



UNIVERSIDAD AUTÓNOMA
DE AGUASCALIENTES

CENTRO DE CIENCIAS BÁSICAS

MAESTRÍA EN CIENCIAS EXACTAS, SISTEMAS Y DE LA
INFORMACIÓN

ÁREA: INGENIERÍA DE SOFTWARE

TESIS

Identificación de las Mejores Prácticas de Ingeniería de
Software Utilizadas por las Micro, Pequeñas y Medianas
Empresas Desarrolladoras de Software en el Estado de
Aguascalientes

PRESENTA

L.I. Gustavo Ponce Chavoyo

COMITÉ DE TESIS

Director de Tesis: Dra. Laura Garza González

Asesor: Dr. Juan Muñoz López

Asesor: Dr. Jaime Muñoz Arteaga

Cd. Universitaria, Agosto 2010



Centro de Ciencias Básicas



L.I. GUSTAVO PONCE CHAVOYO
ALUMNO (A) DE LA MAESTRÍA EN CIENCIAS
EXACTAS, SISTEMAS Y DE LA INFORMACIÓN
CON ESPECIALIDAD EN INGENIERÍA DE SOFTWARE
P R E S E N T E .

Estimado (a) Sr. (ta) Ponce:

Por medio de este conducto me permito comunicar a Usted que habiendo recibido los votos aprobatorios de los revisores de su trabajo de tesis titulada **“Identificación de las Mejores Prácticas de Ingeniería de Software utilizadas para las Micro, Pequeñas y Medianas Empresas desarrolladoras de Software en el Estado de Aguascalientes”**, de acuerdo con su contenido y para dar cumplimiento a lo establecido en el Artículo 162 (Fracciones I, II y III) y Artículo 175 (Fracciones I y II) del Reglamento de docencia de la U.A.A., me permito comunicarle que puede imprimir dicho documento y continuar con los trámites para la presentación de su examen de grado.

Sin otro particular me permito saludarle muy afectuosamente.

A T E N T A M E N T E
Aguascalientes, Ags., 6 de agosto de 2010
“SE LUMEN PROFERRE”
EL DECANO

DR. FRANCISCO JAVIER ÁLVAREZ RODRÍGUEZ



c. c. p.- Archivo

FJAR,mjda



Dr. Francisco Javier Álvarez Rodríguez:
Decano del Centro de Ciencias Básicas

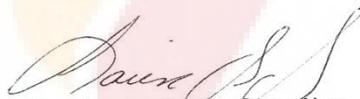
Por medio de este conducto, autorizo al tesista:

L.I. GUSTAVO PONCE CHAVOYO

La impresión de su documento de tesis con título **“Identificación de las Mejores Prácticas de Ingeniería de Software utilizadas por las Micro, Pequeñas y Medianas Empresas desarrolladoras de software en el Estado de Aguascalientes”**, ya que cumple con los requisitos de contenido y forma exigidas en la Universidad Autónoma de Aguascalientes.

Sin más por el momento, quedo a sus órdenes para cualquier aclaración al respecto.

ATENTAMENTE



Dra. Laura Garza González
Director de Tesis



Dr. Juan Muñoz López
Asesor de Tesis



Dr. Jaime Muñoz Arteaga
Asesor de Tesis

Aguascalientes, Ags. Agosto 2010

AGRADECIMIENTOS.

Agradezco a Dios, a mis padres, hermanas, familia y amigos por el apoyo que me han brindado en todos los proyectos que he emprendido en mi vida.

A mi directora de tesis, asesores, maestros y compañeros de clases que gracias a sus críticas, opiniones y consejos ayudaron a mejorar y enriquecer este trabajo de investigación.

A la Universidad Autónoma de Aguascalientes en la cual me desarrolle como profesionista y que hoy me da la oportunidad de culminar mis estudios de maestría.

A la Maestra Paty Rangel Jiménez por sus consejos y asesorías en el trabajo de tratado de datos e interpretación estadística.

A cada una de las empresas que mostraron interés en participar en la investigación, esperando que los resultados les sean de gran utilidad.

Al Consejo Nacional de Ciencia y Tecnología (CONACYT) por su apoyo durante mis estudios de maestría.

RESUMEN.

El presente trabajo de investigación comprende el estudio sobre mejores prácticas de ingeniería de software implementadas por la industria de software en el estado de Aguascalientes. El estudio fue llevado a cabo en organizaciones que cumplieron con el perfil: “Micro, Pequeñas y Medianas Empresas (MiPyMEs) que tienen como actividad principal el desarrollo de software”. Fue realizado con el objetivo de conocer si en ciertas etapas de desarrollo de sus productos existe una menor o mayor implementación de mejores prácticas, conocer el comportamiento de utilización de mejores prácticas en base a las características demográficas de las organizaciones y finalmente conocer si existe interés por la implementación de un modelo de trabajo de mejores prácticas a corto plazo.

El estudio fue realizado a través del diseño, validación y aplicación de un cuestionario en el cual se midieron un total de 31 mejores prácticas basadas en su mayoría en los niveles 2 y 3 de maduración del Modelo de Capacidad y Madurez Integrado (CMMI), cada una de estas prácticas fue categorizada por la etapa en la que son utilizadas. Los resultados mostraron una mayor utilización en las organizaciones que cuentan con un modelo de trabajo de mejores prácticas sin embargo el comportamiento general de la industria de software demostró una alta utilización durante la fase de administración de proyectos de software, una utilización promedio en las fases de análisis, diseño y codificación de software y una muy baja utilización de mejores prácticas en la etapa de pruebas de software.

ÍNDICE.

1. INTRODUCCIÓN..... 1

1.1 Descripción del Contexto del Problema de Investigación..... 1

1.2 Relevancia y Justificación de la Investigación..... 3

1.3 Descripción General de Contribuciones de la Investigación..... 5

1.4 Descripción General del Enfoque, Métodos y Técnicas de Investigación..... 5

2. ESTRUCTURACIÓN DE LA INVESTIGACIÓN..... 6

2.1 Descripción Específica del Problema de Investigación..... 6

2.2 Objetivos de la Investigación..... 7

2.3 Hipótesis de la Investigación..... 7

2.4 Descripción Específica del Enfoque, Métodos y Técnicas Específicas de Investigación Usadas..... 8

3. MARCO TEÓRICO..... 11

3.1 Revisión de Teorías Bases Usadas..... 11

3.1.1 Micro, Pequeñas y Medianas Empresas..... 11

3.1.2 Ingeniería de Software..... 13

3.1.3 Mejores Prácticas..... 14

3.1.4 Marcos de Trabajo de Mejores Prácticas..... 16

3.1.5 Ciclo de vida de Software..... 18

3.1.6. Gestión de Proyectos de Software..... 19

3.1.7 Análisis de Software..... 20

3.1.8 Diseño de Software..... 22

3.1.9 Codificación de Software..... 22

3.1.10 Pruebas de Software..... 23

3.2 Revisión de Principales Trabajos Relacionados..... 24

3.3 Resumen de Contribuciones y Limitaciones de Trabajos Relacionados..... 25

3.4 Comparación Trabajos Relacionados, Trabajo Propuesto..... 26

4. DESARROLLO Y VALIDACIÓN..... 27

4.1 Diseño y Desarrollo de Solución..... 27

4.2 Pruebas y Validación..... 31

5. REPORTE Y DISCUSIÓN DE RESULTADOS..... 38

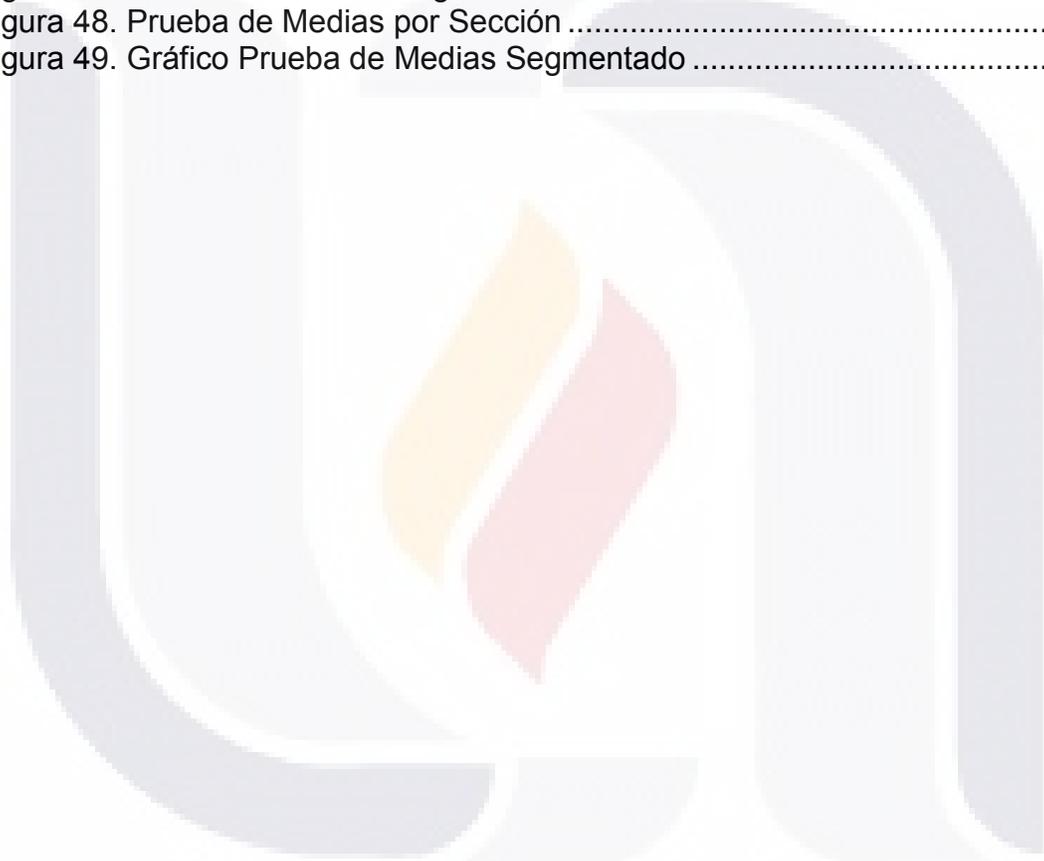
5.1 Técnicas y Análisis de Resultados.....	38
5.2 Descripción de Contribuciones.	40
5.3 Limitaciones y Problemas Encontrados.....	74
6. CONCLUSIONES.....	75
6.1 Conclusiones de Resultados Obtenidos.....	75
6.2 Conclusiones del Método de Investigación Empleado.	79
6.3 Conclusiones de Aprendizaje Personal.	79
6.4 Conclusiones y Recomendaciones Finales.	80
BIBLIOGRAFÍA.....	81
ANEXOS.....	84
A.1 Empresas de TI por Estado Registradas en México.	84
A.2 Software Best Practice (ESSI) Questionnaire	85
A.3 Investigación sobre las prácticas de ingeniería de software en México	91
A.4 Software Engineering Practices in Singapore	95
A.5 Instrumento de Evaluación.....	98
A.6 Listado de Empresas	104
ÍNDICE DE TABLAS	
Tabla 1. Empresas de Aguascalientes con modelo de mejores prácticas.....	2
Tabla 2. Perfil de la MiPyME en México	12
Tabla 3. Clasificación de las Micro, Pequeñas y Medianas Empresas	12
Tabla 4. Tipos de Requerimientos de Software	21
Tabla 5. Estudios Relacionados	24
Tabla 6. Contribuciones y Limitaciones de Estudios Relacionados.....	25
Tabla 7. Comparación Trabajos Relacionados-Propuesto	26
Tabla 8. Variables de Medición.....	30
Tabla 9. Totales de Variables del Instrumento de Evaluación.....	30
Tabla 10. Análisis de Confiabilidad – Administración de Proyectos de Software... ..	32
Tabla 11. Análisis de Confiabilidad – Análisis de Software	33
Tabla 12. Análisis de Confiabilidad – Diseño de Software.....	34
Tabla 13. Análisis de Confiabilidad – Codificación de Software	35
Tabla 14. Análisis de Confiabilidad – Pruebas de Software	36
Tabla 15. Correlaciones Tamaño de las Organizaciones	63
Tabla 16. Correlaciones Tamaño de las Organizaciones por Secciones.....	64
Tabla 17. Correlaciones Tiempo de Existencia de las Organizaciones	65
Tabla 18. Correlaciones Tiempo de Existencia por Sección.....	65
Tabla 19. Porcentajes, Interés de Implementación.....	67

Tabla 20. Prueba K-Muestras Independientes Kruskal-Wallis.....	70
Tabla 21. Estadísticos de Contraste.....	70
Tabla 22. Prueba K-Muestras Independientes Kruskal-Wallis.....	72
Tabla 23. Estadísticos de Contraste.....	73
Tabla 24. Mejores Prácticas en Orden de Utilización	76

ÍNDICE DE FIGURAS

Figura 1. Clasificación de empresas del estado de Aguascalientes.....	3
Figura 2. Tamaño de empresas de Tecnologías de Información.....	4
Figura 3. Modelo de Investigación	5
Figura 4. Metodología de Investigación	8
Figura 5. Empleado con los que cuenta la organización.....	41
Figura 6. Empleados Involucrados en Ingeniería de Software.....	41
Figura 7. Tiempo de existencia de la organización	42
Figura 8. Implementación actual de modelo de mejores prácticas.....	42
Figura 9. Planeación de implementación de modelo de mejores prácticas.....	43
Figura 10. Asignación de Líder de Proyecto	44
Figura 11. Costo y tiempo del desarrollo.....	44
Figura 12. Capacitación a nuevos elementos del proyecto.....	45
Figura 13. Monitoreo de avance de proyecto.....	45
Figura 14. Revisiones periódicas	46
Figura 15. Métricas sobre el desempeño del producto	46
Figura 16. Beneficios y viabilidad del proyecto	47
Figura 17. Documento estandarizado para obtención de requerimientos.....	48
Figura 18. Proceso definido en cambio de requerimientos	48
Figura 19. Definición de ciclos de desarrollo.....	49
Figura 20. Modelación de requerimientos	49
Figura 21. Tecnologías existentes del mercado	50
Figura 22. Diseño de interfaces y reportes	51
Figura 23. Normalización de bases de datos	51
Figura 24. Utilización de diccionarios de datos	52
Figura 25. Notaciones formales para el diseño del proyecto	52
Figura 26. Utilización de patrones de arquitectura y diseño.....	53
Figura 27. Utilización de herramientas CASE	53
Figura 28. Acoplamiento y cohesión del diseño	54
Figura 29. Control de versiones	55
Figura 30. Prototipado del producto	55
Figura 31. Ambientes de desarrollo.....	56
Figura 32. Control del código fuente	56
Figura 33. Documentación técnica.....	57
Figura 34. Estándares de codificación	57
Figura 35. Plan de pruebas.....	58
Figura 36. Pruebas independientes	59

Figura 37. Revisiones de pares.....	59
Figura 38. Pruebas con usuarios finales	60
Figura 39. Uso de casos de prueba	60
Figura 40. Herramientas automatizadas de pruebas	61
Figura 41. Correlación Tamaño de la Organización, Gestión de Proyectos	64
Figura 42. Correlación Tiempo de Existencia de la Organización	66
Figura 43. Interés de Implementación Modelo de Mejores Prácticas.....	67
Figura 44. Contingencia Tamaño de la Empresa e Interés de Implementación	68
Figura 45. Contingencia Tiempo de Existencia, Interés de Implementación	68
Figura 46. Prueba de Medias por Sección	69
Figura 47. Gráfico de Medias Segmentado Sección/Modelo	71
Figura 48. Prueba de Medias por Sección	72
Figura 49. Gráfico Prueba de Medias Segmentado	73



TESIS TESIS TESIS TESIS TESIS

1. INTRODUCCIÓN.

1.1 Descripción del Contexto del Problema de Investigación.

Las condiciones actuales de mercado exigen a la industria de software en todo el mundo cambiar la forma en que realiza la producción de sus sistemas informáticos, enfocándose en el control de los procesos involucrados en el ciclo de desarrollo de software y el mejoramiento de calidad. Las organizaciones mexicanas que han logrado posicionarse como proveedoras de servicios de tecnologías de información a nivel mundial, han implementado a partir de marcos de trabajo tales como Capability Maturity Model (CMM), Capability Maturity Model Integration (CMMI) y Modelo de Procesos para la Industria del Software (MoProSoft) un conjunto de **mejores prácticas** para facilitar, controlar y mejorar la manera en que efectúan sus productos.

El gobierno mexicano busca aumentar la competitividad del país teniendo como una de sus principales estrategias generar una industria exitosa de software, por lo que dentro del **Plan Nacional de Desarrollo** [1], se plantea el fomento a la industria y el mercado de Tecnologías de la Información, teniendo definidas las siguientes metas para el año 2013:

- Lograr una producción anual de software de 5,000 millones de dólares.
- Alcanzar el promedio mundial de gasto en tecnologías de información.
- Convertir a México en el líder latinoamericano de desarrollo de software y contenidos digitales en español.

A partir de las siguientes estrategias:

1. Promover las exportaciones y la atracción de inversiones.
2. Educación y formación de personal competente en el desarrollo de software, en cantidad y calidad convenientes.
3. Contar con un marco legal promotor de la industria.
4. Desarrollar el mercado interno.
5. Fortalecer a la industria local.
6. Alcanzar niveles internacionales en capacidad de procesos.
7. Promover la construcción de infraestructura básica y de telecomunicaciones.

Conforme al Directorio de Empresas de Tecnologías de Información (**DETI**) [2], Aguascalientes ocupa el octavo lugar con **49 empresas** registradas como proveedoras de servicios de TI (**Anexo A.1**) de las cuales 3 cuentan con algún modelo de mejora de procesos implementado.

Empresa	Modelo	Nivel de Madurez
Active Intelligence	CMM	3
Hildebrando Software Factory	CMMI	5
Softtek	CMMI	5

Tabla 1. Empresas de Aguascalientes con modelo de mejores prácticas.

En el estado de Aguascalientes acorde al Sistema de Información Empresarial Mexicano (**SIEM**), se cuentan con 6,784 empresas de las cuales el 99% caen en la clasificación de micro, pequeñas o medianas empresas [3].

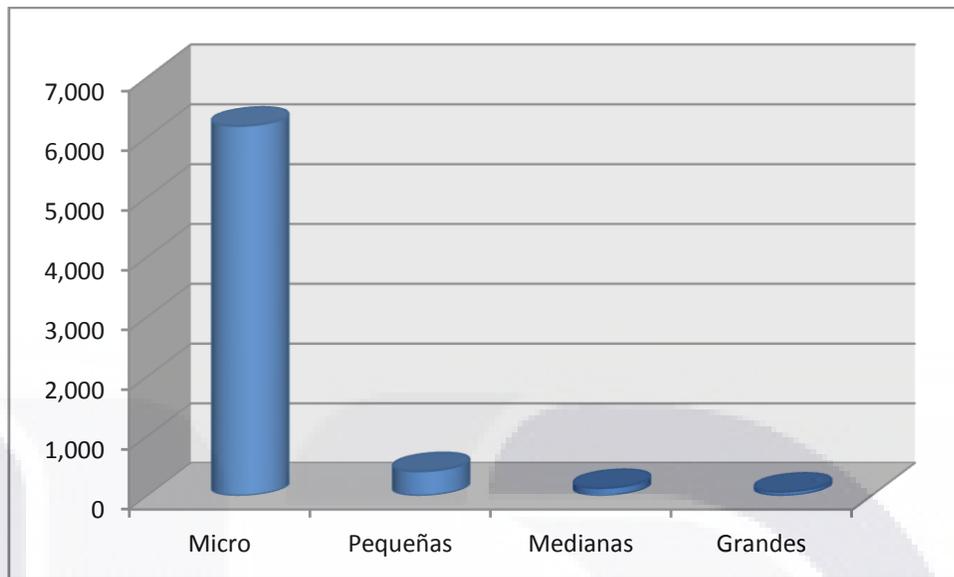


Figura 1. Clasificación de empresas del estado de Aguascalientes.

El uso de mejores prácticas de manera ordenada y guiada dentro de las MiPyMEs puede llegar a ser una manera efectiva de crecimiento, con la finalidad de buscar un mayor nivel de competitividad a nivel nacional y mundial.

1.2 Relevancia y Justificación de la Investigación.

Dentro del DETI se encuentran registradas 49 empresas en el estado de Aguascalientes de las cuales tres cuentan con algún marco de trabajo de **mejores prácticas** implementado, lo que significa que aproximadamente solo el 6% de la industria de software en el estado realiza de manera formal el uso de mejores prácticas. A nivel nacional se refleja el mismo comportamiento debido a que solo se alcanza el 8% en este rubro [2].

De acuerdo a estudios realizados por la Subsecretaría de Industria y Comercio y la Secretaría de Economía dentro del Programa para el Desarrollo de la Industria de Software Existen 2,095 Empresas de Tecnologías de Información de las cuales el

83% cuenta con menos de 50 empleados y el 81% de estas empresas requiere mejora en sus procesos [4].



Figura 2. Tamaño de empresas de Tecnologías de Información.

Las micro, pequeñas y medianas empresas (**MiPyMEs**) mexicanas desarrolladoras de software se enfrentan a la problemática de no contar con los elementos necesarios para la implementación de modelos de calidad, debido al esfuerzo y recursos que esto implica.

Este trabajo de investigación busca identificar las mejores prácticas utilizadas por las MiPyMEs del estado de Aguascalientes desarrolladoras de software a partir de la generación, validación y aplicación de un instrumento de evaluación basado en las mejores prácticas propuestas en los niveles 2 y 3 de CMMI y en tres investigaciones relacionadas al tema propuesto en esta investigación, con la finalidad de proporcionar un panorama del nivel de implementación de las mejores prácticas durante las fases de análisis, diseño, codificación, pruebas y administración de proyectos de software así como la relación que existe en base a las características demográficas de las organizaciones.

1.3 Descripción General de Contribuciones de la Investigación.

En el presente trabajo de investigación buscara llegar a resultados sobre el estado en el que se encuentra la implementación de mejores prácticas de ingeniería de software en las empresas que cumplen con el perfil de ser MiPyMEs en la industria de software en el estado de Aguascalientes, con la finalidad de que esto pueda llegar a ser útil en la creación y crecimiento de este tipo de organizaciones así como tener un punto de referencia para futuras investigaciones sobre mejores prácticas.

1.4 Descripción General del Enfoque, Métodos y Técnicas de Investigación.



Figura 3. Modelo de Investigación

Generación de Instrumento de evaluación en base a las prácticas específicas de los niveles 2 y 3 de CMMI y de tres investigaciones relacionadas al trabajo presente de investigación.

Validación a partir de la respuesta de los alumnos de la materia de Ingeniería de Software de la Universidad Autónoma de Aguascalientes, realizando un análisis con la técnica de confiabilidad de consistencia interna KR-20.

La investigación es de carácter descriptivo llevándose a cabo con el proceso de investigación cuantitativa propuesto por Robert Hernández Sampieri.

2. ESTRUCTURACIÓN DE LA INVESTIGACIÓN.

2.1 Descripción Específica del Problema de Investigación.

Existe poco interés mostrado por la industria de software nacional en la implementación de marcos de trabajo formales para la utilización de mejores prácticas de ingeniería de software tales como CMM, CMMI o MoProSoft debido a los recursos y esfuerzo que esto implica.

La clave para lograr un crecimiento en la industria de software nacional y ampliar los servicios proporcionados fuera de nuestras fronteras es buscar implementar mejores prácticas, las cuales se reflejen en el control y mejora de los procesos que están involucrados en el ciclo de desarrollo de un producto de software.

Este trabajo de investigación busca identificar en las MiPyMEs las mejores prácticas de ingeniería de software utilizadas en la generación de sus productos y/o servicios en base a las fases de desarrollo (análisis, diseño, codificación y pruebas), la gestión de proyectos y la información demográfica de las empresas.

Con la finalidad de que los resultados obtenidos lleguen a ser útiles en el crecimiento de la industria de software nacional.

2.2 Objetivos de la Investigación.

Objetivo Principal:

1. Identificar un panorama sobre las mejores prácticas implementadas por las MiPyMEs desarrolladoras de software en el estado de Aguascalientes

Objetivos Específicos:

1. Identificar la influencia de la implementación de mejores prácticas en base a la estructura organizacional de las empresas.
2. Conocer si existe un interés por implementar un marco de trabajo formal de mejores prácticas a corto plazo (Menor a 2 años).
3. Conocer si existe una mayor implementación de mejores prácticas en cierta etapa del ciclo de desarrollo del producto.

2.3 Hipótesis de la Investigación.

H1: La utilización de mejores prácticas de ingeniería de software está influenciada por el tamaño de la organización.

H2: La utilización de mejores prácticas de ingeniería de software está influenciada por el tiempo de existencia de la organización.

H3: Existe interés por implementar un marco de trabajo de mejores prácticas en un plazo menor a 2 años.

H4: Existen diferencias significativas en la utilización de mejores prácticas de ingeniería de software entre las fases de desarrollo de un producto.

2.4 Descripción Específica del Enfoque, Métodos y Técnicas Específicas de Investigación Usadas.

La investigación que se realizo es de carácter cuantitativo con un alcance de tipo descriptivo los cuales consisten en describir fenómenos, situaciones, contextos y eventos detallando como son y se manifiestan. [5]

Los estudios descriptivos buscan especificar las propiedades, características y los perfiles de personas, grupos, comunidades, procesos, objetos o cualquier otro fenómeno que se someta a un análisis. [6]

La metodología de investigación que se llevara a cabo en el presente trabajo es el proceso de investigación cuantitativa propuesto por Roberto Hernández Sampieri en su libro Metodología de la Investigación. [5] A continuación se bosqueja y describe cada una de sus etapas.



Figura 4. Metodología de Investigación

Planteamiento del Problema: En esta etapa se busca definir los objetivos de la investigación, las preguntas de investigación, la justificación de la investigación, la viabilidad de la investigación y la evaluación de deficiencias en el conocimiento del problema.

Elaboración del Marco Teórico: En esta etapa se realizan actividades que consisten en la detección de la literatura y otros documentos, inicio de revisión de dicha literatura, consulta en profundidad de la literatura y finalmente la extracción y recopilación de la información de interés de la literatura.

Definición del Alcance de la Investigación: Esta etapa busca que se defina el alcance y propósito de la investigación a partir de los siguientes estudios que se pueden llevar a cabo: estudios exploratorios, estudios descriptivos, estudios correlacionales, estudios explicativos o una combinación de los anteriormente mencionados.

Formulación de Hipótesis: En esta etapa se definen las hipótesis a partir de los objetivos y preguntas propuestas en el trabajo de investigación. Las hipótesis pueden ser de tipo nulas, alternativas o estadísticas.

