



UNIVERSIDAD TECNOLÓGICA DE LA MIXTECA

“Adaptación de las Metodologías Tradicionales Cascada y Espiral para la Inclusión de Evaluación Inicial de Usabilidad en el Desarrollo de Productos de Software en México.”

TESIS

**PARA OBTENER EL GRADO DE
MAESTRO EN MEDIOS INTERACTIVOS**

PRESENTA

ING. CARLOS GERARDO PRIETO ÁLVAREZ

DIRECTOR DE TESIS

DR. JOSÉ FIGUEROA MARTÍNEZ

CO-DIRECTOR DE TESIS

MC. MARIO ALBERTO MORENO ROCHA

HUAJUAPAN DE LEÓN, OAX.; 15 DE SEPTIEMBRE DEL 2015

Dedicatoria

A mis padres Ermita y Gerardo por ser mi más grande motivación.

A mi sobrino Eithan, que espero encuentre una pasión en la ciencia.

A mis amigos, por motivarme en este proceso.

Agradecimientos

A mi director de tesis, Dr. José Figueroa Martínez, por su paciencia, amistad, conocimientos y tiempo dedicado a este proyecto de investigación.

A mi madre, que siempre me apoyó durante este proceso, por ser un ejemplo para mí y por siempre ayudarme a cumplir mis metas.

A mi padre, por ser una inspiración en mi trabajo, apoyarme cuando más lo necesito y siempre ver por mi salud.

Al profesor y amigo Mario Alberto Moreno Rocha y la profesora Iliana Herrera Arellano, por brindarme su apoyo y conocimientos desde el primer día que llegue a esta universidad.

A Itzel Eliasub Vázquez Osorio y familia por su amistad y compañía desde mi llegada a la universidad.

A la Universidad Tecnológica de la Mixteca, por proporcionarme las facilidades y apoyo económico durante mi estancia en este campus.

Contenido

Dedicatoria	3
Agradecimientos	4
Índice de figuras	8
Índice de tablas	9
Glosario	12
Introducción	13
1 Resumen del Protocolo de Tesis	14
1.1 Definición del problema	14
1.2 Justificación	15
1.3 Hipótesis	16
1.4 Objetivo general	16
1.5 Objetivos específicos	16
1.6 Alcances y limitaciones	16
1.7 Metas	17
1.8 Metodología.....	18
2 Marco Teórico	22
2.1 Principales metodologías en la industria de software.....	22
2.2 Metodología en Cascada.....	23
2.3 Metodología en Espiral.....	23
2.4 Acerca del costo total de un producto de software	24
2.4.1 Eficiencia en las metodologías tradicionales.....	27
2.5 Conceptos y pruebas de Usabilidad.....	28
2.6 Industria del software en México	31
2.7 Eficiencia, Eficacia y Satisfacción	32
2.8 Modelo de aceptación tecnológica por parte del usuario (TAM).....	33
2.9 Adaptaciones a los modelo actuales	33
2.10 Conclusiones del Marco Teórico.....	34
3 Desarrollo de la investigación.....	36
3.1 Prácticas de Usabilidad dentro de las empresas en México	36
3.2 Análisis inicial de la información y estudio con empresas en México.....	37

3.3 Estructura del problema.....	43
3.4 Diseño de la adaptación.....	43
3.4.1 Adaptación de las tareas de Usabilidad dentro del modelo en Cascada	45
3.4.2 Adaptación de las tareas de Usabilidad al modelo en Espiral.....	50
3.5 Diseño de la herramienta prototipo	56
3.5.1 Diseño de Interacción.....	57
3.5.2 Descripción de casos de uso.....	58
3.5.3 Wireframes iniciales de la herramienta	60
3.5.4 Diagrama de la base de datos y modelo relacional	66
3.5.5 Evaluación inicial de la herramienta	68
3.6 Versión funcional del prototipo.....	69
3.6.1 Descripción de los usuarios objetivo.....	74
3.6.2 Diseño de la interfaz de usuario.....	75
3.7 Diseño de las pruebas al modelo	75
4 Resultados y análisis experto.....	77
4.1 Primera Prueba: Intel Jalisco	78
4.1.1 Interpretación de resultados.....	81
4.2 Segunda Prueba: IBM Ciudad de México	81
4.2.1 Interpretación de resultados.....	84
4.3 Tercera Prueba: Desarrollo UCOL.....	85
4.3.1 Interpretación de resultados.....	88
4.4 Cuarta Prueba: Desarrollo Oaxaca	88
4.4.1 Interpretación de resultados.....	90
4.5 Quinta Prueba: Desarrollo Monterrey	90
4.5.1 Interpretación de resultados.....	92
4.6 Sexta Prueba: Desarrollo Puebla	93
4.8 Estadística.....	95
4.9 Diferencias en el proceso.....	97
5 Conclusiones y Anotaciones.....	100
5.1 Comprobación de la Hipótesis.....	100
5.2 Anotaciones	100
5.3 Implicaciones Teóricas y Políticas	101
5.4 Discrepancias.....	101

5.5 Confidencialidad.....	102
5.6 Trabajos Futuros.....	102
5.6.1 Capacitación.....	103
5.6.2 Recopilación de información.....	103
5.7 Limitaciones Del Estudio.....	103
5.8 Acuerdo de Confidencialidad.....	104
ANEXO A.....	105
ANEXO B.....	108
ANEXO C.....	110
ANEXO D.....	111
Bibliografía.....	112

Índice de figuras

Figura 1.1.- Procedimiento a Seguir en la Metodología.	19
Figura 1.2.- Fases del Diseño Centrado en Usuario Versión Extendida.	20
Figura 2.1- Modelo en Cascada y su Relación con una Base de Tiempo.	23
Figura 2.2- Modelo en Espiral.	24
Figura 2.3- Gráfica con el costo real de un producto de Software	25
Figura 2.4- Resultado de un estudio con más de 500 empresas a nivel mundial sobre metodologías utilizadas.	26
Figura 2.5- Gráfica de costos aproximados durante un proyecto de software.	27
Figura 2.6- Resultado de un estudio con más de 500 empresas a nivel mundial sobre metodologías utilizadas.	28
Figura 2.7: Obtención de información utilizando el modelo TAM.	33
Figura 2.8- Comparación entre el Modelo Lean UX y Diseño Centrado en el Usuario.	34
Figura 3.1: Resultado de las diferentes etapas en las que se participa.	38
Figura 3.2: Resultados de las etapas con mayores dificultades.	39
Figura 3.3: Principales medios utilizados para capacitación y entrenamiento.	39
Figura 3.4: Principales metodologías utilizadas.	40
Figura 3.5: Acercamiento directo con el usuario.	41
Figura 3.6: Principales tareas de Usabilidad conocidas.	42
Figura 3.7: Importancia del acercamiento con el usuario.	42
Figura 3.8: Estructura tradicional del modelo Cascada.	44
Figura 3.9: Etapas de modelo en Cascada y sus tareas mínimas por etapa.	44
Figura 3.10: Tareas básicas de Usabilidad en conjunto con las tareas del modelo Cascada.	45
Figura 3.11: Adaptación de Tareas de Usabilidad en la etapa de requerimientos.	46
Figura 3.12: Adaptación de tareas de Usabilidad en la etapa de análisis y diseño.	47
Figura 3.13: Adaptación de tareas de Usabilidad a la etapa de implementación.	48
Figura 3.14: Adaptación de tareas de Usabilidad dentro de la etapa de integración.	49
Figura 3.15: Adaptación de tareas de Usabilidad a la etapa de operación y mantenimiento.	49
Figura 3.16: Cuadrantes y etapas del modelo en Espiral. (Boehm, 1983).	51
Figura 3.17: Adaptación de tareas de Usabilidad a la etapa de identificación de Objetivos.	52
Figura 3.18: Adaptación de tareas de Usabilidad a la etapa de análisis de riesgo.	53

Figura 3.19: Adaptación de tareas de Usabilidad a la etapa de desarrollo del producto-----	54
Figura 3.20: Adaptación de tareas de Usabilidad a la etapa de operación y mantenimiento.----	55
Figura 3.21: Características del dispositivo requerido y opciones de la herramienta.-----	56
Figura 3.22: Diagrama UML de interacción.-----	58
Figura 3.23: Pantalla inicial y Panel de proyectos.-----	61
Figura 3.24: Registro de un nuevo proyecto y vista de las etapas.-----	62
Figura 3.25: Vista de las tareas dentro de la etapa.-----	63
Figura 3.26: Adaptación de tareas de Usabilidad a la etapa de operación y mantenimiento.----	64
Figura 3.27: Vista de un proyecto con modelo en Espiral.-----	65
Figura 3.28: Visualización de una segunda iteración y miembros del proyecto.-----	66
Figura 3.29: Modelo relacional propuesto para la aplicación.- -----	67
Figura 3.30: Jerarquía de la información dentro del modelo relacional.-----	68
Figura 3.31: Pruebas de prototipos de baja fidelidad con usuarios.-----	69
Figura 3.32: Pantalla de inicio de sesión.-----	70
Figura 3.33: Panel de proyectos activos y avances.-----	71
Figura 3.34: Modelo dentro del proyecto seleccionado.-----	72
Figura 3.35: Panel de tareas de Usabilidad por hacer.-----	73
Figura 3.36: Gestos considerados para moverse en pantalla.-----	74
Figura 3.37: Diseño de la prueba a la adaptación usando la herramienta.-----	76
Figura 4.1: Interfaz de acceso para el grupo 1.-----	78
Figura 4.2: Interacción con el menú de registro para el grupo 1.-----	79
Figura 4.3: Diseño de la interfaz de acceso para el grupo 2-----	80
Figura 4.3: Mensaje de confirmación para el grupo 2.-----	80
Figura 4.5: Diseño de la interfaz para el grupo 1.-----	82
Figura 4.6: Interfaz principal para el Grupo 2.-----	83
Figura 4.7: Mensaje de aviso sobre un proceso en espera. -----	83
Figura 4.8: Formulario de inicio de sesión. -----	84
Figura 4.9: Formulario de inicio de sesión. -----	85
Figura 4.10: Formulario de registro. -----	86
Figura 4.11: Formulario de inicio de sesión. -----	87
Figura 4.12: Formulario con validación. -----	87

Figura 4.13: Formulario de inicio de sesión. -----	88
Figura 4.14: Formulario de inicio de sesión. -----	89
Figura 4.15: Formulario de inicio de sesión. -----	91
Figura 4.16: Formulario de inicio de sesión. -----	92
Figura 4.17: Formulario de inicio de sesión. -----	93
Figura 4.18: Formulario de inicio de sesión. -----	94
Figura 4.19: Grafica comparativa con los resultados de Eficiencia. -----	95
Figura 4.20: Grafica comparativa con los resultados de Eficacia. -----	96
Figura 4.21: Grafica comparativa con los resultados de Satisfacción. -----	96
Figura 4.22: Grafica comparativa con los resultados de TAM (Technology Acceptance Model). -----	97
Figura 4.23: Comparación entre el proceso seguido por los grupos siguiendo la <u>adaptación</u> y los grupos con documentación libre. -----	99

Índice de tablas

Tabla 1: Obtención de información utilizando el modelo TAM. -----	21
Tabla 2: Niveles Esperados para Utilidad percibida y Percepción de Facilidad de uso. -----	21
Tabla 3: Validación por Heurísticas de Nielsen. -----	30
Tabla 4: Resultado de la Encuesta sobre medios apropiados para el prototipo. -----	56
Tabla 5: Porcentajes Obtenidos en la Evaluación Intel 1. -----	79
Tabla 6: Porcentajes Obtenidos en la Evaluación Intel 2. -----	81
Tabla 7: Porcentajes Obtenidos en la Evaluación IBM 1. -----	82
Tabla 8: Porcentajes Obtenidos en la Evaluación IBM 2. -----	84
Tabla 9: Porcentajes Obtenidos en la Evaluación UCOL 1.-----	86
Tabla 10: Porcentajes Obtenidos en la Evaluación UCOL 2. -----	87
Tabla 11: Porcentajes Obtenidos en la Evaluación Oaxaca 1. -----	89
Tabla 12: Porcentajes Obtenidos en la Evaluación Oaxaca 2. -----	89
Tabla 13: Porcentajes Obtenidos en la Evaluación Monterrey 1.-----	91

Glosario

Producto de Software: Se define como un producto diseñado para un usuario. Este producto debe hacer uso de recursos de cómputo y puede medirse de acuerdo a la cantidad de elementos que lo componen (IEEE,1991).

Sistema de Software: Se define como un conjunto de productos de software bajo el mismo objetivo de cumplir una función de cómputo. Para que un producto de software sea considerado un sistema, debe contar con al menos dos o más componentes que hagan uso de los recursos nativos del equipo de cómputo.(IEEE,1991).

Introducción

Desde los inicios del desarrollo de software, hace más de 40 años, los costos que representa la producción de software en pequeña y gran escala son uno de los factores por los cuales muchas empresas optan por tener su propio departamento de desarrollo de herramientas de software. (Bevan, 2008)

La creación de metodologías de desarrollo que permitan una mejor calidad en los productos de software ha creado una gran competencia entre las empresas desarrolladoras de software, casos como la creación de metodologías ágiles dentro de grandes empresas ha dado la idea de que aquellos grupos de desarrollo que utilicen una metodología optimizada para desarrollar software son los que prevalecerán y generan más ganancias en los mercados de desarrollo.

Actualmente existen diversas metodologías de las cuales el equipo de desarrollo puede elegir dependiendo las necesidades del proyecto, sin embargo, muchas empresas optan por apegarse a metodologías tradicionales como los son Cascada y Espiral, el problema comienza cuando se quieren aplicar nuevas herramientas las cuales no son del todo compatibles con estos modelos.

En México existe un gran número de empresas que desean mejorar sus procesos de desarrollo pero no tienen la capacidad necesaria para utilizar modelos completamente nuevos por lo que requieren hacer uso de metodologías tradicionales modificadas para su beneficio.

La inclusión de un módulo de evaluación de Usabilidad podría traer grandes beneficios para la calidad del producto (Nielsen, 1994) y aminorar costos en mantenimiento. Es por eso que es importante generar cambios en las metodologías tradicionales para incluir esta evaluación sin crear problemas dentro del manejo de recursos, tiempo y presupuesto del proyecto.

Para poder adaptar los conceptos de Usabilidad a los modelos tradicionales es necesario utilizar una herramienta adecuada que permita la modificación de tareas en las diferentes etapas del desarrollo con el fin de tener un seguimiento continuo de los cambios. (Norman D. A., 2010).

En este trabajo se presentan algunos indicadores sobre de las metodologías de desarrollo actuales utilizadas dentro de las principales empresas de software producto de una encuesta selectiva, el análisis de artículos recopilatorios en la materia y finalmente el diseño de una herramienta que contemple la adaptación de conceptos de Usabilidad a las metodologías tradicionales para la obtención de mayores niveles de aceptación final del producto por parte del usuario.

1 Resumen del Protocolo de Tesis

1.1 Definición del problema

Teniendo en cuenta toda esta información y tomando como referencia la manera en que los sistemas de pruebas han ido evolucionando (Meerts, 2013), la falta de pruebas de calidad para la obtención de niveles de Usabilidad principalmente Eficiencia, Eficacia y Satisfacción referente al producto final durante el proceso de desarrollo, es parte importante del problema relacionado al número de proyectos terminados con deficiencias, mayormente en la parte de aceptación final del producto por parte del usuario , ya que analizando las razones por las que se crearon otros modelos como XP (Extreme Programming) y Lean, es necesario tener en cuenta al usuario desde el diseño inicial. (J. Coplien, 2010; G. Succi & M. Marchesi, 2001; K. Pries & J. Quigley, 2011).

Realizando pruebas con diferentes empresas del área de tecnologías de la información, se ha encontrado que el uso de una metodología de apoyo, como lo es el Diseño Centrado en el Usuario, disminuye el índice de errores relacionados al diseño de las interfaces lo cual se traduce en mejores niveles en facilidad de uso, Eficiencia, Eficacia y Satisfacción. Esto ha llevado a distintos investigadores en el área de Interacción Humano-Computadora a buscar distintas maneras de implementar tareas relacionadas a la Usabilidad dentro de los modelos tradicionales, todo esto sin dejar fuera o modificando el objetivo principal de cada etapa(Weinschenk, 2012).

Una prueba básica de Usabilidad que contenga pruebas con usuarios, análisis de tareas, especificación de usuarios y resultados estadísticos iniciales con base en cuestionarios TAM (Technology Acceptance Model) proporciona información suficiente para mejorar el diseño y aminorar los errores en etapas posteriores de evaluación, el problema se presenta cuando no se hace una correcta adaptación de este procedimiento a la metodología utilizada por lo que los tiempos disponibles para la evaluación es demasiado corto (Krug, 2006).

Es por todo lo anterior que el presente avance del proyecto de tesis estará orientado a proponer una adaptación a las metodologías tradicionales de Cascada y Espiral, considerando aspectos de Usabilidad (Eficiencia, Eficacia y Satisfacción) que vendrían a contrarrestar la falta de inclusión del usuario durante el proceso de desarrollo de productos de software.

Basado en nuestra investigación inicial podemos plantear las siguientes preguntas de investigación:

- ¿Es posible incluir módulos de evaluación de Usabilidad en metodologías tradicionales como Cascada y Espiral?

- ¿Puede una adaptación de evaluación de Usabilidad a los modelos tradicionales Cascada y Espiral mejorar la calidad del producto en aspectos de facilidad de uso, Eficiencia, Eficacia y Satisfacción?

1.2 Justificación

De lo explicado anteriormente, el problema con la falta de una adaptación apropiada para la inclusión de pruebas de Usabilidad en los modelos tradicionales Cascada y Espiral está costando recursos importantes a las empresas que todavía trabajan con estas metodologías (Weinschenk, 2012). Incluir conceptos de pruebas con usuarios en otras metodologías como lo es el Diseño Centrado en el Usuario ha traído beneficios en la calidad final del producto en cuanto a métricas de Eficiencia, Eficacia y Satisfacción, reducción en la corrección de errores de diseño y beneficios en la aceptación final del producto por parte del usuario (Weinschenk, 2012).

En los últimos 10 años, se han realizado varias adaptaciones a diferentes modelos de desarrollo, principalmente a modelos ágiles. (Coplien, J. 2010), (Succi, G., & Marchesi, M. 2001), (Pries, K., & Quigley, J. 2011). Tomando en cuenta la cantidad de problemas existentes explicados anteriormente, el único acercamiento que se realiza actualmente a una adaptación de conceptos de Usabilidad en modelos tradicionales es la realización de pruebas de Usabilidad al final del proceso de desarrollo que corresponde a la parte de mantenimiento. Esto como se explica brevemente en el marco teórico, sólo corrige problemas de diseño y no cumple con una parte muy importante que es la prevención temprana de errores.

Actualmente dentro del mercado de productos de software para pruebas podemos encontrar distintas alternativas funcionales para los diferentes tipos de procedimientos existentes. Estos diferentes productos de software varían según el instrumento que se desea utilizar, la etapa en el desarrollo en la que se encuentra el producto y el tipo de parámetros que se buscan medir.

Para poder medir parámetros como la cantidad de métricas heurísticas nivel de retención del usuario y niveles de Usabilidad, es necesario contar con herramientas que nos proporcionen una manera de obtener información por parte de los usuarios además de una herramienta estadística apropiada que permita compartir los resultados obtenidos con los diferentes miembros del equipo de desarrollo durante el proceso (Seffah, 2005).

1.3 Hipótesis

Es posible la inclusión de conceptos y estudios de Usabilidad en las fases con mayor problema de los modelos de desarrollo tradicionales Cascada y Espiral para productos de software, a través de una herramienta que guíe a los desarrolladores en las adaptaciones propuestas?.

1.4 Objetivo general

Incluir tareas de Usabilidad en las etapas con mayor problema dentro de los modelos tradicionales de desarrollo de productos de software Cascada y Espiral a través de una herramienta que guíe a los desarrolladores durante su ejecución.

1.5 Objetivos específicos

1. Realizar un análisis inicial de las empresas dedicadas al desarrollo de productos de software en México.
2. Identificar las etapas en el desarrollo de software de los métodos tradicionales en donde se dan los problemas que afectan la calidad del producto final en cuanto a facilidad de uso, Eficiencia, Eficacia y Satisfacción.
3. Definir las tareas de Usabilidad mínimas para su uso dentro de cada etapa de las metodologías.
4. Modificar las tareas realizadas en las diferentes etapas del desarrollo de productos de software haciendo la adaptación adecuada de evaluación de Usabilidad en los métodos tradicionales.
5. Diseñar la versión prototipo de la herramienta de evaluación como resultado de la adaptación propuesta a los métodos tradicionales.
6. Comparar la herramienta propuesta en escenarios previos y posteriores a la adaptación realizada.
7. Evaluar los resultados obtenidos después de la adaptación propuesta.

1.6 Alcances y limitaciones

- El estudio inicial únicamente incluirá empresas en México con las que se tenga acercamiento principalmente México DF, Monterrey, Jalisco, Puebla, Colima y Oaxaca.

- Las pruebas de la adaptación se llevarán a cabo con pequeños grupos de desarrollo (menos de 6 integrantes).
- Debido a los tiempos para el desarrollo de una tesis de maestría en la institución, esta deberá estar concluida para el mes de Septiembre del 2015.
- Se utilizarán dos metodologías tradicionales como objeto de estudio, la metodología en Cascada y la metodología en Espiral.
- La adaptación a estas metodologías se presentará como una herramienta en su versión prototipo que será aplicada a dos grupos de desarrollo disponibles en las empresas con las que se tiene acercamiento.
- Los participantes de las pruebas pueden cambiar debido a la disponibilidad de los mismos y cambios en las fechas de entrega

1.7 Metas

- Aplicar dos cuestionarios (método Delphi) a un grupo empresas disponibles en México como acercamiento inicial para conocer la situación actual de estas empresas de desarrollo.
- Escribir un análisis de resultados a partir de las respuestas por parte de las empresas.
- Construir la adaptación a los modelos Cascada y Espiral generando un esquema visual donde se muestre el modelo original y el modelo adaptado.
- Construir el prototipo inicial de baja fidelidad de una aplicación para dispositivo móvil que contenga la guía de adaptación.
- Realizar el prototipo de alta fidelidad que contenga elementos funcionales sobre el dispositivo seleccionado.
- Realizar dos pruebas con el prototipo de alta fidelidad con 10 grupos de desarrollo de diferentes empresas.
- Analizar los resultados de las pruebas buscando que se cumpla un nivel mayor a 75% para los indicadores de Eficiencia y Eficacia.
- Escribir un reporte de los resultados de las pruebas de Usabilidad para los prototipos de baja y alta fidelidad.

