

ARTÍCULO

## METODOLOGÍAS HÍBRIDAS PARA DESARROLLO DE SOFTWARE: UNA OPCIÓN FACTIBLE PARA MÉXICO

Eréndira Miriam Jiménez Hernández y Sandra Dinora Orantes Jiménez



## Metodologías híbridas para desarrollo de software: una opción factible para México

### Resumen

En este artículo se muestra que las compañías de desarrollo de software en México son candidatas a usar la nueva tendencia en el área de Ingeniería de Software: las Metodologías Híbridas. Esto se obtuvo a partir de un estudio realizado con las empresas mexicanas dedicadas a desarrollar software. Además se incluyen algunos resultados estadísticos referentes al uso de prácticas de Ingeniería de Software en dichas empresas.

**Palabras clave:** Ingeniería de Software, Metodología Ágil, Metodología Híbrida, Metodología Tradicional.

## Hybrid methodologies for software development: a feasible option for Mexico

### Abstract

This paper shows that software development companies in Mexico are candidates to use the new trend in the area of Software Engineering: Hybrid Methodologies. This was obtained from a study with Mexican companies dedicated to developing software. Besides, it includes some statistical results concerning to the usage of Software Engineering practices in these companies.

**Keywords:** Software Engineering, Agile Methodology, Hybrid Methodology, Traditional Methodology.

### Introducción

De acuerdo con el Diccionario de la Real Academia Española (2011), la palabra *metodología* es un conjunto de métodos que se siguen en una investigación científica o en una exposición doctrinal.

En el área de Ingeniería de Software, el término *metodología* (Pressman, 2005) se refiere a un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo de sistemas computacionales.

Así, se espera que al utilizar una metodología para desarrollo de software, ésta pueda proveer un conjunto de prácticas y herramientas que faciliten el proceso de desarrollo, ofreciendo un producto con alta calidad, seguro y que satisfaga las expectativas del cliente.

Actualmente existen muchas metodologías, las cuales se pueden dividir en dos tipos principales: Ágiles y Tradicionales. Sin embargo las Metodologías Híbridas están marcando la nueva tendencia en el área de Ingeniería de Software, al considerar algunas de las mejores características de ambas metodologías.

Algunos ejemplos de Metodologías Tradicionales son: RUP (*Rational Unified Process*, Proceso Racional Unificado, IBM, 2011), OMT (*Object Modeling Technique*, Técnica de Modelado de Objetos, Rumbaugh *et al.*, 1990 y Yourdon, 1976). En este tipo de metodologías se considera la importancia de la documentación del sistema, lo cual permite entender, extender y darle mantenimiento al software. Además, estas metodologías proporcionan un orden y una estructura bien definida para el desarrollo del software. Sin embargo, para que estas metodologías funcionen adecuadamente, se requiere de un alto grado de disciplina por parte de todos los miembros del

equipo de desarrollo. Asimismo, no se tiene respuesta rápida a cambios; se genera documentación innecesaria, y se requiere mucho tiempo en el modelado del sistema. Por lo tanto las Metodologías Tradicionales no consideran que el análisis, el diseño y la construcción son impredecibles en la mayoría de las veces (Piattini et al., 2007).

Por otro lado, están las Metodologías Ágiles, las cuales presentan respuestas rápidas y efectivas al cambio; tienen un plan de proyecto flexible, y muestran simplicidad, de manera general, en el desarrollo. Sin embargo, tienen la desventaja de generar poca documentación y no hacer uso de métodos formales. Algunos ejemplos de este tipo de metodologías, son: XP (eXtreme Programming, Programación Extrema, Beck, 2010) y Scrum (Scrum group, 2011).

Las Metodologías Híbridas retoman las ventajas de los dos tipos de metodologías anteriores. De esta manera, son una combinación de las mejores prácticas existentes dentro de ellas.

La nueva tendencia en Ingeniería de Software es diseñar Metodologías Híbridas. Esta propuesta es atribuida a Ivar Jacobson, uno de los tres creadores de UML (Unified Modeling Language, Lenguaje Unificado de Modelado, Object Management group, 2011); creador de UP (Unified Process, Proceso Unificado), y ahora creador de EssUP (Essential Unified Process). EssUP (Jacobson, 2011) es una metodología híbrida que combina RUP con Scrum.

Por lo tanto, diseñar una metodología híbrida para que las empresas de desarrollo de software en México la puedan utilizar, es una buena opción para incrementar su productividad.

Sin embargo, antes de crear una nueva metodología híbrida para desarrollo de proyectos de Software en México, es necesario saber si ésta realmente podría representar una opción factible, así como conocer cuáles son las prácticas de Ingeniería de Software que se emplean en las empresas mexicanas para ser incluidas.

Si se desea utilizar en una empresa una metodología híbrida, es conveniente tener en cuenta los resultados de este estudio, obtenido de manera formal por medio de una prueba de hipótesis, que arroja elementos de gran utilidad en la búsqueda de mejorar el desarrollo de software. Esto redundará en un menor costo, un menor tiempo de desarrollo, calidad y mantenimiento de software. Las hipótesis a probar, son:

### Hipótesis

$H_0:$	$p_1 \geq 0.50$
$H_1:$	$p_1 < 0.50$

Donde:

$p_1$ : Proporción de empresas desarrolladoras de software con una inclinación hacia metodologías híbridas.

