos técnicos podrán crear casos de prueba en una interfaz fácil de usar y convertir esos casos de prueba en unidades de prueba accesibles a los desarrolladores.

- IBM Rational

  Es una de las herramientas mas usadas por las empresas, ya que adapta la metodología RUP (Rational Unified Process).

  Brinda una solución completa, cubriendo todas las disciplinas del ciclo de vida del Software; logrando así un desarrollo integrado del negocio [41].

  Entre las principales soluciones que se aplican para la tarea de Verificación y Validación que nos ofrece la suite de IBM tenemos las siguientes:

  Requisite Pro

  Funtional/Perfom Tester

  ClearQuest

  ClearCase

2.6 Técnicas de Verificación y Validación de Software.

- Pruebas Funcionales y Análisis

Esta técnica consiste en ejecutar parte o todo el sistema para validar que el requisito ó los requisitos del usuario se encuentren cubiertos [12].

- Pruebas Estructurales y Análisis
  Nos permite examinar la lógica de las unidades y puede usarse para apoyar los requerimientos del software para una prueba de cobertura, es decir, cuánto del programa se ha ejecutado [12].

- Error-Oriented
  Está basada en algo muy simple: enfocarse en los errores que aparecerán durante y después de la elaboración del Software para poder mitigar su impacto [42].

- Testing and Analysis
  Son actividades de Análisis y Pruebas que ocurren durante el proceso de desarrollo y evolución de un sistema de software, desde los primeros requerimientos entregados por los usuarios y su subsiguiente evolución. La calidad depende en parte de un buen proceso de software, no solo de Análisis y Pruebas; no existe una cantidad de Análisis y Pruebas que pueda recuperar la pobre calidad que se levanta en otras actividades del proceso de Software. Por otro lado una característica esencial del proceso de software que origina productos de alta calidad es que el Análisis y Prueba es completamente integrado y no luego del proceso [43].

- Estrategias de Integración

La estrategia de integración es una técnica sistemática para construir la estructura del programa mientras al mismo tiempo, se lleva a cabo pruebas para detectar errores asociados con la interfase. El objetivo es tomar los módulos probados en unidad y estructurar un programa que esté de acuerdo con el que dicta el diseño [44].

- Hybrid Approaches

  Es el proceso mediante el cual se combinan mas de una Técnica al proceso de V&V, el objetivo de esto es aprovechar las ventajas de cada una de las técnicas combinadas, contribuyendo de esta manera a elaborar un software de calidad [12].

- Análisis de Flujo de Transacción

  El flujo de transacción se caracteriza por el movimiento de datos a través de un camino de llegada que convierte la información del mundo exterior en una transacción. Se evalúa la transacción y de acuerdo con su valor, el flujo sigue por uno de los muchos caminos de acción [45].

- Análisis de Stress

  Es el proceso de determinar la habilidad de un sistema o dispositivo de mantener un cierto nivel de efectividad bajo las condiciones desfavorables. El proceso puede involucrar pruebas cuantitativas hechas en un laboratorio, como medir la frecuencia de errores o caídas del sistema. [46]

- Análisis de Falla
  Permite mitigar el impacto de una Falla previamente identificada en cualquier fase de del proceso de desarrollo de Software. Se evalúan las posibles causas y el origen [47].

- Análisis de Concurrencia
  Permite describir las múltiples actividades del software ocurriendo simultáneamente. Normalmente es empleado para el desarrollo de aplicaciones Cliente/Servidor, en el cual me permite simular la conexión al sistema de manera simultánea [48] [49].

- Análisis de Algoritmos
  Provee estimaciones teóricas para los recursos que necesitan cualquier algoritmo que resuelva un problema dado. Estas estimaciones resultan ser bastante útiles en la búsqueda de algoritmos eficientes [50].

- Análisis de Cobertura
  Está asociado con las pruebas Unitarias, nos permite conocer la magnitud de los cambios que se van a llevar a cabo en el proceso de desarrollo.

  Debemos tener en cuenta que la cobertura nos proporciona información sobre el código de la aplicación que se ha ejecutado, lo que muchas veces obviamos es que esa ejecución no tienen por qué estar provocada por la ejecución de pruebas unitarias [51].

- Análisis de Flujo de Datos

El diagrama de flujo de datos DFD (o de burbujas) se utiliza como herramienta gráfica para la descripción del flujo de la información en las diferentes etapas del proceso de desarrollo de software.

Mediante esta técnica nos permite tener una visión global del sistema a desarrollar ya que podemos pronosticar las variables antes de que sean escritas en el programa [45].

- Inspecciones

  Esta técnica permite la evaluación de los requerimientos del software, diseño del software, o código; son examinados por una persona que conoce el modelo de negocio, el cual identifica los errores de estándares de desarrollo, y otros problemas.

  La inspección empieza encontrando la mayor cantidad de errores para luego registrarlos, pero ninguno se lo corrige en ese momento. Sin embargo, para la siguiente revisión futura se verifica que se encuentren corregidos realmente [52].

- Model Checking

  Es una técnica de verificación para un sistema formal, en la mayoría de las ocasiones derivado del hardware o del software de un sistema informático. El sistema es descrito mediante un modelo, que debe satisfacer una especificación formal descrita mediante una fórmula, a menudo escrita en alguna variedad de lógica temporal [53].

  Esta Técnica se puede expresar como un sistema de transiciones, es decir, un grafo dirigido, que consta de un conjunto de vértices y arcos. Un conjunto de proposiciones atómicas se asocia a cada nodo. Así pues, los nodos representan los estados posibles de un sistema, los arcos posibles evoluciones del mismo, mediante ejecuciones permitidas, que alteran el estado, mientras que las proposiciones representan las propiedades básicas que se satisfacen en cada punto de la ejecución.

Existen herramientas automáticas para realizar el Model checking, basadas en técnicas combinatorias, explorando el espacio de estados posibles; lo que conduce al problema de explosión de estados.

2.7 Métodos de Verificación y Validación de Software.

- Caja Negra

Las pruebas de caja negra se centran en lo que se espera de un módulo, es decir, intentan encontrar casos en que el módulo no se atiene a su especificación. Por ello se denominan pruebas funcionales, y el probador se limita a suministrarle datos como entrada y estudiar la salida, sin preocuparse de lo que pueda estar haciendo el módulo por dentro [12].

Las pruebas de caja negra intentan hallar errores tales como:

Funciones incorrectas o ausentes

Errores de Interfaz

Errores en estructuras de datos o en acceso a BD externas

Errores de rendimiento

- Caja Blanca
  Es un método de prueba llamado también pruebas de Cobertura en el cual se conoce el diseño interno del Software, como ha sido desarrollado a diferencia del método de caja negra en este método se hace énfasis en el detalle de la implementación.

  Su objetivo principal es probar la lógica del programa desde el punto de vista algorítmico [12].

  Básicamente la idea de pruebas de cobertura consiste en diseñar un plan de pruebas en las que se vaya ejecutando sistemáticamente el código hasta que haya corrido todo o la gran mayoría de el, esto que parece complicado es mas aún cuando el programa contiene código de difícil alcance, como por ejemplo manejadores de errores o "código muerto".

- Top-Down
  En el modelo Top-down se formula un resumen del sistema, sin especificar detalles. Cada parte del sistema se refina diseñando con mayor detalle. Cada parte nueva es entonces redefinida, cada vez con mayor detalle, hasta que la especificación completa es lo suficientemente detallada para validar el modelo. El modelo "Top-down" se diseña con frecuencia con la ayuda de "cajas negras" que hacen más fácil cumplir requerimientos aunque estas cajas negras no expliquen en detalle los componentes individuales [12].

- Act-like-a-customer

Es un método de prueba en el cual las pruebas se desarrollan basadas en el conocimiento de cómo los clientes/usuarios usan el Software. Basados en la experiencia podemos afirmar que los productos de software complejos tienen muchos errores y que los clientes/usuarios normalmente encuentran solo un pequeño porcentaje de estos errores [12].

- ATAM
  El método ATAM “Architecture Tradeoff Analysis Method” por sus siglas en Ingles es método estructurado para hacer análisis repetible.

  Este método de evaluación obtiene su nombre no solo porque nos dice cuan bien una arquitectura particular satisface las metas de calidad, sino que también provee ideas de cómo esas metas de calidad interactúan entre ellas [54].

- ADR(Active Design Reviews)
  El método Revision de Diseño Activo, ADR por sus siglas en Ingles, es una técnica efectiva para asegurar la calidad del producto de software. Se usa ADRs principalmente para evaluar planes detallados de unidades coherentes de software, como módulos o componentes.

  Al igual que el método ATAM consta de una serie se preguntas, las preguntas tienden a dirigirse (a) la calidad e integridad de la documentación; y (b) suficiencia, aptitud, y conveniencia de los servicios proporcionados por el diseño [55].

- ARID(Active Reviews for Intermediate Designs)
  El método Revision Activa para Diseño Intermedio, ARID por sus siglas en Ingles; es una combinación entre método ATAM y el método ADR, el cual explota las ventajas de cada uno de estos métodos.

  Este método es adecuado para realizar la evaluación de diseños parciales en las etapas tempranas del desarrollo [55].

2.8 Proceso de Pruebas

2.8.1 Niveles de Pruebas

Las Pruebas de software son los procesos que permiten verificar y revelar la calidad de un producto de software.

A continuación detallamos algunos Niveles de pruebas

- Pruebas Unitarias
  Verifican la funcionalidad y estructura de cada componente individualmente una vez que ha sido codificado [12].

- Pruebas de Integración
  Verifican el correcto ensamblaje entre los distintos componentes una vez éstos han sido probados de forma unitaria, con especial atención a los interfaces, tanto internos como externos [12].

- Pruebas de Sistema
  Prueban a fondo el sistema, comprobando su funcionalidad e integridad globalmente, en un entorno lo más parecido posible al entorno final de producción [12].

- Pruebas de Implantación
  Comprueban el correcto funcionamiento del sistema dentro del entorno real de producción [12].

- Pruebas de Aceptación
  Verifican que el sistema cumple con todos los requisitos indicados y permite que que los usuarios del sistema den el visto bueno definitivo [12].

- Pruebas de Regresión
  Comprueba que los cambios sobre un componente del sistema, no introducen un comportamiento no deseado o errores adicionales en otros componentes no modificados [12].

- Pruebas de Valores de Frontera
  La prueba de Frontera o también conocido como *Boundary test*, es un método para diseñar casos de prueba que se enfocan en pruebas cerca de los valores límites permitidos para ver el comportamiento del sistema [12].

2.8.2 Métodos y Tipos de Pruebas

Funcional

El objetivo de este tipo de prueba es el determinar si una función específica trabaja como lo especificado [12].

Algorítmicos

El objetivo de este tipo de prueba es el determinar si un algoritmo ha sido correctamente implementado [12].

Positivos
El objetivo de este tipo de prueba es en determinar si los resultados de las pruebas son consistentes cuando se enfrenta con entradas adecuadas [12].

Negativos
El objetivo de este tipo de prueba es el determinar si el comportamiento del software es estable cuando se enfrenta con entradas inválidas o acciones con operadores inesperados [12].

Usables
Se enfoca en el usuario, a través de un conjunto de actividades y técnicas, cuyo objetivo es crear un sistema usable para usuarios expertos ó inexpertos [12].

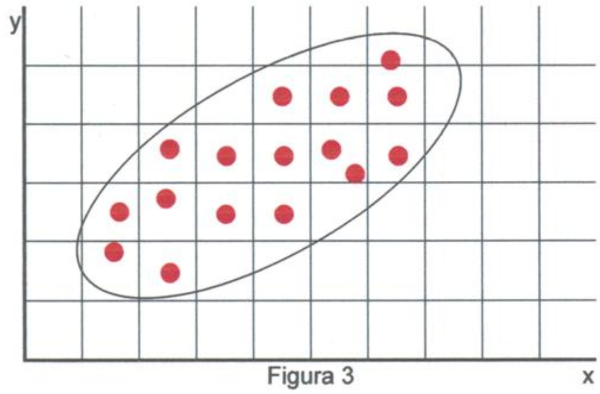
2.9 Herramientas de interpretación de datos

Los datos que son recopilados para visualizar el proceso de desarrollo o investigar causas asignables y potenciales mejoras, a menudo necesitan ser ordenados para poder entender la información obtenida. Esto significa organizar y resumir los datos y buscar patrones, tendencias y relaciones. Herramientas tales como diagramas de dispersión, diagramas causa efecto, histogramas, gráficas de barra son de gran ayuda. **[56]**

2.9.1 Diagramas de Dispersión.

Un diagrama de dispersión, como el que se muestra en la figura 2.9.1, es un gráfico de valores observados que muestra como una variable se ha comportado con respecto a otra. Estos diagramas son usados como un primer paso en la exploración de los datos, especialmente como parte de una búsqueda de relaciones causa-efecto. **[56]**

Figura 2.9.1. Ejemplo de un diagrama de dispersión [56]



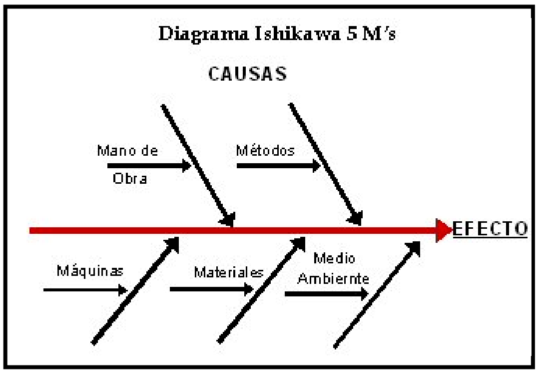
Estos diagramas de dispersión normalmente son utilizados para responder preguntas tales como: ¿El producto de la compañía A trabaja mejor que el de la compañía B? Cuando estos diagramas sugieren que una relación podría existir entre dos variables, su uso es a menudo seguido por métodos estadísticos más formales tales como el análisis exploratorio de datos o análisis de regresión. **[56]**

2.9.2 Diagramas causa-efecto.

Un diagrama causa-efecto es una muestra gráfica que es usada para probar y mostrar relaciones entre un problema (efecto) y sus posibles causas.

Cuando los diagramas causa-efecto son utilizados para explorar el entorno de un proceso, es más conveniente que los diagramas sean ensamblados por gente que trabaja actualmente en el proceso. **[56]**

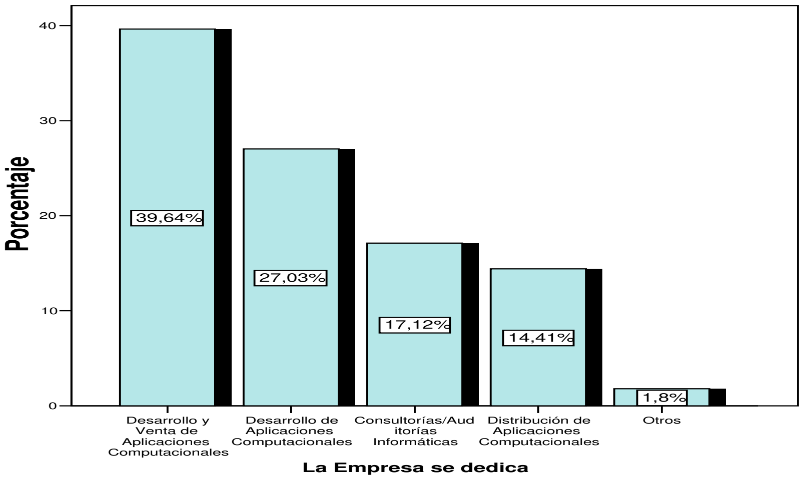
Figura 2.9.2 Ejemplo de un diagrama causa-efecto



2.9.3 Histogramas.

Un histograma toma las mediciones de datos y muestra la distribución de los valores observados. Son creados para agrupar los resultados de las mediciones en "celdas" y contar el número en cada celda. Las alturas de las barras en los histogramas son proporcionales al número de ocurrencias dentro de cada celda, como se muestra en la figura 2.9.3. **[56]**

Figura 2.9.3. Ejemplo de un histograma



Los histogramas son útiles para investigar y resumir el desempeño de un proceso con respecto a límites especificados que el proceso o sus productos deben satisfacer.

2.10 Análisis Univariado.

El análisis univariado en un sentido amplio, se refiere a todos los métodos estadísticos que analizan la distribución de una sola variable. Con el objeto de realizar este tipo de análisis se procederá a utilizar gráficas de barra para mostrar cual es el comportamiento promedio de ciertas variables de importancia. Para esto se obtendrá la media o promedio estándar de dichas variables. Otro método de análisis univariado es observar cuál es la composición porcentual de los elementos de dicha variable, para esto se utilizarán gráficas de barra que permitirán observar dicha composición. [57]

2.11 Análisis Multivariado.

Para este análisis se aplicará el análisis de correlación, el mismo que es un grupo de técnicas estadísticas empleado para medir la intensidad de la relación (correlación) entre dos variables. [57]

El principal objetivo del análisis de correlación consiste en determinar qué tan intensa es la relación entre dos variables. Una medida de esta relación es el coeficiente de correlación, el mismo que fue originado por el investigador Kart Pearson y describe la intensidad de la relación entre dos conjuntos de variables de nivel de intervalo o de nivel de razón. Ya que se le denota con r, con frecuencia se menciona también como ***r*** de Pearson o coeficiente de correlación producto-momento de Pearson. Puede tomar cualquier valor de -1 a +1, inclusive. Un coeficiente de correlación de -1 o de +1 indica correlación perfecta. Un valor calculado de -1 revela que la variable independiente X y la variable dependiente Y están perfectamente relacionadas en forma lineal negativa. **[57]**

CAPÍTULO III

3. DISEÑO DEL ESTUDIO

3.1 Hipótesis

Haciendo un énfasis en las investigaciones realizadas en el campo de la Ingeniería de Software, en la cual se puede observar la tendencia en cuanto a los avances en mejorar la calidad en el proceso de desarrollo; las Hipótesis a probar en el presente estudio son las siguientes:

- Las empresas ecuatorianas no poseen métodos de V&V establecidos para cada fase de desarrollo de Software.
- Las empresas ecuatorianas no poseen herramientas de V&V establecidos para cada fase de desarrollo de Software.
- Las empresas ecuatorianas no cuentan con técnicas de V&V establecidos para cada fase de desarrollo de Software.
- El uso de Herramientas de Distribución libre está relacionado con el tamaño de la empresa.
- El uso de Herramientas Propietarias está relacionado con el tipo de servicio que brindan.

3.2 Metodología para la recolección de la información

En el estudio se decidió realizar una encuesta para obtener la información necesaria, ésta provee de un mecanismo muy práctico y fácil de comprender.

Esta encuesta se la realizó de manera presencial; y las razones que nos llevaron a aplicarla de esta forma fueron:

- Garantizar un alto porcentaje de respuestas, eliminando el envío de encuestas por email que no ha dado resultados favorables en estudios anteriores.
- Obtener resultados claros y precisos mediante la explicación de cada pregunta en caso de ser necesario.

3.3 Población objetivo

Al empezar el proceso de selección de la población objetivo, se analizó en primer lugar la ubicación geográfica de las empresas, aquí se procedió a definir que nos enfocaríamos en empresas asentadas tanto en la ciudad de Quito como en la ciudad de Guayaquil. Esta decisión se la tomó en base a que en estudios anteriores [1-8] se ha demostrado que estas ciudades representan el mercado más grande de empresas desarrolladoras de software del país.