1.8 Metodología

La primera parte de la metodología contempla el análisis de las técnicas actuales, recopilación de información respecto al problema y la especificación de las partes con las que se trabajará para la propuesta.

La primera parte del desarrollo de la investigación consiste en la realización de un cuestionario sobre el proceso actual llevado dentro de varias empresas en el ámbito de desarrollo de productos de software utilizando el método Delphi en al menos dos circulaciones. El estudio contempla principalmente empresas mexicanas en los estados de México DF, Monterrey, Jalisco, Puebla, Colima y Oaxaca. Del estudio se tomarán principalmente aquellas empresas que utilicen únicamente metodologías tradicionales en sus proyectos.

Para la segunda parte se planea utilizar estos datos para crear una adaptación al modelo actual de implementación de pruebas de Usabilidad, esto con el objetivo de crear una versión adaptada sin excluir alguna de las etapas que componen los modelos tradicionales Cascada y Espiral.

Partiendo del diseño inicial se desarrollará una herramienta prototipo en la que se explique cómo realizar la adaptación a los modelos tradicionales. Este prototipo se proporcionará a seis equipos de desarrollo donde los desarrolladores deberán cumplir un perfil básico como programadores. La herramienta será utilizada por la mitad de los equipos y los resultados finales serán medidos por expertos en el área de Usabilidad para determinar si el producto obtenido de la prueba demuestra un cambio significativo en su diseño.

Los resultados de la prueba serán analizados y se comprobará si la herramienta cumple con la hipótesis inicial además de si realmente tiene un beneficio significativo para ser implementado por las empresas mexicanas del ámbito de desarrollo de software que utilicen las metodologías tradicionales Cascada y Espiral.

Lo descrito anteriormente puede verse de manera gráfica en la Figura 1.1 de forma secuencial y con resultados entre cada una de las fases.

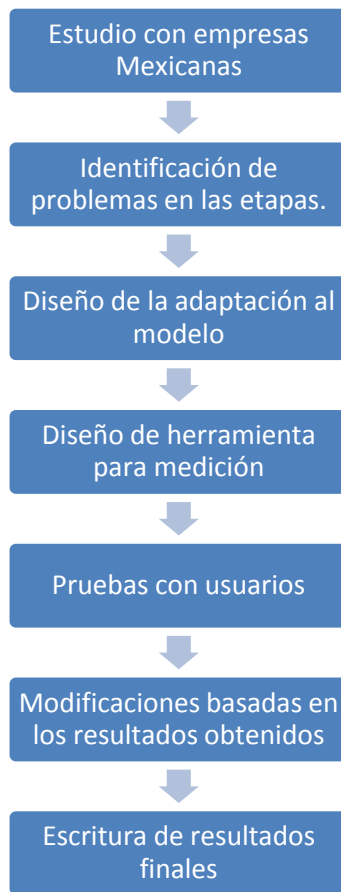


Figura 1.1.- Procedimiento a seguir en la Metodología.

La metodología contemplada para la realización de la herramienta de evaluación es el Diseño Centrado en el Usuario versión extendida (Harper, Rodden, Rogers, & Sellen, 2008). Esta metodología permitirá seguir los pasos y recomendaciones en el área de Usabilidad de manera que el producto tengas las especificaciones necesarias para facilitar el trabajo de adaptación al usuario.

Como podemos ver en la Figura 1.2 con esta metodología podemos iniciar con las fases de entendimiento y estudio del problema, en esta primera parte utilizará la literatura encontrada donde se describe parte de la problemática y se utilizaran resultados producto de un análisis con empresas en el área de desarrollo de software en México.

Una vez obtenidos estos datos podemos diseñar un primer prototipo el cual se construirá con características de baja fidelidad utilizando *Sketches* y *Wireframes*.

La construcción del primer prototipo se realizara en papel y se considerara una versión digital con el objetivo de ser probada de forma remota por algunos equipos de desarrollo.

Para la primera evaluación del prototipo se planea una primera simulación utilizando la técnica Mago de Oz utilizando un pequeño grupo de desarrollo como usuarios de prueba.

Para la segunda iteración se planea producir un prototipo de alta fidelidad, capaz de funcionar en un dispositivo Smartphone.



Figura 1.2.- Fases del Diseño Centrado en Usuario Versión Extendida. (Harper, Rodden, Rogers, & Sellen, 2008)

A partir de los resultados obtenidos del estudio inicial podemos utilizar estos datos para crear una adaptación al modelo actual de implementación de pruebas de Usabilidad, esto con el objetivo de crear una versión adaptada sin excluir alguna de las etapas que componen los modelos tradicionales Cascada y Espiral. Los elementos de Usabilidad utilizados están basados en la definición que la norma ISO 9241-11 proporciona. (Nielsen, 1994)

Partiendo del diseño inicial se desarrolla una herramienta prototipo en la que se explica cómo realizar la adaptación a los modelos tradicionales. Este prototipo se proporciona a los participantes en la parte de pruebas dentro de la investigación.

Los resultados de la prueba son analizados con el objetivo de comprobar si la herramienta cumple con la hipótesis inicial además de si realmente tiene un beneficio significativo para ser implementado por las empresas mexicanas del ámbito de desarrollo de software que utilicen las metodologías tradicionales Cascada y Espiral.

El manejo de estas variables permite conocer si la adaptación y la herramienta cumplen con el objetivo obteniendo niveles esperados como se muestra en la Tabla 1.

Tabla 1: Obtención de información utilizando el modelo TAM (Kowitlawakul, 2005).

Variable	Valor Esperado
Eficiencia(EF)	EF>75%
Eficacia(EFI)	EFI>75%
Satisfacción(SAT)	SAT>60%

Observando cómo los usuarios se enfrentan a tareas interactivas, podemos cuantificar objetivamente la Usabilidad del diseño, contabilizando el número de errores que cometen (Eficacia) o midiendo el tiempo que tardan en completarlas (Eficiencia). Además, preguntando a los usuarios una vez finalizadas sus tareas, podremos medir la Usabilidad subjetiva o percibida, es decir, cómo valoran los usuarios el diseño o cuál es su grado de Satisfacción.

Además de la obtención de variables referentes a Eficiencia Eficacia y Satisfacción, se realizará un cuestionario basado en el Modelo de Aceptación de la Tecnología (TAM) el cual permitirá medir la forma en la que los usuarios llegan a aceptar y utilizar el producto tecnológico resultante. (Kowitlawakul, 2005)

Para las variables de PU y FPU se esperan lo siguientes valores explicados en la siguiente tabla 2:

Tabla 2: Niveles esperados para Utilidad percibida y Percepción de facilidad de uso. (Kowitlawakul, 2005).

Variable	Valor Esperado
PU(Utilidad Percibida)	PU>75%
FUP(Percepción de facilidad de uso)	FUP>80%

2 Marco Teórico

En el presente capítulo se explican las bases teóricas para el entendimiento de este trabajo de investigación. Cada una de las secciones explica los detalles importantes considerados como relevantes para el entendimiento del problema. Las primeras tres secciones explican las principales metodologías existentes y cuáles de estas metodologías fueron seleccionadas como objetos de estudio.

En las siguientes secciones explica el concepto de Eficiencia en las metodologías, los principales puntos de una prueba de Usabilidad, algunos indicadores relevantes sobre la industria de software en México y finalmente las adaptaciones principales a las metodologías de software para la inclusión de conceptos de Usabilidad utilizadas actualmente en la industria.

2.1 Principales metodologías en la industria de software

Una metodología de software se puede explicar como un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo en sistemas de información (Pressman, 1995). La selección de una metodología de desarrollo varía dependiendo de los objetivos, tiempo y recursos con los que cuenta el equipo de desarrollo, tomando en cuenta estas características, desde 1970 se tienen antecedentes de los primeros modelos estandarizados para todo un proceso de desarrollo (Pressman, 1995).

Analizando las características principales de las principales metodologías utilizadas en la industria, podemos referirnos a ellas en dos grupos diferentes: Metodologías tradicionales y Metodologías Ágiles. Para las metodologías tradicionales tenemos los primeros modelos que ganaron popularidad por ser estructurados y centrarse principalmente en la planificación como lo son Cascada, Espiral, Prototipado, RUP (Rational Unified Process) y MSF (Microsoft Solution Framework (Lo, 2011).

Respecto a la metodologías ágiles podemos definir que principalmente se busca un modelo incremental con pequeñas entregas, una menor documentación y una actitud cooperativa entre cliente y desarrollador. Dentro de estas metodologías encontramos el Modelo SCRUM, LEAN, XP (Extreme Programming), ASD (Adaptative Software Development) y Crystal Clear (Lo, 2011).

Dentro de esta investigación nos centraremos en el estudio de dos de las metodologías tradicionales más usadas por los grupos de desarrollo de software: Modelo en Cascada y Modelo en Espiral. Como se explica más adelante, estas dos metodologías cuentan con ciertos lineamientos estructurados en las que posiblemente se pueda implementar conceptos de Usabilidad propios de nuevas metodologías.

2.2 Metodología en Cascada

El primer modelo relevante dentro de la industria de software es el Modelo en Cascada (Pressman, 1995) (denominado así por la posición de las fases en el desarrollo de esta, que parecen caer en Cascada “por gravedad” hacia las siguientes fases) propuesto en 1970, es el enfoque metodológico que ordena rigurosamente las etapas del proceso para el desarrollo de software, de tal forma que el inicio de cada etapa debe esperar a la finalización de la etapa anterior (Pressman, 1995).

Dentro de lo que podemos analizar en base a la experiencia de algunos autores en el área, este modelo es apropiado cuando el proyecto está relacionado con el desarrollo de *mainframes* o sistemas basados en transacciones. Además, de acuerdo a los principios por los cuales se formó la metodología, es eficiente para llevar control de largos presupuestos y se involucran grandes equipos de trabajo. (Pressman, 1995).

De acuerdo al investigador Winston Royce que propuso dicho modelo, los beneficios de esta metodología surgen cuando no existen fechas inmediatas de implementación por lo cual se tiene un tiempo apropiado para desarrollar cada fase. Es importante mencionar que para que este modelo tenga un menor índice de riesgo, es necesario que los requerimientos sean muy claros y se hayan establecido de manera oficial en la primera parte del proyecto Figura 2.1.

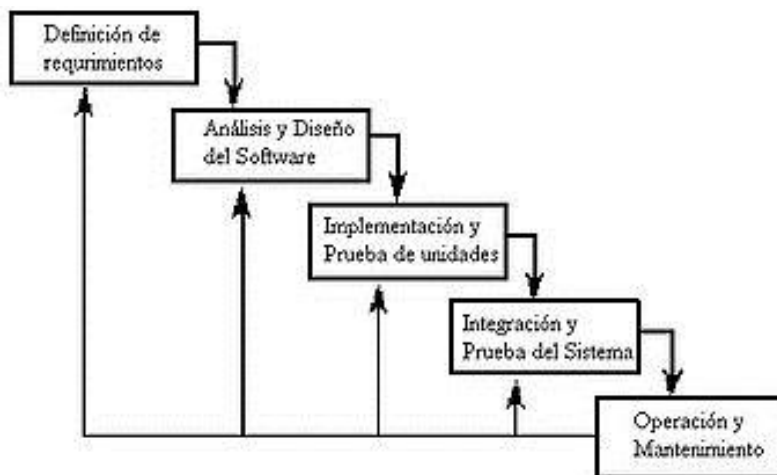


Figura 2.1- Modelo en Cascada y su relación con una base de tiempo. (Pressman, 1995)

2.3 Metodología en Espiral

La primera modificación importante a los modelos de desarrollo de software ocurrió en el modelo presentado por Barry Boehm en 1983 con su metodología conocida como Espiral (Boehm, 1983), cuyas fases se muestran en la Figura 2.2 Este modelo permitía analizar más a fondo cada una de las etapas comprendidas en el desarrollo de un producto de software y

más importante aún, los requerimientos debían ser conocidos mucho antes de su implementación. Esto trajo mejoras en cuanto a la aceptación del producto en el mercado por el usuario así como la adopción final, además, se produjo un recorte en los costos de mantenimiento en al menos un 20% por lo que muchas empresas comenzaron a utilizarlo dentro de sus modelos de desarrollo como lo fue IBM, HP y Xerox (Aspray, 1996).

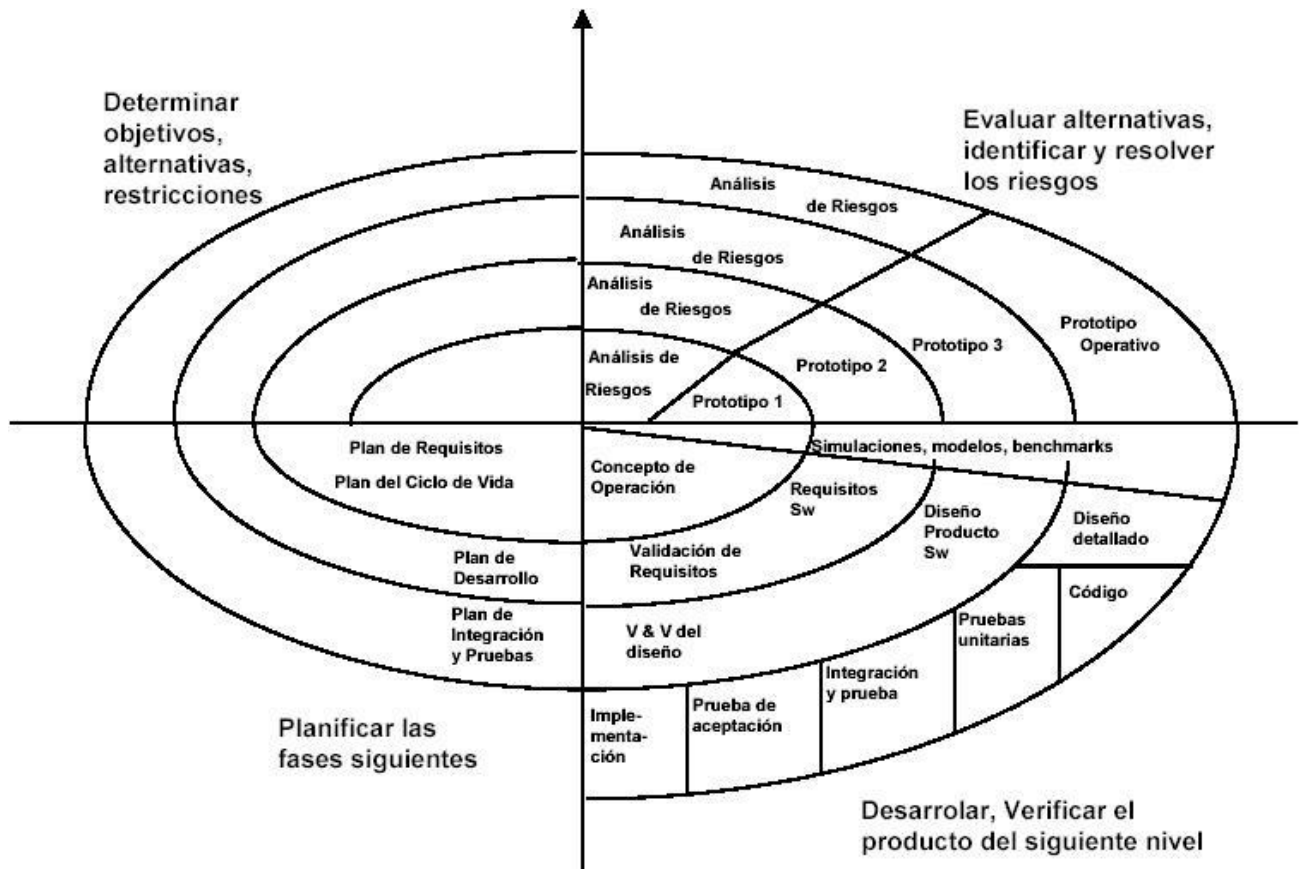


Figura 2.2- Modelo en Espiral. (Boehm, 1983)

Años más tarde continuaron apareciendo versiones modificadas del Modelo en Espiral, todas dependiendo de los objetivos del equipo de desarrollo. La flexibilidad de un modelo representa un factor importante a la hora de la creación de calendario de trabajo ya que ciertas etapas pueden requerir de una extensión como por ejemplo la extensión de una etapa de pruebas con el objetivo de corregir cierta cantidad de errores en etapas tempranas del desarrollo (Foster, 1993).

2.4 Acerca del costo total de un producto de software

Antes de seleccionar una metodología de desarrollo, es necesario conocer a fondo las características del proyecto. Con el tiempo se han ido creando nuevas guías para la selección y modificación de las metodologías.

Los primero en experimentar con modificaciones y modelos para la inclusión de conceptos de Usabilidad con un acercamiento diferente por parte de los desarrolladores experimentados fueron los investigadores Donald Norman y Stephen Draper, ellos propusieron una metodología capaz de contemplar el rol del usuarios en las diferentes etapas de desarrollo (Norman & Draper, 1986). Sin embargo, hasta ese momento se contaba con poca experimentación como para asegurar que esta metodología podía competir con otras más establecidas como la ya mencionada modelo en Espiral.

Durante los siguientes años las empresas más importantes de desarrollo de software a nivel mundial como lo fue IBM y Microsoft centraron su atención en la reducción de costos en los procesos de producción con el objetivo de poder crear programas de bajo costo para la venta a consumidores en sus hogares.

Tomando en cuenta el análisis realizado en 2010 sobre la alineación de tecnologías IT con los objetivos de la empresa (Murphy, 2010), para la parte del análisis del costo real de la producción de un producto de software, encontramos que finalmente el mantenimiento debía ser el principal problema a resolver como lo vemos en la Figura 2.3 ya que incluso el día de hoy, el mismo valor de un mantenimiento continuo supera a la producción inicial del producto tiene un gran impacto sobre el costo final del proyecto.

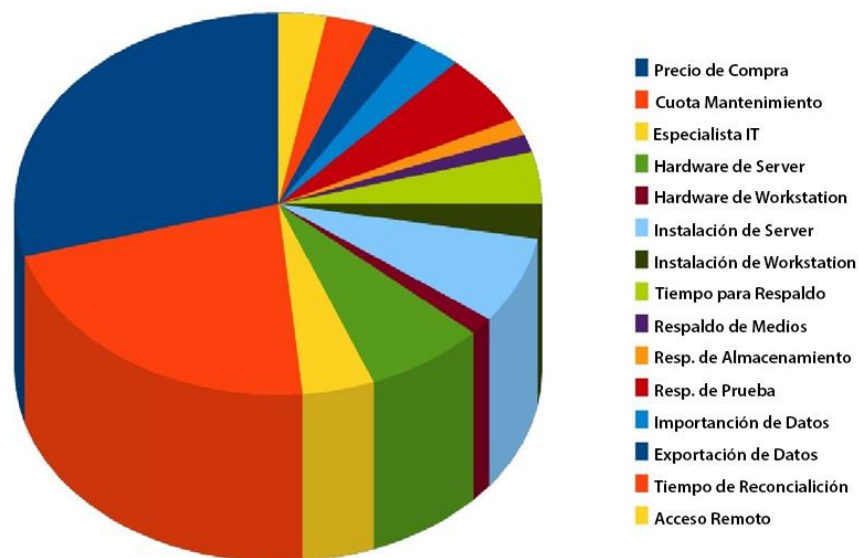


Figura 2.3- Gráfica con el costo real de un producto de software (Murphy, 2010).

La forma en la que se desarrolla software a nivel mundial varía mucho en contexto del país, tipo de empresa y demanda de tecnologías que se tengan en la población. Un ejemplo de esto lo podemos ver analizando el surgimiento de las metodologías ágiles en las que fue necesario crear nuevos modelos de desarrollo para satisfacer las nuevas necesidades de

software con entregas más claras para proyectos que requieran resultados en términos inmediatos (Young, 2011).

Como parte de una comparativa, a partir de un estudio donde los resultados se muestran en la Figura 2.4, podemos ver que el uso de las metodologías ágiles ya toma cierta porción como metodologías principales para el desarrollo pero el Modelo en Cascada sigue siendo la principal metodología utilizada por las empresas de desarrollo (Young, 2011).

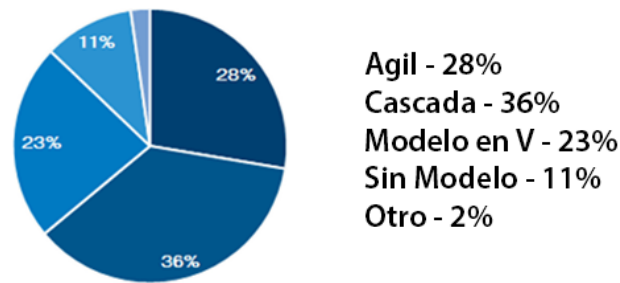


Figura 2.4- Resultado de un estudio con más de 500 empresas a nivel mundial sobre metodologías utilizadas. (Young, 2011)

Aunque existen otros tipos de metodologías recomendadas por organismos responsables de la revisión de estándares de calidad. Se prevé que su uso se vea reducido cada vez más como es el caso del Modelo en V. La razón de que algunos modelos no sean tan populares a nivel mundial podría radicar en el hecho de que no proporcionan buenos resultados si no se utilizan en conjunto con otros modelos de apoyo (Young, 2011).

Analizando las partes principales de proceso de desarrollo es en la parte de producción donde debemos crear una reducción del costo ya sea realizando modificaciones durante alguna de las etapas iniciales para poder evitar grandes cambios que alarguen la duración del proceso de desarrollo, algunos resultados de eso se pueden ver en la Figura 2.5. (Nasa Research JSC, 2011).