La interpretación de las hipótesis es la siguiente:

$H_0:$	El 50% o más de las empresas desarrolladoras de software tienen una inclinación hacia el uso de metodologías híbridas.
$H_1:$	Menos del 50% de las empresas desarrolladoras de software tienen una inclinación hacia el uso de metodologías híbridas.

## Materiales y Métodos

Para poder realizar el estudio se diseñó una encuesta con 19 preguntas, con respuestas de opción múltiple. Esta se incluye en el Apéndice A.

También se recurrió a los datos estadísticos del INEGI (Instituto Nacional de Estadística y Geografía). Según el INEGI, en 2010 se contabilizaron 9540 empresas en México dedicadas al desarrollo de software.

Para calcular el tamaño de la muestra, se aplicó primeramente la encuesta a un grupo piloto de 20 empresas y se obtuvo que:

40%	Se inclinaron por el uso de metodologías híbridas
60%	No se inclinaron por el uso de metodologías híbridas

Estos porcentajes representan los valores estadísticos de p y q, respectivamente, mismos que se utilizan para encontrar el tamaño de la muestra en la siguiente fórmula matemática.

$$n = \frac{Z^2 pqN}{NE^2 + Z^2 pq}$$

Donde:

n	es el tamaño de la muestra
Z	es el nivel de confianza
p	es la variabilidad positiva
q	es la variabilidad negativa
N	es el tamaño de la población
E	es la precisión o el error

Considerando que el nivel de confianza es del 95% y con una población de 9540 empresas, se tiene:

$$n = \frac{(0.95)^2 (0.40)(0.60)(9540)}{(9540)(0.05)^2 + (0.95)(0.40)(0.60)}$$

$$n = 85.86$$

$$n \approx 86$$

De la lista de empresas dedicadas a desarrollar software en México, que proporciona el INEGI, se seleccionaron 86 aleatoriamente para aplicarles la encuesta.

Es importante mencionar que la selección de las empresas a las cuales se les aplicó la encuesta fue en forma aleatoria, que no todas las empresas señaladas por el INEGI se pudieron encontrar y que no todas mostraron interés en participar, de tal manera que en ese caso fueron sustituidas en forma aleatoria por otras empresas, hasta completar las 86 de la muestra.

## Resultados

Después de encuestar a las 86 empresas mexicanas dedicadas a desarrollar software, se obtuvieron los resultados de la tabla que se muestra en el Apéndice B. En dicha tabla sólo se tiene información de las preguntas 3, 4, 6, 7, 8 y 13 de la encuesta, porque estas preguntas son las que permiten determinar el tipo de metodología utilizada por una empresa para desarrollar software. (El nombre de las empresas que corresponden a los números de dicha tabla, se encuentra en el Apéndice C).

Los valores que aparecen como respuesta a las preguntas, fueron asignados dependiendo del tipo de metodología:

Valor	Tipo de metodología
1	Metodología Ágil
2	Metodología Híbrida
3	Metodología Tradicional

Al analizar los datos, se encontró que el número de empresas que prefieren los tres tipos de metodologías se distribuyen de la siguiente manera:

Tipo de metodología	Número de empresas
Metodologías Ágiles	22
Metodologías Híbridas	50
Metodologías Tradicionales	14
<b>TOTAL</b>	<b>86</b>

Se puede observar que el número de empresas que prefieren metodologías híbridas es de 50 y no híbrida, de 36.

Para determinar si una empresa tiene inclinación por usar una determinada metodología, no se empleó una sola pregunta, sino los valores de las preguntas que distinguen a dicha metodología. En las preguntas donde era posible responder más de una opción, se sumó la puntuación y se dividió entre el número de preguntas contestadas para encontrar el promedio.

La prueba de hipótesis antes mencionada, es como se muestra a continuación.

### Hipótesis

$H_0:$	$p_1 \geq 0.50$
$H_1:$	$p_1 < 0.50$

El estadístico de prueba, es:

$$z = \frac{\hat{p} - p}{\sqrt{\frac{p(1-p)}{n}}}$$

Donde:

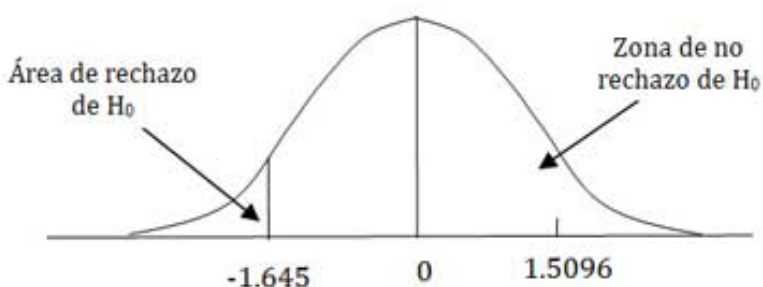
6 -xx

$z$	es el estadístico de prueba
$\hat{p}$	es la proporción de la muestra que tiene inclinación hacia el uso de metodologías híbridas
$p$	es la proporción de la población = 0.50
$n$	tamaño de la muestra

Sustituyendo:

$$z = \frac{\left(\frac{50}{86}\right) - 0.50}{\sqrt{\frac{0.50(1 - 0.50)}{86}}} = 1.5096$$

La información obtenida en el estudio tiene una distribución normal, de tal manera que con un nivel de significancia (o error) del 5% el área de rechazo o no rechazo de la hipótesis nula  $H_0$ , es como se muestra en la siguiente figura:



