



Universidad Nacional Autónoma de México

Programa de Posgrado en Ciencias de la Administración

“Las herramientas open source y de software libre más usadas en la industria desarrolladora de software dentro del Distrito Federal”

T e s i s

Que para optar por el grado de:

Maestro en Administración Campo de Conocimiento: Administración de la Tecnología.

Presenta:

Marco Antonio Galíndez Cortés

Tutor:

M. A. Omar Barragán Fernández

Facultad de Contaduría y Administración

México, D. F., Junio de 2013.



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

“Las herramientas *open source* y de *software* libre más usadas en la industria desarrolladora de *software* dentro del Distrito Federal”

Dedicatoria

A mis padres Rufina Cortés y Ezequiel Galíndez por su apoyo incondicional y nunca dejar de apoyar mis sueños.

A mi hermana Elizabeth Por su gran lección como profesional historiadora, mujer y madre responsable, ya que desde su ámbito de competencia ha creído en mí y siempre ha encontrado el mejor rumbo en todos los sentidos y diferentes fases de su vida.

A mi hermano Miguel Porque sin saberlo, ha contribuido decisivamente en el logro de este trabajo al ser una persona que tiene valores, y demostrar que siempre es mejor cerrar los ciclos que tiene la vida.

A mis tíos José, Joaquín, Genaro, Elena, Consuelo, Lourdes, Guadalupe y Abraham porque desde sus muy diferentes puntos de vista y maneras de comunicar las cosas, nunca han dejado de creer en mí, tenerme una paciencia casi infinita y enseñarme toda la vida con sus ejemplos los valores más importantes.

Agradecimientos:

A mi tutor y sinodal Mtro. Omar Barragán Fernández por mostrarme una dedicación y paciencia muy superior a la que su obligación demandaría.

Al doctor Adrián Mendez Salvatorio y la maestra María Angélica Raya Sánchez, por los momentos que pudieron compartir conmigo.

A mis profesores en la maestría, Dr. Francisco Ballina Ríos, mtra. Celia Luz González Fernández, Dra. María de Lourdes Marquina Sánchez, mtra Rita Aurora Fabregat Tinajero, mtra Patricia Guadalupe Olea Salas, Dra Alejandra Herrera, Dr. Carlos Eduardo Purga Murguía, Dr. Roberto E. López Martínez, Dr. Enrique Medellín Cabrera, mtro Alejandro Zárate Paredes, Dr. Raúl Ojeda Villagomez, por hacer de esta estancia la más fortalecedora en mis conocimientos académicos.

A las personas que tuve el privilegio de conocer o reencontrar en esta etapa, y espero poder mantener su confianza para el resto de mi vida, Antonio Cabrera, Anaid Guevara, Andrei Perez, Gustavo Becerril, Naxhieli Maldonado, David Vázquez, Fernando Gamboa, Itzelsweet Suarez, Jose Antonio, Mario y Yesica, Francisco (paco), Álvaro, Emilio, Omar... Omar, Claudette (Alexa), Guillermo, Mauricio, Ulises Martinez, Soledad (Sol) González, Erika Martínez, Paty Huacuja, Angélica Hernández, Samira Macias, Guillermo Ruíz, Jocelyn Gonzalez, Rosa Emilia, Yadira Mondragón, Yolanda Tellez, Verónica Gonzalez, Israel Hernández, Rodolfo López. A mis demás amigos, compañeros y conocidos, Gracias por todo...

A la UNAM, por darme la oportunidad de estudiar esta maestría y permitirme conocer todo el mundo de opciones que ofrece para aquel con verdadera hambre de conocimiento.

A todas las personas que me apoyaron de cualquier manera para la finalización de este estudio, gracias a todos.

Tabla de contenido

INTRODUCCIÓN	1
CAPITULO 1	¡Error! Marcador no definido.
Metodología	4
Justificación:	4
Preguntas de investigación.....	8
Objetivos	9
Hipótesis	9
Matriz de congruencia	9
Relevancia social de la investigación	10
¿Quién se beneficiará con los resultados de esta investigación?	10
¿De qué modo?	10
¿Qué proyección social tiene?	10
Implicaciones prácticas	10
¿Ayudará a resolver algún problema?.....	10
¿Qué tan conveniente es la investigación?	11
Metodología empleada	11
Diseño de la investigación.	11
Elaboración del instrumento.	12
Redefiniciones fundamentales:	12
Unidades de análisis.....	13
Construcción del instrumento.	14
Confiabilidad y Validez.....	21
CAPITULO 2	23
INTRODUCCIÓN	23
¿QUÉ ES EL SOFTWARE?	23
Software.	23
DIFERENTES TIPOS DE SOFTWARE	25
<i>Software</i> Propietario.....	25
Freeware.....	26
Shareware.....	26
Warez.....	26
<i>Software</i> Semilibre.....	26
Software libre.....	27
<i>Open source</i> (código abierto).....	29
DIFERENCIAS DE SOFTWARE LIBRE Y DE CÓDIGO ABIERTO	30
Ventajas del <i>software</i> libre y del <i>software</i> de código abierto.	31
Desventajas del software libre y de código abierto.....	38
Ventajas del software de código cerrado.....	41
Desventajas de los programas de código cerrado.....	42
CONCLUSIONES	43
CAPÍTULO 3	45
INTRODUCCIÓN:	45
¿Qué es una empresa?	45
Propósito de la empresa	47
Características de la empresa.	47

El emprendedor.....	47
Clasificación de las empresas.....	48
Pequeñas y medianas empresas.....	50
La pyme en el mundo.....	51
Las pymes en México.....	53
Penetración del internet.....	56
La industria desarrolladora de software en el mundo.....	58
La industria del software en México.....	61
EL PROGRAMA PROSOFT.....	61
CONCLUSIONES.....	63
CAPÍTULO 4: Competitividad en la industria del <i>software</i>.....	65
Introducción.....	65
Competitividad.....	65
Competitividad en país.....	66
Competitividad en la industria.....	67
Competitividad en empresa.....	68
Elementos de medición de competencia del software.....	69
Modelos de evaluación de madurez de procesos de desarrollo de software.....	72
Ingeniería de <i>software</i>	73
Modelos de proceso de software.....	74
Modelos de evaluación de <i>software</i>	75
INDICADOR DE COMPETITIVIDAD DE SOFTWARE.....	84
CONCLUSIONES.....	85
CAPÍTULO 5.....	87
Introducción.....	87
Resultados:.....	88
Datos del estado de la empresa.....	90
Preguntas con respecto a las herramientas informáticas.....	94
Análisis de resultados.....	107
Resultado.....	113
GLOSARIO:.....	116
Bibliografía.....	119

ÍNDICE DE TABLAS

TABLA 2.1: TABLA DONDE NOS MUESTRA QUE DE LAS 500 COMPUTADORAS MAS VELOCES, 455 TRABAJAN CON GNU/LINUX Y SOLAMENTE 6 CON EL SISTEMA OPERATIVO MICROSOFT WINDOWS.	38
TABLA 3.1 CLASIFICACIÓN DE LAS EMPRESAS DE ACUERDO CON LA OCDE.	49
TABLA 3.2 CLASIFICACIONES DE LAS EMPRESAS PROPUESTA POR LA CCE.	49
TABLA 3.3 CLASIFICACIÓN DE LA MICRO, PEQUEÑA Y MEDIANA EMPRESA VIGENTES A PARTIR DEL 30 DE JUNIO DEL 2009 (BARRAGÁN, 2011) EN MÉXICO.	50
TABLA 3.4 PORCENTAJE DE EMPRESAS POR TAMAÑO EN EL SECTOR DE MANUFACTURA.	51
TABLA 3.5 CONFLICTOS QUE AQUEJAN A LA PYME MEXICANA.	54
TABLA 3.6 MEDIDA EN QUE LAS EMPRESAS ABSORBEN NUEVAS TECNOLOGÍAS, LA CALIFICACIÓN MÁXIMA ES DE 7.	59
TABLA 4.1 FACTORES DE COMPETITIVIDAD NACIONAL.	67
TABLA 4.2 COMPONENTES CON LOS QUE DEBE DE CONTAR LOS PROCESOS DE CREACIÓN DE <i>SOFTWARE</i>	70
TABLA 4.3 ATRIBUTOS ESENCIALES DEL BUEN <i>SOFTWARE</i>	71
TABLA 4.4 RETOS QUE ENFRENTA EL <i>SOFTWARE</i>	71
TABLA 4.5 REPRESENTACIÓN DE LOS DIFERENTES NIVELES QUE POSEE EL MODELOS CMMI.	78
TABLA 4.6 PROCESOS POR LOS QUE ESTA CONFORMADO MOPROSOFT.	82
TABLA 5.1 CUÁLES SON LAS HERRAMIENTAS DE CÓDIGO ABIERTO O LIBRE QUE UTILIZA MÁS FRECUENTEMENTE.	108
TABLA 5.2 LISTE POR FAVOR LAS HERRAMIENTAS DE TIPO OPEN SOURCE QUE CONCIDERE SON TAN COMPETITIVAS COMO LAS DE CÓDIGO CERRADO.	109
TABLA 5.3 TABLA DE LAS HERRAMIENTAS OPEN SOURCE QUE AÚN NO ESTÁN LO SUFICIENTE DESARROLLADAS PARA TRABAJAR CON ELLAS.	110

ÍNDICE DE GRÁFICAS

GRÁFICA 2.1.- NOS MUESTRA QUE APROXIMADAMENTE EL 91% DE LAS COMPUTADORAS EN LOS ESTADOS UNIDOS HACEN USO DEL SISTEMA OPERATIVO MICROSOFT WINDOWS.	40
GRÁFICA 3.1 USUARIOS DE INTERNET 1995-2010.....	57
GRÁFICA 3.2: CONEXIONES A INTERNET SUPERIORES A 5 MEGAS/S.....	57
GRÁFICA 3.3: COMPETITIVIDAD VS NIVEL DE PENETRACIÓN DE BANDA ANCHA POR PAÍS (2010).	58
GRÁFICA 3.4: GASTOS GLOBALES DE TIC, LAS CANTIDADES ESTÁN EN BILLONES DE DÓLARES ESTADUNIDENSES.....	59
GRÁFICA 4.1 REPRESENTACIÓN GRÁFICA DE LAS CAPAS POR LAS QUE ESTA CONSTITUIDA LA INGENIERÍA DE <i>SOFTWARE</i>	73
GRÁFICA 4.2: ETAPAS DE CICLO DE DESARROLLO.....	82
GRÁFICA 5.1: EDADES DE LOS ENCUESTADOS.....	90
GRÁFICA 5.2 NIVEL DE ESTUDIOS.....	91
GRÁFICA 5.3: PUESTO DENTRO DE LA EMPRESA.....	92
GRÁFICA 5.4: NÚMERO DE EMPLEADOS.....	93
GRÁFICA 5.5: CONEXIÓN A INTERNET.....	93
GRÁFICA 5.6: NORMA ISO 9126 DE MANERA GRÁFICA. GRÁFICA TOMADA DE (ALVAREZ ARRIAGA, 2011).	94
GRÁFICA 5.8: BRINDA LOS RESULTADOS CORRECTOS.....	95
GRÁFICA 5.9: PUEDE INTERACTUAR CON OTRO <i>SOFTWARE</i>	96
GRÁFICA 5.10: SEGURIDAD DE ACCESO.....	96
GRÁFICA 5.11: EVITA FALLOS POR MALA PROGRAMACIÓN.....	97
GRÁFICA 5.12: EVITA FALLOS POR MALA PROGRAMACIÓN.....	97
GRÁFICA 5.13: CAPACIDAD DE RECUPERAR DATOS EN CASO DE FALLO.....	98
GRÁFICA 5.14: CUENTA CON LA DOCUMENTACIÓN SUFICIENTE.....	99
GRÁFICA 5.15: ES INTUITIVO EL <i>SOFTWARE</i>	99
GRÁFICA 5.16: ES FÁCIL DE UTILIZAR.....	100
GRÁFICA 5.17: ES ATRACTIVO VISUALMENTE.....	100
GRÁFICA 5.18: PROPORCIONA TIEMPOS DE RESPUESTA.....	101
GRÁFICA 5.19: UTILIZA SOLO LOS RECURSOS NECESÁRIOS.....	102
GRÁFICA 5.20:ES FÁCIL DETERMINAR FALLOS EN EL <i>SOFTWARE</i>	102
GRÁFICA 5.21:ES FÁCIL IMPLEMENTAR ALGUNA MODIFICACIÓN.....	103
GRÁFICA 5.22:EVITA EFECTOS INESPERADO POR APLICAR MODIFICACIONES.....	103
GRÁFICA 5.23: EN CASO DE SER MODIFICADO PUEDE SER VALIDADO.....	104
GRÁFICA 5.24: PUEDE SER ADAPTADO A DIFERENTES SO.....	105
GRÁFICA 5.25: ES FÁCIL DE INSTALAR.....	105
GRÁFICA 5.26: EL <i>SOFTWARE</i> PUEDE COEXISTIR CON EL <i>SOFTWARE</i> O ENTRA EN CONFLICTOS.....	106
GRÁFICA 5.27: EL <i>SOFTWARE</i> PUEDE SER UTILIZADO EN LUGAR DE OTRO PROGRAMA EN LA MISMA COMPUTADORA.....	106
GRÁFICA 5.28: RESPUESTAS A LA PREGUNTA, ¿CUÁLES SON LAS HERRAMIENTAS DE CÓDIGO ABIERTO O LIBRE QUE UTILIZA MÁS FRECUENTEMENTE?.....	108
GRÁFICA 5.29: RESPUESTAS A LA PREGUNTA, LISTE POR FAVOR LAS HERRAMIENTAS DE TIPO <i>OPEN SOURCE</i> QUE CONSIDERE SON TAN COMPETITIVAS COMO LAS DE CÓDIGO CERRADO.....	109
GRÁFICA 5.30: RESPUESTAS A LA PREGUNTA, LISTE POR FAVOR LAS HERRAMIENTAS <i>OPEN SOURCE</i> QUE CONSIDERA QUE AÚN NO SON LO SUFICIENTE ESTABLES PARA TOMARLAS COMO PRIMERA OPCIÓN.....	110
GRÁFICA 5.31: RESPUESTAS A LA PREGUNTA, ¿CONSIDERA QUE SU PRODUCTIVIDAD ES LA MISMA CUANDO TRABAJA CON HERRAMIENTAS DE CÓDIGO ABIERTO O LIBRES?.....	111

INTRODUCCIÓN

Poco a poco las computadoras se están posicionando como una herramienta casi imprescindible para todos, la penetración de los equipos de cómputo según datos de la OECD para el año del 2009 es de más del 26.8%[1], cada día se están vendiendo más equipos de cómputo alrededor del planeta, y por lo tanto también el *software* con el que cuentan, ya que una computadora por sí sola no tiene mayor utilidad si es que no se tienen todos los programas informáticos requeridos, y para crear estos programas es necesario contar con personas con el conocimiento para desarrollarlos, por esto mismo han surgido innumerables compañías alrededor del mundo que tienen como finalidad crear *software* de calidad.

A pesar que existen muchas empresas que se dedican a la creación de *software* realmente el mercado esta ampliamente concentrado en unas cuantas compañías internacionales, como por ejemplo Microsoft, con su producto estrella Microsoft Windows, Oracle que es la segunda empresa más grande desarrolladora de software, IBM, HP, entre de otras; todas estas empresas son gigantes tecnológicos, que cuentan con grandes recursos económicos y humanos para llevar a cabo cualquier proyecto, como por ejemplo la creación de una nueva herramienta informática y garantizar que por determinado tiempo esta contará con el servicio necesario para garantizar su buen funcionamiento.

El objetivo principal de este estudio es el dejar en claro que las empresas desarrolladoras de software ubicadas dentro del Distrito Federal ya están utilizando estas herramientas, pero sobre todo conocer cuales son las que se están utilizando más porque los resultados obtenidos son iguales o superiores a su contraparte de código cerrado.

Igualmente importante con este estudio será el conocer que herramientas de código cerrado aún son superiores a las opciones libres o de código abierto, esto mostrará un espectro claro para cualquier persona que desee emprender un negocio de desarrollo de software o que ya tenga o esté en una empresa de este ramo, este estudio le dará el conocimiento de las mejores opciones que puede adquirir para su organización.

Para poder alcanzar esta meta, fue necesario el dividir en diferentes partes esta investigación, el primer capítulo nos explica claramente todo el marco metodológico de la misma, las preguntas de investigación, el objetivo, la síntesis y el instrumento que se creó para poder realizar las encuestas que se aplicaron a las diferentes organizaciones a lo largo de la misma. En este capítulo principalmente se dejó en claro qué es lo que se investigaría, porque, y de qué manera se realizará este trabajo.

El capítulo dos se explican en claro la mayoría de los términos que se utilizarán a lo largo de la misma, como por ejemplo que es el *software* libre, el de

código abierto, porque en todo este trabajo se haría uso indistinto entre estos dos términos, aclarar que el *software* libre no tiene que ser gratuito, que los programas gratuitos no tienen que ser considerados como libres automáticamente, las licencias principales que existen dentro del *software* libre para proteger los programas que existe, entre otros términos.

El capítulo tres es un estudio de las pymes (pequeñas y medianas empresas), tanto a nivel internacional como nacional, aquí observamos que las organizaciones no son un sujeto de estudio sencillo a cualquier nivel de comparación, ya sea internacional como por ejemplo comparar las pequeñas organizaciones entre países, ya que no existe un modelo homogéneo entre de estos (excepto el de la unión europea), ya que el medio ambiente en el que se desenvuelven son muy diferentes de un país a otro, o completamente nacional, como comparar las empresas de Sonora con otro en Oaxaca, ya que las realidades que se viven pueden ser muy diferentes. Se explica en este capítulo cuales tipos de organización serían las estudiadas, y se investiga de manera general la realidad a la que se están enfrentando las pequeñas empresas desarrolladoras de *software* en este momento.

El capítulo cuatro se trata de cómo medir la competencia dentro de las empresas desarrolladoras de *software*, ya que no existe ninguna obligación para que una persona que esté desarrollando un código tenga que hacer esta tarea de una manera establecida, existen algunas reglas que han sido probadas ampliamente que el seguirlas trae beneficios, estas reglas se han transformado en certificaciones como por ejemplo CMMI creada por el ejército de los Estados Unidos y la universidad de Carnegie Mellon, o también está ProSoft creada en México tomando en cuenta que la mayoría de las empresas desarrolladoras de *software* son pequeñas o medianas. Estas buenas prácticas son estudiadas con detenimiento en este capítulo y determinamos cuál de todas nos ayuda más en este trabajo y definimos el porque de esta decisión.

Finalmente en el capítulo cinco llevamos a cabo la investigación que se gestó durante la creación de este trabajo, realizamos las encuestas necesarias para conocer el modo de pensar de las empresas desarrolladoras de *software* dentro del Distrito Federal al respecto de las herramientas libres y de código abierto, les preguntamos que tan cómodamente realizan sus labores diarias con ellas, que herramientas consideran que aún no están lo suficiente desarrolladas como para tomarlas como primera opción para sus tareas, y también que herramientas creen que ya son lo suficiente robustas para realizar sus labores. En este capítulo se muestran todos los resultados obtenidos, y se llegan a las conclusiones de cuales programas libres sería necesario que las personas que están laborando en una empresa desarrolladora de *software* se tome un tiempo para probarlas, y seguramente los resultados obtenidos por estas serán positivos.

CAPITULO 1

Hoy día las computadoras son una necesidad básica en esta sociedad, como por ejemplo las terminales bancarias que tienen los negocios con las que podemos pagar con la tarjeta de crédito algún producto en el centro comercial, los teléfonos celulares de última generación o también llamados “*smart phones*” que nos permiten mandar un correo electrónico desde cualquier punto con cobertura, que por cierto poco a poco permiten hacer cada vez más actividades y lo concerniente a las llamadas telefónicas es lo último que pregunta la persona que desea adquirirlo, o la computadora con la que cuentan los automóviles modernos, que sin la cual los frenos *ABS* que permiten viajes mas seguros seria imposible.

Los hechos listados en el párrafo anterior no serian posibles si no se contara con un equipo de cómputo para poder ayudarnos en el proceso, por lo tanto es necesario que hoy día contemos con algún conocimiento básico de computación, en la época actual es rara la escuela dentro del área metropolitana que no ofrezcan cursos de computación a sus alumnos como un complemento a su plan de estudio, y enseñan por lo menos las habilidades más comunes para trabajar con el, como puede ser navegar en Internet para mandar y recibir correos electrónicos, y el conocer algún programa de ofimática que nos permita realizar una hoja electrónica.

Por lo tanto seria útil que todos contáramos con los conocimientos informáticos necesarios para poder cubrir nuestras necesidades computacionales elementales, como por ejemplo, el tener en mente que en contadas ocasiones para realizar una actividad en nuestro equipo de cómputo solamente existe una opción, el estar convencidos que el programa que estamos adquiriendo es el exacto para cubrir de la mejor manera nuestros requerimientos y que estamos pagando lo justo por ella, entre otras características más.

El grueso de las personas desconoce que existe una licencia informática llamada “*free software*”, que en su traducción al español es “*software libre*”. Otro llamado “*open source*” que traducido a nuestro idioma es “código abierto”, que son soluciones profesionales de informática que cubren las necesidades listadas anteriormente y otras más, pero con un gran beneficio todas éstas, la mayoría de estos programas son gratuitos y algunos más al momento de su descarga se pide que se haga una donación de la cantidad que se desee para poder descargarlos sin romper ninguna ley de derechos de autor en el proceso. Desgraciadamente buena parte de los usuarios no lo sabe, o algunos creen que no son lo suficientemente serias para pensar en ellas para cubrir sus necesidades informáticas, cosa muy alejada de la realidad.

Esta investigación se centrará en conocer las necesidades informáticas de los pequeños negocios desarrolladores de *software* dentro del Distrito Federal que son cubiertas con herramientas de código abierto, así como también las herramientas de este tipo que aún no son lo suficientemente maduras para que

los empresarios de esta rama, las adquieran como soluciones serias para solventar sus tareas del día a día.

Metodología

Justificación:

Siempre al momento de realizar cualquier presupuesto para adquirir un equipo de cómputo es necesario saber cuanto tendremos que invertir en los programas que debe contener para realizar las tareas por las cuales lo adquirimos, o simplemente para garantizar en lo posible su buen funcionamiento como un programa antivirus, o un lector de archivos .pdf¹ que es el formato que comúnmente están publicados de manera electrónica los artículos científicos.

Es importante para cualquier usuario de cómputo el conocer todas las opciones que tienen a la mano para poder realizar sus actividades, y también que tan desarrolladas están éstas herramientas para estar seguros que lo que estamos adquiriendo es justo lo que necesitamos. Por lo mismo este estudio esta enfocado en averiguar que tan utilizadas son las herramientas de software libre en la industria desarrolladora de software dentro del Distrito Federal para entender de mejor manera que herramientas se pueden tomar como primera opción para realizar las labores diarias que necesita esta industria y puedan tener un ahorro en cuanto a costos de adquisición de software, y de misma manera que herramientas aún no realizan las tareas de la manera que se requiere y la mejor opción seria pagar por la licencia informática de su contraparte y no perder tiempo.

Antecedentes

En la realización de este escrito se emplean los términos “*free software*” o su traducción al español “*software libre*”², “*Open Source*”³ o “código abierto” de la misma manera, éstos son conceptos diferentes y hacen referencia a diferentes tipos de licenciamiento informático, mismos que se definen con el debido detalle en el capítulo 2, y en el glosario al final de la presente investigación. También se entenderá como *software libre* u *open source* a todo aquel que está bajo licencia “*GNU Public License*”⁴ y no a aquel programa de descarga gratuita, ya que ambos son casos diferentes que se explicarán también en el capítulo 2 de esta tesis.

Normalmente una empresa, del giro que sea o con el tamaño que tenga, para poder operar correctamente debe contar con al menos un equipo de cómputo

¹ El formato PDF (Portable Document Format) fue desarrollado a principios de los años 90 pensado en poder compartir documentos entre de equipos y que el formato de los mismos no se perdiera entre de los equipos. Este no es el único formato que puede llevar a cavo esta tarea pero es uno de los más populares entre de los usuarios informáticos.

² Este concepto viene definido con detalle en el glosario de términos al final de la presente investigación.

³ *Idem.*

⁴ *Idem.*

para realizar las tareas cotidianas, como por ejemplo, llevar la administración, navegar por *internet* para encontrar nuevos y mejores proveedores, comunicarse con diferentes personas a través del correo electrónico (*electronic mail* o *e-mail en su forma inglesa*), efectuar la contabilidad de la misma por medio de una hoja de cálculo, entre otras⁵.

Por lo anterior, cuando se está pensando en los costos necesarios para abrir un negocio o bien para el correcto funcionamiento del mismo, es también necesario pensar en las inversiones respecto de los equipos computacionales que se van a adquirir, ya sea en equipo nuevo o bien para la adquisición de diferentes programas de cómputo que forman parte del *software*⁶.

En un país de ingreso mediano alto como México (Banco Mundial, 2011) los costos pueden resultar demasiado elevados para que las personas o las empresas puedan adquirir todo el equipo de cómputo o *periféricos* que desearían, o en su caso todo el *software* que les gustaría para poder mantener a su empresa en un estado de sana competencia ante el mercado internacional con tendencia global que se enfrenta en la actualidad; por este motivo muchas empresas terminan adquiriendo *software* del llamado pirata⁷ o ilegal, mismo que inmediatamente⁸ puede resolver el problema, porque de manera casi instantánea se comienza a realizar el trabajo, pero no estamos resolviendo el asunto de fondo, ya que además de representar un delito informático al violar los derechos de autor correspondientes, porque al momento que las autoridades se lleguen a percatar de esto la persona o empresa se hará acreedora a una pena pecuniaria o multa, que dependiendo del *software* que se haya violado será el monto a cubrir por el concepto mencionado, que incluso podría llevar al cierre de la organización al no poder cubrir el costo de la misma.

Para (Business Software Alliance, 2011) lo anterior también provoca un problema de evasión de impuestos, ya que las personas que venden este tipo de *software* "pirata" no pagan contribuciones fiscales, ahondando más en el problema de la baja recaudación hacendaria que hay en México, y provocando una competencia desleal hacia las compañías plenamente establecidas para la venta y distribución de *software*, ocasionando que éstas no puedan alcanzar las ventas mínimas necesarias para poder continuar en el mercado, y esto puede llevar a que algunas organizaciones empiecen a vender *software* de manera ilegal para no

⁵ Estas actividades son muy variadas y pueden referirse a diseñar planos, hacer simulaciones de redes de cómputo y demás actividades informáticas.

⁶ Se refiere a los distintos programas que necesita el equipo para poder cubrir las labores por las cuales fue adquirido.

⁷ La piratería de *software* es la copia o la distribución no autorizada de programas que cuentan con derecho de autor. Esto puede suceder al reproducir, descargar, compartir, vender o instalar múltiples copias en equipos domésticos o de trabajo. Lo que muchas personas no advierten es que al adquirir *software*, están comprando una licencia para usarlo, y no el *software* como tal. Esa licencia es lo que le permitirá instalar el programa una determinada cantidad de veces, por lo que es importante que la persona lea detenidamente las condiciones de la licencia. Si hace más copias del *software* de lo que la licencia le permite, se incurre en piratería, porque se viola la autorización.

⁸ Se refiere a soluciones prontas que proporciona el sistema o programa.

tener que cerrar o en el peor de los escenarios, cerrando definitivamente.

Erróneamente, la mayoría de las personas en México creen que si compran un equipo de cómputo, éste forzosamente tiene que venir con el sistema operativo⁹ *Microsoft Windows*¹⁰, piensan que es la única manera con la que pueden funcionar todos los equipos de cómputo, nada más alejado de la realidad, ya que existe una gran cantidad de sistemas operativos para interactuar con las computadoras, por ejemplo el mismo *OsX*¹¹, que hace 6 años era impensable tratar de instalar este *software* a otro equipo que no fuera de la compañía Apple Inc.¹², pero como los equipos de esta marca ya funcionan con un procesador fabricado por Intel¹³, esto permite que una cantidad de programas que se crean especialmente para este sistema operativo, funcionen también con los equipos que no son diseñados por esta empresa estadounidense.

Si se desea realizar un trabajo escrito, normalmente se cree que la única solución es realizarlo con la paquetería *Microsoft Office*¹⁴, ello dista mucho de la realidad; existen paqueterías diferentes de ofimática que se utilizan para realizar el trabajo de oficina en equipos de computo como puede ser la paquetería *Star Office*¹⁵ de ORACLE¹⁶, o la versión de descarga gratuita de esa misma suite de ofimática que se llama *OpenOffice.org*¹⁷, (que a partir del 2012 esta empresa liberó su código y ahora es la fundación APACHE la que se encarga de mantener este proyecto) ambas son soluciones de *software* de tipo ofimática como ya se mencionó, es decir, son un procesador de textos, hoja de cálculo y creador de presentaciones electrónicas. Para el sistema operativo *OSX* su solución de ofimática es la paquetería *iWork*¹⁸, la cual cuenta con todas las soluciones marcadas anteriormente, pero sólo se puede instalar en equipos que tengan el sistema operativo mencionado.

Planteamiento del problema

“Actualmente las empresas se enfrentan a una situación en la cual la competencia y la innovación son claves para el presente y futuro de todas las

⁹ El sistema operativo es el programa principal que realiza la mayor gestión de recursos dentro del equipo de cómputo, realiza la distribución de los dispositivos con los que cuenta la computadora como por ejemplo la cámara web, el procesador, el ratón, etc., para que todos los programas que deseamos utilizar tengan lo necesario para funcionar.

¹⁰ Este sistema viene definido con detalle en el glosario de términos al final de la presente investigación.

¹¹ Este sistema operativo viene definido con detalle en el glosario de términos al final de la presente investigación.

¹² Esta empresa viene definida con detalle en el glosario de términos al final de la presente investigación.

¹³ *Idem.*

¹⁴ Esta suite ofimática viene definida con detalle en el glosario de términos al final de la presente investigación.

¹⁵ *Idem.*

¹⁶ Esta empresa viene definida con detalle en el glosario de términos al final de la presente investigación.

¹⁷ Esta suite ofimática viene definida con detalle en el glosario de términos al final de la presente investigación.

¹⁸ *Idem.*

organizaciones” (M.Christensen, 2006). La realidad es que si una empresa no cuenta con la capacidad de innovar y hacer frente a las estrategias de las demás organizaciones, no logrará desarrollarse y probablemente desaparecerá. La Innovación Tecnológica¹⁹ implica cambios técnicos en los servicios y productos de una empresa, llevándola a la competitividad. Recordemos que Michael Porter nos menciona que “La competitividad de una nación depende de la capacidad de su industria para innovar y mejorar” (Porter, 2003).

No genera sorpresa a nadie hoy día que cualquier organización mientras tenga mayor adaptación con las Tecnologías de Información y Comunicaciones, resumidas o abreviadas como TIC, ésta tendrá mayores posibilidades de alcanzar el estadio necesario para comenzar a innovar, pero muchas de las herramientas informáticas diseñadas para tal fin resultan muy costosas con miras a que una pequeña o mediana empresa de servicios pueda adquirirlas y comenzar a obtener sus beneficios.

Lo descrito en el párrafo anterior provoca que las personas que laboran en alguna pequeña empresa, al ver que están perdiendo competitividad frente a la competencia por falta de algún programa informático, adquieran alguna copia ilegal del *software* que requieren, porque este tipo de copias su costo está muy por debajo de las que cuentan con los permisos legales de los derechos de autor correspondientes, esto además de representar un delito por el hecho mismo de violar las leyes mexicanas de derechos de autor, ahonda el problema de recaudación fiscal que existe en este país así como también repercute que México esté catalogado por la BSA²⁰ en su estudio publicado en el 2011 como el doceavo productor de piratería informática en el mundo (Business Software Alliance (BSA) , 2011)²¹.

Lo anterior se debe a que las personas u organizaciones dueñas de los derechos de autor no obtienen las ganancias justas por la venta o utilización de las herramientas que crearon a base de su esfuerzo y dinero, provocando con ello que millones de pesos no pasen por una adecuada regulación fiscal y que muchas empresas dedicadas a la distribución de *software* de manera legal no alcancen las ventas mínimas para su operación, propiciando dos posibles desenlaces, que estas también terminen por vender copias de tipo ilegal, o cerrando sus puertas al no poder con la enorme competencia desleal creada por este tipo de personas u organizaciones amparadas en la falta de aplicación de la ley a nivel nacional en México.

Como ya se comento anteriormente, México ostenta el doceavo lugar en

¹⁹ Se refiere a cualquier invento puesto a la venta en el mercado, es decir, una creación o modificación de cierto producto, así como su introducción dentro de cualquier tipo de mercado aplicable.

²⁰ Business Software Alliance (BSA), es una asociación que actúa legalmente contra piratería informática en más de 60 países de Europa, Asia y América, esta asociación aglutina a las principales compañías del sector que ofrecen programas y servicios informáticos.

²¹ Business Software Alliance. (2011). Eighth Annual BSA Global Software 2010 Piracy Study. Washington DC, USA. Autor. P 5.

piratería informática, pero también tiene el sexto lugar mundial en la producción, distribución y consumo de piratería, según una lista publicada por la Organización para la Cooperación y el Desarrollo Económico (OCDE) en el 2009²², esto coloca al país en “La Lista de Observación Prioritaria” en este rubro de acuerdo con esta organización internacional. Este mismo documento indica que la Asociación protectora de cine y música de México reportó pérdidas estimadas por \$436.4 millones de dólares estadounidenses o USD, únicamente para el año mencionado, ya que 9 de cada 10 películas y *software* que se venden son copias ilegales.

Para ninguna empresa, por pequeña que esta sea, puede ser considerado como opción el hecho que para no perder competitividad frente al mercado adquiera copias ilegales de *software*, pero este tipo de problema se ahonda por la falta de conocimiento y cultura informática, ya que las personas creen que existe exclusivamente una manera de interactuar con los equipos de cómputo, y no cuentan con los conocimientos para evaluar diferentes soluciones informáticas con el fin de tomar la determinación si la herramienta es adecuada para mejorar su productividad y adaptarse mejor a su presupuesto.

Por esto se pretende estudiar una muestra de pequeñas y medianas empresas desarrolladoras de *software* dentro del Distrito Federal que conozcan e implementen diferentes herramientas de tipo *open source*, principalmente aquellas que estén afiliadas a la AMESOL²³, y se analice que herramientas informáticas con esta licencia consideran son lo suficientemente maduras para utilizarlas como primera opción, y cuales aún no tienen la madurez suficiente para poder utilizarlas en lugar del *software* de código cerrado.

Es seguro que varias empresas o personas no estén dispuestos a implementar este tipo de herramientas a sus necesidades, o que algunos programas no están lo suficientemente maduros para cubrir las expectativas necesarias, pero se estudiarán las reacciones obtenidas para que de esta manera conozcamos el tipo de necesidades informáticas que cubren estas herramientas de manera plena.

Preguntas de investigación

Pregunta principal

¿Cuáles son las herramientas *open source* que las empresas pequeñas y medianas desarrolladoras de *software* dentro del DF están utilizando como primera opción informática?

Preguntas secundarias

- ¿Cuáles herramientas de código abierto son las más utilizadas dentro de las

²² Periódico del economista publicado el 21 de octubre 2010, sección industria global.

²³ La AMESOL asociación mexicana de *software* libre es la sociedad de empresas desarrolladoras de herramientas informáticas más grande de México, y por lo tanto se busco de manera especial, la colaboración de sus miembros para la elaboración de este estudio

empresa pequeñas y medianas desarrolladoras de *software*?

- ¿Cuáles necesidades no pueden ser cubiertas por herramientas *open source* en las pequeñas y medianas empresas desarrolladoras de *software* dentro del DF?
- ¿De qué manera las herramientas *open source* ayudan a la competitividad de las empresas que las utilizan?

Objetivos

General

Identificar que herramientas informáticas de tipo *open source* son lo suficientemente desarrolladas para que las empresas productoras de *software* las puedan utilizar como primera opción.

Específicos

- Identificar que herramientas de código abierto son las más utilizadas por las empresas pequeñas y medianas desarrolladoras de *software* del DF.
- Determinar que tipo de necesidades informáticas aún no puede cubrir las herramientas de código abierto para las empresas pequeñas y medianas desarrolladoras de *software*.
- Determinar como apoyan las herramientas *open source* en la competitividad de la pequeña y mediana empresa desarrolladora de *software* ubicada en el DF.

Hipótesis

Principal de trabajo

Existen herramientas de tipo *open source* lo suficientemente desarrolladas para tomarse como primera solución informática para empresas desarrolladoras de *software* dentro del DF.

Hipótesis secundarias.

- Una empresa desarrolladora de *software* puede cubrir ciertas necesidades informáticas en base a herramientas de código abierto.
- Existen ciertas necesidades para las cuales las herramientas de código abierto aún no son lo suficientemente maduras para usarlas.
- Existe empresas pequeñas y medianas desarrolladoras de *software* competitivas que utilizan herramientas de código abierto como su primera opción.

Matriz de congruencia.

Pregunta Principal	Objetivo General	Hipótesis Principal
¿Cuáles son las herramientas <i>open source</i> que las empresas pequeñas y medianas desarrolladoras de <i>software</i> dentro del DF están utilizando como primera opción informática?	Identificar que herramientas informáticas de tipo <i>open source</i> son lo suficientemente maduras para que las empresas pequeñas y medianas desarrolladoras de <i>software</i> las puedan utilizar	Existen herramientas de tipo <i>open source</i> lo suficientemente desarrolladas para tomarse como primera solución informática para empresas desarrolladoras de <i>software</i> dentro del DF.