Posteriormente definimos que el estudio iba a ser realizado en el sector privado y que las características básicas que debían tener las empresas era que se dediquen al desarrollo y/o venta de software y que pertenezcan al mismo sector, que para nuestro estudio sería el sector de software. Esta característica se basó en que al final del estudio se iba a realizar una comparación de datos de las dos ciudades para así poder obtener conclusiones válidas, y para que los datos fueran correctos el estudio debería basarse en empresas con similares características, es decir que pertenecieran a un mismo sector.

Referente a la muestra para la aplicación del instrumento, inicialmente se contaba con una base de datos que fue facilitada por el VLIR – ESPOL, componente 8, la misma que involucraba un total de 160 empresas, ubicadas en las ciudades de Quito, Guayaquil y Cuenca, conteniendo información actualizada de todas las empresas que desarrollan software en el Ecuador.

De la base de datos en mención, se tomó una muestra de 20 empresas desarrolladoras de software, en el mes de Enero del 2008 en la ciudad de Guayaquil; decisión que responde al principio estadístico de que cuando se realizan estas pruebas, es suficiente tomar el 10% del universo de estudio por lo tanto el tamaño de la nuestra muestra es apropiada para poder llegar a conclusiones validas.

A efecto de cumplir con el tamaño de muestra calculado en 20 empresas, fue necesario visitar y/o llamar a un total de 55 empresas con el perfil de la población objetivo del estudio, además de proporcionar a las empresas una carta de colaboración para con el estudio para mas detalles ver el **Anexo 5**. Al término de este ejercicio los resultados de la indagación fueron que:

Ciudad de Guayaquil

Tabla 3.3.1 Detalles de la indagación en Guayaquil

Número de empresas	Resultado de la indagación
1	No desarrolla.
3	Manifestaron interés, pero al final

	nunca participaron
4	Quedaron pendientes de confirmar (nunca confirmaron)
3	No tienen proyecto para aplicar
6	No aceptaron participar (rechazo explícito)
18	Nunca se pudo localizar a la persona encargada
20	Aceptaron participar

A las 20 empresas que aceptaron la participación, se les dio a conocer el alcance de nuestro estudio y como ayudaría a la mejora de los procesos de desarrollo de software y más aspectos, sin embargo, 1 de las empresas no cumplió con el compromiso adquirido.

En lo referente a la ciudad de Quito se consideró un tamaño mucho mayor ya que se trata de un universo de empresas de aproximadamente 98 empresas, para lo cual consideramos un tamaño de la muestra de 38 empresas, por lo que fue necesario visitar y/o llamar a un total de 68 empresas con el perfil de la población objetivo del estudio. Al término de este ejercicio los resultados de la indagación fueron que:

Ciudad de Quito

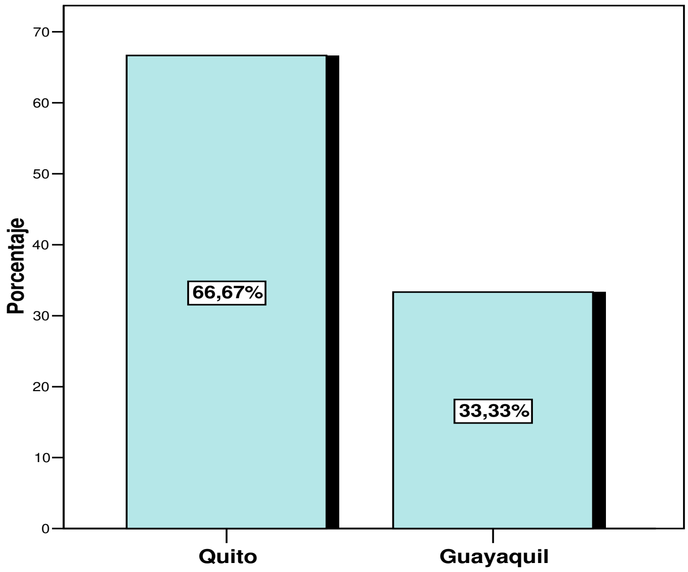
Tabla 3.3.2 Detalles de la indagación en Quito

Número de empresas	Resultado de la indagación
3	No desarrolla.
6	Manifestaron interés, pero al final nunca participaron

4	Quedaron pendientes de confirmar (nunca confirmaron)
2	No tienen proyecto para aplicar
4	No aceptaron participar (rechazo explícito)
11	Nunca se pudo localizar a la persona encargada
38	Aceptaron participar

A las 38 empresas que aceptaron la participación, se les dio a conocer el alcance de nuestro estudio y como ayudaría a la mejora de los procesos de desarrollo de software y más aspectos, sin embargo, 3 de las empresas no cumplieron con el compromiso adquirido.

Figura 3.3.3 Porcentaje de empresas encuestadas por ciudad



3.4 Instrumento de medición

Primero cabe decir que el instrumento de medición es la herramienta fundamental para poder alcanzar los objetivos propuestos. Como hemos mencionado en el numeral 3.2 se utilizó la encuesta.

Este instrumento de medición fue escogido para este estudio por las siguientes razones: **[58]**

- Proporciona mayor rapidez en cuanto a la obtención de resultados.
- Brinda una gran capacidad para estandarización de datos, lo que nos permite realizar un tratamiento informático y análisis estadístico.
- Permite obtener información de cualquier tipo de población.

El tipo de encuesta que se ha escogido es la de encuesta por muestreo, que consiste en elegir a una parte representativa de la población objetivo y trabajar con esta porción.

La estructura de la encuesta se la realizó de una manera segmentada para que sea de fácil comprensión por parte del encuestado ya que estaba conformada de preguntas relacionadas con Métodos, Técnicas y Herramientas de Verificación y Validación. A continuación se listan las secciones en que fue dividida la encuesta:

Sección I: Información General

Sección II: Información Entrevistado

Sección III: Métodos

Sección IV: Técnicas

Sección V: Niveles de Pruebas

Sección VI: Herramientas

La encuesta que se aplicó en este estudio puede ser consultada en el **ANEXO 2**

3.5 Plan de levantamiento de información

Lo primero que tuvimos que definir fue un banco de preguntas que iban a conformar un borrador de la encuesta, para luego elaborar una primera versión de la misma; a continuación se hicieron las correcciones en base a criterios de personas especializadas en el área de Verificación y Validación, generando la versión final de nuestra encuesta.

Una vez definida la encuesta, se procedió a la realización del plan piloto, que consistió en encuestar al menos a cinco empresas, y una vez hecho esto, proceder a realizar el respectivo análisis estadístico. Esto se realizó para comprobar que la encuesta nos iba a proporcionar resultados válidos ver **Anexo1**.

El plan piloto fue ejecutado en la ciudad de Guayaquil, y del mismo se obtuvieron resultados favorables para el estudio.

Una vez finalizado el plan piloto, se prosiguió con la encuesta al resto de la muestra, enfocándose ahora en la ciudad de Quito, una vez confirmadas las citas con las empresas que nos colaboraron, proseguimos a realizar la encuesta con el tamaño de la muestra establecido en el **punto 3.1**

Finalmente se terminó de realizar las encuestas a las demás empresas que formaban parte de la muestra en la ciudad de Guayaquil.

A todas las empresas encuestadas se les proporcionó la encuesta de forma impresa para que sea fácil de llenar.

CAPÍTULO IV

4 ANÁLISIS Y PRESENTACIÓN DE RESULTADOS DEL ESTUDIO

4.1 Presentación de estadísticas

Una vez culminada la fase de recolección de información, se realizó el análisis estadístico de los datos con el fin de obtener información relevante para el contexto del estudio.

Para el análisis estadístico se utilizó el software SPSS por las herramientas que nos brinda este paquete estadístico. Para mayor detalle del Software ver el **Anexo 3**

A continuación presentamos los resultados de las preguntas que nos permitieron comprobar las hipótesis planteadas, los resultados de las demás preguntas se puede observar en el **Anexo 4**

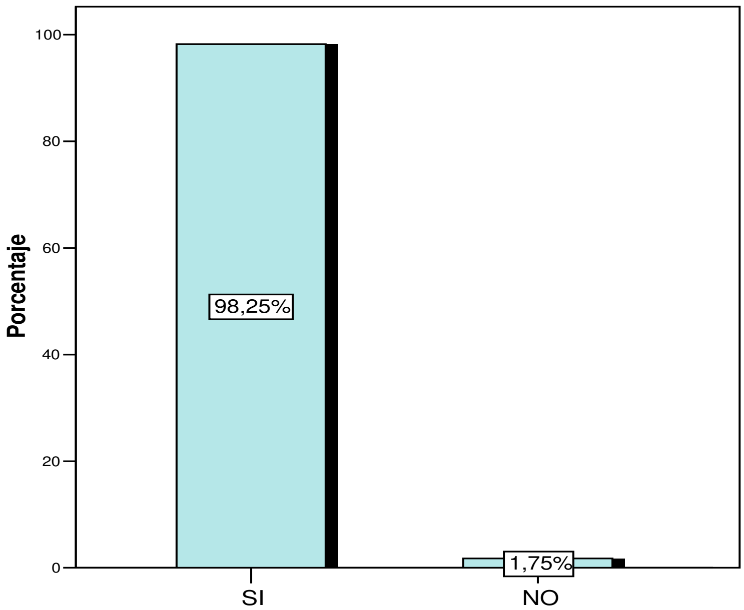
Sección III: **Técnicas**

1.- ¿En su empresa utilizan técnicas de V&V previamente establecidos para llevar a cabo el desarrollo de un proyecto?

a. Si **b. No**

Del total de empresas encuestadas, el 98.25% afirmó que utilizan técnicas de V&V en su empresa, mientras que un 1.75% contestó que no utilizan técnicas de V&V en su empresa. Cabe recalcar que las empresas que no utilizan técnicas de V&V para el proceso de desarrollo de software nos comentaron que a pesar de que en este momento no usaban técnica establecida, estaban revisando la posibilidad de establecer en los próximos meses la utilización de una o varias técnicas que se adapte a su proceso de Negocio ya que sabían de la importancia de las mismas. A continuación se muestra una gráfica estadística de esta pregunta (Figura 4.1.1).

Figura 4.1.1. Estadísticas de la pregunta 1 de la encuesta



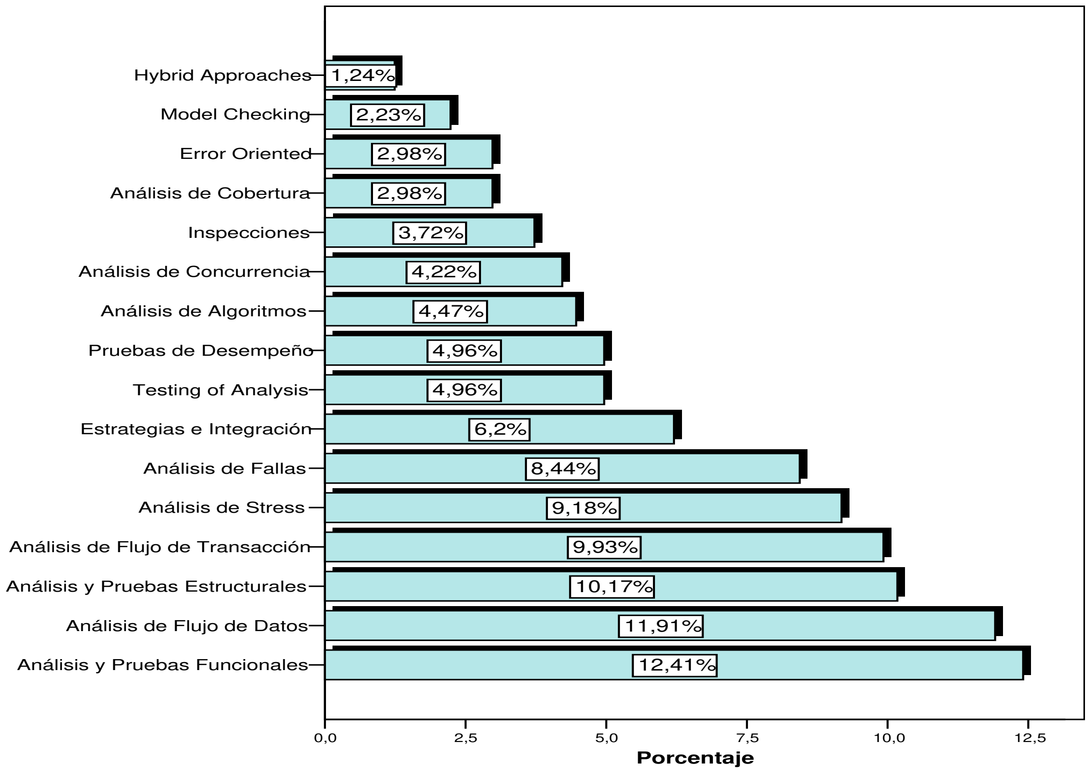
2.- Indique el tipo de Técnicas que utilizan en la empresa

a.- Análisis y Pruebas Funcionales
b.- Análisis y Pruebas Estructurales
c.- Error-Oriented
d.- Testing and Análisis
e.- Estrategias de Integración
f.- Hybrid Approaches
g.- Análisis de Flujo de Transacción
h.- Análisis de Stress
i.- Análisis de Falla
j.- Análisis de Concurrencia
k.- Análisis de Algoritmos
l.- Análisis de Cobertura
m.- Análisis de Flujo de Datos
n.- Inspecciones
ñ.- Pruebas de Desempeño
o.- ModelChecking
p.- Otro

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera, el 12.41% respondió Análisis y Pruebas Funcionales, el 11.91% Análisis de Flujo de Datos, el 10.17% Análisis y Pruebas Estructurales, el 9.93% Análisis de Flujo de Transacción, el 9.18% Análisis de Stress, el 8.44% Análisis de Falla. El resto de las técnicas obtuvieron valores porcentuales pequeños menores al 6% .

A continuación se muestra una gráfica estadística de esta pregunta (Figura 4.1.2).

Figura 4.1.2. Estadísticas de la pregunta 2 sección III de la encuesta

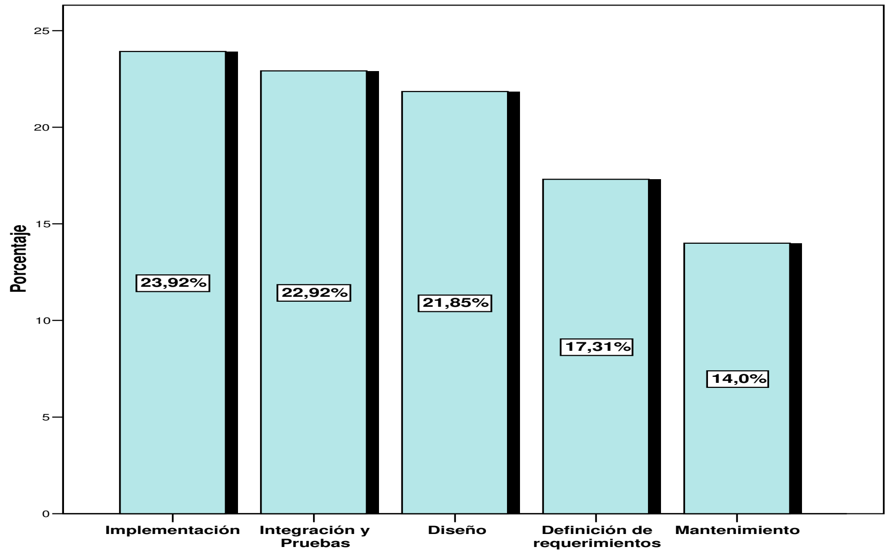


3. De las técnicas seleccionadas en la pregunta anterior señale el tipo de esfuerzo que se aplica en cada fase de desarrollo de la siguiente tabla, donde M es mucho, P es poco y N es ninguno.

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera, el 23.92% de esfuerzo en la fase de implementación, el 22.92% en Integración de Pruebas, el 21.86% Diseño, el 17.31 % Definición de requerimientos, el 14.00 % Mantenimiento.

A continuación se muestra una gráfica estadística de esta pregunta (Figura 4.1.3).

Figura 4.1.3. Estadísticas de la pregunta 3 sección III de la encuesta



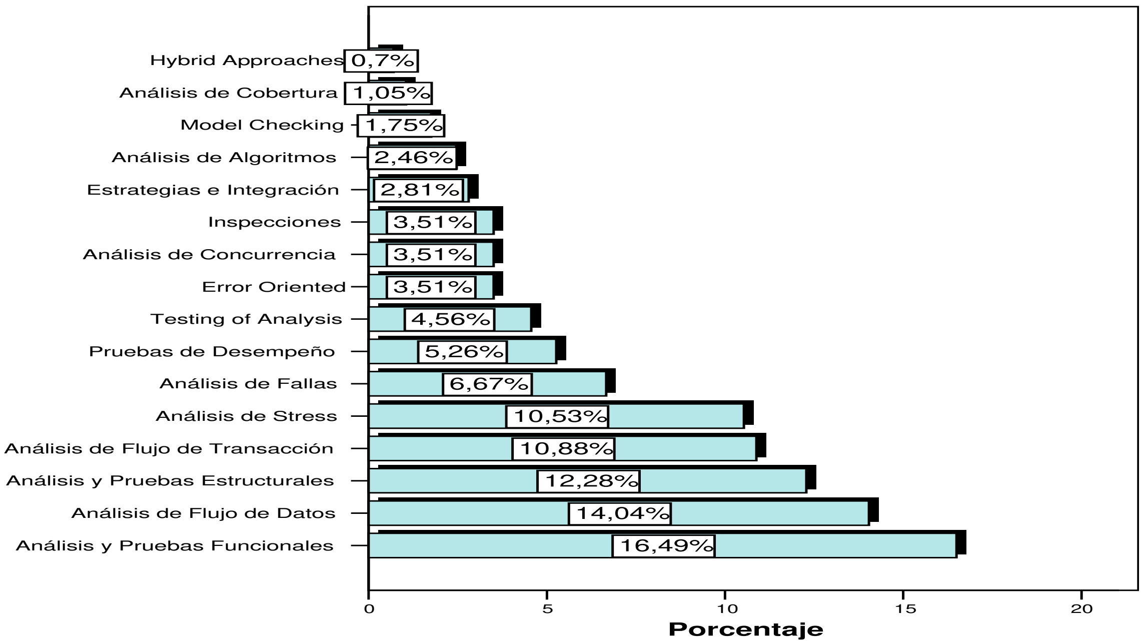
4. ¿Según su criterio, cuales Técnicas cree que aportan más en la Calidad del producto de software final?

a.- Análisis y Pruebas Funcionales
b.- Análisis y Pruebas Estructurales
c.- Error-Oriented
d.- Testing and Análisis
e.- Estrategias de Integración
f.- Hybrid Approaches
g.- Análisis de Flujo de Transacción
h.- Análisis de Stress
i.- Análisis de Falla
j.- Análisis de Concurrencia
k.- Análisis de Algoritmos
l.- Análisis de Cobertura
m.- Análisis de Flujo de Datos
n.- Inspecciones
ñ.- Pruebas de Desempeño
o.- Model Checking
p.- Otro

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera, el 16.49% Análisis y Pruebas Funcionales, el 14.04% Análisis de Flujo de Datos, el 12.28% Análisis y Pruebas Estructurales, el 10.88% Análisis de Flujo de Transacción, el 10.53% Análisis de Stress, el 6.67% Análisis de Falla. El resto de las técnicas obtuvieron valores porcentuales pequeños menores al 6%.