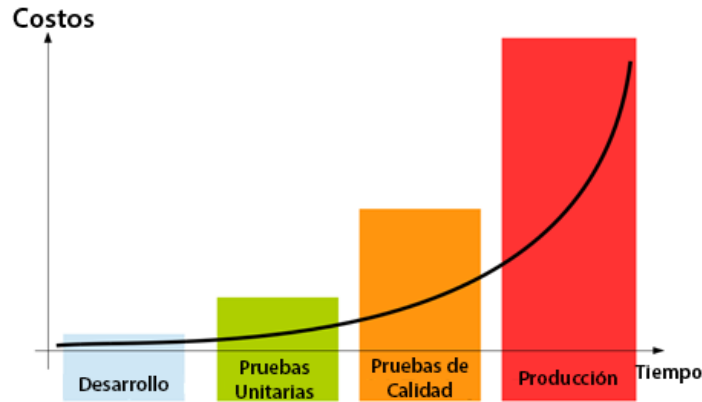


Figura 2.5- Gráfica de costos aproximados durante un proyecto de software. (Nasa Research JSC, 2011)

2.4.1 Eficiencia en las metodologías tradicionales

Entre los principales problemas dentro de las empresas de desarrollo de software, encontramos casos como una ineficiente formación de los equipos de trabajo, falta de disciplina por parte de los desarrolladores, además de un factor importante que en muchos casos termina costando a las empresas una gran cantidad de recursos: el mantenimiento de software (UMBC Research, 2008).

Analizando más a fondo las prácticas que conllevan a una inflación en el mantenimiento de software, encontramos que en muchos casos las empresas llegan a destinar hasta un 60% del presupuesto en cuestiones de mantenimiento y correcciones (Nasa Research JSC, 2011). Esta cantidad de recursos destinados a la corrección de errores de diseño y funcionamiento tiene repercusiones en la calidad del producto, la Satisfacción del cliente y por supuesto las ganancias totales para la empresa desarrolladora.

Analizando el caso de más de 200 empresas de desarrollo (grandes, pequeñas y medianas) encontramos que durante el proceso seguido para la creación de un proyecto de software se utilizan metodologías ágiles que proveen de una revisión continua a cada parte del proyecto. Sin embargo, esto no es suficiente para minimizar en gran parte la cantidad de errores relacionados al diseño de las interfaces (Norman D. A., 2010).

Durante los últimos años, la migración por parte de empresas que utilizaban metodologías tradicionales a modelos ágiles, ha ido creciendo debido a los beneficios que las metodologías ágiles proporcionan, entre ellos, está la inclusión de pruebas de Usabilidad y experiencia de usuario en el proceso.

De acuerdo a los resultados del último reporte realizado por el Standish Group para la medición del desempeño final de distintos proyectos de software por distintas empresas de software alrededor del mundo, el modelo en Cascada ha venido presentando un desempeño

bajo en comparación con el modelo ágil. Como se ve en la Figura 6 únicamente el 14% de los proyectos llevados con esta metodología tuvieron resultados satisfactorios sin dificultad en el proceso, esto significa que se entregó a tiempo y cumpliendo los requerimientos iniciales. En comparación con los modelos ágiles el 42% de los proyectos mostraron resultados satisfactorios (Standish Group International, 2013).

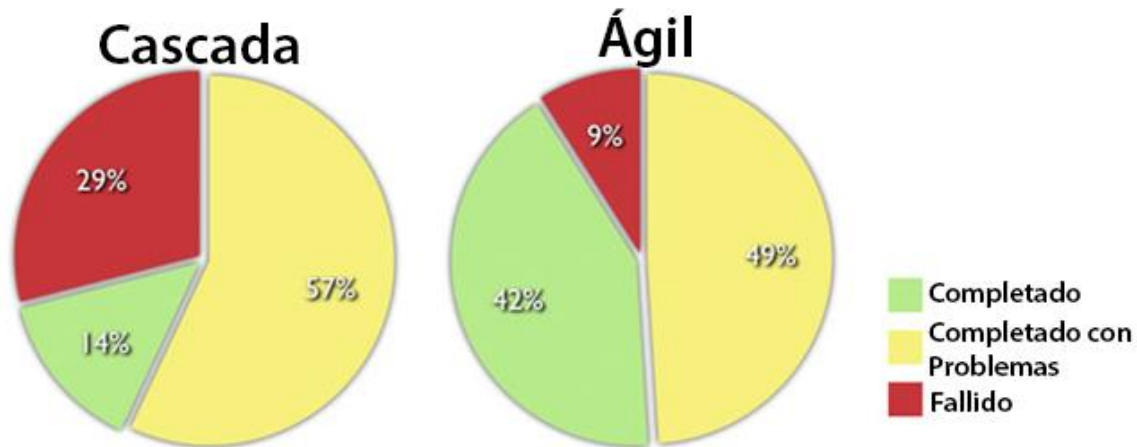


Figura 2.6- Resultado de un estudio con más de 500 empresas a nivel mundial sobre metodologías utilizadas. (Standish Group International, 2013).

2.5 Conceptos y pruebas de Usabilidad

Los diferentes errores encontrados durante la fase de pruebas impactan negativamente en la calidad de un producto de software. Este tipo de errores pueden ser analizados a través de un tipo de pruebas denominadas pruebas de Usabilidad.

Las pruebas de Usabilidad permiten conocer las mejoras para que el usuario final se sienta cómodo con el producto, estas pruebas se realizan en etapas finales de producción pero dentro de las buenas prácticas es recomendable que se realicen a la par del proceso de desarrollo, de esta manera se garantiza que el producto cumple con ciertos estándares en sus interfaces (Nielsen, 1994).

Estudio contextual: Este tipo de estudio está pensado para la obtención directa de información sobre el usuario utilizando las prácticas de observación directa, Estudio de la población y análisis psicológico del usuario final.

La observación directa del usuario final permite determinar el uso práctica que tendrá la aplicación una vez finalizada. De acuerdo a el análisis de prácticas de este método es posible obtener buenos resultados en un periodo de 7 a 14 días (Hartson,2012) (Nodder, 2013).

Análisis de tareas: Esta herramienta nos permite lista el procedimiento actual que tiene el usuario para llevar a cabo su trabajo. Para poder realizar un análisis de tareas es necesario crear definir el flujo de trabajo que deberá seguir el usuario dentro de la aplicación, la cantidad de actividades y opciones que debe realizar y una descripción explícita de la tarea. De acuerdo a la literatura sobre la práctica del análisis de tareas, es posible obtener resultados relevantes en al menos 3 días (Hartson, 2012) (Nodder, 2013).

Entrevistas con usuarios: Este tipo de tarea permite obtener información directa de cuestionarios aplicados al usuario final. A diferencia de la recolección de información a través de los *stakeholder*, las entrevistas con usuarios finales muchas veces revelan detalles sobre la claridad del concepto de la aplicación y los requisitos mínimos que debe tener el software. Recomendable 4 días (Hartson, 2012) (Nodder, 2013).

Diseño de personas: Esta técnica denominada Personas permite la construcción de arquetipos que representan a usuarios del producto y sus necesidades. El objetivo de los mismos es poner un nombre y dar forma humana a los segmentos meta.

El poder dar nombre y cara a un usuario principal, permite aclarar los objetivos y necesidades que este tiene, al final, estas necesidades reales deben estar cubiertas por el producto que se está desarrollando. De acuerdo al análisis de prácticas de Usabilidad es posible obtener resultados a partir de los 2 días (Nielsen, 2013) (Nodder, 2013).

Prototipo de baja fidelidad: Esta modalidad de prototipo permite la creación de una primera versión del producto donde se expongan las principales características que tendrá una vez finalizado.

El objetivo de crear un prototipo de baja fidelidad es el poder probar la idea con usuarios reales justo al principio del desarrollo por lo que el prototipo no tendrá tantas opciones interactivas. En este tipo de prototipado es posible evaluar el color, tamaño, utilidad y opciones principales para poder obtener una rápida retroalimentación por parte del consumidor. De acuerdo al análisis de prácticas de Usabilidad, es posible obtener resultados en 4 días (Hartson, 2012) (Nodder, 2013).

Wireframes: Este tipo de técnica permite crear una guía visual de lo que serán las opciones dentro del sistema final principalmente la interfaz de usuario. Con los wireframes es posible determinar los tipos de información que será mostrada, la cantidad de las funciones disponibles, las prioridades relativas de la información y las funciones, las reglas para mostrar ciertos tipos de información y el efecto de los distintos escenarios en la pantalla. De acuerdo al análisis de prácticas de Usabilidad es posible obtener resultados en aproximadamente días (Amowitz, 2007) (Nodder, 2013).

Paseo cognitivo: A diferencia de la evaluación heurística en la que el experto se basa en una serie de principios y buenas prácticas (las heurísticas), el recorrido cognitivo permite al experto detectar los problemas a medida que hace una simulación de la ejecución de las tareas como si fuera un usuario. Esto hace que los problemas detectados sean habitualmente problemas realmente relevantes. De acuerdo al análisis de prácticas de Usabilidad es posible obtener resultados en 3 días (Faulkner,1998) (Nodder, 2013).

Pruebas con usuarios: Este tipo de pruebas hace uso de una selección de usuarios finales para que prueban el producto en desarrollo. Las pruebas con usuarios pueden llamar también pruebas de análisis en la que los usuarios representativos trabajan en tarea típicas

Utilizando el sistema o prototipo, aquí los evaluadores se encargan de visualizar como la herramienta se encarga de realizar las tareas a los usuarios.

Otro tipo de enfoque que se le pueden dar a las pruebas con usuarios es de tipo indagación. La identificación de requerimientos, tanto los del usuario como los de nuestro producto, son indispensables en una etapa temprana de un proceso de desarrollo que culminará en la Satisfacción de una necesidad del usuario, quien con Eficiencia y efectividad habrá de realizar las funciones que ese producto le ofrece. De acuerdo al análisis de prácticas de Usabilidad en la industria es posible obtener resultados a partir de 5 días (Nielsen, 1994) (Nodder, 2013).

Pruebas heurísticas globales: Las pruebas heurísticas en Usabilidad refieren al análisis experto que se realiza siguiendo los lineamientos profesionales comprobados en el área. En este caso los más relevantes son las “reglas heurísticas” establecidas por Jacob Nielsen mostradas en la Tabla 3, las cuales recopilan los principales problemas dentro del diseño de productos de software principalmente sistemas web. (Nielsen, 2010).

Tabla 3: Validación por Heurísticas de Nielsen. (Nielsen,2010)

Número	Descripción de Prueba Heurística
1	Visibilidad del estado del sistema
2	Relación entre el sistema y el mundo real
3	Control y libertad del usuario
4	Consistencia y estándares
5	Prevención de errores

6	Reconocimiento antes que recuerdo
7	Flexibilidad y Eficiencia de uso
8	Estética y diseño minimalista
9	Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de errores
10	Ayuda y documentación

Analizando la documentación sobre prácticas de Usabilidad es posible obtener resultados en 3 días (Nielsen, 2010)

Una prueba básica de Usabilidad que contenga pruebas con usuarios, análisis de tareas, especificación de usuarios y resultados estadísticos iniciales en base a cuestionarios TAM (Technology Acceptance Model) (Kowitlawakul,2005) proporciona información suficiente para mejorar el diseño y aminorar los errores en etapas posteriores de evaluación, el problema se presenta cuando no se hace una correcta adaptación de este procedimiento a la metodología utilizada por lo que los tiempos disponibles para la evaluación es demasiado corto (Krug, 2006).

2.6 Industria del software en México

Respecto al panorama de las empresas de desarrollo en México, de acuerdo con estimaciones realizadas por ESANE consultores en colaboración con INEGI sobre el número total de empleados y empresas de la Industria del Software en México, el número aproximado de empresas de la industria mexicana del software podría ser del orden de 9,540 empresas (ESANE, 2004).

De la población total de empresas en México dedicada el desarrollo de software, únicamente alrededor de 1,500 están registradas en padrones como PROSOFT, afiliadas al AMITI y a la AMCIS.

Este tipo de estadística únicamente describe de manera muy general la situación de la industria en el país, cuestiones como el registro dentro de padrones de calidad dificulta el poder analizar más afondo la manera en que los proyectos son desarrollados y si en verdad cumplen con las expectativas de los clientes (ESANE, 2004).

Dentro de los escasos estudios encontrados respecto al estado de las empresas de software en México encontramos que de acuerdo a la UNAM (Jiménez, 2012) de un estudio con 85

empresas como muestra significativa el 65% por ciento de las empresas tienen en 1 y 5 años de antigüedad lo cual se considera joven si tomamos en cuenta la antigüedad de empresas consideradas como grandes (SE, 2004).

Los proyectos que más realizan estas empresas son de tipo multimedia y sistemas web muy por encima de sistemas distribuidos, sistemas en tiempo real y sistemas de control. Al ser empresas muy pequeñas muchas veces cuentan con proyectos cortos sin mucho análisis de riesgo por lo que la mayoría cuenta con proyectos que se completan en promedio en 3 meses.

Una estadística un tanto controversial es que de todas las empresas analizadas el 29% menciona que no en todos los proyectos se cuenta con un plan de desarrollo a utilizar como guía y un 5% definitivamente no utilizan algún plan de proyecto durante el ciclo de vida del producto (Jiménez, 2012).

Respecto al tamaño del equipo de trabajo el promedio de integrantes es de 15 personas. Esto puede estar relacionado al hecho de que las empresas grandes propiamente mexicana son pequeñas y las consideradas grandes con más de 200 empleados son prácticamente escasas o son grupos de desarrollo provenientes de empresas americanas.

2.7 Eficiencia, Eficacia y Satisfacción

Las pruebas de Usabilidad permiten evaluar “la capacidad de un software de ser comprendido, aprendido, usado y ser atractivo para el usuario, en condiciones específicas de uso” (ISO/ICE 9126) o “el grado de que un producto puede ser usado por usuarios específicos para lograr las metas con efectividad, Eficiencia y Satisfacción en un contexto de uso determinado” (ISO 9241-11).

Dentro del análisis de Usabilidad y calidad de software es necesario tener en cuenta al menos 3 características relacionadas a los niveles de Usabilidad:

Eficiencia: Nivel obtenido de la mediación total del tiempo en que tarda un usuario en realizar las tareas que se propone con el sistema.

Eficacia: Se refiere al número de tareas realizadas correctamente por el usuario por un periodo de tiempo.

Satisfacción: Este indicador nos permite conocer que tan satisfecho está el usuario con los resultados. ¿Qué tan agradable y sencillo le ha parecido al usuario la realización de las tareas?

2.8 Modelo de aceptación tecnológica por parte del usuario (TAM)

El modelo TAM (Technology Acceptance Model) permite medir el grado de aceptación por parte del usuario hacia un producto tecnológico. Este modelo permite obtener 2 variables adicionales que nos permiten comprobar que realmente se cumple con parámetros de Usabilidad y que el usuario encuentra útil el producto de software. Las variables utilizadas en esta parte son las siguientes.

- **PU (Perceived usefulness, Utilidad Percibida):** definida como el grado en que una persona cree que el uso de un determinado sistema mejora su rendimiento en el trabajo.
- **FUP (Perceived ease-of-use, Percepción de facilidad de uso).** Definida como el grado en que una persona cree que utilizando un sistema en particular, podrá liberarse del esfuerzo que le conlleva realizar un trabajo.

Como se puede ver en la Figura 2.7 estas dos variables PU y FUP provienen de variables externas las cuales únicamente se clasifican como aquellos factores referentes al grado de conocimiento del programador en cuanto a diseño y al grado de manejo de la tecnología por parte del usuario.

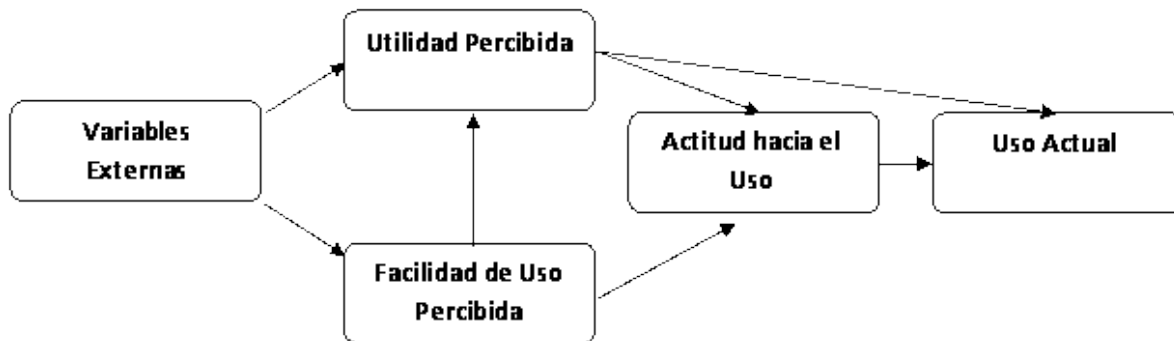


Figura 2.7: Obtención de información utilizando el modelo TAM. (Kowitlawakul, 2005)

2.9 Adaptaciones a los modelos actuales

Actualmente se han intentado varias adaptaciones a diferentes modelos de desarrollo principalmente para modelos ágiles. Tomando en cuenta la cantidad de problemas existentes explicados anteriormente, es posible incluir pruebas de Usabilidad si conocemos a detalle la progresión que el desarrollo llevará durante su tiempo total.

Dentro del mercado de software para pruebas podemos encontrar distintas alternativas funcionales para los diferentes tipos de procedimientos existentes. Estos productos de diferentes características varían según el instrumento que se desea utilizar, la etapa en el desarrollo en la que se encuentra el producto y el tipo de parámetros que se buscan medir.

Para poder medir parámetros como la cantidad de métricas heurísticas, nivel de retención del usuario y funcionalidad de la experiencia de usuario es necesario contar con herramientas que nos proporcionen una manera de obtener información por parte de los usuarios además de una herramienta estadística apropiada.

El modelo LEAN UX (Gothelf, 2013) busca crear una modificación para los modelos ágiles que quizás puedan aplicarse de alguna manera a los métodos tradicionales. Este modelo parte de una modificación al UCD como vemos en la Figura 2.8. Dicho modelo busca reducir el tiempo entre la abstracción del problema proporcionar una respuesta de diseño para establecer algunos requerimientos como parte de un proceso exploratorio.

Incluir las fases de pensar, hacer y revisar a través de pequeños ciclos iterativos permite tener diferentes prototipos que arrojan resultados durante todo el proceso de desarrollo.

El modelo LEAN UX puede servir de referencia para realizar una modificación a los modelos tradicionales la cual incluya tareas básicas de Usabilidad.

La razón por la cual la mayoría de las nuevas modificaciones únicamente se aplican a las metodologías ágiles es por la facilidad y flexibilidad que éstas representan, por lo que se tiene más margen de error a la hora de una implementación (Gothelf, 2013).

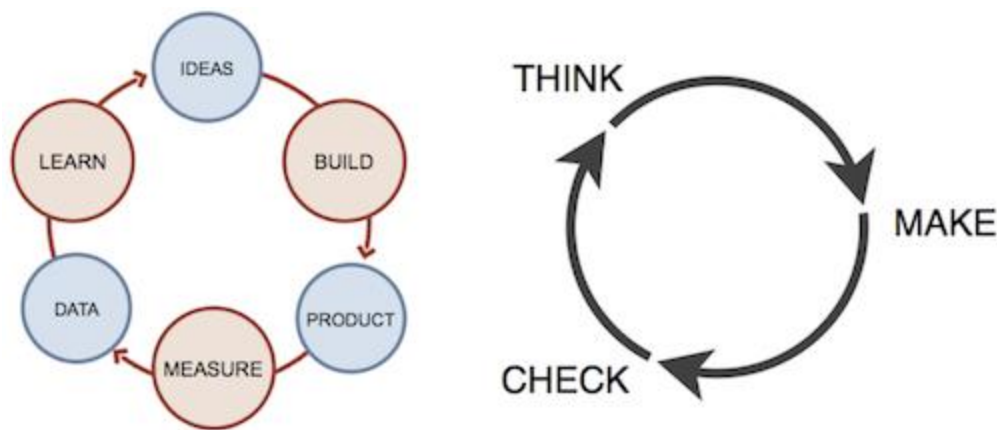


Figura 2.8- Comparación entre el Modelo Lean UX y Diseño Centrado en el Usuario. (Gothelf, 2013).

2.10 Conclusiones del Marco Teórico

La evolución de las metodologías para el desarrollo de software está basada en las necesidades del mercado, la demanda de productos novedosos y la manera en que trabajan los equipos de desarrollo.

La creación de modificaciones a las metodologías tradicionales de desarrollo deriva comúnmente en la creación de un modelo totalmente nuevo.

En este caso, se busca la adaptación de dos de los modelos tradicionales Cascada y Espiral. Aunque dichos modelos tienen más de 20 años en el mercado, de acuerdo al estudio realizado aún son utilizadas por muchos equipos de desarrollo a nivel nacional e internacional.

Lograr una adaptación adecuada de conceptos de Usabilidad puede reducir costos en mantenimiento correctivo, además de incrementar los niveles de Eficiencia, Eficacia y Satisfacción por parte del usuario, todo esto sin la necesidad de involucrar más de un modelo en el desarrollo y sin la necesidad de cambiar de metodología a mitad del proyecto.

3 Desarrollo de la investigación

En este capítulo se explica a detalle cada una de las etapas que se llevaron a cabo con el objetivo de comprobar la validez de nuestra hipótesis inicial: “Es posible la inclusión de conceptos y estudios de Usabilidad en las etapas con mayores problemas dentro de los modelos de desarrollo de software tradicionales Cascada y Espiral, todo esto con el apoyo de una herramienta que guíe a los desarrolladores en las adaptaciones propuestas”.

Para poder dar respuesta a nuestra hipótesis de investigación, fue necesario separar nuestro trabajo en 4 partes principales. En la primera parte (Sección 3.2.1) se realizó un análisis inicial con empresas en México con el objetivo de conocer la situación actual de la industria de software con respecto a prácticas de Usabilidad.