Preguntas Secundarias	Objetivos Específicos	Hipótesis Secundarias
¿Cuáles herramientas de código abierto son las más utilizadas dentro de las empresa pequeñas y medianas desarrolladoras de <i>software</i> ?	Identificar que herramientas de código abierto son las más utilizadas por las empresas pequeñas y medianas desarrolladoras de <i>software</i> del DF.	Una empresa desarrolladora de <i>software</i> puede cubrir ciertas necesidades informáticas en base a herramientas de código abierto.
¿Cuáles necesidades aún no pueden ser cubiertas por herramientas <i>open source</i> a las pequeñas y medianas empresas desarrolladoras de <i>software</i> dentro del DF?	Determinar que tipo de necesidades informáticas aún no puede cubrir las herramientas de código abierto para las empresas pequeñas y medianas desarrolladoras de <i>software</i> .	Existen ciertas necesidades para las cuales las herramientas de código abierto aún no son lo suficientemente maduras para usarlas.
¿De qué manera las herramientas <i>open source</i> ayudan a la competitividad de las empresas que las utilizan?	Determinar como apoyan las herramientas <i>open source</i> en la competitividad de la pequeña y mediana empresa desarrolladora de <i>software</i> ubicada en el DF.	Existe empresas pequeñas y medianas desarrolladoras de <i>software</i> competitivas que utilizan herramientas de código abierto como su primera opción.

Relevancia social de la investigación

¿Quién se beneficiará con los resultados de esta investigación?

A todas las empresas que trabajan dentro del ramo de desarrolladoras de *software*, para que conozcan las mejores herramientas y las más usadas dentro de este giro así como también los beneficios que obtendrían por utilizarlas.

¿De qué modo?

Indagando que *software* utilizan las empresas y estudiar cuales son los beneficios que obtienen las organizaciones por su uso.

¿Qué proyección social tiene?

Como ayuda las herramientas *open source* a la pequeña y mediana empresa desarrolladora de *software* mexicana del DF para mejorar su competitividad frente al mercado.

Implicaciones prácticas

¿Ayudará a resolver algún problema?

Ayudará sobre todo a reducir de manera significativa los costos de adquisición de equipo nuevo de cómputo o de *software* para las pequeñas y medianas empresas desarrolladoras de programas de cómputo, además de esta manera probablemente se reduciría de forma importante la adquisición de programas llamados “pirata” o “ilegal” y disminuiría el problema que se tiene en este país en relación con la alta evasión fiscal.

¿Qué tan conveniente es la investigación?

Esta investigación será muy conveniente para todas las personas y/o empresas que desean adquirir un equipo de cómputo (*hardware*) o *software* para su negocio, ya que con el conocimiento de las diferentes herramientas existentes para cubrir parte de las necesidades más comunes, o en el mejor de los casos sus necesidades completas, podrían ahorrarse una cantidad significativa de dinero y dejar de lado la terrible dependencia tecnológica que existe en el mundo hacia los diferentes productos informáticos, principalmente extranjeros y de grandes empresas.

Metodología empleada.

Para el Marco Teórico se realizó un estudio informacional para identificar el estado del arte respecto a elementos de estado de las TIC (Tecnologías de la Información y Comunicaciones) en el estado actual de las pymes mexicanas que se ve con detenimiento en el capítulo tres, así como también un estudio de las diferentes herramientas de código abierto y de los programas de software libre y open source así como también las herramientas que no están bajo estas licencias como el software de código restrictivo, el freeware, entre otras que se estudia en el capítulo dos, y un estudio de diferentes herramientas para medir la competitividad en las empresas desarrolladoras de software en el capítulo cuatro.

El alcance de la investigación es descriptivo al especificar tareas que se están llevando a cabo bajo el uso de herramientas de código abierto así como también las tareas que éstas por cualquier motivo no puedan realizarse con estos programas de cómputo. Es exploratoria porque no se encontró ningún estudio o tabla relacionada con el uso de herramientas informáticas por parte de las pequeñas y medianas empresas desarrolladoras de software dentro del DF, así como también en las diferentes entrevistas que se realizaron a miembros destacados dentro de la rama de desarrollo de software dentro del Distrito Federal, ninguno pudo señalar algún estudio previo que se halla escrito al respecto. También es cuantitativa porque se están recabando datos que serán estudiados en profundidad en el capítulo cinco.

Diseño de la investigación.

La investigación se llevó a cabo en las empresas pequeñas y medianas desarrolladoras de *software* dentro del DF principalmente a las empresas pertenecientes al grupo llamado “KaraOKulta – La comunidad de desarrolladores de Videojuegos en Latinoamérica”, perteneciente a la red social de facebook, a las empresas empadronadas a la organización de la CANACINTRA²⁴ como empresas prestadoras de servicios informáticos y a la AMESOL con datos obtenidos por encuesta a las empresas que así me lo permitieron, poniendo principal énfasis a aquellas que usan de manera consciente y cotidiana las herramientas de tipo *open source*, porque todas las personas que usamos un equipo de cómputo somos usuarios de *software* de licencia libre como por ejemplo la máquina virtual de java,

²⁴ Esta organización se explicará con el debido detalle en unidades de análisis.

que gran parte del contenido en internet está diseñado con este lenguaje de programación y si no lo tuviéramos sería imposible el poder verlo, o también el protocolo DNS, este protocolo lo usamos para conectarnos a internet, y todas las compañías que brindan servicio de conexión a internet lo utilizan para sus suscriptores.

Elaboración del instrumento.

La construcción del instrumento de medición se llevó a cabo de acuerdo a los lineamientos establecidos por (Hernandez Sampieri, Fernández Collado, & Baptista Lucio, 2006), de acuerdo a las siguientes fases.

Fase 1:

Redefiniciones fundamentales sobre propósitos y participantes.

Fase 2:

Revisión de la literatura para medir las variables de interés.

Fase 3:

Identificación de los conceptos y variables a medir, así como sus indicadores.

Fase 4:

Decisión con respecto a que formato será utilizado.

Fase 5:

Construcción del instrumento, generación de ítems, codificación y niveles de medición de los reactivos.

Fase 6:

Prueba piloto ensayo y probabilidad.

Fase 7:

Versión final, posibles cambios y versión definitiva.

Fase 8:

Entrenamiento de las personas que van a calificarlos.

Fase 9:

Obtener la autorización para aplicarlo.

Fase 10:

Administración del instrumento. Preparación de los datos para el análisis, confiabilidad, validez, objetividad y análisis de la información.

Redefiniciones fundamentales:

1.- ¿Qué va a ser medido?. Elementos que proporciona el *software* libre y *open source* para la pyme desarrolladora de *software* en el Distrito Federal.

2.- ¿Qué o quienes van a ser medidos? La mayor cantidad de pymes desarrolladoras de *software* dentro del Distrito Federal, tomando principalmente a los miembros de la AMESOL (Asociación Mexicana de *Software* Libre), la CANACINTRA (Cámara Nacional de la Industria de Transformación) y el grupo de “KaraOKulta – La comunidad de desarrolladores de Videojuegos en Latinoamérica”, perteneciente a la red social de facebook.

3.- ¿Cuándo se van a aplicar las encuestas? Las encuestas se van a aplicar durante los meses de noviembre, diciembre del año 2011 y enero del 2012.

4.- ¿Dónde se van a aplicar las encuestas? Las encuestas se van a aplicar por internet principalmente, pero cuando los factores lo permitan también se aplicaran en el lugar de trabajo o el algún punto que la persona nos especifique.

5.- ¿Cuál es el propósito de recolectar los datos? Identificar la documentación, funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y transportabilidad que proporcionan las herramientas *open source* y de *software* libre a las empresas desarrolladoras de *software* dentro del Distrito Federal, así como también determinar cuales son las herramientas de licencia libre más usadas por la industria y cuales aún no pueden competir en el campo profesional.

6.- ¿Qué tipo de datos se van a obtener? Datos descriptivos de las empresas de tipo ordinales, porque las preguntas cuentan con una serie de niveles que expresan una actitud de acuerdo o desacuerdo con respecto a las características con las que cuenta el *software* que utilizan.

Unidades de análisis.

Como se verá más adelante en el capítulo cuatro cuando se estudie a profundidad la importancia de las TIC dentro de las pymes así como también la manera en la que éstas ayudan en la competitividad, midiendo la competitividad en cuanto a costos, tiempo y adaptabilidad de las herramientas informáticas entre de otros factores, pero como principal objeto de estudio para esta investigación se tomó las empresas pertenecientes a la AMESOL, CANACINTRA y “KaraOKulta – La comunidad de desarrolladores de Videojuegos en Latinoamérica”, perteneciente a la red social de facebook, ya que estas enfocan sus esfuerzos en la creación de herramientas de tipo informático, y algunas de las empresas el *software* que desarrollan lo licencian como *software* libre (que en el capítulo dos se estudiarán con detalle todas las diferencias que existen entre las licencias de tipo *software* libre y las de código abierto entre otras).

En síntesis las unidades de análisis son las diversas empresas desarrolladoras de *software* que su domicilio fiscal se encuentre dentro de la demarcación del Distrito Federal, pero prestando mayor atención a las afiliadas a cualquiera de las organizaciones citadas anteriormente.

¿Qué es la AMESOL?

“Es un organismo no gubernamental sin fines de lucro que se encuentra fundado bajo la filosofía del *software* libre y que además abraza valores tales como el respeto, la tolerancia y el bien común, entendiéndose estos términos dentro del entorno empresarial mexicano. Por lo cuál a partir de su creación en Marzo del 2003, esta asociación a luchado constantemente por ser un punto de partida por dar forma e identidad a la industria por medio de la comunidad empresarial dedicada al *software* libre (amesol.com.mx, 2011)²⁵”.

²⁵ Extracto tomado directamente de la página de la AMESOL, (última revisión 4-11-2011).

¿Qué es CANACINTRA?

En 1941 un grupo de empresarios industriales se percataron de la necesidad de crear una agrupación que representara el sector industrial del país, y que estuviera integrada por ellos mismos, con la finalidad de aglutinar los intereses, objetivos y planes de ese sector de la sociedad. Así fue como nació la Cámara Nacional de la Industria de la Transformación.

Esta es una organización que hasta el día de hoy después de más de 70 años está conformada por 80 delegaciones por todo el país divididas en 9 regiones, 5 oficinas en la ciudad de México, una sede nacional representada por 12 sectores industriales y 97 ramas especializadas además de contar con 60 comisiones integradas por 12 comités directivos encargados de representar a la industria afiliada ante el gobierno .

KaraOKulta

El nombre completo de este grupo es “KaraOKulta – La comunidad de desarrolladores de Videojuegos en Latinoamérica”, este es un grupo que se encuentra dentro de la página de facebook.com. Es un grupo que nació para poder tener un medio de comunicación veloz con la mayor cantidad de empresas desarrolladoras de video juego.

Es la comunidad más grande y activa de esta rama de empresas desarrolladoras de *software* que pude encontrar dentro de mi investigación, y me ha sido de mucha ayuda por que con solamente ingresar a la red social, ya podía empezar a obtener retroalimentación de su parte.

Construcción del instrumento.

Los reactivos fueron desarrollados tomando como base el siguiente cuadro de variables. Cada uno de ellos ha sido redactado de la manera más clara posible y directa para asegurarse de que las respuestas son las más apegadas a la realidad de la empresa.

Las variables fueron tomadas principalmente del estándar ISO 9126 que en el capítulo cuatro se explica con detalle la razón de tomar las características con las que debe de contar un buen *software* para que en esta investigación se tome como variables para medir la competencia.

Cuadro de variables -

Tabla 1.2.- Cuadro de variables.

Variable	Definición	Reactivo	Categoría
Nombre de la empresa	Nombre de la organización en la que se labora	Nombre de su compañía	Abierta
Puesto dentro de	El puesto que tiene dentro de	Puesto dentro de la	Abierta

la empresa	su empresa	empresa	
Edad.	La edad del encuestado	Edad.	1.- Menor de 25
			2.- 25 a 35.
			3.- 35 a 45.
			4.- 45 a 55.
			5.- Mayor de 55.
Nivel de estudios.	Nivel de estudios del encuestado	Nivel de estudios.	1.- Primaria terminada.
			2.- Secundaria terminada.
			3.- Bachillerato terminado.
			4.- Licenciatura terminada.
			6.- Estudios de posgrado.
Banda Ancha	Cuenta con una conexión a internet de banda ancha	¿Cuál es la velocidad de conexión de internet que tiene en su negocio?	1.- Menor a 1M. 2.- Entre de 1M y 2M. 3.- Entre de 4M y 5M. 4.- Mayor a 5M.
Pregunta filtro	La persona encuesta utiliza herramientas de tipo open source.	¿Utiliza herramientas de código abierto o software libre?.	1.- Si. 2.- No.
Listado	Que la persona encuestada nos liste cuales son las herramientas de este tipo que utiliza	¿Cuáles son las herramientas de código abierto o libre que utiliza más frecuentemente?	Abierta.
Tamaño de la empresa	Determinar en que tipo de empresa labora el encuestado	¿Cuántos empleados tiene la organización en la que laboras?	1.- Hasta 10. 2.- 11 a 50. 3.- 51 a 100. 4.- Más de 100.
Funcionabilidad			
Adecuación	Determinar si la herramienta utilizada cuenta con la suficiente adecuación.	¿El software libre (open source) que utiliza tiene la capacidad de brindarle un conjunto	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.

		apropiado de funciones?	
Exactitud.	Determinar si la herramienta utilizada cuenta con la suficiente exactitud.	¿El software libre (open source) que usa le brinda los resultados correctos por los que la adquirió?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Interoperabilidad.	Determinar si la herramienta utilizada cuenta con la suficiente interoperabilidad.	¿El software libre (open source) que utiliza puede interactuar con al menos otro software?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Seguridad de acceso.	Determinar si la herramienta utilizada cuenta con la suficiente seguridad de acceso.	¿El software libre (open source) tiene la capacidad para proteger la información de manera que las personas no autorizadas no puedan leerlos o modificarlos, al tiempo que no se le niega el acceso al personal autorizado?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Fiabilidad.			
Madurez	Determinar si la herramienta utilizada cuenta con la suficiente madurez	¿El software libre (open source) evita fallar por resultados de mala programación?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Tolerancia a fallos.	Determinar si la herramienta utilizada cuenta con la suficiente tolerancia a fallos.	¿El software libre (open source) cuenta con un nivel especificado de prestaciones en caso de fallos o de infringir sus interfaces especificados?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Recuperabilidad.	Determinar si la herramienta utilizada cuenta con la suficiente recuperabilidad.	¿El software libre (open source) cuenta con un nivel de prestaciones especificado y tiene la capacidad de recuperar los datos directamente afectados en caso de fallo?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Usabilidad.			
Comprensibilidad.	Determinar si la herramienta utilizada cuenta con la suficiente comprensibilidad.	¿El software libre (open source) permite al usuario entender si es	1.- Si. 2.- La mayoría 3.- Algunos.

		adecuado y cómo puede ser usado para unas tareas o condiciones de uso particulares?	4.- Ninguno.
Facilidad de aprendizaje.	Determinar si la herramienta utilizada cuenta con la suficiente facilidad de aprendizaje.	¿El software libre (open source) permite al usuario aprender sobre su aplicación?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Operabilidad.	Determinar si la herramienta utilizada cuenta con la suficiente operabilidad.	¿El software libre (open source) permite al usuario operarlo y controlarlo?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Atractivo.	Determinar si la herramienta utilizada cuenta con el suficiente atractivo.	¿El software libre (open source) es atractivo al usuario?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Eficiencia.			
Tiempo de respuesta	Determinar si la herramienta utilizada cuenta con el suficiente tiempo de respuesta.	¿El software libre (open source) proporciona tiempos de respuesta, tiempos de proceso y potencia apropiados, bajo condiciones determinadas?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Utilización de los recursos.	Determinar si la herramienta utilizada cuenta con el suficiente utilización de recursos.	¿El software libre (open source) usa las cantidades y tipos de recursos adecuados cuando lleva a cabo su función bajo condiciones determinadas?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Mantenibilidad.			
Analizabilidad.	Determinar si la herramienta utilizada cuenta con la suficiente analizabilidad.	¿El software libre (open source) se le puede diagnosticar fácilmente deficiencias o causas de fallos?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Cambiabilidad	Determinar si la herramienta utilizada cuenta con la suficiente cambiabilidad.	¿El software libre (open source) permite fácilmente que una determinada modificación sea implementada?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.

Estabilidad	Determinar si la herramienta utilizada cuenta con la suficiente estabilidad.	¿El software libre (open source) cuenta con la capacidad para evitar efectos inesperados debidos a modificaciones?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Facilidad de pruebas.	Determinar si la herramienta utilizada cuenta con la suficiente facilidad de pruebas.	¿Capacidad del software libre (open source) que permite que si es modificado sea validado?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Transportabilidad.			
Adaptabilidad.	Determinar si la herramienta utilizada cuenta con la suficiente adaptabilidad.	¿El software libre (open source) tiene la capacidad para ser adaptado a diferentes entornos específicos, sin aplicar acciones o mecanismos distintos de aquellos proporcionados para este propósito por el propio software considerado?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Facilidad de instalación.	Determinar si la herramienta utilizada cuenta con la suficiente facilidad de instalación.	¿Es de fácil instalación?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Coexistencia.	Determinar si la herramienta utilizada cuenta con la suficiente coexistencia.	¿El software libre (open source) puede coexistir con otro software independiente, en un entorno común, compartiendo recursos comunes?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Recambiabilidad.	Determinar si la herramienta utilizada cuenta con la suficiente recambiabilidad.	¿El software libre (open source) puede ser usado en lugar de otro programa, para el mismo propósito, en el mismo entorno?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Productividad	Determinar si el encuestado considera que las herramientas le dan la suficiente productividad.	¿Considera que su productividad es la misma cuando trabaja con herramientas de código abierto o libres?.	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.

Desarrollo	Conocer que herramientas informáticas considera el encuestado son lo suficientemente desarrolladas como las de código cerrado.	Liste por favor las herramientas de tipo open source que considere son tan competitivas como las de código cerrado	Abierta
Desarrollo	Conocer que herramientas el encuestado considera que aún le falta desarrollo.	Liste por favor que herramientas de código abierto aún no son lo suficiente competitivas como su equivalente de código cerrado.	Abierta

Instrumento.

Cuestionario herramientas de código abierto desarrolladoras de *software* en las PYME's.

Nombre de su compañía	Abierta
Puesto dentro de la empresa	Abierta
Edad.	1.- Menor de 25 2.- 25 a 35. 3.- 35 a 45. 4.- 45 a 55. 5.- Mayor de 55.
Nivel de estudios.	1.- Primaria terminada. 2.- Secundaria terminada. 3.- Bachillerato terminado. 4.- Licenciatura terminada. 6.- Estudios de posgrado.
¿Cuál es la velocidad de conexión de internet que tiene en su negocio?	1.- Menor a 1M. 2.- Entre de 1M y 2M. 3.- Entre de 4M y 5M. 6.- Mayor a 5M.
¿Utiliza herramientas de código abierto o software libre?.	1.- Si. 2.- No.
¿Cuáles son las herramientas de código abierto o libre que utiliza más frecuentemente?	Abierta.
¿Cuántos empleados tiene la organización en la que laboras?	1.- Hasta 10. 2.- 11 a 50. 3.- 51 a 100. 4.- Más de 100.
¿El <i>software</i> libre (<i>open source</i>) que utiliza tiene la capacidad de brindarle un conjunto apropiado de funciones?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) que usa le brinda los resultados correctos por los que la adquirió?	1.- Si. 2.- La mayoría 3.- Algunos.

	4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) que utiliza puede interactuar con al menos otro <i>software</i> ?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) tiene la capacidad para proteger la información de manera que las personas no autorizadas no puedan leerlos o modificarlos, al tiempo que no se le niega el acceso al personal autorizado?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) evita fallar por resultados de mala programación?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) cuenta con un nivel especificado de prestaciones en caso de fallos o de infringir sus interfaces especificados?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) cuenta con un nivel de prestaciones especificado y tiene la capacidad de recuperar los datos directamente afectados en caso de fallo?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) permite al usuario entender si es adecuado y cómo puede ser usado para unas tareas o condiciones de uso particulares?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) permite al usuario aprender sobre su aplicación?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) permite al usuario operarlo y controlarlo?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) es atractivo al usuario?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) proporciona tiempos de respuesta, tiempos de proceso y potencia apropiados, bajo condiciones determinadas?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) usa las cantidades y tipos de recursos adecuados cuando lleva a cabo su función bajo condiciones determinadas?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.

¿El <i>software</i> libre (<i>open source</i>) se le puede diagnosticar fácilmente deficiencias o causas de fallos?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) permite fácilmente que una determinada modificación sea implementada?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) cuenta con la capacidad para evitar efectos inesperados debidos a modificaciones?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿Capacidad del <i>software</i> libre (<i>open source</i>) que permite que si es modificado sea validado?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) tiene la capacidad para ser adaptado a diferentes entornos específicos, sin aplicar acciones o mecanismos distintos de aquellos proporcionados para este propósito por el propio <i>software</i> considerado?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿Es de fácil instalación?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) puede coexistir con otro <i>software</i> independiente, en un entorno común, compartiendo recursos comunes?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿El <i>software</i> libre (<i>open source</i>) puede ser usado en lugar de otro programa, para el mismo propósito, en el mismo entorno?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
¿Considera que su productividad es la misma cuando trabaja con herramientas de código abierto o libres?	1.- Si. 2.- La mayoría 3.- Algunos. 4.- Ninguno.
Liste por favor las herramientas de tipo open source que considere son tan competitivas como las de código cerrado	Abierta
Liste por favor que herramientas de código abierto aún no son lo suficiente competitivas como su equivalente de código cerrado.	Abierta

Muchas gracias por su participación.

Confiabilidad y Validez.

La confiabilidad puede ser medida y permite comprobar los indicadores de la investigación pueden estar relacionados entre sí, además que las medidas están libres de errores aleatorios. Así, se puede decir que la confiabilidad es la

exactitud de la medición, puesto que este instrumento está basado sólo en la descripción y exploración de hechos.

Por otro lado, la Validez se refiere al grado en que un instrumento de medición mide realmente lo que pretende medir y no otros factores. (Puga Murguía, 2002) Ésta se puede analizar desde tres aspectos.

- a) Validez de Contenido: Grado en que una prueba representa el universo de reactivos del cual se extrajo y es útil sobre todo para evaluar la utilidad de las pruebas.
- b) Validez de Constructo: Juicio de lo apropiado de las inferencias extraídas de las puntuaciones de la prueba (instrumento de medición) respecto a posiciones individuales en una variable llama constructo.
- c) Validez de Consistencia Interna: Evidencia de homogeneidad del instrumento de medición.

De este modo, se puede sustentar la validez de contenido al considerar que los reactivos fueron extraídos de una base sólida como lo es el Marco Teórico, donde se identifica el estado del arte del mismo, pues incluso el cuadro de variables sustenta este razonamiento.

CAPITULO 2

INTRODUCCIÓN.

El motivo de este capítulo es expresar los conceptos técnicos que se van a estar utilizando de manera frecuente a lo largo de toda la investigación, como lo es el concepto de *software* libre y el de *open source*, que son conceptos que en cuanto a la filosofía que manejan son diferentes pero en el ámbito de trabajo parece señalarnos lo mismo, a su vez estos dos también son totalmente diferentes al *software* tipo *freeware* pero como normalmente tanto los programas de código abierto y libres se pueden descargar de manera gratuita es muy normal que el usuario promedio confunda estos tres.

También se hablará de otros conceptos igual de importantes como las demás licencias con las que puede ser protegido un programa como lo es la licencia de código cerrado (como por ejemplo esta licencia es con la que esta protegido Microsoft Windows) y algo muy importante dentro de todo esto, las ventajas y desventajas que contiene cada una de las licencias. Ya que normalmente no pensamos que todo producto que adquirimos viene acompañado con las ventajas (que son las razones por la que adquirimos esa herramienta) pero que a su vez también vienen con desventajas inherentes en cada una de ellas.

Lo que se pretende con este capítulo no solamente es que se entienda la parte técnica de esta investigación si no que también comprenda el lector que dentro de las soluciones informáticas que nos rodean no solamente existen las mas populares como las que son distribuidas por las empresas de Microsoft o Apple, si no que hay un gran número de empresas que están tratando todos los días de colocar sus productos en el mercado y que las soluciones que nos pueden proporcionar compiten en calidad con estas, y probablemente sean superiores en cuanto a precio, pero como estas empresas no cuentan con los recursos suficientes para hacer campañas agresivas de publicidad y mercadotecnia quedan casi inmediatamente fuera de toda posibilidad de poder entrar al mercado, y como normalmente este tipo de compañías caen dentro de la categoría de pymes, normalmente no pueden cumplir todos los requerimientos exigidos por el cliente (como puede ser el gobierno).

¿QUÉ ES EL SOFTWARE?

En el capítulo pasado ya estudiamos a grandes rasgos qué es el *software* libre, pero será necesario el ahondar mas en la explicación de este concepto para que quede firme porque es un término fundamental en esta investigación, para comenzar con esto primero explicaremos lo que es el *software*.

Software.

“No tiene esencia física, solo una representación objetiva. El *software* es una idea o una serie de ideas, es decir, el producto de la imaginación o intelecto

de alguna persona” (Rosch, 1996). También llamado programa es simplemente la parte de la computadora que no esta compuesta por ninguna pieza física, radica completamente dentro del disco duro de la misma, y es la interfaz que permite al usuario el interactuar con la parte física del equipo.

Para la Real Academia Española *software* es: “Conjunto de programas, instrucciones o reglas informáticas para ejecutar ciertas tareas en una computadora”. Como podemos observar existen diferentes definiciones similares para éste término, pero una de las mas citadas dentro del mundo de la informática es la que el IEEE²⁶ le ha dado en su estándar 729, “Es el conjunto de los programas de cómputo, procedimientos, reglas, documentación y datos asociados que forman parte de las operaciones de un sistema de operación”. Con esta definición podemos asentar que es todo lo intangible de un equipo de cómputo, es el conjunto de instrucciones que permite su utilización.

Hoy día el *software* se ha transformado en una de las principales herramientas a nivel global, existen pocos aparatos electrónicos que no cuenten con el, como por ejemplo todos los celulares que existen tienen diferentes programas que sirven para darnos algún servicio con ellos, los automóviles modernos todos cuentan con computadoras y todas ellas tienen *software* para hacer el viaje mas seguro y mas agradable, como la misma radio con la que cuenta el automóvil o el dispositivo de frenado ABS (Anti Blokier System) que nos brinda una manera más segura de viajar.

También esta siendo el generador de fortunas más importantes del planeta, como por ejemplo Bill Gates que es la segunda persona más rica del planeta (según revista Forbes²⁷) siendo la base de su fortuna el *software*, específicamente el sistema operativo Microsoft Windows que es el sistema operativo más usado en el mundo.

Existen diferentes tipos de *software*, como aquel que solamente nos ayuda a escuchar alguna estación de radio dentro del automóvil o aquellos con los que podemos hacer una hoja de texto como Microsoft Office, u otros que juegan un papel mucho mas crítico como aquellos con los que cuentan los aviones modernos para ayudar a los pilotos en su tarea de llevar a los pasajeros a buen destino, también existen diferentes clasificaciones dentro de los mismos, pero la clasificación que vamos a utilizar para esta investigación será la que se desglosa dependiendo de la licencia con la que halla sido registrado, como lo es el *software* libre o el *software* propietario.

²⁶ Institute of Electrical and Electronics Engineers, (instituto de ingenieros eléctricos y electrónicos) es una institución fundada en 1884, por Thomas Alba Edison y otros ingenieros renombrados de la época. Es la asociación mas grande de ingenieros en el campo de la electricidad y carreras afines del mundo. Tiene facultades suficiente para la creación de normas y estándares en este campo además de informática, mecatrónica, redes de comunicación, entre otras más.

²⁷ La revista Forbes esta especializada en el mundo de las finanzas y los negocios. Todos los años emite una lista donde están inscritos las personas con la fortuna personal mas grande del mundo según sus cálculos.

DIFERENTES TIPOS DE SOFTWARE.

Software Propietario.

Este es uno de los términos que más controversia dentro del mundo de la informática puede ocasionar, porque no existe un término aceptado por la comunidad informática en su totalidad, por lo tanto hay diferentes maneras para definirlo como las siguientes: *software* no libre, *software* restrictivo, *software* privativo, *software* de código cerrado, *software* propietario y de código restrictivo entre otras.

La principal característica con la que cuenta este tipo de *software* es que resulta imposible ver el código fuente²⁸ con el que han sido creados, evitando que algún usuario que lo ocupe pueda ver la manera en la que están escritas las instrucciones y pueda copiar o alterar de alguna manera como están escritas estas. Provocando con esto que si algún programa tuviera un error o si el usuario final deseara alterar de alguna manera al programa para su mayor beneficio no lo pueda hacer o que sea muy difícil el lograrlo, además que seguramente esto sería una tarea ilegal. Por lo tanto el término que se utilizará para denominar este tipo de programa en este trabajo de investigación será el de “*software* de código cerrado”, no quiero decir con esto que las demás maneras de identificar esta licencia sean incorrectas o que sean poco asertivas, pero con este término se puede identificar fácilmente la principal característica del mismo.

Este tipo de *software* es el más común, la mayoría de los programas que usamos en nuestro que hacer diario con las computadoras tiene una licencia de código cerrado, como por ejemplo Microsoft Windows que es el sistema operativo con el que funciona la mayoría de las computadoras, Microsoft Office que es el *software* de ofimática con el que hacemos nuestros escritos y hojas de trabajo, Internet Explorer que es un navegador web, AutoCad que es el programa con el que la mayoría de los arquitectos e ingenieros realizan sus planos entre otros muchos más.

La empresa de Microsoft no es la única compañía que produce este tipo de *software*, AutoCad le pertenece a AutoDesk, el sistema operativo de OSX le pertenece a Apple Inc, así como también le pertenece la licencia de iTunes programa que sirve para escuchar música, ver videos. entre otras cosas, así como también el navegador web Safari, GarageBand que sirve para grabar y editar música, entre otros programas más. Todos los programas aquí listados tienen licencias de tipo código cerrado, y fueron creadas por diferentes empresas de esta rama.

Dentro de esta categoría, el de *software* de código cerrado existen algunas

²⁸ El código fuente es toda la lista de instrucciones que componen al programa final que nos ofrecen, son todas las instrucciones de programación que están acomodadas de una manera ordenada y sistematizada para que cumplan con el propósito por el cuál fue elaborado del programa.

sub categorías dentro del mismo, (Culebro, Gómez, & Torres, 2006) listan los siguientes.

Freeware.

No tiene una definición clara y precisa, sin embargo suele usarse para clasificarse al *software* que podemos descargar, usarlo y copiarlo libremente pero no así alterarlo por que no nos muestra su código fuente. Un ejemplo de este tipo de *software* es Flash Player de la empresa Adobe, es la herramienta informática que usamos para ver videos por la web pero el *freeware* no es *software* libre.

Shareware.

Este es el caso de *software* que podemos descargar u obtener por cualquier medio y somos libres de utilizarlo por un tiempo determinado, o que sus características no están completas mientras no paguemos la cantidad que nos indique su creador. Pero nuevamente no podemos alterar ninguna de sus característica porque no viene acompañado de su código fuente, ejemplo de este tipo de *software* son aquellos que podemos descargar versiones de prueba.

Abandonware.

El abandonware es el *software* cuyos derechos de autor ya no son defendidos o que ya no está siendo vendido por la compañía que lo creó, (Carranza Torres, 2004) y por lo mismo se dice que ha sido abandonado.

Warez.

Éste término es muy usado en la jerga de las personas con conocimientos en informática para aludir a un programa que ha sido violada su seguridad para evitar que sea alterado su código, y éste pueda ser instalado y ejecutado sin las medidas de seguridad diseñadas por su creador, también se puede referir a que ha sido alterado su código y ahora se le a agregado o quitado una característica a la que originalmente no estaba diseñado. Los *warez* son tipos de *software* que son distribuidos en violación a la licencia de derechos de autor, causando de esta manera que su distribución sea ilegal. La principal justificación que dan los *crackers*²⁹ para permitirse hacer este tipo de trabajos es en la injusticia que provoca el esconder información, y que solamente las personas con los recursos suficientes pueden acceder a esta. Ejemplos de este tipo de *software* son todos los programas informáticos que estén en diferentes puntos para su descarga, y que al momento de instalarlos ya no piden un número de serie para su instalación.

Software Semilibre.

Contiene las mismas características que el *software* libre para los usuarios individuales, entidades educativas y organizaciones sin fines de lucro, pero

²⁹ Éste término es utilizado cuando un experto en informática toma sus conocimientos para hacer tareas poco éticas con ellas, como por ejemplo entrar a computadoras ajenas de una empresa o persona y robar información de ella, como tomar números de cuentas bancarias con sus contraseñas o cuentas de correo de los trabajadores de la empresa.

prohíbe esas libertades para su uso comercial o empresarial, un ejemplo de este tipo de programas es el manejo de bases de datos MySQL este *software* es libre si deseas utilizarlo para cuestiones de escuela o no lucrativas, pero si deseas obtener soporte por parte de la empresa para fines lucrativos tienes que pagar por su uso.

Software libre.

Utilizando la definición que la misma *Free Software Foundation*³⁰ le otorga: “El *software* libre es una cuestión de la libertad de los usuarios de usar, copiar, distribuir, estudiar, cambiar y mejorar el *software* que tiene a su alcance. Más precisamente se refiere a los programas que cuenta con las 4 libertades³¹”.

Libertad 0.- Libertad de correr el programa para cualquier propósito.

Libertad 1.- La libertad de estudiar como funciona el programa, y cambiarlo si es que se desea para que haga lo que el usuario quiera que haga. El acceso al código fuente es una condición para poder hacer esto.

Libertad 2.- La libertad de distribuir copias para que puedas ayudar a tus conocidos.

Libertad 3.- La libertad de distribuir copias de tu programa modificado a otros. Haciendo esto puedes darle a la comunidad entera la oportunidad de beneficiarse de tus cambios. El acceso al código fuente es una condición para poder hacer esto.

Dentro del *software* libre hay, a su vez, matices que son necesarios el tomarlos en cuenta. Como lo es el *software* de dominio público que significa que no está protegido por los derechos de autor (o *copyright* en inglés) provocando con esto que se pudieran hacer versiones no libres del mismo, mientras tanto que el *software* libre protegido por el *copyleft*³² impide a los redistribuidores incluir algún tipo de restricción a las libertades propias del *software* así concebido, es decir, que garantizan a las demás versiones que se hagan de ese programa que seguirán siendo libres.

También es muy conveniente el no confundir el *software* libre con el *software* gratuito, es muy cierto que mayormente el *software* libre se distribuye de manera gratuita pero esta no es una característica que esté listada dentro de las cuatro libertades con las que debe de contar, además que hay cierto *software* que se distribuye de manera gratuita y está muy lejos de ser libre, como por ejemplo todas las versiones de prueba de programas, estas versiones solamente son

³⁰ Es una organización sin ánimo de lucro cuyo objetivo es promocionar la libertad de los usuarios de computación, y defender los derechos de los usuarios de software libre, también es la organización fundadora y que brinda apoyo a toda aquella persona o empresa que haga o use software libre.

³¹ Dato tomado de la página de la Free Software Foundation con dirección electrónica: <http://www.gnu.org/philosophy/free-sw.html> (última revisión 13-06-2011).

³² Copyleft es el término que define a un grupo de licencias que garantizan que el producto que tiene alguna de estas contiene las 4 libertades para ser considerado software libre, y es una “broma” que el propio Richard Stallman creó como contra posición al “copyright” de los derechos de autor.

programas que te permiten trabajar hasta cierto número de días con ellos, o solo funcionan algunas características del original, y si deseamos gozar de los beneficios de este programa en su totalidad, tenemos que pagar o hacer lo necesario que quiera el autor para poder adquirirlo, este concepto ya fue estudiado en unas cuantas paginas anteriores en este capítulo.

El *software* libre comenzó en pequeñas comunidades de entusiastas de la computación que tenían la firme idea de querer cambiar la industria del *software*, al paso del tiempo no solamente ha alcanzado atraer a personas, si no también a empresas clave en cuanto a la industria de la computación como es IBM³³, INTEL³⁴, NOKIA entre de otras.

Inicialmente el *software* libre fue muy popular en los enormes servidores, estaba limitado a ser un sistema que no era visible a los usuarios finales, pero a través del trabajo de cientos de voluntarios ahora es un sistema que es usado por miles de usuarios en todo el mundo, como simple ejemplo podemos observar al navegador web MOZILLA FIREFOX, que en su página de descargas nos muestra que desde la publicación de la versión 4 de este navegador hasta el día de hoy, ya se ha descargado más de 214,000,000³⁵ (y contando) en el mundo entero, y la principal fuente de descargas de este navegador son usuarios finales que lo desean para su uso diario y pueden no saber que están utilizando *software* libre.