**Figura 1.** Área bajo la curva de la distribución normal

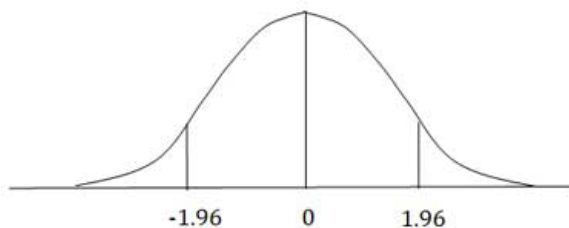
NOTA: El valor del área bajo la curva normal (-1.645) se obtuvo de tablas estadísticas

Se observa en la figura 1 que el estadístico de prueba cae en la zona de no rechazo. Por lo tanto no se rechaza  $H_0$  y se dice que:

“El 50% o más de las empresas desarrolladoras de software tiene una inclinación hacia el uso de metodologías híbridas”.

Para corroborar la prueba de hipótesis se realizó un intervalo de confianza para un nivel de significancia del 5%. Los límites de dicho intervalo se muestran a continuación:

$$z_{\frac{\alpha}{2}} = z_{\frac{0.05}{2}} = z_{0.025} = 1.96$$



**Figura 2.** Intervalo de confianza

Lo cual implica que con un error del 5% la proporción de desarrolladores de software que prefieren metodologías híbridas se encuentran entre el 47.77% y el 68.5%

$$\hat{p} - z_{\frac{\alpha}{2}} \sqrt{\frac{pq}{n}} < p < \hat{p} + z_{\frac{\alpha}{2}} \sqrt{\frac{pq}{n}}$$

$$\frac{50}{86} - 1.96 \sqrt{\frac{\left(\frac{50}{86}\right)\left(1 - \frac{50}{86}\right)}{86}} < p < \frac{50}{86} + 1.96 \sqrt{\frac{\left(\frac{50}{86}\right)\left(1 - \frac{50}{86}\right)}{86}}$$

$$0.4770 < p < 0.6855$$

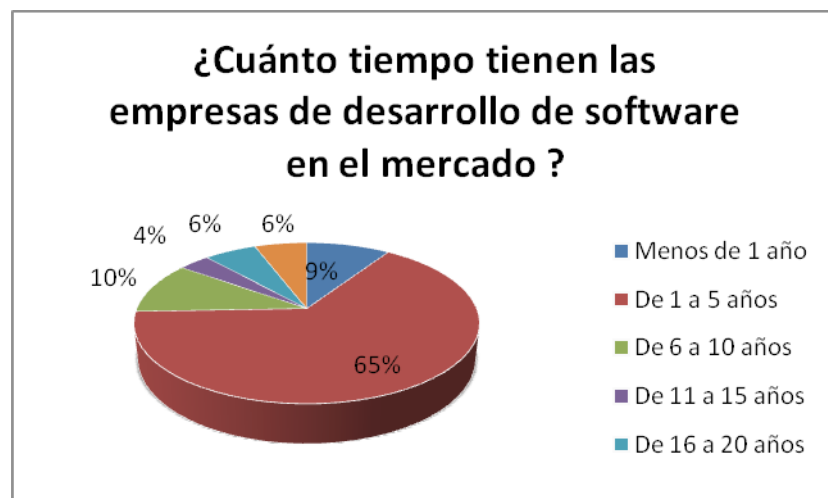
### Otros resultados obtenidos

Además de probar la hipótesis planteada, se obtuvo información muy valiosa con respecto a las empresas dedicadas al desarrollo de software en México, como la siguiente:

#### Información

La mayoría de empresas dedicadas a desarrollar software en México son aún muy jóvenes, como se puede observar en la figura 3. El 65% de las empresas tienen entre 1 y 5 años de antigüedad.

#### Gráfica



**Figura 3.** Tiempo en el mercado de las empresas



Los proyectos de software que más se desarrollan son sistemas multimedia y Web. Véase la figura 4.

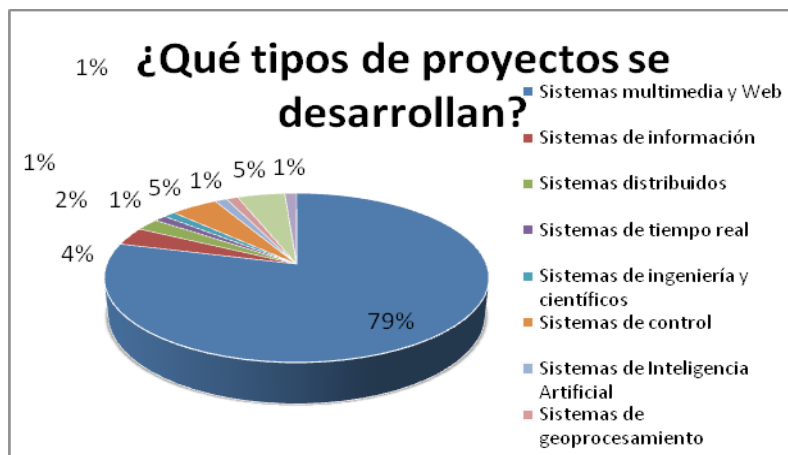


Figura 4. Tipos de proyectos que se desarrollan

El tiempo de desarrollo de la mayoría de proyectos oscila entre 2 y 3 meses, tal como se muestra en la figura 5.