En la segunda parte del proceso de investigación (Sección 3.2.2), se muestra un análisis de la información obtenida en la primera etapa, utilizado para complementar las características básicas necesarias que debe tener nuestra adaptación a los modelos.

En la siguiente etapa se explica el diseño del modelo teórico, el cual fue construido utilizando la información obtenida hasta ese momento.

Durante la última etapa del procedimiento de investigación se explican las características y la construcción de una herramienta prototipo que permite guiar a los desarrolladores durante el uso de las metodologías adaptadas.

3.1 Prácticas de Usabilidad dentro de las empresas en México

Para la primera parte de la investigación, fue necesario aplicar una encuesta a empresas en México con el objetivo de conocer la situación actual de las prácticas de Usabilidad en el desarrollo de software.

La muestra significativa para esta parte se estableció tomando en cuenta el registro de 1,150 empresas en México por parte de la Secretaria de Economía (ESANE, 2004), (SE, 2004). De esta población total, se tomaron únicamente aquellas empresas que se dedican principalmente al desarrollo de software y con las que se tiene suficiente información de contacto. Todo esto resulta en una población de 575 Empresas de las cuales únicamente se utiliza el 10% para obtener resultados relevantes.

$$\frac{575}{10} = 57.5$$

La localización geográfica de estas 60 empresas consideradas en la primera etapa de este trabajo se encuentra en los estados de Nuevo León, Puebla, Colima, Oaxaca y el Distrito Federal.

Para la etapa final de la investigación referente a pruebas de Usabilidad con el prototipo, se utilizó una muestra de 6 centros de investigación y empresas de las cuales se seleccionaron 12 equipos de desarrollo. La distribución de esta selección de usuarios para esta etapa de pruebas consta de 2 equipos por cada empresa participante. Cada uno de los equipos está formado por 3 integrantes los cuales cuentan con un conocimiento básico en programación de aplicaciones de escritorio en al menos 2 lenguajes diferentes.

En total de los 12 equipos, se tiene la participación de 36 programadores con un nivel básico de experiencia en el área de desarrollo.

Para el área de evaluación heurística de los resultados de la prueba con los equipos de desarrollo, se cuenta con la participación de 3 profesores investigadores en el área de Interacción Humano-Computadora.

3.2 Análisis inicial de la información y estudio con empresas en México

En el Capítulo 2 se analizaron las técnicas actuales de desarrollo de software con aspectos de Usabilidad, recopilación de información respecto al problema y la especificación de las partes con las que se trabajó para la propuesta. Este análisis se realizó tomando en cuenta principalmente los reportes anuales realizados por *Standish Group* donde se recopilan 300 casos de estudio. (Standish Group International, 2013).

Del total de casos de estudio analizados se tomaron en cuenta aquellos proyectos que utilizan el modelo Cascada y Espiral como principal metodología. Otro factor a tomar en cuenta es el tamaño del proyecto, aquí se separaron en 3 categorías principales: pequeño, mediano y grande.

Los proyectos de tamaño pequeño refieren un desarrollo menor a 6 meses en el que participan máximo 30 personas. Los proyectos de tamaño mediano refieren a proyectos con una duración mayor a 6 meses en el que participan más de 60 personas en su desarrollo. Los proyectos de tamaño grande refieren a proyectos con una duración mayor a 2 años que involucran a más de 120 personas en su desarrollo (Standish Group International, 2013).

Como producto de este análisis se determinó que para cada categoría se tienen 35 casos de referencia relacionados a proyectos pequeños, 50 casos en los que el proyecto se determinó de tamaño medio y 25 casos en los que el proyecto es de tamaño grande.

Después de realizar el análisis de la documentación actual sobre la problemática dentro de los proyectos de desarrollo de software, se realizó un estudio dentro de varias empresas en el ámbito de desarrollo de software utilizando el método Delphi en al menos dos circulaciones.

El estudio inicial con empresas mexicanas considera una muestra significativa de 50 empresas (explicado en la Sección 3.1) localizadas en los estados de México DF, Monterrey, Jalisco, Puebla, Colima y Oaxaca. Del estudio se tomarán principalmente aquellas empresas que utilicen únicamente metodologías tradicionales en sus proyectos. El uso del método Delphi para esta parte nos permite disminuir la dispersión y precisar la opinión media consensuada.

Las empresas y grupos participantes en este estudio inicial son: Intel México, IBM México, HP Labs, TATA, iGate, Grupo Salinas, CompuSoluciones, MakerLab, Microsoft Openness, Crowd Interactive, Desarrollo Tecnológico del Estado de Colima, ITESO Lab, Usaria, SCRUM Alliance y Softtek.

El resultado de las encuestas comienza con una descripción del puesto de trabajo que ocupan los participantes. Como se puede ver en la Figura 3.1 los participantes ocupan puestos de implementación y desarrollo principalmente esto se debe a que actualmente es donde mayor contratación se tiene.

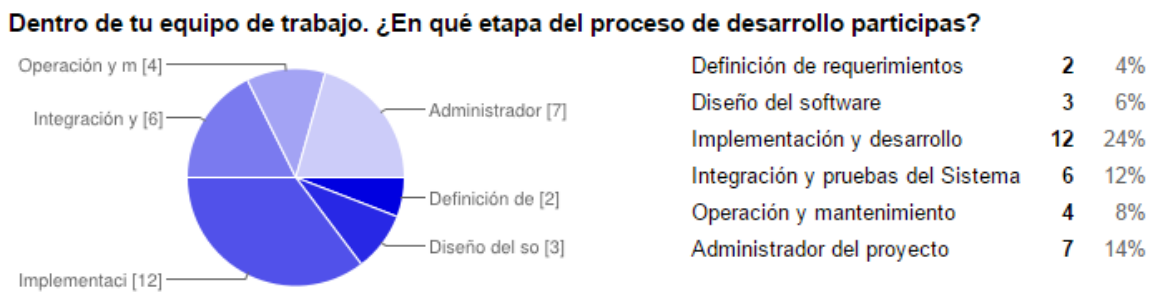


Figura 3.1: Resultado de las diferentes etapas en las que se participa.

El siguiente resultado es uno de los más importantes para poder establecer prioridades en el diseño de la adaptación. En la Figura 3.2 se listan las diferentes etapas del proceso de desarrollo en las principales metodologías. El resultado determina que en la etapa de requerimientos, diseño e integración es en donde se presentan mayores dificultades. Esto se debe a la falta de métodos y cambios que surgen durante el desarrollo.

La parte de implementación no presenta mayor problema de acuerdo a los administradores de proyecto debido a que la mayoría de los errores en la codificación se detectan en la siguiente parte de pruebas.

¿En que etapa del proceso se presentan mayores dificultades?

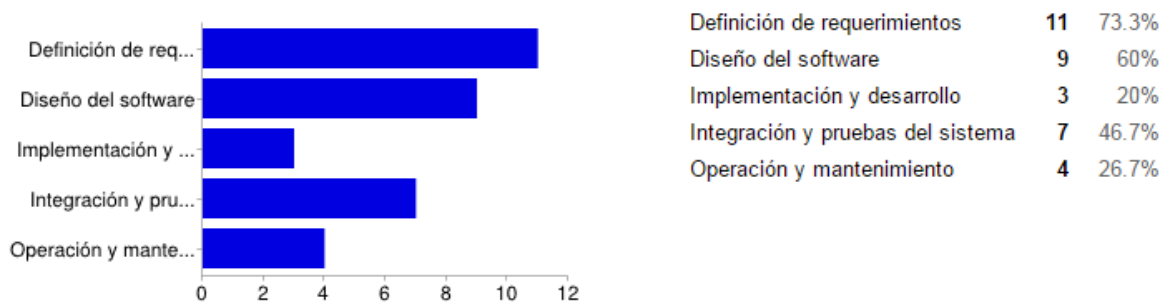


Figura 3.2: Resultados de las etapas con mayores dificultades.

Respecto a las herramientas utilizadas para la capacitación y entrenamiento, se encontró que la mayoría de los desarrolladores/diseñadores utilizan documentación en línea debido a su facilidad de acceso y contenido. El resultado más destacado de esta parte es el hecho de que la documentación impresa ya no es un recurso tan importante como lo fue en años pasados. De acuerdo a los administradores de proyecto se debe a que hoy es necesario que puedan acceder a la información desde cualquier lugar además de que ciertos componentes requieren de mucho contenido por lo que es posible que el desarrollador/diseñador invierta mucho tiempo en su consulta.

Como parte de tu capacitación y entrenamiento. ¿Qué tipo de herramientas utilizas?

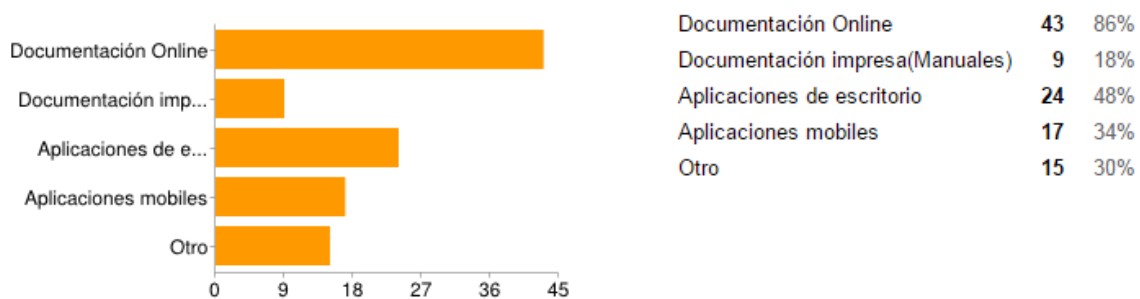


Figura 3.3: Principales medios utilizados para capacitación y entrenamiento.

Respecto a las metodologías de desarrollo utilizadas en las empresas mexicanas tenemos que la metodología ágil SCRUM es la más utilizada actualmente como se puede ver en la Figura 3.4, esto se debe a la popularidad que han tenido las metodologías ágiles en los últimos años. En segundo plano tenemos el Modelo en Cascada y el modelo en Espiral los cuales de acuerdo a los jefes de desarrollo son utilizados como modelos de apoyo para proyectos grandes que requieren más de 8 meses de desarrollo constante.

De las siguientes metodologías de desarrollo. ¿Con cuáles has trabajado?

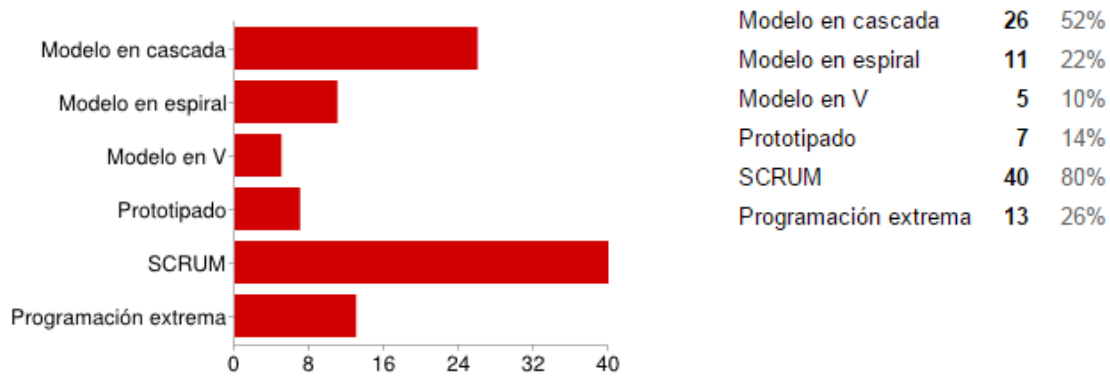
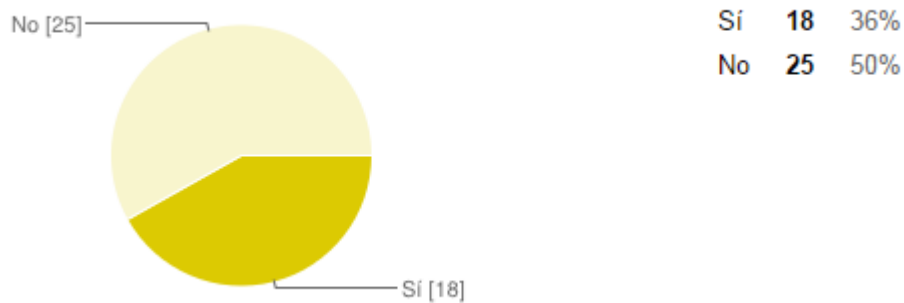


Figura 3.4: Principales metodologías utilizadas.

Para determinar si los participantes tienen un acercamiento directo con el usuario final nos encontramos que en su mayoría no tienen un acercamiento esperado durante el desarrollo. En algunos casos el participante no llega a conocer las características completas del usuario final y se apegan a los requisitos del *stakeholder* como se puede ver en la Figura 3.5.

Como parte de las razones por la cual no se tiene un acercamiento directo, encontramos que en la mayoría de los casos alguien más durante el proceso de desarrollo lo realiza. En la mayoría de los casos el personal encargado de los requerimientos se apegan a los requisitos pensados por el *stakeholder* por lo que durante todo el proceso se siguen este tipo de lineamientos los cuales no reflejan las necesidades del usuario final.

Como parte de tu trabajo ¿Tienes acercamiento directo con el usuario final?



Razón por al cual no tienes acercamiento directo con el usuario

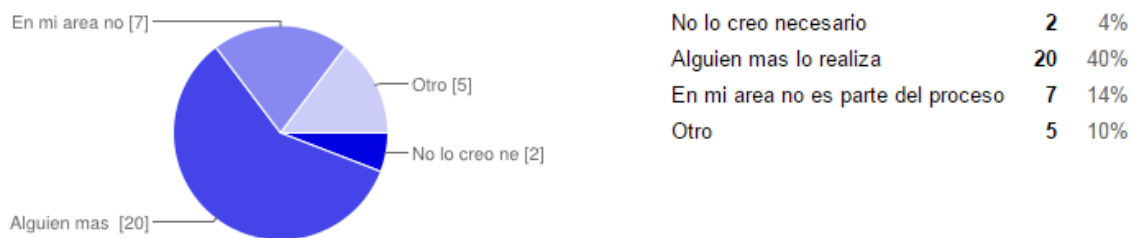


Figura 3.5: Acercamiento directo con el usuario.

Para determinar si los participantes tienen conocimiento de los diferentes conceptos que forman parte de la experiencia de usuario, se listaron las principales tareas que son compatibles con los modelos de desarrollo ágil y tradicional.

Como podemos observar en la Figura 3.6 las principales tareas conocidas son las entrevistas, pruebas de Usabilidad y evaluación TAM. Aquí es importante tomar en cuenta que el hecho de que hayan escuchado hablar de ellas no quiere decir que sepan cómo aplicarlas o que las hayan realizado en algún proyecto.

De las siguientes tareas de usabilidad. ¿Cuáles conoces?

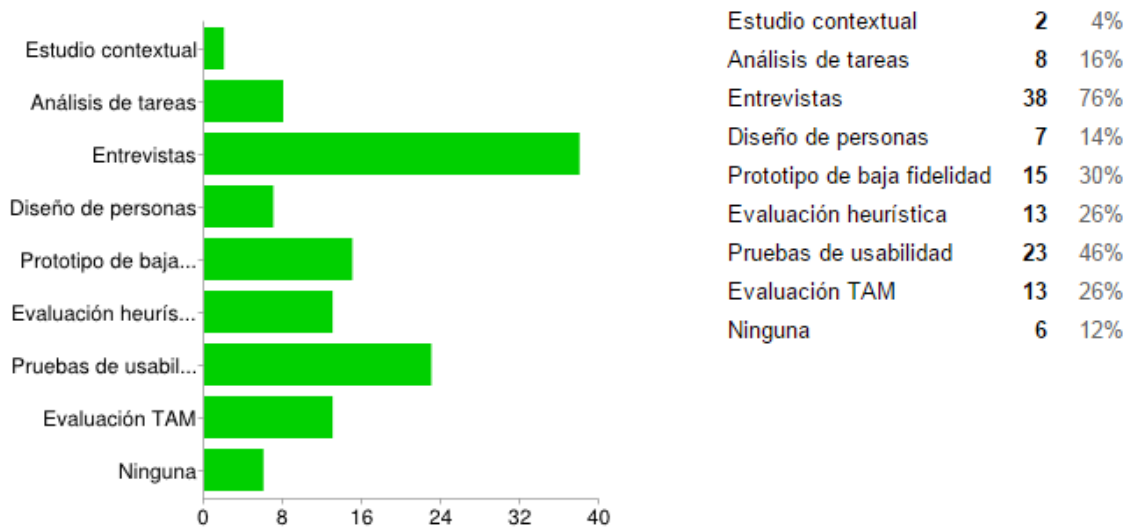


Figura 3.6: Principales tareas de Usabilidad conocidas.

El último dato interesante resultado del estudio inicial, revela que los participantes encuentran de gran importancia contar con información completa por parte del usuario final. Sin embargo, esta sólo les es útil en ciertas partes del proceso. Aquí existe una relación con aquellas etapas con las que se tiene mayor problema que son requerimientos, diseño y pruebas. Es necesario que el diseño de la adaptación este pensado para que la información fluya en todo momento entre los distintos departamentos y esté disponible para cuando sea requerida.

¿Crees que es importante utilizar información del usuario en todo el proceso de desarrollo?

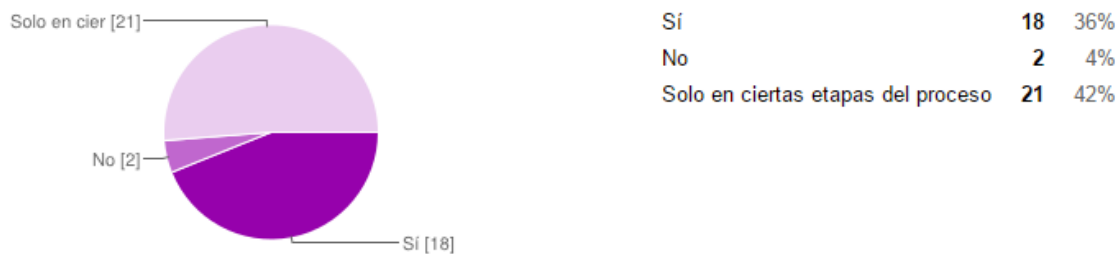


Figura 3.7: Importancia del acercamiento con el usuario.

3.3 Estructura del problema

Teniendo en cuenta lo explicado en la Sección 2.8 y tomando como referencia la manera en que los sistemas de pruebas han ido evolucionando (Meerts, 2013), la falta de pruebas de calidad para la obtención de niveles de Usabilidad principalmente Eficiencia, Eficacia y Satisfacción referente al producto final durante el proceso de desarrollo, es parte importante del problema relacionado al número de proyectos terminados con deficiencias, mayormente en la parte de aceptación final del producto por parte del usuario.

Analizando las razones por las que se crearon otros modelos como XP (*Extreme Programming*) y *Lean*, es necesario tener en cuenta al usuario desde el diseño inicial. (Coplien, J. 2010), (Succi, G., & Marchesi, M. 2001), (Pries, K., & Quigley, J. 2011).

3.4 Diseño de la adaptación

La solución que se propone como acercamiento al problema permite incluir las tareas de Usabilidad explicadas en la Sección 2.7 en las etapas problemáticas de los modelos tradicionales de desarrollo de software Cascada y Espiral a través de una herramienta que guíe a los desarrolladores durante su ejecución.

La adaptación permitirá cumplir con los requerimientos básicos que incluyen el poder adaptar los conceptos básicos de Usabilidad en todas las etapas del modelo Cascada y Espiral.

Como se puede ver en la Figura 3.8 el modelo original denominado Cascada cuenta con una serie de etapas secuenciales. Estas etapas se respetan y no se altera el orden en que se desarrollan además de contemplar las tareas internas que requiere cada una.

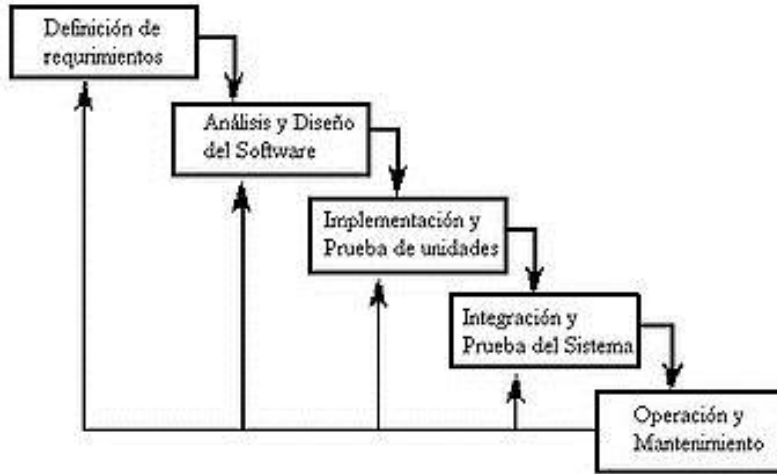


Figura 3.8: Estructura tradicional del modelo Cascada. (Pressman, 1995)

Otro aspecto que se busca como producto de esta investigación es el permitir que a través de esta adaptación se puedan modificar las tareas realizadas en las diferentes etapas del desarrollo de software, haciendo la adaptación adecuada en los modelos tradicionales utilizados. Como se puede ver en la Figura 3.9 el modelo Cascada cuenta con una serie de etapas básicas para poder funcionar.