Ahora ya se ha convertido en un sistema usado por millones de personas en todo el mundo, como en las computadoras personales, teléfonos celulares, agendas y más. Aunque usar el *software* libre es similar a usar cualquier otro, este es un legado que le pertenece a toda la humanidad, cada persona en el mundo tiene derecho a usarlo, modificarlo y copiarlo de la misma manera que los autores de este mismo. Es un legado que tiene propietario pero este no tiene interés en crear un monopolio con su conocimiento, de la misma manera que las leyes básicas de la física o las matemáticas, no existe ningún monopolio y no tienes que pagar para su uso.

Un factor importante en los últimos años ha sido la adopción del *software* libre en los gobiernos. Algunos lo han adoptado para ahorrar dinero, otros por cuestión de seguridad, unos más para la creación de industrias locales y fortalecer a la economía nacional, para que de esta manera se creen empleos dentro del país y no sea una empresa transnacional que solamente tiene contratado a un comisionista que se encarga de cerrar los tratos de manera local, y que todo su

³³ Este año IBM cumple 100 años de existencia, y entre de los logros mas significantes que en lista por su permanencia en le mercado, cita la incursión al software libre entre de ellas.

³⁴ El 23 de junio del 2009 Intel y Nokia anuncian una relación estratégica para crear el sistema operativo Meego, que esta diseñado para trabajar en teléfonos celulares y protegido con una licencia open source. Dato tomado de la sala de prensa electrónica de Intel con dirección electrónica: <http://www.intel.com/cd/corporate/pressroom/emea/spa/424533.htm> (última revisión 16-06-2011).

³⁵ Dato tomado de la página del centro de descargas de FIREFOX 4 con dirección electrónica: <http://glow.mozilla.org/#> (última revisión el 13-06-2011).

equipo de producción y pago de impuestos lo hace en otro país.

Licencia GNU/GPL

La licencia *General Public License* (GPL), fue creada por la *Free Software Foundation* en 1989³⁶, y esta diseñada para asegurar la libre distribución, modificación, uso del *software*, asegurarse que recibirá el código fuente si es que así lo quiere el usuario, y todo programa que este protegido por ésta es considerado *software* libre. Actualmente se encuentra en su versión 3 liberada al público desde el 2007.

Open source (código abierto).

Durante el año de 1998, un grupo de expertos informáticos encabezados por Eric S Raymond³⁷ y Bruce Perens³⁸ junto con otros miembros involucrados en el movimiento del *software* libre, decidieron crear la iniciativa conocida como *Open Source Initiative* y crearon el término de *Open Source* (código abierto en español) en contra posición al término de *Free*³⁹ *Software* para que les pareciera este entorno de programación más atractivo a las empresas, ya que el término de “Free” (en ingles tiene dos acepciones que es libre y gratis), le resulta muy complicado a las empresas el aceptar este término, y la mayoría de los directivos en cuanto lo escuchaban inmediatamente presentaban negativas al mismo.

La mayoría de las empresas ahora prefieren este término de *open source* para evitar dar la percepción de estar creando *software* gratis, ya que aún el *software* libre no tiene que ser forzosamente gratuito, y para poner énfasis en el valor diferencial que el *software* que están proveyendo viene acompañado con su código fuente para que el usuario final pueda modificarlo a sus necesidades específicas.

Bruce Parens, miembro de la *Open Source Initiative* y antiguo coordinador de la distribución de Debian⁴⁰, creó una lista de condiciones que debe de cumplir un programa para poder ser considerado como *open source*. Estas condiciones son muy similares a la lista de condiciones de *software* libre, ya que desde el

³⁶ Dato tomado de la página de gnu.org sitio oficial de la licencia GNU/GPL, con dirección electrónica: <http://www.gnu.org/licenses/licenses.html#GPL> (última revisión 16-06-2011).

³⁷ Es un experto en informática, una de sus principales aportaciones al proyecto de software libre ha sido la creación de la publicación llamada La catedral y el Bazar, también a contribuido con código a la librería ncurses para el compilador GCC que es el principal compilador de código C dentro del software libre. Co fundador del Open Source Initiative.

³⁸ Experto informático que lideró el sistema operativo Debian basado en GNU/Linux, además es un experto en gráficos y animación por computadora, trabajo por 20 años para Pixar. Es el autor del término open source (código abierto) y cofundador de Open Source Initiative.

³⁹ El problema encontrado con la palabra *free* en ingles es que tiene dos significados, libre y gratis, esto provoca confusiones en este idioma, ya que las personas angloparlantes piensan que para un software tenga la licencia de free software esta obligado a que sea gratuito, pero como ya se ha mencionado, esta no es una característica que este listada dentro de las cuatro libertades.

⁴⁰ Debian es una distribución de GNU/Linux y es un sistema operativo completo, reside dentro de el puro software libre, pero que también tiene soporte para *software* que no lo sea.

comienzo de este proyecto todos los miembros de este, han aceptado que el padre de este movimiento es la *Free Software Foundation*, solamente lo han querido matizar un poco para adaptarlo al ámbito empresarial.

Lista de condiciones para que un programa sea considerado de código abierto.

- 1.- Libre distribución: No se puede impedir la venta o distribución del programa o parte de él. Así mismo tampoco se puede exigir el pago de un canon o tasa de cambio de su distribución por parte de terceros.
- 2.- Código fuente: El programa debe de incluir su código fuente y no se debe de restringir de modo alguno su distribución.
- 3.- Trabajos derivados: No debe impedirse realizar trabajos derivados o modificaciones del programa y debe permitirse que estos sean distribuidos bajo los mismo términos del *software* original.
- 4.- Integridad del código fuente original: Puede exigirse que una versión modificada del programa tenga un nombre y número de versión diferente que el programa original para poder proteger al autor original de la responsabilidad de estas versiones.
- 5.- No discriminación contra personas o grupos: Las condiciones de uso del programa no pueden discriminar contra una persona o grupo de personas.
- 6.- No discriminación contra usos: No se puede negar a ninguna persona hacer uso del programa para ningún fin, ya sea comercial o militar.
- 7.- Distribución de la licencia: Los derechos contenidos dentro del programa deben aplicarse a todos aquellos programas con los que esta siendo distribuido para su necesidad de ejecución.
- 8.- La licencia no debe de ser específica de un producto: Los derechos garantizados al usuario del programa no deben depender de que el programa forme parte de una distribución o paquete particular de *software*.
- 9.- La licencia no debe de restringir otro *software*: La licencia no debe poner restricción a otro *software* que sea distribuido junto con la licencia de *software*. Por ejemplo, la licencia no debe insistir que todos los otros programas distribuidos en el mismo medio deben ser programas de código abierto.
- 10.- La licencia debe ser tecnológicamente neutral: No puede existir ninguna disposición de la licencia que obligue al uso de una tecnología concreta.

DIFERENCIAS DE SOFTWARE LIBRE Y DE CÓDIGO ABIERTO.

Como podemos observar no existen grandes variantes entre de *software* libre y de código abierto, si no más bien podemos decir que su mayor diferencia es filosófica, ya que para el *software* libre pueda identificarse como tal, es necesario que tenga la licencia creada por Richard Stallman llamada *GNU GPL* que actualmente se encuentra en su versión 3, ésta licencia impide a los redistribuidores de *software* que añadan a su licencia cualquier restricción que no esté en la licencia original. Esta metodología conocida como "*copyleft*" impide a cualquier usuario tomar *software* libre o parte de él y lo quiera cambiar a *software* de código cerrado. Por lo tanto la *Free Software Foundation* solamente acepta que

un *software* sea libre si éste está protegido por la licencia *GPL*, negando con esto que el *software* con licencia *open source* pueda ser conocido como libre.

El *software* de código abierto es entendido por algunos como una modalidad de *software* libre, o como el término general que engloba a todo el *software* libre, lo cierto es que nace como una separación de la *Free Software Foundation* en 1998.

Conviene aclarar que la *Open Source Initiative* no tiene una licencia en específico para ellos en comparación con la *Free Software Foundation* con su licencia *GNU/GPL*, solamente tienen el conjunto de 10 directrices para la clasificación de licencias relativas a aplicaciones y productos de *software*, para este propósito la *Open Source Initiative* a elaborado una marca de certificación (*OSI Approved*) para determinar que un producto cumple con sus términos y una certificación (*OSI License Approved*) para determinar que una licencia cumple con sus definiciones.

Este es el principal motivo por lo que en este trabajo de investigación se va a tomar los términos de *software* libre y *open source* como iguales, y no se va a hacer diferencia entre de ellos al momento de referirse a los programas, ya que su principal diferencia radica en cuanto a la licencia por la que están protegidos y la finalidad de la presente investigación es conocer la aceptación del mismo dentro de la pymes desarrolladoras de *software* dentro del Distrito Federal, no si el empresario conoce la diferencia entre los dos términos.

Ventajas del *software* libre y del *software* de código abierto.

Un objetivo importante de este estudio es el que el usuario pueda conocer una mirada general del *software* hoy día, y una parte importante de esto es que pueda conocer las ventajas y desventajas de estos dos tipos de licenciamiento, ya que normalmente no encontramos la solución perfecta si no mas bien la que mas cubre nuestras necesidades. Para trabajar en este apartado primero listare las ventajas del *software* libre que (Culebro, Gómez, & Torres, 2006) listaron en su libro *software libre Vs software propietario*⁴¹, pero las explicaciones que acompañan a los puntos son de creación propia.

Bajo costo de adquisición y libre uso.

El *software* que usamos normalmente no esta a la venta, lo que adquirimos cuando compramos un CD con un programa, es una licencia de uso donde esta estipulado en el contrato muy claramente lo que se puede hacer con ese *software* así como también lo que no. Una buena analogía de esto es cuando se esta rentando una casa, la casa nunca va a ser propiedad del inquilino ni tampoco puede hacer uso libre del inmueble, solamente tiene permitidas las tareas que están listadas en el contrato de arrendamiento, y si llegara a hacer alguna tarea que no estuviera listada en el contrato, éste podría ser penado por su casero, y si fuera considerada grave la falta puede ser expulsado de la misma.

⁴¹ Culebro, J. Gómez, W. Torres, S. (2006). *Software libre vs Software propietario Ventajas y desventajas*. (1ra ed). México.

Esto no sucede así con el *software* libre, una vez que se paga por algún programa este le pertenece completamente a la persona que pago por el, ya que si esta adquiriendo también el código fuente, permitiéndole hacer todos los cambios que crea pertinente para cubrir todas sus necesidades específicas, sin necesidad de tener que ir con la persona que lo programó y pedirle que le haga cambios necesarios para nosotros.

Por lo tanto cuando adquirimos una solución bajo licencia *open source* o *GPL* los costos de uso inmediatamente se ven reducidos, porque normalmente junto con una solución informática con licencia de código cerrado también viene estipulado el costo mensual que se debe de pagar para poder continuar usando el programa, más los costos de mantenimiento, los de adaptación del *software* y todos los demás que de aquí se generen.

Y una de las grandes ventajas que nos proporciona el que venga el programa acompañado con su código fuente esta en que no viene con costo asociado. Reafirmo este punto, por el *software* libre solamente se paga una vez y nada mas, solamente se debe de pagar por adaptar el programa a nuestras necesidades específicas a cualquier persona con los conocimientos suficientes para esto, con licencias de código cerrado solamente la empresa que vende el *software* es la única que puede resolver problemas del sistema, porque nunca va a mostrar el código de su programa, además que sí se llegara a dar el caso que se modificara el programa esto representaría un delito por los derechos de autor, y seguramente dentro de las cláusulas del contrato esto estaría penado y se tendría que pagar multas por este hecho.

Innovación tecnológica.

La ideología principal de los defensores del *software* libre esta en que la información debe ser completamente gratuita y debe de estar al alcance de todo aquel que la solicite, sin distinción alguna. El *software* libre y *open source* tienen como objetivo principal compartir la información trabajando de manera cooperativa, este es el principal modelo con el que la humanidad a avanzado y atrae grandes beneficios como lo son:

- Todo mundo tiene derecho a acceder a su diseño y a aprender de el.
- Todo el mundo tiene derecho a modificarlo.
- Es de libre propagación todas las variantes que se hagan de este.
- Tiende a ser muy eficiente (porque muchas personas pueden ingresar a su código ayudando a reparar problemas de manera completamente altruista).
- Tiende a ser muy robusto (muchas personas tienden a contribuir, porque es de su interés que funcione de manera estable y eficiente el programa).
- Tiende a ser muy diverso (por ejemplo podemos citar que el navegador web MOZILLA FIREFOX ya esta traducido a más de 500 lenguas

- diferentes incluido el maya y el nahuatl).
- Ayuda a generar el deseo de crear nuevo *software* (como están aprendiendo del que ya esta creado las personas se les puede ocurrir crear uno nuevo y que sea mejor que el que están estudiando).
 - Da mucha seguridad porque tenemos el control del *software*.
 - Reutilización del conocimiento (algo que resulta vital para la sociedad del conocimiento en la que vivimos hoy día, ya que es muy ineficiente estar redescubriendo el hilo negro una y otra vez, de esta manera las personas pueden adquirir las lecciones aprendidas por alguien más y mejorarla).
 - Aprender alguna técnica de programación (puede incentivar a las personas con conocimientos técnicos suficientes a aprender diferentes tipos de lenguajes de programación para poder mejorar algún *software* que estén utilizando).
 - Reconocimiento entre colegas (algo también importante en la sociedad del conocimiento radica en tener identificados a los expertos en cualquier tema, ya que así cuando enfrentamos algún problema en específico sabemos con quien podemos acercarnos y encontrar la mejor solución).
 - Aquí no están listadas todas, solo las más comentadas en diferentes bibliografías .

Requisitos de *hardware* menores y durabilidad en las soluciones.

No se puede generalizar de manera fehaciente, pero normalmente las soluciones *open source* siempre requieren de menor infraestructura para su instalación y adaptación. Ya que normalmente una de las metas de todas las empresas es que los usuarios estén constantemente consumiendo, por lo tanto buscan la manera de hacer que sus programas consuman más recursos tanto de procesador, memoria RAM y de disco duro, esto provoca que una computadora con unos 3 o 4 años desde su fabricación sea considerada como muy lenta, aunque todo el equipo funcione a la perfección, esto es causado a que el *software* con el que cuenta ya no puede trabajar en el equipo porque no tiene la infraestructura que su programador nos obliga a tener, causando de esta manera que las soluciones tengan menor durabilidad.

Con el *software* libre esto no sucede, ya que si alguna persona enfrenta el problema que desea instalar algún programa específico y no pueda porque necesita mas poder de procesamiento su computadora, si cuenta con los conocimientos suficientes o contrata a alguien que los tenga puede entrar al código y depurar las opciones que no le sean necesarias, haciendo mas ligero el programa y ayudando a que pueda funcionar sin necesidad de muchos recursos físicos.

Otro punto a tomar en cuenta es que algún autor de código cerrado podría llegar a decidir dejar de dar soporte a alguna arquitectura informática dada, provocando de esta manera que sus usuarios tengan que adquirir las plataformas

que si seguirán siendo soportadas sí es que tienen la que dejará de funcionar. Un ejemplo de esto ocurre con el *software* de Flash Player de la empresa Adobe, que a finales del año 2011 anunció que ya no le daría soporte a su versión de BlackBerry, obligando a esto que la propia empresa de Research in Motion (dueña de la marca Black Berry) anunciara a todos sus clientes que se prepararan para migrar a otras soluciones. Con el *software* de código abierto esto no puede pasar, porque las personas inmediatamente pueden ingresar al código y seguir dando soporte al sistema ellos mismos, si tienen el conocimiento suficiente o puedan contratar a quien si lo tenga para ayudarlos.

Escrutinio público.

Por la manera en la que están definidas las 4 libertades que acompañan a todo *software* libre y las reglas que tiene el código abierto, todo programa debe de venir acompañado por su código fuente, y que todas las personas que así lo deseen pueden aprender del mismo y modificarlo en el proceso. Esto ayuda a que todo el *software* libre este expuesto al escrutinio de cualquiera, ayudando a que si hay un error en el código este pueda ser reparado de manera inmediata y que la persona pueda compartir con todos la solución que creó. Un ejemplo de esto es escuchar que se han encontrado huecos de seguridad en cierto programa, pero si este es de código cerrado es imposible tratar de corregirlo uno mismo, tenemos que esperar hasta que el fabricante tenga los recursos necesarios y encuentre la solución, haciendo de esta manera que la corrección se retrase demasiado mientras más personas resultarán afectadas.

Independencia del proveedor.

Este es un punto que la mayoría de los gobiernos que desean empezar a migrar sus plataformas informáticas a soluciones de código abierto remarcan, cuando se adquiere un *software* de código cerrado el cliente esta completamente ligado al proveedor del mismo, provocando que si se desea agregar una nueva función o alterar alguna que ya se tuviera, o reparar algún error que tenga el mismo, se tiene que esperar hasta que el fabricante así lo quiera o pueda. También podemos llegar a extremos como lo puede ser que el fabricante tome la decisión de dejar de brindar soporte a alguna versión determinada de sus programas, obligando a todos sus clientes a migrar a otra diferente y probablemente proveída por el mismo, obligando a que sus clientes terminen comprando lo que el quiere vender no lo que ellos necesitan, y los clientes están completamente indefensos a las determinaciones del fabricante, provocando con esto una dependencia tecnológica tremenda e injustificada, porque el sistema no puede dar un solo paso si el fabricante no lo desea. Todo este escenario es inexistente con el *software* libre, ya que el cliente puede contratar a cualquier empresa o persona con los conocimientos técnicos necesarios para que empiece a dar manteniendo al sistema con el que cuenta, ayudando con esto a tener una independencia técnica sana y los clientes cuentan con el poder de decisión de escoger a quien prefieran para brindarles soporte.

Industria local.

Algo que todos los gobiernos están obligados es a apoyar a la industria local. Cuando sacan una licitación para adquirir una solución informática es normal que las primeras en meter presupuestos sean las empresas transnacionales de *software* que trabajan en todo el mundo, es cierto que estas empresas tienen contratados a miles de personas, pero no todos esos puestos de trabajo están en el país, lo común es que esas empresas solamente tengan contratado a un puñado de personas en el país y su única función es la de estar al tanto de los contratos que se están presentando de manera local, y de mandar todos los puntos a las oficinas principales que normalmente son extranjeras para que desde allá se calculen todos los pormenores, como tiempo necesario para desarrollar el programa, costos, puesta a punto, etc; y finalmente todo lo que se pagó es dinero que sale del país y los impuestos que generan en México son los mínimos. Probablemente la cotización que se le pagó a esta empresa resulto ser más barata que la que presentaron los competidores nacionales, pero ésta empresa nacional si crea empleos locales, los impuestos los paga en el país y el dinero queda fluyendo dentro de las fronteras.

Con soluciones de código abierto se pueden generar empresas locales ya que al momento de adquirir el sistema de este tipo, cualquier empresa con el personal adecuado puede brindar el servicio de soporte, mantenimiento y solución al sistema, y si este proveedor llegara a fallar por cualquier motivo se puede cambiar a alguno más. Una manera de poder hacer esto pudiera ser que en vez de contratar a una sola empresa con un sistema de varios millones de pesos para realizar una solución en específico, contratar a varias empresas con requerimientos mas pequeños para que una fabrique la solución, otra más le de mantenimiento, otra este arreglando problemas, etc.

Datos personales y seguridad.

Uno de los puntos fundamentales que todos los estados están obligados a proveer es la seguridad en la información que todos aportamos, como por ejemplo la cantidad que pagamos de impuestos, pagos fiscales, numero de pasaporte, número de cédula profesional, entre de otros, este tipo de información son muy sensibles y si es que llegaran a caer en malas manos podrían causar grandes daños a los ciudadanos por un error de algún funcionario público.

Pero también un grave problema podría ser que datos muy importantes del mismo estado pudieran ponerse en peligro, como los son los programas llamados “*snnifers*⁴²”, que son programas que sirven para monitorear el tráfico que esta circulando por una red, y saber cuales son las páginas que estoy visitando, y hasta los datos que estoy colocando dentro de la misma como nombres de usuario y contraseñas.

⁴² Un software de tipo snnifer son programas especializados en monitorear los sucesos a su alrededor, principalmente la información que se distribuye en la computadora huésped. Haciendo de esta manera que algún usuario mal intencionado pueda acceder a información sensible.

Problemas mucho peores como la instalación de “*malware*⁴³” o “*spyware*⁴⁴”, este tipo de programas se caracterizan porque las funciones que realizan dentro del equipo donde están instalados siempre van encaminadas en dañar al usuario, como por ejemplo abrir puertas traseras para poder monitorear sus actividades, y no solamente las que están desarrollándose en internet, si no todo lo que hace fuera de la misma como copiar los documentos con los que está trabajando o hasta poder ver completamente lo que esta en su monitor de la computadora. Este tipo de programas son los que mas daños provocan, y siempre es tarea del administrador de la red y de los equipos de cómputo estar muy atentos para evitar que puedan instalarse este tipo de programas, y en el caso de ya estar instalado removerlo inmediatamente del equipo.

Con *software open source* esto no es posible que suceda, ya que en dado caso de que se llegara a detectar que el programa esta realizando actividades fuera de lo común inmediatamente se puede revisar su código fuente y descubrir cuál es el problema, pero con el *software* cerrado esto es imposible, solamente se puede revisar cuales son las puertas traseras que se abren y cuanta información se esta transmitiendo pero no cuál ni adonde se esta dirigiendo esta.

Por estas causas se le puede considerar como circunstancia de seguridad nacional el usar solo programas *open source* porque toda actividad que ejecute el programa puede ser estudiada, comprendida y si llegara a ser necesario, modificarla para que el usuario se sienta seguro con el uso de esta herramienta.

Adaptación del *software*.

Una de las principales características que nunca se debe de perder de vista cuando adquirimos un *software* con licencia de código cerrado es que estamos adquiriendo precisamente eso, una licencia, realmente no estamos adquiriendo el programa si no el permiso de uso, y como cualquier otro permiso de uso tenemos prohibido tratar de hacer alguna modificación a la propiedad, ya que solo tenemos licencia de uso y no nos pertenece de ninguna manera el producto por el que estamos pagando.

Entonces estamos pagando por el uso del *software* completo, incluidos los errores que vengan con el mismo, y tenemos estrictamente prohibido cualquier intento de cambio, aunque estemos tratando de resolver un problema que hasta el propietario del mismo resultaría beneficiado.

⁴³ También conocido como *badware* en ingles, este tipo de programas tiene como objetivo el infiltrarse y dañar al equipo de computo donde este es alojado sin el consentimiento del propietario, como por ejemplo el listar cuales son los programas que tiene el usuario, los sitios que visita en internet o peor aun, los nombres de usuario y las contraseñas que utiliza.

⁴⁴ *Spyware* es una variedad de *malware*, que tiene como objeto especifico el estar listando los nombres de usuario y contraseñas que utiliza el usuario de la computadora sin su consentimiento, para después enviar estos datos a un tercero que puede hacer mal uso de esa información.

Esto no pasa con *software* de código abierto, ya que todos los programas vienen acompañados de su código fuente permitiéndonos adaptarlo tanto como sea necesario para el cliente, o para uno mismo si se cuenta con los conocimientos necesarios, esto proporciona una de las mayores ventajas del *software* libre.

Y las licencias de código cerrado principalmente están diseñadas para atar de manos a los usuarios, casi se puede decir que ve a sus usuarios como enemigos más que como clientes, ya que lo normal es que el cliente quede satisfecho con el producto y normalmente se le invita a usarlo como mejor le convenga, con este tipo de licencia principalmente busca amenazar a sus clientes para que solo use su programa de cierta manera, y si quiere modificarlo puede ser demandado aunque el resultado que se busca le resulte beneficioso al fabricante mismo.

No hay virus.

Uno de los problemas que constantemente todos los usuarios están preocupados con sus equipos de cómputo son los virus informáticos, siempre son una amenaza a la que todos están expuestos, y para tratar de protegerse es necesario el desembolsar recursos para adquirir e instalar un antivirus en nuestro equipo. Como la mayoría del *software* que instalamos en sistemas *open source* podemos ver su código fuente, sí se llegara a ver un mal funcionamiento en algún programa se puede ingresar a ver donde esta el error e inmediatamente cambiar las líneas de código necesarias para que deje de realizar la función contraria al usuario e inmediatamente copiar la solución para que todos la corrijan.

Esto no quiere decir que estén exentos de programas tipos “*malware*” o “*spyware*”, es más, en los años recientes cada vez aparecen más estos tipos de programas para los sistemas operativos diferentes de Windows, pero virus informáticos que borren información contenida en el disco duro de la computadora no.

Aumenta la competencia y la innovación.

Como nos lo explica muy bien el maestro Edgar Buenrostro Mercado (2010)⁴⁵ en una entrevista para la página de Internet linuxparatodos.net⁴⁶, está demostrado que cuando existe una herramienta de código abierto para algún nicho de mercado, las herramientas que tienen licencia de código cerrado generan actualizaciones de manera más rápida y están mas atentas a satisfacer de manera

⁴⁵ La entrevista completa que se le hizo al Maestro Edgar Buenrostro Mercado el día 26-05-2010 se puede escuchar en la siguiente dirección electrónica: <http://radiocontempo.wordpress.com/2010/05/26/entrevista-con-el-mtro-edgar-buenrostro-e-impacto-social-del-software-libre/> (última revisión 23-07-2011).

⁴⁶ La página de Internet <http://linuxparatodos.net> es un blog especializado en temas de *software* libre y *open source*, se dedica principalmente a proporcionar las últimas noticias que se producen en el campo y a dar cursos de capacitación en diferencias herramientas que cuenten con estas licencias.

más inmediata los problemas que tienen sus usuarios.

Este fenómeno siempre resultara provechoso para los usuarios ya que las organizaciones que desarrollan la herramienta de tipo cerrada constantemente estarán tratando de crear una mayor diferenciación con su competidor mediante la innovación, ya que con el precio lo más común será que la herramienta libre sea gratis, pero sí con su programa esté constantemente colocando nuevas opciones y haciendo más fácil la labor al usuario.

Esta es solamente una lista breve de las mayores ventajas que nos ofrece el *software* libre, si continuamos pensando al respecto podríamos listar aún más, pero estos son ejemplos suficiente ilustrativos para tener una idea clara de los beneficios que siempre vendrán acompañados con este tipo de herramienta, pero también tienen desventajas que también vendrán. Pero un buen ejemplo de las grandes ventajas que nos ofrecen estas soluciones se puede ver en la tabla 2.1 que el 91% de las súper computadoras en el mundo trabajan con alguna distribución de GNU/Linux, sistema operativo que es licenciado como *software* libre.

Tabla 2.1: Tabla donde nos muestra que de las 500 computadoras mas veloces, 455 trabajan con GNU/Linux y solamente 6 con el sistema operativo Microsoft Windows.

Operating system Family	Count	Share %	Rmax Sum (GF)	Rpeak Sum (GF)	Processor Sum
Linux	455	91.00 %	53455235	78424464	6441088
Windows	6	1.20 %	459520	563535	63140
Unix	23	4.60 %	1776736	2284365	127536
BSD Based	1	0.20 %	122400	131072	1280
Mixed	15	3.00 %	3116134	3776512	1146880
Totals	500	100%	58930025.59	85179949.00	7779924

Fuente: Tabla de elaboración propia con datos de la página www.top500.org⁴⁷.

Desventajas del software libre y de código abierto.

Curva de aprendizaje más pronunciada.

Uno de los principales problemas de este tipo de *software* esta en que generalmente es un poco mas difícil su uso, como unos ilustra muy claramente la investigación desarrollada por (San Martín Alonso, Sales Arasa,, & Peirats Chacón, 2010) que en dos escuelas a nivel primaria en una comunidad Valenciana los profesores que repentinamente de un ciclo escolar al otro se vieron forzados a cambiar las herramientas informáticas que normalmente estaban acostumbrados a

⁴⁷En la página de internet llamada www.top500.org crea una lista mes con mes de los equipos de cómputo más rápidos, y en la lista del mes de junio del 2011 publica esta tabla para mostrar los sistemas operativos mas usados dentro de estos computadores:
<http://www.top500.org/list/2011/06/100> (última revisión 12 de junio del 2011).

utilizar a otras de código abierto, todos presentaron un alto grado de resistencia al cambio, ya que se sintieron relegados en la toma de decisiones de su propio centro de trabajo, principalmente los profesores con mayor edad que argumentaban que pusieron su mayor interés en el aprendizaje de las pasadas herramientas, y que veían todo ese esfuerzo tirado a la basura porque ahora estaban obligados a aprender otras herramientas que consideraban más complicadas.

El *software* libre normalmente no tiene garantía proveniente del autor.

Esta cláusula se presenta normalmente cuando se adquiere una herramienta gratis, ya se explico ampliamente que no es obligatoria la gratuidad en el *software* libre o de código abierto, cuando se contrata a una empresa para que desarrolle una herramienta a la medida, al momento de diseñar el contrato un punto que no puede faltar es que tipo de garantías les ofrece al cliente, por lo tanto si pueden haber garantías tan amplias como el contrato lo especifique. Pero si es una herramienta que se descargo por internet de un servidor ftp público, todo ese *software* viene sin ninguna garantía por parte del creador.

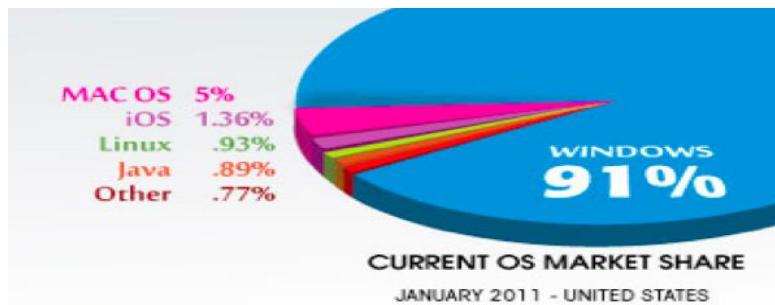
Normalmente los programas libres sus interfaces gráficas no son tan intuitivas como las de código cerrado. Este es el punto que principalmente el autor de este trabajo de investigación cree que es el factor principal por el cuál las herramientas de código abierto no han entrado de manera masiva a las computadoras personales. La mayoría de las personas que diseña este tipo de programas son voluntarios, normalmente no reciben compensación económica alguna por todas las horas que están trabajando, esto provoca que se enfoquen casi exclusivamente en que la herramienta funcione y punto, el aspecto gráfico pasa a segundo plano u otro aún mas inferior.

Además estas herramientas están pensadas en que el usuario que las va a utilizar cuenta con cierto grado de conocimiento técnico o que son expertos en el tema. Y hasta ahora no ha habido muchos diseñadores que estén dispuestos a trabajar de manera voluntaria para hacer que alguna herramienta luzca una interfaz mas atractiva e intuitiva.

Este es el punto porque en el campo de cómputo de alto rendimiento la gran mayoría de los equipos trabajan bajo *software* libre⁴⁸, pero el grueso de la población mundial tiene instalado programas de código cerrado, ya que los programas son más agradables visualmente y son más intuitivos al momento de usarse, principalmente debido a que este tipo de empresas pueden invertir fuertes cantidades de dinero en contratar a un grupo de diseñadores gráficos para que su única tarea sea crear diseños agradables e interfaces fáciles de usar.

⁴⁸ En la página de internet llamada top500.org crea una lista mes con mes de los equipos de cómputo más rápidos, y en la lista del mes de junio del 2011 las primeras 10 computadoras trabajan con el sistema operativo GNU/Linux. La dirección electrónica de esta lista es: <http://www.top500.org/list/2011/06/100> (última revisión 12 de junio del 2011).

Gráfica 2.1.- Nos muestra que aproximadamente el 91% de las computadoras en los Estados Unidos hacen uso del sistema operativo Microsoft Windows.



Fuente.- Fragmento de una gráfica hecha por www.makeuseof.com⁴⁹

No existen compañías únicas que respalden toda la tecnología.

Este es un punto por el cuál puede resultar muy complicado el que puedan obtener contratos muy grandes dentro del gobierno por ejemplo, ya que comúnmente las empresas que se dedican a desarrollar *software* de código abierto suelen ser pymes y esto provoca que se especialicen en ciertos procesos, provocando que para poder fabricar una herramienta que sea muy amplia sería necesario el contratar a dos o más empresas que trabajen en conjunto para poder desarrollarla, esto provoca que se caiga en franca desventaja con empresas transnacionales que tienen a cientos de empleados y pueden disponer de las personas necesarias para fabricar el producto en menor tiempo.

Faltan controladores para algunos dispositivos.

Una práctica que aún resulta común para los fabricantes de *hardware*, es el no liberar sus controladores para las piezas que fabrican. Por lo tanto algunas piezas de cómputo pueden resultar imposibles de instalar bajo herramientas de código abierto, ya que solamente funcionan bajo uno o dos sistemas operativos, esto es algo que puede resultar muy inconveniente a la hora de querer migrar a otra plataforma diferente porque puede ser muy penoso o imposible el configurar de manera confiable todos tus dispositivos.

Únicamente los proyectos importantes o populares cuentan con soporte.

Como ya se comentó previamente una de las principales fuentes de soporte para las herramientas de tipo *open source* son personas que trabajan como voluntarios, no cobran un centavo por todas las horas que pueden invertir en crear o darle mantenimiento a algún programa, esto provoca que sí este grupo de personas pierde el interés porque apareció una herramienta nueva que puede hacer mejor la labor o ya no puede invertir mas tiempo en brindar mantenimiento a

⁴⁹ La página www.makeuseof.com es una página estadounidense especializada en software, la liga para poder observar la gráfica original es: <http://www.makeuseof.com/tag/history-operating-systems-infographic/> (última revisión 13 de julio del 2011).

la misma, el *software* queda a la deriva junto con todos los usuarios que trabajan con esa herramienta. un caso es el servidor web APACHE, que es *software* libre, pero pueden mantener su proyecto adelante gracias a que tienen un contrato con IBM que les paga una cantidad a cambio que le estén brindando mantenimiento constante al servidor *web*. Por lo tanto, existen clientes que no pueden correr el riesgo que algunas de sus herramientas queden sin mantenimiento.

Como queda de manifiesto el *software* libre tiene muchas ventajas que pueden dar lugar a grandes beneficios, pero no existe ninguna herramienta libre de errores, así como también las herramientas de código cerrado tienen ventajas y desventajas.

Ventajas del software de código cerrado.

Control de calidad.

Algo que siempre tendrá más los programas de código cerrado que los de código abierto es el control de calidad porque como son un equipo muy cerrado el que trabaja para producirlo, se puede controlar de una manera mas ordenada su producción, sobre todo por el punto que las personas son pagadas por la creación del mismo, son personas que están dedicadas de manera competa en el programa que están creando, y algo normal en los programadores que crean *software* libre es que lo hacen en tiempos libres, de manera altruista sin la mínima intención de lucrar, obligándolos a buscar una fuente de ingresos que les garantice sus gastos y no pueden trabajar de manera ordenada en el proyecto que están desarrollando.

Personal altamente capacitado.

Las grandes empresas de *software* están constantemente buscando a personas que sean hábiles programadores, y como el conocimiento humano es en estos días el activo mas valioso, la manera con la que atraen a estos programadores es ofreciendo sueldos competitivos, esto hace que las personas se preparen intensamente en estos puntos y grandes programadores de la industria están dentro de las nóminas de estas empresas.

Uso común por los usuarios.

Este es otro de los puntos que considero una de las mayores causas por las cuales el *software* de código cerrado tiene tan fuerte penetración dentro de los hogares de las personas, ya que cada día es más común trabajar desde el hogar, y lo que quieren las personas es que la misma interface que tienen en su trabajo sea la que tienen en sus hogares y viceversa, por lo tanto, cuando quieren comprar un equipo para su uso personal siempre buscarán el que tenga el mismo sistema operativo con el que están más acostumbrados y con el que saben trabajar mejor.

Como ejemplo nuevamente voy a citar la investigación de (San Martín Alonso, Sales Arasa,, & Peirats Chacón, 2010), donde nos hace mención que gran parte de los profesores que presentan descontento con el cambio de plataforma

(incluyéndose a sí mismo) ya tenían una cantidad de programas que ellos mismos recolectaban en la red en tiempos libres en sus hogares para ayudarse en su trabajo como docente, y al momento de hacer el cambio todos esos programas se volvían inservibles y no podían hacer el mismo proceso de recolección nuevamente porque no consideran tener los conocimientos necesarios para poder instalar programas en la nueva plataforma.

Software para aplicaciones muy específicas.

Otro punto que también es muy importantes para el uso de las herramientas cerradas es que existen programas que no se pueden usar en otro sistema operativo, son programas muy específicos y algunos pueden ser vitales como por ejemplo el SAE, que no funciona en otra plataforma que no sea Microsoft Windows.

Se encuentra fácilmente ayuda al respecto.

Como los sistemas operativos cerrados son los más utilizados es muy fácil encontrar soporte técnico, sí se llega a presentar algún error en el equipo por *software* cualquier técnico puede brindar apoyo, pero es más complicado encontrar a personas que tengan experiencia en solucionar problemas con sistemas *open source*.

Desventajas de los programas de código cerrado.

Secreto del código fuente.

Lo que más quieren proteger las empresas que producen programas de código cerrado es precisamente las líneas por las que esta conformado su programa, y esto siempre va a ser un enorme problema para el usuario final, porque al adquirir el *software* nunca podrá tratar de adaptar el programa a sus necesidades específicas, está obligado a acercarse al propietario del programa y que le pague exclusivamente a el para que los cambios que el requiere puedan ser elaborados, esto lo deja en clara dependencia tecnológica hacia su proveedor de *software* y cae dentro de un monopolio tecnológico que le podría llegar a resultar al cliente muy perjudicial.