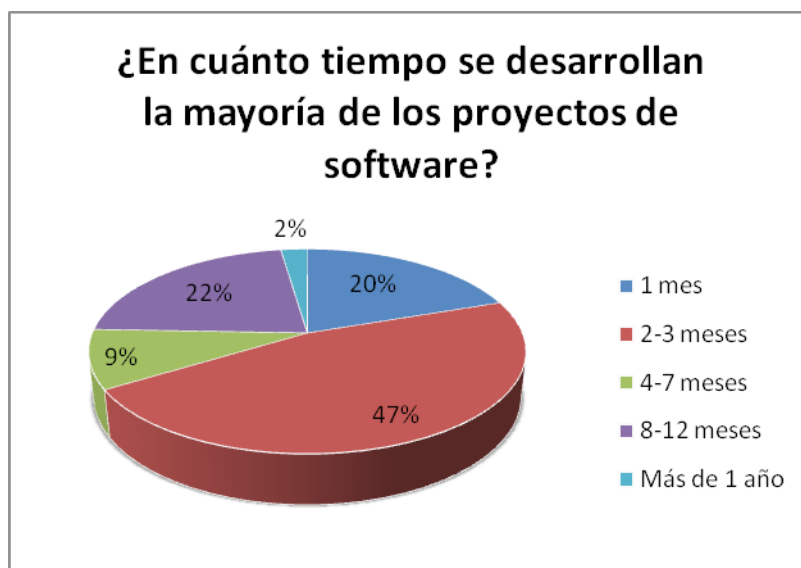


Figura 5. Tiempo de desarrollo de software

En la mayoría de los desarrollos de software sí se sigue un plan de proyecto que sirve como guía, como se observa en la figura 6.

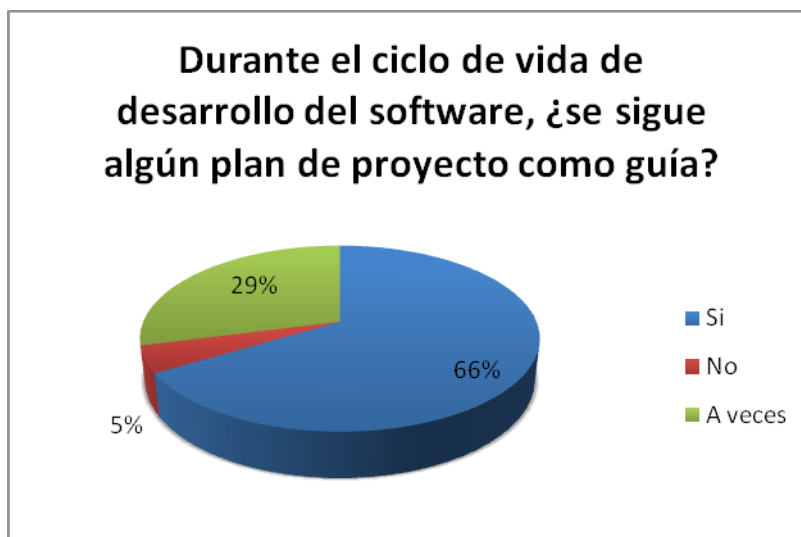


Figura 6. Uso de un plan de proyecto durante el desarrollo de software

En la figura 7 se observa que, la mayoría de las veces, no se realiza un estudio de factibilidad antes de iniciar el desarrollo del software.

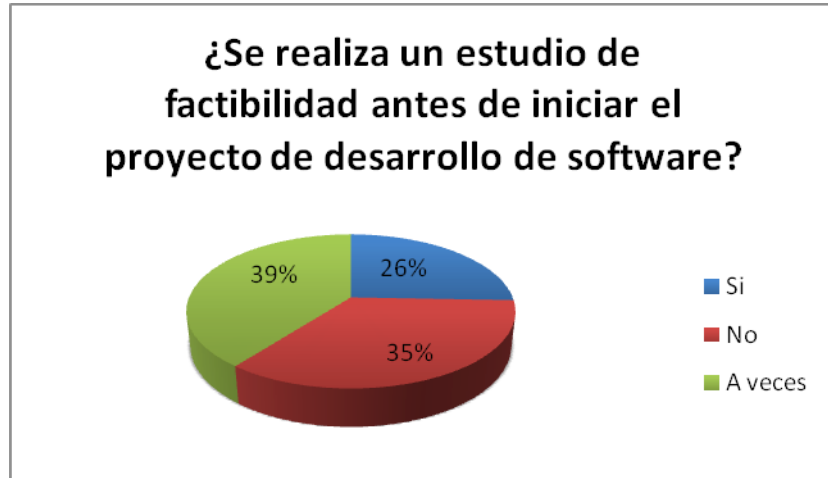


Figura 7. Realización de estudios de factibilidad

En la figura 8 se puede observar que las empresas dicen que sus clientes prefieren que se les entregue poca documentación, pero que se reduzca el tiempo de desarrollo.