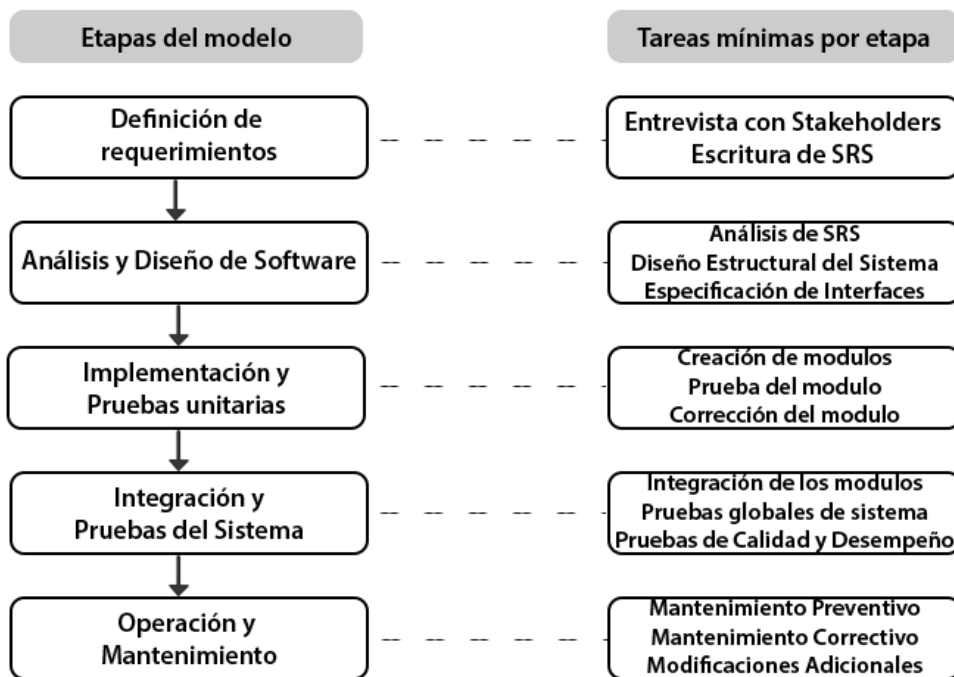


Figura 3.9: Etapas de modelo en Cascada y sus tareas mínimas por etapa.

Dicha estructura en conjunto con las tareas mínimas requeridas por etapa son las utilizadas por las empresas desarrolladoras de software a la hora de elegir el modelo en Cascada como base de un proyecto.

Al utilizar únicamente dichas tareas de acuerdo a la documentación y problemática descrita en la Sección 2.8, se obtienen niveles por debajo del 50% en cuanto a Eficiencia, Eficacia y Satisfacción en la evaluación del producto final además de tener un nivel por debajo del 40% en cuanto a PU (Utilidad Percibida) y FPU (Percepción de Facilidad de Uso) los cuales definen principalmente si un producto cuenta un nivel de Usabilidad aceptable por encima del 60 %.

Al incluir tareas básicas de Usabilidad como se muestra en la Figura 3.10 es posible que se obtengan mejores niveles en cuanto a estas variables siempre y cuando se respete la estructura completa del modelo y no se sustituyan por completo las tareas originales.

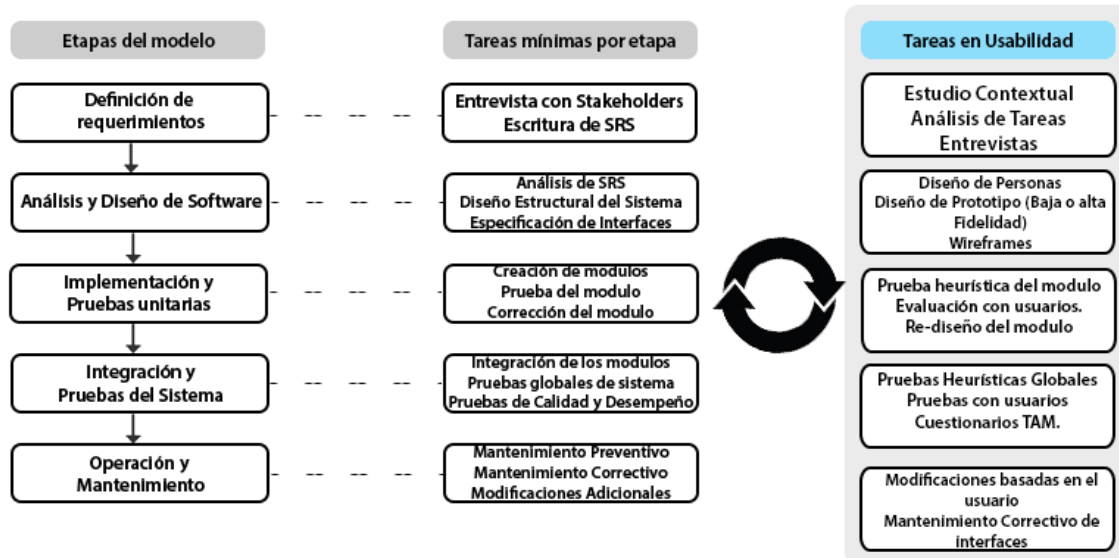


Figura 3.10: Tareas básicas de Usabilidad en conjunto con las tareas del modelo Cascada.

3.4.1 Adaptación de las tareas de Usabilidad dentro del modelo en Cascada

Dentro de cada etapa es posible intercambiar las tareas originales propuestas en el modelo con el objetivo de incluir información por parte del usuario.

A través del análisis de los documentos recopilatorios de casos de estudio como el publicado por Nielsen Project y Standish Group es posible determinar cuáles son las tareas más importantes que se deben respetar dentro de cada etapa y cuales tienen cierta flexibilidad.

Para ilustrar el concepto de tareas estrictas y tareas flexibles, se identifican las tareas estrictas con un punto azul dentro de los diagramas contenidos en esta sección. Las tareas estrictas no deben sufrir algún cambio en cuanto a su producto final ya que al sufrir una modificación la etapa completa pierde su propósito. (Pressman, 1995).

Las tareas flexibles que están identificadas con un punto verde dentro de los diagramas pueden ser intercambiadas siempre y cuando no interfieran completamente con las tareas estrictas dentro de la etapa.

Como resultados del estudio inicial, es necesario tomar en cuenta que el tiempo dentro de cada tarea depende de la ruta de trabajo que se haya definido para el proyecto. Este factor es tomado en cuenta a la hora de realizar la sustitución ya que no es recomendable exceder el número de días que se planteó inicialmente para la etapa completa con el fin de cumplir con todas las tareas de Usabilidad.

Para la primera parte del proceso de desarrollo en Cascada se tomaron 3 tareas básicas de Usabilidad como primer acercamiento.

Las tareas propuestas incluyen: Estudio contextual el cual se recomienda tenga una duración de 7 a 14 días como mínimo (Hartson, 2012), Análisis de tareas en el cual se recomienda que dure mínimo 3 días (Hartson, 2012) y Entrevistas con usuarios con una duración recomendable de al menos 4 días (Hartson, 2012). En la Figura 3.11 es posible observar el orden en que se recomienda realizar la sustitución.

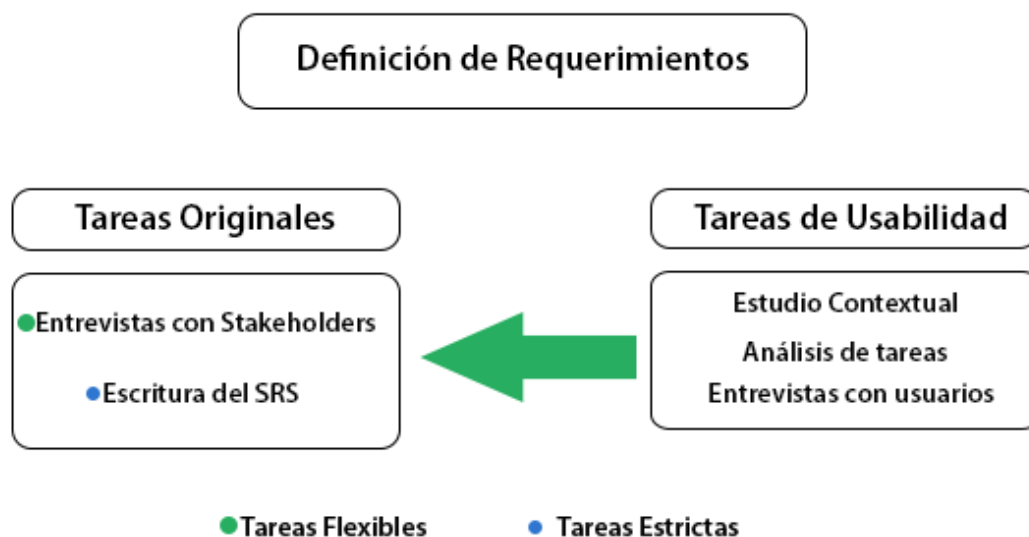


Figura 3.11: Adaptación de Tareas de Usabilidad en la etapa de requerimientos.

La etapa de análisis y diseño se proponen 3 tareas básicas de Usabilidad las cuales comprenden el uso de la técnica Diseño de Personas en la que se recomienda al menos se tengan 2 días para obtener resultados (Nielsen, 2013). El diseño de Prototipo de baja fidelidad el cual requiere de al menos 4 días para poder incluir la mayoría de los elementos de interacción (Hartson, 2012).

La última tarea de esta etapa es el diseño de Wireframes que puede ser de mucha ayuda a la hora de proporcionar una idea clara sobre los elementos básicos del producto. Para esta tarea se recomienda tomar mínimo 4 días (Amowitz, 2007). En la Figura 3.12 es posible verificar el orden de inclusión dentro del esquema original.

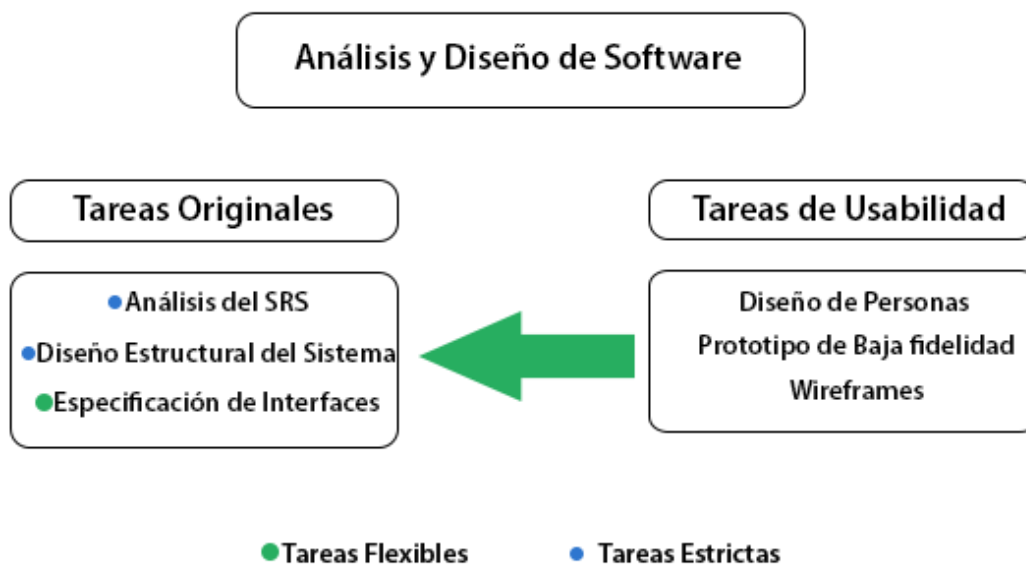


Figura 3.12: Adaptación de tareas de Usabilidad en la etapa de análisis y diseño.

En la etapa de implementación y pruebas unitarias se especifican tareas de exploración como lo es el Paseo cognitivo el cual se recomienda dure al menos 3 días (Faulkner, 1998). La inclusión de Pruebas con Usuarios puede definirse desde el comienzo ya que al llegar a esta etapa se buscan usuarios potenciales que cumplan el perfil al cual va dirigido el producto, aquí varía el tiempo necesario para obtener resultados por lo cual se recomienda que el proceso dure mínimo 5 días para grupos entre 5 y 20 usuarios (Nielsen, 1994).

La última tarea básica de Usabilidad dentro de esta etapa es el Re-diseño del módulo el cual debe incluir los cambios importantes reportados para mejorar la experiencia de uso, aquí se recomiendan al menos 14 días (Hartson, 2012).

En la Figura 3.13 es posible ver las reglas de adaptación las cuales sirven de guía al desarrollador. Aquí es posible determinar obtener indicadores de Eficiencia y Eficacia de la aplicación.

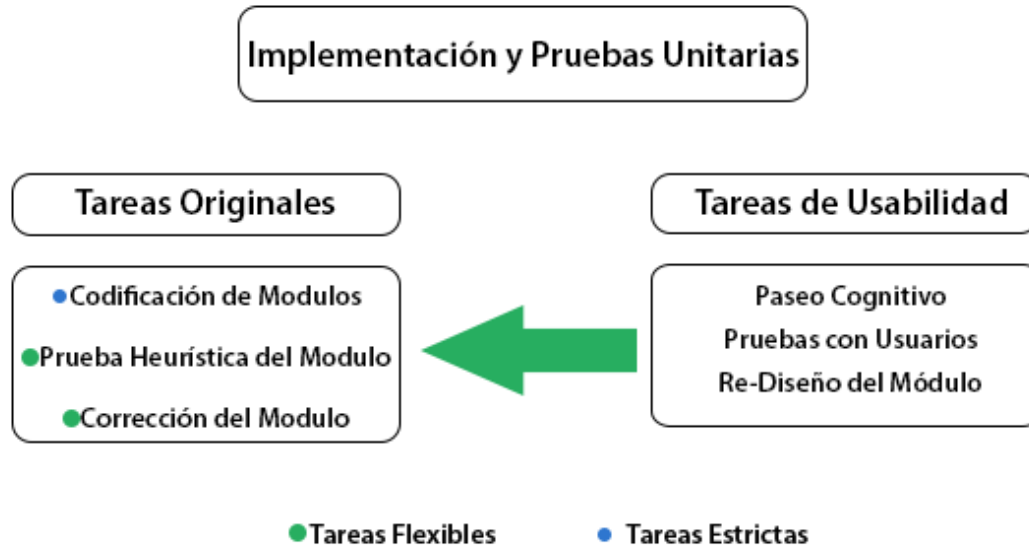


Figura 3.13: Adaptación de tareas de Usabilidad a la etapa de implementación.

Durante la etapa de investigación inicial se determinó que la tercera etapa con mayor problema en el proceso de desarrollo es la integración y pruebas globales del sistema. Para esta etapa se propone la inclusión de Pruebas Heurísticas globales las cuales deben realizarse con el objetivo de mejorar el flujo entre los diferentes elementos de la interfaz.

Algunas tareas secundarias propuestas son la realización de una segunda ronda de pruebas con usuarios y un cuestionario TAM para determinar si el producto cumple con las expectativas del usuario. En la Figura 3.14 podemos ver las reglas de adaptación las cuales no permiten obtener mediciones relacionada a la Satisfacción del usuario.

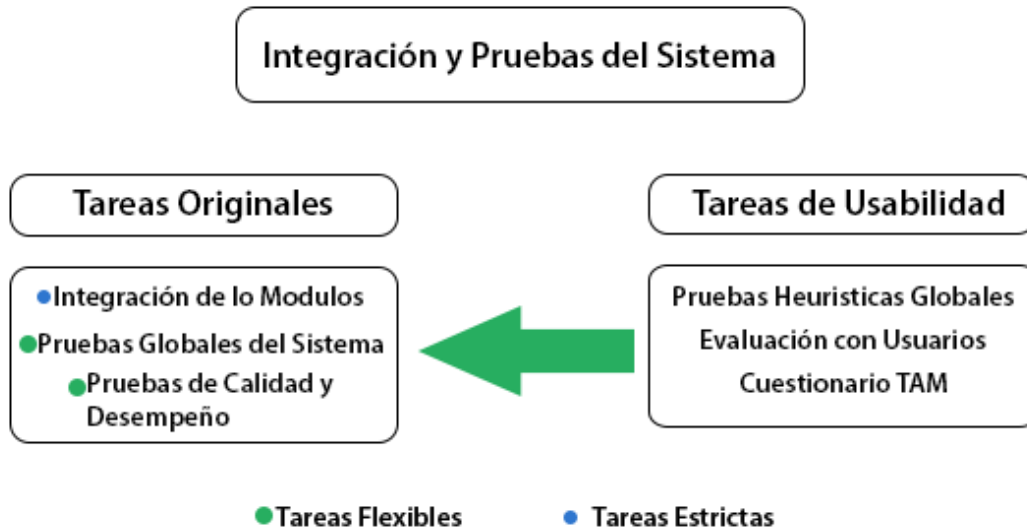


Figura 3.14: Adaptación de tareas de Usabilidad dentro de la etapa de integración.

La última etapa del modelo en Cascada contiene 3 tareas originales de las cuales 2 son de carácter flexible. Aquí se propone incluir las tareas de Usabilidad basadas en modificaciones al producto y mantenimiento correctivo.

En la Figura 3.15 se observa que es posible incluir opciones de mantenimiento para corregir problemas en la interfaz de usuario. Toda la información recopilada hasta el momento permitirá realizar las correcciones de manera más eficiente y facilita el poder rastrear un problema durante el proceso.



Figura 3.15: Adaptación de tareas de Usabilidad a la etapa de operación y mantenimiento.

3.4.2 Adaptación de las tareas de Usabilidad al modelo en Espiral.

Modificaciones basadas en el usuario: Para esta etapa es necesario contar con información producto del análisis del trabajo con el usuario. La cantidad de información que se tiene hasta el momento permitirá determinar la cantidad de cambios en la interfaz o funcionamiento interno del sistema.

Para la inclusión de esta tarea dentro del modelo original es necesario haber realizado pruebas con usuario reales en la etapa anterior de pruebas del sistema.

Mantenimiento correctivo de las interfaces: Durante el tiempo que dure el mantenimiento del sistema, es necesario determinar si los problemas encontrados a lo largo de esta etapa son referentes a la interfaz o a funciones específicas del sistema.

En caso de determinar que el mantenimiento será para corregir funciones relacionadas al rendimiento, se recomienda omitir la información obtenida a partir de los usuarios y proseguir con la ruta establecida originalmente para el proyecto.

A diferencia del modelo en Cascada, el modelo en Espiral nos permite trabajar con cuatro cuadrantes principales referentes a objetivos, análisis de riesgos, desarrollo y planeación. Un factor importante a la hora de definir el modo de adaptación de las tareas de Usabilidad es el hecho de poder tener iteraciones a lo largo del proyecto. (Boehm, 1983)

La flexibilidad de este modelo agrega nuevos elementos con los cuales podemos apoyarnos como un análisis de riesgos antes de iniciar el desarrollo.

Una de los puntos de apoyo que se utilizaron para realizar la adaptación, es la similitud que se tiene con el modelo en Cascada para la parte de desarrollo. Esta estructura permite tomar en cuenta las buenas prácticas que se llevan en el modelo en Cascada para obtener un mejor resultado con el beneficio de poder realizar iteraciones.

La distribución del modelo en Espiral en 4 diferentes cuadrantes (Figura 3.16) permite al administrador de proyecto especificar cuál es el resultado esperado durante cada uno de ellos. En general el modelo tiene varias características similares al modelo en Cascada, esto proporciona una ventaja a los desarrolladores que hayan trabajado anteriormente con este modelo y quieran cambiar a un proceso iterativo fácil de rastrear.

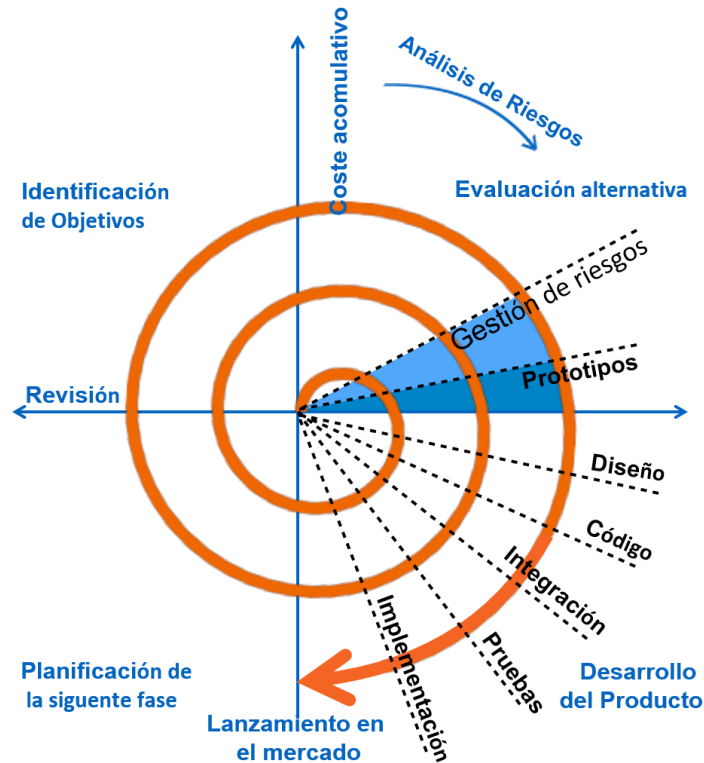


Figura 3.16: Cuadrantes y etapas del modelo en Espiral. (Boehm, 1983)

Antes de comenzar el proceso de adaptación para un proyecto con modelo en Espiral, es necesario tener una idea de que es lo que se pretende obtener ya sea un prototipo inicial, una aplicación completa, un esquema de datos o un módulo perteneciente a un producto de software ya desarrollado.

Hacer un análisis general antes de comenzar a trabajar con este modelo y su adaptación permite al equipo determinar si esta es la metodología adecuada para ellos.

El primer paso dentro de la adaptación al modelo en Espiral es la identificación de objetivos. A diferencia del modelo original, aquí podemos incluir 3 tareas principales de Usabilidad. El estudio contextual y entrevistas con usuarios son sustitutos potenciales para la tarea original de entrevistas con los *stakeholders* que como vemos en la Figura 3.17 es de carácter flexible.

El análisis de tareas en este caso no solo permite al administrador de proyecto tener un documento de requerimiento en forma sino que ayuda a trazar el camino para las siguientes iteraciones. Si se realiza al menos dos de estas tareas, es posible establecer una serie de riesgos iniciales necesarios para la siguiente etapa del modelo.