Quedar sin soporte técnico.

También este problema se presenta en los programas de código cerrado, porque también es muy posible que el fabricante de un *software* en específico quede fuera de mercado y se vea obligado a cerrar su empresa, esto haría que todos sus clientes queden a la deriva en cuanto al mantenimiento de la herramienta que habían adquirido y se vean obligados a tener que pagar por la adquisición de una nueva. Sírvese nuevamente como ejemplo el caso de Adobe y BlackBerry listado anteriormente.

Alta cantidad de virus.

Este problema se presenta principalmente en el sistema operativo Microsoft

Windows, hay una cantidad de millones de virus informáticos programados para atacar esa plataforma, y siempre es un riesgo el infectarnos con algunos de ellos por el simple hecho de abrir un archivo equivocado que nos llegó por correo o nos lo pasaron por algún medio de almacenamiento.

Nulificación de desarrollo tecnológico de la industria nacional.

Las grandes empresas de *software* no tienen sus centros de desarrollo en el país, haciendo que las herramientas de uso masivo sean programadas en otros lugares fuera de las fronteras nacionales y ninguna empresa local pueda enfrentar una competencia tan desigual y esto hace que se detenga o sea muy lento el desarrollo tecnológico nacional.

Creación de posibles monopolios tecnológicos.

El hecho que pocas empresas puedan tener la infraestructura de crear algunas herramientas de cómputo y que solamente una pueda mantener una herramienta en específica siempre va a resultar el problema de que se llegue a generar algún monopolio que sea el único al que se pueda acudir para adquirir alguna solución, ya está demostrado que nunca es sano para ninguna economía que existan monopolios dentro de sus fronteras, por lo mismo todos los países cuentan con leyes anti monopolios, por lo tanto siempre será una obligación de los gobiernos el estar muy atentos para evitar a toda costa la creación de alguno y mucho menos el apoyarlos de manera directa o indirecta.

CONCLUSIONES

Es muy importante que siempre tengamos en mente todos los problemas que tienen las tecnologías que estamos adquiriendo, ya que solamente tenemos en mente las ventajas que nos van a ofrecer, pero nunca debemos de olvidar que siempre todas las herramientas vienen acompañadas con sus defectos, como pueden ser los errores de programación que todo *software* tiene, hasta el día de hoy no existe ninguna herramienta informática que pueda decirse que este completamente libre de ellos, por esta causa siempre es necesario tener el dato de que tan costoso resulta el mantener esa solución y estar seguros que los problemas que nos va a solucionar van a ser superiores que los costos que tenemos que invertir por el derecho a tenerla.

Todas las grandes empresas de *software* son extranjeras y los recursos económicos con los que cuenta son amplios, por lo tanto pueden invertir en fuertes campañas publicitarias para que todos los usuarios conozcan sus soluciones y lleguen a ser las más populares, además que la mayoría de programas que producen este tipo de empresas son de código cerrado. En contra parte podremos ver que la mayoría de empresas que producen programas con alguna licencia libre, son pequeñas y medianas, o son personas individuales que comienzan con la fabricación de una herramienta y poco a poco se van uniendo más, y éstas no cuentan con los recursos necesarios para poder pagar por toda una campaña de publicidad, y la solución informática que proponen normalmente solo las personas especializadas en el tema la conocen, pero el grueso de los usuarios informáticos

normalmente piensan que solamente existe una manera de hacer las tareas y solo con las herramientas que ellos conocen.

Esto nos trae un gran problema de dependencia tecnológica, ya que la mayor parte del *software* usado no solamente en el hogar de manera personal, si no también en la industria y el gobierno es creado por un puñado de empresas, todas ellas extranjeras, y provocando esto que la industria de desarrollo de *software* este muy limitada, provocando que les cueste mucho trabajo el poder colocar las habilidades que ellas pueden aportar a la sociedad, haciendo que solamente puedan ingresar en mercados pequeños, o si le trabajan al gobierno, solo pueden obtener contratos pequeños que no les permite poder ir creciendo de una manera sana y garantizar de esta manera su estabilidad en el tiempo.

Como pudimos ver no existe ninguna herramienta informática que se encuentre libre de error alguno, o que se garantice de manera plena que por todo el tiempo que se desee esta valla a contar con el respaldo del fabricante, ya sean programas libres o de código cerrado siempre existe la posibilidad de que aparezcan errores o se le deje de dar el mantenimiento adecuado para garantizar su buen funcionamiento. Pero como las grandes empresas pueden invertir fuertes sumas de dinero en llevar a las personas responsables en tomar la decisión de que *software* es el que se va a adquirir a sus propias conferencias, cuesta gran trabajo convencerlos que tu programa es igualmente bueno.

Es necesario que por parte de alguna institución gubernamental se empiece una campaña de concienciación a los usuarios que existe todo un abanico de posibles soluciones informáticas, y que muchas de estas pueden ser de manufactura nacional y en caso de no serlo, que se busque una manera de producirla, para ir reduciendo de manera paulatina esta dependencia tecnológica desproporcionada y comenzar a producir desarrollo tecnológico nacional de alta calidad, y para poder alcanzar estas metas aquí listadas todo el *software* libre será de gran ayuda.

CAPÍTULO 3

INTRODUCCIÓN:

El estudio de las empresas es un tema muy vasto que puede llegar a ser abrumador por la cantidad de datos que se manejan de ella, y lo complicado que resulta el poder digerirlos y mostrarlos de una manera coherente y estructurada.

En este capítulo específicamente vamos a tratar el estudio de las pequeñas y medianas empresas a nivel internacional y nacional, y específicamente ahondaremos aún más en la industria de *software* en estos dos niveles. Este tema desde la simple clasificación de la pequeña y mediana empresa a nivel internacional se dificulta, ya que no existe ninguna clasificación a nivel internacional aceptada de manera general, cada país realiza su propia clasificación según sus necesidades, y en el caso específico de la unión europea han podido materializar una clasificación para ese bloque, pero si tratamos de ajustar esa clasificación para nuestro caso nacional podría llegar a resultar ridícula, ya que los niveles de ventas para los países europeos en nuestro país podrían llegar a verse como estratosféricos, pero en su realidad local están bien establecidos. También para el caso nacional, la columna vertebral de la clasificación de las empresas recae principalmente en el número de empleados que cuenta ésta, y como segundo punto toma en cuenta el nivel de ventas que alcanza anualmente, pero lo principal es su número de empleados.

Vamos a estudiar de manera detallada la industria del *software* a nivel internacional y nacional, y veremos que esta industria es una de las más dinámicas actualmente y como países de tasa media, como lo es India y Singapur han alcanzado a conglomerar una importante parte de este mercado, pero como México poco a poco a un ritmo constante esta alcanzando cada año una tasa cada vez mayor de este mercado.

¿Qué es una empresa?

Para algunas personas la palabra empresa o empresario puede sonar como una promesa, el máximo de estabilidad económica o lo más que puede alcanzar el ser humano para garantizar su supervivencia y el medio para alcanzar un nivel de vida superior, pero al mismo tiempo para otras personas esta definición puede sonar a la principal razón de las desigualdades sociales tan asentadas hoy día. Éstas palabras se escuchan constantemente pero a pesar que es tan utilizada es muy común que signifiquen cosas muy diferentes de una persona a otra.

Probablemente la raíz de este problema se pueda establecer que no existe una clara definición para la palabra empresa, el diccionario de la real academia española cuenta con varias definiciones, además que es normal que la gran mayoría de teóricos de la economía y la administración escriban su definición propia.

“(Del it. *impresa*).

1. f. Acción o tarea que entraña dificultad y cuya ejecución requiere decisión y esfuerzo.
2. f. Unidad de organización dedicada a actividades industriales, mercantiles o de prestación de servicios con fines lucrativos.
3. f. Lugar en que se realizan estas actividades.
4. f. Intento o designio de hacer algo.
5. f. Símbolo o figura que alude a lo que se intenta conseguir o denota alguna prenda de la que se hace alarde, acompañada frecuentemente de una palabra o mote.” (Real Academia Española).

También existen otras definiciones para este término que vale la pena anotar:

- Sociedad mercantil o industrial, creada y sostenida por el poder público, dato tomado de (García-Pelayo Gross, 2010).
- Unión de personas, objetos físicos y recursos económicos con el propósito de producir artículos o prestar servicios para obtener utilidades⁵⁰.
- Es una acometida, un intento organizado y complejo para alcanzar un designio, en el que se comprometen trabajo, recursos y dinero (Gaviria Cock, 2006).
- Es una entidad económica donde se combinan dinámicamente factores que son necesarios para el proceso de producción, estos factores esenciales son: el capital, el trabajo y la dirección empresarial⁵¹.
- Asociación de varios individuos para la realización de obras materiales negocios o proyectos de importancia, concurrendo comúnmente a los gastos que ofrezcan y participando todos de las ventajas que reportaren (Encyclopaedia Britannica, 1993).

Para (Huertas Méndez, 2009) las características con las que cuenta la empresa son:

- a.- Cuentan con recursos humanos, de capital, técnico y financieros.
- b.- Realizan actividades económicas referentes a la producción, distribución de bienes y servicios que satisfacen necesidades humanas.
- c.- Combinan factores de producción a través de los procesos de trabajo, de las relaciones técnicas y sociales de la producción.
- d.- Planean sus actividades de acuerdo a los objetivos que desean alcanzar.
- e.- Son una organización social muy importante que forman parte del ambiente económico y social de un país.
- f.- Son un instrumento importante del proceso de crecimiento de desarrollo económico y social.
- g.- Para sobrevivir debe de competir con otras empresas, lo que exige: modernización, racionalización y programación.
- h.- El modelo de desarrollo empresarial reposa sobre las nociones riesgo, beneficio y mercado.

⁵⁰ Nacional Financiera. La empresa y el empresario. México. s/n.

⁵¹ Programa educativo visual. Enciclopedia educativa. Tomo 9. México.

i.- Son el lugar donde se desarrollan y combinan el capital y el trabajo, mediante la administración e integración de la organización.

Me parece que como última definición de empresa y la que se utilizará para este trabajo de investigación es la que nos proporciona Cuauhtémoc (2002) en su tesis de maestría.

“Las empresas son sociedades que llevan a cabo acciones concretas, ordenadas y reguladas por las leyes que ponen en marcha a grupos multidisciplinarios de individuos tanto para dar respuesta a sus necesidades particulares como a las comunidades en las que habitan” (Hernández Suárez, 2002).

Propósito de la empresa

Dentro de la economía global actual la actividad empresarial es la que genera la riqueza de los países, y a esta riqueza la denominan valor agregado. El valor agregado es la diferencia entre el valor monetario de producción empresarial y el valor monetario de los bienes y servicios adquiridos a otras unidades productivas. La empresa consigue este valor agregado transformando las materias primas mediante el trabajo de las personas y el capital productivo que tenga disponible, de manera que su finalidad es la obtención de un producto final de mayor valor, el cuál sirve para cubrir las necesidades de los consumidores finales. Esto da lugar a lo que se conoce como flujo circular de la renta de un país.

Características de la empresa.

Ante nuestra realidad actual, es fundamental que tengamos presente en todo momento la importancia de las pymes no solamente a nivel local, como por ejemplo país si no más bien a nivel mundial. A fin de contar con una clara imagen de cual es nuestro objeto de estudio y de su importancia en el crecimiento económico vamos a estudiar las pequeñas y medianas empresas (pyme o sme en inglés) en el mundo y de manera local, al mismo tiempo vamos a tratar de identificar sus características.

El emprendedor.

Algo que comparten todas las definiciones anteriores es que siempre el concepto de empresa está ligado al de llevar a cabo una acción, por lo tanto la persona que empieza a realizar las acciones se le puede denominar como emprendedor, ya que esta persona fue el que emprendió la realización de la idea que se tenía en mente, no necesariamente el tuvo que ser la primera persona en imaginarla pero sí fue el primero en hacer los arreglos para llevarla a cabo.

Si es que llegara a faltar la figura del emprendedor, entonces todo queda como una simple idea que nadie se atreve a tratar de llevarla a cabo, por lo tanto la figura del emprendedor es fundamental para que pueda nacer cualquier empresa. Los preparativos para llevar la idea a cabo pueden ser tan grandes o

pequeños como la idea misma, a estos preparativos normalmente se le conoce como proyecto, donde su finalidad es dar un orden lógico de los pasos a seguir para llevar a cabo la idea y puede ser tan simple o complejo como la idea misma y la manera de trabajar del emprendedor, por lo tanto, la figura del empresario aparece como una pieza básica pues es el elemento conciliador de los distintos intereses.

Otro punto que también tenemos que poner en consideración esta que nadie puede realizar todas las tareas que son necesarias para llevar a buen puerto una idea, siempre es necesario el poder contar con el apoyo de otras personas que cuenten con mayores conocimientos en temas mas específicos, como pueden ser temas contables y de hacienda, por lo tanto el emprendedor siempre tendrá que rodearse de un equipo de su confianza para poder asegurarse que todas las tareas serán realizadas de manera satisfactoria.

Hasta aquí estamos hablando de la empresa tomando en cuenta sus funciones sociales, la cuál es un buen punto de partida ya que todas las empresas son fundadas para satisfacer una necesidad específica de la sociedad en la que habita, y va evolucionando conforme esta misma también lo valla haciendo y finalmente muere si es que la labor que desempeña o el producto que comercializa ya no cubren ninguna necesidad.

Para el caso particular de este trabajo de investigación, la definición de empresa tiene principalmente el fin de conceptuar y delimitar el tipo de empresa que se va a estudiar, para determinar cuál de todos los tipos de empresa será estudiado.

Clasificación de las empresas

Las empresas producen de acuerdo a su tamaño o escala de la planta y en consecuencia, la participación en el mercado está condicionada a su volumen de producción. Existen varias maneras de clasificar a las empresas, no existe un índice único que caracterice la dimensión de la empresa de manera adecuada, diferentes organismos internacionales así como también secretarías estatales de cada país se han encargado de crear sus propias clasificaciones, lo cual dificulta de manera considerable su estudio. Suelen ser un extenso espectro de variables los utilizados para su clasificación, como pueden ser:

- 1.- El número de trabajadores que emplean.
- 2.- Tipo de producto.
- 3.- Tamaño de mercado.
- 4.- Inversión en bienes de producción por persona ocupada.
- 5.- El volumen de producción o de ventas.
- 6.- Valor de producción o de ventas.
- 7.- Trabajo personal de socios o directores.
- 8.- Separación de funciones básicas de producción, personal, financieras y ventas dentro de la empresa.
- 9.- Ubicación o localización.

- 10.- Nivel de tecnología de producción.
 - 11.- Orientación de mercados.
 - 12.- El valor del capital invertido.
 - 13.- El consumo de energía.
- Entre otras más.

Cada país define a las pymes usando pautas esencialmente cuantitativas que si bien tiene ciertas características similares, también tienen diferencias significativas. Incluso en algunos casos puede ser que no exista consenso alguno dentro de un mismo país. En la mayoría de los casos, las variables consideradas son cantidad de personal, cifras de ventas y niveles de inversión en activos. Los organismos internacionales con el fin de canalizar los apoyos financieros y logísticos, también se han dado a la tarea de establecer clasificaciones para este sector, como por ejemplo la OCDE (Organización para la Cooperación y Desarrollo Económico) clasifica a las empresas como lo muestra la tabla 3.1 de este capítulo.

Tabla 3.1 Clasificación de las empresas de acuerdo con la OCDE.

Criterio	Muy Pequeñas	Pequeñas	Medianas	Grandes
No. de trabajadores	0 a 9	10 a 49	50 a 249	250 en adelante
Ventas Anuales (euros)	Menor a 2,000,000	Menor a 10,000,000	Menor a 50,000,000	Mayor a 50,000,000

Fuente: Tabla elaboración propia tomando datos de la (Organización para la Cooperación y Desarrollo Económico, 2005).

Otro ejemplo es la Comisión de las Comunidades Europeas que publicó el 20 de mayo del 2003 en el Diario Oficial de la Unión Europea una recomendación para facilitar la clasificación de las pymes, la cuál podemos observar en la tabla número 2.

Tabla 3.2 Clasificaciones de las empresas propuesta por la CCE.

	Empleados	Ventas	Activo
Microempresas	0 - 9	0 - 2,000,000 €	0 - 2,000,000 €
Pequeña	0 - 49	Hasta 10,000,000 €	Hasta 10,000,000 €
Mediana	0 - 249	Hasta 50,000,000 €	Hasta 43,000,000 €

Fuente: Tabla elaboración propia tomando datos del Diario Oficial de la Unión Europea.

En México las empresas han sido clasificadas de acuerdo a su tipo, número de empleados y volumen de ventas, y la secretaria encargada de realizar esta tarea es la Secretaria de Economía que a través del Diario Oficial de la Federación el 30 de junio del 2009 publicó una tabla mostrando la clasificación vigente hasta

el día de hoy.

Tabla 3.3 Clasificación de la micro, pequeña y mediana empresa vigentes a partir del 30 de junio del 2009 (Barragán, 2011) en México.

Tamaño	Sector	Número de trabajadores	Monto de ventas anuales (mdp)	Tope máximo combinado*
Micro	Todas	Hasta 10	Hasta \$4	4.6
Pequeña	Comercio	Desde 11 hasta 30	Desde \$4.01 hasta \$100	93
	Industria y Servicios	Desde 11 hasta 50	Desde \$4.01 hasta \$100	95
Mediana	Comercio	Desde 31 hasta 100	Desde \$100.01 hasta \$250	235
	Servicios	Desde 51 hasta 100		
	Industria	Desde 51 hasta 250	Desde \$100.01 hasta \$250	250

Fuente: Elaboración propia con datos del Diario Oficial de la Federación.

Un dato muy importante es que en México existen aproximadamente 2.9 millones de establecimientos, de los cuales 2.85 millones pertenecen a las micro, pequeñas y medianas empresas (99%). Por lo tanto estas empresas constituyen por sí mismas un sector estratégico para el bienestar del país, ya que ellas crean el 64% de los empleos y contribuyen con el 40% del PIB (Barragán, 2011) (Producto Interno Bruto) del país.

Pequeñas y medianas empresas.

Dentro del entorno globalizado al que se enfrentan hoy día todas las organizaciones del mundo, la lucha por sobrevivir en el mercado es algo que se tiene que tener en mente todos los días, en especial para las micro, pequeña y mediana empresa las cuales necesitarán adecuar sus estrategias competitivas para poder encarar de la mejor manera esta lucha. Para lo cual será necesario adecuar sus estrategias competitivas, de recursos humanos, tecnológicos, organizativos, productos o servicios que esta ofrezca para lograr alcanzar una estabilidad, o en el mejor de los casos superar su estabilidad para buscar crecer en el entorno actual tan dinámico y complejo.

La pyme en el mundo.

Como ya lo hemos explicado en párrafos anteriores puede llegar a resultar una tarea compleja el tratar de hacer un estudio de las pymes de manera global, ya que no existe ningún consenso con respecto a su clasificación, ni siquiera los organismos internacionales que trabajan al respecto han podido llegar a una clasificación en común, pero uno de los criterios que cuenta con mayor aceptación es el instrumento de clasificación propuesto por la OCDE.

En los países miembros de la OCDE un estudio realizado en el año del 2007 se ha establecido que el 95% de los negocios establecidos son micro, pequeñas o medianas empresas. Un dato muy específico de este estudio es que el 99% de los negocios son pymes, otro punto en particular de este estudio es que en México el 90.86% de los negocios dedicados a la manufactura son micro empresas, en la República Checa representan el 90.17%, en Italia es el 82.6%, en la tabla número 4 se puede observar de manera más sencilla algunos de los datos reportados por el informe.

Tabla 3.4 Porcentaje de empresas por tamaño en el sector de manufactura.

Número de empleados	1 a 9	10 a 49	50 a 249	Más de 249
México	90.86	6.01	2.2	0.93
República Checa	90.17	7.13	2.14	0.57
Hungría	86.24	10.18	2.82	0.75
Francia	82.84	13.17	3.15	0.84
Italia	82.63	15.13	1.97	0.27
Portugal	79.77	16.28	3.5	0.44
España	78.03	18.68	2.83	0.46
Holanda	74.49	19.08	5.29	1.13
Austria	72.54	20.63	5.31	1.52
Dinamarca	71.67	20.92	6	1.42
Reino Unido	71.27	21.45	5.91	1.37
Alemania	60.21	29.69	8.03	2.07

Fuente: Tabla realización propia con datos de la OECD (OCDE., 2007).

La importancia de las pymes varían dependiendo del sector, normalmente en el sector servicios cobran mayor importancia, por ejemplo en los servicios de computación, actividades relacionadas y renta de equipo dentro de los miembros de la OCDE más del 90% (OCDE., 2007) son micro empresas. En los últimos años el número de pymes ha crecido de manera muy dinámica incluso más que la

población (OCDE., 2007).

Este tipo de empresas emplean entre el 60% y el 70% de la población económicamente activa y en algunos países son las generadoras de las dos terceras partes de los empleos privados (OCDE., 2007). Estas son solo algunas de las razones por las que este tipo de empresas son consideradas un motor del crecimiento económico, fuente de dinamismo, innovación y productividad.

Existe un gran dinamismo en el surgimiento de nuevas empresas de pequeña dimensión principalmente en los servicios, desgraciadamente estas no sobreviven por largo tiempo. Para el caso de los países miembros de la OCDE aproximadamente el 20% de las nuevas empresas cierran antes de cumplir el primer año, y este porcentaje se agudiza de manera significativa entre los países clasificados como en vía de desarrollo. En América Latina, las actividades de las empresas de pequeña dimensión están asociadas generalmente a negocios familiares y de subsistencia, lo cual limita considerablemente su expansión.

Dentro de las principales características que presentan las nuevas empresas una que destaca y es particularmente interesante es su capacidad a innovar, lo cual a facilitado su internacionalización, adicionalmente de las tradicionales empresas de importación y exportación. Las empresas de pequeña dimensión están apostando más por establecer alianzas estratégicas, fusiones y adquisiciones, así como redes de colaboración interempresarial.

Las nuevas empresas con base tecnológica tienen la capacidad de renovar la tecnología, cerrando brechas tecnológicas y presionando a las grandes empresas a elevar de manera constante su competitividad en su mercado, la OCDE calcula que entre el 30% y el 60% (OCDE., 2007) de las pymes manufactureras son de naturaleza innovadora, de las cuales tienden a ser más innovadoras las ubicadas en servicios intensivos en conocimiento que las de negocios o intermediación financiera.

Para que este tipo de empresas puedan crecer y contribuir de manera significativa al desarrollo económico, se debe de reducir la sobre regulación que normalmente pesa sobre de ellas, así como la carga administrativa que limitan su capacidad emprendedora e innovadora, por lo que una práctica sumamente beneficiosa que deben de buscar todos los gobiernos es focalizar numerosos esfuerzos para la educación de su pueblo y fomentar el espíritu emprendedor en las mismas aulas de todas las instituciones de educación superior con las que cuenten. Al mismo tiempo deben de tener un sencillo acceso al financiamiento, a la tecnología y mercados internacionales para que así las innovaciones que estas produzcan generen los beneficios económicos deseados y puedan desarrollarse de manera sana.

Por lo que respecta al grado de integración empresarial, las pymes han aumentado su participación en redes empresariales, clusters y alianzas, con el fin de aumentar la competencia en los mercados, a través de la reducción de los

costos de operación y transacción, el crecimiento de su productividad y valor agregado, así como la expansión de sus mercados gracias al escalamiento productivo y la incorporación de tecnología en sus procesos. A pesar de su importancia económica y dinamismo, aún no son aprovechadas en toda su capacidad, ya que cuentan con el potencial para aumentar el nivel de vida de la población, reduciendo la pobreza y aumentando el nivel de vida de sus colaboradores, es por estos motivos que los gobiernos locales deben de prestar especial énfasis en encontrar las maneras adecuadas para que estas puedan nacer y desarrollarse de manera sana apoyándose en políticas adecuadas.

Las pymes en México.

En la actualidad las pymes tienen un papel fundamental no solamente en México si no en el mundo, es uno de los motores más importantes para el sano desarrollo de toda economía y constantemente se esta buscando el desarrollo de leyes y políticas para apoyar el nacimiento de estas, además que cuentan con el potencial de incorporarse a sectores de alto contenido tecnológico en los procesos de sectores tradicionales, apoyando en la modernización y escalamiento productivo de la planta industrial. Asimismo debido a su alto impacto en la creación de empleos y poder de arrastre para ayudar a las zonas donde se encuentran ubicadas, constituyen uno de los principales medios para combatir la pobreza y el desempleo.

La mayoría de las pymes no llegan a desarrollarse adecuadamente debido a la falta de una adecuada “cultura empresarial” del emprendedor mexicano, y si a esto se le suma que el ambiente político y económico también se encuentra con una enorme incertidumbre en los últimos años que se han visto influenciados por cuestiones geopolíticas externas, que producen una gran cantidad de consecuencias que afectan directamente su futuro. Un problema también que causa mucho conflicto en las pymes nacionales es la enorme competencia desleal que se produce a través de la importación de productos ilegales de mercados internacionales, la llamada “piratería”, esto produce una enorme cantidad en perdidas para los negocios establecidos y en el peor de los casos, algunos terminan cerrando sus establecimientos.

Algunas de la características que podemos detallar de las pymes mexicanas están:

- Son empresas familiares.
- La falta de recursos económicos les limita su crecimiento, ya que no existe una fuente sencilla de financiamiento.
- Existen pocas o nulas posibilidades de fusionarse con otras empresas para así poder brindar un abanico de opciones más amplio a sus clientes.
- Debido a la inexperiencia administrativa del dueño, éste dedica un número excesivo de horas al trabajo de escritorio y su rendimiento suele

- ser muy bajo.
- La administración no es especializada, es completamente empírica y por lo general la llevan a cabo los propios dueños.
- Las ganancias que obtienen suelen ser muy bajas, y en algunos casos los dueños renuncian al esfuerzo si no alcanzan sus metas esperadas de manera prematura.
- No contratan personal especializado o capacitado por no poder pagar los altos salarios de estos.

Pero no todas características son necesariamente negativas, entre de las positivas podemos citar:

- Asimilan y adaptan con facilidad tecnologías de diversos tipos.
- Con facilidad se adaptan al tamaño del mercado, aumentando o reduciendo su oferta cuando es necesario.
- Debido al tamaño de la empresa el dueño que comúnmente suele ser el gerente conoce bien a sus empleados, facilitando de esta manera la resolución de manera veloz los conflictos que pudieran llegar a aparecer.
- Los dueños suelen tener un amplio conocimiento en el área operativa de la empresa, permitiendo de esta manera aplicar todo su ingenio y experiencia en la solución de problemas que pudiera llegar a presentarse.
- Cuentan con un gran potencial de crecimiento para alcanzar a ser una mediana o gran empresa.
- Se establecen en diversas regiones geográficas, lo cual les permite contribuir al desarrollo regional
- Aunque sea muy redundante en este punto, son generadoras de empleo, atrayendo de esta forma, parte importante de la población económicamente activa.

Con los datos que tenemos aquí listados podemos colocarlos en una tabla para facilitar su estudio, estos datos que colocaremos son los listados por Morales (Morales Castro, 2004), donde nos enumera los conflictos que enfrenta la pyme mexicana.

Tabla 3.5 Conflictos que aquejan a la pyme mexicana.

PROBLEMA	ÁREA QUE AFECTA
1.- Inexperiencia del dirigente. - Sistema de información inadecuado. - Baja capitalización	De lanzamiento.
2.- Aumento de las inversiones ligado al crecimiento. - Endeudamiento exagerado causado por una baja capitalización. - Mala planificación de la tesorería.	De liquidez.
3.- Concentración de la información y del poder. - Falta de personal de nivel intermedio. - Aumento de personal. - Diversificación de las operaciones.	De delegación.

4.- El administrador general percibe a la empresa como una extensión de sí mismo. - El administrador general no escucha a nadie. - El administrador general no controla lo que ocurre	De liderazgo.
5.- Expansión rápida. - Límite de crédito de los proveedores. - Límites de crédito bancarios.	De financiamiento.
6.- Complacencia. - Relajamiento. - Dividendos excesivos.	De prosperidad.
7.- Divergencias entre los accionistas. - Salida del administrador general. - Impuesto de sucesión. - Conflictos excesivos entre el equipo dirigente.	De continuidad.

Fuente: Tabla elaboración propia con datos de Morales (Morales Castro, 2004).

Los puntos con los que debería de contar para que una pyme tenga una cultura laboral adecuada según el propio Morales son:

- Capacitación continua.
- Información suficiente del medio y socio político actual tanto nacional como internacional.
- Conocimiento de la tecnología de punta en el giro que se desenvuelve la empresa.
- Conocimiento del mercado (sus necesidades, satisfactorias, conductas, condiciones sociales y económicas, etc).
- Sistemas de calidad.

Como ya se mencionó anteriormente las micro, pequeñas y medianas empresas juegan un papel sumamente importante en la economía nacional de cualquier país, pero desgraciadamente la mayoría de éstas no llega a desarrollarse sanamente, debido a una falta de cultura empresarial correcta del emprendedor mexicano, aunado al ambiente político y económico adverso, ya no solamente nacional si no internacional, que durante los últimos años se ha visto influenciado por un gran número de variables que afectan su devenir como por ejemplo, que las empresas mexicanas actualmente tienen que enfrentar a una competencia internacional cada vez más capacitada.

En nuestro país aunque se ha tratado de impulsar a las pymes a través de financiamientos que otorga el gobierno federal a través de la banca de desarrollo, cobrando tasas relativamente bajas pero con requisitos muy altos para la mayoría de estas organizaciones, quedando fuera de todo modo de financiamiento institucionalizado, por otro lado los financiamientos que ofrece la banca comercial son a tasas muy elevadas y esto resulta un gran obstáculo para el crecimiento de estas empresas.

Un punto más a tomarse en consideración es que las tecnologías que necesitan las pymes en la mayoría de las ocasiones son desarrollos extranjeros, provocando que los costos de adquisición de estas sean elevados, y que en ocasiones se necesita de personal humano altamente capacitado para poder obtener una buena utilización de las herramientas, este es otro tropiezo con el que debe lidiar la pyme.

Por lo general cualquier organización que no este bien estructurada y que no tenga claras líneas de mando tiende en poco tiempo a desaparecer, provocando de esta manera que el problema de desempleo en el país se agrave, porque las pymes son la principal fuente de empleo no solamente a nivel nacional, si no mundial.

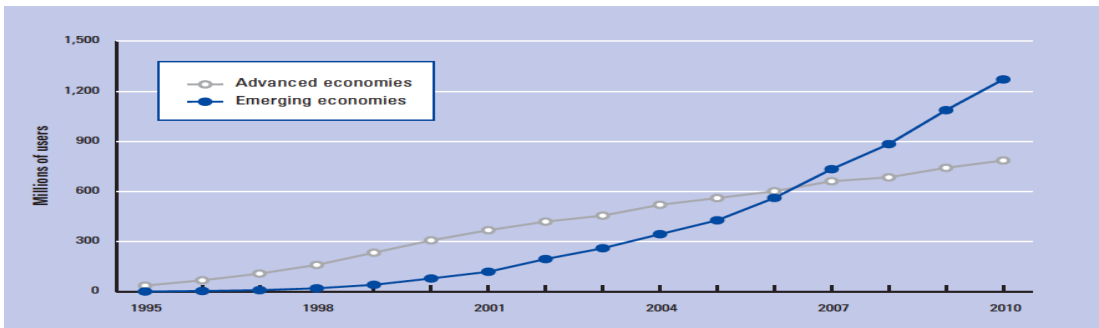
Sin lugar a duda, la revitalización de la economía nacional depende de las empresas, de la creatividad, el ingenio y el compromiso que se este gestando dentro de cada una de ellas, del esfuerzo que este realizando cada uno de los empresarios para crear programas propios de acuerdo a una planeación que este en forma para la correcta función de su empresa.

Penetración del internet

Finalizando la primera década del siglo 21, queda claro que las redes de alta velocidad deben ser una parte integral de la infraestructura básica de cualquier país. De hecho estas constituyen los cimientos de la economía del conocimiento. No solo en las economías más avanzadas la conectividad fortalece la productividad, genera empleos, mejora los cuidados de la salud y el nivel académico y cultural de la población.

El conocimiento y uso de Internet es uno de los aspectos más usados para conocer en que estado se encuentra la conectividad e infraestructura de redes. A causa del rápido crecimiento de la población, los marcadores obtenidos dan una valoración sobre el uso de la gran red por parte de los ciudadanos a nivel mundial (Gráfico 3.1). Como se observa, desde la mitad de la década de los noventa los países más desarrollados tuvieron un incremento muy marcado, pero a partir de los años 2004 y 2005 este incremento cada vez es más pausado, en comparación en el año 2000 el incremento de los usuarios en los países menos desarrollados se comenzó a dar, y a partir del 2005 el incremento es aún más pronunciado y continua hasta el 2010 esta tendencia.

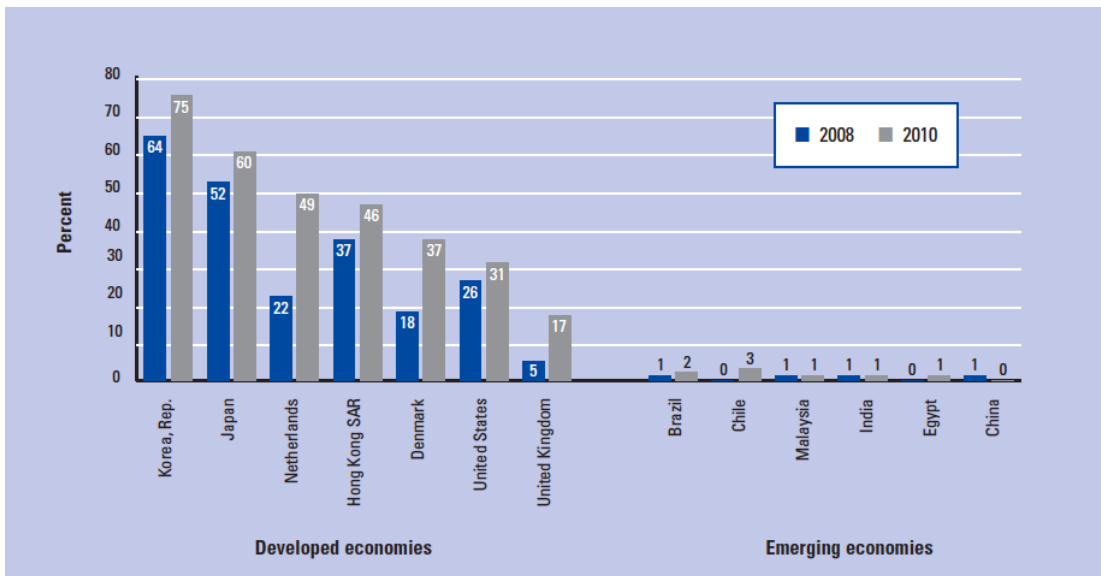
Gráfica 3.1 Usuarios de internet 1995-2010



Fuente: Gráfica tomada de Foro Económico Mundial (World Economic Forum., 2011)⁵².

Y ahora sí estudiamos la gráfica de las conexiones superiores a 5 megas observamos que el país con mayor número de estas es Corea del sur, donde el 75% aproximadamente de todas sus conexiones cuentan con esta velocidad, seguido por Japón, Holanda, Hong Kong, Dinamarca, EEUU, y Reino Unido. Y la lista de los países emergentes, Brasil ocupa el primer sitio con 2%, después Chile, Malasia, India, Egipto y finalmente China, y hablando particularmente de China, es el único país que aparece en la gráfica donde sus conexiones superiores a 5 megas del año 2008 al 2010 se redujeron.

Gráfica 3.2: Conexiones a internet superiores a 5 megas/s.



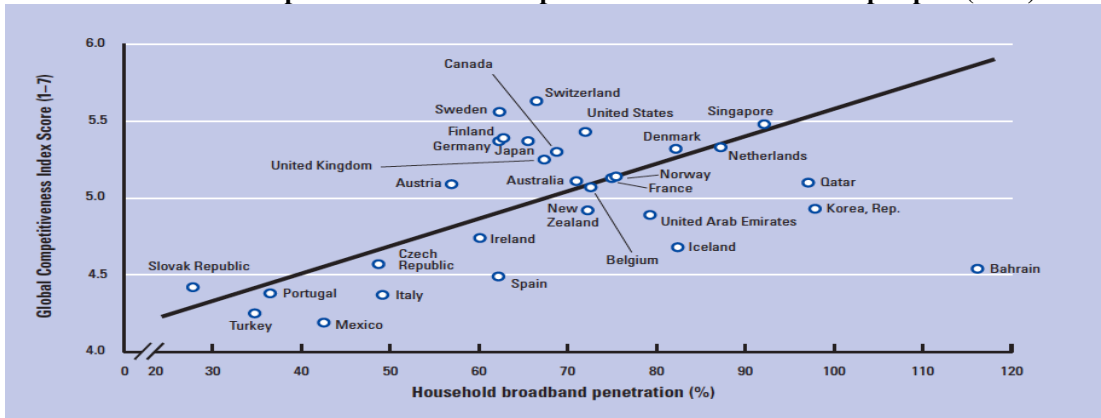
Fuente: Foro Económico Mundial p51 (World Economic Forum., 2011).