A continuación se muestra una gráfica estadística de esta pregunta (Figura 4.1.4).

Figura 4.1.4. Estadísticas de la pregunta 4 sección III de la encuesta



Sección IV: Métodos

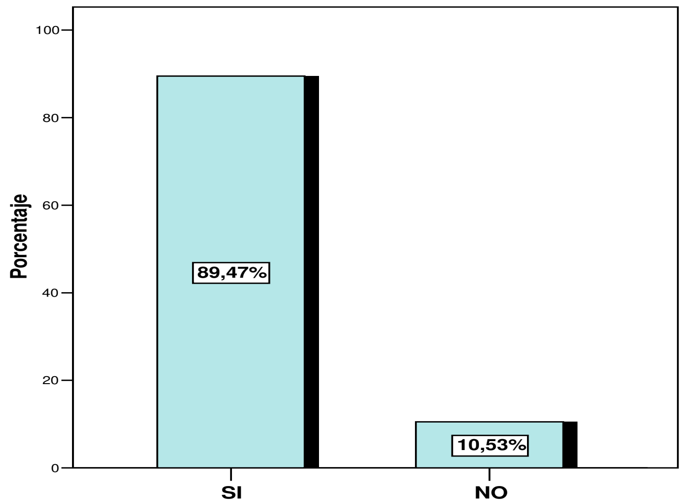
1.- ¿En su empresa utilizan métodos de V&V previamente establecidos para llevar a cabo el desarrollo de un proyecto?

a. Si **b. No**

Del total de empresas encuestadas, el 89.47 respondió que utilizan métodos de V&V en su empresa, mientras que un 10.53% contestó que no utilizan métodos de V&V en su empresa. Cabe recalcar que las empresas que no utilizan técnicas de V&V para el proceso de desarrollo de software nos comentaron que si les gustaría utilizar alguna de las técnicas.

A continuación se muestra una gráfica estadística de esta pregunta (Figura 4.1.5).

Figura 4.1.5. Estadísticas de la pregunta 1 sección IV de la encuesta



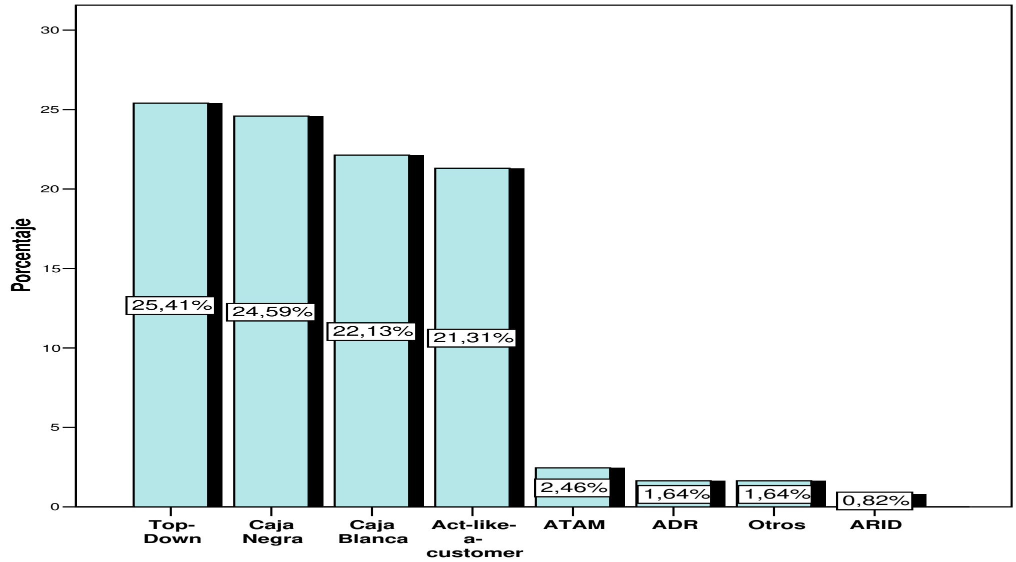
2.- Indique el tipo de Métodos que utilizan en la empresa:

a.- Caja Negra
b.- Caja Blanca
c.- Top-Down
d.- Act-like-a-customer
e.- ATAM
f.- ADR
g.- ARID
h.- Otro

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera, el 25.41% respondió que el método que utilizan es Top-Down, el 24.59% respondió que el método que utilizan es Caja Negra, el 22.13% respondió que el método que utilizan es Caja Blanca, el 21.31 respondió que el método que utilizan es Act- like a Customer, el

6.56% restante esta distribuido con los siguientes métodos como: ATAM, ADR, ARID y Otros como se detalla en el Figura 4.1.6

Figura 4.1.6. Estadísticas de la pregunta 2 sección IV de la encuesta

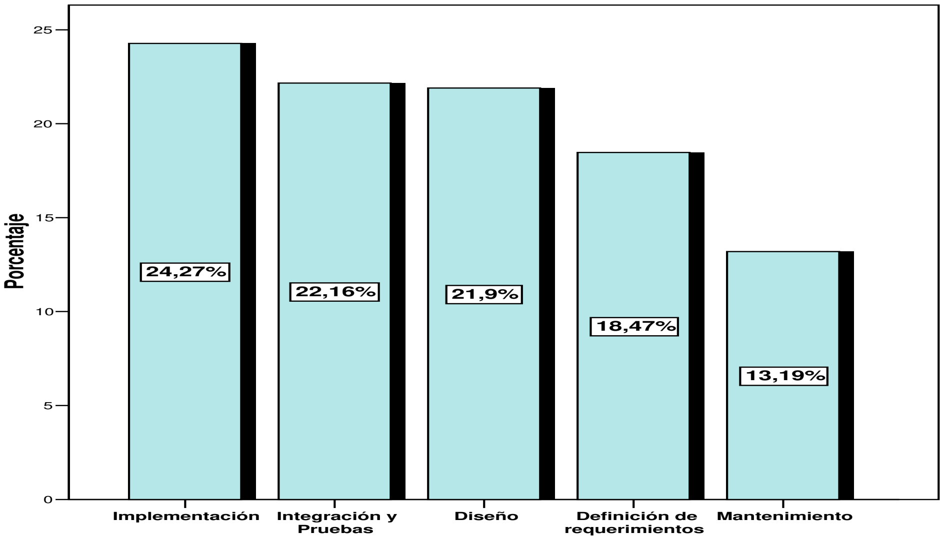


3.- De los métodos seleccionados en la pregunta anterior señale el tipo de esfuerzo que se aplica en cada fase de desarrollo de la siguiente tabla donde M es mucho, P es poco y N es ninguno

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera, el 24.27% de esfuerzo en la fase de Implementación, el 22.18% en Integración de Pruebas, el 21.9% Diseño, el 18.47 % Definición de requerimientos, el 13.19 % Mantenimiento.

A continuación se muestra una gráfica estadística de esta pregunta (Figura 4.1.7).

Figura 4.1.7 Estadísticas de la pregunta 3 sección IV de la encuesta



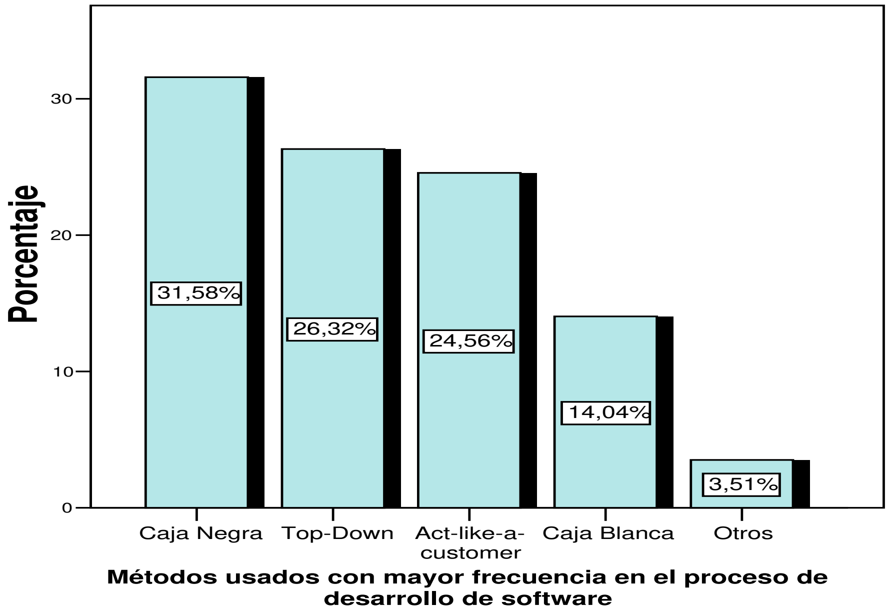
4.- De los métodos anteriormente nombrados cual es usado de manera mas frecuente en el proceso de desarrollo de Software.

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera, 31.58% respondió que el método mas frecuentemente usado es Caja Negra, el 26.32% Top-Down, el 24.56% Act-like-a -Customer, el 14.04% Caja Blanca, en cambio el 3.51% respondieron Otros métodos.

Entre los métodos que se encuentran en esta categoría podemos citar los siguientes: GESTOR y Control Técnico, los dos casos han sido métodos desarrollados por la misma empresa para lo cual sus resultados han cumplido sus expectativas.

A continuación se muestra una gráfica estadística de esta pregunta (Figura 4.1.8).

Figura 4.1.8. Estadísticas de la pregunta 4 sección IV de la encuesta



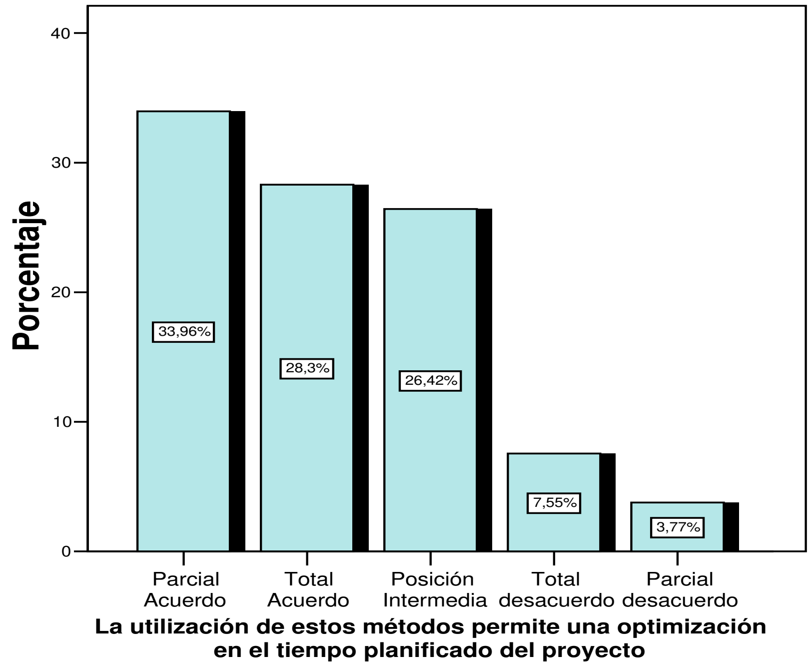
5.- Según su criterio la utilización de estos métodos permite una optimización en el tiempo planificado del proyecto

a. Totalmente Desacuerdo ( )
b. Parcial Desacuerdo ( )
c. Posición Intermedia ( )
d. Parcial Acuerdo ( )
e. Total Acuerdo ( )

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera, 33.96% respondió que la utilización de métodos les permite una optimización en el tiempo planificado fue Parcial Acuerdo, el 28.3% Total Acuerdo, el 26.42% Posición Intermedia, el 7.55% Total desacuerdo, y el 3.77% Parcial Desacuerdo.

A continuación se muestra una gráfica estadística de esta pregunta (Figura 4.1.9).

Figura 4.1.9 Estadísticas de la pregunta 5 sección IV de la encuesta



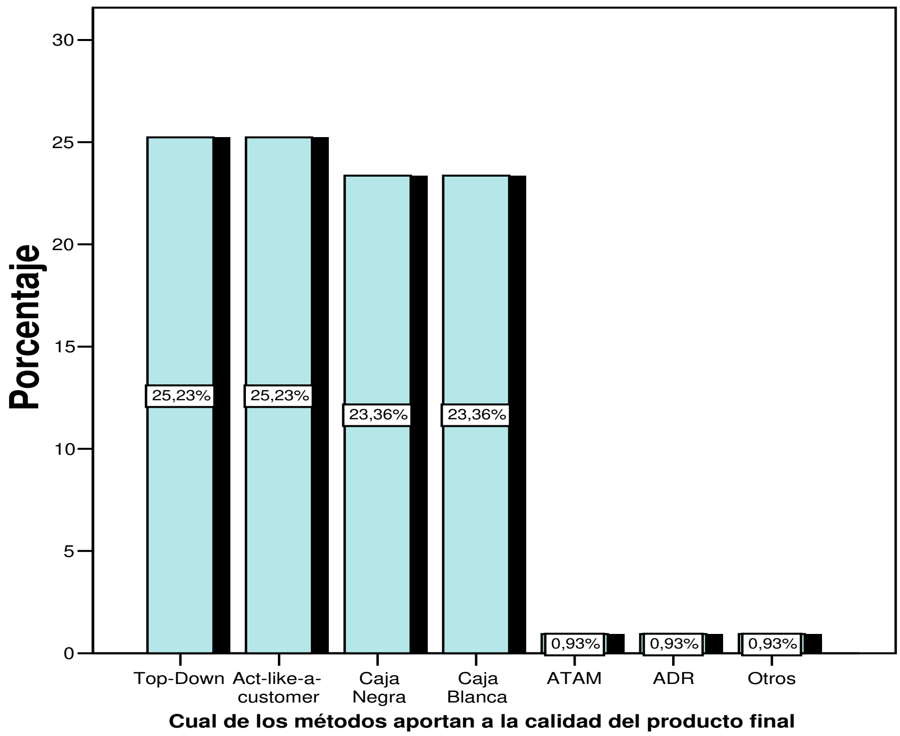
6.- ¿Según su criterio cuales de estos métodos aportan más en la Calidad del producto final?

a.- Caja Negra
b.- Caja Blanca
c.- Top-Down
d.- Act-like-a-customer
e.- ATAM
f.- ADR
g.- ARID
h.- Otro
Indique:__________

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera 25.23% indicaron que los métodos que aportan mas a la calidad del software es Top-Down y Act-like a Customer, el 23.36% Caja Negra y Caja Blanca y con el 0.93% ATAM, ARID y Otros

A continuación se muestra una gráfica estadística de esta pregunta (Figura 4.1.10).

Figura 4.1.10. Estadísticas de la pregunta 6 sección IV de la encuesta



Sección V: Herramientas

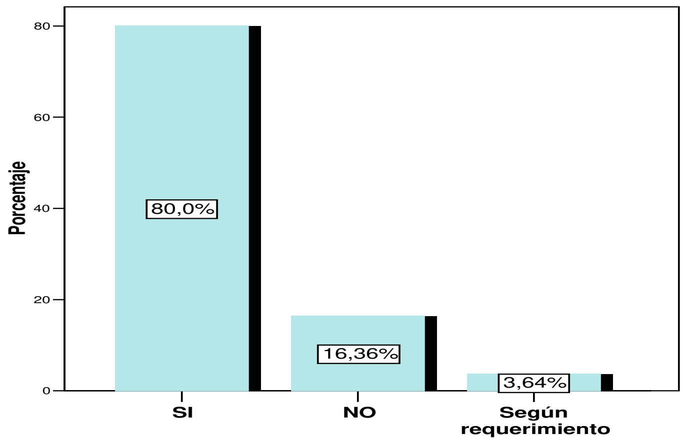
1. En su empresa utilizan Herramientas de V&V previamente establecidas para llevar a cabo el desarrollo de un proyecto?

a. Si b.No

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera 80.0 % indicaron que utilizan Herramientas de V&V previamente establecidas; el 16.36 % que no las utilizan y el 3.64% que según requerimiento.

A continuación se muestra una gráfica estadística de esta pregunta (Figura 4.1.11).

Figura 4.1.11. Estadísticas de la pregunta 1 sección V de la encuesta



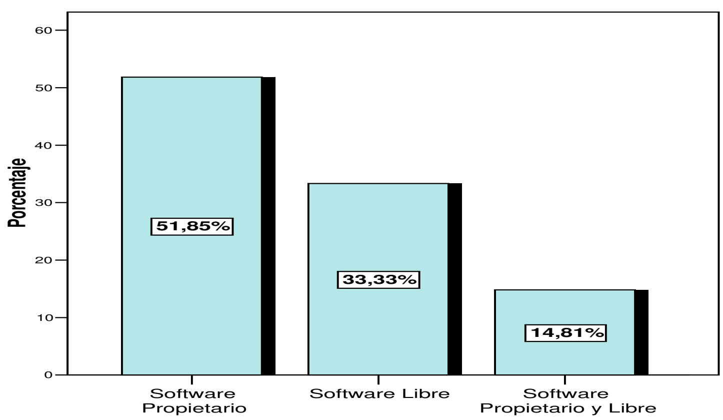
2. Que tipo de herramientas emplean su empresa?

a. Software Propietario b. Software Libre

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera 61.86 % indicaron que utilizan Software Propietario; 33.33% Software Libre y 14.81% Software Propietario y Libre.

A continuación se muestra una gráfica estadística de esta pregunta (Figura 4.1.12).