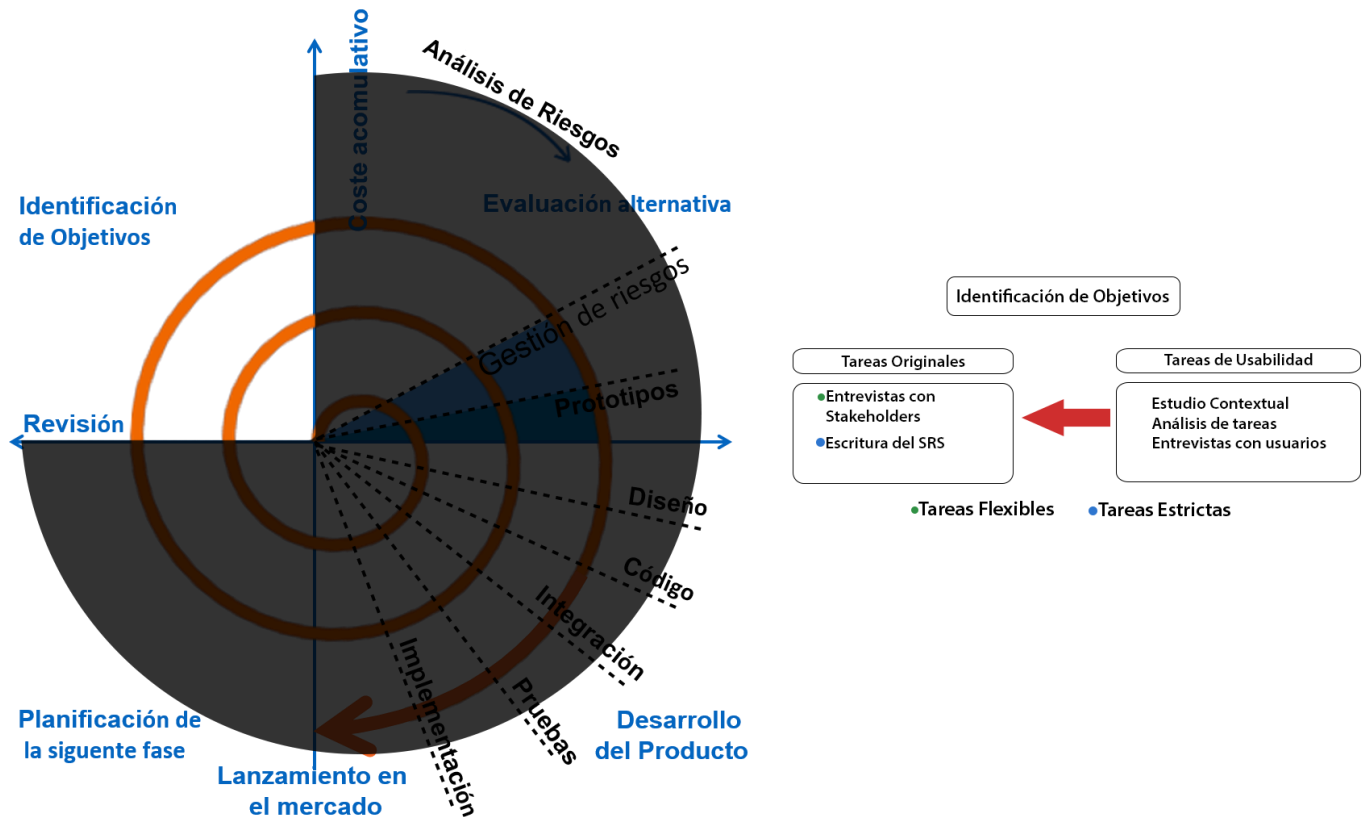


Figura 3.17: Adaptación de tareas de Usabilidad a la etapa de identificación de Objetivos.

Para el segundo cuadrante del modelo en Espiral es necesario contar con un documento de especificaciones para el sistema. La prioridad de los requerimientos y la claridad de las opciones con las que debe contar el sistema permite tener una idea de cuáles son los riesgos que se corren a la hora de omitir o modificar algún requerimiento que resulta ser importante para el usuario.

Como se puede ver en la Figura 3.18, la tarea flexible de retorno de inversión puede especificarse a través de lo que llamamos UR (*User Rejection*). El UR se establece como un indicador producto del estudio inicial con usuarios. Este indicador proporciona que porcentaje de usuarios están de acuerdo en que la propuesta inicial es aceptable y la utilizarían en un futuro.

Un ejemplo del uso de UR sería: Durante el estudio contextual se realiza una encuesta a los usuarios potenciales de la aplicación. En esta encuesta se realizan preguntas de exploración como; ¿El concepto de esta aplicación te parece atractivo?, ¿Pagarías por un servicio con estas características?, ¿Harías uso de una herramienta como esta en tu vida diaria? Las respuestas obtenidas en este estudio te permitirán obtener un porcentaje de cuantos usuarios piensan invertir en tu aplicación.

Una vez obtenido el UR es posible crear un plan de gestión de riesgos adecuado tomando en cuenta que factores deben cumplirse como mínimo para que el usuario realice una inversión.

En esta adaptación al modelo se toma de mayor prioridad que el sistema cuente con las opciones principales solicitadas por el usuario en comparación con las opciones que deben desarrollarse de manera interna como procedimientos de optimización (*Back-end*).

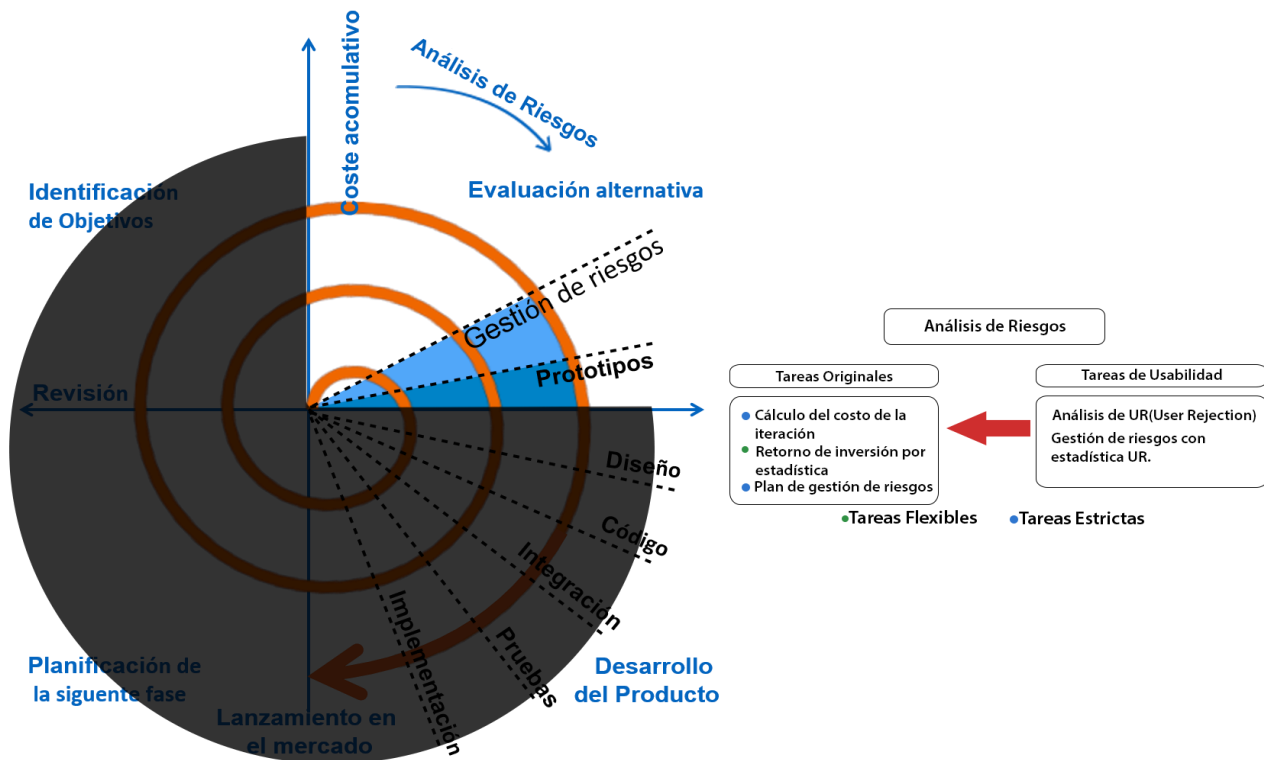


Figura 3.18: Adaptación de tareas de Usabilidad a la etapa de análisis de riesgo.

Si para esta parte del proceso en la primera iteración resulta un UR mayor a 70% es recomendable replantear la idea principal del producto, ya que como se explica en la Sección 2.4.1 del marco teórico, un producto que no cumple con los puntos principales requeridos por un usuario al inicio tiende a presentar mayores problemas en etapas posteriores.

Para el cuadrante tres del modelo adaptado se definen tareas referentes al desarrollo del sistema. En esta parte se tiene cierta similitud con lo que demanda el modelo en Cascada.

Las tareas flexibles en esta parte del proceso (Figura 3.19), incorporan información que se conoce hasta el momento del usuario final y busca crear un primer prototipo con los elementos más significativos.

Cuando se tiene un prototipo inicial dentro de esta etapa, se incluyen tareas de verificación de interfaz y Usabilidad. Los resultados obtenidos en las pruebas son la base para la siguiente parte del proceso y nos permite conocer datos numéricos sobre qué elementos requieren un rediseño.

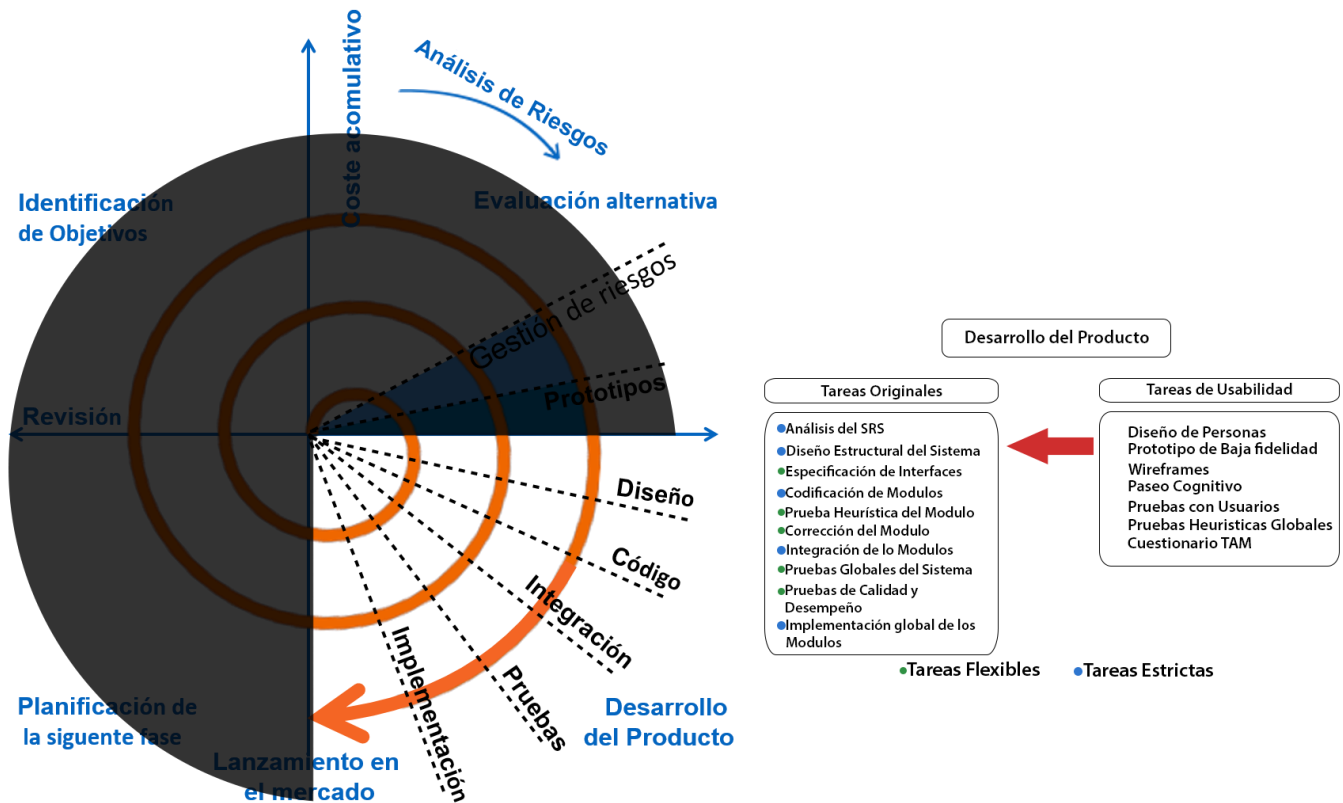


Figura 3.19: Adaptación de tareas de Usabilidad a la etapa de desarrollo del producto.

Para la última parte del proceso, se utilizan tareas de planeación en la que los resultados obtenidos en la parte de pruebas definen la prioridad en los cambios que se tienen que hacer. Los elementos del sistema que resultaron ser confusos, difíciles de operar o inesperados para el usuario pueden requerir de una investigación más a fondo antes de continuar. Como se puede ver en la Figura 3.20, es posible agregar un plan de pruebas adicional si la dinámica de la parte anterior no fue suficiente.

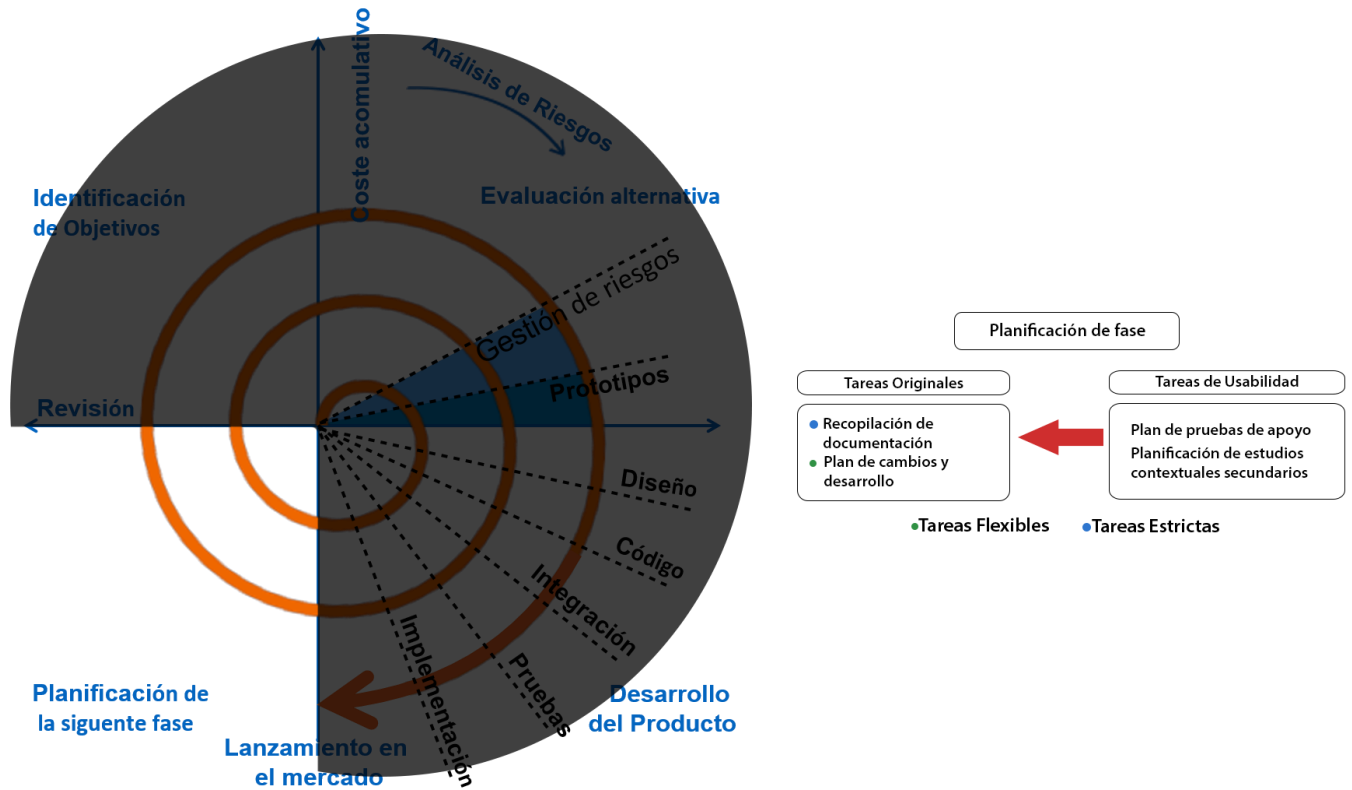


Figura 3.20: Adaptación de tareas de Usabilidad a la etapa de operación y mantenimiento.

Antes de iniciar una nueva iteración es necesario analizar el tiempo restante para completar el proyecto de manera que podamos acortar ciertas tareas o disminuir el número de usuarios para la prueba de Usabilidad en la etapa de desarrollo.

Aunque para este trabajo de investigación se busca definir el proceso de adaptación de ambos modelos tradicionales para la inclusión de tareas de Usabilidad, es necesario respetar aquellas reglas que requieren de métricas en cuanto a costo, tiempo y esfuerzo. De esta forma algunas tareas se conservaron como estrictas y pueden realizarse sin problema.

Para poder hacer un uso completo de esta propuesta es necesario que se incluya el diseño de una herramienta prototipo la cual contenga toda la información necesaria para que los equipos de trabajo puedan guiarse en el proceso de adaptación.

En este caso se plantea el uso de un dispositivo con el cual los equipos de trabajo ya cuenten y estén familiarizados. De acuerdo al estudio inicial con las empresas participantes se determinó que el teléfono inteligente es la herramienta más apropiada con la cual implementar un prototipo. Como se puede ver en la Tabla 4 se consideraron otros medios los cuales pueden ser una alternativa en un futuro.

Tabla 4: Resultado de la encuesta sobre medios apropiados para el prototipo.

Medio Propuesto	Aceptación
PC	86%
Smartphone	34%
Tableta	34%
Papel	18%

3.5 Diseño de la herramienta prototipo

Como última parte es necesario que la herramienta prototipo permita llevar un seguimiento en tiempo real del proceso de adaptación dentro de los equipos de desarrollo, todo esto utilizando interfaces que contengan los datos más relevantes para los usuarios.

Las opciones importantes que debe incluir el prototipo son referentes a la visualización y descripción de cada tarea en Usabilidad utilizando como referencia el modelo. La opción de poder almacenar estadísticas del resultado obtenido después de realizar cada tarea y finalmente el poder visualizar el avance que se tiene referente a cada etapa del modelo Cascada y Espiral (Figura 3.21).

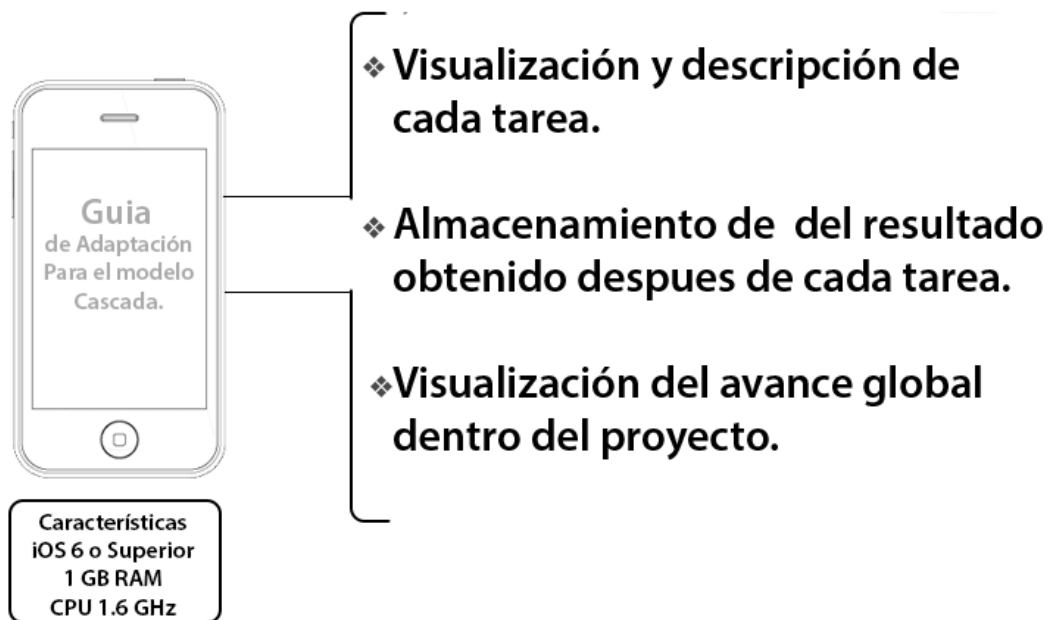


Figura 3.21: Características del dispositivo requerido y opciones de la herramienta.

El diseño del prototipo se construye utilizando el modelo de Diseño Centrado en el Usuario. A través de este modelo podemos realizar modificaciones a la estructura y diseño hasta obtener los resultados deseados todo esto incluyendo los estudios realizados con los usuarios finales.

Para las fases de entendimiento y estudio del problema, se utilizó la información recopilada en la primera fase respecto a las prácticas actuales de desarrollo de software, además se consultaron artículos relevantes para la construcción de un prototipo enfocada a usuarios principiantes. (Arnowitz 2007; Chua 2003)

Para la parte de diseño, se crearon diferentes plantillas tipo *sketch* con el fin de establecer las características visuales de la aplicación. En la siguiente etapa, se construyeron dos versiones de prototipo, una en papel y la otra utilizando herramientas digitales.

3.5.1 Diseño de Interacción

El diseño y construcción de los diferentes módulos con los cuales el usuario tendrá interacción debe cubrir las necesidades básicas de accesibilidad, control de errores y manejo de datos.

Con el objetivo de lograr que la herramienta proporcionara la información necesaria en el momento que el usuario la requiera, fue necesario establecer la relación entre una solicitud y una respuesta por parte del servidor.

El proceso de interacción comienza cuando el servidor proporciona la interfaz para el usuario con el objetivo de que este ingrese su información de ingreso. En la Figura 3.22 se explica el flujo de información que sucede desde el momento en que el usuario comienza a hacer uso de la herramienta hasta que el servidor almacena el trabajo realizado.