Y pasando a la gráfica 3.2 nos muestra la comparación de los países con respecto a la penetración del internet en los hogares, y el nivel de competitividad que le asignó esta misma institución a cada uno de ellos. Por lo tanto mientras más arriba y a la derecha aparezca marcado un país es mejor, ya que mientras

⁵² World Economic Forum. (2011). *The Global Competitiveness Report 2011-2012*. Geneva, Switzerland: Autor.

más alto sea nos indica que tiene un nivel superior de competitividad y mientras más a la derecha este nos dice que el nivel de penetración de internet es mayor.

Gráfica 3.3: Competitividad vs nivel de penetración de banda ancha por país (2010).



Fuente: Foro Económico Mundial p50 (World Economic Forum., 2011).

La gráfica nos muestra que el país con mayor penetración de ancho de banda en los hogares es el Reino de Baréin, que se encuentra en el golfo Pérsico de Asia, después sigue Corea del sur y Qatar. Pero el país que aparece más arriba y a la derecha es Singapur, y nuevamente podemos observar que estos países tanto su área geográfica como su población son pequeños, y nos ayudan a deducir que para poder alcanzar una penetración adecuada de internet en los hogares, se facilita mucho más cuando nos encontramos con poblaciones relativamente pequeñas y no separadas.

Los siguiente países que se encuentran en esa situación son Noruega, Dinamarca, Holanda, Francia, todos países europeos. Y el seguimiento de este escenario nos concluye que son los países más desarrollados los que podemos ver en los primeros lugares.

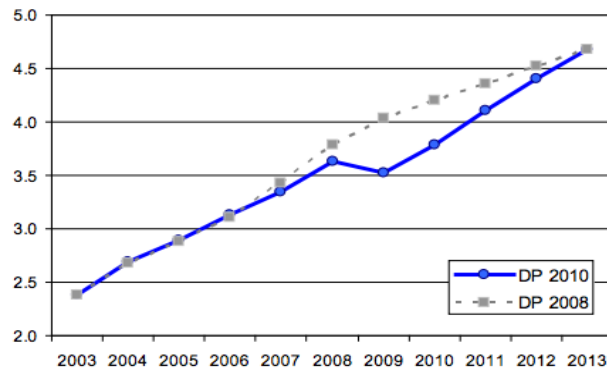
La industria desarrolladora de software en el mundo.

El comportamiento de la industria del *software* en el mundo queda principalmente pautado por los países que más invierten en tecnologías de la información en el mundo, donde los EEUU. tiene un lugar preponderante al respecto, además de ser el país que más demanda genera en este rubro (World Economic Forum., 2011).

Pero como los EEUU están recién saliendo de una crisis económica desatada por el mercado mobiliario en el 2008, en la gráfica 3.4 podemos ver que en el año 2009 la industria presentó una caída aproximada del 3% (WITSA, 2010) en los gastos mundiales de TIC, y actualmente la Unión Europea está inmersa en la crisis más profunda de los últimos años desatada por el enorme déficit de Grecia, esto provocará que toda la industria a nivel global se vea afectada por este vector que aún a principios del 2012, al menos en el caso europeo, los principales

analistas no tomaban en cuenta. Según los datos de (WITSA, 2010), se espera que para el año 2012 hubiera un crecimiento en el mercado del *software* del 7.3% (WITSA, 2010)⁵³. Y en la misma gráfica 3.4 observamos que para el año 2013 sea cuando se puedan volver a retomar el impulso calculado en el 2008.

Gráfica 3.4: Gastos globales de TIC, las cantidades están en billones de dólares estadounidenses.



Fuente: Gráfica tomada de (WITSA, 2010)⁵⁴

En la tabla podemos observar que siguen siendo países desarrollados los que llevan la batuta a nivel mundial con el punto de la absorción de nuevas tecnologías en los negocios, pero también podemos ver que dentro de los primeros 10 de 138 países en total, no están los EEUU, aparecen cuatro países de Europa del norte, que de hecho Islandia tiene el primer sitio, tres países de medio oriente, y tres países asiáticos, el caso de Taiwán es algo especial ya que ellos se quieren representar como un país autónomo pero el gobierno de China no se los permite.

Tabla 3.6 Medida en que las empresas absorben nuevas tecnologías, la calificación máxima es de 7.

Ranking	País	Valor
1	Islandia	6.49
2	Suecia	6.43
3	Japón	6.25
4	Suiza	6.25
5	Arabia Saudita	6.23
6	Noruega	6.2
7	Israel	6.13
8	Qatar	6.11
9	Corea del sur	6.07
10	Taiwán	6.07
24	Puerto Rico	5.62
37	Chile	5.35

⁵³ WITSA. (2010). *Digital Planet Executive Summary*. Taipéi, Taiwán. P. 14.

⁵⁴ WITSA. (2010). *Digital Planet Executive Summary*. Taipéi, Taiwán. P. 14.

41	Panamá	5.31
44	Guatemala	5.24
46	Brasil	5.18
50	Costa Rica	5.14
86	México	4.51

Fuente: Tabla creación propia con datos del Foro Económico Mundial (World Economic Forum., 2011).

También podemos observar que en el caso Latinoamericano el primer puesto le pertenece a Puerto Rico, y los primeros lugares de la región excepto por Chile, Brasil y México son países pequeños, si comparamos sus poblaciones totales no igualan a las personas que vivimos dentro de la zona metropolitana de la Ciudad de México, probablemente este sea un factor beneficioso para poder aumentar el nivel de absorción de nuevas tecnologías, las poblaciones pequeñas, pero Chile es un país más grande geográficamente que México e igualmente Brasil, y ambos tienen una proporción mayor en cuanto a la absorción de estas.

La India ha venido emergiendo como una gran potencia en la producción de *software* en los últimos tiempos. Ya en 2005, por conceptos de subcontratación de servicios y exportaciones de la industria del *software* y la información, el monto fue de 17 mil 200 millones de dólares, cifra que podría elevarse a 60 mil millones anuales para 2012 según pronósticos de Nasscom y McKinsey citados en el estudio creado por Hernández Vismar (Hernández, 2009).

Cada nación realiza importantes acciones en torno a las nuevas tecnologías debido a que esta es la industria que marca y determina las pautas para el desarrollo futuro. Las principales economías marchan a la vanguardia en tal sentido, agrandando cada vez más la brecha tecnológica existente y consumiendo las posibilidades de los países menos favorecidos. No solo la gran diferencia económica determina la diferencia tecnológica, políticas internacionales como la explotación de mano de obra barata y el robo de profesionales altamente capacitados es aún un fenómeno cotidiano que atenta contra las naciones de los países tercermundistas o de economías emergentes que puján fuertemente por desarrollar soluciones y alternativas propias. Las grandes empresas que consolidan el poder en las Nuevas Tecnologías se expanden cada vez más, ocupando nichos de mercado que aún quedan disponibles y con sus fuertes inversiones en investigación y desarrollo dejan en posición cada vez menos ventajosa a naciones que tratan de posicionarse en esta industria.

Como ya se mencionó anteriormente en el año del 2009 se presentó una caída del 3% en toda la industria de TIC, y específicamente en caso del *software* estas alcanzaron un nivel de ventas de \$300 mil millones USD (WITSA, 2010) mundiales, y se estima que para este año las ventas por software sean de 400 mil millones USD (WITSA, 2010).

La industria del software en México.

El gobierno de Fox captó desde el punto de vista empresarial la oportunidad, pero también la necesidad que tenía el país de aprovechar avances previos e impulsar esta industria. Reconociendo el fuerte rezago en comparación con Brasil e incluso Argentina, el Programa para el desarrollo de la industria del software (Prosoft) propuso una estrategia y metas explícitas a alcanzar más allá del sexenio. Esos seis años son singulares para México porque representan la culminación del auge exportador basado en el hardware, al que sigue la primera crisis global de la nueva economía (de las telecomunicaciones e Internet de 2001-2002), complementada con la implacable competencia China. En la reestructuración que se da a comienzos de la primera década del 2000 se advierte la importancia de la producción y servicios de *software* para lograr mayor competitividad primordialmente frente a China.

Otro aspecto es que la parte mayoritaria del mercado, esto es, 94% (Mochi Alemán & Rivera Ríos, 2006) del *software* empaquetado se satisface por medio de importaciones. La empresa doméstica se centra en el *software* a la medida que representa un sexto del tamaño del mercado de productos (Mochi Alemán & Rivera Ríos, 2006). La principal limitante para un mayor desarrollo de este subsector es que los grandes usuarios (empresas y gobierno) han mantenido básicamente el modelo in-house, es decir, producen internamente su propio *software*. Mochi (Mochi Alemán P. , 2006) estima que el total de facturación del *software* in-house es de unos 750 millones de dólares es casi igual al tamaño del mercado en 2002.

La industria doméstica está fragmentada territorialmente y predominan las pequeñas empresas con capacidades limitadas para ampliar la gama de productos y ofrecer soluciones avanzadas. A principios del 2000 había unas 390 empresas registradas, de las cuales la mitad tenían menos de 15 empleados

La calificación de los cuadros de ingenieros y técnicos es baja y constituye un cuello de botella para una rápida expansión de la planta empresarial. Las exportaciones son modestas (se estiman en unos 120 millones en la categoría de empaquetado para 2002), aunque las empresas extranjeras están comenzando a exportar activamente a países con los que México tiene tratados comerciales. En relación a la infraestructura de telecomunicaciones se ha expandido, pero en el uso de las TIC el país ocupa el lugar 47 y el porcentaje del PIB que dedica a ese sector es la mitad de la media mundial (Mochi Alemán & Rivera Ríos, 2006).

EL PROGRAMA PROSOFT.

En el año 2001 el gobierno mexicano declaró la industria del software como un sector estratégico del país, con un área potencial para el desarrollo económico sustentable del país.

En el 2001, tomando en cuenta el mercado interno de México el secretario de economía dentro del Plan Nacional de Desarrollo 2001 – 2006 presentó este

sector como estratégico para el país, y el plan se llamó PROSOFT (programa de desarrollo de software), donde los actores principales son la industria privada y el académico, el industrial y el de servicios y por último el gobierno federal y municipal, las principales metas que se establecieron fueron:

- Aumentar el nivel promedio de inversión en tecnologías de la comunicación del PIB nacional, para alcanzar el promedio de los países miembros de la OCDE.
- Alcanzar una producción anual de \$5000 millones USD anualmente para el año 2010.
- Convertir a México como el líder Latinoamericano en el sector de desarrollo de servicios tecnológicos.

Lanzado en octubre del 2002, el propósito de PROSOFT es aumentar la competitividad del sector tecnológico mediante su fortalecimiento. Inicialmente estaba enfocado en la industria del software, pero desde el 2004 comenzó a cubrir todo lo referente a tecnología de la comunicación.

La parte medular del programa y que tiene una base instrumental clara radica en transferir el mercado cautivo en manos del sector público a las empresas privadas. Complementando lo anterior, se propone la creación de polos de desarrollo apoyados en parques tecnológicos, en los que se inserten las instituciones educativas. Los polos tecnológicos estarían estructurados en torno a empresas extranjeras. El compromiso explícito del gobierno consiste en conceder exenciones tributarias para que se instalen las empresas extranjeras y financiar la infraestructura de comunicaciones, además de promover el enlace con las instituciones educativas.

El gobierno federal identificó correctamente las condiciones necesarias para constituir la industria, pero no asumió el liderazgo de la estrategia y tampoco se comprometió a crear las precondiciones para su desarrollo (transformar al sector educativo y no simplemente reorientarlo, reestructurar la operación del sistema bancario-financiero en su conjunto y asumir directamente el desarrollo masivo de la infraestructura).

Por ejemplo, al proponer transferir la reserva del sector público al sector privado le confiere poderes discrecionales a la Secretaría de Hacienda para determinar la magnitud y ritmo de esa transferencia. Siendo una entidad dominada por cuadros contrarios a la política industrial, es lógico suponer que no le otorgarán prioridad al subprograma, que requiere un fuerte compromiso para coordinar las acciones a varios niveles.

Pero a pesar de los diferentes problemas que aquí están enlistados podemos determinar que este programa de PROSOFT sí a dado resultados positivos un particularmente se encuentra el parque tecnológico de la ciudad de Guadalajara ITESO, donde están establecidas firmas tan importantes como LG, Samsung, Flextronics, Elcoqy Hon Hai, que en el año de 2008 exportó \$12, 000

millones USD y en el 2009 creció 5%⁵⁵ más, dato muy importante ya que en ese año, la economía nacional presentó una caída del 8% del PIB. También empresas como Softtek, empresa creada en Monterrey en 1982, y que en el año 2007 adquirió a la empresa I.T. UNITED, dedicada al desarrollo de software en China, es una de las empresas líderes mexicanas en cuanto al tema de tecnología de la información, se ha ayudado de este programa, y en una entrevista a Fabio Bittencourt director general de esta empresa, resalta su preocupación a la disminución del presupuesto del programa para el año 2010, ya que estaba calculado una partida de \$500 millones MXN, y solamente se estaban concediendo \$388 millones MXN, este ejecutivo comentaba que los empleos producidos por la industria cayeron hasta un 50%⁵⁶ debido a esto.

México por su cercanía a los EEUU, que es el mercado más grande, tiene una ventaja única que ningún otro país que ofrezca servicios de outsourcing informático como lo son India y China puede contar, manejamos los mismo usos horarios, los directivos de las empresas pueden llegar a las plantas mexicanas en un vuelo de unas cuantas horas, y las culturas son más parecidas entre México y EEUU que con respecto a los otros países aquí citados.

CONCLUSIONES

Como se ha estudiado detalladamente en este capítulo las pymes forman parte medular en todas las economías, donde se acentúa una pequeña empresa exitosa se comienzan a gestar empleos y la salud económica de la zona aumenta indudablemente, si esta empresa sigue haciendo bien las cosas y puede llegar a escalar a ser una mediana empresa va a crear aún más empleos, y van a empezar a nacer empresas a su alrededor para brindarle servicios que ella por sí misma no puede realizar, y así la zona comienza a subir aún más de nivel.

También algo que salta a la vista es que la mayor cantidad de empresas que existen en el mundo son pequeñas y medianas, este tipo de organismos puede ser complicado su estudio a nivel internacional, ya que como no existe una clasificación que sea aceptada a este nivel provoca que todas las regiones, como la zona europea, o países particulares creen sus propias definiciones. Este solamente es un ejemplo ilustrativo que nos muestra lo complicado que resulta el estudio, pero también nos muestra la gran cantidad de información que se debe manejar para realizarla.

Hablando más específicamente de la industria del *software*, podemos ver que es un mercado que se encuentra en franco crecimiento, que este es una industria que entra completamente en la llamada economía del conocimiento, ya

⁵⁵ Dato tomado de la página CNN expansión en español, con dirección electrónica: <http://www.cnnexpansion.com/expansion/2009/10/05/Mejor-que-en-India> (última visita 16-06-2012).

⁵⁶ Datos tomados de una entrevista a Fabio Bittencourt por la revista CNN expansión en español con dirección electrónica <http://www.cnnexpansion.com/expansion/2009/10/05/Mejor-que-en-India> (última visita 16-06-2012).

que este producto es realizado mediante el trabajo intelectual del trabajador, y normalmente este tipo de empleos pueden llegar a ser bien remunerados, en el caso de toda la industria de tecnologías de la información ha incrementado una tasa promedio en los últimos cinco años de 12% y para el año 2013 se estima alcanzar una producción de 5,000 millones USD⁵⁷.

Como podemos ver el tamaño de la facturación de la industria no es pequeña, y debemos de ayudarnos de todas las herramientas posibles que existan para apoyarlo, por lo mismo en el capítulo cinco de este trabajo de investigación se estudia a detalle todas las herramientas con licencia libre que las empresas pequeñas y medianas ya establecidas están utilizando para poder alcanzar todas sus metas.

⁵⁷ Dato tomado de la revista CNN expansión en español con dirección electrónica:
<http://www.cnnexpansion.com/mi-carrera/2011/12/29/5-sectores-que-daran-empleo-en-2012>
(última visita 16-06-2012).

CAPÍTULO 4: Competitividad en la industria del *software*.

Introducción

En este capítulo se estudiará como se trata de medir la competitividad en las herramientas informáticas, como se verá más adelante esto es una tarea compleja ya que no existe ningún acuerdo en cuanto a la definición de este término.

Se estudiará en la bibliografía existente por todo lo relacionado en cuando a los temas de competitividad y calidad en el *software*, y se estudiara a fondo dos de los principales métodos utilizados en México para la producción de *software*, los cuales son el CMMI que es un modelo creado por el departamento militar de los Estados Unidos y MOPROSOFT que es un modelo completamente nacional creado para apoyar principalmente a la pequeña y mediana empresa productora de *software* nacionales.

El modelo CMMI es uno de los más utilizados a nivel internacional, pero desde su gestación estuvo pensado para ser usado en empresas de medianas a grandes

Competitividad

Competitividad es un término dinámico que requiere ser definido de acuerdo con los cambios que van surgiendo en el comportamiento de la oferta, de la demanda, de la disponibilidad de los recursos, del desarrollo tecnológico, de las necesidades presentes en la sociedad que a cada momento van evolucionando.

Actualmente una de las áreas más importantes en cuanto a la calidad de todas las pymes es lo concerniente a las tecnologías de la información, ya que su uso se a masificado y que la mayoría cuenta con éstas, ya sea que en una tienda ocupen únicamente un equipo lector de barras para marcar las compras y se les facilite el tener una base de datos para su facturación, así como una empresa de servicios que con la ayuda de éstas herramientas envíen correos electrónicos para mantener la información actualizada que ofrece a sus clientes de una manera sencilla y veloz. Todas estas empresas requieren de soluciones informáticas que sean ajustables a sus necesidades y que los costos de las mismas no les resulten mayores que los beneficios que les están proporcionando.

Para este punto es necesario el poder desarrollar *software* de calidad, afortunadamente para poder definir como tiene que ser el *software* de calidad ISO ya definió muy claramente este concepto, “El conjunto de características de una entidad que le confieren su aptitud para satisfacer las necesidades expresadas y las implícitas” (ISO 8402 (66-001-92)).

Y tenemos a Solarte, Muñoz y Arias⁵⁸ que definen la competitividad en el *software* de una manera muy puntual (Solarte, Muñoz, & Arias, 2009).

“La competitividad en el *software* es sinónimo de eficiencia, flexibilidad, portabilidad, usabilidad, seguridad e integridad.

La calidad del *software* es el conjunto de cualidades medibles y específicas que varía de un sistema a otro, dependiendo del tipo de *software* que se va a desarrollar para determinar su utilidad y existencia. Este desarrollo debe ser confiable, mantenible y flexible para disminuir los costos de mantenimiento y perfeccionamiento durante el tiempo de utilización y durante las etapas del ciclo de vida del *software*.”

Como ya vimos en el capítulo número dos en el apartado que explica las ventajas de usar *software* libre, todas estas características están listadas en la parte de los beneficios que obtienes por usarlo.

Diversos autores proponen que la competitividad puede estudiarse a través de 3 dimensiones: país, industria y empresa.

Competitividad en país.

La primera dimensión, de la competitividad cubre el enfoque de país, que quiere decir: “La habilidad de una economía nacional para producir bienes y servicios que superen las pruebas internacionales. Al mismo tiempo que los ciudadanos pueden alcanzar un estándar de vida creciente y sustentable en el largo plazo”, (World Economic Forum, 2011).

El concepto de competitividad entre países se formula con frecuencia en función del enfrentamiento de un país o bloque comercial con otro, sin embargo, existe poca competitividad frontal entre los países y el hecho de que crezca la economía europea o aumente su prosperidad por ejemplo, no implica que la de Estados Unidos vaya a reducirse de alguna manera, o que no pueda aumentar también simultáneamente. El reto de la competitividad no es la lucha entre Asia, Estados Unidos y Europa, ni siquiera la lucha entre bloques comerciales, la economía global está tan entrelazada hoy día que tiene ya poco sentido hablar de una empresa americana, europea o japonesa [Por ejemplo, Dow Chemical⁵⁹ obtiene más del 50% de sus ingresos fuera de EEUU].

⁵⁸ Solarte, G., Muñoz, L., Arias, B. (2009). *Modelos de calidad para procesos de software*. Scientia Et Technica. Vol XV, Num 42. Agosto. Pp 375-379.

⁵⁹ Dow Chemical es una empresa con presencia en más de 19 países alrededor del mundo, su interés principal reside en obtener innovaciones en el área de la química para obtener sustentabilidad en nuestras vidas, tales como la generación de agua potable, energía renovable y el aumento de la producción agrícola.

Tabla 4.1 Factores de competitividad nacional.

Desempeño económico (economía doméstica)	Eficiencia gubernamental (finanzas públicas)	Eficiencia empresarial (productividad)	Infraestructura básica
Economía doméstica <ul style="list-style-type: none"> • Tamaño • Crecimiento • Riqueza • Pronósticos Comercio internacional e inversión internacional <ul style="list-style-type: none"> • Inversión • Finanzas Empleo Precios	Política fiscal Marco institucional <ul style="list-style-type: none"> • Banco central • Eficiencia del Estado • Justicia y seguridad • Legislación de negocios • Nivel de apertura • Regulaciones de la competencia • Regulaciones laborales • Regulaciones del mercado de capital Educación	Mercado laboral <ul style="list-style-type: none"> • Costos • Relaciones • Disponibilidad y capacidades • Finanzas • Eficiencia del sistema bancario • Eficiencia del mercado de valores • Autofinanciamiento Prácticas gerenciales Impacto de la globalización	Infraestructura tecnológica Infraestructura científica Salud y medio ambiente Sistema de valores

Fuente: Tabla tomada de Herrera (2007) con datos de World Competitiveness Yearbook⁶⁰.

Competitividad en la industria.

La segunda dimensión se refiere a la competitividad a nivel industrial que (Porter M. E., 1991) define como el “conjunto de todas las actividades involucradas a la cadena de valor que contribuyen a aportar beneficios o utilidad para el cliente, lo que puede manifestarse en una forma más eficiente de realizarlas bien, entendiéndose por bien, en una manera que sea más posible agregar mayor valor que los competidores de acuerdo con los siguientes elementos.”

1.- Condiciones de los factores: La disponibilidad de los factores de la producción determina no sólo la capacidad de la industria para proveer los insumos necesarios al mercado, si no incluso para exportarlos.

En la industria del *software* su factor principal es de capital humano altamente capacitado para la ejecución inmediata de los proyectos, requiere también de la tecnología necesaria, metodologías, servicios públicos, infraestructura normativa, marco legal, recursos materiales y financieros para la operación y el crecimiento, etc.

2.- Condiciones de la demanda: Se refiere a la composición de la demanda interior, el tamaño y la tendencia de crecimiento de esta necesidad y los medios para transmitir las ventajas que tiene el sector para el exterior.

⁶⁰ Esta tabla es una lista que todos los años publica International Institute of Management Development (IMD), que es una de las instituciones pioneras en cuando a la educación ejecutiva y con mayor prestigio. Con dirección electrónica: <http://www.imd.org/research/publications/wcy/index.cfm> (última visita 19-09-2011).

Para llevar este proceso adelante, el realizar una investigación de mercado con herramientas como *benchmarking* y promoción son indispensables, así como también el análisis de las competencias y la correcta recopilación de información confiable para el diseño de estrategias globales.

3.- Sectores conexos y auxiliares: Los sectores de apoyo son el gobierno, la industria y la academia e influyen directamente en el desempeño y la determinación de las fuentes de competitividad.

4.- Estrategia, estructura y rivalidad de la empresa: La conjunción de metas, estrategias y formas de organización de las empresas de un sector se derivan de la identificación de sus fuentes de competitividad. La rivalidad en el mercado interior tiene reglas que influyen en el desempeño de las empresas y su búsqueda de la perfección, o bien en su declive.

Las empresas de *software* nacionales tienen alternativas para competir en el mercado interior y en el exterior, la formación de consorcios para juntar capacidades por oportunidad de negocio, alianzas integradoras, etc, permite la unión de fortalezas y la disminución de debilidades ante los retos que establecen las grandes empresas transnacionales, ya que la rivalidad puede determinarse por el precio, la calidad, los recursos humanos entre de otros.

Competitividad en empresa.

La competitividad de una empresa se identifica "... a través de la posición relativa que mantiene con respecto a sus competidores, que pueden encontrarse en el mercado interno o en el mercado externo. La vigilancia permanente de los elementos que mantienen a los competidores en el mercado permiten comprender estratégicamente la relación entre el cliente y los recursos usados por las empresas" (Rothschild, 1990).

Para (Müller, 1992), indica la competitividad como: "es la capacidad sostenida para ganar y mantener una capacidad lucrativa en el mercado", pero podríamos redondear esta idea un poco más y definirla como la deja plasmada (Herrera Mendoza, 2007), "es la capacidad sostenida para ganar y mantener una capacidad lucrativa en el mercado a través del incremento sostenido de la productividad en donde la organización fortalece sus estrategias para atraer más clientes, incrementar la eficiencia y establecer un ritmo de actividades de innovación".

Basándonos en lo anterior, generar ventajas competitivas es un reto para todas las empresas en el mercado ya que como dice (Porter M. , 2003), "es un desempeño más que regular a largo plazo... debido a que no hay equilibrio en la competencia. La competencia es un paisaje sometido a constante cambio en el que continuamente afloran nuevos productos, nuevas formas de comercializarlos, nuevos procesos de fabricación y segmentos de mercado completamente nuevos".

Una definición más de competitividad nos la proporciona (Ronzón Contreras, Vidal Turrubiates, Morcillo Presenda, Lazo de la Vega, & Rosa Alburquerque, 2007), “para una empresa, ser competitivo tiene su base en la capacidad de adaptación a los cambios que demanda el mercado y en el tiempo de respuesta a esos cambios, ya que se espera que a menor tiempo de acción en la toma de decisiones y su ejecución, existe mayor oportunidad de ocupar posiciones competitivas privilegiadas.”

Para el presente estudio se utilizará principalmente el apartado de empresa para estudiar la importancia de este elemento en el desarrollo de las capacidades y fortalezas de las empresas que usan dentro de su plan de negocios principalmente o de manera exclusiva herramientas de tipo *open source* para su operación diaria.

Elementos de medición de competencia del software.

Actualmente no hay medidores o tablas comparativas para ayudarnos a determinar que tan competitivo es un *software*, los medidores que existen en la actualidad en la industria de fabricación de *software* son modelos que nos ayudan a medir la mejora de los procesos relacionados con la creación del mismo, este tipo de modelos cuentan ya hoy día con una madurez y reputación ganada para que la mayoría de las organizaciones que se dedican a la creación de programas de cómputo hagan los esfuerzos necesarios para contar con las certificaciones que los acrediten que tienen implementados esos modelos.

Como por ejemplo la Norma Mexicana de NYCE (Normalización Y Certificación Electrónica A.C.)⁶¹, tiene publicada la norma NMX-I-045-NYCE-2005 mejor conocida como norma MoProSoft⁶², donde establece un marco de referencia común para los procesos del ciclo de vida del *software*, con una terminología bien definida a la que puede hacer referencia la industria del *software*. Contiene procesos actividades y tareas para aplicar durante la adquisición de un sistema que contiene *software*, un producto de *software* puro o un servicio de *software*, y durante el suministro, desarrollo, operación y mantenimiento de productos de *software*. Esta norma incluye también un proceso que puede emplearse para definir, controlar y mejorar los procesos del ciclo de vida del *software*.

Por lo tanto, las diferentes normas mexicanas publicadas por el diario oficial de la federación sirven para medir los procesos por los que están contruidos los programas de cómputo, porque esta demostrado que depende de manera directa estos procesos con la calidad del producto final. Pero ninguna de las dos normas

⁶¹ Normatización y Certificación Electrónica (NYCE), es una asociación sin fines de lucro creada por la unión de empresas líderes de electrónica, telecomunicaciones y tecnologías de la información con la finalidad de que se tomaran en cuenta sus puntos y necesidades en las normatizaciones y en las certificaciones de las Normas Oficiales Mexicanas.

⁶² Modelo de Procesos para la industria del Software (MOPROSOFT), es un modelo para la mejora de evaluación de los procesos de desarrollo y mantenimiento de sistemas y productos de software.

(la NMX-I-059-NYCE-2005⁶³ o EvalSoftProft y la NMX-I-045-NYCE-2005⁶⁴ o MoProSoft) nos dan un calificador para medir la competitividad de los productos informáticos.

Así como se deben de medir las tareas por las cuales fue creado un programa, también los productos de *software* cuentan con atributos asociados que reflejan la calidad de ese *software*, estos atributos no están directamente asociados con lo que el *software* hace, más bien refleja su comportamiento durante su ejecución, su comprensión y estructura del programa fuente, así como también la documentación con la que venga acompañado. Existen muchos ejemplos de estos atributos, pero algunos de los más usados es el tiempo de respuesta que tenga este ante una pregunta del usuario final, la comprensión del código fuente para el cliente, la velocidad de respuesta para solucionar un problema que pudiera surgir, entre otros.

El conjunto de atributos que se espera de un *software* no puede ser el mismo para todos, estos atributos dependen ampliamente de la finalidad por la que fueron creados, como por ejemplo un sistema bancario se espera que sea lo más seguro posible aunque tarde un poco en realizar las operaciones, un juego de video se espera que su respuesta sea lo más veloz aunque presenta algunos errores en ejecución, (inclusive en este tema de los video juegos, existen páginas donde se publican todos los errores que se encuentran en video juegos para que la mayor cantidad de personas los puedan explotar), un sistema telefónico se espera que sea estable, que cuando levantemos un teléfono siempre tengamos tono de marcar y no se este cortando la llamada.

En la siguiente tabla se listan las características esenciales que debe de contar un sistema de *software* bien diseñado según (Sommerville, 2005).

Tabla 4.2 Componentes con los que debe de contar los procesos de creación de *software*.

Descripciones del modelo del sistema.	Descripciones de los modelos del sistema que desarrollará y la notación utilizada para definir estos modelos.	Modelos de objetos, de flujos de estado, de máquinas de estado, etcétera.
Reglas.	Restricciones que siempre aplican a los modelos de sistemas.	Cada entidad de un modelo de sistema debe tener un nombre único.
Recomendaciones.	Heurística que caracteriza una buena práctica de diseño en este método. Seguir estas recomendaciones debe de	Ningún objeto debe tener más de siete sub objetos asociados a el.

⁶³ Publicado en el Diario Oficial de la Federación el día 29/11/2005.

⁶⁴ Publicado en el Diario Oficial de la Federación el día 29/11/2005.

	dar como resultado un modelo del sistema bien organizado.	
Guías en el proceso.	Descripciones de las actividades que deben seguirse para desarrollar los modelos del sistema y la organización de estas actividades.	Los atributos de los objetos deben documentarse antes de definir las operaciones asociadas a un objeto.

Fuente: Tabla elaboración propia con datos de (Sommerville, 2005).

También el mismo autor nos lista los atributos con los que debe de contar, que nos muestra la tabla 4.2

Tabla 4.3 Atributos esenciales del buen *software*.

Mantenibilidad.	El <i>software</i> debe de escribirse de tal forma que pueda evolucionar para cumplir las necesidades de cambio de los clientes. Éste es un atributo crítico debido a que el cambio en el <i>software</i> es una consecuencia inevitable de un cambio en el entorno de los negocios.
Confiabilidad.	La confiabilidad en el <i>software</i> tiene un gran número de características, incluyendo la fiabilidad, protección y seguridad. El <i>software</i> confiable no debe causar daños físicos o económicos en el caso de una falla del sistema.
Eficiencia.	El <i>software</i> no debe hacer que se malgasten los recursos del sistema, como la memoria y los recursos del procesamiento. Por lo tanto la eficiencia incluye tiempos de respuesta y de procesamiento, utilización de la memoria, etcétera.
Usabilidad.	El <i>software</i> debe ser fácil de usar sin esfuerzo adicional por parte del usuario para el que esta diseñado. Esto significa que debe tener una interfaz de usuario apropiada y una documentación adecuada.

Fuente: Tabla elaboración propia con datos de (Sommerville, 2005).

Todas estas características que están listadas anteriormente están pensadas para poder enfrentar los mayores retos que enfrenta el *software* para el presente siglo que apenas esta iniciando, estos retos son:

Tabla 4.4 Retos que enfrenta el *software*.

El reto de la heterogeneidad.	Cada vez más se requiere que los sistemas operen como sistemas distribuidos en redes que incluyen diferentes tipos de computadoras y diferentes clases de sistemas de soporte. A menudo es necesario integrar <i>software</i> nuevo son sistemas heredados más viejos escritos en diferentes lenguajes de programación. El reto de la heterogeneidad es desarrollar técnicas para
-------------------------------	---

	construir <i>software</i> confiable que sea lo suficientemente flexible para adecuarse a esta heterogeneidad.
El reto de la entrega.	Muchas técnicas tradicionales de ingeniería del <i>software</i> consumen tiempo. El tiempo que estas consumen es para producir un <i>software</i> de calidad. Sin embargo los negocios de hoy en día deben tener una gran capacidad de respuesta y cambiar con mucha rapidez. El reto de la entrega es reducir los tiempos de entrega para sistemas grandes y complejos sin comprometer la calidad de los sistemas.
El reto de la confianza.	Puesto que el <i>software</i> tiene relación con todos los aspectos de nuestra vida, es esencial que podamos confiar en el. Esto es especialmente importante en sistemas remotos de <i>software</i> a los que se acceda a través de páginas web o interfaces de acceso web. El reto de la confianza es desarrollar técnicas que demuestren que los usuarios pueden confiar en el.

Fuente: Tabla elaboración propia con datos de (Sommerville, 2005).

Modelos de evaluación de madurez de procesos de desarrollo de software.

Con el avance de la tecnología digital, el *software* cada día es más complejo, más poderoso y con un mayor rango de fracaso al momento de su comercialización, esto es debido a que cada vez más empresas se embarcan en este mercado y cada vez están más desarrolladas y más empresas pueden adquirirlas, son de mayor accesibilidad y son más fáciles de usar. Por este motivo, poco a poco se ha ido desarrollando una lista de buenas prácticas que con el paso del tiempo han demostrado tener amplios beneficios en la creación de los productos informáticos, para que estos sean confiables y estén producidos al mínimo costo.

Hoy día ya existen normas nacionales que explican al desarrollador de *software* cuál debe de ser la manera en la que se debe de realizar los procesos para la programación, como se acaba de explicar anteriormente, pero anterior a estas normas la industria de creación del *software* se vio en la necesidad de tener que crear una serie de buenas prácticas, las cuales de ninguna manera nadie esta obligado a seguir, pero la negativa de no implementarlo dentro de la organización necesariamente significa el encontrarse en desventaja en cuanto a los estándares que se están manejando dentro de la industria del desarrollo de soluciones informáticas, no solamente a nivel local como país si no mundial, el conglomerado de todas estas buenas prácticas están resumidas en una serie de métodos y técnicas para desarrollar y mantener *software*, la cual se conoce en español como ingeniería de *software*.

Ingeniería de *software*

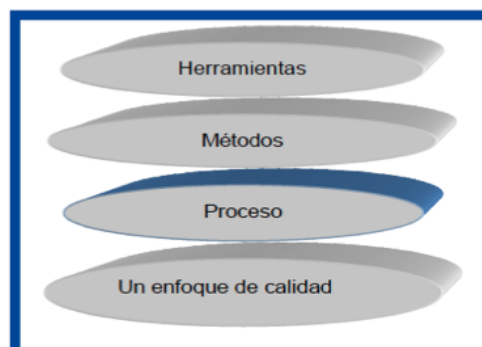
Nuevamente nos encontramos que para este término no existe una terminología aceptada de manera global, según la norma MOPROSOFT se define como “la disciplina tecnológica preocupada en la producción sistemática y mantenimiento de los productos de *software* que son desarrollados y modificados dentro de tiempo y dentro del presupuesto definido. La ingeniería de *software* difiere de la programación convencional en las que se utilizan técnicas de ingeniería para especificar, diseñar, instrumentar, validar y mantener los productos dentro del tiempo y el presupuesto establecido para el proyecto, además esta ingeniería se preocupa por aspectos administrativos que quedan fuera del dominio normal de programación (Alvarez Arriaga, 2011).”

La ingeniería de *software* está definida por la IEEE como “la aplicación de un enfoque sistemático, disciplinado y cuantificable hacia el desarrollo, operación y mantenimiento del *software*⁶⁵”. El motivo principal de este es el presentar a los desarrolladores las técnicas probadas necesarias para reducir la probabilidad de fracaso en la producción de *software*.

La ingeniería del *software* es una tecnología multicapa, la cual contiene elementos que pueden agruparse en varias capas.

En la capa básica se encuentran los componentes o herramientas, que permiten dar un enfoque de calidad al desarrollo completo, sobre estos se encuentran los métodos, después siguen los procesos o dicho de otra manera, las guías que establecen los pasos probados para la generación de los productos que permiten la construcción del *software*. El fundamento de la ingeniería de *software* es la capa del proceso, el proceso en la ingeniería de *software* es la unión que mantiene juntas las capas de tecnología con las superiores para ayudar a un desarrollo racional y oportuno. En la figura 4.1 podemos ver ejemplificado de manera gráfica este concepto de capas en la ingeniería de *software*.

Gráfica 4.1 Representación gráfica de las capas por las que esta constituida la ingeniería de *software*



Fuente: Imagen tomada de (Alvarez Arriaga, 2011) P55.

⁶⁵ IEEE, Estándar IEEE 610.12.