Selección de la Muestra: En esta etapa se realizan actividades para delimitar la población de estudio, selección de la muestra, definir el tamaño óptimo de la muestra e identificar el procedimiento ideal de selección para la muestra (tómola, random, selección sistemática).

Recolección de Datos: Esta es una de las etapas más importantes de la

metodología de investigación ya que se debe realizar la recolección de datos de manera válida y confiable. En esta etapa se realizan las siguientes actividades: redefiniciones fundamentales, revisión enfocada en la literatura, identificación del dominio de las variables a medir y sus indicadores, decisiones claves, construcción del instrumento, prueba piloto, versión final, entrenamiento del personal, autorizaciones y administración del instrumento.

Análisis de Datos: En esta fase se busca realizar el análisis de los datos recolectados de la fase anterior, la cual consta de las siguientes actividades: selección de programa de análisis (SPSS, MatLab, MiniTab, etc.), explorar datos, evaluar confiabilidad y validez lograda por el instrumento de medición, analizar mediante pruebas estadísticas las hipótesis planteadas, realizar análisis adicionales y preparar los resultados para presentarlos.

Reporte de Resultados: Esta fase es la última de la metodología de investigación, la cual tiene como objetivo presentar un documento con los resultados obtenidos durante la investigación el cual contiene el comportamiento de las hipótesis, gráficas, tablas, conclusiones y trabajos futuros.

3. MARCO TEÓRICO.

3.1 Revisión de Teorías Bases Usadas.

3.1.1 Micro, Pequeñas y Medianas Empresas

Las Micro, Pequeñas y Medianas Empresas mexicanas, comúnmente conocidas como MiPyME, representan el 99% de las empresas del país constituyéndose como generadoras de un gran número de empleos, así como la base de la economía [6].

A continuación se presenta el perfil de las MiPyMEs en México:

FACTORES	RESULTADOS
Datos estructurales	Los criterios para establecer el tamaño de una empresa gozan de poco consenso. La inflación cambia los valores financieros o de ventas con mucha rapidez. En las empresas pequeñas y medianas la estructura orgánica es familiar.
Tecnología	La mayor parte utiliza sistemas de producción tradicional, 40% ya está aplicando procesos automatizados.
Tipo de dirigentes	El típico director es un hombre de 44 años, con 14 años de escolaridad y es empresario por herencia. La actitud del dueño-administrador en 50% de los casos indico que el familismo es negativo para las empresas. Algunas empresas mueren con su fundador.
Medio ambiente	Su medio es complejo y sostiene relaciones con otras organizaciones: proveedores, clientes, competencia, asociaciones, bancos, gobierno y despachos profesionales de consultaría.
Problemática a la que deben Enfrentarse	En un orden de importancia atribuida, a juicio de los Empresarios son ocho los problemas más difíciles que actualmente encaran en sus organizaciones: Recursos humanos, deficiencia del gobierno (demasiados controles), falta de seriedad de los proveedores, financiamiento, materias primas (calidad y escasez), mercados, competencia y deficiente organización.
Valores y objetivos	Para la mayoría de los directores el objetivo principal es maximizar las utilidades. Los procedimientos administrativos son sumamente personalizados, es decir, consisten en la vigilancia estrecha sobre las operaciones. La toma de decisiones es centralizada y la ejecución la realizan Jefes de Área, a quienes se les responsabiliza por los resultados obtenidos.

FACTORES	RESULTADOS
Administración de personal	Esta área en un gran problema, la realidad es que existe una actitud muy pasiva en el manejo del personal. El estudio indicó que dentro del reclutamiento, Selección, contratación, y capacitación, se aplican soluciones informales (sin técnica). Sobre la ley de capacitación hay gran escepticismo.
Procesos administrativos	Lo relevante es estar al día acerca de productos y tecnología. Los datos los obtienen mediante: visitas al extranjero, revistas especializadas, información de sus clientes, sobre técnicas administrativas las adquieren por cursos, los aspectos financieros los obtienen con otros industriales y banqueros.

Tabla 2. Perfil de la MiPyME en México [7]

En seguida se presentan los criterios de clasificación para las MiPyMEs acorde con el Sistema de Información Empresarial Mexicano:

Sector	Clasificación	Número de empleados
Servicios	Micro	0 – 10
	Pequeña	11– 30
	Mediana	31 – 100
	Grande	Más de 100
Industrial	Micro	0 – 10
	Pequeña	11 – 50
	Mediana	51 – 250
	Grande	Más de 250
Comercio	Micro	0 – 10
	Pequeña	11 – 50
	Mediana	51 – 250
	Grande	Más de 250

Tabla 3. Clasificación de las Micro, Pequeñas y Medianas Empresas [8]

La importancia de la micro, pequeña y mediana industria no solo puede medirse por el número de establecimientos, también por el capital invertido que representan, el valor de su producción, el valor agregado, las materias primas que consumen, la formación de capital fijo, los empleos que generan y la capacidad de

compra que dan a la población trabajadora mediante los sueldos y salarios [7]. Por ende podemos concluir que las MiPyMEs no solo se definen por su números de trabajadores, si no que están involucrados diversos factores que caracterizan este tipo de organizaciones.

3.1.2 Ingeniería de Software

Los sistemas informáticos se han convertido en piezas fundamentales en el crecimiento de las organizaciones. El alcance de dichos sistemas ha significado un importante cambio en la forma de vida y trabajo actual debido al gran número de procesos que se están sistematizando, todo esto ha llevado a exigir a la industria de software sistemas con calidad que cumplan las expectativas de los usuarios y/o cliente y que se fabriquen dentro de lo planeado en tiempo y recursos. La ingeniería de software es una disciplina que surge a partir de estas necesidades enfocándose en la utilización de herramientas y buenas prácticas para lograr sus objetivos.

Existen diversos autores que han definido a la ingeniería de software a continuación se listan los más reconocidos e importantes:

- La ingeniería de software es el establecimiento y uso de principios sólidos de la ingeniería para obtener económicamente un software confiable y que funcione de modo eficiente en máquinas reales. [9]
- La aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento del software. [10]

- TESIS TESIS TESIS TESIS TESIS
- La ingeniería de software es una disciplina de la ingeniería que comprende todos los aspectos de la producción de software desde las etapas iniciales de la especificación del sistema, hasta el mantenimiento de éste después que se utiliza. [11]

Como se observa existen diversos enfoques que se le ha dado a la ingeniería de software, en donde algunos autores puntualizan en su perspectiva la importancia de ciertos aspectos en relación con otros. En base a las definiciones previamente listadas, esta investigación propone una nueva definición donde se abarcan aspectos de calidad, planeación y control del desarrollo del producto.

“La ingeniería de software es una disciplina perteneciente al área de tecnologías de la información que tiene como finalidad el desarrollo de sistemas informáticos involucrando metodologías de desarrollo, herramientas de ayuda y buenas prácticas que impacten en la generación de productos que cumplan con la calidad y expectativas de los usuarios y/o clientes dentro de los tiempos y con los recursos planeados.”

3.1.3 Mejores Prácticas

Las buenas prácticas son aquellas actividades que con el paso de tiempo han demostrado generar un resultado satisfactorio, estas existen en todos los ámbitos y por lo regular se definen de manera general para que puedan ser aplicadas en diferentes contextos.

Dentro de la ingeniería de software las buenas prácticas han sido identificadas por

TESIS TESIS TESIS TESIS TESIS

grandes organizaciones como Microsoft, Instituto de Ingenieros en Electricidad y Electrónica (IEEE), Sun Microsystems, Instituto de Ingeniería de Software, por mencionar algunos.

Las buenas prácticas se han popularizado debido a los modelos de mejora de procesos ya que estos son un marco de trabajo basado en su uso.

Las actividades que envuelven las buenas prácticas de la ingeniería de software se enfocan en la mejora gradual y continua en el proceso de desarrollo de software. El proceso de mejora de software no debe verse como una meta propia sino que deben estar claramente vinculadas a los objetivos comerciales de la organización. El proceso de mejora de software comienza atendiendo los aspectos organizacionales. Experiencias en el pasado han demostrado que antes de que se hagan inversiones en mejoras tecnológicas (a través de productos como herramientas de apoyo e infraestructura), se necesita resolver cuestiones esenciales del proceso. Se refieren a cómo en realidad se está desarrollando el software: la metodología, los métodos y sobre todo a la organización del proceso de desarrollo y mantenimiento de software.

Encontrar el correcto marco de trabajo organizacional, el correcto modelo de proceso, los correctos métodos o técnicas y la combinación adecuada de habilidades para el equipo de desarrollo es un asunto difícil y una meta a largo plazo de cualquier proceso de mejora. Sin embargo, es un requisito fundamental para el establecimiento de un bien definido y controlado proceso de desarrollo de software.

El desarrollo de software es un proceso llevado por personas, y debido a esta consideración se debe involucrar a todos los actores relacionados en el proceso. La mejora e implementación del proceso debe tener en cuenta todos los aspectos relacionados con las personas (factores humanos), estos son ortogonales a la tecnología y a la metodología por lo cual los enfoques son cruciales para el éxito de la adopción de mejores prácticas.

El éxito del cambio en la gestión incluye la motivación del personal, capacitación y promoción de las contribuciones positivas que el personal pueda realizar.

Los aspectos de personal cubren todos los diferentes grupos que están involucrados en los procesos del desarrollo del software incluido la gestión y los ingenieros de software.

Con el fin de garantizar un entorno adecuado para el éxito de la adhesión a un enfoque de calidad total es imperativo que la alta administración este plenamente consciente de todos los aspectos. Su compromiso y la participación son cruciales para el éxito de la aplicación del proceso de mejora y podría ser necesario aumentar su concientización respecto a esta cuestión. [12]

3.1.4 Marcos de Trabajo de Mejores Prácticas

En la búsqueda de tener una guía que ayudara a generar productos de software que cumplieran con la satisfacción de los clientes surgen los modelos de mejores prácticas y de procesos, los cuales definen un conjunto de mejores prácticas que tienen como objetivo definir, controlar y mejorar los procesos de la organización.

El Instituto de Ingeniería de Software (SEI), en los años ochenta inicio un estudio

de las formas de evaluar las capacidades de los proveedores de software. El resultado de estos estudios fue el “Modelo de Madurez de la Capacidad de Software del SEI” comúnmente conocido como CMM siendo este el primer modelo de mejores prácticas y mejora de procesos.

A partir de este modelo otras organizaciones buscaron desarrollar modelos semejantes surgiendo el modelo SPICE el cual es una aproximación a CMM pero que tiene la característica de ser más flexible e involucrar los procesos entre cliente y proveedor.

También surgió el proyecto Bootstrap el cual tenía como objetivo extender y adaptar el modelo CMM para hacerlo aplicable a un amplio número de compañías. En un intento de integrar la gran variedad de modelos que se habían desarrollado el SEI comenzó un proyecto para desarrollar un modelo de capacidad integrado (CMMI) el cual identifica 24 áreas de procesos en 5 diferentes niveles los cuales involucran metas y prácticas que se encargan del control y la mejora de los procesos. [13]

En México por iniciativa de la Secretaría de Economía y gracias al trabajo de académicos y empresarios mexicanos, encabezados por la Dra. Hanna Oktaba, profesora de la Facultad de Ciencias de la UNAM surge el Modelo de Procesos para la Industria del Software (MoProSoft) el cual es un modelo de procesos para la industria de software nacional, que fomenta la estandarización de operación a través de la incorporación de las mejores prácticas en gestión e ingeniería de software. [14]

3.1.5 Ciclo de vida de Software

El ciclo de vida de un software se traduce como las actividades y recursos que se encuentran involucrados en la generación del mismo. La mayor parte de los ciclos de vida de software se basan en uno de los tres siguientes modelos generales o paradigmas de desarrollo de software [11]:

1. **Enfoque en Cascada.** Representa como fases de proceso la especificación de requerimientos, el diseño del software, la implementación, pruebas. Después de que cada etapa queda definida se firma y el desarrollo continúa con la siguiente etapa.
2. **Desarrollo Iterativo.** Este enfoque entrelaza las actividades de especificación, desarrollo y validación. Un sistema inicial se desarrolla rápidamente a partir de especificaciones muy abstractas. Éste se refina basándose en las peticiones del cliente para producir un sistema que satisfaga las necesidades de dicho cliente. El sistema entonces puede ser entregado. De forma alternativa, se puede reimplementar utilizando un enfoque más estructurado para producir un sistema más sólido y mantenible.
3. **Basado en Componentes.** Esta técnica supone que las partes del sistema existen. El proceso de desarrollo del sistema se enfoca en la integración de estas partes más que desarrollarlas desde el principio.

A continuación un listado con los modelos de ciclos de vida de software

- Cascada pura
- Cascadas modificadas
 - Sashimi (cascada con fases solapadas)
 - Cascada con subproyectos
 - Cascada con reducción de riesgos
 - Cascada incremental
- Codificar y corregir
- Espiral
- Espiral ganar-ganar (winwinspiral)
- Prototipado evolutivo
- Entrega por etapas
- Diseño por planificación
- Entrega evolutiva
- Diseño por herramientas
- Software comercial existente (COTS)
- Métodos ágiles

3.1.6. Gestión de Proyectos de Software

La actividad de gestión en los proyectos de software se ha convertido en una parte esencial en el desarrollo de sistemas ya que una mala gestión usualmente lleva al fracaso del proyecto.

Somerville [13] comenta que los gestores de software son los responsables de la

planificación y temporalización del desarrollo de los productos. Supervisar el trabajo para asegurar que se lleva a cabo conforme al tiempo previsto y al presupuesto.

Las actividades de un gestor de software difieren enormemente de un proyecto a otro todo esto dependiendo de las políticas de la organización y el producto de software a desarrollar. Sin embargo la mayoría de los gestores de software tendrán como responsabilidad las siguientes actividades:

- Redacción de la propuesta
- Planificación y calendarización del proyecto
- Estimación de costos del proyecto
- Supervisión y revisión del proyecto
- Selección y evaluación del personal
- Redacción y presentación de informes

3.1.7 Análisis de Software

El análisis de un software implica las actividades de comprender las necesidades de los clientes. Comúnmente a este proceso se le conoce como Ingeniería de Requerimientos el cual se encarga de descubrir, analizar, documentar y verificar las peticiones del cliente.

Normalmente se realiza una clasificación de los requerimientos para tener un mejor entendimiento de estos, en la tabla 4 se presenta una descripción de los diferentes tipos existentes de acuerdo con Somerville [13].

Tipo de Requerimiento	Descripción
De Usuario	Son declaraciones, en lenguaje natural y en diagramas, de los servicios que se espera que el sistema proporcione y de las restricciones bajo las cuales debe funcionar.
De Sistema	Establecen con detalle las funciones, servicios y restricciones operativas del sistema. El documento de requerimientos del sistema (algunas veces denominado especificación funcional) debe ser preciso. Debe definir exactamente qué es lo que se va a implementar. Puede ser parte del contrato entre el comprador del sistema y los desarrolladores del software.
Funcionales	Son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que éste debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares. En algunos casos, los requerimientos funcionales de los sistemas también pueden declarar explícitamente lo que el sistema no debe hacer.
No Funcionales	Son restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo y estándares. Los requerimientos no funcionales a menudo se aplican al sistema en su totalidad. Normalmente apenas se aplican a características o servicios individuales del sistema.
De Dominio	Son requerimientos que provienen del dominio de aplicación del sistema y que reflejan las características y restricciones de ese dominio. Pueden ser funcionales o no funcionales.

Tabla 4. Tipos de Requerimientos de Software

3.1.8 Diseño de Software

El diseño de software es la etapa donde se crea una representación o modelo del software en el que se proporcionan detalles acerca de las estructuras de datos, las arquitecturas, las interfaces y los componentes del software que son necesarios para implementar el sistema. La importancia del diseño de software radica en que a partir de este se pueden hacer evaluaciones antes de generar código, realizar pruebas y que los usuarios finales se encuentren involucrados a gran escala. [15]

3.1.9 Codificación de Software

La codificación de software se refiere al proceso en el cual se sigue un conjunto ordenado de pasos para llegar a la solución de un problema u obtención de un producto. [15]

En la actualidad existen siete grandes categorías del software de computadora los cuales se listan a continuación:

- Software de sistemas
- Software de aplicación
- Software científico y de ingeniería
- Software empotrado
- Software de línea de productos
- Aplicaciones basadas en Web
- Software de inteligencia artificial

3.1.10 Pruebas de Software

En la fase de pruebas de software se tiene como objetivo descubrir errores cometidos sin darse cuenta al realizar el diseño y la codificación. Las pruebas deben planearse con anticipación y se realizan de manera sistemática.

Las pruebas del software constituyen un elemento de un tema más amplio que suele denominarse verificación y validación. Verificación es el conjunto de actividades que aseguran que el software implemente correctamente una función específica. Validación es un conjunto diferente de actividades que aseguran que el software construido corresponde con los requisitos del cliente. [15]

Boehm lo establece de la siguiente manera:

- Verificación: ¿Estamos construyendo el producto correctamente?
- Validación: ¿Estamos construyendo el producto correcto? [16]

A continuación un listado con las principales tipos de pruebas que se llevan a cabo a los sistemas informáticos:

- Pruebas del sistema
 - Pruebas de integración
 - Pruebas de entregas
 - Pruebas de rendimiento
- Pruebas de componentes
 - Pruebas de interfaces

Para llevar un control de la realización y evaluación de las pruebas de diseñan una serie de documentos en forma de plantillas llamados casos de prueba en los que

se especifica la logística que llevara dicha prueba. Primeo se selecciona una característica del sistema o componente, posteriormente se selecciona un conjunto de entradas que ejecutan dicha característica y finalmente se documentan las salidas esperadas o rangos de salida. [13]

3.2 Revisión de Principales Trabajos Relacionados.

Nombre del Estudio	Autor(es)
<ul style="list-style-type: none"> • Software Engineering in Europe: A Study of Best Practices. [17] 	<ul style="list-style-type: none"> - Soumitra Dutta - Michael Lee - Luk Van Wassenhove
<ul style="list-style-type: none"> • Software Engineering Practices in Singapore. [18] 	<ul style="list-style-type: none"> - Dany C.C Poo - Mui Ken Chung
<ul style="list-style-type: none"> • Investigación sobre las prácticas de ingeniería de software en México. [19] 	<ul style="list-style-type: none"> - María de la Luz Villalobos Hernández - Agustín Francisco Gutiérrez Tornés
<ul style="list-style-type: none"> • Prácticas de ingeniería de software e un entorno académico. [20] 	<ul style="list-style-type: none"> - Eugenia Marquez - Gabriela Gaetan - Viviana Saldaño - Patricia Yañez

Tabla 5. Estudios Relacionados

En la tabla 5 se presentan los estudios base los cuales fueron de utilidad en la realización de la presente investigación.

3.3 Resumen de Contribuciones y Limitaciones de Trabajos Relacionados.

Estudio	Contribuciones	Limitaciones
Software Engineering in Europe: A Study of Best Practices.	El resultado de la investigación ayude a las organizaciones desarrolladoras de software a mejorar el enfoque de su tiempo disponible en recursos como la ingeniería de software, sus administradores, líderes y planes para la mejora.	El tamaño de la población fue de 397 organizaciones de diferentes sectores en 20 países europeos, debido a una amplia variación de datos se tuvo que discriminar cierta información.
Software Engineering Practices in Singapore.	La mayoría de las organizaciones desarrolladoras de sistemas de software utilizan técnicas de Ingeniería de software. La utilización de dichas técnicas están concentradas en las fases de análisis y diseño. En las etapas de codificación, revisión/pruebas y mantenimiento la utilización de técnicas de ingeniería de software es muy baja. La utilización de herramientas CASE es bastante baja.	De las 240 organizaciones seleccionadas para participar en la investigación, solo 54 fueron respuestas validas
Investigación sobre las prácticas de ingeniería de software en México.	Las organizaciones que desarrollan software en México son en su mayoría pequeños y microempresarios que no se interesan por utilizar métodos de control de sus proyectos, su perspectiva radica en cumplir sólo con los requerimientos en las fechas estipuladas, sin ofrecer a los clientes servicios de valor agregado.	La población no representa todas las instituciones desarrolladoras de software. Se tuvo que ajustar el instrumento de evaluación debido a la falta de roles en las organizaciones. Se descartaron 28 evaluaciones por no estar correctamente contestadas quedando 72 en total.
Prácticas de ingeniería de software e un entorno académico.	Perspectivas de los estudiantes: Posibilidad de tratar con un cliente real. Apreciación de trabajo en equipo. Desarrollar dentro de un marco de un proceso de desarrollo. Falta de información sobre nuevas tecnologías. El esfuerzo real necesario para cumplir con las estimaciones de los proyectos.	Falta de tiempo para concluir los proyectos.

Tabla 6. Contribuciones y Limitaciones de Estudios Relacionados

En la tabla 6 se presentan las contribuciones y limitaciones de los estudios relacionados que fueron de utilidad en la realización de la presente investigación.

3.4 Comparación Trabajos Relacionados, Trabajo Propuesto.

Nombre del Estudio	Basado en algún modelo	Enfocado a la industria	Instrumento de Evaluación	Enfoque MiPyMEs	Longitudinal
• Software Engineering in Europe: A Study of Best Practices. [17]	X	✓	✓	X	✓
• Software Engineering Practices in Singapore. [18]	X	✓	✓	X	X
• Investigación sobre las prácticas de ingeniería de software en México. [19]	X	✓	✓	X	X
• Prácticas de ingeniería de software e un entorno académico. [20]	X	X	X	X	X
• Trabajo de Tesis, Mejores Prácticas de Ingeniería de Software.	✓	✓	✓	✓	X

Tabla 7. Comparación Trabajos Relacionados-Propuesto

En la tabla 7 se presenta una comparativa de los estudios relacionados con el presente trabajo propuesto analizando características de metodología, enfoque y tipo de investigación.

4. DESARROLLO Y VALIDACIÓN.

4.1 Diseño y Desarrollo de Solución.

La metodología de investigación que se propuso en este trabajo, busca identificar la utilización de mejores prácticas en la industria de software, basándose en la generación, validación y aplicación de un instrumento de evaluación el cual está dividido en las siguientes secciones:

- Datos demográficos (5 Preguntas)
- Administración de proyectos de software (6 Preguntas)
- Análisis de software (6 Preguntas)
- Diseño de software (7 Preguntas)
- Codificación de software (6 Preguntas)
- Pruebas de software (6 Preguntas)

La distribución de las secciones fue realizada en base a las fases en común con que cuentan los diversos modelos y técnicas de desarrollo de software donde se identificaron las fases de análisis, diseño, codificación y pruebas. La sección de administración de proyectos se adhirió debido a que esta es llevada en paralelo al desarrollo de los productos.

Cada una de estas secciones cuenta con una serie de preguntas las cuales midieron el nivel de utilización de mejores prácticas. El instrumento de evaluación fue elaborado a partir de los **niveles 2 y 3 del modelo CMMI** el cual es un marco de trabajo de mejores prácticas, también se tomaron en cuenta **3 investigaciones relacionadas con el tema** y finalmente se propusieron **4 preguntas** en la

investigación. A continuación la información a detalle de la construcción del instrumento de evaluación.

- CMMI, Nivel 2 y 3 (17 preguntas)
- Instituto Europeo de Software (9 Preguntas) (**Anexo A.2**)
- Instituto Politécnico Nacional, Investigación sobre las prácticas de ingeniería de software en México. (4 Preguntas) (**Anexo A.3**)
- Software Engineering Practices in Singapore, National University of Singapore (2 Preguntas) (**Anexo A.4**)
- 4 Preguntas Propuesta en la Investigación

Teniendo el instrumento evaluación un total de 36 preguntas. (**Anexo A.5**)

En la tabla 8 se presenta la tabla de las variables a medir:

Referencia	Sección / # Pregunta	Variable
European Software Institute	Datos demográficos/1	Empleados de la organización
European Software Institute	Datos demográficos/2	Empleados involucrados en la Ingeniería de Software
Instituto Politécnico Nacional	Datos demográficos/3	Tiempo de la organización
Instituto Politécnico Nacional	Datos demográficos/4	Marco de trabajo de mejores prácticas
Propuesta en la Investigación	Datos demográficos/5	Interés en implementar un marco de trabajo de mejores prácticas
European Software Institute	Gestión de Proyectos/1	Líder de proyecto
CMMI, Planeación del Proyecto(PP), SP 1.4 Estimar esfuerzo y costo del proyecto	Gestión de Proyectos/2	Estimación tiempo y costos
CMMI, Planeación del Proyecto(PP), SP 2.5 Planificar la adquisición de conocimiento y habilidades	Gestión de Proyectos/3	Capacitación
CMMI, Planeación del Proyecto(PP), SP 2.7 Establecer el plan del proyecto	Gestión de Proyectos/4	Planeación documentada
CMMI, Planeación del Proyecto(PP), SP 2.2 Identificar los riesgos del proyecto	Gestión de Proyectos/5	Riesgos de proyecto
European Software Institute	Gestión de Proyectos/6	Revisiones periódicas
CMMI, Medición y Análisis (MA), SP 1.2 Especificar métricas	Gestión de Proyectos/7	Métricas

Referencia	Sección / # Pregunta	Variable
European Software Institute	Análisis de Sistemas/1	Viabilidad y Beneficios del proyecto
CMMI, Desarrollo de Requerimientos (RD), SP 1.2 Desarrollar los Requerimientos del Cliente	Análisis de Sistemas/2	Obtención de requerimientos
CMMI, Administración de Requerimientos (REQM), SP 1.3 Administrar cambios a los requerimientos	Análisis de Sistemas/3	Cambio en los requerimientos
CMMI, Planeación del Proyecto(PP), SP 1.3 Definir el ciclo de vida del proyecto	Análisis de Sistemas/4	Ciclo de vida del proyecto
CMMI, Desarrollo de Requerimientos (RD), SP 3.3 Analizar Requerimientos	Análisis de Sistemas/5	Modelación de requerimientos
Instituto Politécnico Nacional	Análisis de Sistemas/6	Nuevas tecnologías
National University of Singapore	Diseño de Sistemas/1	Diseño de Interfaces
National University of Singapore	Diseño de Sistemas/2	Normalización de datos
European Software Institute	Diseño de Sistemas/3	Diccionarios de datos
CMMI, Solucion Técnica (TS), SP 3.2 Desarrollar la Documentación de Soporte, ESI (European Software Institute)	Diseño de Sistemas/4	Herramientas de diseño
Propuesta en la Investigación	Diseño de Sistemas/5	Patrones de diseño
European Software Institute	Diseño de Sistemas/6	Herramientas CASE
Propuesta en la Investigación	Diseño de Sistemas/7	Acoplamiento y Cohesión
CMMI, Administración de la Configuración(CM), SP 1.2 Establecer un sistema de administración de la configuración	Codificación/1	Control de cambios
CMMI, Solución Técnica (TS), SP 1.1 Desarrollar Soluciones Alternativas y Criterios de Selección	Codificación/2	Prototipado
CMMI, Integración del Producto (PI), SP 1.2 Establecer el Ambiente de Integración	Codificación/3	Ambientes de desarrollo
CMMI, Administración de la Configuración(CM), SP 2.1 Monitorear Pedidos de Cambio	Codificación/4	Control sobre el código fuente
CMMI, Solución Técnica (TS), SP 3.2 Desarrollar la Documentación de Soporte	Codificación/5	Documentación técnica
Propuesta en la Investigación	Codificación/6	Estándares (convenciones)

Referencia	Sección / # Pregunta	Variable
European Software Institute	Pruebas/1	Plan de pruebas
CMMI, Validación(VA), SP 1.2 Establecer el Ambiente de Validación	Pruebas/2	Pruebas independientes
CMMI, Verificación (VE), SP 2.2 Conducir la Revisión de Pares	Pruebas/3	Revisión de pares
CMMIm Validación(VA), SP 2.1 Realizar la Validación	Pruebas/4	Pruebas usuario final
Instituto Politécnico Nacional	Pruebas/5	Casos de pruebas
European Software Institute	Pruebas/6	Herramientas automatizadas de pruebas

Tabla 8. Variables de Medición

Variables a Medir a Partir del Instrumento de Evaluación
18 Preguntas CMMI (Nivel 2 y 3)
9 Preguntas ESI (European Software Institute)
4 Preguntas IPN, Investigación sobre las prácticas de ingeniería de software en México.
2 Preguntas Software Engineering Practices in Singapore, National University of Singapore
4 Preguntas Propuesta en la Investigación
37 Preguntas en Total

Tabla 9. Totales de Variables del Instrumento de Evaluación

En la tabla 9 se presenta la procedencia de cada una de las preguntas que forman parte del instrumento de evaluación.