Figura 4.1.12. Estadísticas de la pregunta 2 sección V de la encuesta

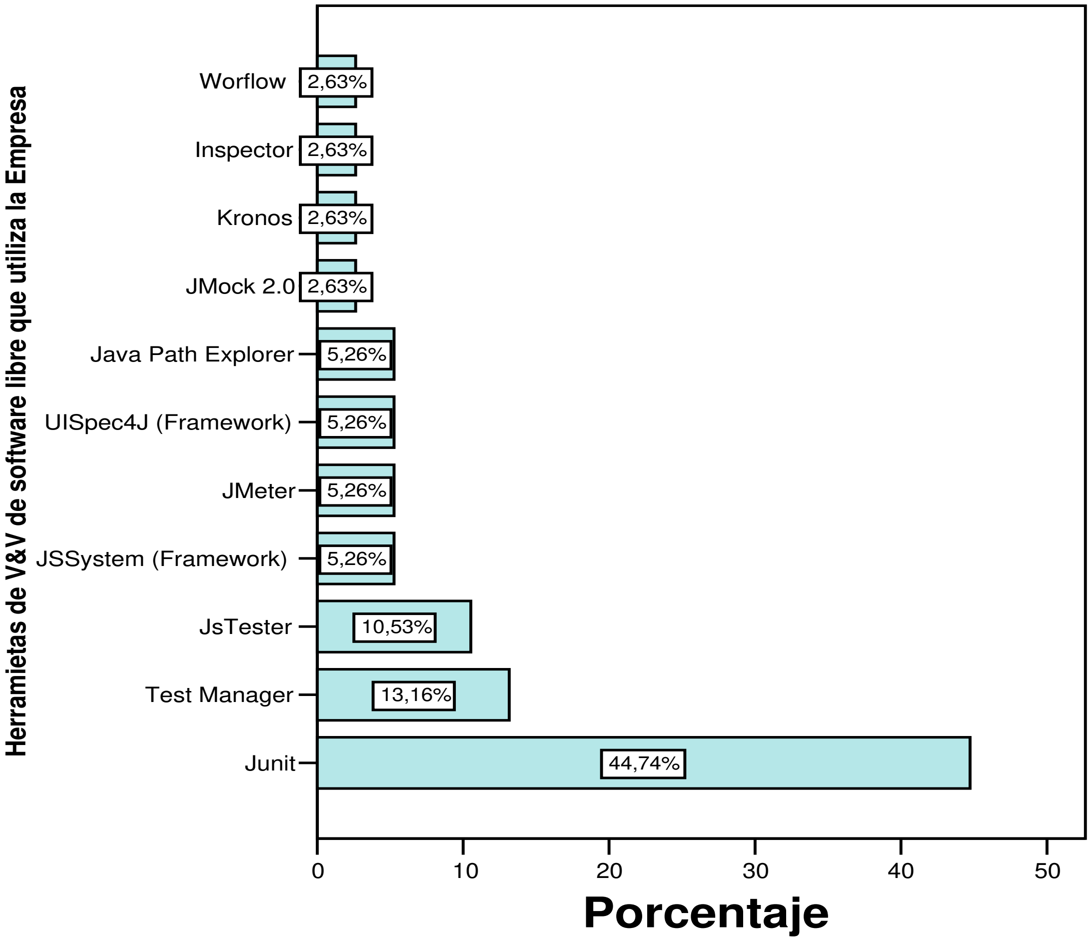


3. Seleccione las herramientas de V&V de software Libre que utilizan en su empresa. Si en su empresa no usan herramientas libres, por favor continúe con la pregunta 6 .

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera 44.74 % indicaron que utilizan Junit**; 13.16** Test Manager; 10.53% JsTester; 5.26% JsSystem (Framework), JMeter, UISpec4J (Framework) y Java Path Explorer; 2.63% JMock 2.0, Kronos, Inspector y Worflow.

A continuación se muestra una gráfica estadística de esta pregunta (Figura 4.1.13).

Figura 4.1.13. Estadísticas de la pregunta 3 sección V de la encuesta

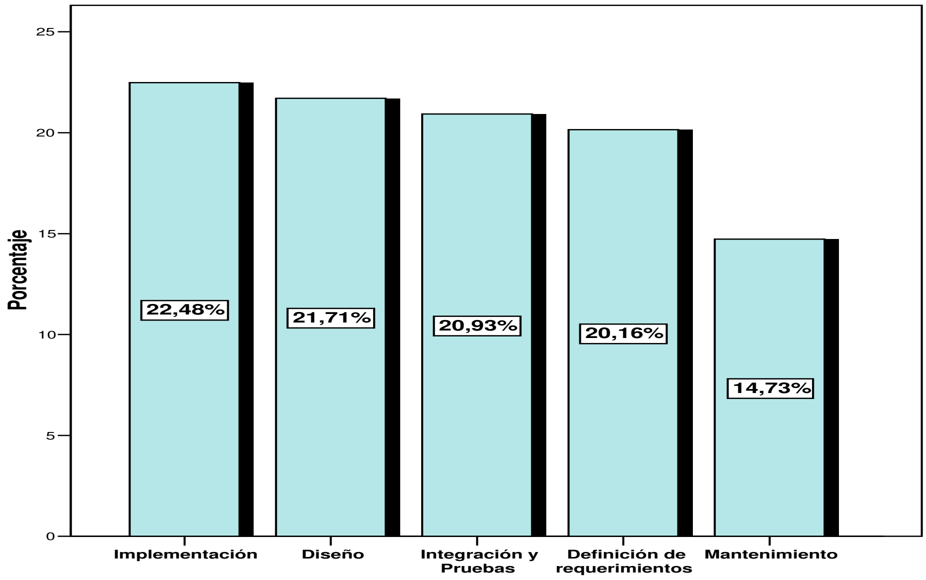


4.- De las herramientas seleccionadas en la pregunta anterior señale el tipo de esfuerzo que se aplica en cada fase de desarrollo de la siguiente tabla donde M es mucho, P es poco y N es ninguno.

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera, el 22.48% de esfuerzo en la fase de Implementación, el 21.7% en Diseño, el 20.93% Integración y Pruebas, el 20.16% Definición de requerimientos, el 14.73% Mantenimiento.

A continuación se muestra una gráfica estadística de esta pregunta (Figura 4.1.14).

Figura 4.1.14. Estadísticas de la pregunta 4 sección V de la encuesta



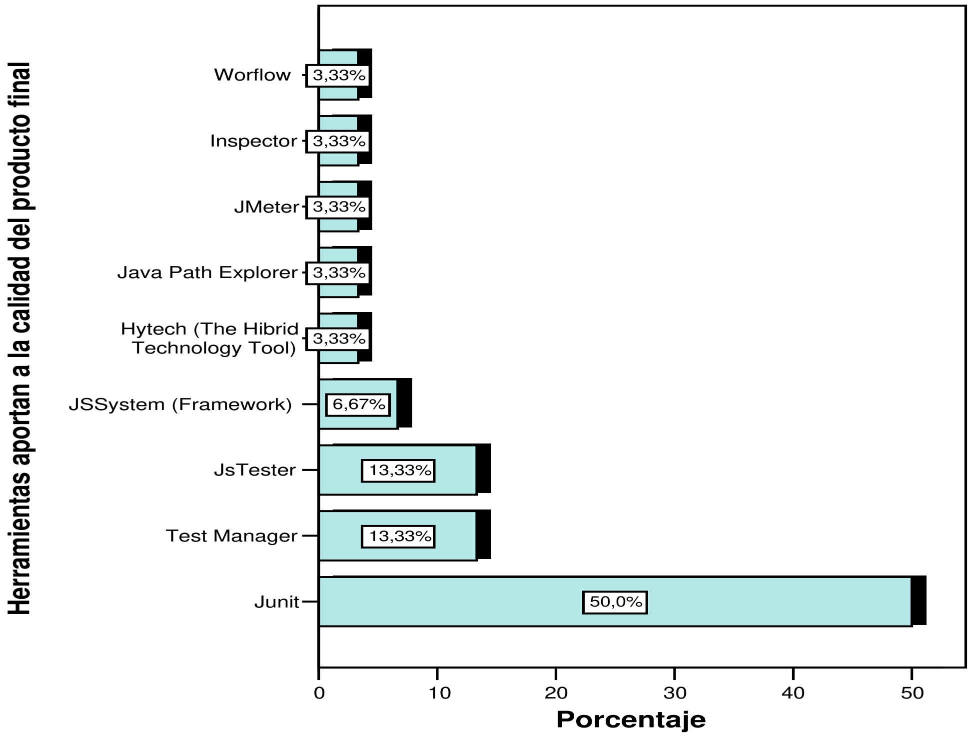
5.- Cuales Herramientas cree a su criterio que aportan más en la Calidad del producto final

a.- UISpec4J (Framework)
b.- JUnit
c.- TestManager
d.- Java Path Explorer
e.- JMock 2.0
f.- Selenium
g.- JsTester
h.- JSystem (Framework)
i.- DDTUnit (Framework)
j.- Jameleon
k.- T2
l.- Kronos
m.- HyTech (The HibridTechnology Tool)
n.- SMV (Symbolic ModelVerifier)
o.- Spin
p.- UPPAAL
q.- Otro
Indique:______________________

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera 50.0 % indicaron que Junit; Test Manager y JsTester el 13.33 %; JsSystem (Framework) 6.67 %; HyTech (The HibridTechnology Tool), Java Path Explorer, JMeter, Inspector y Worflow con el 3.33%.

A continuación se muestra una gráfica estadística de esta pregunta (Figura 4.1.15).

Figura 4.1.15. Estadísticas de la pregunta 5 sección V de la encuesta



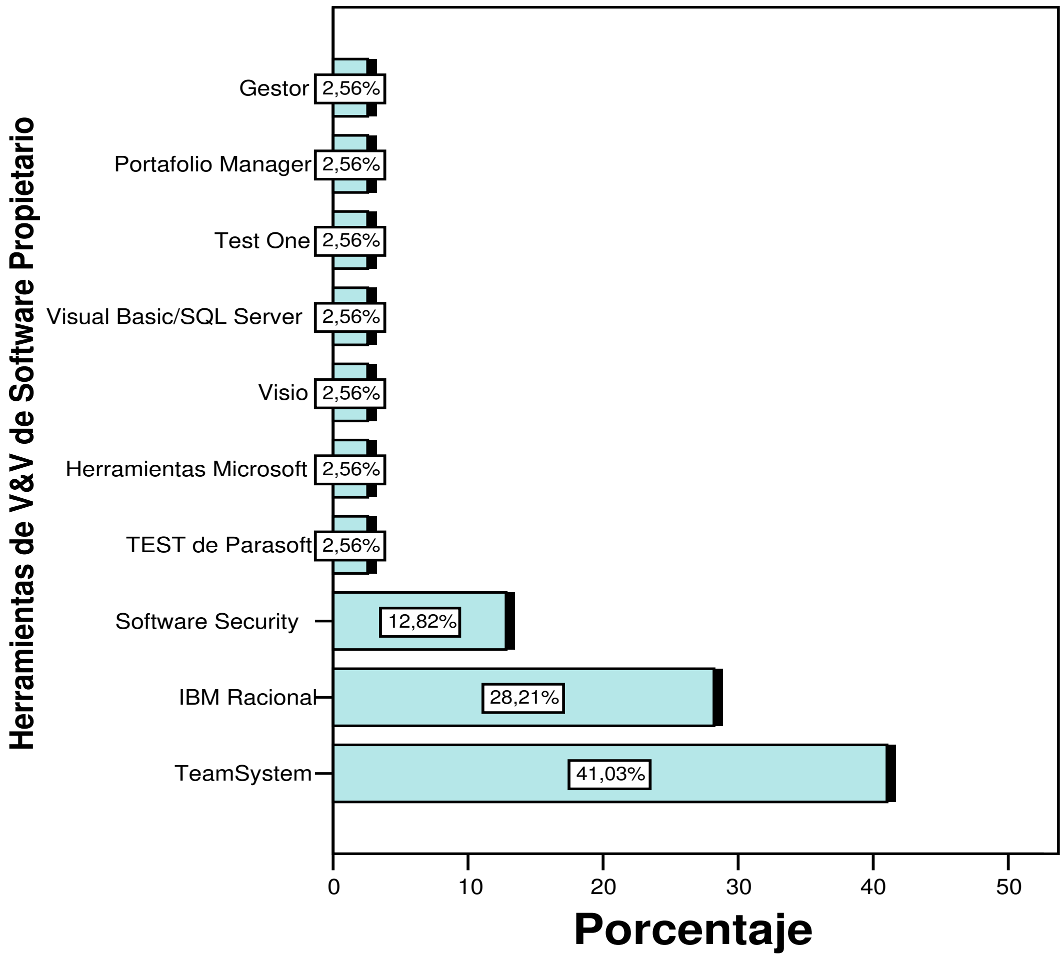
6 Seleccione las herramientas de V&V de software Propietario que utiliza en su empresa. Si en su empresa no usan herramientas de V&V de software propietario, pase a la siguiente sección

a.- Team System
b.- checKing de ALS
c.- Open Load Tester
d.- QACenter de Compuware
e.- Security Tester de Fortify
f.- SOATest de Parasoft
g.- Software Security
h.- Manager de Fortify
i.- .TEST de Parasoft

j.-Webking de Parasoft
k.- IBM Racional
l.-Otro
Indique:____________________

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera 41.03 % indicaron que utilizan Team System; 28.21 IBM Rational; 12.18% Software Security; 17.92 corresponden a otras herramientas que se detallan en la Figura 4.1.16.

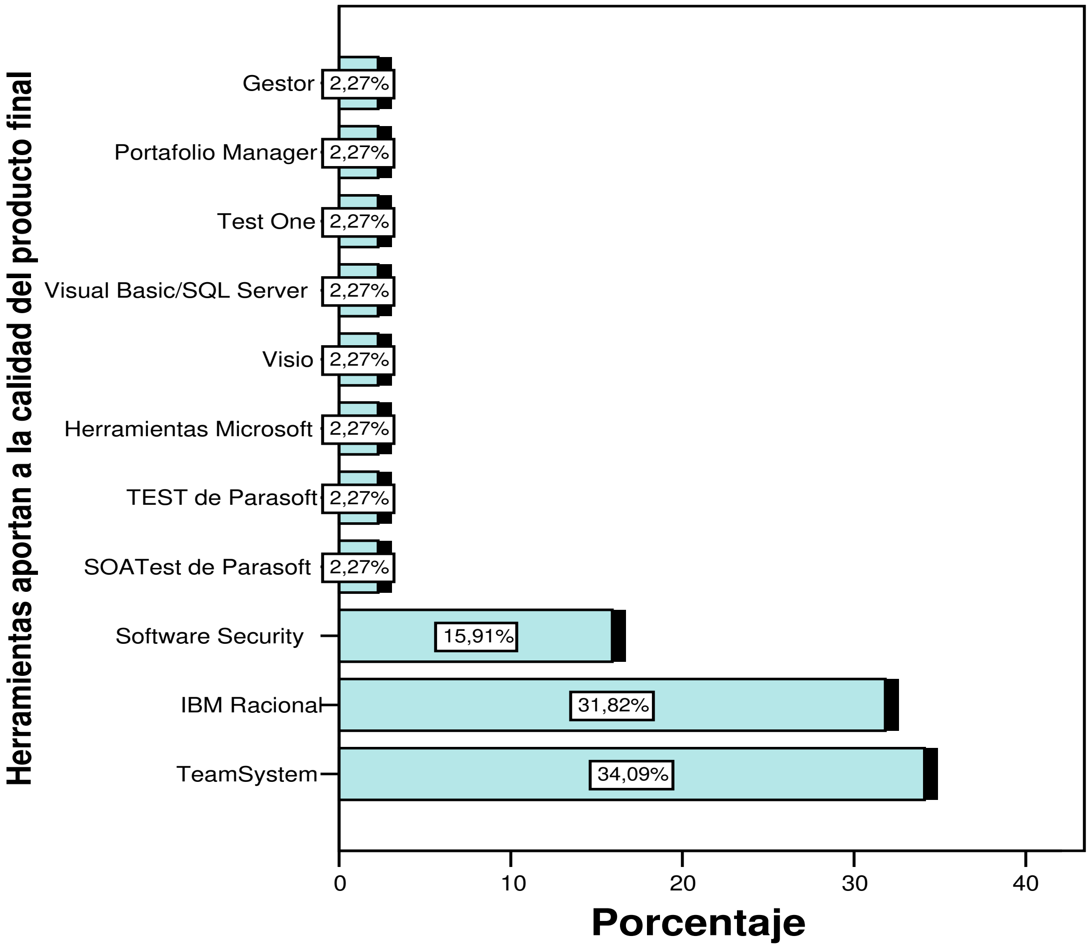
Figura 4.1.16. Estadísticas de la pregunta 6 sección V de la encuesta



8.- ¿Cuáles de estas herramientas cree a su criterio que aportan más a la calidad del producto Final?

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera 34.09 % indicaron que utilizan Team System; 31.82 IBM Rational; 15.91% Software Security; 18.16 corresponden a otras herramientas que se detallan en la Figura 4.1.17.

Figura 4.1.17. Estadísticas de la pregunta 8 sección V de la encuesta



4.2 Pruebas de verificación de hipótesis

En el presente apartado procedemos a verificar las hipótesis, las mismas que fueron detallas en el Capítulo 3 Sección 3.1, para el efecto se ha utilizado el fundamento teórico detallado en el Capítulo 2 Sección 2.9, Sección 2.10 y Sección 2.11.

Prueba de la hipótesis 1

La primera hipótesis a demostrar en el presente estudio responde al planteamiento hipotético referido en el Capitulo 3, que textualmente dice:

"Las empresas ecuatorianas no poseen métodos de V&V establecidos para cada fase de desarrollo de Software."

Donde se plantea como Hipótesis Nula H_0

H_0: Métodos que utiliza la empresa es independiente al tipo de esfuerzo que emplean en cada fase de desarrollo

Vs

La Hipótesis alternativa H_1

H_1: Métodos que utiliza la empresa y el tipo de esfuerzo que emplean no son independientes

Pruebas de chi-cuadrado

Tabla 4.2.1 Tabla de resultados de Hipótesis 1

	Valor	Gl	Sig. asintótica (bilateral)
Chi-cuadrado de Pearson	5,390(a)	4	,250
Razón de verosimilitud	5,408	4	,248
Asociación lineal por lineal	3,729	1	,053
N de casos válidos	57		

En la Tabla notamos que el p-valor es de 0,250; lo que vemos que a nivel de significación de 0,05, se debe de aceptar la hipótesis nula de independencia entre estas variables. Por lo tanto el método que utiliza la empresa es independiente del tipo de esfuerzo que emplean.

Prueba de la hipótesis 2

La segunda hipótesis a demostrar en el presente estudio responde al planteamiento hipotético referido en el Capitulo 3, que textualmente dice:

“Las empresas ecuatorianas no poseen herramientas de V&V establecidos para cada fase de desarrollo de Software”

Donde se plantea como Hipótesis Nula H_0

H_0: Herramientas que utiliza la empresa es independiente al tipo de esfuerzo que emplean en cada fase de desarrollo

Vs

La Hipótesis alternativa H_1

H_1**:** Herramientas que utiliza la empresa y el tipo de esfuerzo que emplean no son independientes

Pruebas de chi-cuadrado

Tabla 4.2.2 Tabla de resultados de Hipótesis 2

	Valor	gl	Sig. asintótica (bilateral)
Chi-cuadrado de Pearson	4,138(a)	4	,388
Razón de verosimilitud	4,158	4	,385
Asociación lineal por lineal	1,536	1	,215

N de casos válidos	57		

En la Tabla notamos que el p-valor es de 0,388; lo que vemos que a nivel de significación de 0,05, se debe de aceptar la hipótesis nula de independencia entre estas variables. Por lo tanto la herramienta que utiliza la empresa es independiente del tipo de esfuerzo que emplean.

Prueba de la hipótesis 3

La tercera hipótesis a demostrar en el presente estudio responde al planteamiento hipotético referido en el Capitulo 3, que textualmente dice:

"Las empresas ecuatorianas no cuentan con técnicas de V&V establecidos para cada fase de desarrollo de Software"

Donde se plantea como Hipótesis Nula H_0

H_0: Técnicas que utiliza la empresa es independiente al tipo de esfuerzo que emplean en cada fase de desarrollo

Vs

La Hipótesis alternativa H_1

H_1: Técnicas que utiliza la empresa y el tipo de esfuerzo que emplean no son independientes

Pruebas de chi-cuadrado

Tabla 4.2.3 Tabla de resultados de Hipótesis 3

	Valor	gl	Sig. asintótica (bilateral)

Chi-cuadrado de Pearson	1,554(a)	2	,460
Razón de verosimilitud	1,556	2	,459
Asociación lineal por lineal	1,537	1	,215
N de casos válidos	57		

En la Tabla notamos que el p-valor es de 0,460, lo que vemos que a nivel de significación de 0,05, se debe de aceptar la hipótesis nula de independencia entre estas variables. Por lo tanto la técnica que utiliza la empresa es independiente del tipo de esfuerzo que emplean.