Para poder llevar a cabo todos los pasos anteriormente listados son necesarias el uso de diferentes herramientas, que incluyen lenguajes de programación, *software* para la correcta configuración y control del proyecto, entre otros.

Existen diferentes modelos de buenas prácticas en los procesos de creación de programas informáticos (CMM, CMMI, ISO/IEC 15504, ISO 9001:2000, MoProSoft), y la finalidad de estos procesos es el estandarizar en el grado de lo posible, las mejores prácticas en gestión en ingeniería de *software* que se han encontrado en cuando a su desarrollo, y todos estos modelos están basados en los procesos que nos señala esta materia de estudio, reitero, esta disciplina no busca homogeneizar de ninguna manera el *software* que se produce, esta tarea sería imposible, lo que busca es homogeneizar los procesos por los que están constituidos, y que se ha demostrado con el tiempo que las empresas que los manejan, los productos informáticos producidos cuentan con una alta calidad. Vamos a presentar a continuación los modelos más utilizados como garantía de buenas prácticas para la fabricación de *software* en México.

Modelos de proceso de software.

Los procesos para desarrollar *software* normalmente contienen las actividades de comunicación, planeación, modelado, construcción y arranque. Cualquiera de estas actividades pueden aplicarse de manera concurrente, dependiendo totalmente de la naturaleza del proyecto a desarrollar, para esta decisión existen modelos que nos indican la secuencia en la que el proceso puede ser creado. Pero sin importar cuál sea, la calidad del sistema desarrollado estará altamente influenciado por procesos usados para su elaboración

Para Álvarez define proceso como “un conjunto de prácticas relacionadas entre sí, llevadas a cabo a través de roles, que utilizando recursos e insumos producen un satisfactorio de negocio para el cliente” (Alvarez Arriaga, 2011).

También Álvarez define modelo de proceso como “son guías que presentan las mejores prácticas para el desarrollo del producto en cuestión, su propósito es guiar a las organizaciones en la selección de estrategias de mejora, determinando la madurez del proceso actual para identificar puntos importantes a atacar y mejorar tanto el proceso como la calidad del *software*”.

Todo esto es muy importante porque esta comprobado que la aplicación de modelos de procesos eventualmente lleva a la mejora de calidad del producto final, debido a que se aumenta la capacidad de los procesos. Por esto mismo es de vital importancia que los modelos que se implementen ya hayan sido probados y que están basados en estándares de calidad.

Vale la pena mencionar algunos modelos importantes para la evaluación de la calidad del producto de *software* en términos generales para tener en cuenta la amplia diversidad que existen actualmente y lo importante que son estos para la industria.

Modelos de evaluación de *software*

Modelo en cascada.

El modelo en cascada o ciclo de vida clásico, sugiere un enfoque sistemático, secuencial hacia el desarrollo del *software*, que se inicia con la especificación de requerimientos del cliente y que continúa con la planeación, el modelado, la construcción y el despliegue para culminar en el soporte del *software* terminado. Es un enfoque metodológico que ordena las etapas del ciclo de vida del *software*, de tal forma que el inicio de cada etapa debe de esperar a la finalización total de la inmediata anterior (Pressman, 2005).

Son 5 las actividades por las que se compone este proceso las cuales son:

- **Requerimientos del cliente:** Recopilación de requisitos, inicio de proyecto.
- **Planeación:** Estimación, seguimiento.
- **Modelado:** Análisis, diseño.
- **Construcción:** Código, pruebas.
- **Despliegue:** Entrega, soporte, retroalimentación.

Estas cinco actividades son útiles durante el desarrollo de programas pequeños, la creación de grandes aplicaciones en la red y en la ingeniería de sistemas basados en computadoras grandes y complejas. Algo importante de resaltar de este proceso es que en la vida real raramente un proyecto sigue una secuencia puramente lineal, lo que provoca en una cantidad de casos una implementación del mismo y que termine en fracaso el proyecto.

Modelo en espiral.

Es un modelo de tipo incremental, el cuál combina elementos del modelo en cascada aplicado en forma iterativa.

“El modelo en espiral es un enfoque realista para el desarrollo del *software* y de sistemas a gran escala. Como el *software* evoluciona conforme avanza el proceso, el desarrollador y el cliente entienden y reaccionan de la mejor manera ante los riesgos en cada etapa evolutiva” (Pressman, 2005).

Este es un modelo que se aplica de manera evolutiva, normalmente la primera entrega de este será una hoja desglosando el sistema, y la siguiente solamente un prototipo muy en ciernes, pero poco a poco esta va tomando fuerza hasta llegar al resultado final.

El modelo en espiral se divide en un número de tareas estructurales, llamadas también regiones de tareas. Generalmente son seis las regiones:

- **Comunicación con el cliente:** Las tareas requeridas para establecer comunicación entre el desarrollador y el cliente.

- **Planificación:** Las tareas definidas para definir recursos, el tiempo y otras informaciones relacionadas con el proyecto.
- **Análisis de riesgo:** Las tareas requeridas para evaluar riesgos técnicos y otras informaciones relacionadas con el proyecto.
- **Ingeniería:** Las tareas requeridas para construir representaciones de la aplicación.
- **Construcción y liberación:** Las tareas requeridas para construir, probar, instalar y proporcionar soporte al usuario.
- **Evaluación del cliente:** Las tareas requeridas para obtener la reacción del cliente según la evaluación de la representación del *software*, creadas durante la etapa de ingeniería y de implementación durante la instalación.

Modelo de Boehm.

El modelo comienza con la utilidad general del *software*, afirmando que el *software* es útil, evitando pérdida de tiempo y de dinero. La utilidad puede considerarse respectivamente a los tipos de usuarios que están involucrados. El primer tipo de usuario queda satisfecho si el sistema hace lo que pretende que haga, el segundo tipo es aquel que utiliza el sistema luego de una actualización y el tercero es el programador que mantiene el sistema.

El modelo presenta una jerarquía de características donde cada una de ellas contribuye a la calidad global, se centra en:

- Sus características operativas.
- Sus capacidades para soportar los cambios.
- Su adaptabilidad a nuevos entornos.
- La evaluación en el desempeño del *hardware*.

El tener sistemas de calidad, significa no solo el tener que instalar la metodología del mismo, si no también sistemas de información para controlar y coordinar el proceso completo, sistemas documentales, entre otros. Por lo tanto, la implantación de cualquier sistema de calidad implica que también el *software* que se emplee lo debe de poseer.

Modelo CMMI

El objetivo principal del modelo integrado de capacidad de madurez (CMMI) es que las empresas dedicadas a la fabricación de programas informáticos logren evaluar el avance de desarrollo de *software*, así como la calidad de la administración de un proyecto, a través de niveles de madurez y capacidad.

El origen de este modelo se debió principalmente a que el entorno de desarrollo de sistemas de *software* adolecía de:

- Retrasos considerables en la planificación.
- Poca productividad.
- Elevadas cargas de mantenimiento.

- Demandas cada vez más desfasadas ante las ofertas.
- Baja calidad y fiabilidad del producto.
- Dependencia de los realizadores.

Aunque este tipo de problemas de múltiples maneras se le ha enfrentado, resulta imposible el poder erradicarlo de manera completa. En la actualidad un estimado realizado por (Paños, 1999), menciona que el mantenimiento de los programas informáticos es un gran problema, pues puede llegar a suponer un importe superior al 60% del total del costo del *software* y que las nuevas tecnologías suponen un enfoque integral del problema, abarcando todas las facetas, que en su mayoría no se consideraban en los desarrollos tradicionales. En lo fundamental los puntos críticos son la reducción de costos y plazos, así como también mantener lo más alto la calidad del producto final.

En el año de 1984 el Departamento de Defensa Americano tenía problemas con el *software* que le proveían, cada proveedor lo hacía como mejor lo podía hacer, y tomaba mucho tiempo y esfuerzo el poder adaptarlo con otro que ya tuvieran previamente, por lo tanto fue aprobado por el congreso del gobierno estadounidense la creación de un organismo de investigación para el desarrollo de modelos de mejora para los problemas en el desarrollo de sistemas de *software*, y así evaluar la capacidad de respuesta y fiabilidad de las compañías que suministraban el *software*.

Así el siguiente año nació el Instituto de Ingeniería de *Software* (SEI⁶⁶), este instituto fue creado por el departamento de defensa estadounidense junto con la Universidad Carnegie Mellon, este instituto empieza a trabajar en un marco de madurez de procesos que permita evaluar a las empresas productoras de *software*.

Este instituto identifica tres dimensiones críticas sobre las que una organización puede enfocarse para mejorar su actividad, las cuales son:

- Personas.
- Proceso.
- Herramientas y equipamiento.

Para el año de 1991 el SEI publica la versión 1.0 del Modelo de Madurez de las Capacidades para el *Software* (SW-CMM) y para el año del 2002 ya cambia de nombre a CMMI como el sucesor de CMM, ahora este nuevo modelo ve de manera más integral los procesos dentro de la organización, para el año del 2006 apareció la versión 1.2 y recientemente en el año del 2010 la versión 1.3 que es la que prevalece hasta este momento.

Dado que este modelo resultó ser de gran utilidad, su uso y aplicación se extendió rápidamente en las organizaciones pertenecientes al desarrollo de *software* tanto nacionales como internacionales. Además que este ha demostrado

⁶⁶ Software Engineering Institute.

una gran utilidad no solo para la industria del *software* ya que a partir del año del 2006 se empezaron a crear diferentes áreas de interés que también cubre el CMMI las cuales son:

- CMMI para el desarrollo (CMMI-DEV), aparece en la versión 1.2 en el año del 2006 y trata de procesos de desarrollo de productos y servicios.
- CMMI para la adquisición (CMMI-ACQ), aparece en la misma versión 1.2 pero esta fue publicada en el año del 2007 y trata de la gestión de la cadena de suministros, adquisición y contratación externa en los procesos de gobierno y la industria.
- CMMI para servicios (CMMI-SVC), esta apareció en la versión 1.3 publicada en el año del 2010 y está diseñado para cubrir todas las actividades que requieren gestionar, establecer y entregar servicios.

Es importante señalar que independientemente del modelo que desee adoptar una organización, las prácticas CMMI deben ser adaptadas a cada caso en particular en función a sus objetivos de negocio, además que el periodo promedio de certificación a nivel cinco para las empresas (el más alto de todos), va de tres a cinco años aproximadamente.

Un punto que siempre se tiene que tener en mente cuando estemos hablando en cuanto a CMMI es que la base de estos es que la calidad de un producto o sistema es consecuencia principal de la calidad de los procesos empleados en el desarrollo y mantenimiento. Por lo tanto CMMI es un conjunto de prácticas para mejorar los procesos empleados.

Cuenta este modelo con 5 niveles de madurez, cada nivel de madurez describe y centra su foco en la madurez de la organización, que esta representado por un modelo de mejora y evaluación escalonado, basado en las actividades de mejora y evaluación en la capacidad de los diferentes procesos utilizados para el desarrollo del producto final. Los diferentes niveles del CMMI están representados en la tabla 4.2.

Tabla 4.5 Representación de los diferentes niveles que posee el modelos CMMI.

Nivel	Definición
Nivel 1.- Inicial.	Los resultados de calidad obtenidos son consecuencia de las personas y de las herramientas que utilizan, no de los procesos, por que no los hay o no los utilizan.
Nivel 2.- Repetible.	Se llevan a cabo prácticas básicas de gestión de proyectos, de gestión de requisitos, control de versiones y de los trabajos realizados por subcontratistas. Los equipos de los proyectos pueden aprovechar las prácticas realizadas para aplicarlas en los nuevos proyectos.
Nivel 3.- Definido.	Los procesos comunes para desarrollo y mantenimiento de software están

	documentados de manera suficiente en una biblioteca accesible a los equipos de desarrollo. Las personas han recibido la formación necesaria para comprender los procesos.
Nivel 4.- Gestionado.	La organización mide la calidad del producto y del proceso en forma cuantitativa en base a métricas establecidas. La capacidad de los procesos empleados, es previsible, y el sistema de medición detectar si las variaciones de capacidad exceden los rangos aceptables para adoptar medidas correctivas.
Nivel 5.- Optimizado.	La mejora continua de los procesos afecta a toda la organización, que cuenta con medios para identificar las debilidades y mejorar la prevención de defectos. Se analizan de forma sistemática datos relativos a la eficiencia de los procesos de software para analizar el coste y los beneficios de las adaptaciones y de las mejoras. Se analizan los defectos de los proyectos para determinar las causas, y su mapeados sobre los procesos.

Fuente.- Tabla elaboración propia en base de los datos obtenidos de (Acosta Sinencio, 2010).

Ventajas de implementar el modelo CMMI.

Para Acosta (Acosta Sinencio, 2010) determina que las empresas que estén certificadas en el modelo de CMMI, que como ya vimos anteriormente no solamente las empresas desarrolladoras de *software* pueden buscar esta certificación, si no cualquier empresa que se dedique al desarrollo, servicios o adquisiciones. Si cuentan con el significaría que pueden evaluar el proceso de desarrollo de sus servicios, la calidad de la administración de un proyecto y posiciona a las organizaciones haciéndolas más competitivas, también aumenta la capacidad de cooperación con empresas, proveedores, socios y clientes.

El valor de producción de *software* y sus servicios ha aumentado considerablemente con la adopción de estrategias de calidad, el estar en constante búsqueda por la adquisición de las certificaciones de calidad del *software* es la mejor manera de competir en un mercado en crecimiento que es cada vez más exigente.

Desgraciadamente resulta aún muy costoso para una pyme mexicana el poder cubrir todos los gastos que son necesarios para poder obtener una certificación CMMI nivel 5, hasta hoy día las empresas que cuentan con esta certificación son empresas grandes como IBM México.

MOPROSOFT

La mayoría de las medidas internacionales para la estandarización de *software* están diseñadas para grandes empresas, lo cual es una gran desventaja de manera inmediata para México porque como se estudió en el capítulo pasado aproximadamente el 98% de las empresas en el país están dentro de la

catalogación de las pequeñas y medianas empresas, y para (Silva Alarcón, 2004) en su estudio realizado aproximadamente el 80% de las empresas desarrolladoras de *software* mexicanas pertenecen a esta categoría. Al ver esta realidad resultaba urgente el crear un modelo de estandarización de procesos de manera local para que se adapte a la realidad que enfrentan las empresas de desarrollo de *software* nacionales. Así fue como el gobierno mexicano por medio de la Secretaría de Economía trabajando en conjunto con la industria nacional e instituciones de educación superior, diseñaron un modelo para la estandarización y mejora de procesos de desarrollo de *software* (MoProSoft).

MOPROSOFT está basado principalmente en los modelos CMM y CMMI y está estructurado de una manera muy sencilla que representa los tres niveles de una organización, alta dirección, gerencia y operación. Cada categoría cuenta con procesos definidos para lograr una implementación del modelo. En resumen se puede decir que MOPROSOFT proporciona paso a paso la implementación de un modelo de calidad en la organización para obtener así el producto deseado que cumpla con los requerimientos necesarios.

La categorización de los procesos de MOPROSOFT identifica tres categorías.

- Alta dirección.
- Gestión de negocio.
- Gerencia.
- Gestión de procesos.
- Gestión de proyectos.
- Gestión de recursos.
- Recursos humanos y ambiente de trabajo.
- Bienes, servicios e infraestructura.
- Conocimiento de la información.
- Operación.
- Administración de proyectos específicos.
- Desarrollo y mantenimiento de *software*.

Alta dirección.

En ella se define el rumbo de la empresa, la estrategia a seguir y la forma como se evaluará el cumplimiento de los objetivos. La información generada en este proceso nos ayuda a contestar la pregunta, ¿hacia donde va la organización?.

Gerencia

Es la encargada de la administración de recursos, proyectos y procesos, es la categoría que brinda más información a la organización, porque nos indica por cada recurso humano o material, donde se encuentra, quien o quienes son los responsables del proceso. Esta categoría además de dividirse en tres procesos también cuenta con tres subprocesos, todos listados anteriormente.

El proceso de gerencia tiene como propósito implantar los procesos de la organización en función de los procesos identificados en el plan estratégico, Así como también definir, plantear e implementar las actividades de mejora de los mismos. Es la administradora de los proyectos, buscando que contribuyan al cumplimiento de los objetivos estratégicos de la empresa y finalmente la gestión de los recursos, busca conseguir y brindar los recursos humanos, infraestructura, ambiente de trabajo y proveedores, así como crear y mantener la base de conocimiento de la organización. De manera general se puede englobar que el fin último de gerencia es el cumplimiento del plan estratégico que establezca la alta dirección.

Operación.

Esta es la última categoría, que es donde se realizan las operaciones que permiten existir a las empresas, el desarrollo de los proyectos, aquí están contenidas las actividades de administración de proyectos específicos y desarrollo y mantenimiento de *software*.

En la actividad de administración de proyectos específicos se establecen y se llevan a cabo de forma continua las actividades que permiten cumplir con los objetivos de un proyecto en tiempo, costo y forma esperados.

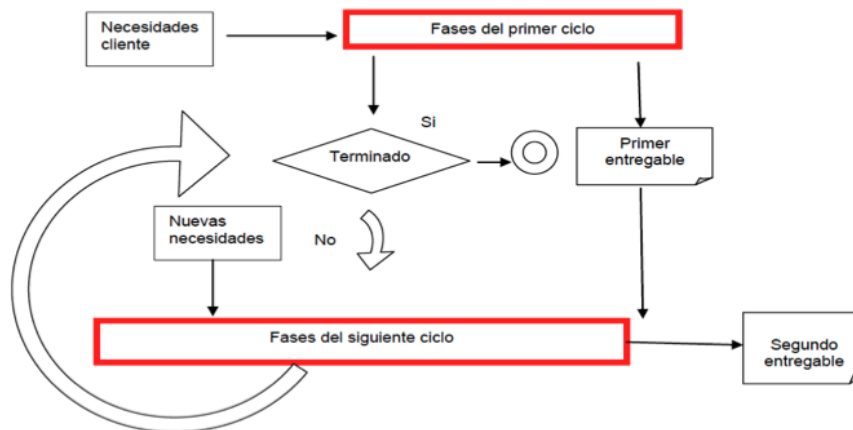
Se define cada etapa de la administración de un proyecto:

- **Planeación:** Actividades cuya finalidad es obtener y mantener el plan de proyecto y el plan de desarrollo que registrarán el proyecto específico, con base en la descripción del proyecto.
- **Realización:** Consiste en llevar a cabo las actividades del plan del proyecto.
- **Evaluación y control:** Consiste en asegurar que se cumplan los objetivos del proyecto. Se supervisa y se evalúa el progreso para identificar desviaciones y realizar acciones correctivas cuando sea necesario.
- **Cierre:** Consiste en entregar los productos de acuerdo a un protocolo de entrega y dar por concluido el ciclo.

Con desarrollo y mantenimiento de *software* se llevan a cabo todas las actividades de análisis, diseño, construcción, integración y pruebas de los productos de *software* ya sean nuevos o modificados cumpliendo con los requisitos solicitados.

El proceso de desarrollo y mantenimiento del *software* se compone de uno o más ciclos de desarrollo como el que se muestra en la figura 4.2.

Gráfica 4.2: Etapas de ciclo de desarrollo.



Fuente: Imagen tomada de (Alvarez Arriaga, 2011) página 32.

Posteriormente cuando ya fue entregado un proyecto el paso final es el mantenimiento del *software*, y MOPROSOFT lo determina como “ es una frase que engloba todo que se usa para denotar las distintas actividades de verificación sucedidas después de la liberación del producto. Las modificaciones se realizan para mejorar, adaptar y corregir errores en los productos de *software*. Mantener la calidad de un producto de *software* a través de ciclos sucesivos de modificaciones y actualizaciones, es un aspecto de importancia fundamental durante el desarrollo de *software*”.

A continuación se muestran los nueve procesos de MOPROSOFT agrupados por categorías, así como su principal objetivo.

Tabla 4.6 Procesos por los que esta conformado MOPROSOFT.

Categoría	Proceso	Objetivo
Alta Dirección (DIR)	Gestión de Negocio (GN)	Establecer la razón de ser de la organización, sus objetivos y condiciones para lograrlos, para lo cual será necesario contemplar las necesidades de los clientes, así como evaluar resultados para poder proponer algún cambio y permitir la mejora continua.
Gerencia (GER)	Gestión de Procesos (GPR)	Establecer los procesos de la organización en función de los procesos requeridos identificados en el plan estratégico. Así como también definir, planificar e implementar las actividades de mejora de los mismos.
Gerencia (GER)	Gestión de proyectos (GPY)	Asegurar que los proyectos contribuyan al cumplimiento de los objetivos y estrategias de la organización.
Gerencia (GER)	Gestión de	Conseguir y dotar a la organización de recursos humano,

	recursos (GR)	infraestructura y ambiente de trabajo, proveedores , también crear y mantener la base de conocimiento de la organización. El objetivo aquí es apoyar con el cumplimiento del plan estratégico de la empresa.
Gerencia (GER)	Recursos humanos y ambiente de trabajo (RHAT)	Proporcionar los recursos humanos adecuados para cumplir las responsabilidades asignadas dentro de la empresa.
Gerencia (GER)	Bienes, servicios e infraestructura (BSI)	Proporcionar proveedores de bienes, servicios e infraestructura que satisfagan los requisitos de adquisición de los procesos y proyectos.
Gerencia (GER)	Conocimiento de la organización (CO)	Conocer y administrar la base de conocimiento que contiene la información y los productos generados por la organización.
Operación (OPE)	Administración de proyectos específicos	Establecer y llevar a cabo sistemáticamente las actividades que permitan cumplir con los objetivos de un proyecto en tiempo y en costo.
Operación (OPE)	Desarrollo y mantenimiento de <i>software</i> (DMS)	Realización sistemática de las actividades de obtención de requisitos de análisis, diseño, construcción, integración y pruebas de producto de <i>software</i> nuevos o modificados cumpliendo con los objetivos.

Fuente: Tabla elaboración propia con datos de Álvarez (Alvarez Arriaga, 2011).

Ventajas de implementar el modelo MOPROSOFT.

Es un modelo fácil de entender y de aplicar en una organización, ya que como lo acabamos de ver, esta dividido en tres partes o categorías, orientada en específico a un área determinada dentro de una organización, permitiéndonos determinar las funciones específicas de cada miembro del equipo para poder realizar un buen producto.

Cuenta con un patrón, el cual sirve de plantilla a los usuarios para documentar cada uno de los procesos realizados. Y estas documentaciones ayudarán al evaluador a determinar que los nueve procesos con los que cuenta este método han sido realizados satisfactoriamente.

En seis meses una empresa puede subir de manera considerable su calidad en producción de *software*, mediante la mejora de su capacidad y madurez en cuanto a sus procesos de desarrollo, ayudando de esta manera a que pueda ofrecer a todos sus clientes un plus en cuanto a sus productos y facilitarle para que pueda llegar a ofrecer innovación.

Al adoptar este modelo las empresas adoptan las mejores prácticas que se han encontrado, específicamente para las pymes mexicanas desarrolladoras de *software*, permitiendo elevar la capacidad de ofrecer sus servicios con calidad.

Y para este autor la principal característica que ofrece el modelo de MOPROSOFT es que fue diseñado desde sus orígenes para las pymes mexicanas desarrolladoras de *software*, ya que todos los otros modelos internacionales y estándares existentes, están pensados principalmente para las grandes empresas, provocando con esto que cuando una pyme trate de implementar alguno de estos modelos le resulte muy costoso y que le tengan que invertir mucho tiempo para tratar de obtenerlo, provocando con esto que pocas empresas desarrolladoras puedan ser candidatas a tratar de implementar estos procesos.

INDICADOR DE COMPETITIVIDAD DE SOFTWARE

Por lo tanto los diferentes modelos e indicadores existentes en el campo de producción de *software* no miden el nivel de competitividad con el que cuentan, sino más bien miden la calidad de los procesos y las prácticas con los que fueron realizados.

Lo que puedo utilizar como instrumento de medición de competitividad es lo que nos apunta el estándar ISO 9126 aprobado en 1992 llamado “ISO 9126: *Software product evaluation: Quality Characteristics and Guidelines for their Use*”.

Este estándar ISO 9126 define un modelo de calidad del *software*, en el que la calidad se define como la totalidad de características relacionadas con su habilidad para satisfacer necesidades específicas o implicadas.

Los atributos los clasifica según seis características, también describe métricas de calidad del *software* basadas en atributos internos y el comportamiento que está teniendo el sistema. Los componentes que toma en cuenta son:

- Funcionalidad.
- Fiabilidad.
- Usabilidad.
- Eficiencia.
- Mantenibilidad.
- Transportabilidad.

Estos serán mis principales vectores de medición que voy a proponer para realizar mi instrumento de medición, de cómo el *software* libre o de código abierto pueden apoyar a una empresa desarrolladora de programas informáticos para alcanzar cualquiera de estas características que debe de tener toda herramienta informática.

La mayoría de las herramientas de tipo código abierto o *software* libre no son desarrolladas por ninguno de estos modelos de trabajo, existen algunas muy claras que si son desarrolladas bajo estos procesos como por ejemplo el navegador web Mozilla Firefox, es la fundación Mozilla la que desarrolla esta herramienta y tiene un equipo de programadores, que siguen todos los pasos de las buenas prácticas aquí mencionados. Otro ejemplo también puede ser la suit de ofimática de Open Office, que aún el año pasado le pertenecía a la empresa de *software* ORACLE, pero que decidió trasladar completamente el proyecto a la fundación de *software* APACHE⁶⁷, estos proyectos están desarrollados bajo este tipo de modelos, pero como se mencionó anteriormente este tipo de prácticas no es la regla dentro del *software* libre ni de código abierto, la mayoría de estas herramientas primero son programadas por una persona o un equipo de personas muy pequeño, probablemente amigos o individuos que se fueron sumando al proyecto poco a poco, que lo hacen sin interés económico alguno mientras son comandadas por una sola persona que va tomando las decisiones conforme van apareciendo.

Pero a pesar de esto, la calidad con la que cuentan la mayoría de las herramientas de *software* libre es muy alta, como lo investiga Eric S. Raymond en su libro “La catedral y el bazar”, en este libro Raymond nos habla de su experiencia como líder de equipo de desarrollo de un *software* de cliente de correo, nos explica como poco a poco se comienza a formar un equipo de manera espontánea, y como de todos ellos recibe aportaciones muy valiosas en forma de comentarios y partes de código para que la agregue, reparan errores, o que tenga nuevas capacidades su cliente de correo, y como esta comunidad prevalece hasta hoy día.

¿En que ayuda principalmente el *software open source*/Libre, para ayudar a estos retos?

Por lo tanto el grueso de las herramientas de *software* libre quizá no este diseñado en base a las buenas prácticas estandarizadas y estudiadas, pero sus resultados están demostrados que son muy positivos, y el tema principal de este estudio es el encontrar cuales herramientas con esta licencia están ayudando a la industria de *software* dentro del Distrito Federal en este momento, y cuales son las necesidades que están cubriendo principalmente.

CONCLUSIONES.

La tarea de medir la calidad con la que cuenta una herramienta informática no es tarea fácil, ni siquiera el concepto ha sido unificado en la literatura o en los diferentes estándares por los que se rige la ingeniería de *software*, diversos son los autores que nos marcan una pausa para determinar tanto la calidad como la competitividad de esta herramienta, pero para ayudarnos a esta tarea ya han sido

⁶⁷ La fundación APACHE nació en 1999 con la creación del servidor web del mismo nombre, esta organización provee del soporte y mantenimiento a la herramienta informática, además de organizar a toda la comunidad ha generado para el buen funcionamiento de esta.

creadas diferentes herramientas y estándares que nos muestran de una manera clara, cuales serian los mejores procesos con los que debería de contar la empresa que lo está fabricando.

No existe ninguna manera de obligar a las empresas desarrolladoras a que sus programadores sigan una manera de hacer el *software* que diseñan, más bien lo que se ha buscado implementar es como “una línea de montaje”, son una serie de procesos que ya han sido estudiados y analizados por mucho tiempo, tanto por las escuelas encargadas de hacer estudios acerca de la ingeniería de *software*, como por la industria misma que se encarga en crear las herramientas informáticas, estos procesos están señalados en diferentes estándares y modelos como el CMMI y el MOPROSOFT.

A pesar que el modelo CMMI esta pensado casi exclusivamente para empresas de medianas a grandes es uno de los más usados en este país, donde la mayoría de las empresas desarrolladoras de *software* son pequeñas, y les resulta casi imposible el poder alcanzar alguna certificación de este proceso por la gran cantidad de puntos que tienen que cubrir, y para las medianas se les presenta como un gran reto que mediante mucho esfuerzo, dinero y tiempo puede llegar a tener.

Por otro lado MOPROSOFT es un modelo creado en este país, en un esfuerzo por el gobierno apoyado por la secretaría de economía y en el académico por investigadores de la UNAM, este modelo desde sus inicios fue pensado para la pequeña y mediana empresa y al paso del tiempo ya se ha transformado en un estándar nacional, es una manera de que una pequeña empresa pueda alcanzar altos niveles de producción, y demuestre que cuenta con los procesos recomendados para crear herramientas informáticas de alta calidad sin la necesidad de invertir tantos recursos ni tiempo para esto.

Las herramientas de código abierto en su mayoría no son creadas bajo los procesos que señalan las buenas prácticas de ingeniería de *software*, pero en base a experiencias por diferentes personas y equipos como el que nos señala Eric S. Raymond en su libro “La catedral y el bazar”, nos ha demostrado que la comunidad que se crea alrededor de estas herramientas produce *software* de muy alta calidad, y las personas que las mantienen son muy leales al proyecto y no lo abandonan fácilmente. Con el paso del tiempo nos han mostrado su gran utilidad y la alta confianza que podemos llegar a poner en ellas, por lo tanto son ya para diferentes empresas opciones de primera mano para realizar sus tareas y cuentan con ellas para todos sus proyectos.

CAPÍTULO 5

Introducción

En el capítulo número uno se han descrito con detenimiento la metodología con la que se ha diseñado y la manera de medir la investigación, las variables mismas que se han propuesto para este fin, así como también los distintos incisos por los que están compuestas las variables y sus posibles respuestas.

En el capítulo dos se analizó con calma todo el estudio del movimiento del *software* libre y de código abierto, se especificó que son dos conceptos distintos en cuanto a su filosofía, pero en cuanto a su utilidad para el usuario final esta no crea mayor diferencia. También se analizaron diferentes tipos de licencias de *software* y se especificaron las diferencias entre de cada una de ellas, así como sus ventajas y sus desventajas.

En el capítulo número tres se estudió a fondo el estado de la pymes, tanto a nivel nacional como internacional, se llegó a la conclusión que no es un ente sencillo de estudiar, ya que no existe ninguna definición que sea compartida a nivel internacional, ya que el estado mexicano tiene una definición particular al respecto, los Estados Unidos manejan la suya propia, así como el continente europeo y diferentes organizaciones internacionales como la OCDE manejan cada cuál su definición particular en el tema, y pueden llegar a resultar hasta contradictorias las definiciones entre de sí. Se determinó que las pymes forman parte importante en la economía global, y particularmente en el caso de México forman más del 98% de todas las empresas nacionales, por lo tanto, es un tema que debe de formar parte de la agenda nacional de cada país.

En el capítulo anterior se estudió a detalle la forma de medir la competitividad y la calidad en el *software*, se reveló que nuevamente estos términos pueden llegar a resultar muy complicados en cuanto al *software*, ya que al ser este un producto de resultado completamente intelectual, el determinar un estándar de qué resultaría correcto y que no en su manufactura es imposible determinar.

Por lo tanto, específicamente en el caso de los Estados Unidos los militares junto con las universidades llegaron a realizar un listado de las mejores prácticas en cuanto a la fabricación de herramientas informáticas, este listado es conocido como CMMI, que es un compendio de las mejores prácticas encontradas para la elaboración de *software*, ya que como se comentó anteriormente, no es posible el establecer cual es la mejor manera que deben de usar los programadores para hacer su trabajo, más bien son los procesos que conforman la creación de la herramienta lo que se ha determinado que influye enormemente en cuanto a la calidad del producto final.

Se estudió que el compendio de buenas prácticas de CMMI fueron diseñadas pensadas principalmente para la gran empresa, esto representa un problema muy particular para el caso de México que como se acaba de mencionar anteriormente, las micro, pequeña y mediana empresa forman más del 98% de

todas las empresas nacionales, por lo tanto, para que una empresa pueda certificarse como CMMI representa un esfuerzo enorme de años y una gran cantidad de recursos económicos, materiales y humanos. Provocando de esta manera para la mayoría de las empresas les resulte inaccesible el poder alcanzar alguna certificación de este tipo, no es un requisito obligatorio para nadie el contar con esta certificación, pero desgraciadamente el no contar con ella coloca a la organización en un plano inferior con respecto de aquellas que sí cuentan con el, por lo mismo, el gobierno federal vía la secretaría de economía junto con académicos de la UNAM, pusieron en marcha un modelo pensado en la realidad de nuestro país, así nació MOPROSOFT, es un modelo que fue creado íntegramente para que las pymes nacionales desarrolladoras de *software* puedan demostrar que las herramientas que producen cuentan con un alto nivel, y que no les resultara en un esfuerzo desproporcionado a su naturaleza para conseguirlo.

Además de estos puntos se establecieron los vectores de evaluación que serán usados para llevar a cabo la investigación.

Resultados:

En este apartado se va a analizar el conjunto de todas las respuestas que fueron posibles recolectar a lo largo del desarrollo de esta investigación, haciéndolos de esta manera datos discretos porque surgieron de un proceso de conteo.

La forma de recolección de datos se realizó principalmente mediante el envío de los reactivos por el correo electrónico, y para los casos que fueron posible se fue personalmente a algún punto de encuentro que dispusiera el entrevistado como su lugar de trabajo o residencia particular.

Un punto importante de esta investigación es que se contó con el apoyo de distintas personas y grupos de desarrolladores de *software* para la ayuda de la distribución de la encuesta y el llenado de la misma, como la encuesta debía de ser llenada por un grupo muy específico de candidatos, todos debían de ser desarrolladores de *software* que utilizarán herramientas de tipo *open source* y que su domicilio fiscal fuera dentro del Distrito Federal, resultó una tarea difícil de afrontar el encontrar organizaciones con estas características, además que por la misma naturaleza de envío de encuestas vía internet el porcentaje de llenado de la misma resulta muy bajo, aproximadamente sólo el 6% de las personas que les llega la solicitud se toma el tiempo necesario de llenar y enviar sus resultados.

Se contó con el apoyo del Dr. José Ángel Gárfias para ingresar al grupo de “KaraOKulta – La comunidad de desarrolladores de Videojuegos en Latinoamérica” dentro de la red social de Facebook, donde se tuvo un alto índice de personas que se tomaron el tiempo necesario para el llenado de la encuesta, así como también el apoyo de la licenciada Alejandra Carpizo Blanco en su puesto de directora de tecnologías de la CANACINTRA que me apoyo dándome datos de contacto de diferentes empresas desarrolladoras de *software* que están inscritas a dicha organización.

Cabe señalar que la herramienta para el análisis de resultados es

enteramente estadística descriptiva para las 31 preguntas que conforman el instrumento, donde solamente 5 preguntas son abiertas, estas son las preguntas 1 donde se le pide que escriba el nombre de la empresa, la número 2 donde se indica que ingrese el puesto que tiene dentro de su lugar de trabajo, la pregunta 7 donde se le pide que indique cuales son las herramientas de código abierto que utiliza más frecuentemente, la 30 donde se le pide que indique que herramientas de código abierto considera son tan competitivas como las herramientas de código cerrado, y por último la pregunta número 31 donde se indica que herramientas de código abierto considera que aún no son tan competitivas como las de código cerrado, estas preguntas se dejaron abiertas porque era necesario que el mismo encuestado nos hiciera sus observaciones precisas.

Se realizó un análisis de cada una de las variables dentro de la matriz de datos apoyados por diferentes herramientas informáticas, las cuales fueron OpenOffice.org, Google docs y Microsoft Office.

La cantidad de organizaciones que apoyaron en la toma de datos de esta investigación es difícil de saber con exactitud, ya que el cuadro señalado para colocar el nombre de la empresa no era obligatorio para enviar los datos, porque algunas personas no desean compartir el nombre de la misma por cuestiones de seguridad. Así como también la mayoría de las personas que participaron tienen el puesto de desarrollador dentro de su organización, por lo tanto, las herramientas que nos describen principalmente están enfocadas a este rubro.

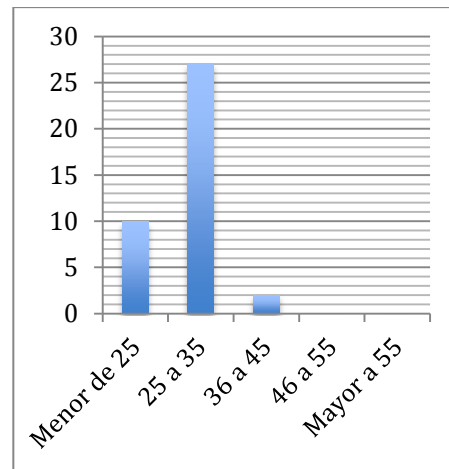
En cuanto a la variable “número de empleados con los que cuenta la organización” es una variable unidimensional ya que solamente recoge una característica, así como también nos sirve para determinar la clasificación de la misma, para categorizar si la organización es micro, pequeña, mediana o grande.