El instrumento de evaluación busca medir la implementación de mejores prácticas en las MiPyMEs del estado de Aguascalientes desarrolladoras de software. No se busca saber el grado de satisfacción de su uso o el tiempo que las han utilizado simplemente saber si actualmente están implementadas en las empresas este conjunto de mejores prácticas.

Se aplicara sobre los datos estadística descriptiva para conocer el comportamiento de las mejores prácticas de manera individual. Se generaran tablas de contingencia para observar el comportamiento de las mejores prácticas de ingeniería de software implementadas en relación a las características

demográficas de las organizaciones.

Se evaluará a partir de índices de correlación el grado de independencia entre el tiempo de existencia de la organización, tamaño de la organización y la utilización de mejores prácticas para cada una de las secciones.

Las hipótesis serán evaluadas a partir de pruebas de hipótesis no paramétricas (k muestras independientes / Kurskal Wallis).

4.2 Pruebas y Validación.

Después de la generación de la primera versión formal del instrumento de evaluación (**Anexo A.5**) se realizó una validación con los alumnos de sexto semestre de la Licenciatura en Tecnologías de Información en la Universidad Autónoma de Aguascalientes donde se identificaron los siguientes aspectos a mejorar:

- Instrucciones mal redactadas para la primera sección.
- Pregunta 3 sección V redactar de manera más entendible.
- Pregunta 5 sección II mal redactada.
- Pregunta 5 sección V mal redactada.
- Pregunta 2 sección IV mal redactada
- Reubicar la sección de ayuda a la parte final de la pregunta, no de la respuesta

En total se complementaron 42 instrumentos de evaluación contestados, de los cuales 7 fueron descartados por su haber sido llenados de manera incorrecta.

Se tomaron los 35 cuestionarios contestados correctamente para aplicarles un análisis de fiabilidad a partir del índice **KR-20** el cual es una medida de

consistencia interna que solo requiere una aplicación del instrumento de evaluación. [5]

En las tablas 10, 11, 12, 13, 14 se presentan los resultados por sección del análisis de fiabilidad basado en el índice RK-20.

Administración de Proyectos de Software

Estadísticos de fiabilidad

KR- 20	KR- 20 basada en los elementos tipificados	N de elementos
.643	.658	6

Estadísticos de la escala

Media	Varianza	Desviación típica	N de elementos
6.91	1.669	1.292	6

Item	KR- 20 si se elimina el elemento
Para cada proyecto de software se tiene asignado un líder de proyecto	.564
Antes de comenzar el desarrollo del producto se realizan estimaciones sobre el costo y tiempo que requerirá su producción	.625
Existe un programa de entrenamiento formal para los nuevos elementos del proyecto, que este diseñado para familiarizarlos con los procedimientos de la organización	.612
Se realiza una planeación documentada formalmente de los tiempos en que se va a llevar el desarrollo de producto con la finalidad de monitorear l avance del proyecto	.590
Se realizan revisiones periódicas formales para verificar el estado de cada proyecto de software y mantener informados a los miembros del proyecto	.578
Se tienen definidas métricas que midan el desempeño del desarrollo del producto	.626

Tabla 10. Análisis de Confiabilidad – Administración de Proyectos de Software

Análisis de Software

Estadísticos de fiabilidad

KR- 20	KR- 20 basada en los elementos tipificados	N de elementos
.754	.755	6

Estadísticos de la escala

Media	Varianza	Desviación típica	N de elementos
8.00	3.529	1.879	6

Item	KR- 20 si se elimina el elemento
Se evalúa formalmente los beneficios y la viabilidad del desarrollo del proyecto de software antes de realizar un contrato con los compromisos	.676
Dentro de la organización se cuenta con un documento estandarizado para la obtención de requerimientos	.708
Se cuenta con un proceso formal para el cambio de un requerimiento previamente definido	.707
Se define un ciclo de desarrollo en base a las características del producto que se va a producir	.715
Se modelan los requerimientos en base a notaciones formales	.768
Antes de iniciar el proyecto, se revisan las nuevas tecnologías existentes en el mercado para determinar la más adecuada a las necesidades del cliente	.727

Tabla 11. Análisis de Confiabilidad – Análisis de Software

Diseño de Software

Estadísticos de fiabilidad

KR- 20	KR- 20 basada en los elementos tipificados	N de elementos
.791	.793	7

Estadísticos de la escala

Media	Varianza	Desviación típica	N de elementos
9.00	4.471	2.114	7

Item	KR- 20 si se elimina el elemento
Son utilizados los diseños de interfaces y reportes tomando en cuenta a los usuarios finales	.776
Se lleva a cabo el proceso de normalización de base de datos para generar un diagrama Entidad-Relación óptimo para el manejo de los datos	.750
Se generan diccionarios de datos para el control y almacenaje detallado de los datos del sistema	.765
Son utilizadas notaciones formales para realizar el diseño de los sistemas	.747
Se toman en cuenta los patrones de arquitectura y diseño para el modelado del sistema	.779
Se utilizan herramientas CASE para realizar el diseño del producto	.763
Al realizar el diseño del sistema se busca que haya bajo acoplamiento y alta cohesión	.770

Tabla 12. Análisis de Confiabilidad – Diseño de Software

Codificación de Software

Estadísticos de fiabilidad

KR- 20	KR- 20 basada en los elementos tipificados	N de elementos
.760	.761	6

Estadísticos de la escala

		Desviación típica	N de elementos
Media	Varianza		
8.23	3.829	1.957	6

Item	KR- 20 si se elimina el elemento
Para cada proyecto se tiene establecido un sistema de control de versiones y cambios para controlar el desarrollo del software	.704
Son utilizadas técnicas de prototipados (evolutivos o desechables) para realizar un desarrollo del producto que cumpla con las expectativas del cliente	.731
Se tiene establecido un ambiente de desarrollo para realizar la integración de las nuevas funcionalidades, antes de integrarlas a la versión estable o final	.732
Se cuenta con un formato específico que lleve el control del código fuente desarrollado	.720
Se tiene en la organización con un formato específico para realizar la documentación técnica del código fuente desarrollado	.703
Se tienen especificados estándares para la escritura de código fuente	.759

Tabla 13. Análisis de Confiabilidad – Codificación de Software

Pruebas de Software

Estadísticos de fiabilidad

KR- 20	KR- 20 basada en los elementos tipificados	N de elementos
.705	.718	6

Estadísticos de la escala

Media	Varianza	Desviación típica	N de elementos
8.23	3.123	1.767	6

Item	KR- 20 si se elimina el elemento
Se realiza un plan de pruebas antes de comenzar el desarrollo del producto	.647
Las pruebas de software son realizadas por un grupo totalmente independiente al equipo de desarrollo	.722
Se tiene establecido que se realicen revisiones de pares a los requerimientos desarrollados	.697
Se realizan pruebas del sistema con los usuarios finales en el ambiente deseado por el cliente antes de su implementación final	.627
Se cuenta con un formato establecido para la definición y llenado de los casos de prueba	.644
Son utilizadas herramientas automatizadas de pruebas	.652

Tabla 14. Análisis de Confiabilidad – Pruebas de Software

La confiabilidad de consistencia interna permite determinar el grado en que los ítems de una prueba están correlacionados entre sí. Si los diferentes reactivos de un instrumento tienen una correlación positiva y, como mínimo, moderada, dicho instrumento será homogéneo. [22]

Existen diferentes procedimientos para estimar la confiabilidad de consistencia interna. Algunos de los más conocidos son los siguientes: Kuder-Richardson; Alpha de Cronbach; Dos mitades, corregido por la fórmula de Spearman-Brown; y método de Hoyt.

Kuder y Richardson (1937) desarrollaron varios modelos para estimar la confiabilidad de consistencia interna de una prueba, siendo uno de los más conocidos la denominada fórmula 20, el cual se representa de la siguiente manera:

$$r_{ii} = \frac{n}{n-1} * \frac{V_t - \sum pq}{V_t}$$

En donde:

r_{ii} = coeficiente de confiabilidad.

N = número de ítems que contiene el instrumento.

V_t = varianza total de la prueba.

$\sum pq$ = sumatoria de la varianza individual de los ítems.

El modelo de Kuder-Richardson es aplicable en las pruebas de ítems dicotómicos en los cuales existen respuestas correctas e incorrectas. [23]

Después de haber realizado la prueba piloto, efectuar el cálculo de fiabilidad del índice KR-20 en cada una de las secciones del instrumento de evaluación y analizar los resultados obtenidos se observó que en la sección de administración de proyectos de software se obtuvo un valor de menor a .7 (.643). A partir de esto se buscó aumentar este valor con la eliminación de elementos de esta sección sin embargo no se logró alcanzar un valor mayor al obtenido inicialmente.

El instrumento de evaluación se aplicó en la población definida para esta investigación (MiPyMEs desarrolladoras de software) con cada una de las secciones definidas inicialmente, observando un comportamiento homogéneo en cada una de las secciones (valor de fiabilidad RK-20 mayor a .7).

5. REPORTE Y DISCUSIÓN DE RESULTADOS.

5.1 Técnicas y Análisis de Resultados.

Dentro de este apartado se describirá y analizará cómo fueron llevadas a cabo las actividades de identificación de empresas que cumplieran con el perfil de estudio de la investigación, la recolección de los datos y el tratado de estos.

Una vez realizado el análisis de confiabilidad a partir del índice RK-20 al instrumento de evaluación, este fue aplicado a las MiPyMEs desarrolladoras de software en el estado de Aguascalientes en base al listado proporcionado por el Consejo de Ciencia y Tecnología del Estado de Aguascalientes (CONCYTEA).

Se contemplaron 21 empresas que cumplieron con el perfil buscado en la investigación (MiPyMES desarrolladoras de software). Se intentó encuestar a cada

TESIS TESIS TESIS TESIS TESIS

una de las empresas y así poder cubrir en su totalidad la base de datos proporcionada por el CONCYTEA, finalmente 15 empresas estuvieron interesadas en colaborar con la investigación por lo que se cubrió el 71% de la población.

Cada una de las empresas fue contactada vía telefónica para solicitar una cita personal en la cual se describían los objetivos de la investigación, se les informaba sobre el respaldo con el que contaba el trabajo de investigación por parte del Centro de Ciencias Básicas de la Universidad Autónoma de Aguascalientes y el Consejo Nacional de Ciencia y Tecnología (CONACYT).

Las empresas que participaron en la investigación respondieron un cuestionario sobre mejores prácticas de ingeniería de software utilizadas en el desarrollo de sus productos o servicios. El cuestionario podía ser contestado en papel o en línea según la preferencia de la empresa.

Una vez recolectados los datos estos fueron volcados para su análisis en el paquete estadístico PASW Statistics versión 18.

Se utilizó estadística descriptiva para conocer el panorama en el que se encuentran las mejores prácticas en la industria de software del estado de Aguascalientes.

Las hipótesis fueron evaluadas a partir de pruebas no paramétricas Kruskal Wallis “K muestras independientes”.

Finalmente se realizaron cruces de variables para observar el comportamiento de las mejores prácticas en base a las características de las empresas.

5.2 Descripción de Contribuciones.

La descripción de las contribuciones hechas en la presente investigación serán presentadas en base a los objetivos establecidos.

Objetivo General

Objetivo 1

Identificar un panorama sobre las mejores prácticas implementadas por las MiPyMEs desarrolladoras de software en el estado de Aguascalientes las cuales podrían llegar a ser útiles en la creación y crecimiento de MiPyMEs enfocadas al desarrollo de software.

Para llegar al cumplimiento de este objetivo a continuación se presenta las características demográficas de las empresas y el comportamiento individual cada una de las mejores prácticas de ingeniería de software a partir de estadística descriptiva categorizado tal como se hizo en el instrumento de evaluación.

Sección I Datos Demográficos

1.- ¿Número de empleados con los que cuenta la organización?

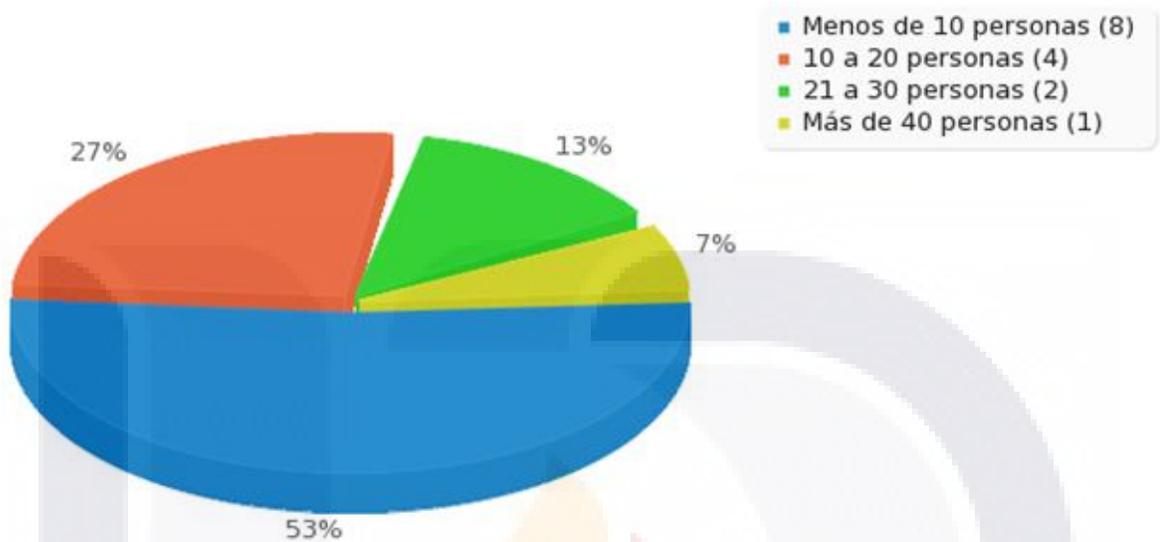


Figura 5. Empleado con los que cuenta la organización

2.- ¿Número de empleados involucrados en la Ingeniería de Software?

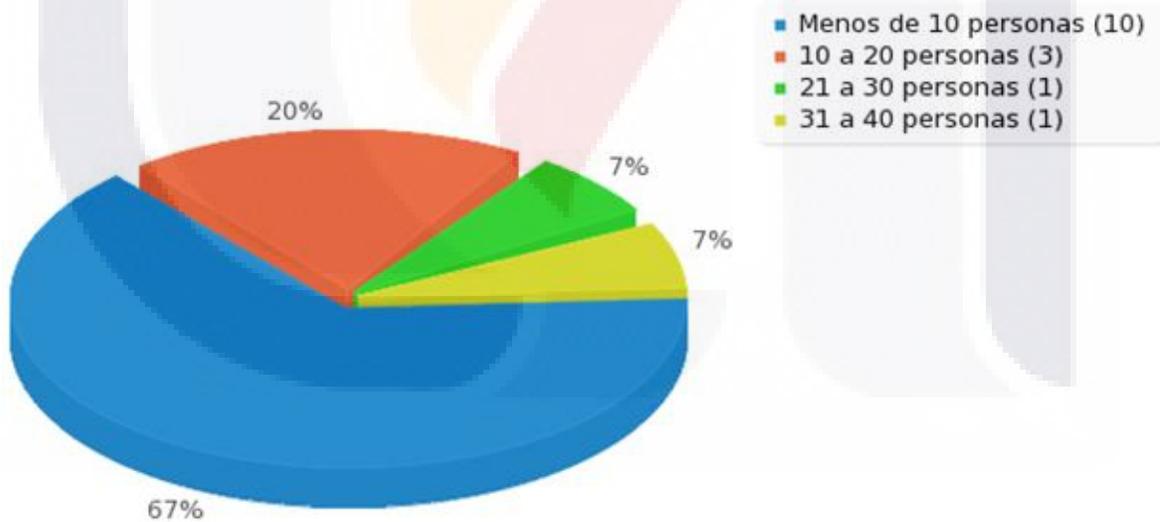


Figura 6. Empleados Involucrados en Ingeniería de Software

3.- ¿Tiempo de existencia de la organización?

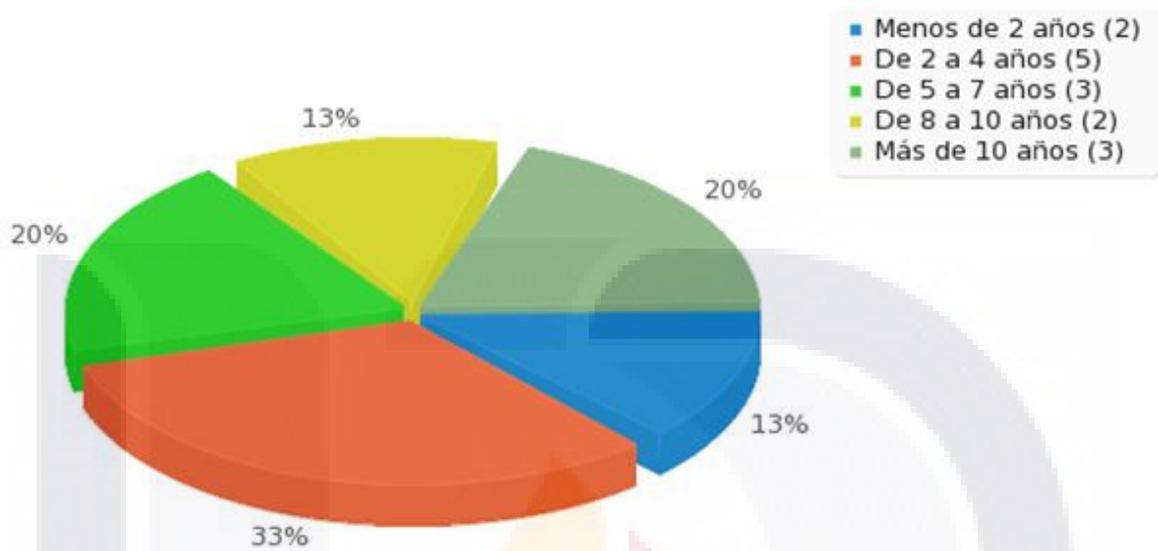


Figura 7. Tiempo de existencia de la organización

4.- ¿La organización actualmente tiene implementado algún modelo de trabajo de mejores prácticas?

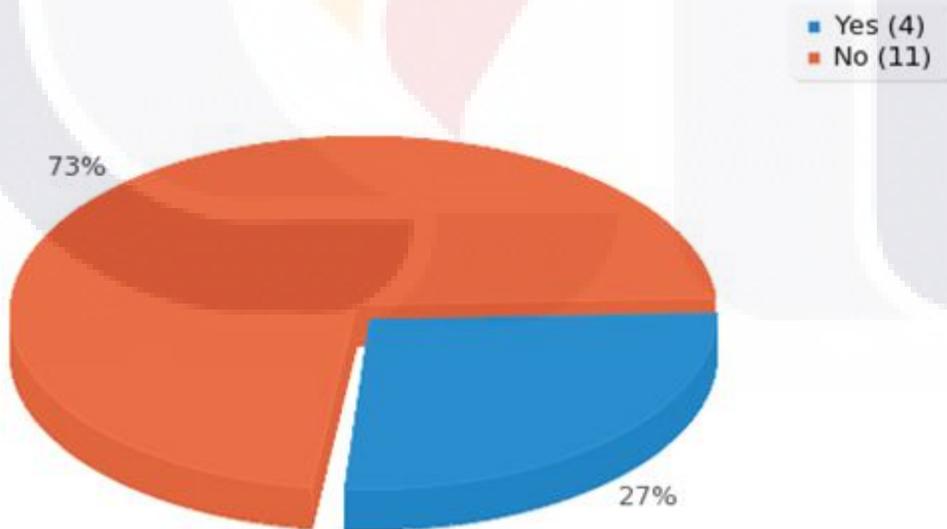


Figura 8. Implementación actual de modelo de mejores prácticas

5.- ¿La organización a corto plazo (Menor a 2 años) tiene formalmente planeado implementar un modelo de trabajo de mejores prácticas?

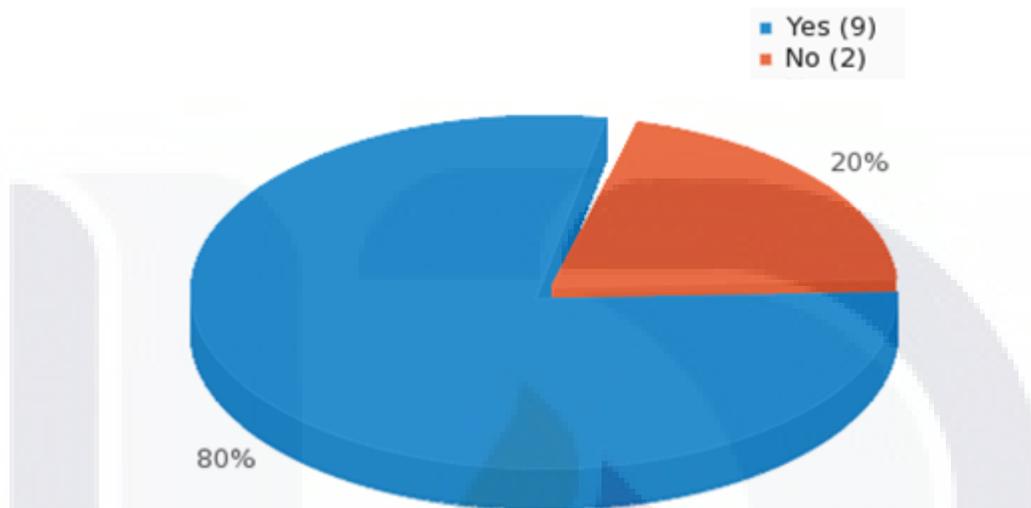


Figura 9. Planeación de implementación de modelo de mejores prácticas

De la Figura 5 a la Figura 9 se presentan en gráficas de pastel los resultados del comportamiento de la primera sección del instrumento de evaluación, la cual abarcó las características demográficas de las organizaciones. La recolección de estos datos tuvo como objetivo y principal funcionalidad la comparación del nivel de implementación de mejores prácticas en base al tamaño de la organización, tiempo de existencia y personal involucrado en los procesos de ingeniería de software.

Sección II Administración de Proyectos de Software

1.- ¿Para cada proyecto de software se tiene asignado un Líder de Proyecto?

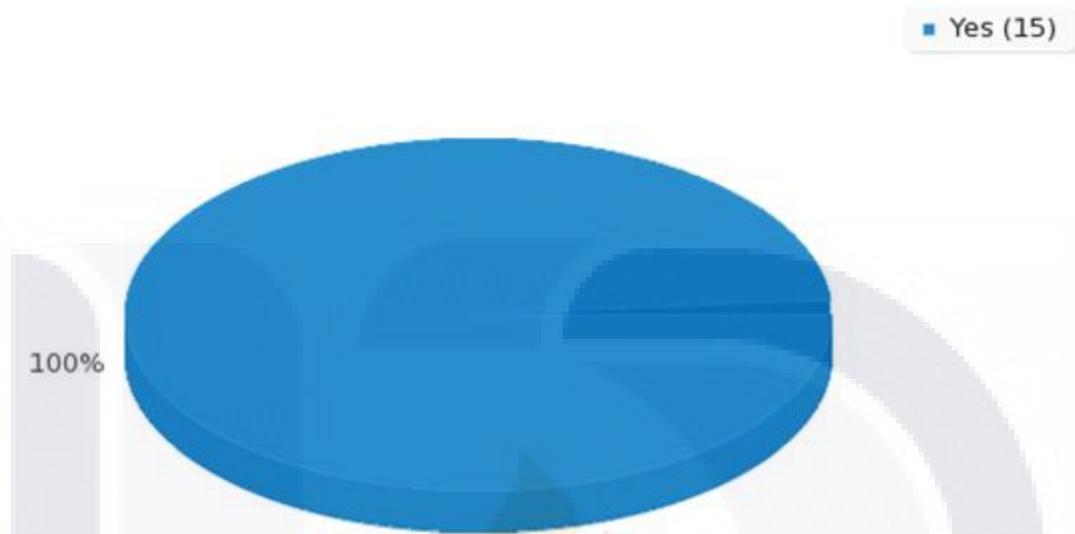


Figura 10. Asignación de Líder de Proyecto

2.- ¿Antes de comenzar el desarrollo del producto se realizan estimaciones sobre el costo y tiempo que requerirá su producción?