Prueba de la hipótesis 4

La cuarta hipótesis a demostrar en el presente estudio responde al planteamiento hipotético referido en el Capitulo 3, que textualmente dice:

"El uso de Herramientas de Distribución libre está relacionado con el tamaño de la empresa"

Donde se plantea como Hipótesis Nula H_0

H_0**:** Tamaño de la empresa es independiente del tipo de herramienta que utiliza

Vs

La Hipótesis alternativa H_1

H_1**:** Tamaño de la empresa y la herramienta que utiliza no son independientes

Pruebas de Chi-cuadrado

Tabla 4.2.4 Tabla de resultados de Hipótesis 4

	Valor	gl	Sig. asintótica

			(bilateral)
Chi-cuadrado de Pearson	13,087(a)	6	,042
Razón de verosimilitud	14,316	6	,026
Asociación lineal por lineal	,341	1	,559
Número de casos válidos	57		

En la Tabla notamos que el p-valor es de 0,042, lo que vemos que a nivel de significación de 0,05, se debe de rechazar la hipótesis nula de independencia entre estas variables. Por lo tanto el tamaño de la empresa depende del tipo de herramienta que utiliza la empresa.

Prueba de la hipótesis 5

La quinta hipótesis a demostrar en el presente estudio responde al planteamiento hipotético referido en el Capitulo 3, que textualmente dice:

"El uso de Herramientas Propietarias esta relacionado con el tipo de servicio que brindan"

Donde se plantea como Hipótesis Nula H_0

H_0 **:** Tipo de herramientas que emplea la empresa es independiente del servicio que brindan

Vs

La Hipótesis alternativa H_1

H_1 **:** Tipo de herramientas que emplea la empresa y el servicio que brindan no son independientes

Pruebas de Chi-cuadrado

Tabla 4.2.5 Tabla de resultados de Hipótesis 5

	Valor	gl	Sig. asintótica (bilateral)
Chi-cuadrado de Pearson	7,978(a)	12	,787
Razón de verosimilitud	11,855	12	,457
Asociacion lineal por lineal	1,308	1	,253
N. de casos válidos	57		

En la Tabla notamos que el p-valor es de 0,787, lo que vemos que a nivel de significación de 0,05, se debe de aceptar la hipótesis nula de independencia entre estas variables. Por lo tanto el tipo de herramientas que emplea la empresa es independiente del servicio que brinda.

4.3 Análisis de indicadores obtenidos

Hipótesis 1

Según los resultados obtenidos con los indicadores de la Prueba de hipótesis los métodos de V&V que utilizan las empresas son independientes del tipo de esfuerzo que emplean; esta independencia demuestra que las empresas si poseen métodos, pero estos no se ven reflejados en cada fase del proceso de desarrollo de software.

Para esta primera hipótesis la conclusión a la que hemos llegado es que dicha hipótesis es verdadera, basados en los resultados de la pregunta 1 y de la pregunta 3 donde se establece si utilizan ó no Métodos de V&V y el grado de utilización del método en cada fase de desarrollo respectivamente. En estas 2 preguntas se observa que aunque la mayoría de empresas mencionó que si utilizan métodos de V&V para el desarrollo de software; estos no se reflejan en las fases de desarrollo.

Otro punto que refuerza la conclusión obtenida es que los métodos mencionados como los más usados por las empresas pueden ser aplicados solo en la etapa de Implementación, por lo que existe una contradicción en sus respuestas, ya que en la pregunta 3 de la sección de Métodos se observa que mencionaron que ponen esfuerzo alto en el uso de estos métodos en las demás etapas de desarrollo.

Hipótesis 2

Según los resultados obtenidos con los indicadores de la Prueba de hipótesis las herramientas que utilizan las empresas son independientes del tipo de esfuerzo que emplean; esta independencia demuestra que aunque las empresas respondieron que si poseen herramientas de V&V establecidas para cada fase de desarrollo de software, estas no se refleja en cada fase de desarrollo, es decir no son usadas en adecuadamente en el proceso de desarrollo.

La segunda hipótesis de nuestro estudio ha quedado demostrada como verdadera, al igual que la hipótesis anterior, la demostración se la ha realizado en base a los resultados de la preguntas 1 y 4 de la sección de Herramientas de la encuesta y de la prueba de hipótesis, para esta hipótesis la conclusión es verdadera en nuestro medio. Aunque la mayoría de las empresas respondieron que si utilizan herramientas de

V&V para el desarrollo de software, esto no se refleja en la utilización en cada fase de desarrollo.

Otro punto que refuerza la conclusión obtenida es que las herramientas mencionadas como las más usadas por las empresas pueden ser aplicadas solo en la etapa de Implementación, por lo que existe una contradicción en sus respuestas ya que en la pregunta 4 de la sección de herramientas mencionan que ponen esfuerzo en el uso de herramientas de V&V en las demás etapas de desarrollo.

Hipótesis 3

Según los resultados obtenidos con los indicadores de la Prueba de hipótesis las técnicas que utilizan las empresas son independientes del tipo de esfuerzo que emplean; esta independencia demuestra que las empresas si poseen técnicas de V&V establecidas para cada fase de desarrollo de software.

Revisando los resultados de la pregunta 1 de la sección de Técnicas de la encuesta, en la cual el 89.47% de las empresas indicaron que si utilizan Métodos de V&V para el desarrollo de software. Aunque la mayoría de las empresas respondieron que si utilizan técnicas de V&V para el desarrollo de software, esto no se refleja la utilización en cada fase de desarrollo. Por lo cual se puede concluir que el valor de la hipótesis planteada es verdadero.

Hipótesis 4

Según los resultados obtenidos con los indicadores de la Prueba de hipótesis el tipo de herramienta que utiliza la empresa depende del tamaño de la empresa.

La cuarta hipótesis la conclusión a la que se ha llegado es que dicha hipótesis es verdadera, la demostración se la ha realizado en base a los resultados de las preguntas 1 de la sección de Información General y la pregunta 2 de la sección de Herramientas de la encuesta y de la prueba de hipótesis, para esta hipótesis la conclusión es verdadera en nuestro medio, ya que a la mayoría de empresas pequeñas utilizan herramientas de V&V de distribución Libre para el desarrollo de software. Esto se debe a que las empresas tienden a ver como un gasto la adquisición de licencias para utilizar una herramienta, por ende optan por utilizar herramientas cuyas licencias son gratuitas y que brindan buenos resultados.

Hipótesis 5

Según los resultados obtenidos con los indicadores de la Prueba de hipótesis el tipo de herramientas que emplea la empresa es independiente del servicio que brinda.

La última hipótesis tenemos que a la conclusión a la que se ha llegado es falsa, la demostración se la ha realizado en base a los resultados de las preguntas 2 de la sección de Información General y la pregunta 2 de la sección de Herramientas de la encuesta y de la prueba de hipótesis, para esta hipótesis la conclusión es falsa en nuestro medio, ya que la selección de herramientas de V&V licenciadas en las empresas no depende del tipo de servicio que las empresas brindan.

4.4 Listado de herramientas, métodos y técnicas de Verificación y Validación de Software adecuadas para el uso de las empresas ecuatorianas desarrolladoras de software

A través de la investigación realizada, proporcionamos a continuación un listado de Métodos, técnicas y herramientas de V&V que la hemos obtenido por medio de la encuesta.

Cabe indicar que del estudio realizado en las ciudades de Quito y Guayaquil se obtuvieron resultados por debajo de lo esperado en lo que se refiere al uso de éstas metodologías, técnicas y herramientas, especialmente en la ciudad de Guayaquil donde se observa una marcada diferencia en esta área:

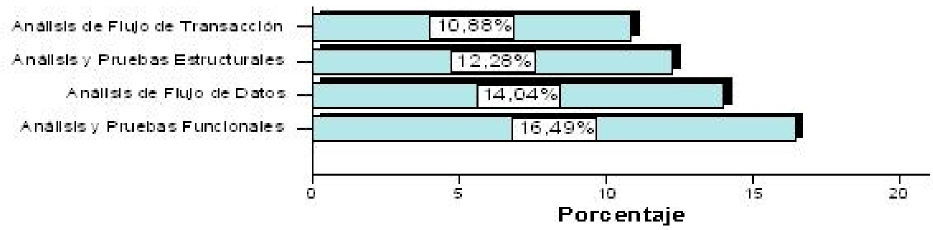
- **Técnicas de V&V que aportan más Calidad a las Empresas Ecuatorianas**
  Como resultado de las entrevistas con los jefes de los departamentos de sistemas de empresas de la ciudad de Quito y Guayaquil, hemos podido indagar sobre técnicas de V&V, que utilizan en sus respectivas empresas para llevar a cabo el

desarrollo de productos de software y a demás aportan a la calidad de software.

Entre las principales Técnicas que las empresas emplean en sus procesos están las siguientes:

1. Análisis y Pruebas Funcionales
2. Análisis de Flujo de Datos
3. Análisis y Pruebas Estructurales
4. Análisis de Flujo de Transacción

Figura 4.4.1. Porcentaje de Técnicas de V&V que aportan mas calidad a empresas Ecuatorianas



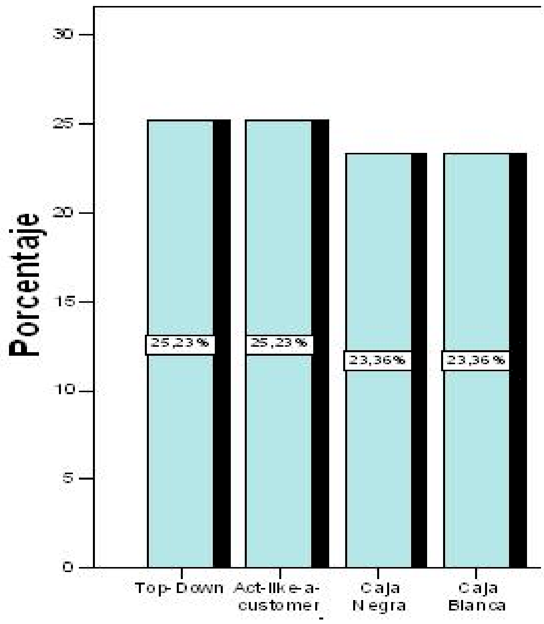
- **Métodos de V&V que aportan más Calidad a las Empresas Ecuatorianas**

Como resultado de las entrevistas con los jefes de los departamentos de sistemas de empresas de la ciudad de Quito y Guayaquil, hemos podido indagar sobre métodos de V&V, que utilizan en sus respectivas empresas para llevar a cabo el desarrollo de productos de software y a demás aportan a la calidad de software.

Entre los principales Métodos que las empresas emplean en sus procesos están las siguientes:

1. Top- Down
2. Act- like a Customer
3. Caja Negra
4. Caja Blanca

Figura 4.4.2 Porcentaje de Métodos de V&V que aportan más calidad a empresas Ecuatorianas

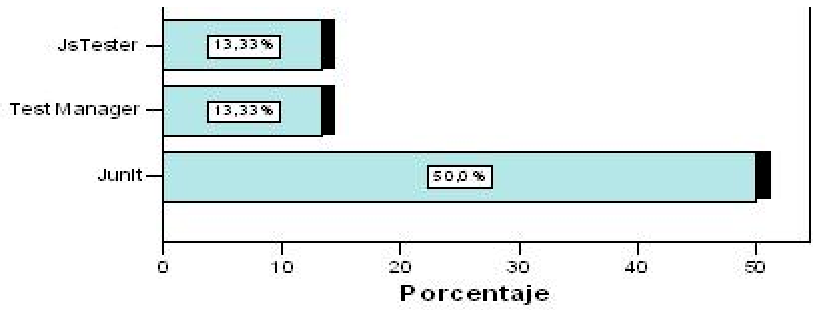


- **Herramientas de V&V de Distribución Libre que aportan más calidad a Empresas Ecuatorianas**
  Como resultado de las entrevistas con los jefes de los departamentos de sistemas de empresas de la ciudad de Quito y Guayaquil, hemos podido indagar sobre las Herramientas de distribución libre de V&V, que utilizan en sus respectivas empresas para llevar a cabo el desarrollo de productos de software y a demás aportan a la calidad de software.

Entre las principales Herramientas de distribución Libre que las empresas emplean en sus procesos están las siguientes:

1. JUnit
2. Test Manager
3. JsTester
4. Act- like a Customer

Figura 4.4.3 Porcentaje de Herramientas de Distribución Libre de V&V que aportan más calidad a empresas Ecuatorianas



- **Herramientas de V&V de Propietarias que más aportan calidad a Empresas Ecuatorianas**

Como resultado de las entrevistas con los jefes de los departamentos de sistemas de empresas de la ciudad de Quito y Guayaquil, hemos podido indagar sobre las Herramientas Propietarias de V&V, que utilizan en sus respectivas empresas para llevar a cabo el desarrollo de productos de software.

Entre las principales Herramientas propietarias que las empresas emplean en sus procesos están las siguientes:

1. Microsoft TeamSystem

2. IBM Rational

3. Software Security

Figura 4.4.3 Porcentaje de Herramientas Propietarias de V&V que aportan más calidad a empresas Ecuatorianas

CONCLUSIONES

1. Que la empresas ecuatorianas si emplean Métodos de V&V para el desarrollo de Software, sin embargo existen empresas que no tienen definido en que etapa del proceso de desarrollo de software emplear un determinado método, por lo cual la aplicación de estos métodos no lo hacen de manera correcta; tomando en cuenta la pregunta 3 de la sección de Métodos tenemos que mayor énfasis le dan a la etapa de Implementación con un porcentaje de 24.27%.
2. Otro punto a considerar es que los métodos seleccionados por las empresas pueden ser aplicados solo en la etapa de Implementación, por lo que existe una contradicción ya que la mayor parte menciona que pone esfuerzo en el uso de métodos de de V&V en otras fases del desarrollo.
3. La mayor parte de las empresas presentes en la encuesta si poseen Herramientas de V&V, pero al igual que en el caso anterior aun no tienen definido en que fase explotar sus beneficios. Como se puede ver en la pregunta referente al tipo de esfuerzo que se aplica a cada fase de desarrollo, la diferencia entre las etapas de Implementación, Integración y Pruebas, Diseño y Definición de requerimientos es de alrededor de 1%, cosa que no sucede en la etapa de mantenimiento en donde el uso de herramientas es mínimo.
4. De las herramientas seleccionadas por las empresas pueden ser aplicados solo en la etapa de Implementación, por lo que existe una contradicción ya que la mayor parte menciona que pone esfuerzo en el uso de herramientas de V&V en otras fases del desarrollo.

5. En cuanto a la utilización de Técnicas de V&V, las empresas en su gran mayoría (98.25) menciono que si utiliza, a demás se observa en la pregunta 3 de la sección de Técnicas que la etapas en las cuales las utilizan son: Implementación, Integración y Pruebas y Diseño con una diferencia pequeña del 1% entre cada etapa.
6. Que a menor número de personas en el área de sistemas como se puede ver en la pregunta 1 de la sección de Información General de la encuesta, las empresas optan por utilizar herramientas de V&V de distribución Libre, lo que ratifica la validez de la Hipótesis 4.

7. La mayoría de empresas encuestadas consideran que la importancia del uso de herramientas de V&V radica en saber cómo y cuándo aplicarlas para lograr el máximo beneficio de ellas, ya que representan una inversión considerable la adquisición de éstas para su empresa.

8. Para finalizar, la principal contribución de esta tesis ha sido la definición de un listado de Metodologías, Herramientas y Técnicas de V&V que se definieron en la sección 4.4, para el proceso de desarrollo, que sirvan para tomar como base para desarrollar sistemas.

RECOMENDACIONES

1. Al momento de finalizar esta tesis, es recomendable seguir profundizando sobre Metodologías, técnicas y Herramientas de V&V para proyectos futuros en el cual se empleen de una forma adecuada dichas metodologías, técnicas y herramientas, además de contar con un departamento capacitado con el resultado de la tesis.

2. Con los resultados de esta encuesta se puede inferir que las pequeñas empresas no se informan correctamente acerca de las nuevas metodologías, técnicas ó herramientas de V&V. Las empresas deben informarse mejor acerca de las innovaciones en el campo de la Ingeniería de software, para así estar en capacidad de desarrollar sistemas que satisfagan y que sean entregados a tiempo y con todos los requerimientos.

3. En las empresas pequeñas ubicadas en la ciudad de Guayaquil, se podría iniciar un trabajo conjunto que involucre el empleo por parte de las empresas de herramientas de V&V de distribución Libre ya que éstas no representan una inversión y mas bien significan una mejora en el proceso de desarrollo de Software.

4. Para finalizar se recomendaría un análisis en base al resultado de este trabajo y seleccionar a empresas en nuestro medio que se dediquen al desarrollo de software con modelos de negocios similares en la cual se utilicen durante todo el proceso de desarrollo

de software y dependiendo la naturaleza de los proyectos una determinada metodología, técnica y herramienta de V&V versus otra empresa de similares características en su negocio en la cual no se utilicen estas metodologías, técnicas y herramientas de V&V, con el objetivo de determinar cuan beneficioso representa el uso de estas metodologías en la actualidad.

BIBLIOGRAFÍA

[1] Perfiles de Productos de Exportación, disponible en http://www.ecuadorexporta.org/productos_down/perfil_producto_software567, última Visita: febrero 2008

[2] Formalismos Ágiles en Ingeniería del Software (2002), disponible en http://self.lcc.uma.es/index.html, última Visita: Marzo 2008

[3] Verificación y Validación: nueva propuesta para la mejora de la calidad del software (2005), disponible en http://www.baquia.com/imprimir.php?id=9778, última Visita: febrero 2008

[4] Ian Sommerville, Ingeniería de Software, séptima Edición. Addison Wesley Press; pp 7-9,384, 469-487

[5] Marco Normativo en el Sector del Software, disponible en http://www.cuti.org.uy/Portals/0/Brasil%20-%20Marco%20Normativo.pdf, última Visita: Septiembre 2007.

[6] Estrategias Productivas Ecuador 2009, Ministerio de Coordinación de la Producción, disponible en http://www.mcpc.gov.ec/, última visita: Marzo 2009

.

[7] Estudio de la Industria de Software, disponible en http://www.aesoft.com.ec, última visita: Mayo 9, 2008

[8] D. Salazar, M. Villavicencio, V. Macías, M. Snoeck, "Estudio Estadístico Exploratorio de las Empresas Desarrolladoras de Software Asentadas en Guayaquil, Quito y Cuenca".

[9] Jacobson, I., Booch, G., Rumbaugh J., El Proceso Unificado de Desarrollo de Software, Addison Wesley 2000.

[10] Mills, H., O´Neill, D., The Management of Software Engineering, IBM Systems, 1980.

[11] Boehm, B. W., A Spiral Model of Software Development and Enhancement, IEEE Computer, 1988.