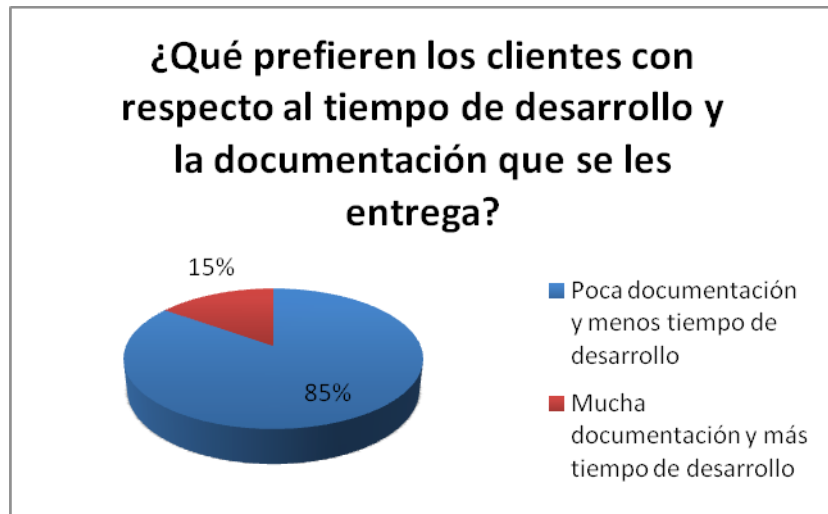


Figura 8. Documentación vs. Tiempo de desarrollo

En la mayoría de las empresas se tiene la filosofía de consultar al cliente sólo después de cada incremento, tal como se muestra en la figura 9.

Sólo una empresa de las encuestadas respondió otra de las opciones, ya que consulta a diario a sus clientes.

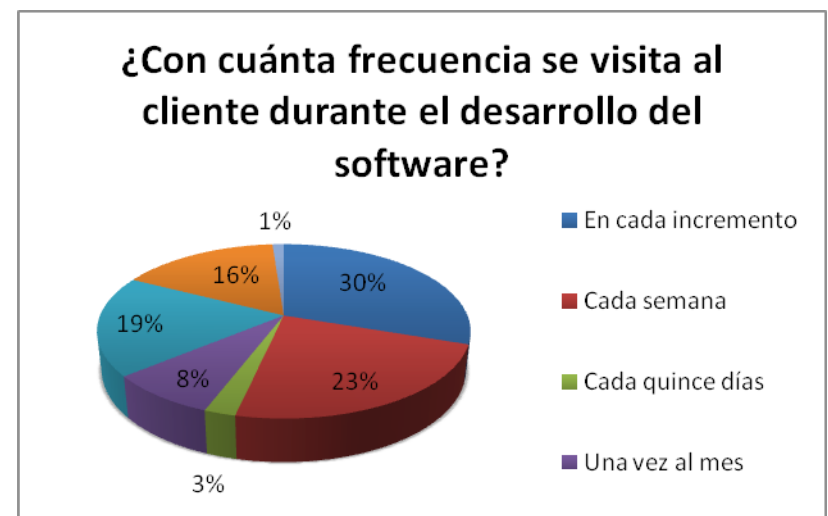


Figura 9. Visitas al cliente

Las empresas de desarrollo de software en México no sólo son jóvenes, sino que en los equipos de desarrollo se cuenta con pocos integrantes, tal como se puede observar en la gráfica de la figura 10.

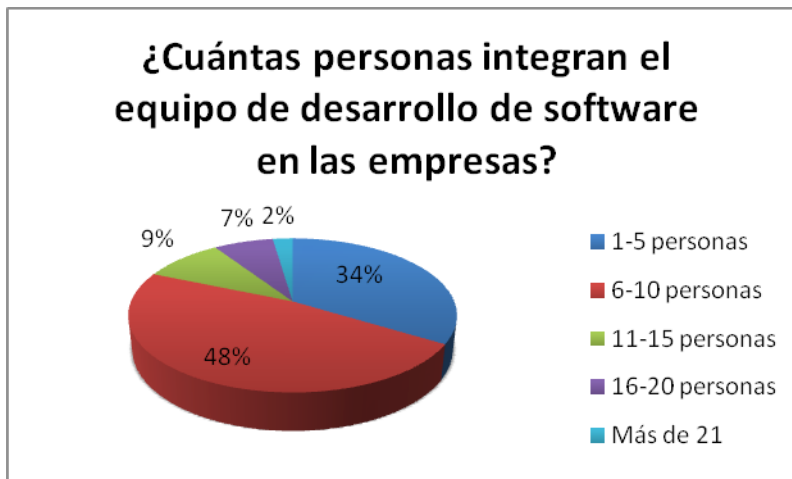


Figura 10. Número de integrantes del equipo de desarrollo

El paradigma de programación que más se utiliza en las empresas de desarrollo de software en México, es el Orientado a Objetos (véase figura 11).

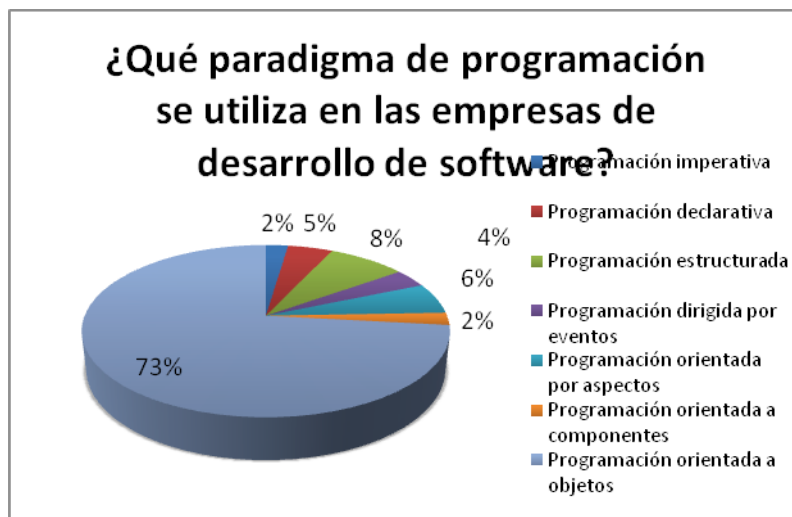


Figura 11. Paradigmas de programación utilizados en las empresas

La técnica de recopilación de requerimientos más utilizada es la lluvia de ideas (véase la figura 12).