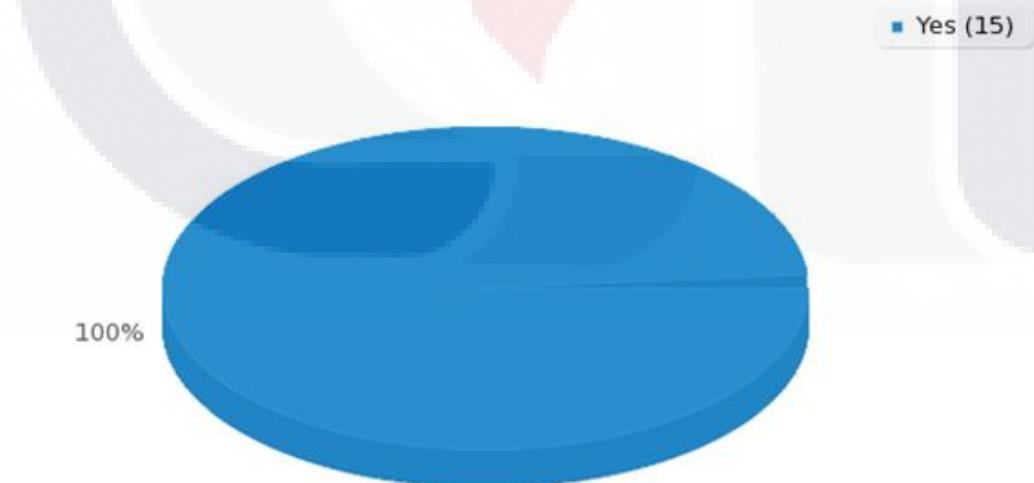


Figura 11. Costo y tiempo del desarrollo

3.- ¿Existe un programa de entrenamiento formal para los nuevos elementos del

proyecto, que este diseñado para familiarizarlos con los procedimientos de la organización?

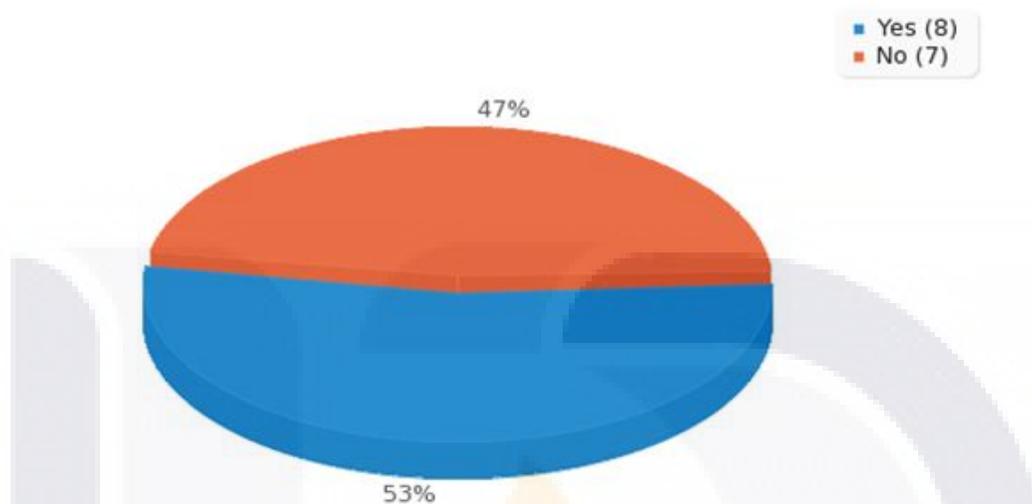


Figura 12. Capacitación a nuevos elementos del proyecto

4.- ¿Se realiza una planeación documentada formalmente de los tiempos en que se va a llevar el desarrollo de producto con la finalidad de monitorear el avance del proyecto?

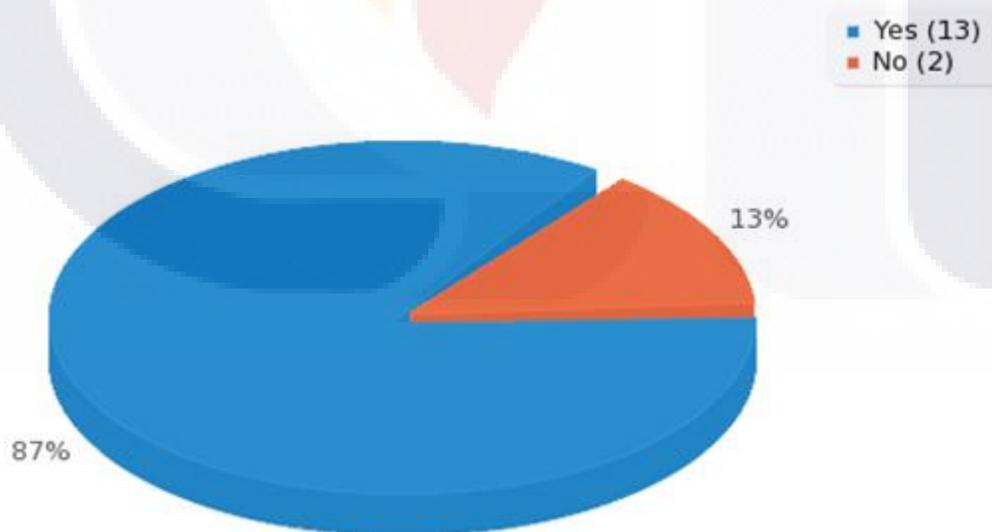


Figura 13. Monitoreo de avance de proyecto

5.- ¿Se realizan revisiones periódicas formales para la verificar el estado de cada proyecto de software y mantener informados a los miembros del proyecto?

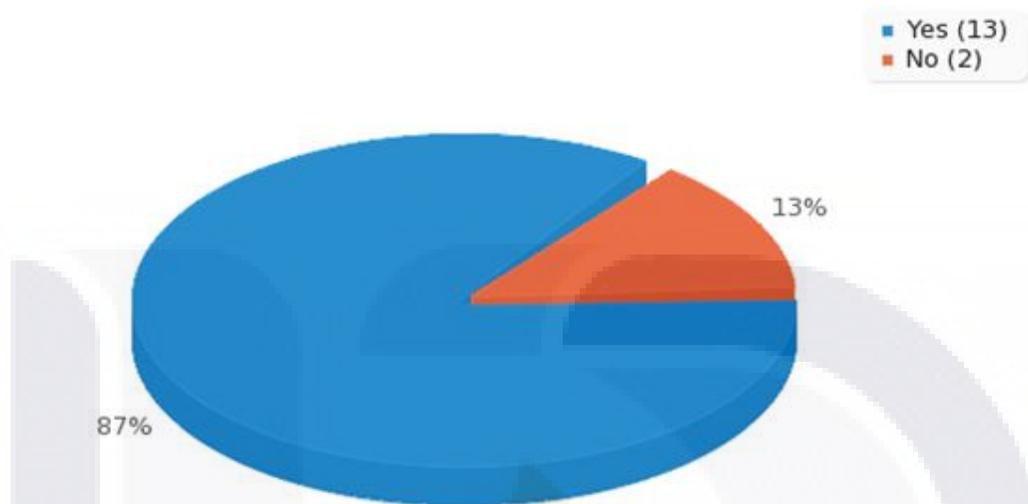


Figura 14. Revisiones periódicas

6.- ¿Se tienen definidas métricas que midan el desempeño del desarrollo del producto?

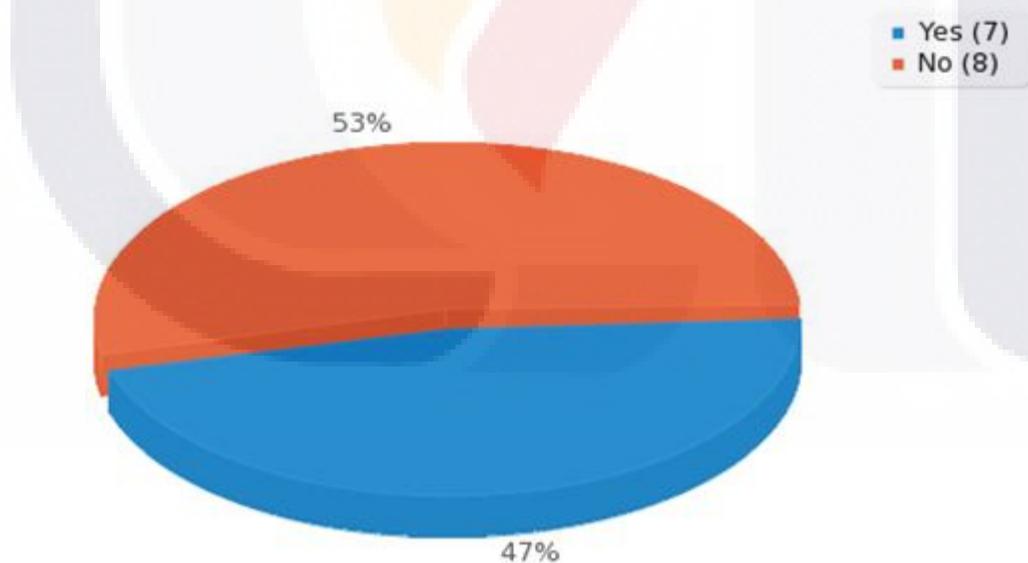


Figura 15. Métricas sobre el desempeño del producto

De la Figura 10 a la Figura 15 se presenta en gráficas de pastel los resultados del comportamiento de la segunda sección del instrumento de evaluación, la cual abarcó la administración de proyectos de software.

La presente sección fue la que mostro el nivel mayor de implementación de mejores prácticas, obteniendo en dos de sus prácticas un 100% de utilización en las organizaciones que participaron en la investigación.

Sección III Análisis de Software

1.- ¿Se evalúa formalmente los beneficios y la viabilidad del desarrollo del proyecto de software antes de realizar un contrato con los compromisos?

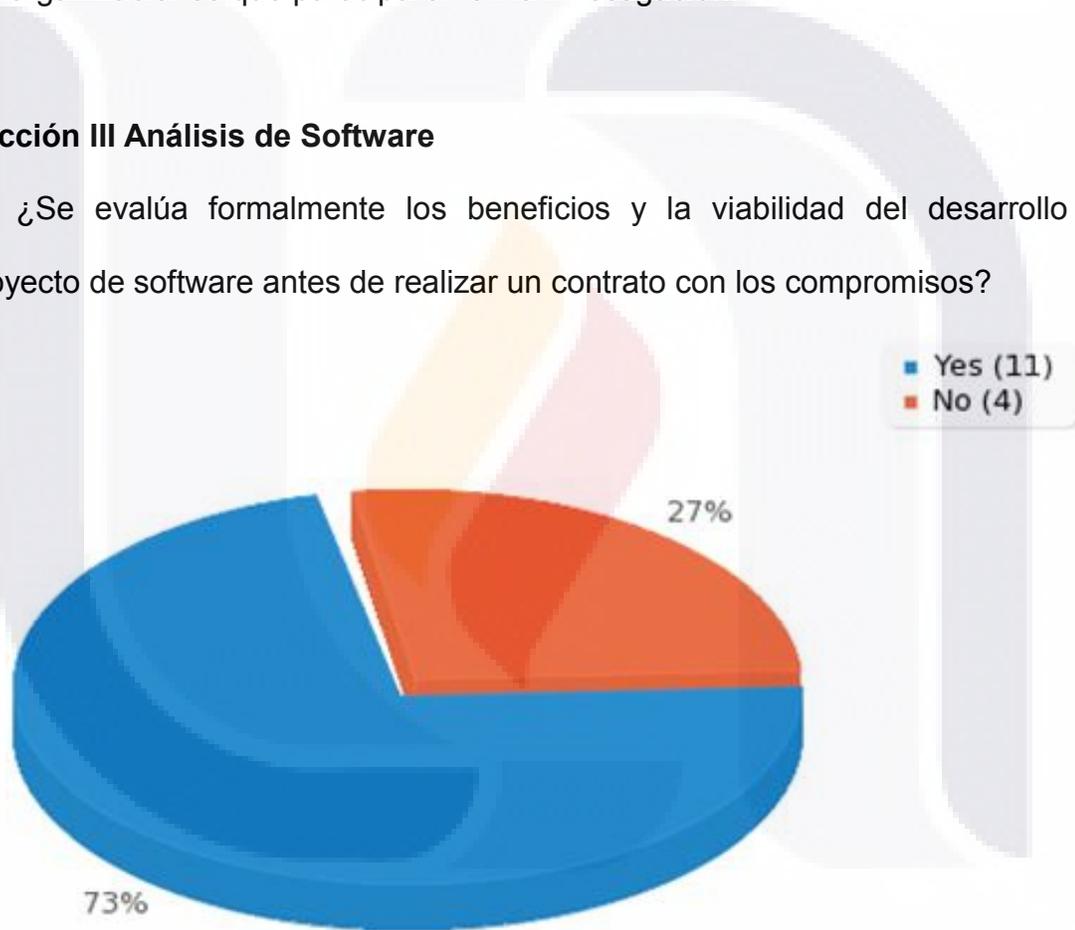


Figura 16. Beneficios y viabilidad del proyecto

2.- ¿Dentro de la Organización se cuenta con un documento estandarizado para la obtención de requerimientos?

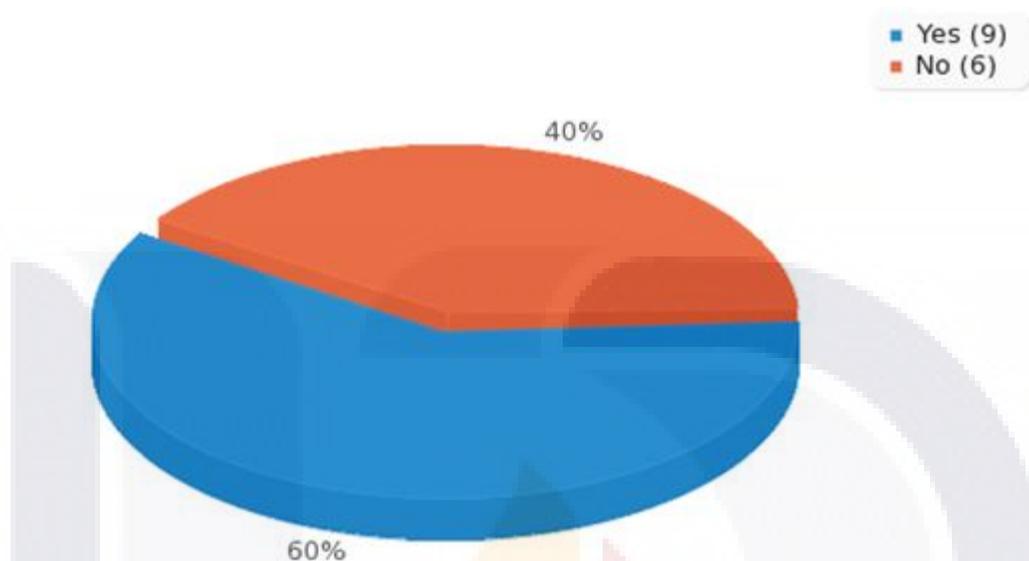


Figura 17. Documento estandarizado para obtención de requerimientos

3.- ¿Se cuenta con un proceso formal para el cambio de un requerimiento previamente definido?

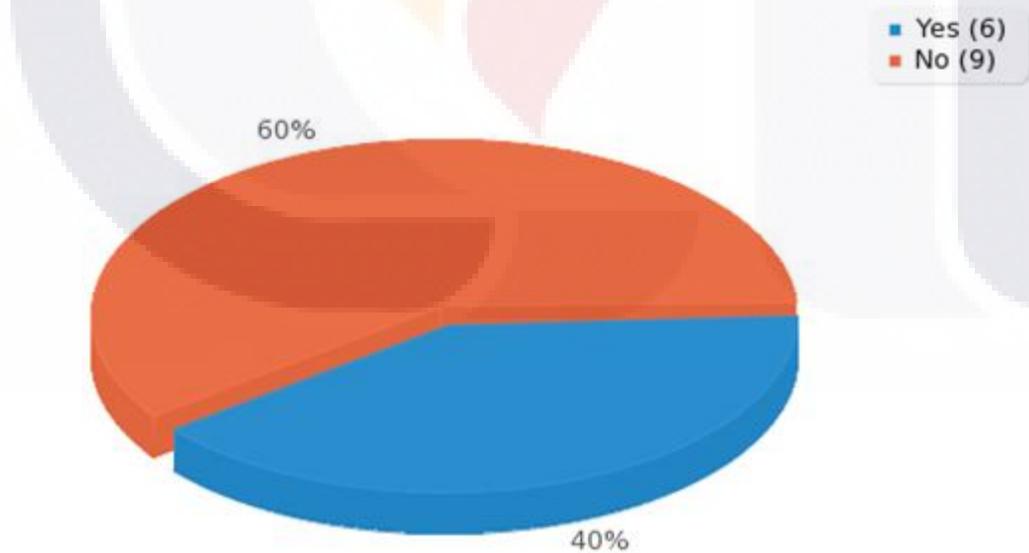


Figura 18. Proceso definido en cambio de requerimientos

4.- ¿Se define un ciclo de desarrollo en base a las características del producto que se va a producir?

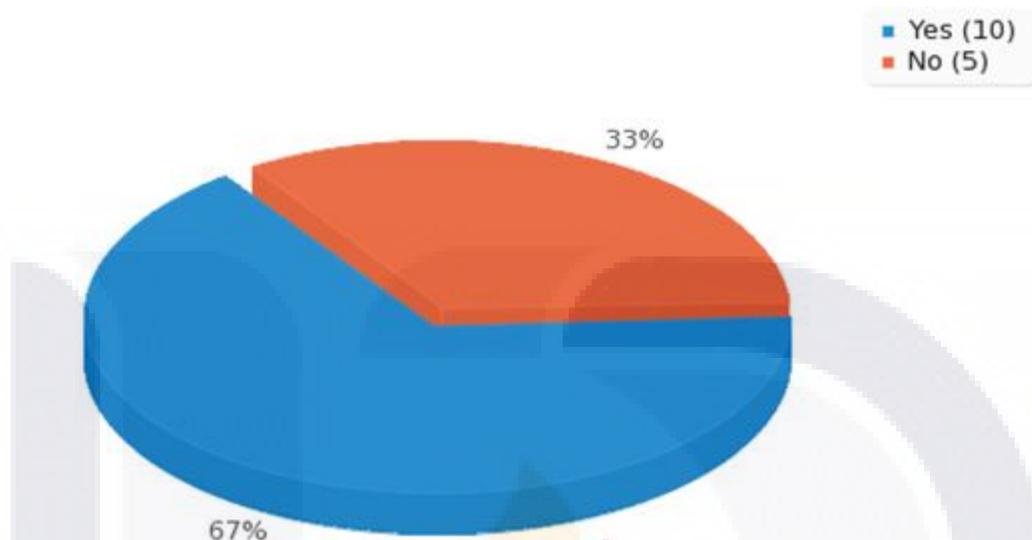


Figura 19. Definición de ciclos de desarrollo

5.- ¿Se modelan los requerimientos en base a notaciones formales?

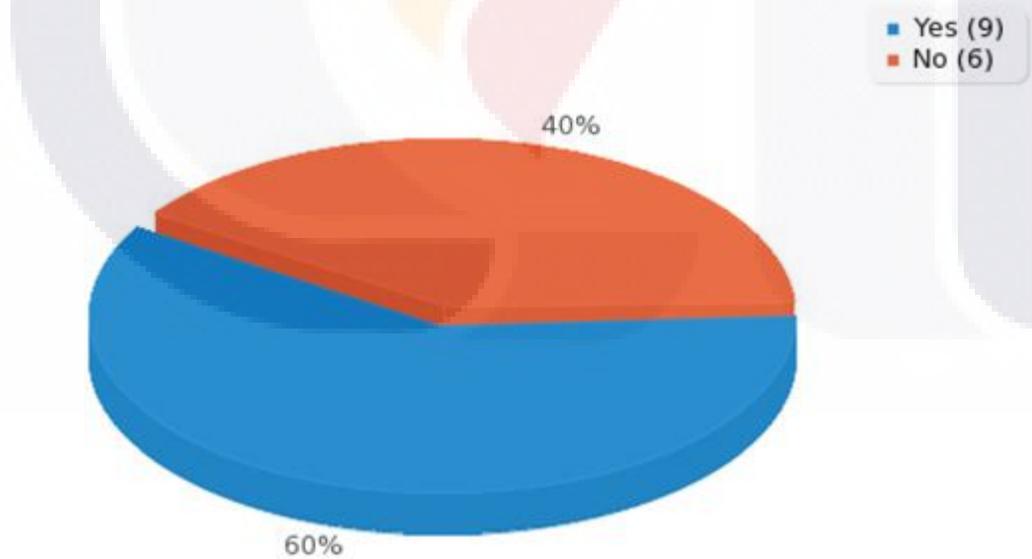


Figura 20. Modelación de requerimientos

6.- ¿Antes de iniciar el proyecto, se revisan las nuevas tecnologías existentes en el mercado para determinar la más adecuada a las necesidades del cliente?

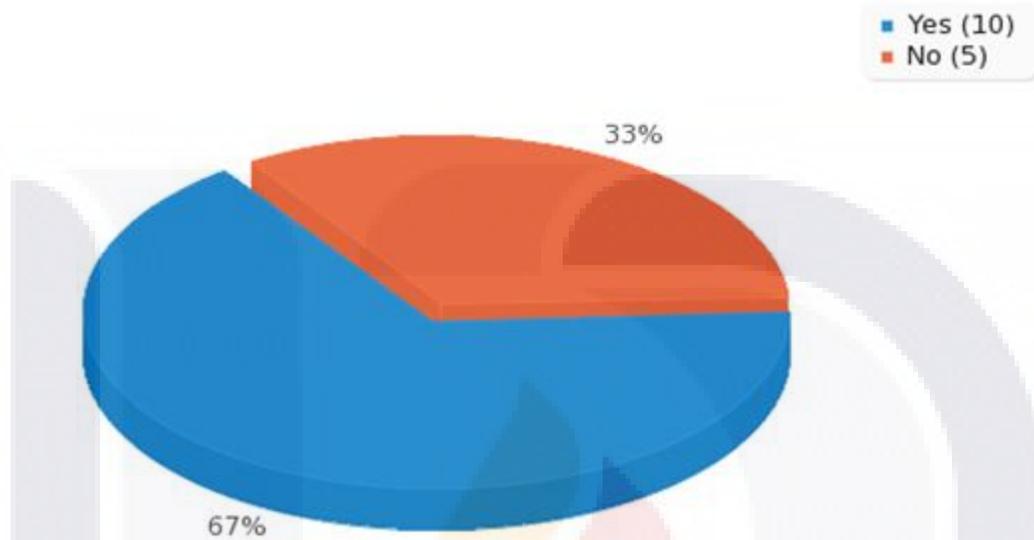


Figura 21. Tecnologías existentes del mercado

De la Figura 16 a la Figura 21 se presenta en gráficas de pastel los resultados del comportamiento de la tercera sección del instrumento de evaluación, la cual abarcó la etapa de análisis de software.

La presente sección mostro un nivel promedio de nivel de implementación (aproximado al 50%) siendo la mejor práctica con mayor utilización la de “análisis y viabilidad del desarrollo del proyecto” y la de menor implementación “la definición de un proceso formal para el cambio de requerimientos previamente establecidos”.

Sección IV Diseño de Software

1.- ¿Son utilizados los diseños de interfaces y reportes tomando en cuenta a los usuarios finales?

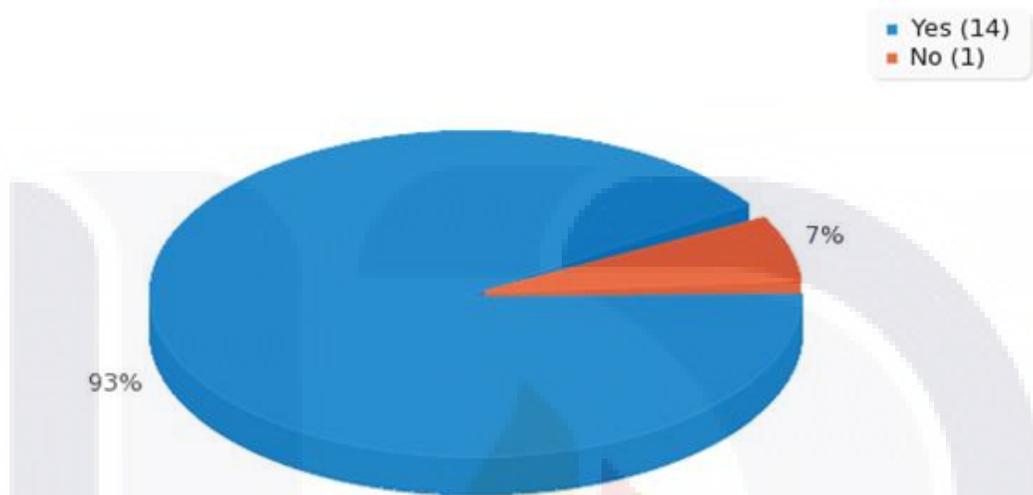


Figura 22. Diseño de interfaces y reportes

2.- ¿Se lleva a cabo el proceso de normalización de base de datos para genera un diagrama Entidad-Relación óptimo para el manejo de los datos?

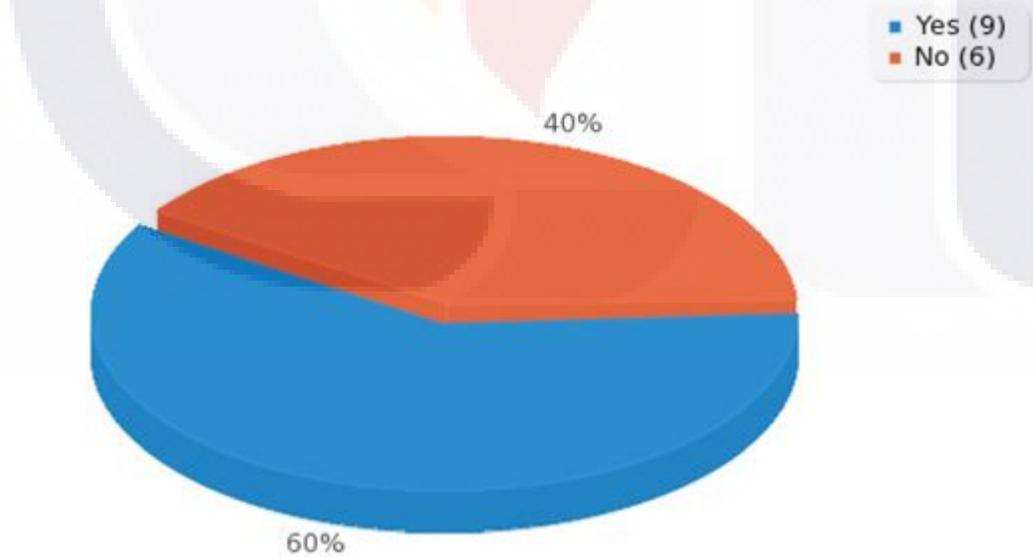


Figura 23. Normalización de bases de datos

3.- ¿Se generan diccionarios de datos para el control y almacenaje detallado de los datos del sistema?