Al momento de diseñar la herramienta se colocó una “pregunta filtro” que fue la que nos dice si utiliza herramientas de código abierto o *software* libre, lo que nos quiere decir esto es que en dado caso que el encuestado conteste negativamente esta pregunta inmediatamente se desechaba todos los datos que nos proporciona, ya que lo que se busca indagar en esta investigación son las herramientas con esta licencia que están siendo utilizadas y que tan competitivas las consideran los encuestados.

Datos del estado de la empresa

Gráfica edad.

Gráfica 5.1: Edades de los encuestados.

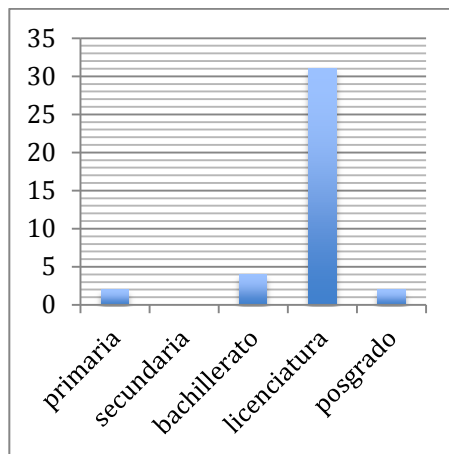


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Como podemos ver en la gráfica 5.1 nos muestra que las edades promedio de las personas encuestadas en este trabajo están en el rango de 25 a 35 años con 27 personas, lo que nos indica que principalmente en este tipo de organizaciones laboran personas jóvenes, y el siguiente rango donde hay más personas son las que tienen menos de 25 años, reforzando todavía más lo que nos demuestra la primer columna, que son personas jóvenes las que se sienten mayoritariamente atraídas a esta industria. En el rango de 36 a 45 años solamente hay 2 personas, pero dentro de las personas encuestadas no hubo un solo trabajador que tuviera más de 45 años. Esto principalmente lo que me indica es que son personas muy jóvenes las que están siendo atraídos a este tipo de labores y las empresas desarrolladoras de *software* deberían de prestar mucha atención a este fenómeno para así aprovechar en mayor medida toda la energía que traen estos nuevos talentos que están ingresando a sus filas.

Gráfica nivel de estudios.

Gráfica 5.2 Nivel de estudios.

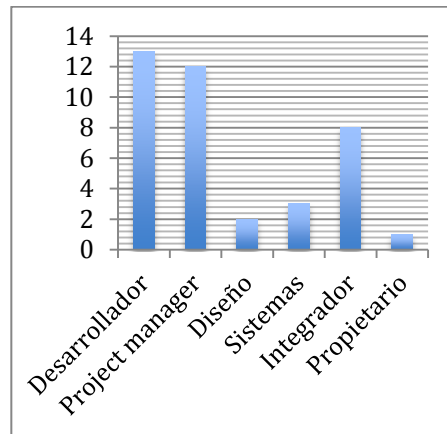


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Ahora lo que podemos ver en la gráfica 5.2 es el nivel de estudios que presentaron los encuestados, lo que salta inmediatamente a la vista es que la mayoría son personas con nivel universitario, 31 personas son las que cuentan con la licenciatura terminada. Y como nos lo muestran los resultados el siguiente nivel en aparecer es el de bachillerato, personalmente pienso que podrían ser personas que ya están en alguna licenciatura pero que aún no la han podido terminar, esta hipótesis se puede reafirmar al ver la gráfica 5.1 que nos muestra las edades de los encuestados que hay 10 personas que son menores a 25 años y que es la segunda edad más común dentro de los encuestados, los siguientes grados académicos que aparecieron fueron el de posgrado con 2 y primaria con la misma cantidad. Lo que principalmente me marcan estos resultados es que las personas que laboran en esta rama de servicios son personas jóvenes y con un grado alto de preparación académica, lo que debería de proporcionarnos un motor eficiente en cuanto a innovación en esta rama, este punto debería de prestar particular interés a las organizaciones para tratar de armar un programa de entrenamiento a sus integrantes, junto con los apoyos necesarios para que las personas se sientan con la confianza suficiente para empezar a proponer ideas nuevas, y así empezar a diferenciarse de la competencia por medio de su capital humano.

Gráfica puesto

Gráfica 5.3: Puesto dentro de la empresa.

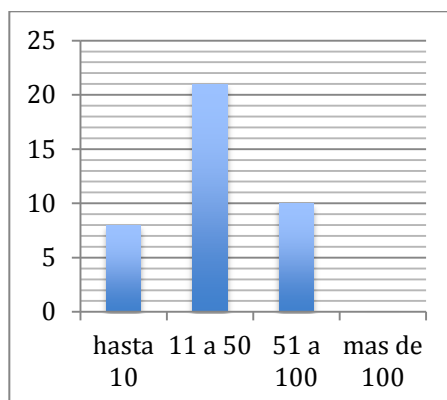


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

En esta gráfica podemos ver que el puesto que ostentan la mayor cantidad de encuestados fue de desarrollador con 13 personas, el siguiente es de project manager con 12 personas y el de integrador con 8, todo este grupo de personas las podríamos colocar dentro de un solo grupo que se puede llamar grupo dedicado a la creación de herramientas informáticas, ya que los desarrolladores, integradores y projects manager son las personas dedicadas a trabajar exclusivamente en el desarrollo de estas, ya sea la creación de nuevas o en darle mantenimiento a alguna ya existente. El puesto que sigue es el de sistemas, este grupo de personas son las que generalmente se encuentran en la parte de infraestructura, que se dedican a que todo el equipo que sea necesario para las labores diarias este funcionando correctamente, y se encarga de administrar los servidores que sean requeridos para que los desarrolladores puedan alcanzar las metas en tiempo y forma. El siguiente grupo es el de diseño, este equipo normalmente trabaja muy de cerca con los desarrolladores, proporcionándoles las imágenes necesarias para que la interfaz que están creando sea atractiva visualmente, y hacer más agradable la herramienta que están creando o adaptando. Por último solamente hay una persona que es dueña de la empresa, este tipo de universo sucedió porque la mayoría de las encuestas fueron enviadas a personas que se encuentran dentro del departamento de desarrollo de *software* y además fueron éstas personas las que se tomaron el tiempo necesario para poder llenar la encuesta.

Gráfica número de empleados.

Gráfica 5.4: Número de empleados.

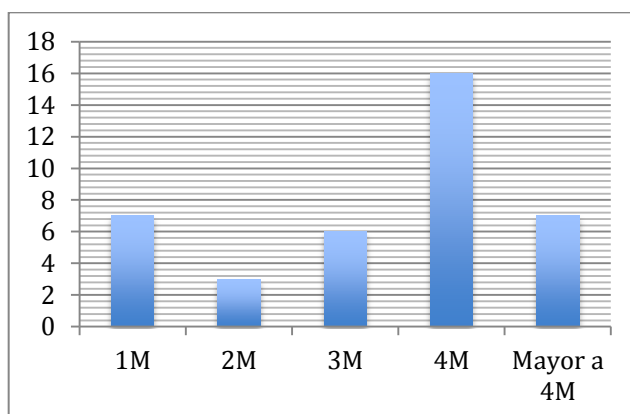


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Esta gráfica nos muestra que principalmente el número de empresas a las que pertenecen los diferentes grupos a los que se les envió la encuesta están dentro de la catalogación de pequeña empresa con 21 organizaciones en este rango, la siguiente es mediana con 10 organizaciones que pertenecen a esta categoría, y finalmente micro con 8. Dentro de las encuestas realizadas existieron varias empresas dentro de la catalogación de empresas grandes con más de 100 empleados, pero esas organizaciones estaban fuera de este trabajo de investigación, por lo tanto sus respuestas tuvieron que ser desechadas. Lo que principalmente nos brinda esta gráfica es que las empresas pertenecientes a la CANACINTRA, a la AMESOL y al grupo KaraOkulta son empresas pequeñas o medianas.

Gráfica conexión a internet.

Gráfica 5.5: Conexión a internet.



Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

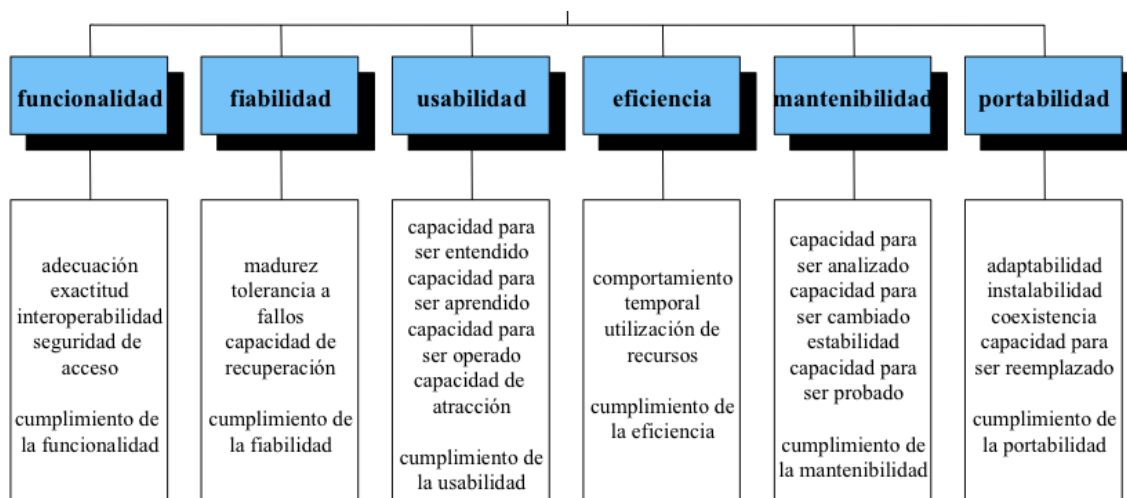
Un punto muy importante que diferentes estudios como los de la OCDE ponen hincapié, es el del ancho de banda de la conexión a internet con la que cuentan los diferentes países miembros, ya que esta demostrado que mientras

más veloz sea la conexión promedio el nivel económico y cultural de la población tiende a aumentar, y con las empresas es mucho más claro que con mejor conexión a la red ésta tiende a ser más productiva. La gráfica nos muestra que la velocidad más común es de 4 megas, personalmente me podría parecer una conexión lenta pensando que la mayoría de las empresas encuestadas están en el rango de pequeña y mediana, por lo tanto puede llegar momentos en que hasta 100 empleados estén tratando de navegar por la red al mismo tiempo y con una conexión de 4 megas resultaría insuficiente. Las siguientes conexiones más comunes son las de mayor a 4 y 1 megas con 7 cada una, las empresas que cuentan con una conexión superior a los 4 megas pueden decir que tienen una buena velocidad para hacer su trabajo pero las que tienen una conexión de 1 mega puede resultar un poco preocupante, sobre todo por el mismo giro al que se dedica la empresa que es el desarrollo de *software*, y para poder realizar bien esta tarea tener una buena conexión a internet es un punto muy importante.

Preguntas con respecto a las herramientas informáticas.

Como se describió en el capítulo uno, los puntos que serán medidos para determinar la calidad que tienen las herramientas informáticas de tipo *open source* que usan las empresas desarrolladoras de *software* son: funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad, son los mismos puntos que están marcados en la norma europea ISO 9126. Pero la manera que se utilizó para medir estos vectores fue preguntando por diferentes características que debe de tener la herramienta, para que la suma de estas nos entregue si cuenta con ese vector o no. Como por ejemplo, el vector de funcionalidad está subdividido en: adecuación, exactitud, interoperabilidad, y seguridad de acceso, estas últimas preguntas fueron las que se realizaron de manera directa a los encuestados, y sí la mayoría de sus respuestas coinciden positivamente, se puede determinar que cuenta con este vector. En la gráfica 5.6 nos muestra todos los vectores que mide la norma y las diferentes sub categorías por las que está compuesto.

Gráfica 5.6: Norma ISO 9126 de manera gráfica. Gráfica tomada de (Alvarez Arriaga, 2011).



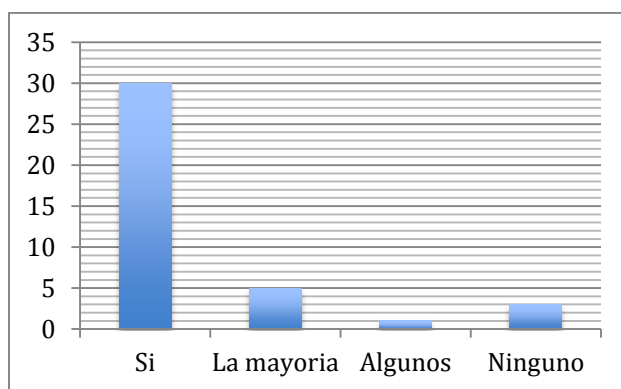
Entonces el análisis de las siguientes preguntas no se pueden analizar de manera individual como las anteriores, se tiene que hacer un análisis de cada

grupo para determinar si los encuestados determinan si las herramientas informáticas de código abierto cuentan con los vectores necesarios para ser consideradas como competitivas o no.

FUNCIONALIDAD

En la pregunta que nos dice “¿El *software* libre (*open source*) que utiliza tiene la capacidad de brindarle un conjunto apropiado de funciones?”, nos proporcionó la siguiente gráfica.

Gráfica 5.7: Capacidad de brindar funciones necesarias.

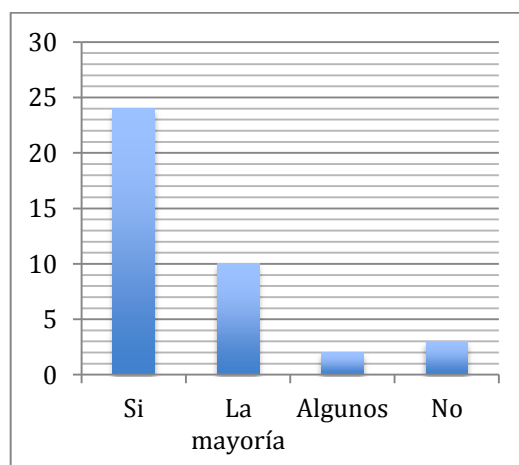


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

En la gráfica 5.7 vemos que la mayoría de las respuestas tiende a que sí cuenta con esta característica.

La siguiente pregunta es ¿El *software* libre (*open source*) que usa le brinda los resultados correctos por los que la adquirió?, genera la siguiente gráfica.

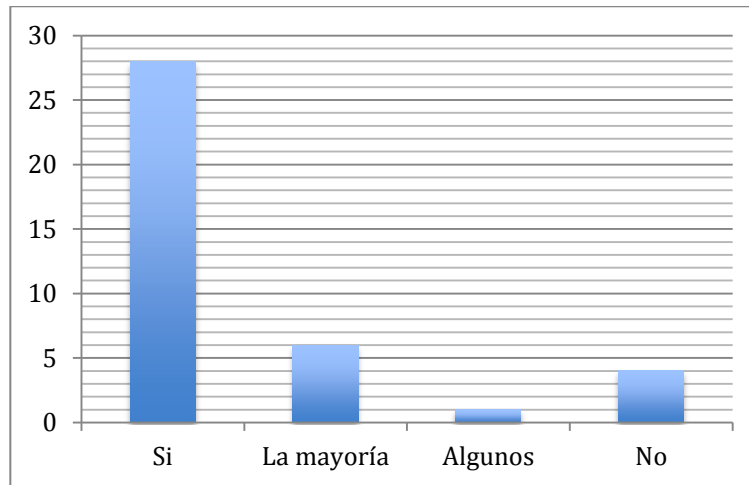
Gráfica 5.8: Brinda los resultados correctos.



Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Nuevamente vemos que la mayoría de los encuestados contestó **positivamente** a esta indicación. La pregunta que continúa es ¿El *software* libre (*open source*) que utiliza puede interactuar con al menos otro *software*?, la que nos proporcionó la siguiente información.

Gráfica 5.9: Puede interactuar con otro *software*.

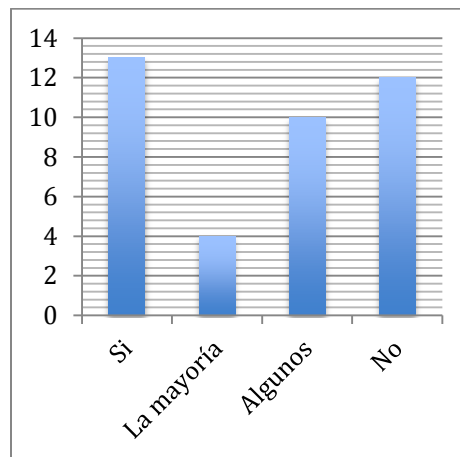


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Nuevamente vemos que las personas encuestadas afirman que las herramientas de código abierto con las que trabajan **pueden interactuar** con otro *software*.

Las últimas de las características necesarias para poder calificar al vector de Funcionalidad es la pregunta ¿El *software* libre (*open source*) tiene la capacidad para proteger la información de manera que las personas no autorizadas no puedan leerlos o modificarlos, al tiempo que no se le niega el acceso al personal autorizado?, la que nos genera la gráfica.

Gráfica 5.10: Seguridad de acceso.



Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Esta gráfica es la única donde las respuestas variaron considerablemente pero aún así la mayoría de las respuestas es **afirmativa**.

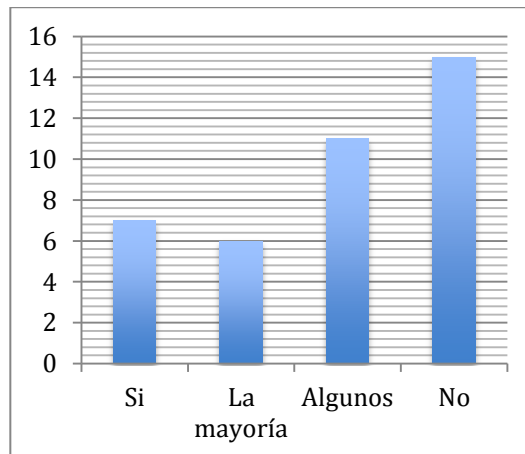
Esto nos conduce a la conclusión que las personas que participaron en este análisis consideran que **las herramientas de código abierto si cuenta con el vector de Funcionalidad**.

FIABILIDAD.

El siguiente vector a medir es el de Fiabilidad el cuál está compuesto por tres características las cuales se analizan a continuación.

La primera de las preguntas por las que esta compuesta es ¿El *software* libre (*open source*) evita fallar por resultados de mala programación?, y los resultados obtenidos por esta pregunta son:

Gráfica 5.11: Evita fallos por mala programación.

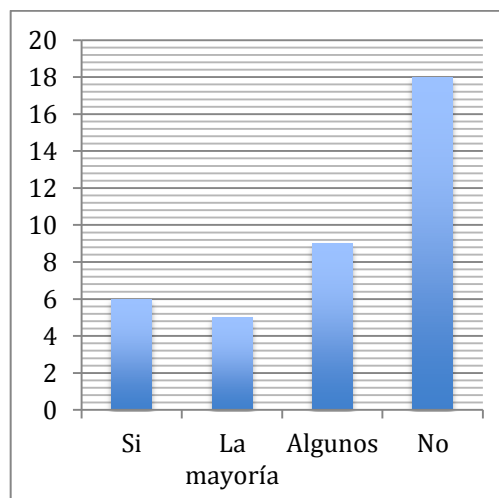


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Aquí resulta muy evidente que las personas encuestadas consideran que la mayoría del *software* libre con el que trabajan **carece de la característica de evitar fallar por resultados de mala programación.**

La siguiente característica por la que se compone este vector está conformada por la pregunta: ¿El *software* libre (*open source*) cuenta con un nivel especificado de prestaciones en caso de fallos o de infringir sus interfaces especificados?, la que nos da los resultados de.

Gráfica 5.12: Evita fallos por mala programación.

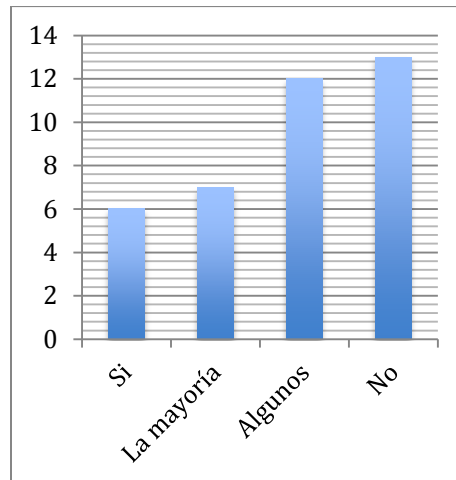


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Nuevamente **en esta respuesta los resultados obtenidos son negativos**, por lo tanto las herramientas de código abierto son la que la mayoría de las personas trabaja tampoco cuenta con la característica de evitar fallos ocasionados por mala programación.

La última de las preguntas por las que se mide este vector es el de: ¿El *software* libre (*open source*) cuenta con un nivel de prestaciones especificado y tiene la capacidad de recuperar los datos directamente afectados en caso de fallo?, y nos genera la gráfica número 5.13.

Gráfica 20 5.13: Capacidad de recuperar datos en caso de fallo.



Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

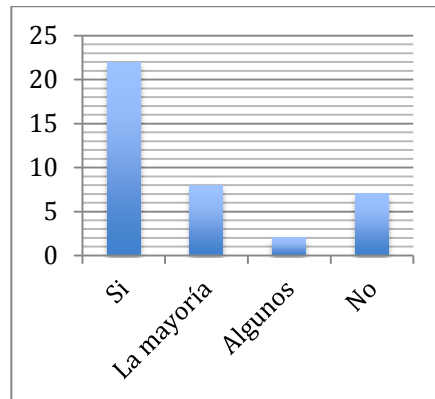
Y nuevamente vemos que las personas consideran que también **carece de esta característica las herramientas de código abierto**. Por lo tanto la conclusión a la que se llega es que las herramientas *open source* con las que trabajan las empresas desarrolladoras de *software* **carece con el vector de Fiabilidad**, teniendo de esta manera una oportunidad enorme de crecer en este punto para así contar con mejores productos con esta licencia.

USABILIDAD.

El siguiente vector a determinar es el de Usabilidad, este está compuesto por 4 preguntas que nos ayudarán a determinar si esta característica la tienen integradas las herramientas *open source*.

La primera pregunta que nos ayudará a esta tarea es: ¿El *software* libre (*open source*) permite al usuario entender si es adecuado y cómo puede ser usado para unas tareas o condiciones de uso particulares?. La que nos genera la gráfica 5.14.

Gráfica 5.14: Cuenta con la documentación suficiente.

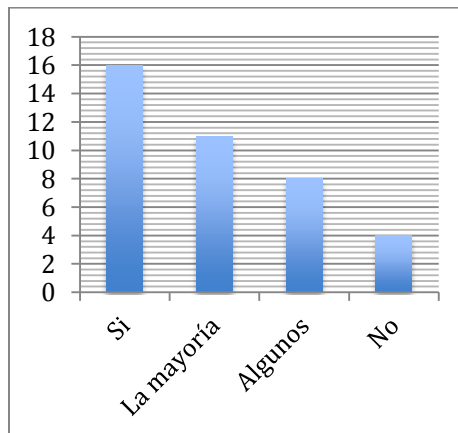


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Esta respuesta es claramente **positiva** ya que 22 personas, más de la mitad, considera que las herramientas *open source* cuentan normalmente con la documentación necesaria para poder trabajar con ellas.

La siguiente pregunta es: ¿El *software* libre (*open source*) permite al usuario aprender sobre su aplicación?, y nos proporcionó los siguientes datos.

Gráfica 5.15: Es intuitivo el *software*.

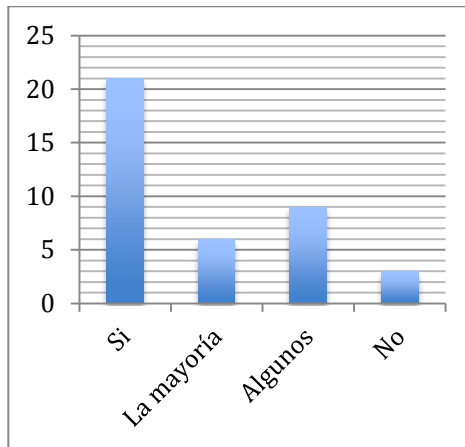


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Aquí nos genera una respuesta **positiva**, ya que 29 personas dan como respuesta que el *software* con el que trabajan es intuitivo en su forma de trabajar.

Una característica más por la que se compone este vector esta medida por la pregunta de: ¿El *software* libre (*open source*) permite al usuario operarlo y controlarlo?, y los resultados obtenidos por esta característica son:

Gráfica 5.16: Es fácil de utilizar.

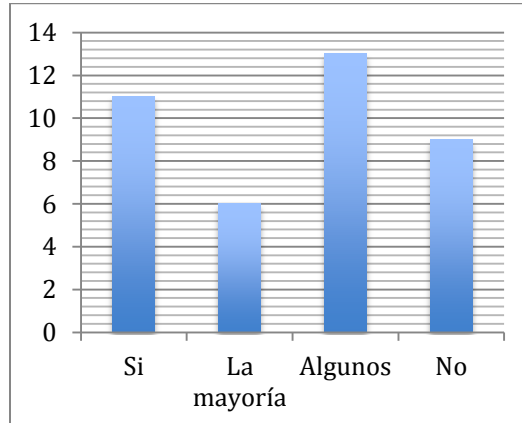


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Nuevamente los resultados obtenidos son **ampliamente positivos** para las herramientas utilizadas, porque 27 personas opinan afirmativamente que las herramientas informáticas con licencia libre son fáciles de utilizar.

La última característica por la que está siendo medido el vector de Usabilidad esta medida por la pregunta: ¿El *software* libre (*open source*) es atractivo al usuario?, y sus resultados fueron:

Gráfica 5.17: Es atractivo visualmente.



Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Esta es la única de todas las características por las que se mide el vector de Usabilidad que resulta un poco complicado determinar si es positiva o negativa, ya que de manera positiva contestaron 17 personas, y de manera negativa 22, no es tan amplio el margen como en todas las otras preguntas anteriores, pero la conclusión a la que se llega es que normalmente las herramientas *open source* **carecen** de una buena presentación visual, que normalmente las herramientas de código cerrado si tienen. Y personalmente ya había comentado este punto en el capítulo dos, en mi opinión personal el mayor problema que enfrentan las herramientas libres es que hay muy pocos diseñadores gráficos que estén dispuestos a dar algo de su tiempo para hacer visualmente más atractivas estas

herramientas, y los programadores no cuentan con las habilidades de diseño suficientes para poder darles ese valor agregado, provocando de esta manera que las personas que no están acostumbradas a usar herramientas con simbología tan minimalista, o completamente en línea de comandos, inmediatamente se apartan y piensan que son muy difíciles.

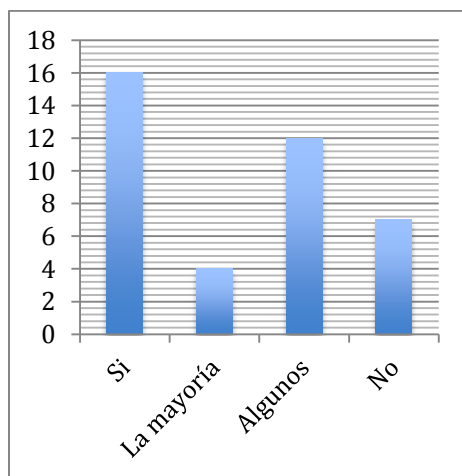
Pero finalmente poniendo en la balanza que para este vector se obtuvo **tres calificaciones positivas de cuatro**, por lo tanto también determinamos que las **herramientas de código abierto cuentan con el vector de Usabilidad**.

EFICIENCIA.

El siguiente vector que mide la norma ISO 9126 es el vector de Eficiencia, este solamente es medido por medio de 2 características, y los resultados de estas preguntas son los siguientes.

La primera pregunta es: ¿El *software* libre (*open source*) proporciona tiempos de respuesta, tiempos de proceso y potencia apropiados, bajo condiciones determinadas?, la que nos da los resultados de:

Gráfica 5.18: Proporciona tiempos de respuesta.

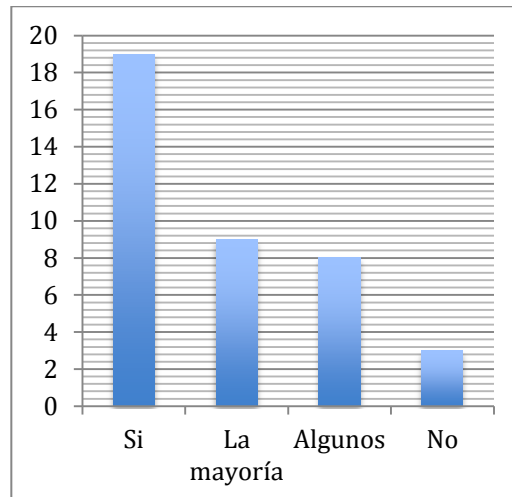


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Las respuestas son mayoritariamente positivas con 20 personas opinando que **sí proporciona tiempos de respuesta** adecuados a sus labores.

Y la pregunta final por la que está determinado este vector es: ¿El *software* libre (*open source*) usa las cantidades y tipos de recursos adecuados cuando lleva a cabo su función bajo condiciones determinadas?, y sus resultados fueron:

Gráfica 5.19: Utiliza solo los recursos necesarios.



Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

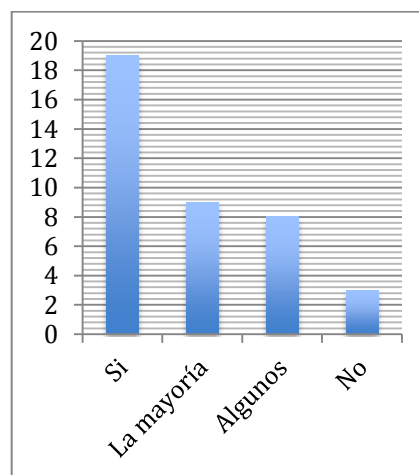
Aquí también los **resultados obtenidos son positivos**, de esta manera llegamos a la conclusión que la mayoría de los encuestados determina que el *software* libre con el que trabajan cuenta con la Eficiencia suficiente para laborar con ellas.

MANTENIBILIDAD.

Un punto más que es necesario medir para determinar la competitividad de las herramientas *open source* es el vector de Mantenibilidad, este vector es medido con la ayuda de 4 preguntas que nos dirán si cuentan las herramientas de código abierto con las características necesarias para agregar este vector.

La primera pregunta es: ¿El *software* libre (*open source*) se le puede diagnosticar fácilmente deficiencias o causas de fallos?, y nos proporciona la gráfica siguiente:

Gráfica 5.20: Es fácil determinar fallos en el *software*.

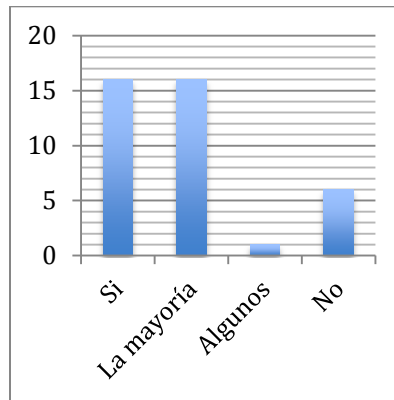


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Esta respuesta es **positiva**, porque 28 personas consideran que cuenta con esta característica.

La siguiente característica esta determinada por la pregunta: ¿El *software* libre (*open source*) permite fácilmente que una determinada modificación sea implementada?, y los resultados obtenidos fueron:

Gráfica 5.21: Es fácil implementar alguna modificación.

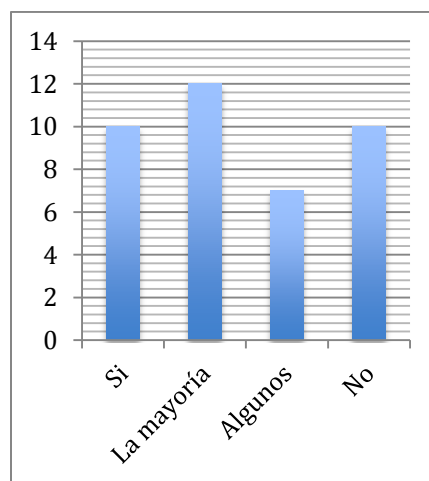


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Esta respuesta es **ampliamente positiva**, ya que 32 personas nos respondieron de manera positiva, mientras que solamente 7 opinaron lo contrario.

El siguiente punto a medir esta definido por la pregunta: ¿El *software* libre (*open source*) cuenta con la capacidad para evitar efectos inesperados debido a modificaciones?, y nos entrega los resultados de:

Gráfica 5.22: Evita efectos inesperados por aplicar modificaciones.



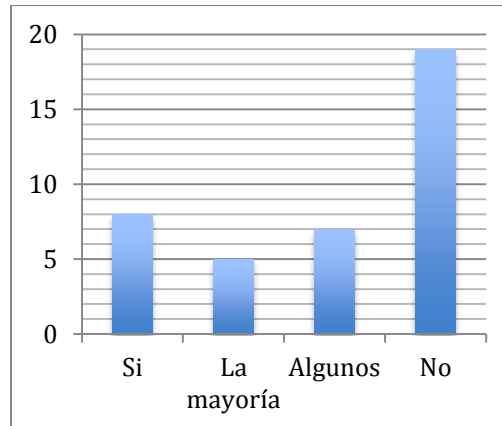
Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Esta gráfica nuevamente no esta tan claro la tendencia de la respuesta, ya que 22 personas opinan de manera positiva en cuanto a esta característica, mientras que 17 opinan lo contrario, es poca la diferencia que existe pero tomando fríamente los resultados, se concluye que **si** cuenta con la característica

preguntada

Finalmente para terminar de concluir si es que cuenta con las características necesarias para determinar si cuenta con el vector de Mantenibilidad se formula la pregunta: ¿Capacidad del *software* libre (*open source*) que permite que si es modificado sea validado?, y los resultados obtenidos fueron:

Gráfica 5.23: En caso de ser modificado puede ser validado.



Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

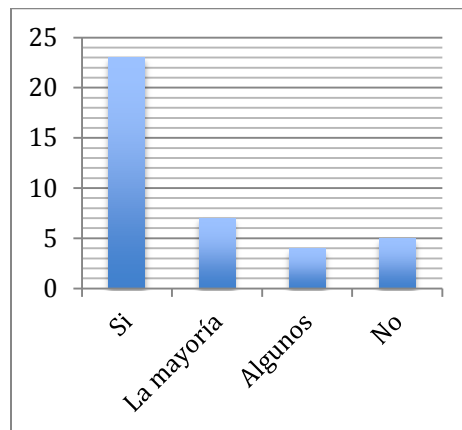
Ahora en esta última pregunta de este vector es **claramente negativa**, con 26 personas opinando que el *software* libre carece de este punto. Pero viendo todas las preguntas por la que esta conformado este vector podemos determinar que **el vector de Mantenibilidad si lo cumple**, pero con la característica de la validación en caso de modificación es un punto de oportunidad para las personas que están desarrollando este tipo de herramientas.

PORTABILIDAD.

El último de los vectores que tiene que ser medido es Portabilidad, este también está compuesto por 4 características, y las preguntas que se realizaron para determinarlos fueron:

La primera pregunta es: ¿El *software* libre (*open source*) tiene la capacidad para ser adaptado a diferentes entornos específicos, sin aplicar acciones o mecanismos distintos de aquellos proporcionados para este propósito por el propio *software* considerado?, y sus respuestas fueron:

Gráfica 5.24: Puede ser adaptado a diferentes SO.

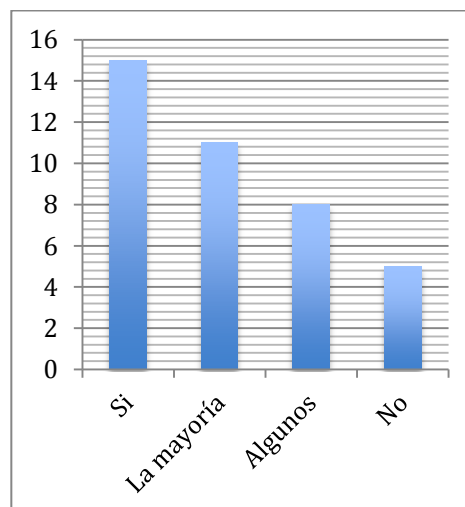


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Esta respuesta es bastante contundente indicando que estas herramientas **son fácilmente exportables** a distintos sistemas operativos, fueron 31 personas las que opinan de esta manera.

La pregunta que sigue es: ¿Es de fácil instalación?, esta es una de las preguntas que personalmente me inquietaba poder ver los resultados, ya que es uno de los puntos que me parece muchas personas le tienen cierto grado de aversión.

Gráfica 5.25: Es fácil de instalar.

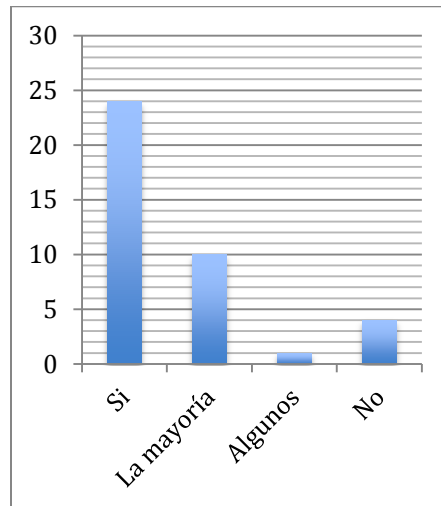


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Los resultados fueron **claramente positivos** a las personas que piensan que no son complicados de instalar, por que 26 personas coinciden con esa opinión.