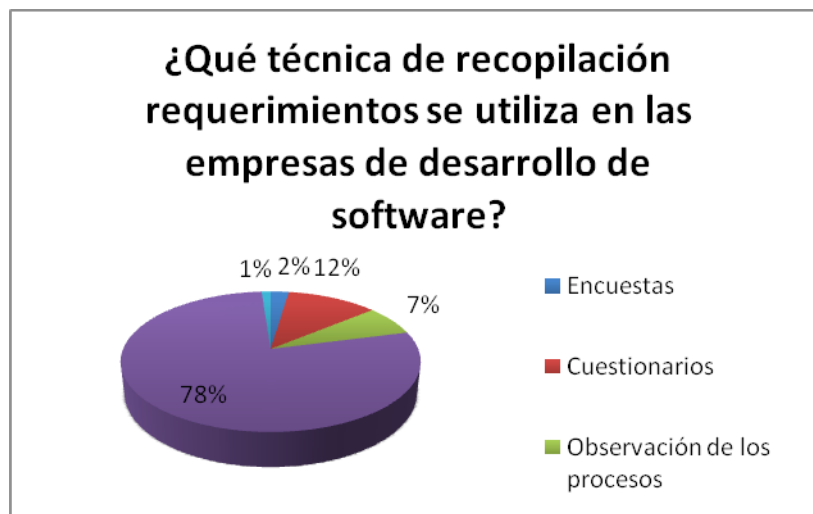


Figura 12. Técnicas de recopilación de requerimientos

En el 59% de las empresas se le asigna más del 50% a la etapa de codificación del software, como se puede observar en la figura 13.

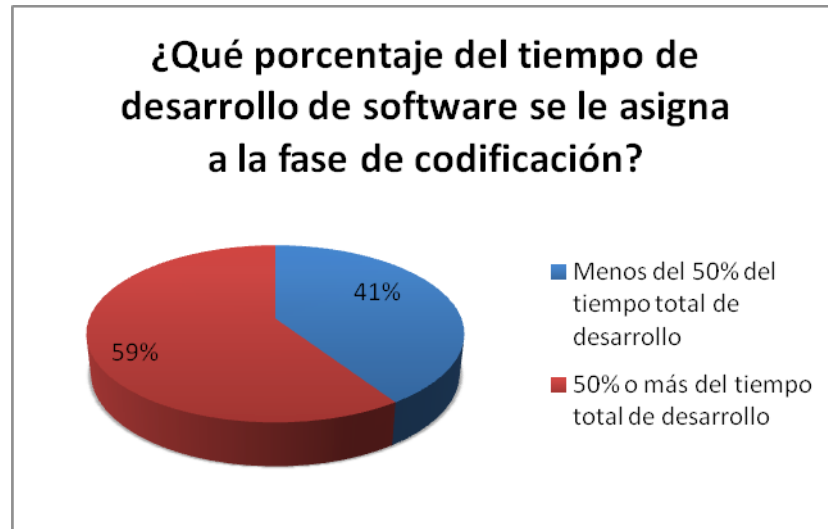


Figura 13. Empresas que le dedican más del 50% del tiempo a la codificación

Como se puede apreciar en la figura 14, la tres metodologías más utilizadas en México para desarrollar software son: RUP, XP y Scrum.



Figura 14. Metodologías de desarrollo de software utilizadas

En México el 82% de las empresas dedicadas a desarrollar software no cuentan con certificación en alguna metodología.

Y la metodología en la cual existen más empresas que cuentan con alguna persona certificada es Scrum, como se muestra en la figura 15.

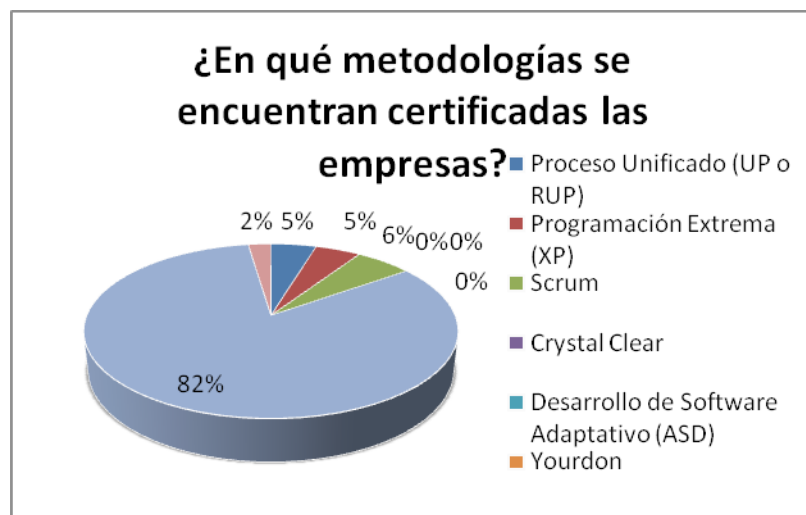


Figura 15. Empresas con certificación en alguna metodología

El tipo de pruebas que se aplican al software son de caja negra en el 88% de las empresas encuestadas (véase figura 16)