La interfaz que se debe diseñar para cumplir con este modelo de interacción debe contener un número limitado de objetos interactivos en pantalla en este caso máximo 5, todo esto para evitar que el diseño para este tamaño de resolución pierda consistencia o despliegue demasiada información que dificulte una correcta lectura.

Otros aspectos como mensajes emergentes, notificaciones en modo de espera y animaciones entre pantallas no forman parte de modelo principal de la herramienta por lo que no se listan como interacciones básicas para un correcto funcionamiento.

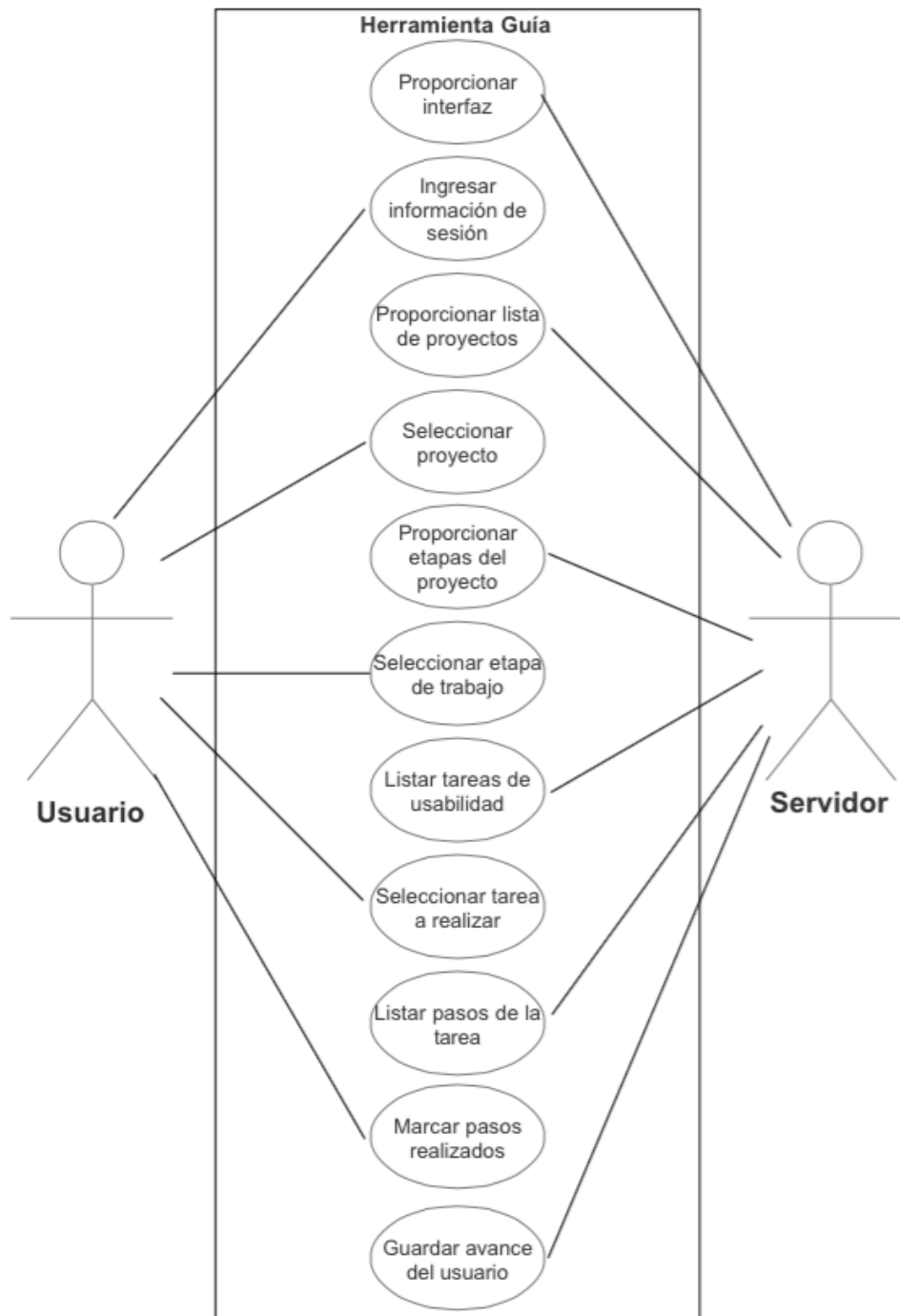


Figura 3.22: Diagrama UML de interacción.

3.5.2 Descripción de casos de uso

Caso de uso: Ingresar información de sesión.

Descripción: El usuario debe ingresar sus datos para inicio de sesión.

Flujo de datos:

- El usuario selecciona los campos para Usuario y Contraseña.
- El usuario escribe sobre el objeto de tipo input.
- El usuario presiona el botón enviar para continuar con el proceso.
- El usuario espera por una respuesta del sistema.

Caso de uso: Seleccionar proyecto.

Descripción: El usuario selecciona un proyecto de la lista presentada por el sistema.

Flujo de datos:

- El usuario visualiza la lista de proyectos relacionados con su cuenta.
- El usuario analiza el avance de los proyectos.
- El usuario presiona alguno de los proyectos de la lista.
- El usuario espera por respuesta del sistema para continuar.

Caso de uso: Seleccionar etapa de trabajo.

Descripción: El usuario selecciona una etapa de trabajo de la lista presentada por el sistema.

Flujo de datos:

- El usuario visualiza la lista de etapa relacionadas al proyecto.
- El usuario selecciona una etapa para trabajar.
- El usuario espera por una respuesta del sistema relacionada a la etapa seleccionada.

Caso de uso: Seleccionar tarea a realizar.

Descripción: El usuario visualiza la lista de tareas relacionada a la etapa de trabajo.

Flujo de datos:

- El usuario visualiza la lista de tareas relacionadas a la etapa de trabajo actual.
- El usuario selecciona una tarea para trabajar.
- El usuario visualiza los detalles de la etapa.
- El usuario selecciona si desea iniciar con esta etapa.
- El usuario espera por una respuesta por parte del sistema para continuar.

Caso de uso: Marcar pasos realizados.

Descripción: El usuario marca los pasos que realiza de la lista presentada en pantalla.

Flujo de datos:

- El usuario visualiza la lista de pasos relacionados a la etapa.
- El usuario lee la descripción del paso a realizar.
- El usuario marca como realizado el paso.
- El usuario selecciona otro paso para realizar.
- El usuario envía el formulario de pasos realizados.
- El usuario espera por una respuesta del sistema para continuar con el proceso.

3.5.3 Wireframes iniciales de la herramienta

Los prototipos iniciales se diseñaron a lápiz y papel con el objetivo de obtener una rápida retroalimentación por parte del usuario. En esta parte se muestran las secciones principales de la interfaz, elementos de interacción y seguimiento de las tareas que se pueden realizar.

La primera parte que se puede ver en la Figura 3.23 muestra la pantalla inicial de la herramienta en la cual se le pide al participante que ingrese un usuario y contraseña.

Es importante determinar quién es el usuario en un inicio debido a que todas las acciones que realice dentro de la aplicación quedarán registradas como avance de proyecto.

La primera interfaz que el usuario utilizará después de iniciar sesión es el panel de proyectos. Aquí el usuario puede revisar los proyectos actuales en los que participa o administra directamente.

Un elemento importante en esta parte, es la posibilidad de crear un nuevo proyecto desde el inicio, el cual debe ser el primer paso antes de elegir una metodología.

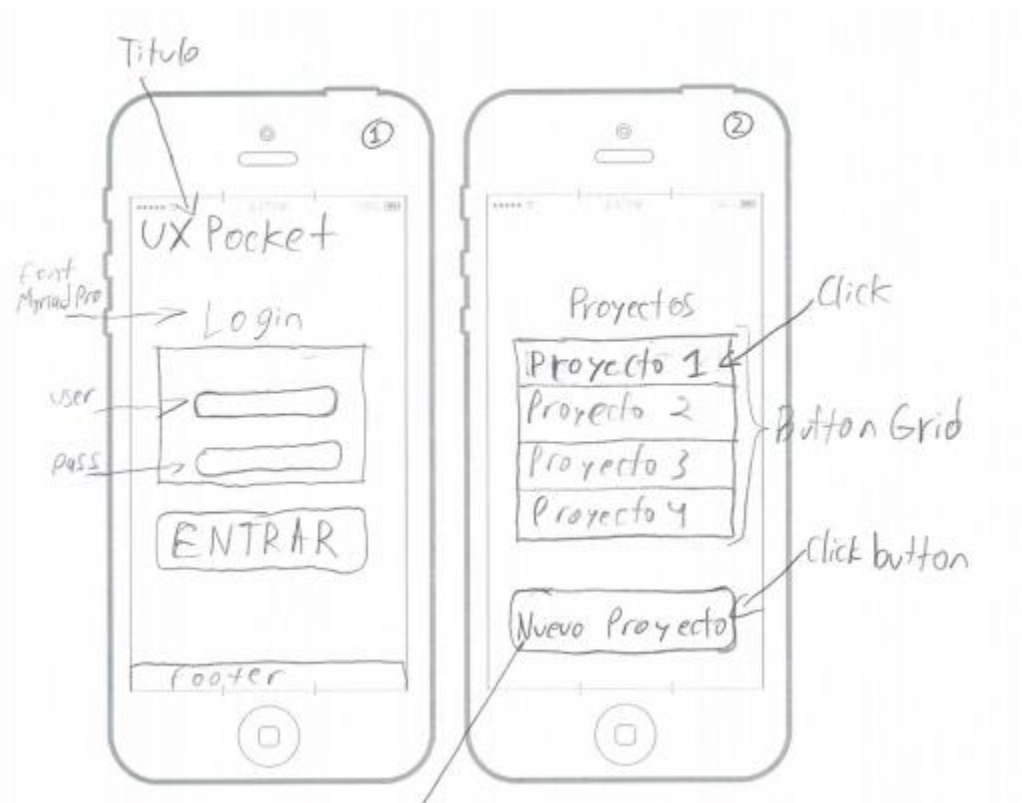


Figura 3.23: Pantalla inicial y Panel de proyectos.

Para registrar un nuevo proyecto se debe contar con dos datos principales que son el nombre y la duración aproximada expresada en días.

La duración del proyecto expresada en días se determinó en base al estudio inicial con desarrolladores los cuales expresaron una preferencia por el manejo en esta unidad a diferencia de semanas o meses.

Después de ingresar el nombre y la duración, el usuario debe seleccionar la metodología principal que llevara dentro del proyecto. Esta selección determina las etapas y muestra el curso que se llevara dentro del proyecto.

Como podemos ver en la Figura 3.24, las etapas se muestran en orden como deben realizarse, además, contienen el nombre y modo de acceso disponible.

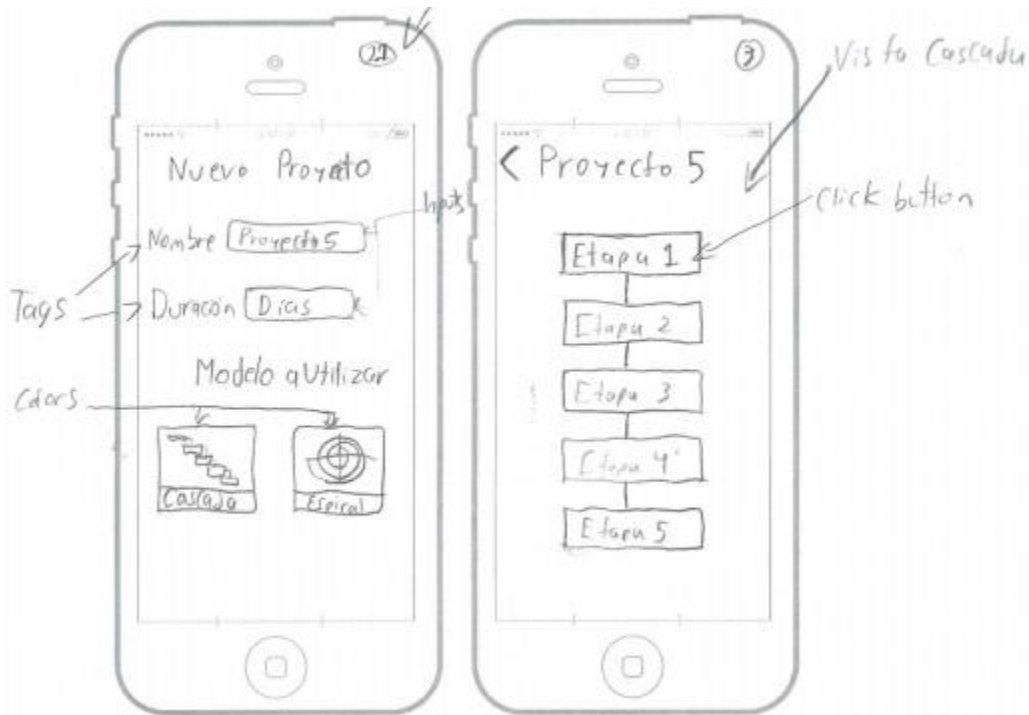


Figura 3.24: Registro de un nuevo proyecto y vista de las etapas.

Las siguientes interfaces que se pueden ver en la Figura 3.25 muestran las opciones que el usuario encontrará al hacer clic dentro de cada etapa.

Al ingresar a una etapa, el usuario deberá proporcionar la cantidad de días que tiene como límite para desarrollar las tareas. Este indicador permite a la herramienta sugerir cuales tareas se recomienda realizar en base al tiempo que se tiene.

Al dar clic sobre una tarea, la herramienta despliega un resumen con la información básica sobre el objetivo de la tarea y el resultado esperado una vez concluida.

Otro elemento que se incluye en esta interfaz es la posibilidad de obtener un resumen del estado actual del proyecto, al cual únicamente se puede ingresar si se encuentra seleccionada la etapa.

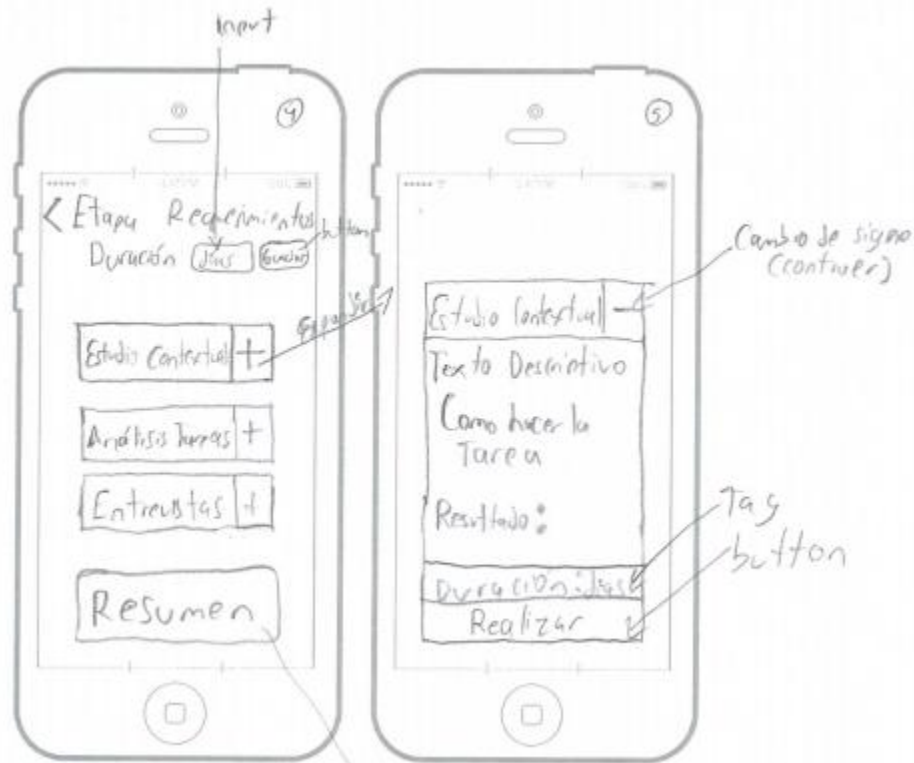


Figura 3.25: Vista de las tareas dentro de la etapa.

Cuando el usuario decide realizar una tarea, la herramienta muestra la serie de pasos que debe seguir para obtener un resultado (Figura 3.26). Los pasos están en orden de realización y cuentan con un control que permite llevar un seguimiento de cada uno de ellos.

Dependiendo del tipo de tarea que debe realizar, el usuario encontrará información relevante, formatos en los que se puede apoyar y enlaces a documentación de apoyo en caso de tener otras dudas que la herramienta no pueda responder.

La vista de resumen permite tener estadísticas actuales del proyecto, en este apartado se separan las tareas completadas de las pendientes y se habilita la opción de poder mandar estos resultado por correo a cualquier colega.

Estos datos se determinaron como los principales en base al estudio inicial con administradores de proyecto.

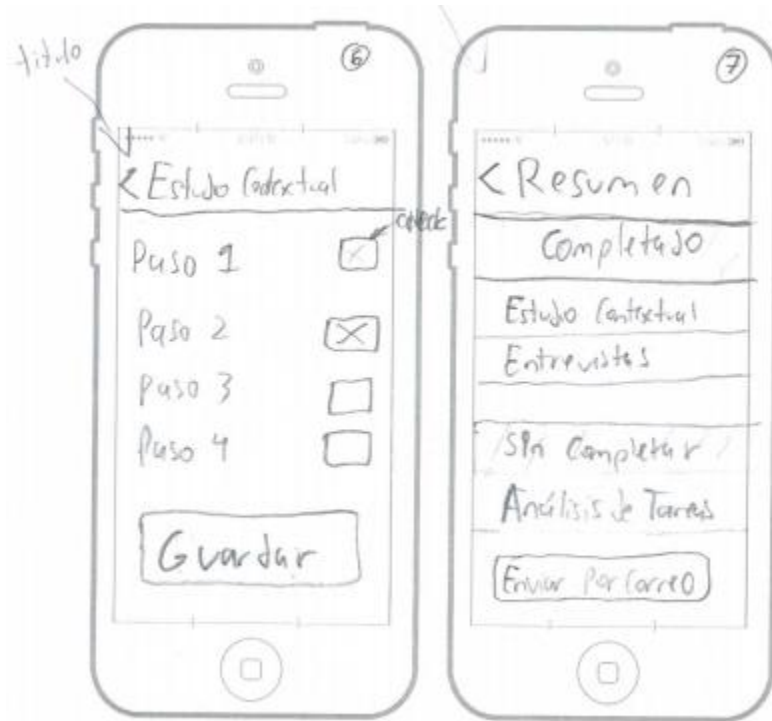


Figura 3.26: Adaptación de tareas de Usabilidad a la etapa de operación y mantenimiento.

En caso de que el usuario decida utilizar una metodología en Espiral, la herramienta permite una fácil selección del cuadrante en el que está trabajando. Como se puede observar en la Figura 3.27, es posible crear nuevas iteraciones o simplemente seleccionar el cuadrante en que se quiere trabajar.

Una vez seleccionado el cuadrante de trabajo, el usuario puede acceder a las tareas pendientes que se sugiere realice. En este panel se utilizan los mismos elementos que en la metodología en Cascada con el fin de mantener un estándar de diseño dentro de la aplicación.

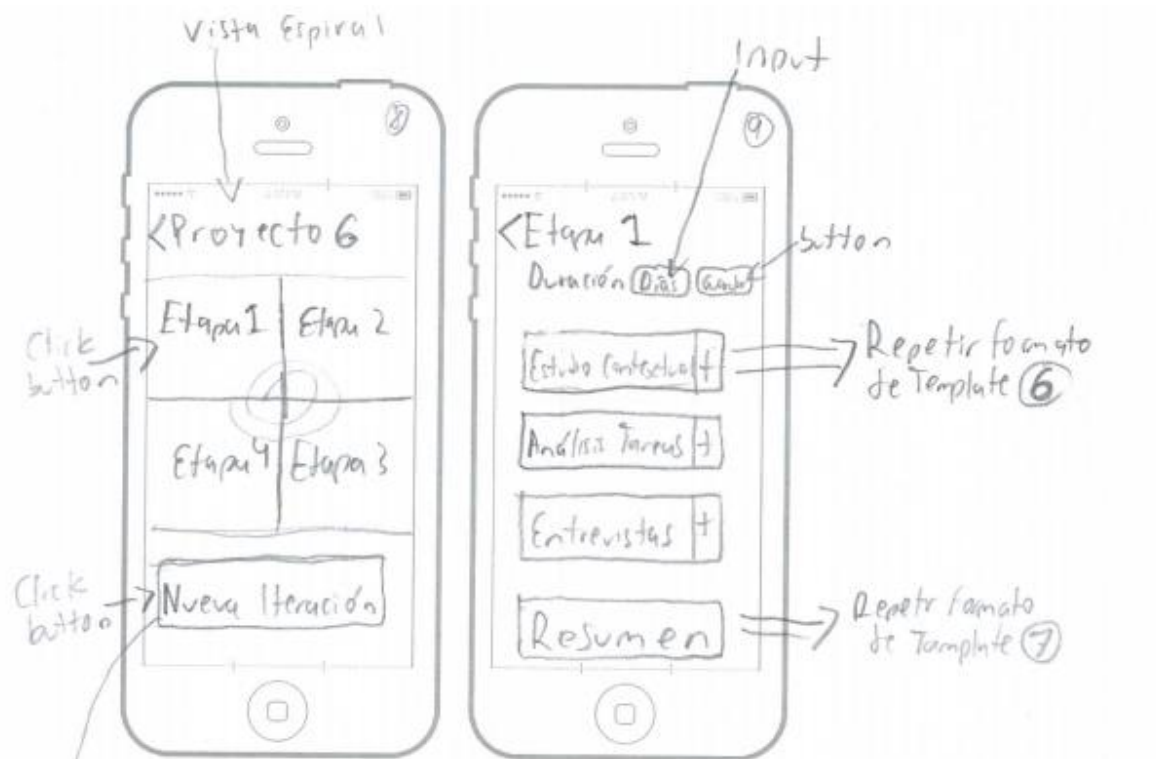


Figura 3.27: Vista de un proyecto con modelo en Espiral.