[12] Steven R. Rakitin (2001). Software Verification and Validation for

Practitioners and Managers, 2da edición Artech House

[13] Clases de Ingeniería de Software. Ing. Verónica Macías
Tema: Verificación y Validación de Software

[14] Validación Cruzada disponible en http://es.wikipedia.org/wiki/Validaci%C3%B3n_cruzada, última visita: Agosto 2008

[15] Herramienta de distribución libre UISpec, disponible en http://www.uispec4j.org/index.html, última visita: Agosto 2008

[16] Herramienta de distribución libre JUnit, disponible en http://java-source.net/open-source/testing-tools/junit, última visita: Julio 2008

[17] Herramienta de distribución libre TestManager, disponible en http://www.mbtech-group.com/cz/electronics_solutions/test_engineering/provetechta_overview/testmanager.html, última visita: Mayo 2008

[18] Klaus Havelund, Grigore Ro_su, "Java PathExplorer - A Runtime Verication Tool, Moffett Field – California- USA, disponible en http://ti.arc.nasa.gov/m/pub, última Visita: julio 2008

[19] Herramienta de distribución libre JMock 2.0, disponible en http://www.jmock.org/, última Visita: marzo 2008

[20] Herramienta de distribución libre Selenium, disponible en http://www.jourmoly.com.ar/automatizando-pruebas-selenium-ide/, última Visita: abril 2008

[21] Herramienta de distribución libre JsTester, disponible en http://jstester.sourceforge.net/, última Visita: Mayo 2008

[22] Herramienta de distribución libre JSystem, disponible en http://www.roseindia.net/opensource/opensourcesoftware.php?id=334, última Visita: Junio 2008

[23] Herramienta de distribución libre JSystem, disponible en http://ostatic.com/33933-software-opensource/jsystem, última Visita: Julio 2008

[24] Herramienta de distribución libre DDTUnit, disponible en http://ddtunit.sourceforge.net/, última Visita: Mayo 2008

[25] Herramienta de distribución libre Jameleon, disponible en http://jameleon.sourceforge.net/index.html, última Visita: Junio 2008

[26] Herramienta de distribución libre T2, disponible en http://www.cs.uu.nl/wiki/WP/T2Framework, última Visita: Abril 2008

[27] Herramienta de distribución libre Kronos, disponible en http://www-verimag.imag.fr/TEMPORISE/kronos/, última Visita: Julio 2008

[28] Herramienta de distribución libre HyTech, disponible en http://embedded.eecs.berkeley.edu/research/hytech/, última Visita: Junio 2008

[29] Herramienta de distribución libre SMV, disponible en http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/MS-qest2004.pdf, última Visita: Marzo 2008

[30] Herramienta de distribución libre Spin, disponible en http://spinroot.com/spin/whatispin.html, última Visita: Julio 2008

[31] Herramienta de distribución libre UPPAAL, disponible en http://www.uppaal.com/, última Visita: Junio 2008

[32] Herramientas Propietarias TeamSystem, disponible en http://msdn.microsoft.com/en-us/vsts2008/products/default.aspx, última Visita: Abril 2008.

[33] Herramientas Propietarias ChecKing de ALS disponible en http://www.als-es.com/home.php?location=herramientas/checKing, última Visita: Abril 2008.

[34] Herramientas Propietarias Open Load Tester, disponible en http://www.als-es.com/home.php?location=herramientas/entorno-pruebas/open-load-tester, última Visita: Junio 2008.

[35] Herramientas Propietarias QACenter de Compuware, disponible en http://www.als-es.com/home.php?location=herramientas/entorno-pruebas/qacenter, última Visita: Abril 2008.

[36] Herramientas Propietarias Security Tester de Fortify, disponible en http://www.als-es.com/home.php?location=herramientas/entorno-pruebas/security-tester, última Visita: Mayo 2008.

[37] Herramientas Propietarias SOATest de Parasoft, disponible en http://www.parasoft.com/jsp/products/home.jsp?product=SOAP&itemId=101, última Visita: Mayo 2008.

[38] Herramientas Propietarias Software Security, disponible en http://www.als-es.com/home.php?location=herramientas/entorno-pruebas/software-security-manager, última Visita: Mayo 2008.

[39] Herramientas Propietarias .TEST de Parasoft, disponible en http://www.parasoft.com/jsp/products/home.jsp?product=TestNet&itemId=135, última Visita: Abril 2008.

[40] Herramientas Propietarias Webking de Parasoft, disponible en http://www.parasoft.com/jsp/products/home.jsp?product=WebKing&itemId=86, última Visita: Abril 2008.

[41] Rational Software Inc., RUP - Rational Unified Process, www.rational.com , 2003. , última Visita: Mayo 2008.

[42] Técnica de Verificación Orientada al Error, disponible en http://www.versioncero.com/articulo/141/poe-programacion-orientada-a-errores, última Visita: Febrero 2008.

[43] Técnica de Verificación Testing and Analisys, disponible en http://ix.cs.uoregon.edu/~michal/book/, última Visita: Marzo 2008

[44] Facultad de Informática Universidad de La Coruña, Estrategias de Integración disponible en http://www.fic.udc.es/HarvestExternalData.do?operation=marks.courses, última Visita: junio 2008

[45] Técnica de Verificación Análisis de Flujo de Transacción, disponible en http://exa.unne.edu.ar/depar/areas/informatica/anasistem2/public_html/apuntes/de2.pdf, última Visita: Junio 2008

[46] Software Quality Topics, "Stress Testing", http://searchsoftwarequality.techtarget.com/, última Visita: noviembre 2008

[47] Universidad de los Andes, "Análisis de Falla de Componentes de Ingeniería", Gustavo Tovar, disponible en http://revistaing.uniandes.edu.co/index.php?idr=15&ids=12&ida=149&ri=3d0a9885ce52282410239cd384dcc36d, última Visita: Agosto 2008

[48] Análisis de Concurrencia, disponible en http://www.mitecnologico.com/Main/ElModeloDesarrolloConcurrente, última Visita: noviembre 2008

[49] Análisis de Concurrencia, disponible en http://diario.de/osjobu/ciclos-de-vida.html, última Visita: noviembre 2008

[50] Wikidepia, "Análisis de Algoritmos", disponible en http://es.wikipedia.org/wiki/An%C3%A1lisis_de_algoritmos, última Visita: noviembre 2008

[51] Análisis de Cobertura, disponible en http://geeks.ms/blogs/rcorral/archive/2008/08/13/cobertura-de-c-243-digo-m-225-s-all-225-de-las-pruebas-unitarias.aspx, última Visita: Julio 2008

[52] Inspecciones, disponible en http://www.eici.ucm.cl/Academicos/ygomez/descargas/calidad/, última Visita: Julio 2008

[53] Wikipedia, "Model Checking", disponible en http://es.wikipedia.org/wiki/Model_checking, última visita: Agosto 2008

[54] Evaluación de Arquitecturas disponible en http://www.dc.uba.ar/materias/arq-soft/2007/cuat1/descargas/19.Evaluacion-v0.0.pdf/view última visita: Septiembre 2008

[55] Active Reviews for Intermediate Designs, disponible en http://www.sei.cmu.edu/publications/documents/00.reports/00tn009.html, última visita: Mayo 2008

[56] William Florac, Anita Carleton, "Measuring the Software process", 1999, Software Engineering Institute, Addison Wesley.

[57] Mason y Lind, "Estadística para Administración y Economía", 1992, Alfaomega.

[58] Wikipedia, "Encuesta", disponible en: http://es.wikipedia.org/wiki/Encuesta, última visita: Julio 2007 Que es Encuesta

[59] La industria del Software en Brasil: ¿Un mercado interno fuerte puede promover las exportaciones?, disponible en: http:// www.iadb.org/, última visita: Enero 2009

[60] VILLAVICENCIO, Monica, IZQUIERDO, Edgar, SNOECK, Monique, Aspects Regarding Management of Software Projects that Influence the Quality of Software Products Released: The Case of Ecuador, Memoria Técnica V Jornadas Iberoamericanas de Ingeniería de Software e Ingeniería del Conocimiento (JIISIC '06), p27-35, Volumen 5, Número 1, Puebla, México, 2006, ISBN 970-94770-0-5

[61] GONZALEZ, Raúl, HERNANDEZ, Hennry, VILLAVICENCIO, Mónica, Desarrollo de un Código de Métricas para Pequeñas Empresas Ecuatorianas Desarrolladoras de Software, Memoria Técnica VI Jornadas Iberoamericanas de Ingeniería de Software e Ingeniería del Conocimiento (JIISIC '07), Volumen 6, Número 1, p221 – 227, Lima, Perú, 2007, ISBN 9972-2885-1-7 (Esta es de COMPETISOFT)

ANEXO 1: Encuesta Piloto aplicada al Estudio

Estudio estadístico de las empresas desarrolladoras de Software de Quito y Guayaquil.

I. Información General

Nombre de la Empresa: ______________________________ Fecha: ______________
Dirección: ____
Ciudad: ____________ Teléfono: ___________

e – mail: _______________________

1. – De acuerdo con cada uno de los siguientes títulos académicos, determine el número de empleados en el área de desarrollo de sistemas.

a.- Bachiller	
b.- Programador/ Analista de sistemas	
c.- Ingeniero en Sistemas	
d.- Master	
e.- Doctor (Ph.D.)	
f.- Otro Indique: ______________________	

2.- Su empresa se dedica a (marque con una X todas las que se aplican):

a. Consultorías/auditorías informáticas	
b. Desarrollo de aplicaciones computacionales (contratista)	
c. Distribución de aplicaciones computacionales (compra a otros para luego vender)	
d. Desarrollo y venta de aplicaciones computacionales desarrolladas por la empresa	
e. Otro Indique: _____________________	

3.- Su empresa tiene algún tipo de certificación?

a)	Si	
b)	No	
c)	En proceso	

Si su respuesta fue Si o en Proceso especifique que tipo de certificación es?

____

4.- En su empresa qué paradigma utilizan a la hora de desarrollar un software (Marque con una X las que se aplican)?

a. Orientado a Objetos	
b. Estructurado	

c. Otro. ☐
Por favor, indique:
____

II. Información Entrevistado

Nombre del entrevistado: ____________________	Edad: _________	Sexo: ______
Teléfono: _________________	e-mail: __________________________	

1.- Indique su nivel de educación más alto:

a.- Bachiller	
b.- Programador/Analista de sistemas	
c.- Ingeniero en Sistemas	
d.- Master	
e.- Doctor (Ph.D.)	
f.- Otro Indique: ___________________________	

2.- Indique el cargo que desempeña en la empresa:
____

III. Métodos

1) En su empresa utilizan Métodos V&V previamente establecidos para llevar a cabo el desarrollo de un proyecto?
   a. Si b. No

2) Indique el tipo de Métodos que utilizan en la empresa :

a.- Caja Negra	
b.- Caja Blanca	
c.- Top-Down	
d.- Act-like-a-customer	
e.- ATAM	
f.- ADR	
g.- ARID	
h.- Otro Indique: ___________	

3) De los métodos seleccionados en la pregunta anterior señale el tipo de esfuerzo que se aplica en cada fase de desarrollo de la siguiente tabla donde M es mucho, P es poco y N es ninguno.

Métodos	Definición de requerimientos			Diseño			Implementación			Integración y Pruebas			Mantenimiento		
	M	P	N	M	P	N	M	P	N	M	P	N	M	P	N
Caja Negra															
Caja Blanca															
Top-Down															
Act-like-a-customer															
* ATAM															
* ADR															
* ARID															

Otro: Indique ________															

*ATAM.- Architecture Tradeoff Analysis Method
*ADR.- Active Design Review
*ARID.- Active Review of Immediate Design

4) De los métodos anteriormente nombrados cual frecuentemente es usado en el proceso de desarrollo de Software.

---

5) Según su criterio la utilización de estos métodos permite una optimización en el tiempo planificado del proyecto.

a. Totalmente Desacuerdo ( )
b. Parcial Desacuerdo ( )
c. Posición Intermedia ( )
d. Parcial Acuerdo ( )
e. Total Acuerdo ( )

6) Cuales Métodos cree a su criterio que aportan más en la Calidad del producto final?

a.- Caja Negra	
b.- Caja Blanca	
c.- Top-Down	
d.- Act-like-a-customer	
e.- ATAM	
f.- ADR	
g.- ARID	
h.- Otro Indique: __________	

IV Técnicas

1) En su empresa utilizan Técnicas V&V previamente establecidos para llevar a cabo el desarrollo de un proyecto?
a. Si b. No

2) Indique el tipo de Técnicas que utilizan en la empresa

Pruebas Funcionales y Análisis	
Pruebas Estructurales y Análisis	
Error-Oriented	
Testing and Análisis	
Estrategias de Integración	
Hybrid Approaches	
Análisis de Flujo de Transacción	
Análisis de Stress	
Análisis de Falla	
Análisis de Concurrencia	
Model Checking	
Otro: _______________	

3) De las técnicas seleccionadas en la pregunta anterior señale el tipo de esfuerzo que se aplica en cada fase de desarrollo de la siguiente tabla, donde M es mucho, P es poco y N es ninguno.

Técnicas	**Definición de requerimientos**			**Diseño**			**Implementación**			**Integración y pruebas**			**Mantenimiento**		
	M	**P**	**N**	**M**	**P**	**N**	**M**	**P**	**N**	**M**	**P**	**N**	**M**	**P**	**N**
Pruebas Funcionales y Análisis															
Pruebas Estructurales y Análisis															
Error-Oriented															
Testing and Analysis															
Estrategias de Integración															
Hybrid															

Approaches															
Análisis de Flujo de Transacción															
Análisis de Stress															
Análisis de Falla															
Análisis de Concurrencia															
Model Checking															
Otro Indique ____________															

4) Según su criterio, cuales Técnicas cree a su criterio que aportan más en la Calidad del producto de software final?

Pruebas Funcionales y Análisis	
Pruebas Estructurales y Análisis	
Error-Oriented	
Testing and Análisis	
Estrategias de Integración	
Hybrid Approaches	
Análisis de Flujo de Transacción	
Análisis de Stress	
Análisis de Falla	
Análisis de Concurrencia	
Model Checking	
Otro: ________________	

V. Niveles de Pruebas

1. El tipo de Niveles de pruebas que utilizan en la empresa es (Marque con una X todas las que se aplican):

a. Pruebas de Módulos (Unidades)	
b. Pruebas de Integración	
c. Pruebas de Sistema	

d. Pruebas de Regresión
e. Pruebas de Implantación
f. Pruebas de Aceptación
g. Pruebas de Valores de Frontera
h. Otro.
Por favor, indique:
____

2) Quienes realizan las pruebas (Marque con una X todas las que se aplican):

a. Persona independiente que desarrolla el sistema
b. Equipo de personas destinadas al proceso de pruebas
c. La persona que desarrolla el sistema
g. Otro.
Indique: _______________________________________

3) Por favor, indique el número de personas designadas para la tarea de V&V de:

a.- Proyecto pequeño *
b.- Proyecto mediano **
c.- Proyecto grande ***

* Tiempo para la culminación entre 1 y 6 meses
** Tiempo para la culminación entre 7 y 15 meses
*** Tiempo para la culminación más de 15 meses

Nota: Considere condiciones normales

4) En su criterio el tiempo de retraso de un proyecto es directamente proporcional a la disminución de tiempo y esfuerzo dedicado a las actividades de verificación y validación

Totalmente Desacuerdo	( )
Parcial Desacuerdo	( )
Posición Intermedia	( )
Parcial Acuerdo	( )
Total Acuerdo	( )

VI. Herramientas

1. En su empresa utilizan Herramientas de V&V previamente establecidas para llevar a cabo el desarrollo de un proyecto?
   a. Si   b. No

2. Que tipo de herramientas emplean su empresa?

a. Software Propietario b. Software Libre

3. Seleccione las herramientas de V&V de software Libre que utilizan en su empresa. Si en su empresa **no** usan herramientas libres, por favor continúe con la pregunta 6.

a.- UISpec4J (Framework)	
b.- JUnit	
c.- TestManager	
d.- Java Path Explorer	
e.- JMock 2.0	
f.- Selenium	
g.- JsTester	
h.- JSystem (Framework)	
i.- DDTUnit (Framework)	
j.- Jameleon	
k.- T2	
l.- Kronos	
m.- HyTech (The HibridTechnology Tool)	
n.- SMV (Symbolic ModelVerifier)	
o.- Spin	
p.- UPPAAL	
q.- Otro Indique:____________________	

4. De las herramientas seleccionadas en la pregunta anterior señale el tipo de esfuerzo que se aplica en cada fase de desarrollo de la siguiente tabla donde M es mucho, P es poco y N es ninguno.

Herramientas	**Definición de requerimientos**			**Diseño**			**Implementación**			**Integración y Pruebas**			**Mantenimiento**		
	M	**P**	**N**	**M**	**P**	**N**	**M**	**P**	**N**	**M**	**P**	**N**	**M**	**P**	**N**
UISpec4J (Framework)															
JUnit															
TestManager Java Path Explorer															
JMock 2.0															
Selenium															
JsTester															

JSystem(Framework)															
DDTUnit (Framework)															
Jameleon															
T2															
Kronos															
HyTech															
SMV															
Spin															
UPPAAL															
Otro: Indique:____															

5. Cuales Herramientas cree a su criterio que aportan más en la Calidad del producto final?

a.- UISpec4J (Framework)	
b.- JUnit	
c.- TestManager	
d.- Java Path Explorer	
e.- JMock 2.0	
f.- Selenium	
g.- JsTester	
h.- JSystem (Framework)	
i.- DDTUnit (Framework)	
j.- Jameleon	
k.- T2	
l.- Kronos	
m.- HyTech (The HibridTechnology Tool)	
n.- SMV (Symbolic ModelVerifier)	
o.- Spin	
p.- UPPAAL	
q.- Otro Indique:______________________	

6. Seleccione las herramientas de V&V de software Propietario que utiliza en su empresa. Si en su empresa **no** usan herramientas de V&V de software Propietario continúe con la pregunta 9.

a.- Team System	
b.- ChecKing de ALS	
c.- Open Load Tester	
d.- QACenter de Compuware	
e.- Security Tester de Fortify	
f.- SOATest de Parasoft	
g.- Software Security	
h.- Manager de Fortify	
i.- .TEST de Parasoft	
j.- Webking de Parasoft	
k.- IBM Racional	
l.- Otro: Indique:____________________	

7. De las herramientas seleccionadas en la pregunta anterior señale el tipo de esfuerzo que se aplica en cada fase de desarrollo de la siguiente tabla donde M es mucho, P es poco y N es ninguno.

Herramientas	Definición de requerimientos			Diseño			Implementación			Integración y Pruebas			Mantenimiento		
	M	P	N	M	P	N	M	P	N	M	P	N	M	P	N
TeamSystem															
checKing de ALS															
Open Load Tester															
QACenter de Compuware															
Security Tester de Fortify															
SOATest de Parasoft															
Software Security															
Manager de Fortify															
.TEST de Parasoft															
Webking de Parasoft															
IBM Racional															
Otro: Indique___															

8. Cuales Herramientas cree a su criterio que aportan más en la Calidad del producto final?

a.- Team System	
b.- ChecKing de ALS	
c.- Open Load Tester	

d.- QACenter de Compuware	
e.- Security Tester de Fortify	
f.- SOATest de Parasoft	
g.- Software Security	
h.- Manager de Fortify	
i.- .TEST de Parasoft	
j.- Webking de Parasoft	
k.- IBM Racional	
l.- Otro: Indique: ____________________	

9. Según su criterio cuales son los aspectos por los cuales se utiliza una u otra herramienta/método/técnica de V&V, donde M es mucho, P es poco y N es ninguno.