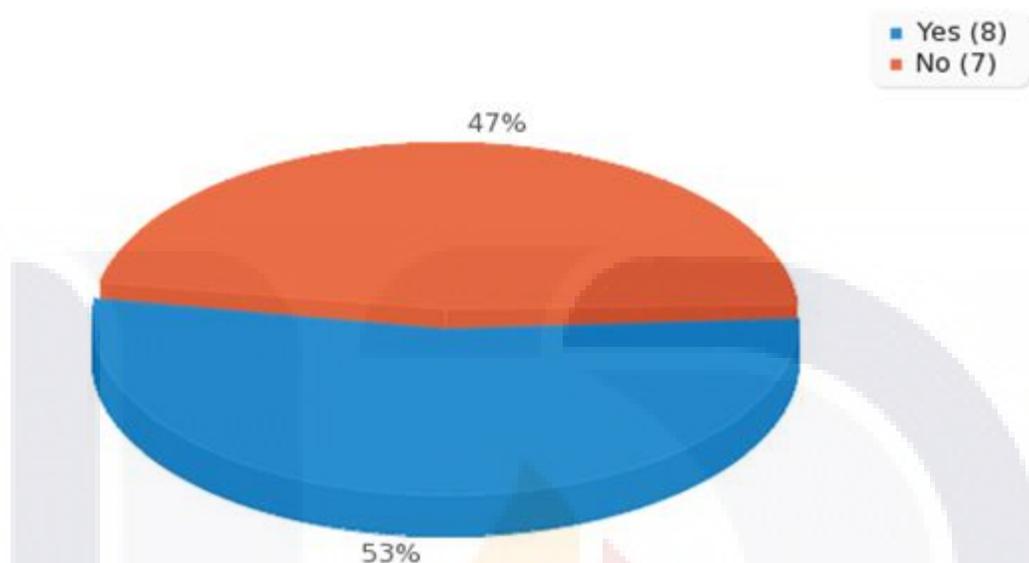


Figura 24. Utilización de diccionarios de datos

4.- ¿Son utilizadas notaciones formales para realizar el diseño de los sistemas?

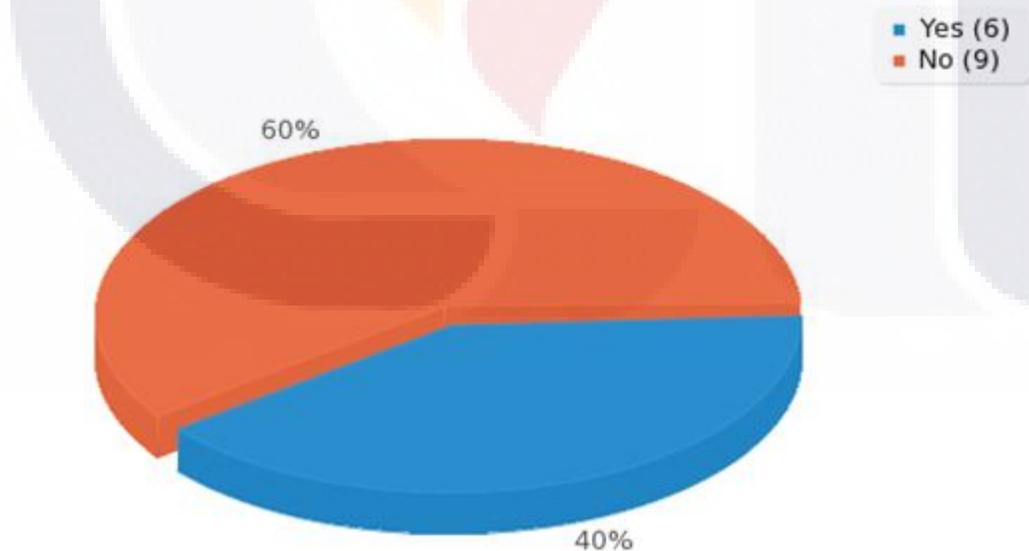


Figura 25. Notaciones formales para el diseño del proyecto

5.- ¿Se toman en cuenta los patrones de arquitectura y diseño para el modelado del sistema?

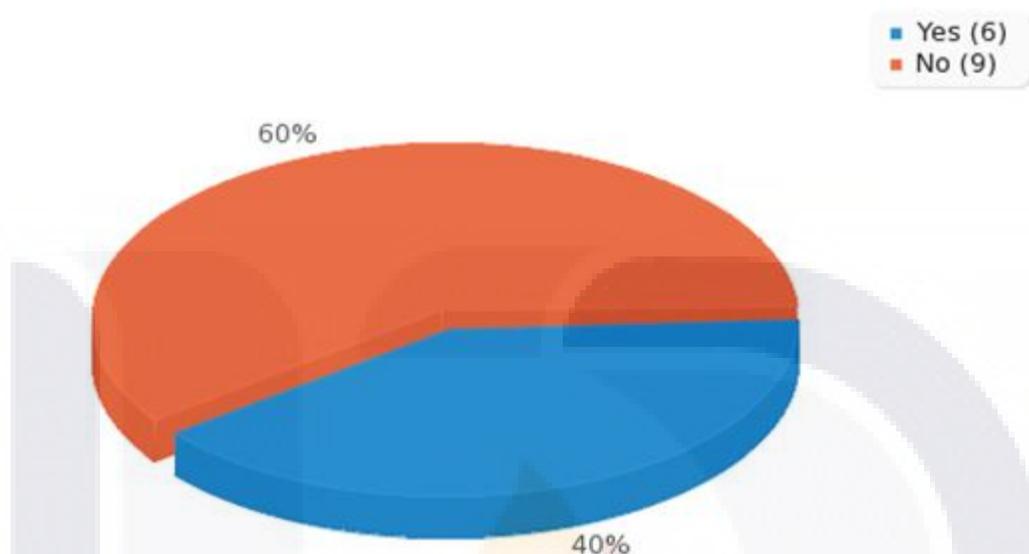


Figura 26. Utilización de patrones de arquitectura y diseño

6.- ¿Se utilizan herramientas CASE para realizar el diseño del producto?

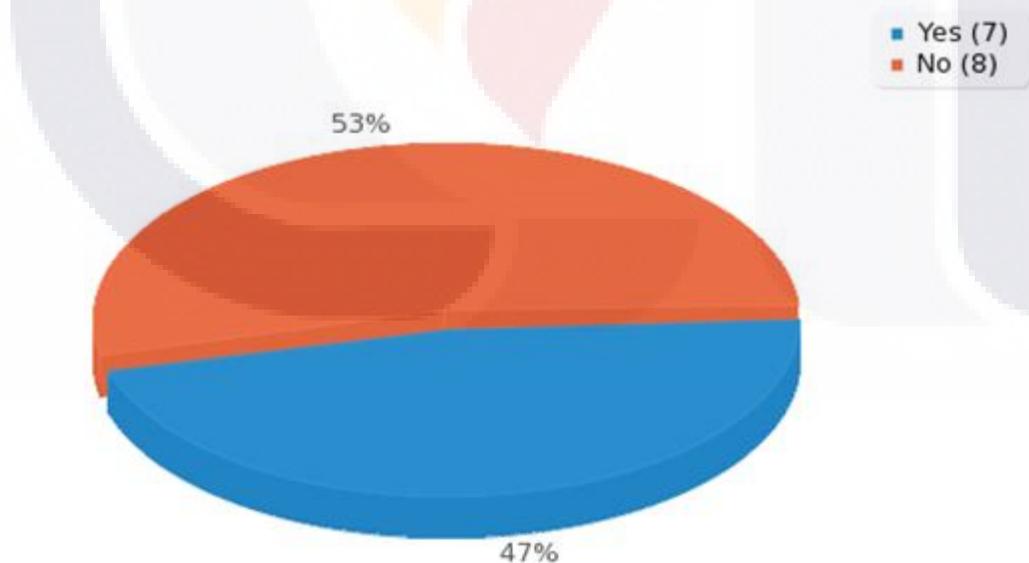


Figura 27. Utilización de herramientas CASE

7.- ¿Al realizar el diseño del sistema se busca que haya bajo acoplamiento y alta cohesión?

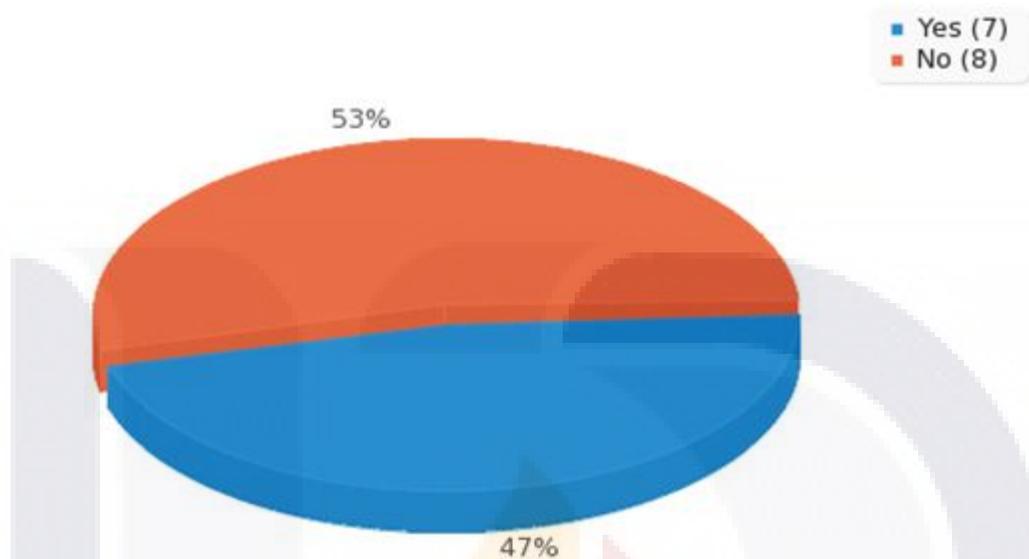


Figura 28. Acoplamiento y cohesión del diseño

De la Figura 22 a la Figura 28 se presenta en gráficas de pastel los resultados del comportamiento de la cuarta sección del instrumento de evaluación, la cual abarcó la etapa de diseño de software.

La presente sección mostro un nivel promedio de nivel de implementación (aproximado al 50%) siendo la mejor práctica con mayor utilización la de “diseño de interfaces y reportes” y las de menor implementación “la utilización de patrones de arquitectura y diseño” y “modelación del diseño del proyecto”.

Sección V Codificación de Software

1.- ¿Para cada proyecto se tiene establecido un sistema de control de versiones y cambios para controlar el desarrollo del software?

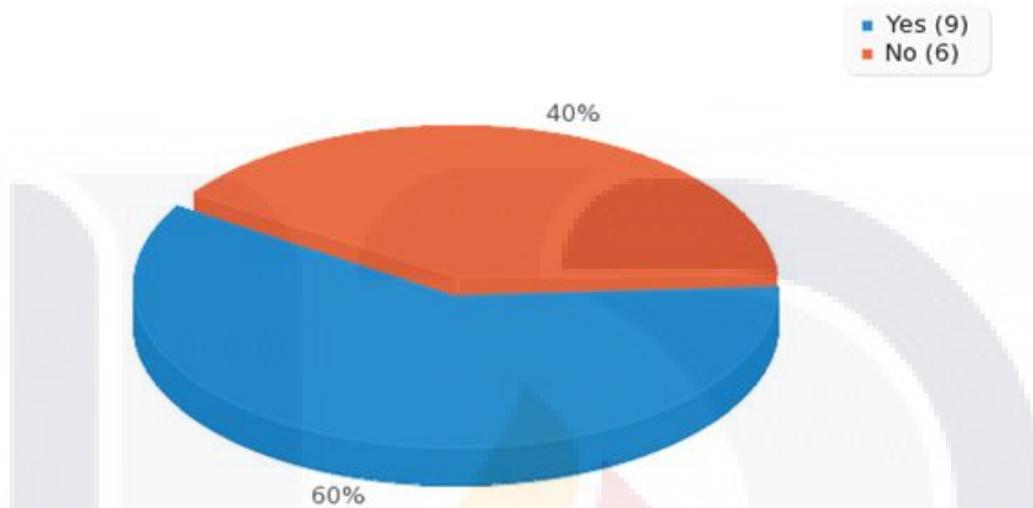


Figura 29. Control de versiones

2.- ¿Son utilizadas técnicas de prototipados (Evolutivos o Desechables) para generar un producto que cumpla con las expectativas del cliente?

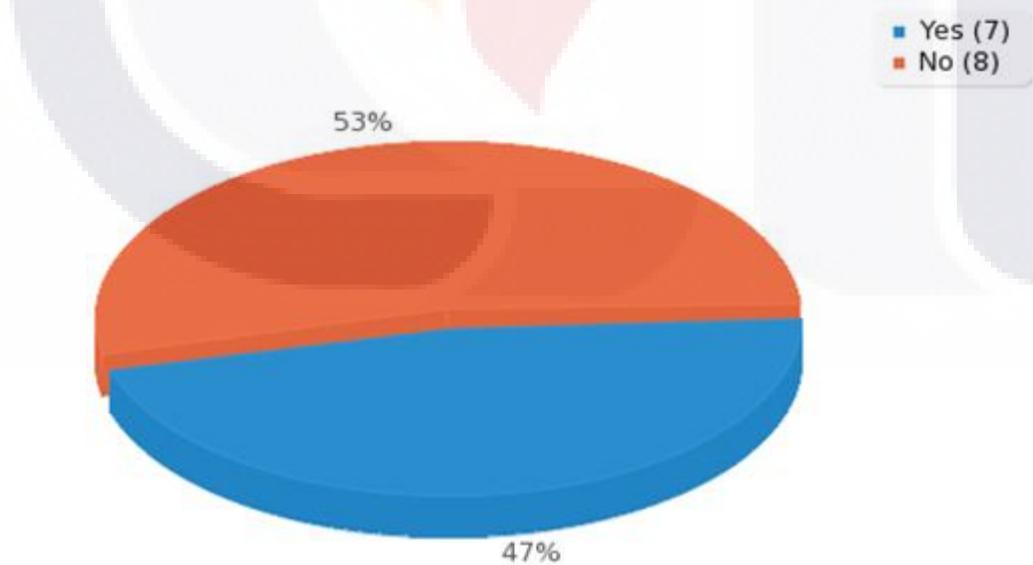


Figura 30. Prototipado del producto

3.- ¿Se tiene establecido un ambiente de desarrollo para realizar la integración de las nuevas funcionalidades, antes de integrarlas a la versión estable o final?

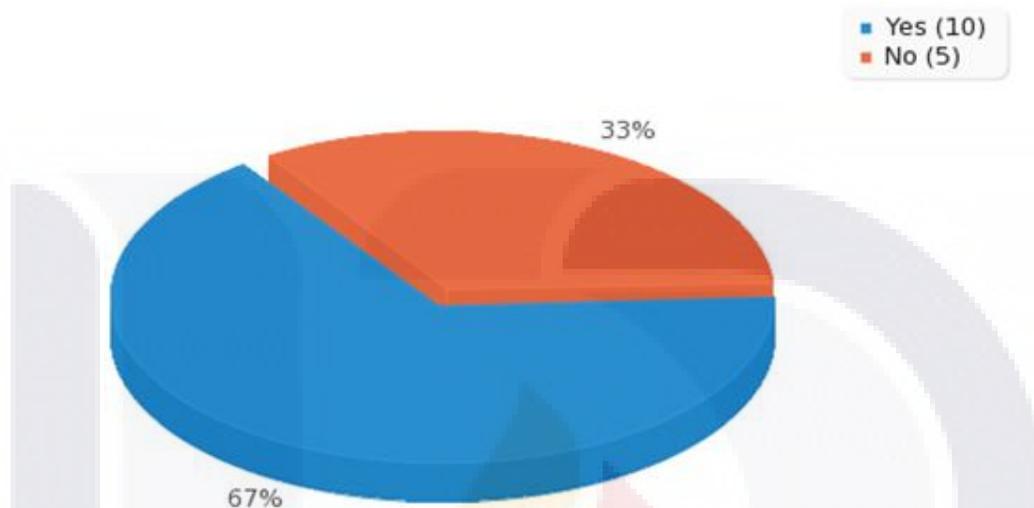


Figura 31. Ambientes de desarrollo

4.- ¿Se cuenta con un formato específico que lleve el control del código fuente desarrollado?

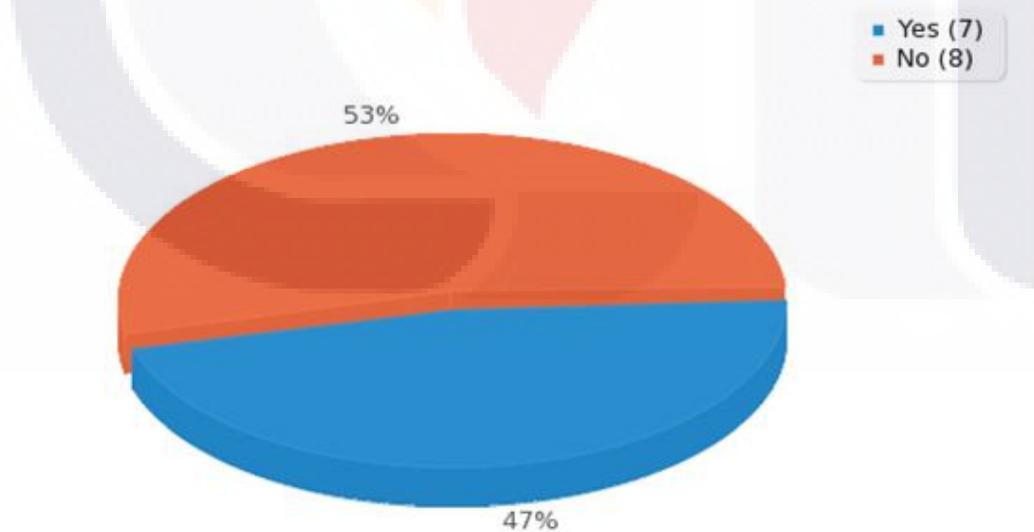


Figura 32. Control del código fuente

5.- ¿Se cuenta en la organización con un formato específico para realizar la documentación técnica del código fuente desarrollado?

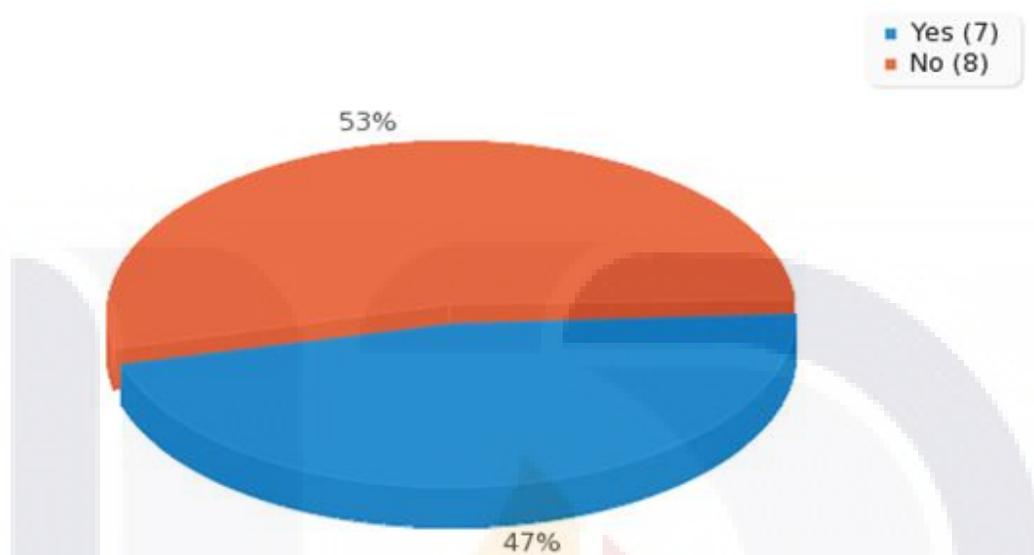


Figura 33. Documentación técnica

6.- ¿Se tienen especificados estándares para la escritura de código fuente?

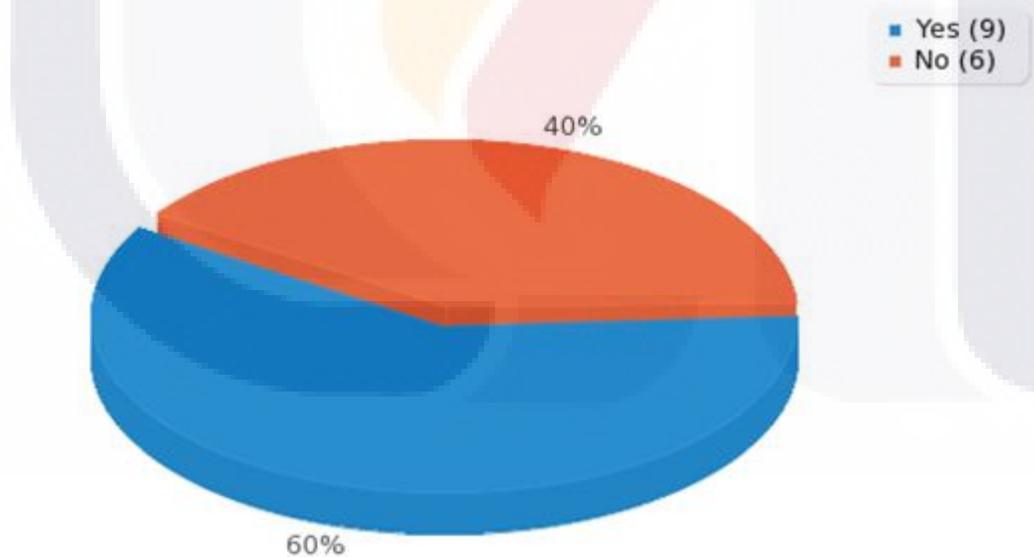


Figura 34. Estándares de codificación

De la Figura 29 a la Figura 34 se presentan en gráficas de pastel los resultados del comportamiento de la cuarta sección del instrumento de evaluación, la cual abarcó la etapa de codificación de software.

La presente sección mostro un nivel promedio de nivel de implementación (aproximado al 50%) siendo la práctica con mayor utilización la de “ambientes de desarrollo de software”.

Sección VI Pruebas de Software

1.- ¿Se realiza un plan de pruebas antes de comenzar el desarrollo del producto?

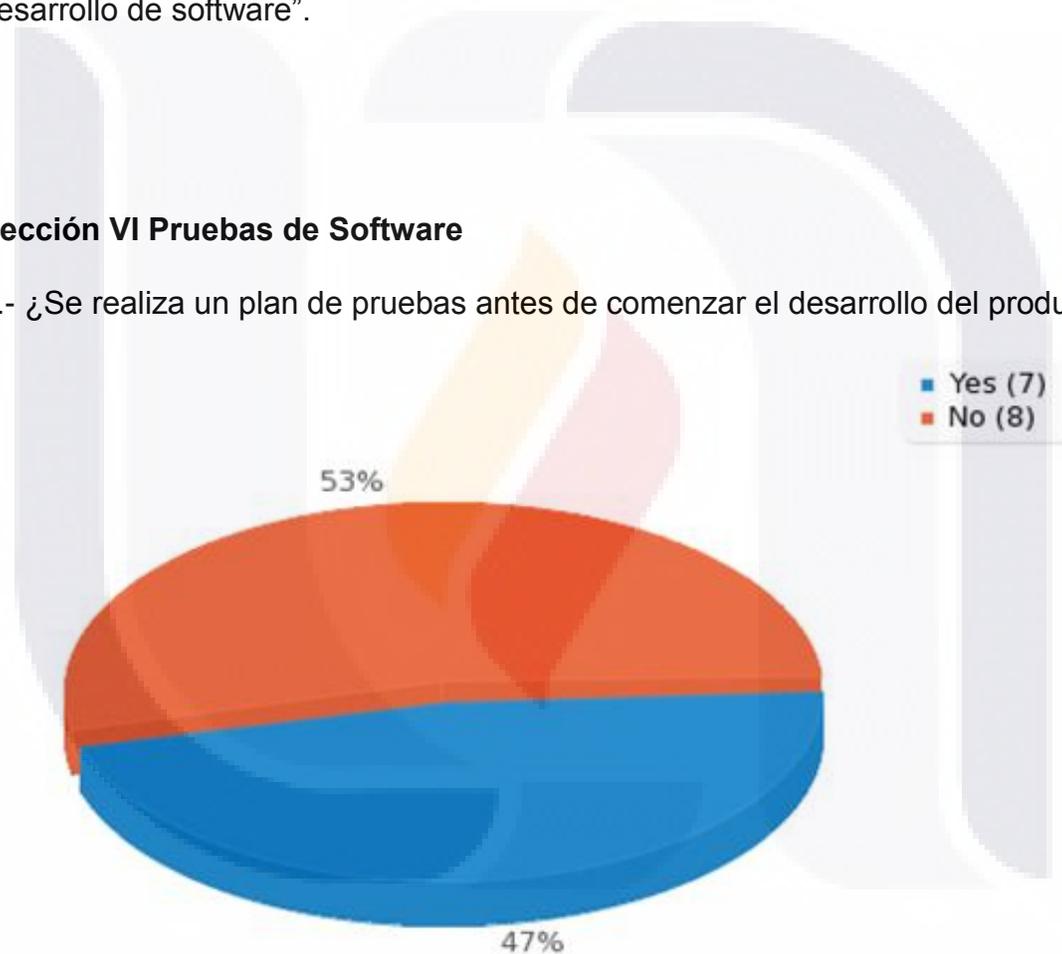


Figura 35. Plan de pruebas

2.- ¿Las pruebas de software son realizadas por un grupo totalmente independiente al equipo de desarrollo?