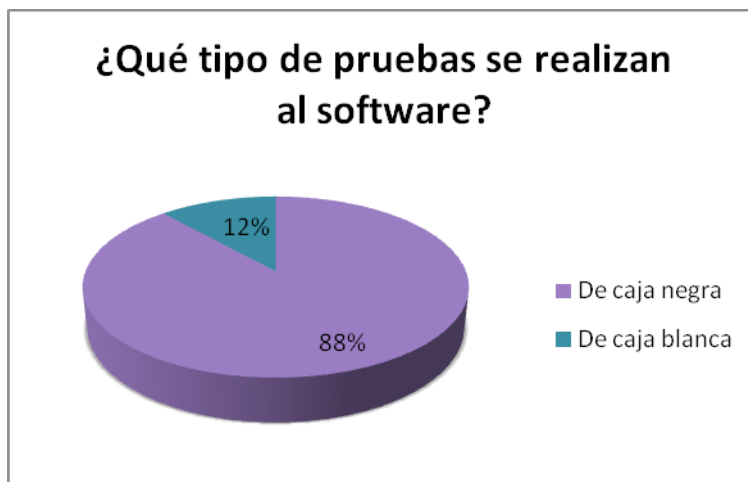


Figura 16. Pruebas aplicadas al software

En el 87% de las empresas no se hace uso de algún modelo de gestión y aseguramiento de la calidad (véase figura 17). Y de las empresas que aplican alguno el más utilizado es el ISO 9000-2000.

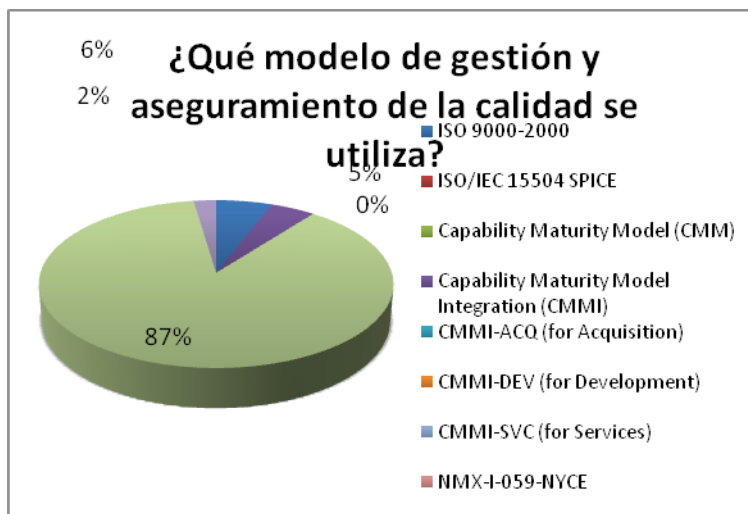


Figura 17. Modelos de gestión y aseguramiento de la calidad utilizados

Dos de las empresas encuestadas señalaron que hacen uso de otro modelo, una de ellas utiliza ITIL y la otra Lean&Agile.

En la figura 18, se muestra que la mayoría de empresas utiliza UML para modelar el software.

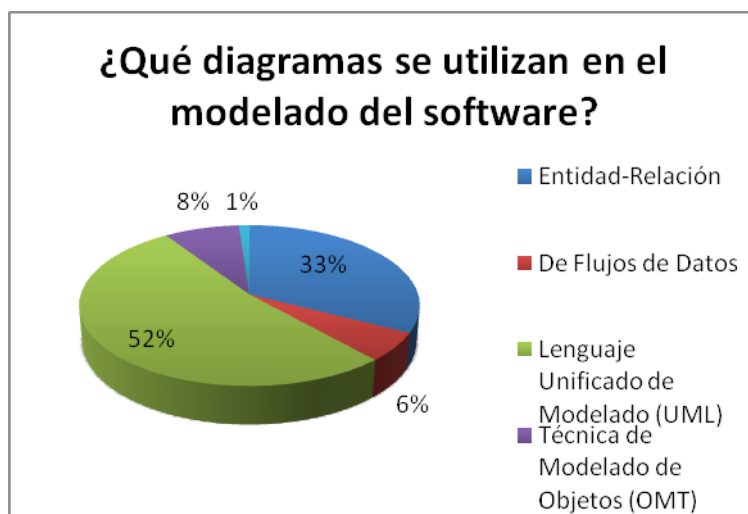


Figura 18. Diagramas utilizados para el modelado.

En la figura 19 se puede observar que el 61% de las empresas entrega al cliente, como parte de la documentación, sólo el manual de usuario.