Criterios	**Método**			**Técnica**			**Herramienta**		
	M	**P**	**N**	**M**	**P**	**N**	**M**	**P**	**N**
Minimizar Costos									
Optimizar Tiempo									
Optimizar Recursos									
________									
________									
________									
________									

ANEXO 2: Encuesta Final aplicada al Estudio

Estudio estadístico de los métodos, técnicas y herramientas de Verificación y Validación de Software usadas por las empresas desarrolladoras de Software de Quito y Guayaquil

I. Información General

Nombre de la Empresa: ______________________________ Fecha: __________
Dirección: ____
Ciudad: ____________ Teléfonos: ___________________________________
Email: ____

1) De acuerdo con cada uno de los siguientes títulos académicos, determine el número de empleados en el área de desarrollo de sistemas.

a.- Bachiller	
b.- Programador / Analista de Sistemas	
c.- Ingeniero de Sistemas	
d.- Master	
e.- Doctor (Ph.D)	
f.- Otro Indique: ______________________________	

2) Su empresa se dedica a (marque con una X todas las que se aplican):

a.- Consultorías / auditorías informáticas	
b.- Desarrollo de aplicaciones computacionales (contratista)	
c.- Distribución de aplicaciones computacionales (compra a otros para luego vender)	
d.- Desarrollo y venta de aplicaciones computacionales desarrolladas por la empresa	
e.- Otro Indique: ______________________________	

3) Su empresa ¿tiene algún tipo de certificación?

a.- Si	
b.- No	
c.- En proceso	

Si su respuesta fue Si o en Proceso especifique que tipo de certificación es?

____

4) En su empresa ¿qué paradigma utilizan a la hora de desarrollar un software?

a.- Orientado a Objetos	
b.- Estructurado	
c.- Otro Indique: ______________________________	

II. Niveles de Pruebas

1) El tipo de niveles de pruebas que utilizan en la empresa es (Marque con una X todas las que se aplican):

a.- Pruebas de Módulos (Unidades)	
b.- Pruebas de Integración	
c.- Pruebas de Sistema	
d.- Pruebas de Regresión	
e.- Pruebas de Implantación	
f.- Pruebas de Aceptación	
g.- Pruebas de Valores de Frontera	
h.- Otro Indique: ____________________	

2) Quiénes realizan las pruebas (Marque con una X todas las que se aplican):

a.- Persona independiente que desarrolla el sistema	
b.- Equipo de personas destinadas al proceso de pruebas	
c.- La persona que desarrolla el sistema	
d.- Otro Indique: ____________________	

3) Por favor, indique el número de personas designadas para la tarea de V&V de:

a.- Proyecto pequeño *	
b.- Proyecto mediano **	
c.- Proyecto grande ***	

4) En su criterio el tiempo de retraso de un proyecto es directamente proporcional a la disminución de tiempo y esfuerzo dedicado a las actividades de verificación y validación

a.- Total desacuerdo	
b.- Parcial desacuerdo	
c.- Posición intermedia	
d.- Parcial acuerdo	
e.- Total acuerdo	

5)¿Cuáles son los criterios por los cuales su empresa decide (o escogería) utilizar una herramienta/método/tecnología de V&V específica? Por favor indique en que medida este criterio afecta la desición, tomando en cuenta que M es mucho, P es poco y N es ninguno.

Criterio	M	P	N
a.- Minimizar costos			
b.- Optimizar tiempo			
c.- Optimizar recursos			
d.- Mejorar calidad			
e.-			
f.-			
g.-			

III. Técnicas

1) En su empresa utilizan técnicas de V&V previamente establecidas para llevar a cabo el desarrollo de un proyecto?

a.- Si   b.-No

2) Indique el tipo de técnicas que utilizan en la empresa

a.- Análisis y Pruebas Funcionales	
b.- Análisis y Pruebas Estructurales	
c.- Error-Oriented	
d.- Testing and Analysis	
e.- Estrategias de Integración	
f.- Hybrid Approaches	
g.- Análisis de Flujo de Transacción	
h.- Análisis de Stress	
i.- Análisis de Fallas	
j.- Análisis de Concurrencia	
k.- Análisis de Algoritmos	
l.- Análisis de Cobertura	
m.- Análisis de Flujo de Datos	
n.- Inspecciones	
ñ.- Pruebas de Desempeño	
o.- Model Checking	
p.- Otro Indique: ____________________	

3) De las técnicas seleccionadas en la pregunta anterior señale el tipo de esfuerzo que se aplica en cada fase de desarrollo de la siguiente tabla, donde M es mucho, P es poco y N es ninguno.

	Definición de requerimientos			**Diseño**			**Implementación**			**Integración y Pruebas**			**Mantenimiento**		
Técnicas	M	P	N	M	P	N	M	P	N	M	P	N	M	P	N
Análisis y Pruebas Funcionales															
Análisis y Pruebas Estructurales															
Error-Oriented															
Testing and Analysis															
Estrategias de Integración															
Hybrid Approaches															
Análisis de Flujo de Transacción															
Análisis de Stress															
Análisis de Falla															
Análisis de Concurrencia															
Análisis de Algoritmos															
Análisis de Cobertura															
Análisis de Flujo de Datos															
Inspecciones															
Pruebas de Desempeño															
Model Checking															
Otro Indique:															

4) Según su criterio, cuáles de estas técnicas aportan más a la calidad del producto final?

a.- Análisis y Pruebas Funcionales	
b.- Análisis y Pruebas Estructurales	
c.- Error-Oriented	
d.- Testing and Analysis	
e.- Estrategias de Integración	
f.- Hybrid Approaches	
g.- Análisis de Flujo de Transacción	
h.- Análisis de Stress	
i.- Análisis de Fallas	
j.- Análisis de Concurrencia	
k.- Análisis de Algoritmos	
l.- Análisis de Cobertura	
m.- Análisis de Flujo de Datos	
n.- Inspecciones	
ñ.- Pruebas de Desempeño	
o.- Model Checking	
p.- Otro Indique: ______________________________	

IV. Métodos

1) En su empresa utilizan métodos de V&V previamente establecidos para llevar a cabo el desarrollo de un proyecto?

a.- Si b.-No

2) Indique el tipo de métodos de V&V que utilizan en su empresa:

a.- Caja Negra	
b.- Caja Blanca	
c.- Top-Down	
d.- Act-like-a-customer	
e.- ATAM	
f.- ADR	
g.- ARID	
h.- Otro Indique: ______________________________	

3) De los métodos seleccionados en la pregunta anterior señale el tipo de esfuerzo que se aplica en cada fase de desarrollo de la siguiente tabla donde M es mucho, P es poco y N es ninguno.

	Definición de requerimientos			**Diseño**			**Implementación**			**Integración y Pruebas**			**Mantenimiento**		
Método	M	P	N	M	P	N	M	P	N	M	P	N	M	P	N
Caja Negra															
Caja Blanca															
Top-Down															
Act-like-a-customer															
ATAM															
ADR															
ARID															
Otro Indique:															

4) De los métodos anteriormente nombrados cuál es usado de manera más frecuente en el proceso de desarrollo de Software?:

____

5) Según su criterio, la utilización de estos métodos permite una optimización en el tiempo planificado del proyecto?

a.- Total desacuerdo	
b.- Parcial desacuerdo	
c.- Posición intermedia	
d.- Parcial acuerdo	
e.- Total acuerdo	

6) Según su criterio, cuáles de estos métodos aportan más a la calidad del producto final?

a.- Caja Negra	
b.- Caja Blanca	
c.- Top-Down	
d.- Act-like-a-customer	
e.- ATAM	
f.- ADR	
g.- ARID	
h.- Otro Indique: ______________________________	

V. Herramientas

1) En su empresa utilizan herramientas de V&V previamente establecidas para llevar a cabo el desarrollo de un proyecto?

a.- Si b.-No

2)Qué tipo de herramientas emplean en su empresa??

a.- Software Propietario b.-Software Libre

3) Seleccione las herramientas de V&V de software libre que utiliza en su empresa. Si en su empresa no usan herramientas de V&V de software libre, por favor pase a la pregunta 6 de esta sección.

a.- UISpec4J (Framework)	
b.- JUnit	
c.- Test Manager	
d.- Java Path Explorer	
e.- JMock 2.0	
f.- Selenium	
g.- JsTester	
h.- JSystem (Framework)	
i.- DDTUnit (Framework)	
j.- Jameleon	
k.- T2	
l.- Kronos	
m.- HyTech (The HibridTechnology Tool)	
n.- SMV (Symbolic Model Verifier)	
ñ.- Spin	
o.- UPPAAL	
p.- Otro Indique: ______________________	

4) De las herramientas seleccionadas en la pregunta anterior señale el tipo de esfuerzo que se aplica en cada fase de desarrollo de la siguiente tabla, donde M es mucho, P es poco y N es ninguno.

	Definición de requerimientos			**Diseño**			**Implementación**			**Integración y Pruebas**			**Mantenimiento**		
Herramientas	M	P	N	M	P	N	M	P	N	M	P	N	M	P	N
UISpec4J (Framework)															
JUnit															
Test Manager															
Java Path Explorer															
JMock 2.0															
Selenium															
JsTester															
JSystem (Framework)															
DDTUnit (Framework)															
Jameleon															
T2															
Kronos															
HyTech															
SMV															
Spin															
UPPAAL															
Otro Indique:															

5) Cuáles de estas herramientas cree a su criterio que aportan más a la calidad del producto final?

a.- UISpec4J (Framework)	
b.- JUnit	
c.- Test Manager	
d.- Java Path Explorer	
e.- JMock 2.0	
f.- Selenium	
g.- JsTester	
h.- JSystem (Framework)	
i.- DDTUnit (Framework)	
j.- Jameleon	
k.- T2	
l.- Kronos	
m.- HyTech (The HibridTechnology Tool)	
n.- SMV (Symbolic Model Verifier)	
ñ.- Spin	
o.- UPPAAL	
p.- Otro Indique: ____________________	

6) Seleccione las herramientas de V&V de software propietario que utiliza en su empresa. Si en su empresa no usan herramientas de V&V de software propietario, pase a la siguiente sección.

a.- TeamSystem	
b.- checKing de ALS	
c.- Open Load Tester	
d.- QACenter de Compuware	
e.- Security Tester de Fortify	
f.- SOATest de Parasoft	
g.- Software Security	
h.- Manager de Fortify	
i.- .TEST de Parasoft	
j.- Webking de Parasoft	
k.- IBM Racional	
l.- Otro Indique: ____________________	

7) De las herramientas seleccionadas en la pregunta anterior señale el tipo de esfuerzo que se aplica en cada fase de desarrollo de la siguiente tabla, donde M es mucho, P es poco y N es ninguno.

	Definición de requerimientos			**Diseño**			**Implementación**			**Integración y Pruebas**			**Mantenimiento**		
Herramientas	M	P	N	M	P	N	M	P	N	M	P	N	M	P	N
Team System															
checKing de ALS															
Open Load Tester															
QACenter de Compuware															
Security Tester de Fortify															
SOATest de Parasoft															
Software Security															
Manager de Fortify															
.TEST de Parasoft)															
Webking de Parasoft															
IBM Racional															
Otro Indique:															

8) Cuáles de estas herramientas cree a su criterio que aportan más a la calidad del producto final?

a.- TeamSystem	
b.- checKing de ALS	
c.- Open Load Tester	
d.- QACenter de Compuware	
e.- Security Tester de Fortify	
f.- SOATest de Parasoft	
g.- Software Security	
h.- Manager de Fortify	
i.- .TEST de Parasoft	
j.- Webking de Parasoft	
k.- IBM Racional	
l.- Otro Indique: ____________________	

VI. Información Entrevistado

Nombre del Entrevistado: ____________________ Edad: ______ Sexo: ____
Teléfonos: ____________________ Email: ____________________

1) Indique su nivel de educación más alto:

a.- Bachiller	
b.- Programador / Analista de Sistemas	
c.- Ingeniero de Sistemas	
d.- Master	
e.- Doctor (Ph.D)	
f.- Otro Indique: ____________________	

2) Indique el cargo que desempeña en la empresa:

____

ANEXO 3: Software SPSS

SPSS Base 15.0

El software estadístico integral

Statistical Package for the Social Sciences (SPSS) es un programa estadístico informático muy usado en las ciencias sociales y las empresas de investigación de mercado para cálculos estadísticos, aunque incluye un sin número de utilidades.

Fue creado en 1968 por Norman H. Nie, C. Hadlai (Tex) Hull y Dale H. Bent. Entre 1969 y 1975 la Universidad de Chicago por medio de su National Opinión Research Center estuvo a cargo del desarrollo, distribución y venta del programa. A partir de 1975 corresponde a SPSS Inc.

Actualmente, la estadística ha adquirido, de manera progresiva, una mayor relevancia en todos los sectores universitarios y, en general, en la sociedad.

A diferencia de otros paquetes estadísticos, SPSS es más fácil de utilizar, tiene un menor coste total de propiedad y abarca todas las etapas del proceso analítico. SPSS Base es una parte integral de este proceso, pues provee funciones para el acceso, gestión, preparación y análisis de los datos, y presentación de informes de resultados. Le permite trabajar con módulos adicionales y otros productos de la familia SPSS, que ofrecen la funcionalidad necesaria para la planificación, recopilación de datos, puesta en marcha y adicionalmente incrementa la funcionalidad para manejar otras áreas que también abarca SPSS.

Antes de que pueda analizar sus datos, necesita prepararlos para el análisis. SPSS incluye numerosas técnicas y características que permiten una fácil preparación de los datos. A continuación se resumen sólo algunas de las características más resaltantes de la gestión de datos de SPSS.

Con SPSS Base podrá configurar la información del diccionario de datos (por ejemplo, etiquetas de valores y tipos de variables) y preparar sus datos para el análisis más rápidamente gracias a la herramienta “Definir propiedades de las variables”. SPSS ofrece una lista de valores y sus recuentos permitiendo así añadir la información deseada. Una vez configurado el diccionario, podrá utilizar la información que este contiene con la herramienta "Copiar propiedades de datos". El diccionario de datos funciona como una plantilla que puede utilizar en otros archivos de datos y aplicar a otras variables del mismo archivo. SPSS facilita la detección de casos duplicados, de forma que puedan ser eliminados antes del análisis. Utilice la herramienta “Identificar casos duplicados” para establecer los parámetros y señalar los duplicados de modo que pueda seguirles la pista.

SPSS permite hacer estadísticas resumidas y la matemática de filas y columnas muy sencillas por ello ofrece un amplio rango de procedimientos estadísticos para el análisis básico incluyendo: recuentos, tablas de contingencia, conglomerados, estadísticos descriptivos, análisis de factor, regresión lineal, análisis de conglomerados y regresión ordinal. Una vez completado el análisis, podrá incluir datos en su base de datos con el “Asistente para exportar a bases de datos”. Para una mayor capacidad de análisis aproveche la amplia variedad de módulos que puede añadir a SPSS, como “SPSS Modelos de Regresión” y “SPSS Modelos Avanzados” que se centran en el análisis de los datos.

Genere fácilmente los gráficos más utilizados, como SPLOMs (diagramas de dispersión matriciales), histogramas y pirámides poblacionales, con el Constructor de gráficos. La interfaz visual para la creación de los gráficos permite crear un gráfico arrastrando las variables y elementos al lienzo de creación de dicho gráfico. También puede utilizar un método abreviado basado en algún gráfico existente en la galería. Podrá ver una vista preliminar simplificada del gráfico a medida que se construye. Los usuarios avanzados tendrán a su alcance un rango más amplio de gráficos y opciones utilizando el Lenguaje de Producción de Gráficos (GPL).

El sistema de presentación de gráficos proporciona control tanto en la etapa de creación como en la de edición, contribuyendo a aligerar la carga de trabajo en un ambiente de producción. Después de crear un gráfico, podrá utilizar sus especificaciones para crear cientos de gráficos similares.

SPSS tiene un sistema de ficheros en el cual el principal son los archivos de datos (extensión. SAV). Aparte de este tipo existen otros dos tipos de uso frecuente:

- Archivos de salida (output, extensión. SPO): en estos se despliega toda la información de manipulación de los datos que realizan los usuarios mediante las ventanas de comandos. Son susceptibles de ser exportados con varios formatos (originalmente HTML, RTF o TXT, actualmente la versión 15 incorpora la exportación a PDF junto a los formatos XLS y DOC que ya se encontraban en la versión 12)
- Archivos de sintaxis (extensión. SPS): Casi todas las ventanas de SPSS cuentan con un botón que permite hacer el pegado del proceso que el usuario desea realizar. Lo anterior genera un archivo de sintaxis donde se van guardando todas las instrucciones que llevan a cabo los comandos del SPSS. Este archivo es susceptible de ser modificado por el usuario. Muchos de los primeros usuarios del SPSS suelen escribir estos archivos en vez de utilizar el sistema de pegado del programa.

Existe un tercer tipo de fichero: el fichero de scripts (extensión. SBS). Este fichero es utilizado por los usuarios más avanzados del software para generar rutinas que permiten automatizar procesos muy largos y/o complejos. Muchos de estos procesos suelen no ser parte de las salidas estándar de los comandos del SPSS, aunque parten de estas salidas. Buena parte de la funcionalidad de los archivos de scripts ha sido ahora asumida por la inserción del lenguaje de programación Python en las rutinas de sintaxis del SPSS. Procedimientos que antes solo se podían realizar mediante scripts ahora se pueden hacer desde el sintaxis mismo.

El programa cuando se instala trae un determinado número de ejemplos o utilidades de casi todos los ficheros en cuestión. Estos son usados para ilustrar algunos de los ejemplos de uso del programa.

Se pueden generar estos archivos de sintax con la ayuda del programa mismo, pues en casi todas las ventanas donde se realizan tareas en el SPSS existe un botón "Paste". Este botón cierra la ventana en cuestión y guarda la sintaxis de las acciones seleccciondas en dicha ventana. Una vez salvado este archivo es susceptible de modificación.

La sintaxis tal cual se presenta a continuación fue producida directamente con el SPSS. Este programa le da un formato legible a la sintaxis, formato que el software en algunos casos no requiere para su correcto uso.

Otra peculiaridad de las sintaxis del SPSS es que no son "case sensitive". Ante lo cual es común ver sintaxis escritas en solo mayúsculas, solo minúsculas o bien una combinación propia de cada usuario. Esta situación se modifica para aquellas personas que hacen uso de Python dentro de sus sintaxis, pues este es un lenguaje sensitivo a las variaciones entre mayúsculas y minúsculas. Esto obliga a estos usuarios a escribir sintaxis con mayor cuidado.

ANEXO 4: Análisis de la Encuesta

ANÁLISIS DE LA ENCUESTA

El presente anexo presenta los resultados de las preguntas planteadas en la encuesta.