La siguiente pregunta es: ¿El *software* libre (*open source*) puede coexistir con otro *software* independiente, en un entorno común, compartiendo recursos comunes?, que nos dan las respuestas de:

Gráfica 5.26: El *software* puede coexistir con el *software* o entra en conflictos.

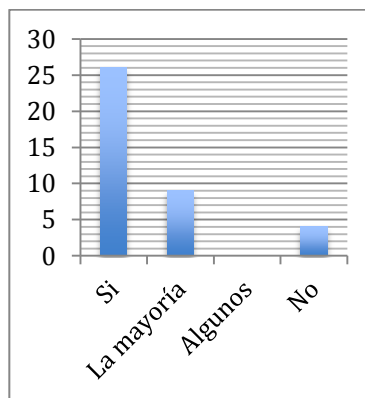


Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Aquí nuevamente las respuestas están indicando **claramente que si**, ya que 34 personas consideran que las herramientas *open source* pueden convivir sencillamente con diferentes programas sin causar conflicto alguno.

Para finalizar con el análisis de este vector se utilizó la pregunta de: ¿El *software* libre (*open source*) puede ser usado en lugar de otro programa, para el mismo propósito, en el mismo entorno?, y se concluye que:

Gráfica 5.27: El *software* puede ser utilizado en lugar de otro programa en la misma computadora.



Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Finalmente para terminar la parte de medición en la calidad del *software* y para terminar este vector, tenemos las respuestas de la característica que **sí podemos utilizar este programa** en lugar de otro diferente y que puedan coexistir en la misma computadora como positivo, 35 personas coinciden que sí es posible mientras solo 4 opinan lo contrario, y con esto llegamos fácilmente a la conclusión que las herramientas de código abierto **sí cuentan con el vector de Portabilidad**.

Análisis de resultados.

Haciendo un recuento de todos los vectores y los resultados son:

- Funcionalidad, si lo tiene
- Fiabilidad, no lo tiene (área de oportunidad).
- Usabilidad, si lo tiene, pero puede ser mejorado en la parte de hacerlo visualmente más amigable.
- Eficiencia, si lo tiene.
- Mantenibilidad, si lo tiene, pero debe de trabajarse más en la parte de la validación en caso de modificación.
- Portabilidad, si lo tiene.

Por lo tanto las herramientas *open source* con las que trabajan las empresas desarrolladoras de *software* localizadas dentro del Distrito Federal, cuentan con la mayoría de los vectores requeridos para cumplir con la norma europea de calidad ISO 9126. Pero se tiene que trabajar fuertemente en el vector de Fiabilidad, entre de otros puntos.

Las últimas preguntas que se realizaron son 3 preguntas abiertas y una cerrada que son las preguntas que me indicaran cuales son las herramientas de licencia libre que más se están usando dentro de esta rama económica, y cuales no se usan, las preguntas son:

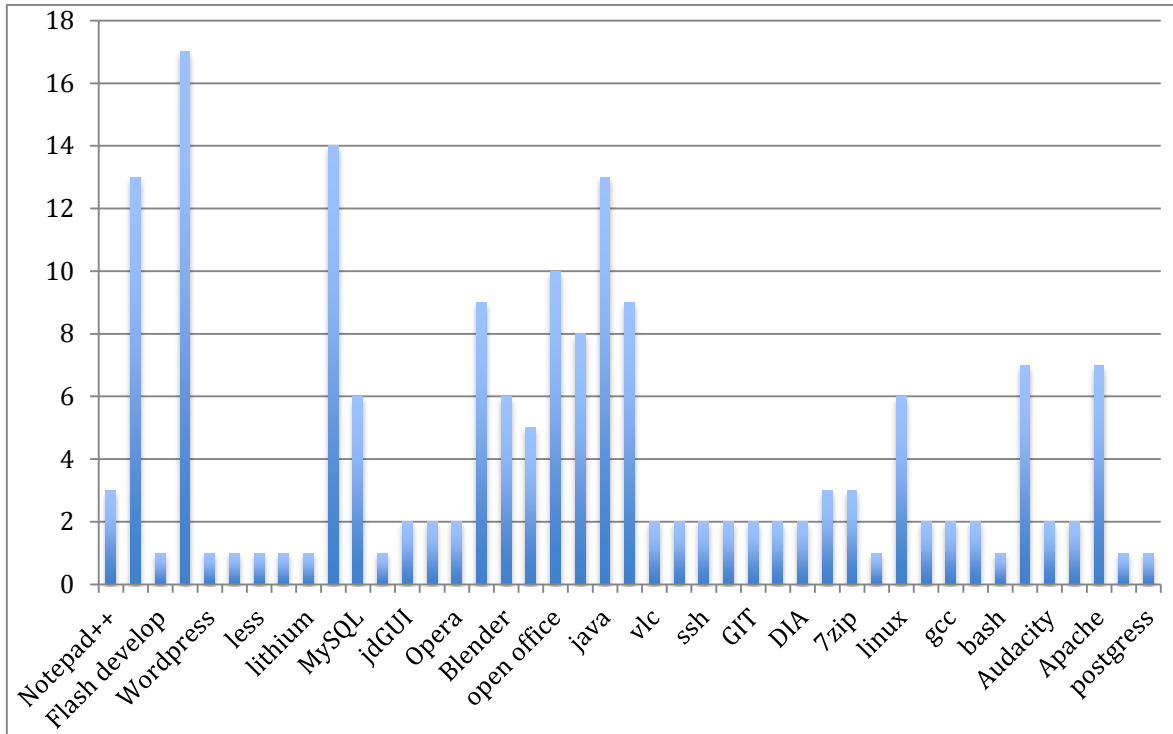
- ¿Cuáles son las herramientas de código abierto o libre que utiliza más frecuentemente?.
- ¿Considera que su productividad es la misma cuando trabaja con herramientas de código abierto o libres?.
- Liste por favor las herramientas de tipo *open source* que considere son tan competitivas como las de código cerrado
- Liste por favor que herramientas de código abierto aún no son lo suficiente competitivas como su equivalente de código cerrado.

Ya que estas preguntas nos indican cuáles herramientas son las más utilizadas en la industria actual de desarrollo de *software* y que tan cómodos se sienten los trabajadores con ellas. Las respuestas fueron.

Herramientas informáticas que se usan con más frecuencia.

¿Cuáles son las herramientas de código abierto o libre que utiliza más frecuentemente?.

Gráfica 5.28: Respuestas a la pregunta, ¿Cuáles son las herramientas de código abierto o libre que utiliza más frecuentemente?.



Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Tabla 5.1 Cuáles son las herramientas de código abierto o libre que utiliza más frecuentemente.

Notepad++	3
Eclipse	13
Flash develop	1
Firefox	17
Wordpess	1
Smfforum	1
less	1
coffescript	1
lithium	1
PHP	14
MySQL	6
HxD	1
jdGUI	2
Filezilla	2
Opera	2
mono	9
Blender	6

Google Chrome	9
vlc	2
Synfigstudio	2
ssh	2
asterisk	2
GIT	2
subversion	2
DIA	2
foxite reader	3
7zip	3
Weka	1
linux	6
Android toolkit	2
gcc	2
g++	2
bash	1
kate	7

Netbeans	5
open office	10
Gimp	8
java	13

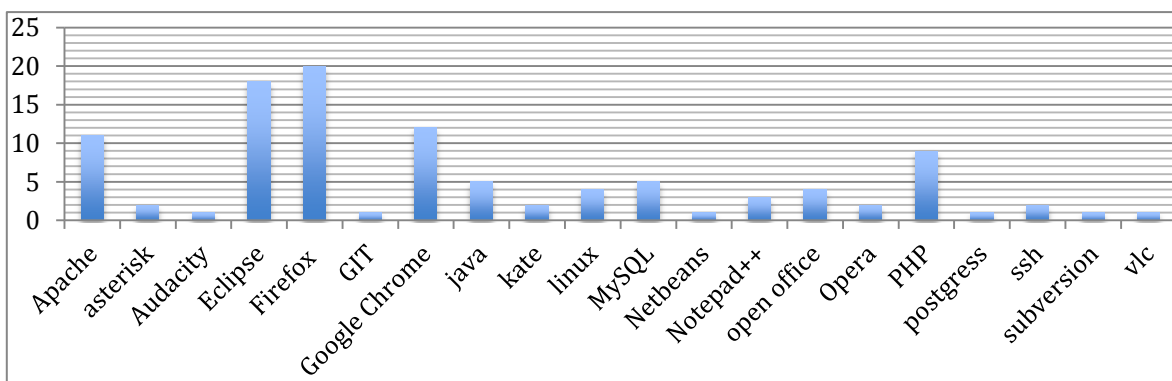
Audacity	2
Sugarsync	2
Apache	7
VNC	1
postgress	1

Fuente: Tabla creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Herramientas libres que son tan competitivas como las de código cerrado

La gráfica para la pregunta de “Liste por favor las herramientas de tipo *open source* que considere son tan competitivas como las de código cerrado” es la siguiente.

Gráfica 5.29: Respuestas a la pregunta, Liste por favor las herramientas de tipo *open source* que considere son tan competitivas como las de código cerrado.



Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Y la tabla donde se obtuvo el gráfico anterior es la siguiente.

Tabla 5.2 Liste por favor las herramientas de tipo *open source* que considere son tan competitivas como las de código cerrado.

Apache	11
asterisk	2
Audacity	1
Eclipse	18
Firefox	20
GIT	1
Google Chrome	12
java	5
kate	2
linux	4

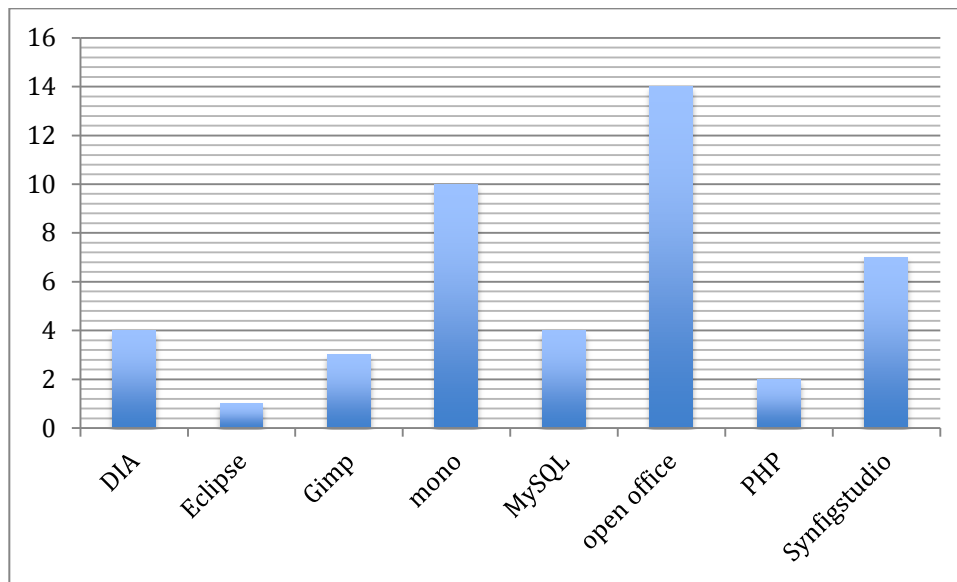
MySQL	5
Netbeans	1
Notepad++	3
open office	4
Opera	2
PHP	9
postgress	1
ssh	2
subversion	1
vlc	1

Fuente: Tabla creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Herramientas libres que aún no son tan competitivas como las de código cerrado.

La siguiente pregunta es: Liste por favor que herramientas de código abierto aún no son lo suficiente competitivas como su equivalente de código cerrado. que produjo la siguiente gráfica.

Gráfica 5.30: Respuestas a la pregunta, Liste por favor las herramientas open source que considera que aún no son lo suficiente estables para tomarlas como primera opción.



Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Y la tabla donde están integrados todos estos datos que aparecen en la gráfica 5.30 es:

Tabla 5.3 Tabla de las herramientas open source que aún no están lo suficiente desarrolladas para trabajar con ellas.

DIA	4
Eclipse	1
Gimp	3
mono	10
MySQL	4
open office	14
PHP	2
Synfigstudio	7

Fuente: Tabla creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

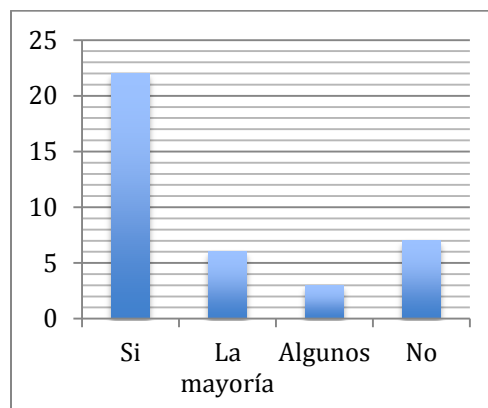
En base a esta tabla podemos apreciar que las **herramientas que menos recomendarían probar para una empresa desarrolladora de software** esta **OpenOffice.org**, desgraciadamente esta herramienta parece ser la gran faltante de resultados, por lo tanto, si alguna empresa esta pensando en comenzar a utilizar esta herramienta para llevar su día a día, quizás resulte mejor el pensarlo con detenimiento en primera quien es la persona que la utilizará y cuantos

conocimientos informáticos tiene esta persona, porque de lo contrario puede resultar contra productivo el integrarla a nuestras soluciones informáticas. **El siguiente programa que necesita de mucho trabajo de desarrollo es mono**, este *software* se utiliza en vez de *.NET*, que es un lenguaje propio de Microsoft y que solamente se puede implementar en el sistema operativo de Windows, a continuación sigue **syngfig studio** que es utilizado para crear animaciones tipo *flash*, al parecer **es una mejor solución el pagar la licencia de la empresa** Adobe para utilizar su kit de desarrollo que el libre. A continuación sigue **el programa de MySQL** que se usa para crear bases de datos, es muy cierto que cuando comienzas a trabajar con bases de datos grandes este *software* no funciona muy bien, se hace necesario adquirir programas de código cerrado como puede ser Oracle, que está demostrado que este programa sirve muy bien para este fin El programa de **DIA** personalmente no lo conozco, pero funciona para generar diagramas de flujo para coordinar las labores de trabajo de un grupo desarrollador de *software*, será mejor buscar una solución diferente a este. Seguimos con **GIMP**, que este es un buen programa para diseño, pero al parecer **las personas que se dedican a esta labor prefieren mucho más las herramientas cerradas** como PHOTOSHOP u otros. Después **continúa PHP** que extraña mucho que halla aparecido en esta lista, porque fue mencionada 9 veces en la lista de los programas mejores para empezar a ser usados en la industria de desarrollo de *software*, además que fue nombrado en esta lista por 2 personas diferentes, quizá sea debido a que algunas librerías de java pueden realizar todas las funciones de PHP y prefieran el uso de estas. Por último aparece **Eclipse**, que también me llamó la atención, pero solamente fue mencionada una sola vez en la lista, quizá la nombró una personas que prefiera otros lenguajes de programación como PHP o Perl.

La productividad es la misma usando herramientas libres que de código cerrado

La última pregunta que falta de listar es la pregunta de ¿Considera que su productividad es la misma cuando trabaja con herramientas de código abierto o libres?. La cual nos proporcionó la siguiente gráfica.

Gráfica 5.31: Respuestas a la pregunta, ¿Considera que su productividad es la misma cuando trabaja con herramientas de código abierto o libres?.



Fuente: Gráfica creación propia con datos obtenidos por las encuestas realizadas para esta investigación.

Donde **la respuesta mayoritaria es positiva**, porque 29 personas

coincidieron **que si son igualmente productivos** o la mayoría de las veces, mientras que solamente **10 personas lo calificaron de manera negativa**, aunque desgraciadamente es un porcentaje considerable de personas que aún consideran que trabajan de manera mucho menos productiva con herramientas libres, que los que lo hacen de manera indistinta con unas o con otras. Este fenómeno sin duda alguna sería un tema interesante el poder determinar porque las personas que contestaron de manera negativa cuáles son los principales motivos de su respuesta, pero este ya es un tema que queda fuera del alcance de este estudio.

Resultado.

Las conclusiones principales a las que se llegó con esta investigación son variadas, la pregunta principal de este estudio era ¿Cuáles son las herramientas *open source* que las empresas pequeñas y medianas desarrolladoras de *software* dentro del DF están utilizando como primera opción informática?, ya quedó claro en este estudio que los programas principales que se están utilizando en este tipo de empresas son los diferentes navegadores de internet como lo es Mozilla Firefox, Google Chrome, pero también diferentes herramientas de desarrollo de *software* como lo es java a través de su interfaz Eclipse, el lenguaje de programación PHP, y el servidor web de Apache, estas son las principales herramientas informáticas que se están utilizando y que los usuarios profesionales de cómputo se sienten satisfechos con los resultados.

Para la primera pregunta secundaria qué es ¿Cuáles herramientas de código abierto son las más utilizadas dentro de las empresa pequeñas y medianas desarrolladoras de *software*?. Se reitera que son muchas y muy diversas las herramientas que se están utilizando, desde programas que pueden solo servir como un simple editor de texto, pero que a la vez este editor de texto es una herramienta que ayuda significativamente al programador, porque tiene muchas funciones integradas para hacer el trabajo lo más sencillo posible como lo es el programa Kate, o herramientas que pueden ser tan enormes como uno lo necesite como lo es la suite de programación Eclipse, ya que el programador decide cuantos plugins necesita instalar. Por lo tanto esta pregunta quedó contestada desde el capítulo número cinco, al momento que se analizaron las herramientas más utilizadas en esta rama de la industria informática.

La siguiente pregunta, ¿Cuáles necesidades aún no pueden ser cubiertas por herramientas *open source* de las pequeñas y medianas empresas desarrolladoras de *software* dentro del DF?. Queda claro que el programa que aún le falta mucho por recorrer es el de *Openoffice.org*, es de llamar la atención esto porque este *software* le pertenece a la empresa de Oracle ,y esta empresa de *software* es la segunda más grande del mundo en esta rama, solamente Microsoft es mayor, y desde mi percepción yo soy un usuario de esa herramienta y no me parece una mala elección de programa de ofimática, y puedo decir que hay algunas tareas que prefiero realizar en esta herramienta que en *Microsoft Office*, además durante casi 3 años exclusivamente trabajé con esta herramienta. Resultó ser una gran sorpresa que este programa apareciera como la menos recomendable para que una persona la trate de ingresar a su negocio de desarrollo de *software*.

Otro programa que al parecer aún le falta un buen camino por recorrer es el lenguaje de programación *mono*. Este es un lenguaje de programación que apareció para poder utilizar la plataforma .NET bajo de diferentes sistemas operativos aparte de Microsoft Windows, ya que este lenguaje le pertenece a esta empresa. Personalmente creo que no tardará mucho tiempo para que la tecnología de *mono* empiece a tomar un grado suficiente de madurez y las empresas desarrolladoras lo utilicen más, digo esto porque ya se ha generado una comunidad muy proactiva alrededor de esta herramienta, que hace no muchos años prácticamente sólo los aventureros de la programación se atrevían a utilizarla

y trataban de entender su funcionamiento.

Un *software* que no me sorprendió que apareciera en la lista negativa es MySQL, ya que de todos es sabido que esta herramienta no está garantizada para poder trabajar con bases de datos que sean muy grandes, para cosas pequeñas funciona muy bien, como ejercicios escolares o bases de datos que no lleguen a los miles, pero sí se empieza a llegar a miles de datos, va a ser necesario que se planee la manera de migrar toda la base de datos a otra plataforma porque este programa no está diseñado para poder manejar bases de datos tan grandes.

Por lo tanto las necesidades informáticas que aún no están del todo cubiertas son diferentes lenguajes de programación, programas de ofimática y bases de datos principalmente.

La pregunta de investigación final, ¿De qué manera las herramientas *open source* ayudan a la competitividad de las empresas que las utilizan?, se puede contestar de la siguiente manera.

Siguiendo el análisis que nos presenta la norma ISO 9126 de todas las características que debe de contar un *software* para decir si este cuenta con calidad suficiente, se llegó a la conclusión que cuenta con todos los puntos excepto con el de fiabilidad, los demás puntos como el de funcionabilidad, usabilidad, eficiencia, mantenibilidad y el de transportabilidad los tienen. El *software* libre cuenta con algunos puntos de manera contundente como el de funcionabilidad, que obtuvo todas las preguntas de manera positiva, en cambio el vector de mantenibilidad, fallo en la validación en caso de realizar un cambio al *software*, pero con la ayuda de este análisis se puede determinar que las herramientas *open source* principales que utilizan las empresas desarrolladoras de *software* son competitivas, e indudablemente estas herramientas harán una gran labor en cualquier empresa que esté dispuesta a darles la oportunidad de probarlas.

Un punto interesante que sobresale en este estudio, es que sería un buen tema de análisis el porqué algunas personas sin importar que tan desarrollada esté una herramienta de tipo *open source* aún se niega a tratar de darle cualquier oportunidad, y de manera automática argumenta que no funciona la opción que se le ofrece, y que no va a poder desempeñar sus funciones de manera correcta si no se le instala el *software* que él está acostumbrado a usar. Desgraciadamente este tema de investigación ya queda fuera del alcance de este estudio y queda esperando que alguien le llegue a interesar tomar este nuevo sendero de investigación.

Otro punto más que sería interesante tomar es realizar este estudio para diferentes ramas de la industria y de servicios. Y así crear poco a poco un repositorio de conocimientos de que herramienta informática libre o de código abierto podemos utilizar para nuestra organización, con la seguridad que las otras empresas del mismo ramo ya están utilizando y los resultados obtenidos de esta son positivos.

Un punto más que también quedó fuera de este trabajo es el porque el

personal de las organizaciones prefiere instalar *software* pirata antes que buscar una opción de código abierto, ya hemos mencionado que la opción de instalar *software* ilegal ni siquiera debería de ponerse sobre la mesa ya que es una infracción directa a las leyes de derechos de autor, que en caso de ser observado por las autoridades competentes derivará en una multa inmediata. Es obvio que las personas argumentarán que el precio de los programas les resulta demasiado elevado, lo que se investigaría es porque no buscan mejor una herramienta que les resulte similar a la primera sin violar ninguna ley de propiedad intelectual, antes que violar la ley y adquirir una herramienta de manera ilegal.

Otro tema igual de interesante es que en algunos países con mayor desarrollo tecnológico al de México, ya no esta en duda que las herramientas de *software* libre atraen a las organizaciones grandes beneficios, como por ejemplo las casas de bolsa más grandes del mundo como el NYSE en los Estados Unidos, la bolsa de Japón, la de Londres entre otras, utilizan un sistema operativo basado en RedHat/Linux, y además que en estos lugares ya se esta empezando a hablar de *Open Hardware*, otro concepto donde ya está implícito que el *software* debe de ser libre, pero no nada más éste, sino también el gabinete que lo contiene debe de ser un circuito electrónico que cualquiera pueda utilizar y tratar de cambiar si así lo desea. Este sería otro campo más de estudio que esta investigación ya no puede cubrir.

Finalmente viendo el estudio en su totalidad, y tratando de alcanzar una máxima conclusión con la ayuda de este, se verifica que existen herramientas informáticas libres lo suficientemente desarrolladas y soportadas por comunidades grandes para que la persona que desee hacer la prueba con ellas tenga gratas experiencias, ya que los resultados que se están obteniendo son positivos, pero es necesario que se tome un tiempo para realizar una investigación de cuales labores si puede llevar a cabo con ellas, ya que existen algunas que por el momento aún no pueden dar los beneficios deseados, o mejor aún, que ingrese a alguna comunidad de desarrolladores de cualquier herramienta libre en la que esté particularmente interesado, para ser un miembro activo de ésta y con su ayuda se puedan alcanzar los beneficios que todos desean obtener.

GLOSARIO:

Software libre.- El software libre (en inglés free software, esta denominación también se confunde a veces con gratis por el doble sentido del inglés free en castellano) es la denominación del software que respeta la libertad de los usuarios sobre su producto adquirido y, por tanto, una vez obtenido puede ser usado, copiado, estudiado, cambiado y redistribuido libremente.

Según la Free Software Foundation, el software libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, modificar el software y distribuirlo modificado. El software libre suele estar disponible gratuitamente, o al precio de costo de la distribución a través de otros medios; sin embargo no es obligatorio que sea así, por lo tanto no hay que asociar software libre a "software gratuito" (denominado usualmente freeware), ya que, conservando su carácter de libre, puede ser distribuido comercialmente ("software comercial"). Análogamente, el "software gratis" o "gratuito" incluye en ocasiones el código fuente; no obstante, este tipo de software no es libre en el mismo sentido que el software libre, a menos que se garanticen los derechos de modificación y redistribución de dichas versiones modificadas del programa.

Tampoco debe confundirse software libre con "software de dominio público". Éste último es aquel software que no requiere de licencia, pues sus derechos de explotación son para toda la humanidad, porque pertenece a todos por igual. Cualquiera puede hacer uso de él, siempre con fines legales y consignando su autoría original. Este software sería aquel cuyo autor lo dona a la humanidad o cuyos derechos de autor han expirado, tras un plazo contado desde la muerte de este, habitualmente 70 años. Si un autor condiciona su uso bajo una licencia, por muy débil que sea, ya no es del dominio público.

Open source.- Es el término con el que se conoce al software distribuido y desarrollado libremente. El código abierto tiene un punto de vista más orientado a los beneficios prácticos de compartir el código que a las cuestiones morales y/o filosóficas las cuales destacan en el llamado *software* libre.

Intel: Intel Corporation es el más grande fabricante de chips semiconductores basado en ingresos. La compañía es la creadora de la serie de procesadores x86, los procesadores más comúnmente encontrados en la mayoría de las computadoras personales. Intel fue fundada el 18 de julio de 1968 como Integrated Electronics Corporation (aunque un error común es el de que "Intel" viene de la palabra intelligence) por los pioneros en semiconductores Robert Noyce y Gordon Moore, y muchas veces asociados con la dirección ejecutiva y la visión de Andrew Grove.

Apple: Apple Inc. es una empresa multinacional estadounidense que diseña y produce equipos electrónicos y software. Entre los productos de hardware más conocidos de la empresa se cuenta con equipos Macintosh, el iPod, el iPhone y el iPad. Entre el software de Apple se encuentran el sistema operativo Mac OS X, el sistema operativo iOS, el explorador de contenido multimedia iTunes, la suite iLife (software de creatividad y multimedia), la suite iWork (software de productividad), Final Cut Studio (una suite de edición de vídeo profesional), Logic Studio (software para edición de audio en pistas de audio), Aperture (software para editar imágenes RAW), y el navegador web Safari .

La empresa opera más de 250 tiendas propias en nueve países, miles de distribuidores (destacándose los distribuidores premium o Apple Premium Resellers) y una tienda en línea (disponible en varios países) donde se venden sus productos y se presta asesoría técnica.

OSX: OS X es un sistema operativo desarrollado y comercializado por Apple Inc. que ha sido incluido en su

gama de computadoras Macintosh desde 2002. Es el sucesor del Mac OS 9 (la versión final del OS Classic), el sistema operativo de Apple desde 1984. Está basado en UNIX, y se construyó sobre las tecnologías desarrolladas en NeXT entre la segunda mitad de los 80's y finales de 1996, cuando Apple adquirió esta compañía. Desde la versión Mac OS X 10.5 Leopard para procesadores Intel, el sistema tiene la certificación UNIX 03.

iWork: es la suite ofimática de aplicaciones creada por Apple para el sistema operativo Mac OS X. Contiene un procesador de texto llamado Pages , un paquete de presentaciones llamado Keynote y una hoja de cálculo llamada Numbers . Aunque iWork fue en un principio un rumor de ser la actualización o reemplazo de la aplicación de Apple llamada AppleWorks, Apple actualmente la comercializa como "construyendo un sucesor de AppleWorks ", pero no extiende las funcionalidades base de datos y herramientas de dibujo de AppleWorks, no pretende competir en forma directa con la versión para Mac de Microsoft Office que se considera más profesional; en cambio, ofrece herramientas con un punto de vista diferente, enfocado al usuario doméstico (presentación en vez de análisis) a un precio más bajo.

GNU General Public License: La Licencia Pública General de GNU o más conocida por su nombre en inglés GNU General Public License o simplemente sus siglas del inglés GNU GPL, es una licencia creada por la Free Software Foundation en 1989 (la primera versión), y está orientada principalmente a proteger la libre distribución, modificación y uso de software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios. Existen varias licencias "hermanas" de la GPL, como la licencia de documentación libre de GNU (GFDL), la Open Audio License , para trabajos musicales, etcétera, y otras menos restrictivas, como la LGPL, o la LGLP (Lesser General Public License, antes Library General Public License), que permiten el enlace dinámico de aplicaciones libres a aplicaciones no libres.

Oracle Corporation: Es una de las mayores compañías de software del mundo. Sus productos van desde bases de datos hasta sistemas de gestión. Cuenta además, con herramientas propias de desarrollo para realizar potentes aplicaciones.

Dell: Es una compañía multinacional estadounidense que desarrolla, fabrica, vende y da soporte a computadoras personales, servidores, switches de red, programas informáticos, periféricos y otros productos relacionados con la tecnología. En 2008 tenía 95.000 empleados en todo el mundo. La corporación creció durante los 80 y los 90 para convertirse durante varios años en el vendedor de PC y servidores más grande del mundo. En 2008 ocupó el segundo lugar, después de Hewlett-Packard Company.

HP: Hewlett-Packard es una de las mayores empresas de tecnologías de la información del mundo. Fabrica y comercializa hardware y software además de brindar servicios de asistencia relacionados con la informática. Se dedicaba a la fabricación de instrumentos de medida electrónica y de laboratorio. Hoy en día es la empresa líder en venta de impresoras.

NEC: Es una compañía multinacional de tecnología y comunicaciones japonesa. Nec proporciona soluciones IT (tecnologías de información) y soluciones de comunicaciones a empresas, servicios de telecomunicaciones y al gobierno.

IBM: Es una empresa multinacional que fabrica y comercializa herramientas, programas y servicios relacionados con la informática. Tiene una presencia principal en prácticamente todos los segmentos relacionados con las tecnologías de la información; de hecho, en los años recientes, más de la mitad de sus

ingresos vienen de sus ramas de consultoría y servicios, y no de la fabricación de equipos. Además es una firme patrocinadora del software libre.

Hitachi: Es una empresa con bases en Japón. La compañía fue fundada en 1910 como una tienda de reparación de equipos electrónicos. Hitachi produce una gran variedad de electrónica de consumo y proporciona productos para otras fábricas por ejemplo circuitos integrados y otros semiconductores.

Bibliografía

Carranza Torres, M. (2004). *Problemática Jurídica del Software Libre*. (Primera ed.). Buenos Aires, Argentina: LexisNexis.

Culebro, J., Gómez, W., & Torres, S. (2006). *Software libre vs Software propietario Ventajas y desventajas*. (Primera ed.). Mexico, Mexico, Mexico: S/N.

Acosta Sinencio, B. (2010). *Implementación de un modelo de calidad “Modelo Integrado de Capacidad y Madurez” (CMMI, Capability Maturity Model) en pequeñas y medianas empresas de software mexicanas*. CU, Ciudad de México, México: UNAM.

Alvarez Arriaga, S. (2011). *MOPROSOFT un modelo básico para el éxito en el desarrollo y mantenimiento del software*. (UNAM, Ed.) CU, México DF, México: UNAM.

Barragán, O. (2011). El gobierno corporativo: una opción para el crecimiento y el desarrollo de las medianas empresas en México. Ponencia presentada en el 1er Congreso de investigación financiera “Las finanzas a 200 años de la independencia”. *El gobierno corporativo: una opción para el crecimiento y el desarrollo de las medianas empresas en México. Ponencia presentada en el 1er Congreso de investigación financiera “Las finanzas a 200 años de la independencia”*. (pág. 19). Ciudad de México: UNAM.

Business Software Alliance (BSA) . (2011). *Eighth Annual BSA Global Software 2010 Piracy Study* . Washington DC, Washington DC, USA: Business Software Alliance .

Business Software Alliance. (2011). *2010 Piracy Study* (Octava ed.). (BSA, Ed.) Washington DC, EE.UU.: BSA.

Encyclopaedia Britannica. (1993). *Encyclopaedia Britannica: World Data (Annual)*. (E. Britannica, Ed.) Indiana, Indiana, EEUU: Encyclopaedia Britannica.

Garcia-Pelayo Gross, R. (2010). *Diccionario Larousse usual enciclopédico*. (Primera ed.). (Larousse., Ed.) México., México., México.

Gaviria Cock, J. R. (2006). DE EMPRESAS Y EMPRESARIOS: Reflexión en forma al sentido y significado de estos conceptos desde la etimología y desde la teoría económica. *Semestre Económico* , 9 (17), 127-137.

Hernández Suárez, C. (2002). *Tecnología para la pequeña y mediana empresa mexicana*. Ciudad de México, DF, Mexico: UNAM.

Hernández, V. S. (2009). LA INDUSTRIA DEL SOFTWARE. ESTUDIO A NIVEL GLOBAL Y AMÉRICA LATINA. *Revista académica de economía* (116), 1.

Hernandez Sampieri, R., Fernández Collado, C., & Baptista Lucio, P. (2006). *Metodología de la Investigación*. (Cuarta ed.). Mexico, Mexico, Mexico: Mc. Graw-Hill/ Interamericana Editores S.A. De C.V.

- Herrera Mendoza, A. (2007). *Las fuentes de competitividad de las empresas mexicanas de desarrollo de software a la medida*. UNAM. México DF. México. DF, México: UNAM.
- Huertas Méndez, F. (2009). El software libre como elemento de desarrollo de la Administración electrónica. . *IDP. Revista de internet, Derecho y Política*. (8), 36-48.
- <http://amesol.com.mx>. (4 de 11 de 2011). *amesol.com.mx*. Recuperado el 4 de 11 de 2011, de Asociación Mexicana de Software Libre: amesol.com.mx
- M.Christensen, C. (2006). The Ongoing process of Building a Theory of Disruption. *The Journal of Product Innovation Management* , 23, 39-55.
- Mochi Alemán, P. (2006). La Industria del Software en México en el contexto internacional y latinoamericano. *MITT/UNAM* , 1.
- Mochi Alemán, P., & Rivera Ríos, M. A. (2006). Nueva modalidad de desarrollo y tecnologías de la información. *Economía informa*. (343), 97-107.
- Morales Castro, A. (2004). *PyME'S financiamiento, inversión y administración de negocios: casos prácticos*. Mexico, DF, Mexico: Gasca Sicco.
- Müller, G. (Diciembre de 1992). Transformaciones productivas y competitividad: aspectos conceptuales y metodológicos”, ponencia presentada en el Seminario-taller Política Tecnológica y Competitividad Agrícola en América Latina. *Instituto Interamericano de Cooperación para la Agricultura* , 14-15.
- OCDE. (2007). *Getting it right OECD Perspectives on Policy Challenges in Mexico*. (Primera ed.). Paris, Paris, France: OECD.
- Organización para la Cooperación y Desarrollo Económico. (2005). *OECD SME and Entrepreneurship Outlook 2005*. Washington, DC, EEUU: Organización para la Cooperación y Desarrollo Económico.
- Paños, Á. A. (1999). Reflexiones sobre el papel de la información como recurso competitivo de la empresa. *Anales de Documentación* , 21-38.
- Porter, M. E. (1991). Towards a Dynamic Theory of Strategy. *Strategic Management Journal* , 12, 95–117.
- Porter, M. (2003). *Ventaja Competitiva: Creación y Sostenimiento de un Desempeño Superior*. México, México, México: Continental.
- Pressman, S. R. (2005). *Ingeniería de Software* (6 ed.). DF, México, México: McGraw-Hill.
- Puga Murguía, C. E. (2002). *Clima laboral : construccion de una herramienta tecnologico-administrativa*. Mexico, Mexico, Mexico: UNAM.
- Ronzón Contreras, J. J., Vidal Turrubiates, L. B., Morcillo Presenda, F. A., Lazo de la Vega, H. T., & Rosa Alburquerque, A. (2007). Modelo para medir la eficacia de proyectos informáticos en una dependencia académica de la Universidad Juárez Autónoma de Tabasco, México. Resultados preliminares bajo la perspectiva de alumnos y asesores.

Visión gerencial , 6, 117-129.

Rosch, W. (1996). *Todo sobre multimedia*. . México., México., México.: Prentice Hall Hispanoamericana.

San Martín Alonso, Á., Sales Arasa, C., & Peirats Chacón, J. (Enero de 2010). POLÍTICAS SOBRE EL SOFTWARE LIBRE EN EL CONTEXTO EDUCATIVO ESPAÑOL. *Pixel-Bit. Revista de Medios y Educación* , 29-40.

Silva Alarcón, A. (2004). *Modelos de calidad: La industria del software en México*. CU, DF, México: UNAM.

Solarte, G., Muñoz, L., & Arias, B. (2009). Modelos de calidad para procesos de software. *Scientia Et Technica* , XV (42), 375-379.

Sommerville, I. (2005). *Ingeniería del software*. (Séptima ed.). Madrid, España: Pearson Educación, S. A.

WITSA. (2010). *Digital Planet Executive Summary*. Taipéi, Taiwán: WITSA. .

World Economic Forum. (2011). *The Global Competitiveness Report 2011-2012*. Genova, Switzerland: World Economic Forum.