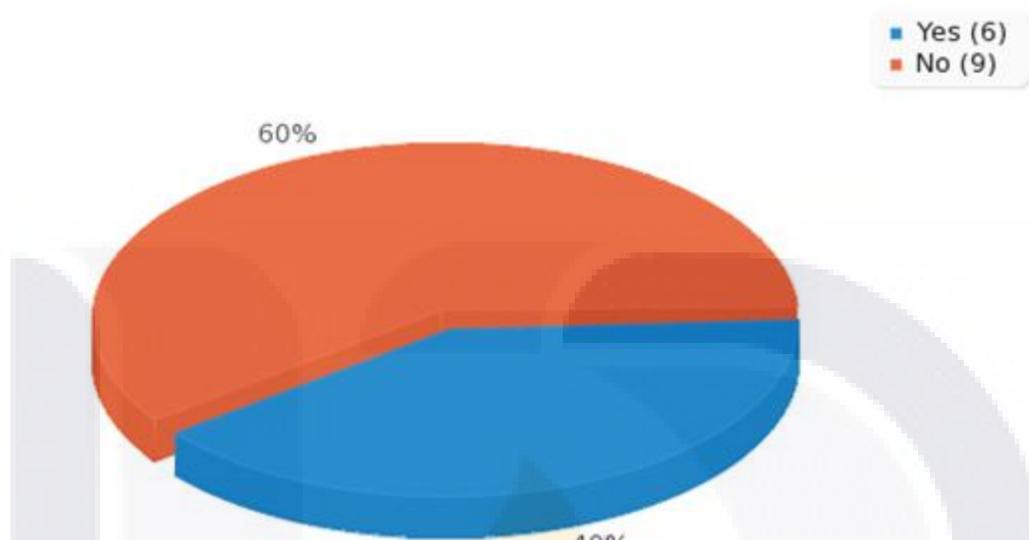


Figura 36. Pruebas independientes

3.- ¿Se tiene establecido que se realicen revisiones de pares a los requerimientos desarrollados?

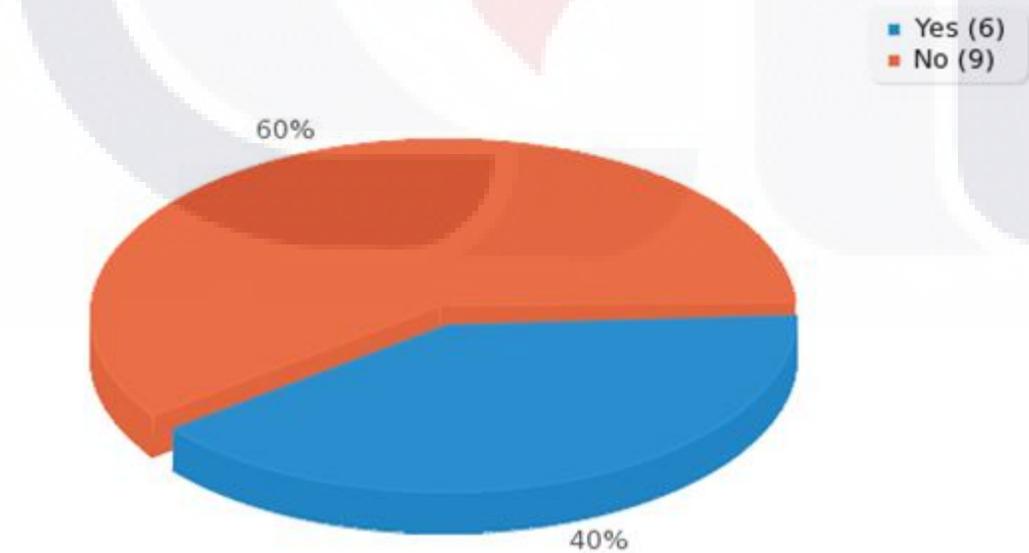


Figura 37. Revisiones de pares

4.- ¿Se realizan pruebas del sistema con los usuarios finales en el ambiente deseado por el cliente antes de su implementación final?

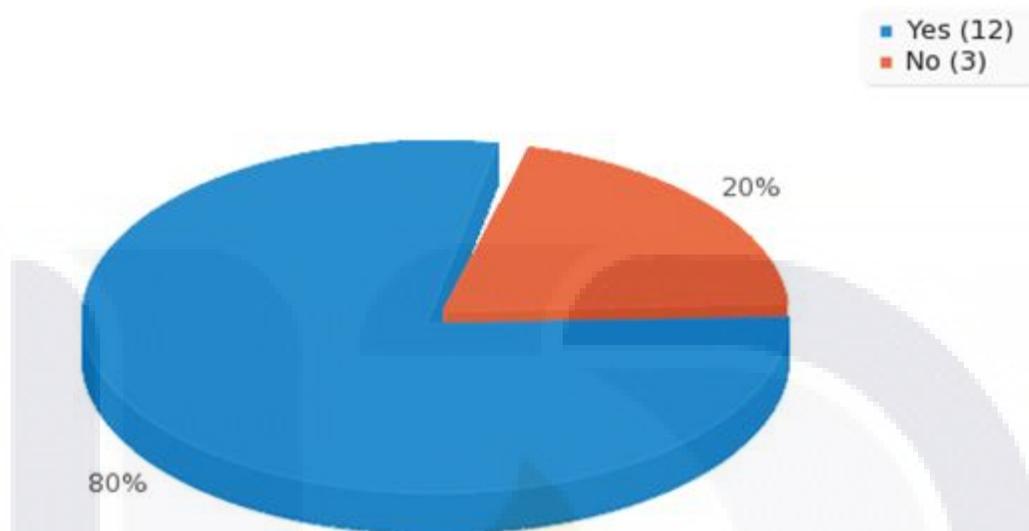


Figura 38. Pruebas con usuarios finales

5.- ¿Se cuenta con un formato establecido para la definición y llenado de los casos de prueba?

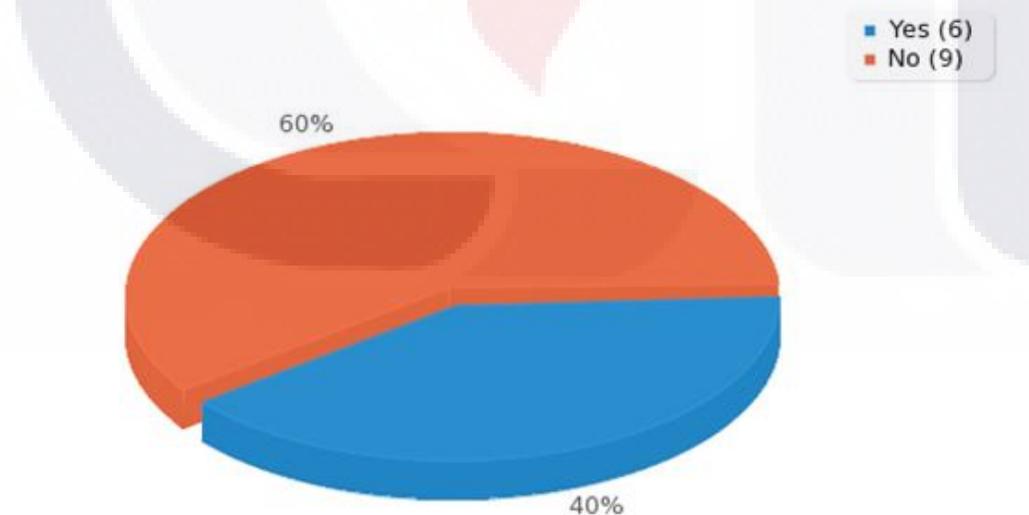


Figura 39. Uso de casos de prueba

6.- ¿Son utilizadas herramientas automatizadas de pruebas?

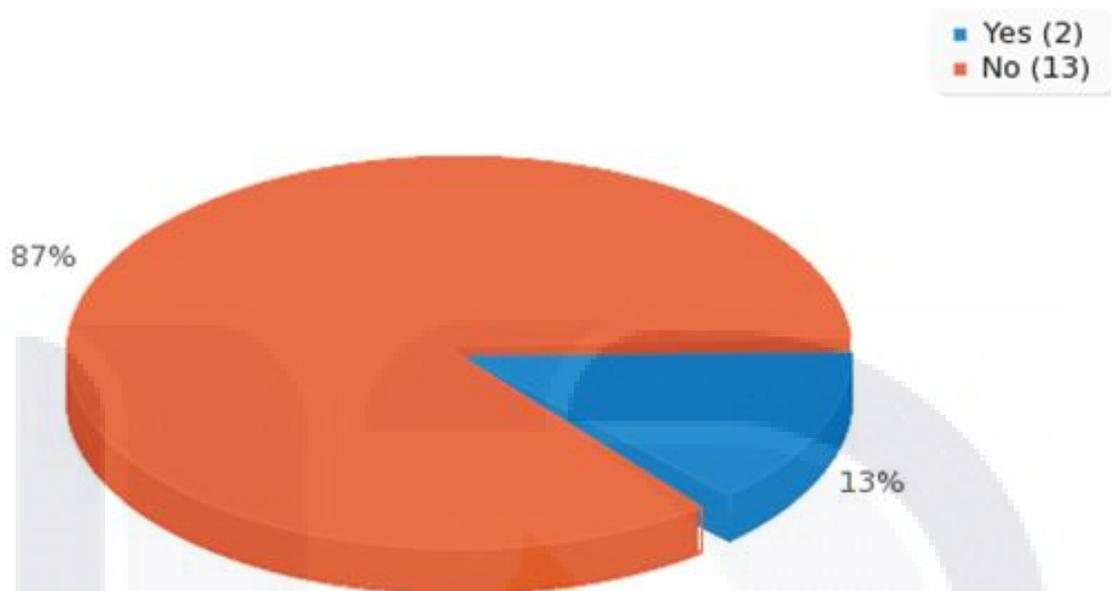


Figura 40. Herramientas automatizadas de pruebas

De la Figura 35 a la Figura 50 se presentan en gráficas de pastel los resultados del comportamiento de la quinta y última sección del instrumento de evaluación, la cual abarcó la etapa de pruebas de software.

La presente sección mostro el nivel más bajo de implementación en la cual solamente una de las prácticas supero el 50% de implementación.

Se realizaron gráficos de pastel en base a porcentajes resultantes sobre las características demográficas de las organizaciones y para cada una de las mejores prácticas de ingeniería de software con la finalidad de conseguir una mejor interpretación de los resultados obtenidos.

Al realizar un análisis detallado para cada una de las mejores prácticas de ingeniería de software observamos la alta utilización de mejores prácticas en actividades de gestión de proyectos de software, una utilización promedio en las etapas de análisis, diseño y codificación de software y finalmente una baja utilización de mejores prácticas en la fase de pruebas de software.

Las características demográficas de las empresas en su mayoría reflejan ser empresas que cuentan con menos de 10 empleados y que su tiempo de existencia no sigue una tendencia sobre un determinado tiempo de haber sido creadas.

A partir de estos resultados se puede analizar el nivel de implementación de las mejores prácticas en forma individual, observar cuales son las más utilizadas y cuales simplemente tienen un nivel de utilización casi nulo.

Objetivos Específicos

Objetivo 1

Identificar la influencia de la implementación de mejores prácticas en base a la estructura organizacional de las empresas.

Para el cumplimiento de este objetivo se definieron dos hipótesis las cuales fueron evaluadas en base al índice de correlación Rho de Spearman oscilando entre -1 y 1 (-1 = Correlación negativa, 0 = Sin Correlación, 1 = Correlación Positiva) con un nivel de significancia de 0.05

H1: La utilización de mejores prácticas de ingeniería de software está influenciada por el tamaño de la organización.

			Número de empleados con que cuenta la organización	Totales Ajustados
Rho de Spearman	número de empleados con que cuenta la organización	Coefficiente de correlación	1.000	.318
		Sig. (bilateral)	.	.248
		N	15	15
Totales Ajustados		Coefficiente de correlación	.318	1.000
		Sig. (bilateral)	.248	.
		N	15	15

Tabla 15. Correlaciones Tamaño de las Organizaciones

Como nos indica el índice de correlación Rho de Spearman (ver tabla 15), a medida que incrementa el número de empleados con los que cuenta la organización se aprecia que no aumenta la utilización de mejores prácticas de

ingeniería de software, con un valor de $p=.248$, $R =.318$, por lo cual la hipótesis es rechazada.

Para obtener un panorama más profundo se evaluaron las correlaciones de cada una de las secciones con respecto al tamaño de la empresa.

Correlaciones

			número de empleados con que cuenta la organización					
Rho de Spearman	número de empleados con que cuenta la organización	Coefficiente de correlación Sig. (bilateral)	1.000	.688**	.163	.225	.275	.253
				.005	.562	.420	.322	.364
		N	15	15	15	15	15	15

Tabla 16. Correlaciones Tamaño de las Organizaciones por Secciones

Como observamos en la Tabla 16 de correlaciones de Rho de Spearman, a medida que incrementa el número de empleados con los que cuenta la organización se aprecia que únicamente aumenta la utilización de mejores prácticas de ingeniería de software durante la fase de gestión de proyectos de software (ver Figura 11), con los un valor de $p=.005$, $R =.688$

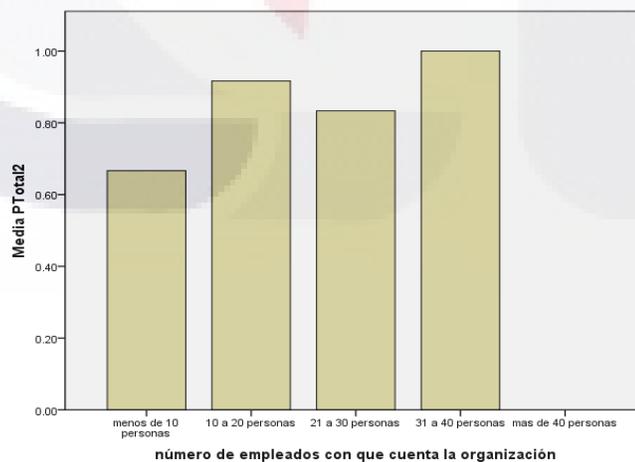


Figura 41. Correlación Tamaño de la Organización, Gestión de Proyectos

H2: La utilización de mejores prácticas de ingeniería de software está influenciada por el tiempo de existencia de la organización.

Correlaciones

			Tiempo de existencia de la organización	Totales Ajustados
Rho de Spearman	tiempo de existencia de la organización	Coefficiente de correlación	1.000	-.200
		Sig. (bilateral)	.	.474
		N	15	15
Totales Ajustados		Coefficiente de correlación	-.200	1.000
		Sig. (bilateral)	.474	.
		N	15	15

Tabla 17. Correlaciones Tiempo de Existencia de las Organizaciones

Como nos indica el índice de correlación Rho de Spearman (Tabla 17), a medida que incrementa el tiempo de existencia de la organización, se aprecia que no aumenta la utilización de mejores prácticas de ingeniería de software, con un valor de $p = .474$, $R = -.200$, por lo cual la hipótesis es rechazada.

Para obtener un panorama más profundo se evaluaron las correlaciones de cada una de las secciones con respecto al tamaño de la empresa (ver tabla 18).

Correlaciones

			tiempo de existencia de la organización	PTotal2	PTotal3	PTotal4	PTotal5	PTotal6
Rho de Spearman	tiempo de existencia de la organización	Coefficiente de correlación	1.000	.152	-.411	.048	.196	-.305
		Sig. (bilateral)	.	.588	.128	.866	.483	.269
		N	15	15	15	15	15	15

Tabla 18. Correlaciones Tiempo de Existencia por Sección

Como observamos en la tabla 18 de correlaciones de Rho de Spearman a medida que incrementa el tiempo de existencia de la organización, ninguna de las secciones estudiadas aumenta la utilización de mejores prácticas de ingeniería de software (Ver Figura 12).

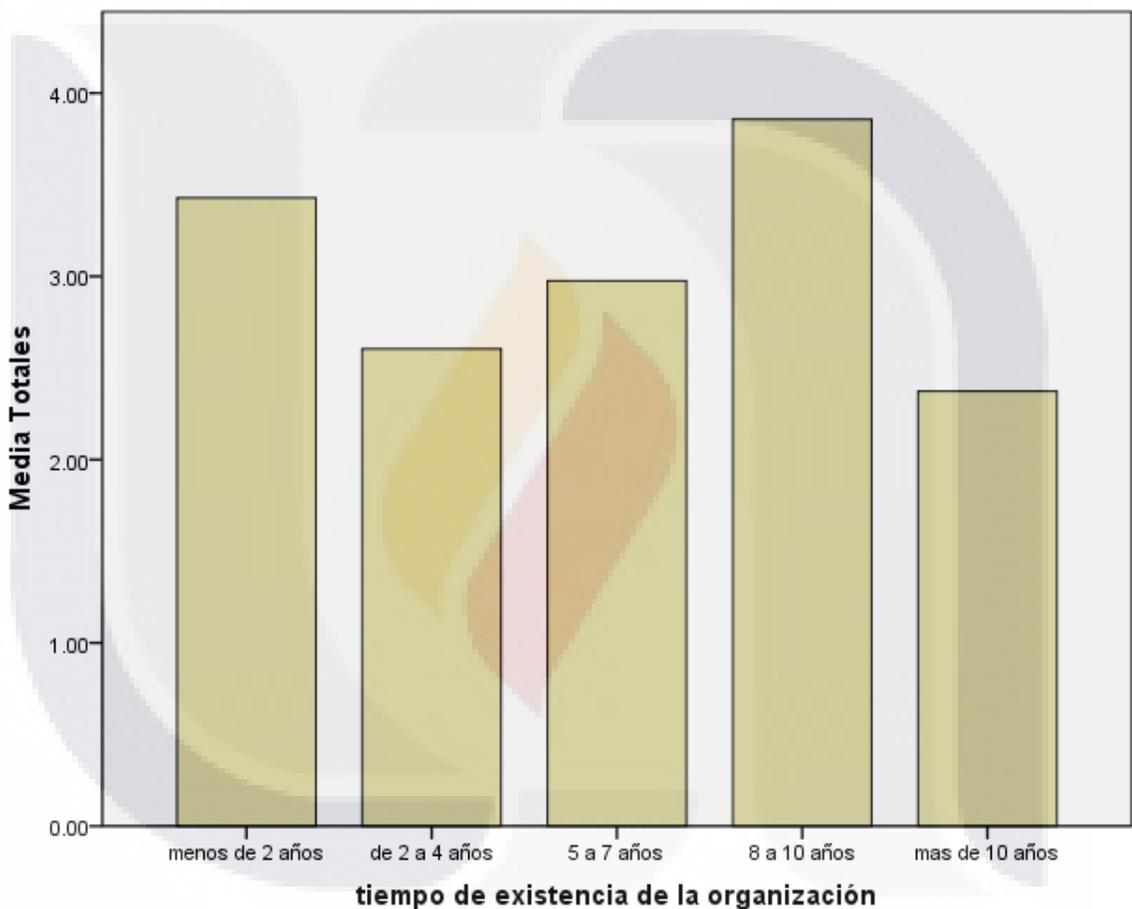


Figura 42. Correlación Tiempo de Existencia de la Organización

Objetivo 2

Conocer si existe un interés por implementar un marco de trabajo formal de mejores prácticas a corto plazo (Menor a 2 años).

H3: Existe interés por implementar un marco de trabajo de mejores prácticas en un plazo menor a 2 años.

En la Tabla 19 se presentan los resultados obtenidos de las empresas que no cuentan con un modelo de mejores prácticas de ingeniería de software, observando en la Figura 13 que un 81.8 % de las organizaciones tienen planeado implementar un modelo de mejores prácticas en un lapso menor a dos años.

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	0	2	18.2	18.2	18.2
	1	9	81.8	81.8	100.0
	Total	11	100.0	100.0	

Tabla 19. Porcentajes, Interés de Implementación

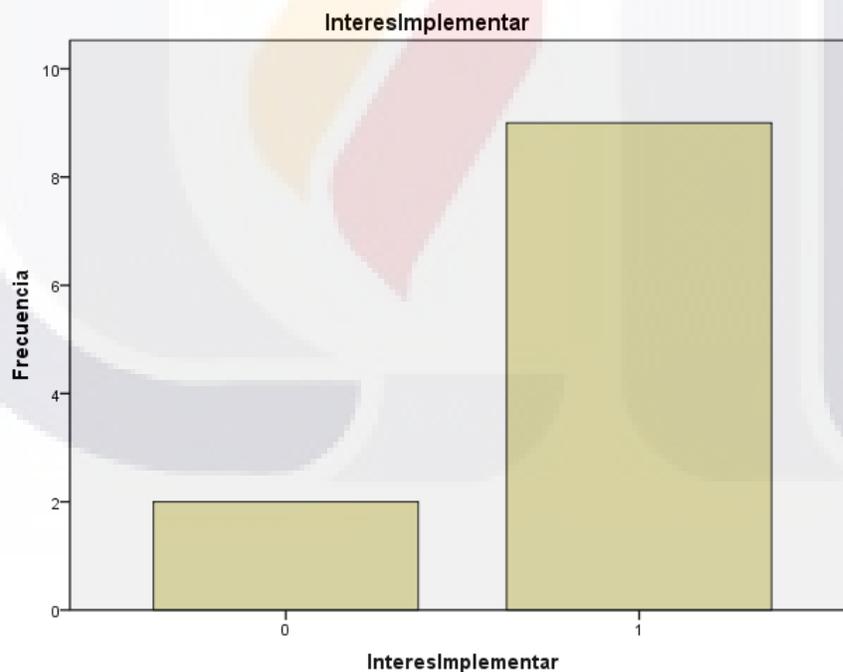


Figura 43. Interés de Implementación Modelo de Mejores Prácticas

La Figura 14 muestra que las organizaciones que no se encuentran interesadas en implementar un modelo de mejores prácticas son empresas menores a 10 personas.

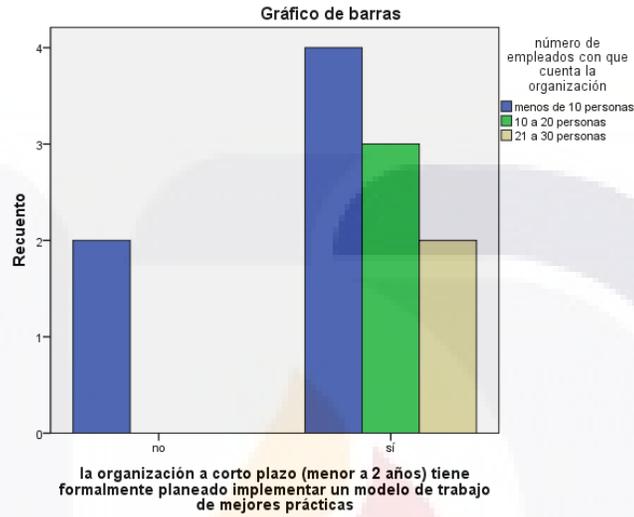


Figura 44. Contingencia Tamaño de la Empresa e Interés de Implementación

La Figura 15 muestra que las organizaciones que no se encuentran interesadas en implementar un modelo de mejores prácticas son empresas que se encuentran en el intervalo de dos a siete años de haber sido creadas.

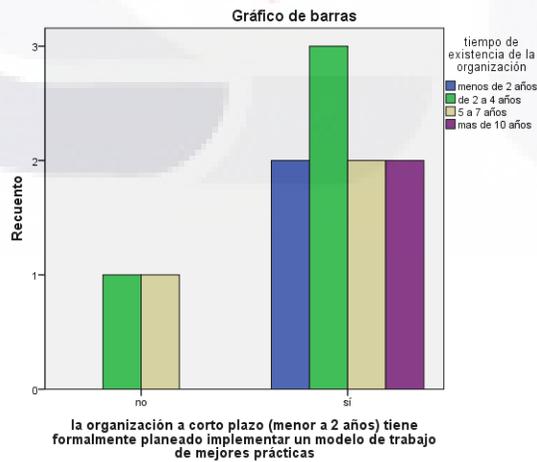


Figura 45. Contingencia Tiempo de Existencia, Interés de Implementación

Objetivo 3

Conocer si existe una mayor implementación de mejores prácticas en cierta etapa del ciclo de desarrollo del producto.

H4: Existen diferencias significativas en la utilización de mejores prácticas de ingeniería de software entre las fases de desarrollo de un producto.

Para poder llegar a cumplir el objetivo definido se realizaron pruebas de hipótesis no paramétricas de k muestras independientes Kruskal-Wallis, a continuación se presenta el análisis de la prueba.

H0: M3 = M4 = M5 = M6

Ha: No son significativamente iguales

Nivel de significancia: $\alpha = 0.05$

Se rechaza la hipótesis nula en caso de que el valor de $p \leq 0.05$

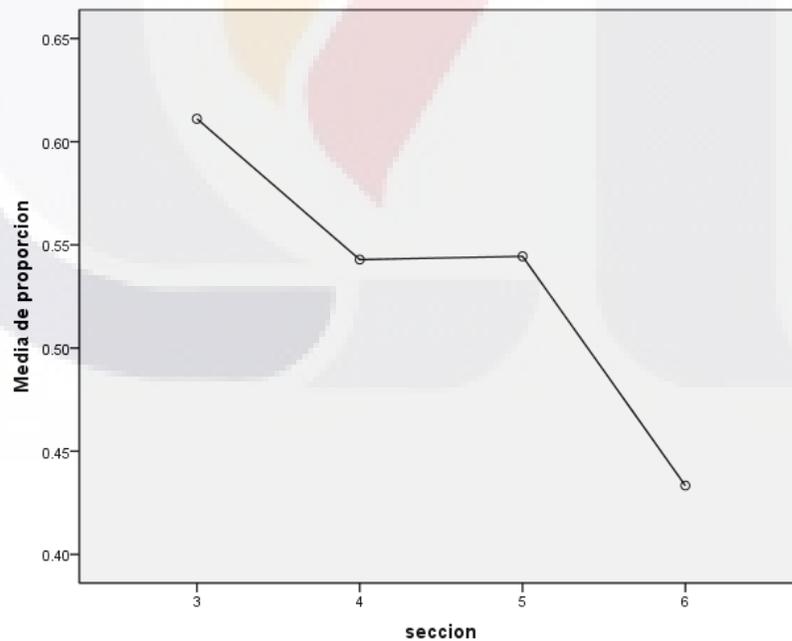


Figura 46. Prueba de Medias por Sección

sección		N	Rango promedio
Proporción	3	15	35.30
	4	15	31.50
	5	15	31.03
	6	15	24.17
	Total	60	

Tabla 20. Prueba K-Muestras Independientes Kruskal-Wallis

La tabla 20 muestra los rangos promedios de proporciones de las secciones 3 a 6 del instrumento de evaluación, con la finalidad de que estos puedan ser comparados de manera estadística.

	proporcion
Chi-cuadrado	3.223
gl	3
Sig. asintót.	.359

Tabla 21. Estadísticos de Contraste

Como se observa en la Figura 16 y conforme a la Tabla 21, debido a que el valor de p 0.359 es mayor a α (0.05) es aceptada la hipótesis nula por lo que se concluye que no existe una diferencia en la utilización de mejores prácticas por etapa de desarrollo del producto.

Gráfico de utilización de mejores prácticas de ingeniería de software segmentado por aquellas empresas que cuentan con un modelo de mejores prácticas implementado (color verde) y por aquellas empresas que no cuentan con el (color azul).