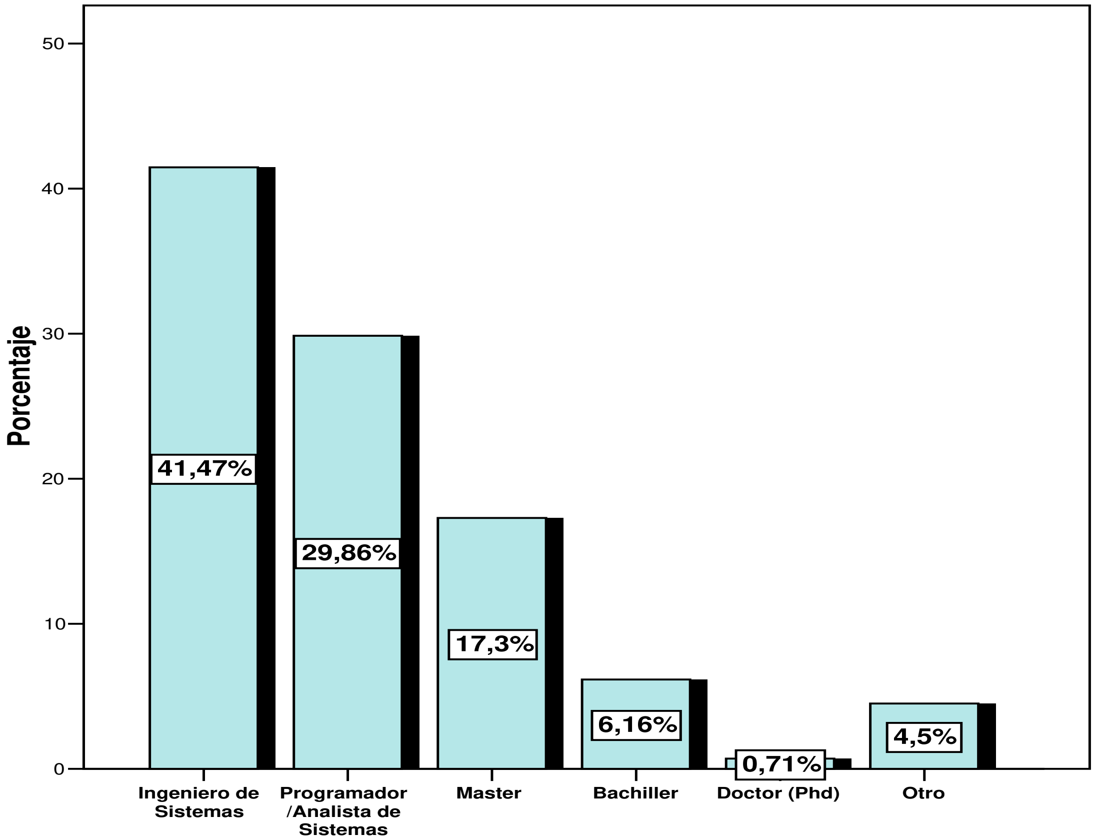
Sección I: **Información General**

1.- De acuerdo con cada uno de los siguientes títulos académicos, determine el número de empleados en el área de desarrollo de sistemas.

a.- Bachiller
b.- Programador/ Analista de sistemas
c.- Ingeniero en Sistemas
d.- Master
e.- Doctor (Ph.D.)
f.- Otro

Del tamaño de la muestra tomada para este estudio el 41.7% de las empresas encuestadas respondió que sus empleados poseen el titulo de Ingeniero en Sistemas, el 29.86% respondió que sus empleados tienen el grado de Programador ó Analista de Sistemas, el 17.3% respondió que poseían empleados con estudios de Pos-Grado como lo es una Maestría, el 6.16% de las empresas respondió que sus empleados tienen el titulo de Bachiller, el 0.71% de las empresas respondió que sus empleados tienen el grado de PhD, en cambio el 4.5% respondió que sus empleados poseían otros tipos niveles académicos no precisamente relacionados al área de la informática.

De aquel 4.5% que respondieron que sus empleados poseían otro tipo de niveles académico se destacan los siguientes: Diseñadores Gráficos, Ing. Comercial, Admr. Empresas, Ing. en Procesos, Lic. Sistemas.



2.- **Su empresa se dedica a:**

a. Consultorías/auditorías informáticas
b. Desarrollo de aplicaciones computacionales (contratista)
c. Distribución de aplicaciones computacionales (compra a otros para luego vender)
d. Desarrollo y venta de aplicaciones computacionales desarrolladas por la empresa
e. Otro
Indique:_______________________

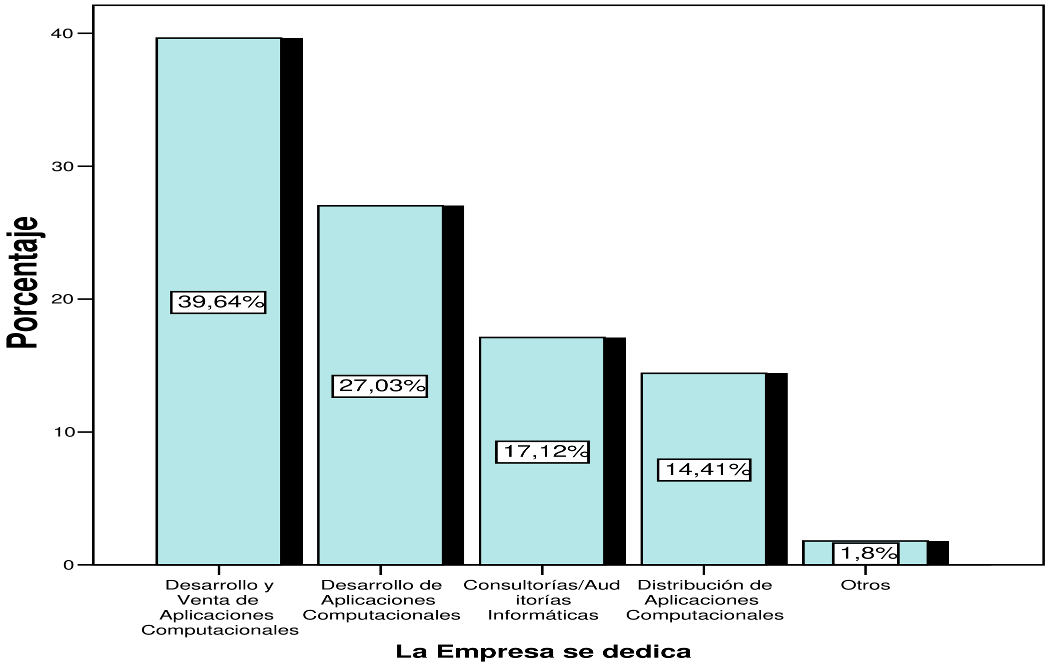
Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera, el 39.64% respondió que se dedicaba al Desarrollo y Venta de Aplicaciones Computacionales, el 27.03% respondió que se dedicaba al Desarrollo de Aplicaciones Computacionales, el 17.12% respondió que se dedicaba a la Consultoría ó Auditorias Informáticas, el 14.41% respondió que se dedicaba a la Distribución de Aplicaciones

Computacionales, el 1.8% respondió que se dedicaba a Otras Actividades

Del 1.8% que respondió que se dedicaba a Otras aéreas, se distingue la siguiente: Venta de Hardware.

Las estadísticas correspondientes a esta pregunta pueden verse a continuación (Gráfico 4.1.2).

Gráfico 4.1.2. Estadísticas de la pregunta 2 sección I de la encuesta



3.- ¿Su empresa tiene algún tipo de certificación?

a) Si
b) No
c) En proceso

Los datos obtenidos de esta pregunta son los siguientes: El 63.16% de las empresas no poseían una certificación, el 22.81% si poseían un tipo de certificación, en cambio que el 14.04% se encontraban en proceso de certificación.

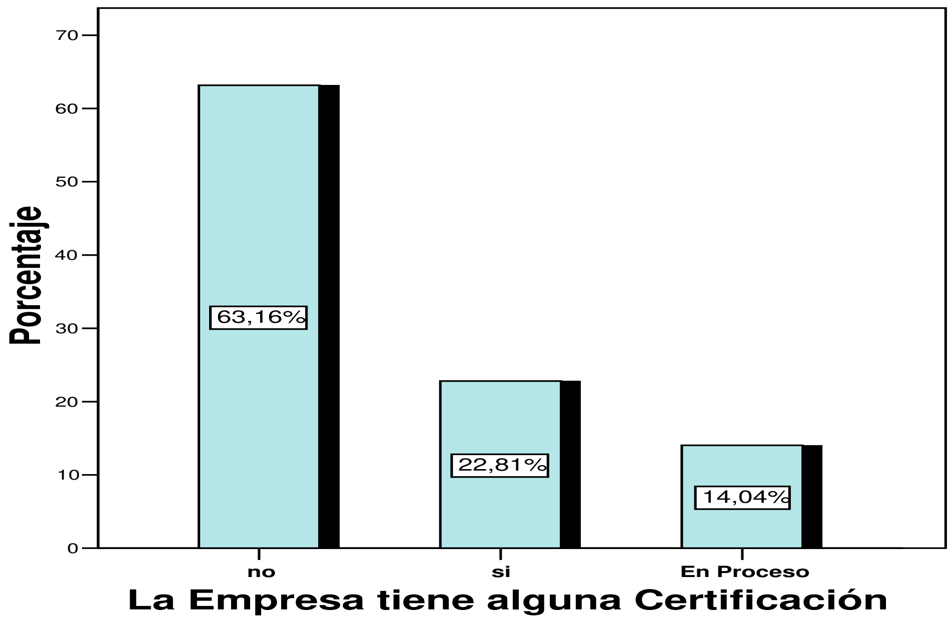
Del 22.81% de las empresas que respondieron que si poseían certificación se distinguen los siguientes tipos: ISO 9001:2000, Microsoft, SUN JAVA y ESRI.

Respecto al 14.04% de las empresas que respondieron que se encontraban en proceso de certificación indicaron se han iniciado las primeras gestiones correspondientes para la obtención de la misma, a continuación de detallan los tipos de certificaciones.

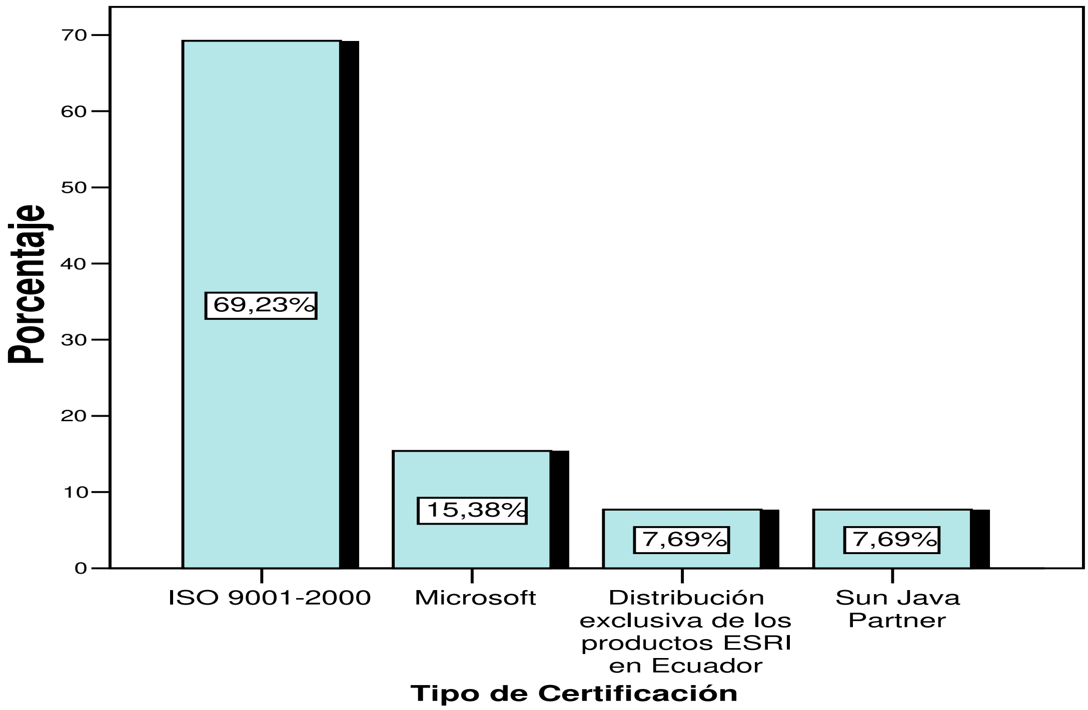
ISO 9001:2000, Microsoft y CMMI.

Las estadísticas correspondientes a esta pregunta pueden verse a continuación (Gráfico 4.1.3).

Gráfico 4.1.3. Estadísticas de la pregunta 3 sección I de la encuesta



Del 22.81% de empresas que respondieron que poseían una certificación se detallan en la grafica.



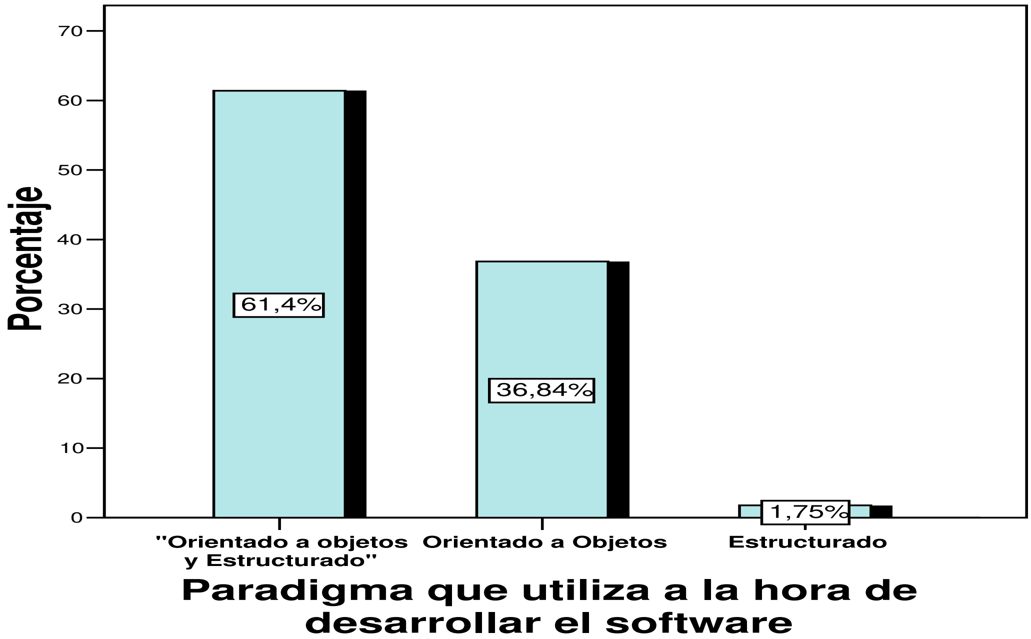
4.- En su empresa qué paradigma utilizan a la hora de desarrollar un software

a. Orientado a Objetos
b. Estructurado
c. Otro.

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera, el 61.4% respondieron que utilizan los dos paradigmas planteados tanto el Orientado a Objetos como el estructurado, el 36.84% respondió que utiliza el paradigma Orientado a Objetos, finalmente el 1.75% de las empresas respondió que utilizan el paradigma Estructurado.

Las estadísticas correspondientes a esta pregunta pueden verse a continuación (Gráfico 4.1.4).

Gráfico 4.1.4. Estadísticas de la pregunta 4 sección I de la encuesta

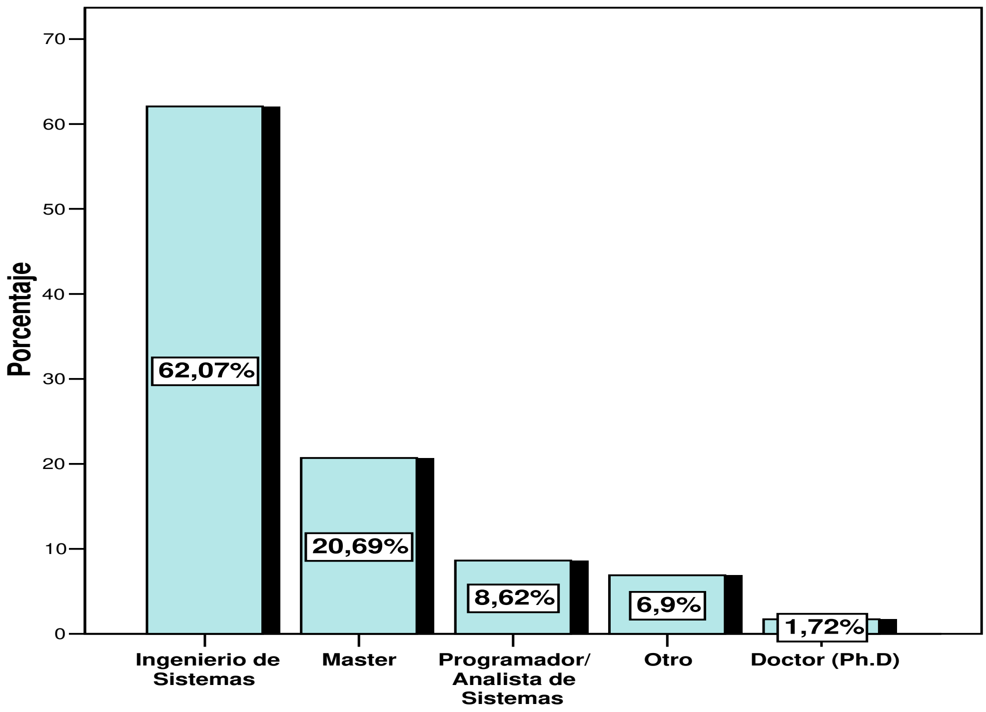


Sección II: **Información Entrevistado**

1.- Indique su nivel de educación más alto:

a.- Bachiller
b.- Programador/ Analista de sistemas
c.- Ingeniero en Sistemas
d.- Master
e.- Doctor (Ph.D.)
f.- Otro
Indique:____________________________

Con respecto a esta pregunta las empresas consultadas contestaron de la siguiente manera, 62.07% respondió que poseen un Titulo de Ing. en Sistemas, el 20.69% Master, 8.62% Analista Programador, el 6.9% Otro y el 1.72% respondió que poesía un titulo de Ph.D.



Porcentaje
70
60
50
40
30
20
10
0
62,07%
20,69%
8,62%
6,9%
1,72%
Ingenierio de Sistemas
Master
Programador/ Analista de Sistemas
Otro
Doctor (Ph.D)

ANEXO 5: Carta para las Empresas

La Escuela Superior Politécnica del Litoral (ESPOL), con el financiamiento del centro de Investigación Científica y Tecnológica (CICYT), se encuentran trabajando en un proyecto de investigación sobre metodologías de Software. Este proyecto esta orientado a brindar apoyo a la industria del Software en el Ecuador para la difusión, uso y evaluación de diferentes Métodos, Técnicas y Herramientas de Verificación y Validación que garanticen la calidad en el proceso de Desarrollo de Software.

Por lo expuesto solicitamos una entrevista con el Gerente de Sistemas/Director de Proyectos de su empresa para la realización de una encuesta que será de mucha utilidad para conocer las necesidades actuales de las empresas de desarrollo de Software de nuestro medio; y en base a este conocimiento generar nuevos proyectos que cubran estas necesidades.

Atentamente,

Ing. Verónica Macias
Directora
Grupo de Investigación de Ingeniería de Software
FIEC-ESPOL

Freddy Tituana
Estudiante
FIEC-ESPOL