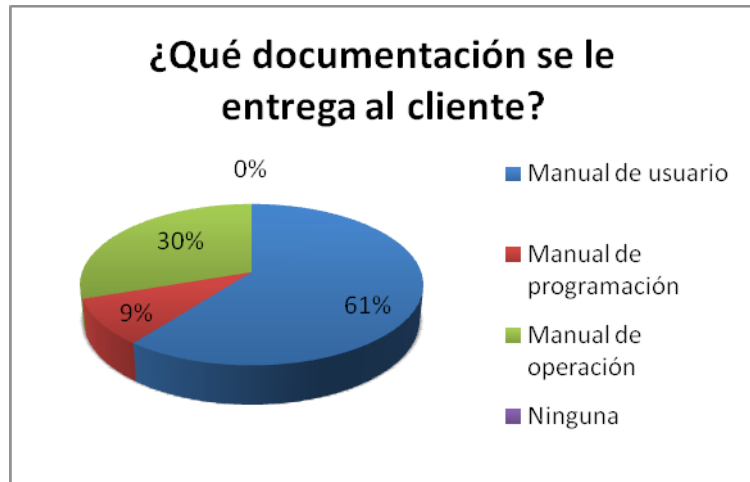


Figura 19. Documentación entregada al cliente

En el 52% de las empresas dedicadas a desarrollar software en México se hace uso de la Gestión de Riesgos o Problemas, siendo así la gestión más utilizada (véase figura 20).



Figura 20. Gestiones utilizadas durante el desarrollo de software

## Conclusiones

Se puede observar en los resultados de la investigación que las empresas que se dedican a desarrollar software en México, no solamente son candidatas para usar metodologías híbridas, sino que seguramente también obtendrán mejores resultados al desarrollar software con una metodología híbrida.

Es recomendable que se diseñe y utilice una metodología híbrida para el desarrollo de software, que integre prácticas de Ingeniería de Software de las tres metodologías más usadas por las empresas en México: RUP, XP y Scrum, tomando los elementos que proporcionen mayores ventajas a las empresas, de acuerdo a las características del software a desarrollar, de los desarrolladores, hardware y tiempo requerido para el desarrollo, entre otros elementos.

Por lo general los equipos desarrolladores de software en México están integrados por alrededor de 10 elementos; el tiempo de desarrollo del software es de 2 a 3 meses, y el software desarrollado

son aplicaciones Web. Estas características, de acuerdo a la investigación, permiten poner en práctica una metodología híbrida, pero esto no implica que dicha metodología no se pueda aplicar a grandes consorcios de desarrollo de software, ya que tomando en cuenta las características propias de la empresa y personal de desarrollo, se puede poner en práctica una metodología híbrida que seguramente dará buenos resultados.

En México, por lo general, las empresas desarrolladoras de software no realizan un estudio de factibilidad y no utilizan un modelo de gestión y aseguramiento de la calidad. Seguramente una metodología híbrida bien analizada, discutida y diseñada para tomar de los distintos modelos de desarrollo de software las características más ventajosas, ayudará a estas empresas a elaborar mejores planes de desarrollo del software, que permitan una mejor calidad de sus productos a menor costo y con un tiempo de desarrollo menor.

En esta investigación se puede observar que una metodología híbrida será de gran ayuda para las empresas desarrolladoras de Software de la República Mexicana, pero dadas las similitudes de México con los países latinoamericanos, seguramente estos resultados se pueden extrapolar también para esos países, con las respectivas diferencias propias de cada una de esas naciones, empresas de desarrollo de software y grupos de trabajo.

En resumen esta investigación muestra que es factible utilizar una metodología híbrida para desarrollar software en México. Además, proporciona información que puede ser útil para el diseño de la misma, o bien, al tratarse de prácticas existentes dentro de las empresas dedicadas a desarrollar software, esta información puede ser empleada para otras investigaciones relacionadas con el área de Ingeniería de Software en México.

## Bibliografía

Beck, Kent. *Extreme programming (XP): a gentle introduction*. USA: Mc Graw Hill, 2010.

Diccionario de la Real Academia Española. "Metodología".  
<[http://buscon.rae.es/draeI/SrvltConsulta?TIPO\\_BUS=3&LEMA=cultura](http://buscon.rae.es/draeI/SrvltConsulta?TIPO_BUS=3&LEMA=cultura)> [Consulta: 3  
septiembre 2011]

IBM. "IBM Rational Unified Process (RUP)". <<http://www-01.ibm.com/software/awdtools/rup/>>  
[Consulta: 18 septiembre 2011]

INEGI. "Estadística". <<http://www.inegi.org.mx>> . [Consulta: Febrero 14, 2011]

Jacobson, Ivar. "The Essential Unified Process (EssUP)". <[http://www.ivarjacobson.com/process\\_improvement\\_technology/essential\\_unified\\_process\\_software/](http://www.ivarjacobson.com/process_improvement_technology/essential_unified_process_software/)> [Consulta: 5 septiembre 2011]

Object management group. "UML". <<http://www.uml.org>>. [Consulta: 14 septiembre 2011]

Piattini *et al.* *Análisis y Diseño de Aplicaciones Informáticas de Gestión*. Una perspectiva de Ingeniería de Software. España: Ra-Ma, 2007.

Pressman, Roger. *Software Engineering: a practitioner's approach*. USA: Mc Graw Hill, 2005.

Rumbaugh *et al.* *Object-Oriented Modeling and Design*. USA: Prentice-Hall, 1990.

Scrum group. “Scrum”. < <http://www.scrum.org/>>. [Consultado: 2 octubre 2011]

Yourdon, Edward. Techniques of Program Structure and Design. USA: Prentice-Hall, 1976.