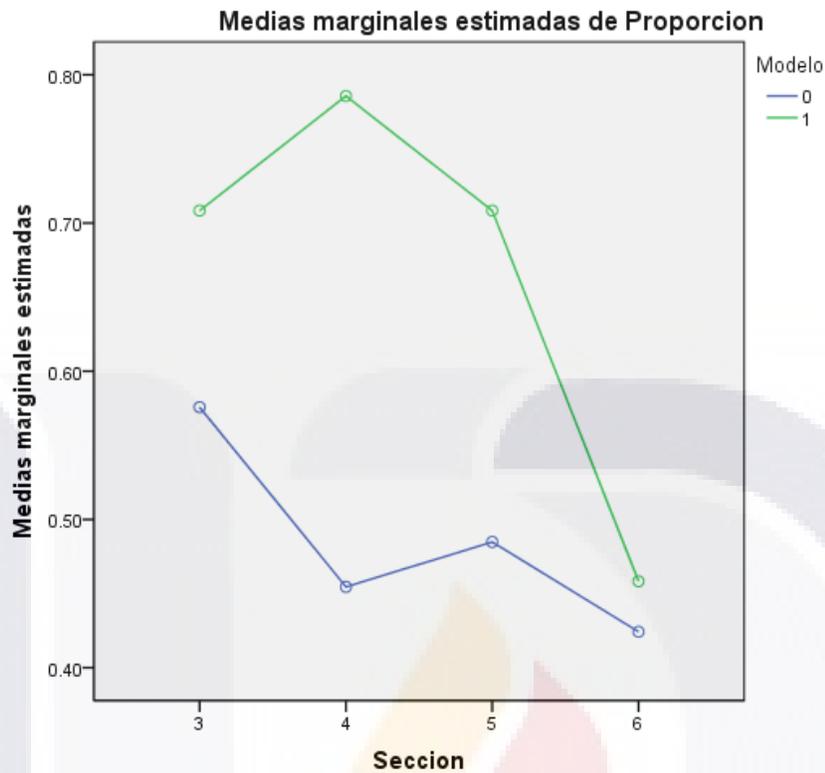


Figura 47. Gráfico de Medias Segmentado Sección/Modelo

Como se observa en la Figura 17 la utilización de mejores prácticas de ingeniería de software en el estado de Aguascalientes tiende a disminuir en la etapa de pruebas de software siendo esta su utilización casi nula.

Una vez realizada esta prueba de hipótesis se realizó una réplica adhiriendo la sección de administración de proyectos de software para observar su comportamiento. A continuación se presentan los resultados.

H0: $M_2 = M_3 = M_4 = M_5 = M_6$

Ha: No son significativamente iguales

Nivel de significancia: $\alpha = 0.05$

Se rechaza la hipótesis nula en caso de que el valor de $p \leq 0.05$

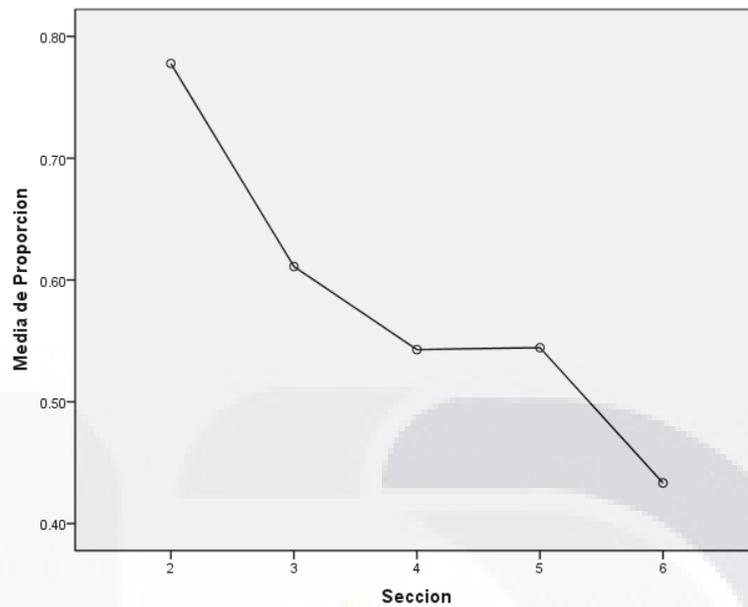


Figura 48. Prueba de Medias por Sección

	Sección	N	Rango promedio
Proporción	2	15	53.00
	3	15	39.87
	4	15	35.73
	5	15	34.47
	6	15	26.93
	Total	75	

Tabla 22. Prueba K-Muestras Independientes Kruskal-Wallis

La tabla 22 muestra los rangos promedios de proporciones de las secciones 2 a 6 del instrumento de evaluación, con la finalidad de que estos puedan ser comparados de manera estadística.

	Proporción
Chi-cuadrado	11.867
gl	4
Sig. asintót.	.018

Tabla 23. Estadísticos de Contraste

Como se observa en la Figura 18 y conforme a la Tabla 23, debido a que el valor de p 0.018 es menor a α (0.05) es rechazada la hipótesis nula por lo que se concluye que existe diferencia en la utilización de mejores prácticas por etapa de desarrollo del producto incluida la gestión de proyectos de software.

Gráfico de utilización de mejores prácticas de ingeniería de software segmentado por aquellas empresas que cuentan con un modelo de mejores prácticas implementado (color verde) y por aquellas empresas que no cuentan con el (color azul).

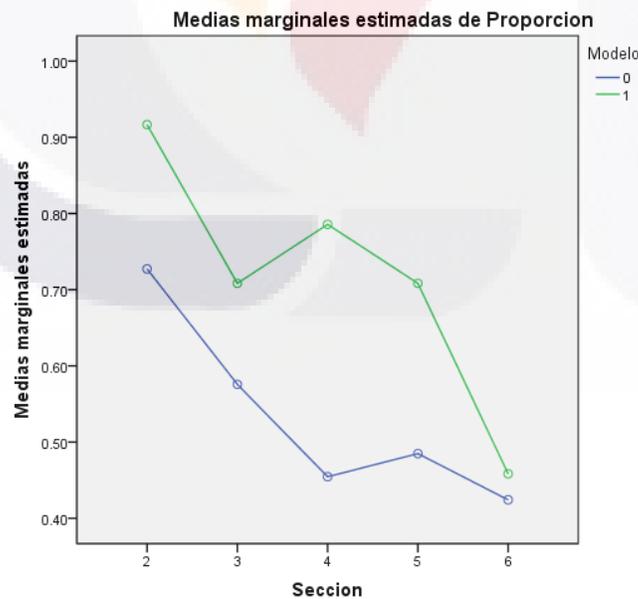


Figura 49. Gráfico Prueba de Medias Segmentado

Como se observa en la Figura 19 la utilización de mejores prácticas de ingeniería de software en el estado de Aguascalientes tiende a disminuir en la etapa de pruebas de software siendo esta su utilización casi nula.

5.3 Limitaciones y Problemas Encontrados.

Durante el transcurso de la investigación se enfrentaron con diversos aspectos que llegaron a impactar el desarrollo óptimo de la investigación. A continuación se describe cada uno de ellos:

- Se obtuvo una base de datos de empresas desarrolladoras de software en el Estado de Aguascalientes totalmente desactualizada y con información errónea por parte de la Secretaria De Desarrollo Económico del Estado (SEDEC).
- Con el apoyo del CONCYTEA y del director del clúster de tecnologías de información del estado el Ingeniero Pedro Alberto Tinoco Ariza se logró la actualización del listado de las empresas desarrolladoras de software en el estado, obteniendo un total de 21 organizaciones que cumplieran con el perfil de la investigación.
- Seis de las 21 empresas no se mostraron interesadas en apoyar la investigación.
- Debido al tipo de escala utilizada (dicotómica) y las hipótesis planteadas se requirió profundizar en temas sobre pruebas de hipótesis no paramétricas (k muestras independientes / Kruskal Wallis) las cuales requieren de un tratado especial.

6. CONCLUSIONES.

6.1 Conclusiones de Resultados Obtenidos.

Las conclusiones del presente trabajo serán presentados a partir de los objetivos definidos en el inicio de la investigación e hipótesis evaluadas a partir de estos.

Objetivo General

Objetivo 1: Identificar un panorama sobre las mejores prácticas implementadas por las MiPyMEs desarrolladoras de software en el estado de Aguascalientes las cuales podrían llegar a ser útiles en la creación y crecimiento de MiPyMEs enfocadas al desarrollo de software.

Se observó que más del 50% de las empresas están constituidas por menos de 10 empleados. Siendo en su mayoría empresas que tienen entre 2 y 4 años de existencia.

A continuación se presentará en la Tabla 24 las mejores prácticas en orden de utilización por fase en las que fueron clasificadas, en la cual se indica de manera subrayada aquellas que superaron el 50% de utilización y en cursiva aquellas que no.

Sección	Mejores Prácticas en Orden de Utilización
Administración de Proyectos de Software	<u>Líder de Proyecto</u> <u>Estimación Costo/Tiempo</u> <u>Planeación Documentada</u> <u>Revisiones Periódicas</u> <u>Entrenamiento</u> <i>Métricas</i>

Sección	Mejores Prácticas en Orden de Utilización
Análisis de Software	<u>Beneficios y Viabilidad del Proyecto</u> <u>Ciclo de Desarrollo del Producto</u> <u>Nuevas Tecnologías</u> <u>Notación Para el Modelado del Análisis</u> <u>Formato Para Obtención de Requerimientos</u> <i>Proceso En Cambio de Requerimientos</i>
Diseño de Software	<u>Diseño de Interfaces</u> <u>Normalización de Datos</u> <u>Diccionarios de Datos</u> <i>Herramientas CASE</i> <i>Acoplamiento y Cohesión</i> <i>Notaciones Para el Modelado del Diseño</i> <i>Patrones de Diseño y Arquitectura</i>
Codificación de Software	<u>Ambientes de Desarrollo</u> <u>Convenciones (Estándares)</u> <u>Sistema de Control de Cambio</u> <i>Documentación Técnica del Código</i> <i>Documentación de Control de Cambio</i> <i>Prototipado</i>
Pruebas de Software	<u>Pruebas Usuarios Finales</u> <i>Plan de Pruebas</i> <i>Revisiones de Pares</i> <i>Casos de Prueba</i> <i>Pruebas Independientes</i> <i>Automatización de Pruebas</i>

Tabla 24. Mejores Prácticas en Orden de Utilización

Como se describe en la tabla 24 las fases que cuenta con un mayor número de mejores prácticas implementadas son las secciones de administración de proyectos de software y análisis de software alcanzando 5 de 6 prácticas un porcentaje mayor al 50%. En contraste la fase de pruebas de software presentó una baja implementación de mejores prácticas, obteniendo sólo una de las mejores prácticas de esta sección un porcentaje mayor al 50%.

Objetivos Específicos

Objetivo 1: Identificar la influencia de la implementación de mejores prácticas en base a la estructura organizacional de las empresas.

Se definieron dos hipótesis para el cumplimiento de este objetivo, la primera relacionada al tamaño de la organización y la segunda al tiempo de existencia de la organización. Después de haber sido evaluadas dichas hipótesis a partir de pruebas de correlación con el índice de Rho de Spearman se concluye que no existe un aumento en la utilización de mejores prácticas en base al tamaño y tiempo de existencia de las organizaciones.

Posteriormente se pasó al realización de pruebas de hipótesis independientes para cada una de las secciones evaluadas en la investigación en las cuales se encontró que la sección de administración de proyectos de software es en la única que aumenta la utilización de mejores prácticas de ingeniería de software en base al tamaño de la organización, más no así en el tiempo de existencia de éstas.

Objetivo 2: Conocer si existe un interés por implementar un marco de trabajo formal de mejores prácticas a corto plazo (Menor a 2 años).

El 73% de las empresas no cuentan con un modelo de mejores prácticas. De las empresas que no cuentan con un modelo de mejores prácticas el 81.8% manifestaron interés por implementarlo en un lapso menor a dos años.

Este porcentaje manifiesta el interés de las empresas en mejorar la forma en la que realizan sus productos y así poder llegar a competir en mercados más amplios.

El porcentaje que no está interesado en la implementación de un modelo de mejores prácticas son empresas que cuentan con menos de 10 empleados y cuentan con menos de 4 años de haber sido creadas.

Objetivo 3: Conocer si existe una mayor implementación de mejores prácticas en cierta etapa del ciclo de desarrollo del producto.

El cumplimiento de este objetivo se logró a partir de la categorización de cada una de las secciones y la evaluación de pruebas de hipótesis no paramétricas con la comparación de sus respectivas medias y varianzas.

Se realizó una primera evaluación de la prueba de hipótesis tomando en cuenta las secciones de análisis, diseño, codificación y pruebas mostrándonos que no existen diferencias significativas en su comportamiento, sin embargo al hacer la comparación con la sección de administración de proyectos se observa una mayor implementación de mejores prácticas mostrando diferencias significativas con la sección de pruebas de software.

Posteriormente se contrastó el comportamiento de las empresas que cuentan con un modelo de mejores prácticas y aquellas que no lo cuentan, mostrándonos que

disminuye considerablemente la utilización de prácticas durante la etapa de pruebas de software para ambas tipo de organizaciones.

6.2 Conclusiones del Método de Investigación Empleado.

El trabajo realizado siguió la metodología de investigación cuantitativa propuesta por Roberto Hernández Sampieri en su libro Metodología de la Investigación. El método de investigación fue la guía para poder llevar a cabo una investigación ordenada y con objetivos bien definidos desde un principio, lo cual fue esencial para la terminación del presente trabajo de investigación en tiempo y forma debido a la definición y de cada una de las tareas que involucro este trabajo de investigación.

6.3 Conclusiones de Aprendizaje Personal.

Durante el desarrollo de este proyecto de investigación se logró obtener un crecimiento personal a la par que este fue desarrollado. El trabajo se realizó en el transcurso de dos años en los cuales se cumplió con cada uno de los requisitos que la universidad exigió así como de tener la disciplina necesaria para cumplir con los objetivos definidos inicialmente en la investigación. Se conocieron y debatieron diferentes perspectivas del tema en conjunto del tutor de tesis y asesores, maestros, compañeros de clases y organizaciones participantes. Cada uno de los elementos mencionados ayudó a la mejora continua del presente trabajo el cual se espera que llegue a tener un impacto positivo en el crecimiento de la industria de software del estado de Aguascalientes.

6.4 Conclusiones y Recomendaciones Finales.

La utilización de mejores prácticas dentro de las MiPyMes durante la fabricación de sus productos y/o servicios beneficia al control del desarrollo de éstos y aumenta su cartera de clientes potenciales. La industria de software en el estado de Aguascalientes demostró tener una actitud positiva en la colaboración con la investigación debido al interés mostrado en los resultados que se pudieran llegar a generar, todo esto con la finalidad de que sean útiles en su crecimiento organizacional.

Como trabajo futuro se buscará realizar una réplica (estudio con característica longitudinal) para poder llegar a generar un punto de referencia que indique la evolución en la forma de trabajo de las organizaciones y el crecimiento de la industria de software.

La realización de un estudio con base en aquellas empresas que tienen formalmente planeado la implementación de un modelo de mejores prácticas a corto plazo (menor a 2 años) verificando si esto se realiza de manera óptima y en caso contrario estudiar cuáles fueron los factores involucrados para que esto no ocurriera.

Un trabajo futuro más ambicioso es el de generar un modelo que cuente con cada una de las mejores prácticas categorizadas por etapa de desarrollo del producto, que se cuente con un soporte estable ya sea documentado o por medio de consultores y en consecuente se generen bases de datos de conocimiento tomando en cuenta experiencias previas las cuales facilitarían en gran medida la implementación de mejores prácticas.

BIBLIOGRAFÍA.

- [1] Economía, Secretaría de., *Programa PROSOFT v1.3*. 2006.
- [2] —. DETI. *Directorio de Empresas de Tecnologías de Información*. [En línea] 25 de Marzo de 2009. <http://www.edigital.economia.gob.mx/deti/>.
- [3] Mexicano, Sistema de Información Empresarial., [En línea] 26 de Marzo de 2009. <http://www.siem.gob.mx/siem2008/estadisticas/EstadoTamano.asp?p=1>.
- [4] Economía, Subsecretaría de Industria y Comercio Secretaría de., [En línea] <http://www.google.com.mx/url?sa=t&source=web&ct=res&cd=6&ved=0CBoQFjAF&url=http%3A%2F%2Fwww3.diputados.gob.mx%2Fcamara%2Fcontent%2Fdownload%2F4564%2F24830%2Ffile%2FPresentacion%2520del%2520PROSOFT.pdf&rct=j&q=Programa+para+el+Desarrollo+de+la+Industria+de>.
- [5] Sampieri, Dr. Roberto Hernández., *Metodología de la Investigación*. s.l. : McGraw-Hill, 2006. 970-10-5753-8.
- [6] Danhke, G. L., *Investigación y Comunicación*. s.l. : McGraw-Hill, 1989.
- [7] Mexicano, Sistema de Información Empresarial., [En línea] [Citado el: 26 de Marzo de 2009.] <http://www.siem.gob.mx/siem2008/estadisticas/EstadoTamano.asp?p=1>.
- [8] Valencia, Joaquín Rodríguez., *Administración de Pequeñas y Medianas Empresas*. México : ECAFSA, 2002.
- [9] Economía, Secretaría de., Sistema de Información Empresarial Mexicano. [En línea] 24 de Marzo de 2009. <http://www.siem.gob.mx/siem2008/>.

- TESIS TESIS TESIS TESIS TESIS
- [10] Randall, P. Naur y B., *Software Engineering: A Report on a Conference Sponsored by the NATO Science Committee*. s.l. : NATO.
- [11] IEEE., *IEEE Standar Collection: Software Engineering*. s.l. : IEEE, 1993.
- [12] Somerville, Ian., *Ingeniería del software*. s.l. : Pearson Addison Wesley, 2005.
- [13] Initiative, European Systems and Software., *Software Best Practice (ESSI). Supporting the growth and the spread of the Information Society*. [En línea] Mayo de 1997. <http://cordis.europa.eu/esprit/src/essi.htm>.
- [14] Somerville, Ian., *Ingeniería del software*. s.l. : Pearson Addison Wesley, 2005. 7 edición.
- [15] Alarcón, Armando Silva., *Modelos de calidad. La industria del software en México*. [En línea] <http://www.enterate.unam.mx/Articulos/2004/Enero/modelos.htm>.
- [16] Pressman, Roger S., *Ingeniería del Software*. Sexta Edición. s.l. : Mc Graw Hill, 2005. 970-10-5473-3.
- [17] Boehm, B., *Software Engineering Economics*. s.l. : Prentice Hall, 1981.
- [18] Soumitra Dutta, Michael Lee, Luk Van Wassenhove., *Software Engineering in Europe: A Study of Best Practices*. s.l. : IEEE, 1999.
- [19] Danny C.C. Poo, Mui Ken Chung., *Software Engineering Practices in Singapore*. s.l. : Department of Information Systems, School of Computing, National University of Singapore, 1995.
- [20] María de la Luz Villalobos Hernández, Agustín Francisco Gutiérrez Tornés., *Investigación sobre las prácticas de ingeniería de software en México*. s.l. :

Instituto Politécnico Nacional, 2001.

[21] Eugenia Marquez, Gabriela Gaetan, Viviana Saldaño, Patricia Yañez., *Prácticas de Ingeniería de software en un entorno académico*. Argentina : JEITICS, 2005.

[22] Loevinger, J. A., *A systematic approach to the construction and evaluation of test of ability*. 1947.

[23] Freed N. Kerlinger, Howard B. Lee., *Foundations of behavioral research*.



ANEXOS.

A.1 Empresas de TI por Estado Registradas en México.

Estado	No. Empresas
Distrito Federal	368
Nuevo León	158
Jalisco	133
Veracruz	76
Estado De México	71
Puebla	70
Sonora	56
Aguascalientes	49
Sinaloa	48
Baja California	45
Guanajuato	38
Querétaro	36
Tabasco	36
Coahuila	31
Colima	31
Yucatán	30
Chihuahua	26
Michoacán	23
Zacatecas	22
Morelos	21
Tamaulipas	19
Tlaxcala	18
Oaxaca	14
Chiapas	12
Durango	12
Quintana Roo	12
San Luis Potosí	9
Baja California Sur	8
Hidalgo	8
Guerrero	6
Campeche	4
Nayarit	2

A.2 Software Best Practice (ESSI) Questionnaire

Guidelines

1. Please attempt to answer ALL questions either YES, NO, or N.A. (Not Applicable). N.A. may be a valid response for those questions which are applicable for only certain types of developer. Questions of this type are indicated by brackets around the number of the question, e.g.(1.8), and by an "Applies to" comment set with italics.

2. "Don't know" or "don't understand the question" responses should be reflected as a NO.

3. Please answer according to normal organisational practice - not ideal practice or according to unimplemented standards.

General Information

Organisation Type ¹
Country ²
Number of Employees
Number of Employees involved in Software Engineering
Organisation primary involvement in software industry (delete those descriptions which do not apply): Software user (developed in-house) Software user (developed by a 3rd party)
Software vendor (producing off-the-shelf systems) Software vendor (producing custom software systems) Research & Development institute or university Interest Group (e.g. professional society or standards body) Other (please specify):
Industrial Sector ³
Community of Interest ⁴ (delete those which do not apply) Business Orientated Systems Technically Orientated Systems Embedded and Control Systems

1 Specify one of the following codes: 'U' for University, 'R' for Research Institute, 'I' for Industry, 'A' for Public Administration, 'O' for Other.

2 A list of country codes is supplied with Form 2 of section 6.1 of the 'Software Best Practice (ESSI) - Specific Information Document'.

3 A list of codes for industrial sectors is supplied with Form 2 of section 6.1 of the 'Software Best Practice (ESSI) - Specific Information Document'.

4 'Business Orientated Systems' would e.g. include the software process of all traditional MIS and EDP systems, found in banks, insurances, industry and public administration; for instance sales and marketing support systems, personnel management and administration systems, financial information systems, reservation systems, office automation systems, etc.

'Technically Orientated Systems' would e.g. include the development of systems aimed at engineering professionals, which may be used in a business or control environment; for instance computer aided design/manufacturing systems, computer aided software engineering tools, geographic information systems, simulation systems, document image processing systems, etc.

'Embedded and Control Systems' would e.g. include all software applications which are embedded within systems, which the user may or may not be aware of their existence; for instance in industrial automation and process control systems, signal processing systems, telecommunications systems, consumer products, etc.

Section 1 - Organisational Issues (8 questions)

<p>1.1 Does each software project have a nominated software project manager? Comments:</p>	(YES/NO)
<p>1.2 Does the software project manager report to a business project manager responsible for the overall benefit of the project to the business? Comments:</p>	(YES/NO)
<p>1.3 Does a Software Quality Assurance (SQA) function exist within an independent reporting line from software development project management? Comments:</p>	(YES/NO)
<p>1.4 Is a change control function established for each software project? Comments:</p>	(YES/NO)
<p>1.5 Is there a required training programme for all newly-appointed software managers which is designed to familiarise them with in-house software project management procedures? Comments:</p>	(YES/NO)
<p>1.6 Is there a procedure for maintaining awareness of the state-of-the-art in CASE or software engineering technology? Comments:</p>	(YES/NO)
<p>1.7 Is there a procedure for ensuring that appropriate levels of user/customer/marketing input is made throughout the project? Comments:</p>	(YES/NO)
<p>(1.8) Where other non-software resources are critical to the success of the project is there a procedure for ensuring their availability according to plan? <i>Applies to: Organisations with projects dependent on a number of non-software resources such as hardware design specialists, computer operations staff, data administrators, or computer network staff.</i> Comments:</p>	(YES/NO/N.A.)

Section 2 - Standards and Procedures (13 questions)

<p>2.1 Do management formally assess the benefits, viability, and risk of each software project prior to making contractual (or internal) commitments?</p> <p>Comments:</p>	(YES/NO)
<p>2.2 Do management formally conduct periodic reviews of the status of each software project?</p> <p>Comments:</p>	(YES/NO)
<p>(2.3) Are there procedures to ensure that external software subcontracting organisations, if any, follow a disciplined software development process?</p> <p><i>Applies to: Those organisations which contract out software development to other organisations.</i></p> <p>Comments:</p>	(YES/NO/N.A.)
<p>2.4 For each project, are independent audits (such as inspections or walkthroughs) conducted for each major stage in the software development process?</p> <p>Comments:</p>	(YES/NO)
<p>2.5 Are common coding standards applied to each software project?</p> <p>Comments:</p>	(YES/NO)
<p>(2.6) Is there a documented procedure for estimating software size (such as "Lines of Source Code") and thus for using productivity measures?</p> <p><i>Applies to: Organisations developing predominantly in a single language such as C or Cobol.</i></p> <p>Comments:</p>	(YES/NO/N.A.)
<p>2.7 Is a formal procedure used to produce software development effort, schedule, and cost estimates?</p> <p>Comments:</p>	(YES/NO)
<p>2.8 Is a formal procedure (such as a review or handover with sign-off) used whenever a deliverable (such as a user statement of requirements or system requirements) is passed from one discrete group to another (e.g. <i>user</i> to <i>analyst</i> to <i>designer</i>) to ensure it is properly understood?</p> <p>Comments:</p>	(YES/NO)
<p>(2.9) Is there a mechanism to ensure that the systems projects selected for development qualitatively or quantitatively support/alleviate the organisation's business objective/problems.</p>	(YES/NO/N.A.)
<p>(2.10) Are there procedures to ensure that the functionality, strengths, and weaknesses of the "system" which the software is replacing are formally reviewed?</p> <p><i>Applies to: Organisations whose software is intended to replace a previously computer-based mechanical or clerical set of tasks.</i></p> <p>Comments:</p>	(YES/NO/N.A.)

2.11 Does test planning commence prior to programming beginning based on the user requirements and high-level design documents? Comments:	(YES/NO)
2.12 Is independent testing conducted by users (or appropriate representatives) under the guidance of Software Quality Assurance before any system or enhancement goes live? Comments:	(YES/NO)
2.13 Is there a procedure to check that the system configuration (i.e. the programs and any data) passing user acceptance testing is the same as that which is implemented for liveoperation and that no changes are made directly to a "live" version of any system (other than through modification to its development version)? Comments:	(YES/NO)

Section 3 - Metrics (8 questions)

3.1 Are records of actual project resourcing and timescales versus estimates maintained (at individual resource/resource-type level) and regularly analysed/fed-back into the estimating and scheduling procedures? Comments:	(YES/NO)
(3.2) Are records of software size maintained for each software configuration item, over time, and fed-back into the estimating process? <i>Applies to: as per 2.6.</i> Comments:	(YES/NO/N.A.)
3.3 Are statistics on the sources of errors in software code gathered and analysed for their cause, detection and avoidance measures? Comments:	(YES/NO)
3.4 Are statistics on test efficiency (% of errors actually detected by an activity against the maximum theoretically possible) gathered and analysed for all testing stages in the development process? Comments:	(YES/NO)
3.5 Is "earned value" project tracking used throughout the software development process (actual versus planned deliverables analyses, designed, unit tested, system tested, acceptance tested over time) to monitor project progress? Comments:	(YES/NO)
(3.6) Are estimates made and compared with actuals for target computer performance (e.g. memory utilisation, processor throughput and file/channel I/O and disk usage)? <i>Applies to: Where the user/customer has explicit or implicit performance expectations or computer resource constraints.</i> Comments:	YES/NO/N.A.)
3.7 Are post-implementation software problem reports logged and their resolution effectively tracked and analysed? Comments:	(YES/NO)
3.8 Do records exist from which (and requiring nothing extra) all current versions and variants of software systems and their components can be quickly and accurately reconstructed in the development environment? Comments:	(YES/NO)

Section 4 - Control of the Development Process (6 questions)

<p>4.1 Are estimates, schedules and subsequent changes produced only by the project managers who directly control the project resources and are fully aware of their abilities and availabilities? Comments:</p>	<p>(YES/NO)</p>
<p>4.2 Does the overall business project manager gain agreement and sign-off from all parties (YES/NO)who have produced detailed estimates and schedules before publishing or revising a consolidated project plan? Comments:</p>	<p>(YES/NO)</p>
<p>4.3 Is there a procedure for controlling changes to the software requirements, designs and documentation? Comments:</p>	<p>(YES/NO)</p>
<p>4.4 Is there a procedure for controlling changes to the code and specifications? Comments:</p>	<p>(YES/NO)</p>
<p>4.5 Is there a mechanism for assuring that regression testing (i.e. the forced re-run of all previous tests prior to any new tests) is routinely performed during and after initialimplementation? Comments:</p>	<p>(YES/NO)</p>
<p>4.6 Do procedures exist to ensure that every required function is tested/verified? Comments:</p>	<p>(YES/NO)</p>

Section 5 - Tools and Technology (7 questions)

<p>5.1 Are software tools used to assist in forwards and/or backwards tracing of software requirements to software designs through to code? Comments:</p>	(YES/NO)
<p>5.2 Are design notations such as SADT used in program design? Comments:</p>	(YES/NO)
<p>5.3 Are automated testing tools used (for example for capturing and replaying tests, or for) ensuring logic paths coverage? Comments:</p>	(YES/NO)
<p>5.4 Are software tools used for tracking and reporting the status of the software/subroutines in the software development library? Comments:</p>	(YES/NO)
<p>5.5 Are prototyping methods used in ensuring the requirements elements of the software? Comments:</p>	(YES/NO)
<p>(5.6) Is a data dictionary available for controlling and storing details of all data files and their fields? Applies to: Where systems developed have significant file/database content. Comments:</p>	(YES/NO/N.A.)
<p>5.7 Are software tools used for project planning, estimating, scheduling, and critical path analysis? Comments:</p>	(YES/NO)

A.3 Investigación sobre las prácticas de ingeniería de software en México

María de la luz Villalobos Hernández. ESCA-IPN
Agustín Francisco Gutiérrez Tornés CIC-IPN

Sección I. Antecedentes

Instrucción: Marque con una X la opción seleccionada.

1. ¿Cuál de las siguientes opciones describe su posición actual (puede marcar varias)

- Líder de proyecto o de equipo
- Consultor
- Programador/analista
- Admón redes
- Soporte técnico
- Otro _____

2. ¿En cuál de las siguientes actividades se desempeña (puede marcar varias)?

- Requerimientos de software
- Aseguramiento de calidad de software
- Diseño de software
- Administración de la configuración
- Código y pruebas de unidades
- Mejoras del proceso de software
- Pruebas e integración
- otro _____

3. ¿Cuál es el giro de su empresa?

- Sistemas multimedia y web.
- Sistemas de información (sistemas de gestión, SMBD, SMBDOO, etc.).
- Sistemas distribuidos.
- Sistemas de tiempo real.
- Sistemas de ingeniería(CAD/CAM) y científicos.
- Sistemas de control (Sistemas de adquisición de datos).
- Sistemas de Inteligencia Artificial(sistemas expertos, etc.).
- Sistemas de geoprocusamiento
- Asesoría y consultoría.
- Capacitación.
- Docencia e investigación.
- otro _____

4. Experiencia laboral:

¿Cuánto tiempo lleva en la present software?
 ____ años

5. ¿De cuántas personas se compone el área de desarrollo de software en su empresa? ()
 a)1-5 b)6-10 c)11-15 d)16-21 e)21 o más

6.-¿El tiempo promedio de duración de sus proyectos es de...? ()
 a)1 mes b)2-4 meses c)5-7 meses d)8-12 meses e)más de 12 mes

Sección 2 Gestión de proyectos de Software

Instrucción: Elija la opción de acuerdo a los criterios siguientes:

- Si- La práctica está bien establecida y se ejecuta consistentemente como un procedimiento estándar de operación.
- No- La práctica no está bien establecida o es realizada inconsistentemente.
- No aplica – Posee el conocimiento requerido acerca del proyecto u organización pero usted siente que la pregunta no aplica para el proyecto.
- Desconoce – Ud. posee incertidumbre de cómo responder a la pregunta.

Utilice los espacios para comentarios para cualquier aclaración acerca de sus respuestas a las preguntas.

1. ¿Para elaborar un proyecto se sigue algún plan documentado que guíe el desarrollo?

Sí No No aplica Desconoce

2. ¿Dentro de los proyectos, quienes son las personas encargadas de colaborar con el cliente para establecer el sistema de requerimientos?

- Líder de proyecto
- Líder comercial
- Líder de proyecto y comercial
- Otro _____

3. ¿Al inicio del proyecto se revisan las nuevas tecnologías existentes en el mercado para determinar la más adecuada a las necesidades del cliente?

Sí No No aplica Desconoce

Comentarios:

4. ¿En cuanto a la distribución de esfuerzos, que porcentaje le asigna a cada fase (el total debe ser 100%)?

- Análisis y diseño
- Codificación
- Prueba y depuración

5. ¿La capacitación proporcionada es la necesaria para desarrollar las bases y el conocimiento requerido para desempeñar sus actividades?

Sí No No aplica Desconoce

6. ¿Mantiene informados a todos los miembros del equipo acerca de los cambios del proyecto(ej.: aquellos que realizaron el trabajo y aquellos que son los responsables del trabajo)?

Sí No No aplica Desconoce

7. ¿Los resultados de los proyectos actuales se comparan con las estimaciones iniciales de los planes de software en términos de... ? (puede marcar varias)

- Tamaño del proyecto
- Costos
- Duración cronológica del proyecto
- Recursos humanos
- Tecnología

8. ¿Cuáles de las siguientes actividades realiza para mantener un control del proyecto? (puede marcar varias)

- reuniones periódicas sobre el estado del proyecto
- evaluar los resultados de todas las revisiones realizadas en todo el proceso
- identificar si las tareas se han alcanzado en las fechas programadas
- reuniones informales con los miembros para conocer sus valoraciones subjetivas

9. ¿Para realizar las pruebas del sistema, cuáles de los siguientes pasos realiza?

- Establece objetivos de las pruebas
- Diseña los casos de prueba
- Prueba los casos
- Evalúa los resultados de las pruebas

10. ¿Qué tipo de pruebas aplica a un proyecto?

- pruebas que demuestren la función es completamente operativa
- pruebas que aseguren que las operaciones internas se ajustan a las especificaciones
- otra _____

11. ¿La empresa contrata personal exclusivo para las pruebas del sistema?

- Sí No No aplica Desconoce

12. ¿Cuál de los siguientes modelos de gestión y aseguramiento aplica?

- ISO 9001 e ISO 9000-3
- ISO SPICE
- ISO Tick-IT
- Capability Maturity Model (CMM)
- otro _____
- ninguno

Instrucción. En las siguientes tres cuestiones seleccione de la lista de valores anotando el número que crea conveniente para cada opción. **(1-Bajo 2- Medio 3-Alto, 4- Muy Alto)**

13. De acuerdo a su nivel de importancia los siguientes factores de calidad que describen a un software son:

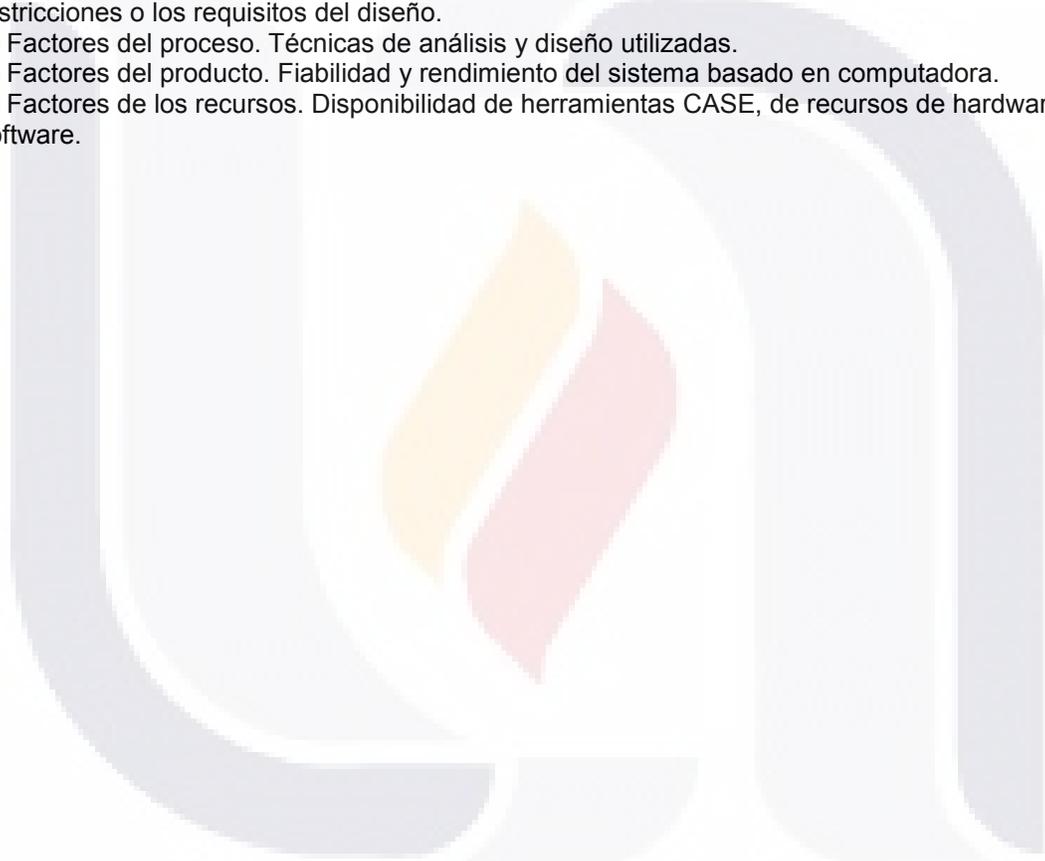
- Confiabilidad (Que no falle, que no tenga errores)
- Eficiencia (El grado en que se aprovechan los recursos a disposición del producto de software)
- Funcionalidad (Se refiere al grado en que el software satisface los requerimientos, hace lo esperado)
- Mantenibilidad (¿Puedo actualizarlo?)
- Portabilidad (¿Podré usarlo en otro ambiente computacional?)
- Reusabilidad (¿Puede una parte o todo utilizarse en otro software?)
- Usabilidad (Se refiere a la facilidad de uso.)

14. De las siguientes razones, anote el número que a su criterio le corresponda a cada opción que señale los motivos por los cuales se debe medir el software.

- () Indicar la calidad del producto
- () Evaluar la productividad de la gente que desarrolla el producto
- () Evaluar el uso de nuevos métodos y herramientas de ingeniería de software
- () Establecer una línea base para futuras estimaciones

15. De acuerdo a su experiencia anote el número que corresponda a los factores que inciden en la productividad del software

- () Factores humanos. La cantidad y la experiencia del personal de desarrollo.
- () Factores del problema. La complejidad del problema a resolver y el número de cambios en las restricciones o los requisitos del diseño.
- () Factores del proceso. Técnicas de análisis y diseño utilizadas.
- () Factores del producto. Fiabilidad y rendimiento del sistema basado en computadora.
- () Factores de los recursos. Disponibilidad de herramientas CASE, de recursos de hardware y de software.



A.4 Software Engineering Practices in Singapore

Software Engineering Practices in Singapore
 Danny C.C. Poo and Mui Ken Chung
 Department of Information Systems
 School of Computing
 National University of Singapore
 3, Science Drive 2
 Singapore 117543

Instrumento de Evaluación

Sección 1

Datos demográficos

Sección 2

Prácticas de Ingeniería de Software

Metodología de desarrollo de software

1. ¿La organización utiliza una metodología formal para el desarrollo de sus productos?
(Si/No)
2. ¿Cuánto tiempo se ha utilizado la metodología de desarrollo?
(Menos de un año/1 -2 Años/3 – 4 Años/ Más de 4 años)
3. ¿Qué modelo de proceso de software sigue la metodología?
(Secuencial (Modelo en Cascada)/Espiral (Cascada + Análisis de Riesgo)/Incremental/Prototipado/Otro)

Técnicas de Ingeniería de Software y Herramientas CASE

1. ¿Técnicas Utilizadas en la fase de análisis de requerimientos?
(Diagramas de flujo de datos, Diccionarios de datos, Diagramas entidad relación, diagramas de flujo, normalización de datos, Prototipado, diagramas de descomposición de procesos, diagramas de dependencia entre procesos, desarrollo de aplicaciones conjunto(JAD), diagrama de ciclo de vida, análisis orientado a objetos)
2. ¿Técnicas utilizadas en la fase de Diseño del sistema?
Diseño de interfaces, Diseño de reportes, Organigrama, Tablas ó arboles de decisiones, Diagramas de jerarquía de procesos, Diagramas jerárquicos de entradas y salidas, análisis del volumen de transacciones, diagramas de acción,

diagramas del flujo de dialogo, diseño orientado a objetos (Booch Diagrams), Diagramas Warnier-Orr, Diagramas Nassi-Schneiderman.

3.- ¿Técnicas utilizadas en la etapa de de codificación?
Especificaciones del programa, 4GL, 3GL

4.- ¿Técnicas utilizadas en la etapa de pruebas?
Walkthroughs, caja blanca, pruebas de integración, pruebas de caja negra.

Mantenimiento de Software

¿Técnicas utilizadas en la etapa de mantenimiento?
Procedimiento de administración de cambios de software
Proceso de administración de cambios en la documentación
Procedimiento de la administración de la configuración

Entrenamiento y Educación en Ingeniería de Software

¿La organización provee entrenamiento formal en las metodologías de ingeniería de software?
(Si/No)

¿Tipo de entrenamiento que se proporciona en la organización?
Formal (llevado a cabo por entrenadores externos/Llevado a cabo por entrenadores de la organización)
Informal (Durante el trabajo/autodidáctico(videos, documentos, etc)

¿Frecuencia con que ocurren las siguientes situaciones?

- El software implementado cumple con un alto grado de satisfacción del usuario
- El software desarrollado es fácil de mantener
- Los proyectos de software con completados dentro de los costos estimados
- Los proyectos de software son complementados dentro de los tiempos definidos

(Nunca/algunas veces/usualmente/siempre)

Sección 3

Percepción de prácticas de Ingeniería de Software

¿Las mejores prácticas de ingeniería de software permiten un mejor control sobre el desarrollo y mantenimiento de software?

¿Las mejores prácticas de ingeniería de software mejoran la calidad de los sistemas desarrollados y en mantenimiento?

¿Las mejores prácticas de ingeniería de software mejoran el mantenimiento y la adaptabilidad del software?

¿Las mejores prácticas de ingeniería de software reducen el tiempo de mantenimiento y desarrollo?

¿Las mejores prácticas de ingeniería de software han ayudado a la organización en mejorar su ventaja competitiva?

¿La metodología de trabajo adoptada en la organización es fácil de usar?

¿La metodología de ingeniería de software orientada a objetos ayudara de manera benéfica a la organización?

¿Las mejores prácticas de ingeniería de software ayudan a reducir los costos?
(De acuerdo/Neutral/Desacuerdo)

¿Falta de personal experimentado en la utilización de herramientas y metodologías de ingeniería de software?

¿Falta de entrenamiento formal en metodologías y herramientas de ingeniería de software?

¿Falta de un apropiado ambiente y herramientas que soporten las metodologías de ingeniería de software?

¿Sistemas heredados que no son compatibles con las nuevas metodologías de ingeniería de software?

¿Complejidad de las mejores prácticas de ingeniería de software?

¿Falta de soporte por parte de la dirección?

¿Curva de aprendizaje prolongada?

¿Inversión financiera muy alta?

¿Objetivos no claros para la adopción de las mejores prácticas de ingeniería de software?

¿Beneficios inciertos acerca de la adopción de mejores prácticas de ingeniería de software?

¿Fracasos previos en intentos de implementar mejores prácticas de ingeniería de software?

¿Metodología definida es inapropiada para los proyectos actuales?

A.5 Instrumento de Evaluación



UNIVERSIDAD AUTÓNOMA
DE AGUASCALIENTES

Maestría en Ciencias Exactas, Sistemas y de la
Información. Área: Ingeniería de Software.

Sección I. Datos Demográficos

1.- ¿Número de empleados con los que cuenta la organización?

- Menos de 10 personas
- 10 a 20 personas
- 21 a 30 personas
- 31 a 40 personas
- Más de 40 personas

2.- ¿Número de empleados involucrados en la Ingeniería de Software?

- Menos de 10 personas
- 10 a 20 personas
- 21 a 30 personas
- 31 a 40 personas
- Más de 40 personas

3.- ¿Tiempo de existencia de la organización?

- Menos de 2 años
- De 2 a 4 años
- De 5 a 7 años
- De 8 a 10 años
- Más de 10 años

4.- ¿La organización actualmente tiene implementado algún modelo de trabajo de mejores prácticas? (CMM, CMMI, MoProSoft.)

- Sí
- No

Nota: Sólo conteste esta pregunta si su respuesta anterior fue No.

5.- ¿La organización a corto plazo (Menor a 2 años) tiene formalmente planeado implementar un modelo de trabajo de mejores prácticas? (CMM, CMMI, MoProSoft)

- Sí
- No

Instrucciones

1. Por favor, responda a TODAS las preguntas con Si o No.
2. En caso de "no entender la pregunta", esta deberá ser contestada con No.
3. Por favor, responda de acuerdo a las prácticas normales de la organización.

Sección II. Administración de Proyectos de Software

1.- ¿Para cada proyecto de software se tiene asignado un Líder de Proyecto?

- Sí
 No

2.- ¿Antes de comenzar el desarrollo del producto se realizan estimaciones sobre el costo y tiempo que requerirá su producción?

- Sí
 No

3.- ¿Existe un programa de entrenamiento formal para los nuevos elementos del proyecto, que este diseñado para familiarizarlos con los procedimientos de la organización?

- Sí
 No

4.- ¿Se realiza una planeación documentada formalmente de los tiempos en que se va a llevar el desarrollo de producto con la finalidad de monitorear el avance del proyecto? (Graficas de Gantt, Diagramas de Hitos, PERT)

- Sí
 No

5.- ¿Se realizan revisiones periódicas formales para verificar el estado de cada proyecto de software y mantener informados a los miembros del proyecto?

- Sí
 No

6.- ¿Se tienen definidas métricas que midan el desempeño del desarrollo del producto?

- Sí
 No

Sección III. Análisis de Software

1.- ¿Se evalúa formalmente los beneficios y la viabilidad del desarrollo del proyecto de software antes de realizar un contrato con los compromisos?

- Sí
 No

2.- ¿Dentro de la Organización se cuenta con un documento estandarizado para la obtención de requerimientos? (IEEE, Definido por la organización)

- Sí
 No

3.- ¿Se cuenta con un proceso formal para el cambio de un requerimiento previamente definido?

- Sí
 No

4.- ¿Se define un ciclo de desarrollo en base a las características del producto que se va a producir? (Espiral, Cascada, Evolutivo, etc.)

- Sí
 No

5.- ¿Se modelan los requerimientos en base a notaciones formales? (Diagramas de Casos de Uso, SADT)

- Sí
 No

6.- ¿Antes de iniciar el proyecto, se revisan las nuevas tecnologías existentes en el mercado para determinar la más adecuada a las necesidades del cliente?

- Sí
 No

Sección IV. Diseño de Software

1.- ¿Son utilizados los diseños de interfaces y reportes tomando en cuenta a los usuarios finales?

- Sí
 No

2.- ¿Se lleva a cabo el proceso de normalización de base de datos para generar un diagrama Entidad-Relación óptimo para el manejo de los datos? (1FN, 2FN, 3FN, 4FN, 5FN)

- Sí
 No

3.- ¿Se generan diccionarios de datos para el control y almacenaje detallado de los datos del sistema?

- Sí
 No

4.- ¿Son utilizadas notaciones formales para realizar el diseño de los sistemas? (UML, SADT)

- Sí
 No

5.- ¿Se toman en cuenta los patrones de arquitectura y diseño para el modelado del sistema? (Modelo Vista Controlador (MVC), Patrones Gang Of Four)

- Sí
 No

6.- ¿Se utilizan herramientas CASE para realizar el diseño del producto? (starUML, MySQL Migration Toolkit, Microsoft Visio, etc.)

- Sí
 No

7.- ¿Al realizar el diseño del sistema se busca que haya bajo acoplamiento y alta cohesión?

- Sí
 No

Sección V. Codificación de Software

1.- ¿Para cada proyecto se tiene establecido un sistema de control de versiones y cambios para controlar el desarrollo del software? (CVS, Subversion, SourceSafe, ClearCase, Darcs, Bazaar, Plastic SCM, Git, Mercurial, etc.)

- Sí
 No

2.- ¿Son utilizadas técnicas de prototipados (Evolutivos o Desechables) para generar un producto que cumpla con las expectativas del cliente?

- Sí
 No

3.- ¿Se tiene establecido un ambiente de desarrollo para realizar la integración de las nuevas funcionalidades, antes de integrarlas a la versión estable o final? (Comúnmente conocidas como Etapa, Desarrollo y Producción.)

- Sí
 No

4.- ¿Se cuenta con un formato específico que lleve el control del código fuente desarrollado? (Comúnmente con la siguiente información: Responsable, Fechas de creación y modificación, estado, etc.)

- Sí
 No

5.- ¿Se cuenta en la organización un formato específico para realizar la documentación técnica del código fuente desarrollado?

- Sí
 No

6.- ¿Se tienen especificados estándares para la escritura de código fuente? Comúnmente conocidas como convenciones, (UpperCamelCase, lowerCamelCase, Beans, etc.)

- Sí
 No

Sección VI. Pruebas de Software

1.- ¿Se realiza un plan de pruebas antes de comenzar el desarrollo del producto?

- Sí
 No

2.- ¿Las pruebas de software son realizadas por un grupo totalmente independiente al equipo de desarrollo?

- Sí
 No

3.- ¿Se tiene establecido que se realicen revisiones de pares a los requerimientos desarrollados? (Comúnmente conocidos como Peer Reviews)

- Sí
 No

4.- ¿Se realizan pruebas del sistema con los usuarios finales en el ambiente deseado por el cliente antes de su implementación final?

- Sí
 No

5.- ¿Se cuenta con un formato establecido para la definición y llenado de los casos de prueba? (Comúnmente conocidos como Test Cases)

- Sí
 No

6.- ¿Son utilizadas herramientas automatizadas de pruebas? (JUnit, XQual Studio, ComfortTester, vTest)

- Sí
 No

A.6 Listado de Empresas

Empresa	Dirección	Teléfono	Sitio Web
Agsoft	Felipe Serrano 361, Nuestra señora de la asunción	1392277	www.agsoft.com.mx
BitMore	Av. Universidad #1005 Int. 5	1533619	www.bitmoretechnologies.com
Centro de Solución Empresarial	Héroe de Nacozari Sur 2420 Int. 1, Rinconada del Parque	9782885	www.cse.com.mx
DaComp	Roma 207, Fraccionamiento del Valle	9154861	www.dacompsc.com
AVG Asociados	República Perú 604-312, Fracc. Santa Elena	9785055	www.avgasociados.com.mx
Eb-Soluciones	Mar Caspio 631, Fraccionamiento Colinas del Río	9964746	www.eb-soluciones.com
EIT	Manuel J. Clouthier 2275-5	1221239	www.eit.com.mx
Grupo Inco	Av. Aguascalientes Sur No. 2625, Jardines de las Fuentes	9781408	www.grupoinco.net
Innova Consultoría	Av. Universidad 1001, Bosques del Prado Norte Piso 6. Int. 2	9966260	www.innovaconsultoria.com.mx
IT Builder	Ed. Torre Plaza Bosques Piso 11 Int. 7	1532636	www.itbuilder.com.mx
KaelSoft	Canal interceptor 408	1533179	www.kaelsoft.com
NTS	Av. Aguascalientes Norte 503-F Fracc. Bosques del Prado Sur	1533096	www.newts.com.mx
Chiva Sentada	Sin Dirección	1392277	www.chivasentada.com
PyMSis	Clarisas # 344 Fracc. Villa Teresa	9257884	www.pymsis.com.mx
QuarkSoft	Av. Ciencia y Tecnología Sin Numero Parque Tecnopolo	9650020	www.quarksoft.net
Grupo MX	Av. Universidad #1009 Int. 304 Bosques del Prado Norte	1534111	www.grupomx.com
Traffic System	Valparaíso 207 Col. La Fuente	9169857	www.trafficsystem.com.mx
Desarrolla	Ciencia y Tecnología 102, Parque Tecnopolo	1533090	www.desarrolla.com.mx
FacturaNet	Av. Independencia #1926-3, Fracc. Trojes de Oriente	9966763	www.facturanet.net
IntecPlan	Parque Industrial Tecnopolo Pocitos L-4	9784115	www.intecplan.com.mx
Sprocket	Sin Dirección	9120038	www.sprocket.com.mx
Softtek	Av. Eugenio Garza Sada #902 Parque Industrial Tecnopolo	9107800	www.softtek.com
Hildebrando	Sierra de Las Palomas 101 Bosques del Prado Sur	9961500	www.hildebrando.com.mx
Dextra	Jesús Rivera Franco #507 Ciudad Industrial	9967541	www.dextratech.com