Cuando el proyecto se encuentra en una segunda iteración, el usuario puede visualizar el cuadrante nuevamente e ingresar a cada uno de ellos para seguir realizando tareas. Un apartado adicional que los usuarios solicitaron es la vista de miembros participantes.

En la Figura 3.28 podemos observar un despliegue de información con el nombre del participante y una opción para mandar documentos directamente.

Adicionalmente es posible marcar el proyecto como finalizado desde la interfaz de miembros.

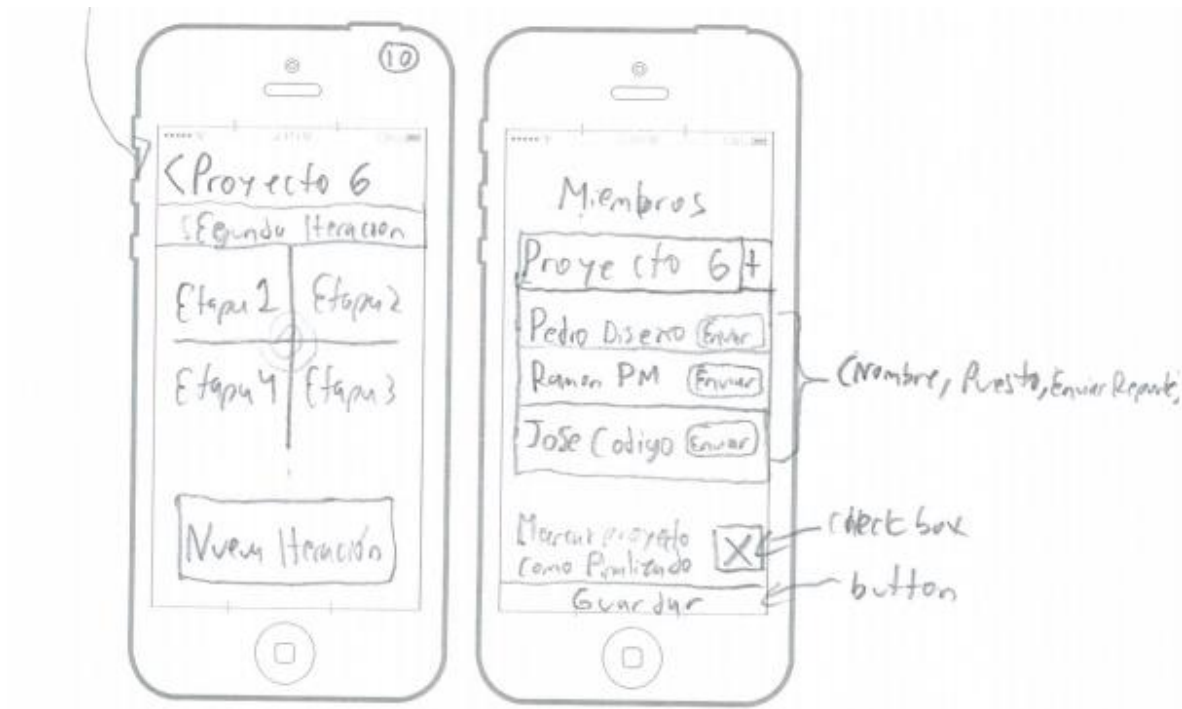


Figura 3.28: Visualización de una segunda iteración y miembros del proyecto.

3.5.4 Diagrama de la base de datos y modelo relacional

Para poder almacenar la información necesaria para cada etapa dentro de la herramienta, es necesario utilizar un motor de base de datos que funcione en conjunto con el lenguaje de programación PHP y permita un diseño relacional que funcione. (Sklar, 2014).

El motor utilizado para el modelo relacional funciona con SQL, lo cual permite administrar diferentes relaciones entre el usuario y los diferentes proyectos en los que este puede participar.

El esquema que se puede observar en la Figura 3.29 ilustra la relación que existe entre el proyecto y el modelo que se decide utilizar. Dependiendo del modelo seleccionado, se enlazan los datos de las diferentes etapas con cada una de sus tareas respectivamente.

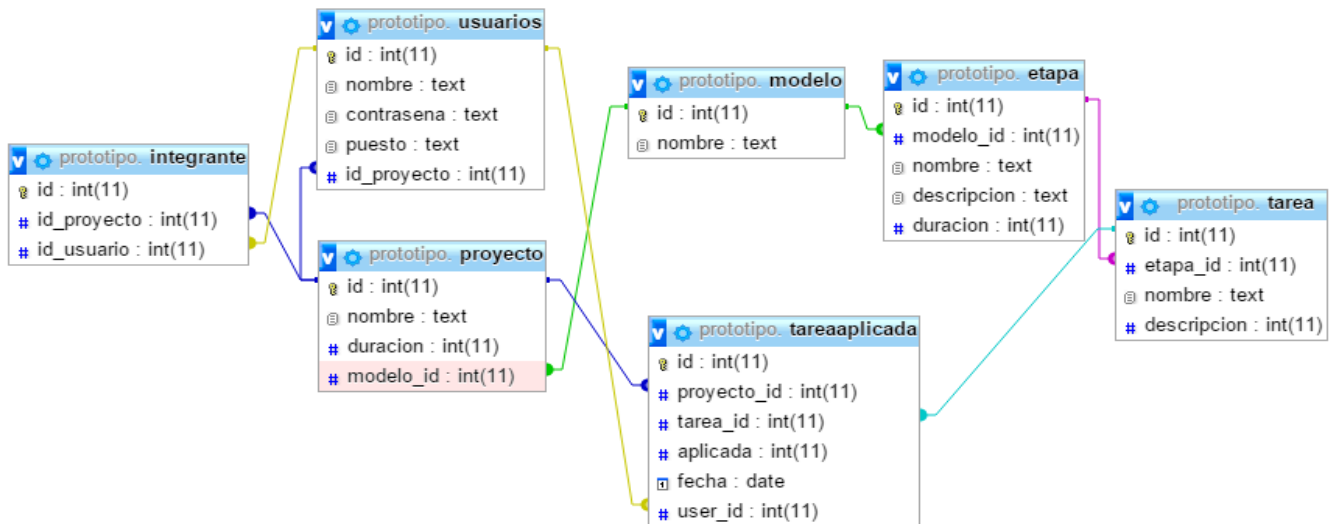


Figura 3.29: Modelo relacional propuesto para la aplicación.

Con este diseño propuesto como modelo relacional, es posible utilizar una de las limitantes más importantes para cada etapa, la cual hace referencia al tiempo total para cada una de ellas y lo compara con el tiempo total que dura el proyecto.

Con el objetivo de facilitar el seguimiento y aprendizaje de la adaptación, el modelo relacional separa el proceso en diferentes tareas las cuales incluyen pasos internos para realizarse.

En la Figura 3.30 se ilustra la jerarquía que tienen las etapas, tareas y pasos dentro de un proyecto.

Esta separación y jerarquía se diseñó pensando en un modelo instruccional para que el usuario pudiera seguir y rastrear de forma fácil en que parte del proceso se encuentra, además es posible delegar y planear de forma clara como se planea llevar el proceso (Stephenson, 2004).

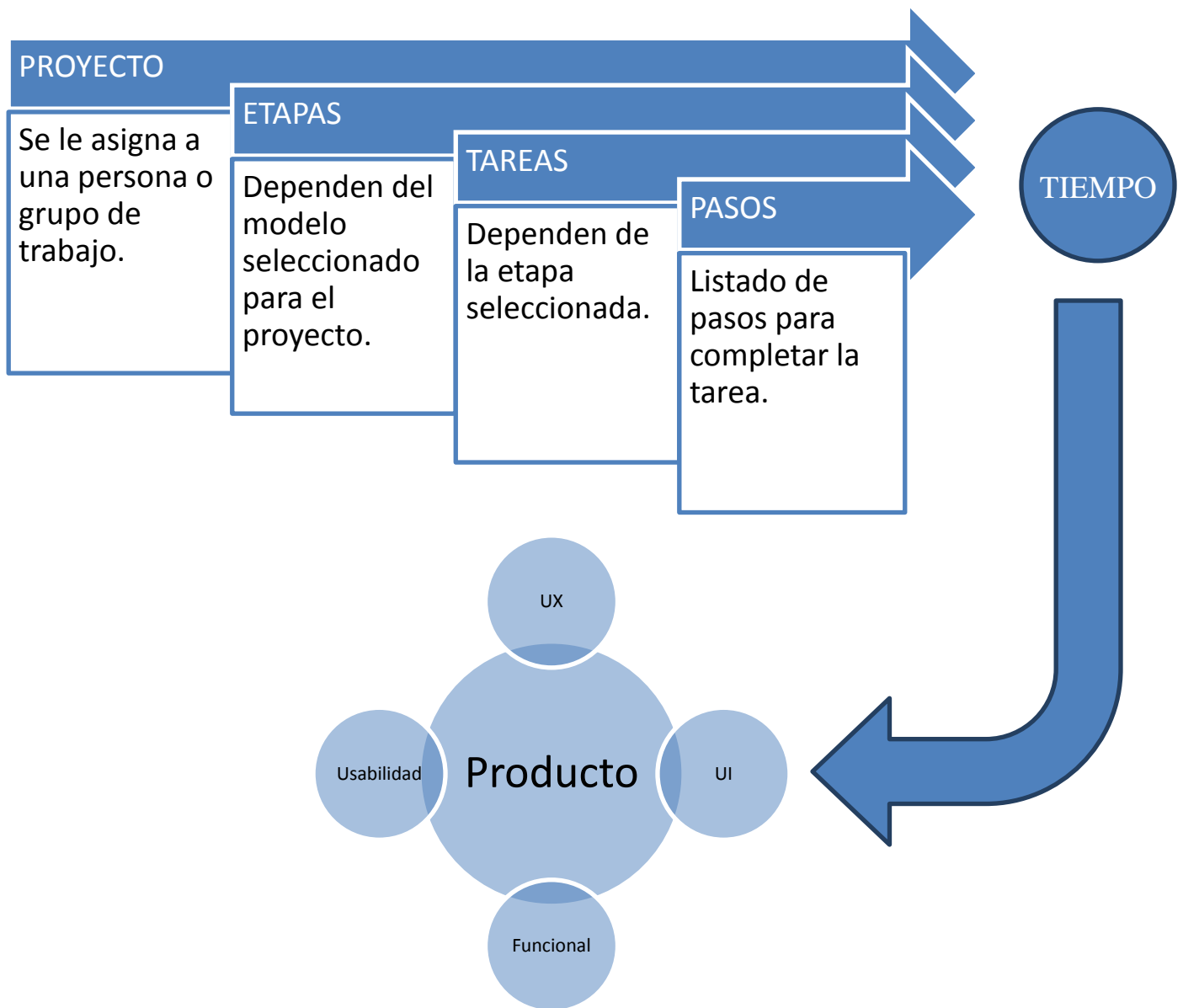


Figura 3.30: Jerarquía de la información dentro del modelo relacional.

3.5.5 Evaluación inicial de la herramienta

Durante la etapa de evaluación del prototipo, se llevaron a cabo pequeñas pruebas iniciales donde se presenta a los usuarios cada una de las interfaces propuestas para la herramienta. El objetivo de esta evaluación fue determinar qué elementos debían modificarse para que el usuario final no tenga problema a la hora de usar el prototipo final de la aplicación.

Las pruebas iniciales de la herramienta se realizaron con usuarios con un perfil relacionado a tecnologías de la información con experiencia mínima en desarrollo de aplicaciones. En la

Figura 3.31 se puede observar 2 de las 5 pruebas realizadas al concepto de interfaz y elementos de interacción.



Figura 3.31: Pruebas de prototipos de baja fidelidad con usuarios.

Los resultados de las pruebas fueron utilizados para crear una segunda versión del prototipo esta vez con elementos interactivos y en un formato compatible con un dispositivo móvil con resolución de 320 x 480 píxeles.

3.6 Versión funcional del prototipo

Esta versión del prototipo cuenta con los cambios sugeridos por los participantes de las pruebas iniciales. Cada uno de los elementos en las interfaces cuenta con una resolución apropiada para dispositivos móviles con una resolución común de 320 x 480 píxeles.

La interfaz de inicio cuenta con los elementos básicos de usuario basado email y contraseña (Figura 3.32). Esta versión cuenta con validación para mostrar si el usuario y contraseña no coinciden.



Figura 3.32: Pantalla de inicio de sesión.

Una vez el usuario ingresa al sistema, se muestran los proyectos actuales en los que participa, una fotografía de su perfil como usuario y la barra de menú con opciones adicionales. Las notificaciones se muestran en la esquina superior izquierda por su importancia en la jerarquía visual (Figura 3.33).

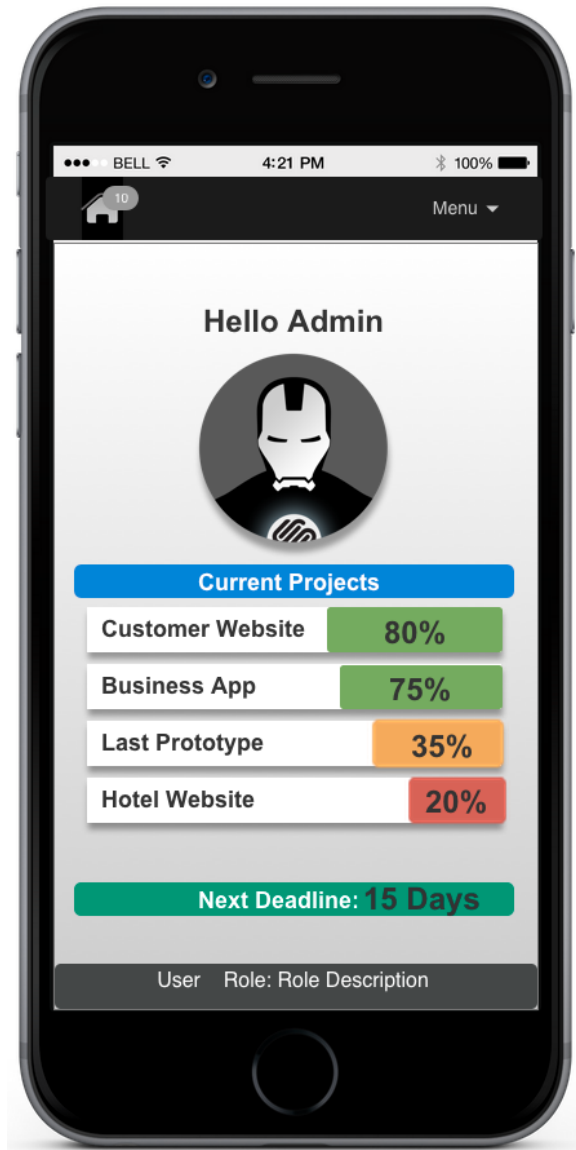


Figura 3.33: Panel de proyectos activos y avances.

Al seleccionar un proyecto se despliega en pantalla la metodología con un esquema del modelo utilizado en el proyecto. Como podemos ver en la Figura 3.34, es necesario mostrar en que iteración se encuentra actualmente con el objetivo de centrar prioridades en las tareas a realizar.

Al presionar sobre un cuadrante se despliegan los componentes y tareas que componen esa parte del proceso.



Figura 3.34: Modelo dentro del proyecto seleccionado.

Dentro de la etapa seleccionada se encuentra el contenido de las tareas que se proponen realizar. En la Figura 3.35 se muestra el diseño basado en una lista de objetos marcables.

Utilizando el concepto de listas TODO (Tareas Por Hacer) es posible llevar el control de lo que el desarrollador hace, además de mostrar detalles relevantes en caso de tener dudas adicionales sobre cómo hacer una tarea.

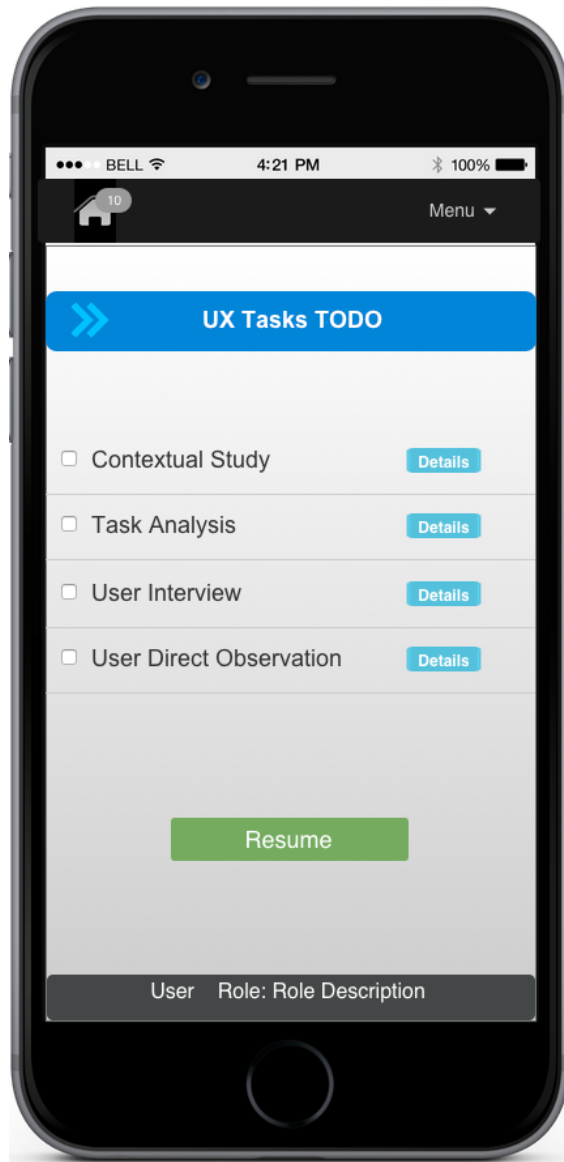


Figura 3.35: Panel de tareas de Usabilidad por hacer.

Para moverse entre las pantallas es posible utilizar el menú o realizar gestos con el dedo, indicando que arrastrar a la izquierda provoca que regreses a la pantalla anterior. Arrastrar el dedo a la derecha provoca que ingreses a la pantalla siguiente (Figura 3.36).

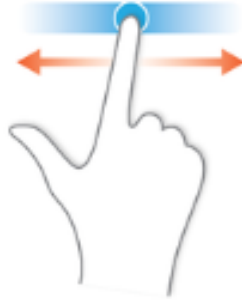


Figura 3.36: Gestos considerados para moverse en pantalla.

3.6.1 Descripción de los usuarios objetivo

Los usuarios a los que va dirigida la aplicación se separan principalmente en 2 tipos: primarios y secundarios.

Los usuarios primarios son aquellos que harán uso extenso de la aplicación, utilizarán la mayor parte de los recursos puestos en pantalla y serán los principales consumidores del servicio.

Utilizando la técnica de proto-personas descrita en la sección 2.5 es posible describir de forma resumida un perfil primario.

Nombre: José Ramírez Sandoval.

Ocupación: Desarrollador Web.

Descripción: José es un desarrollador web que dedica la mayor parte de tiempo a programar la parte interna de los sistemas. Algunas veces le toca programar parte de la interfaz del sistema. José menciona que no es muy bueno con las cuestiones de diseño y que muchas veces sus interfaces cuentan con problemas a la hora de ser evaluadas.

José necesita apoyarse de guías de diseño y tutoriales que encuentra en internet para poder cumplir con los requerimientos mínimos de usabilidad. Sin embargo, esto no es suficiente debido a que la mayoría del material es muy complejo y difícil de entender.

El perfil secundario para el cual se establecería el diseño es para el usuario de tipo administrador de proyectos. Este perfil únicamente requiere que la lista de pendientes permita marcar el avance global de los elementos.

3.6.2 Diseño de la interfaz de usuario

El diseño de los elementos de la interfaz de usuario se eligió pensando en el perfil descrito en la sección 3.6.1. Estos elementos deben ser familiares y fáciles de reconocer desde un primer acercamiento.

El estilo y paleta de colores se basa en el estilo de diseño tipo *flat* (Pratas, 2014) y los colores corresponden a un conjunto de lineamientos incluidos en las librerías Bootstrap (Spurlock, 2013). Esta librería es muy común en el ambiente de desarrollo web por lo que el esquema de colores para indicar si un objeto fue marcado exitosamente o se encuentra en estado de emergencia es muy fácil de identificar.

Un ejemplo del uso de colores estandarizados en esta librería se puede observar en la imagen 3.33 donde los proyectos cuentan con una barra de progreso la cual está separada en tres estados: Buen estado (Verde #5cb85c), estado intermedio (Amarillo#j0ad4e) y mal estado (Rojo #d9534f).

3.7 Diseño de las pruebas al modelo

Para poder probar la adaptación a los modelos tradicionales Cascada y Espiral se diseñó una prueba para los 12 equipos participantes.

Las pruebas se realizan en equipos de 3 integrantes, por cada empresa se seleccionan 2 equipos de desarrollo que tendrán como objetivo diseñar e implementar un sistema de Login.

El Login es uno de los productos de software que contiene mayor peso en un sistema debido a la cantidad de objetos y validaciones que debe contener (Nielsen G, 2011).

Este tipo de producto de software debe contener un nivel de retroalimentación de acuerdo al usuario objetivo, un diseño de formulario adecuado y un contexto visual que servirá de guía para un sistema de software completo, estas características lo vuelve el ejemplo más completo sobre un producto de software integral para web (Nielsen G, 2011).

Por esta razón se pensó que en un lapso de 90 minutos diseñaran e implementaran un sistema con las características que decidieran fueran las apropiadas. Para esta parte se pide a los equipos de trabajo que utilicen el Anexo C (Formato de Persona) como referencia para las necesidades del usuario.

De los dos equipos por empresa, a uno se le entregó un prototipo funcional de la herramienta a la cual denominamos “UX Pocket”. Al otro equipo se le permitió elegir cualquier medio de documentación que ellos prefirieran ya sea desktop, móvil o impreso si así lo desean.

En la Figura 3.32 se explica que del resultado esperado se realiza una evaluación de los niveles de Eficiencia, Eficacia, Satisfacción y TAM. Esta evaluación del trabajo de los equipos es realizada por expertos en el área usando la técnica de evaluación a ciegas con el objetivo de no contaminar los resultados al hacer una comparación.

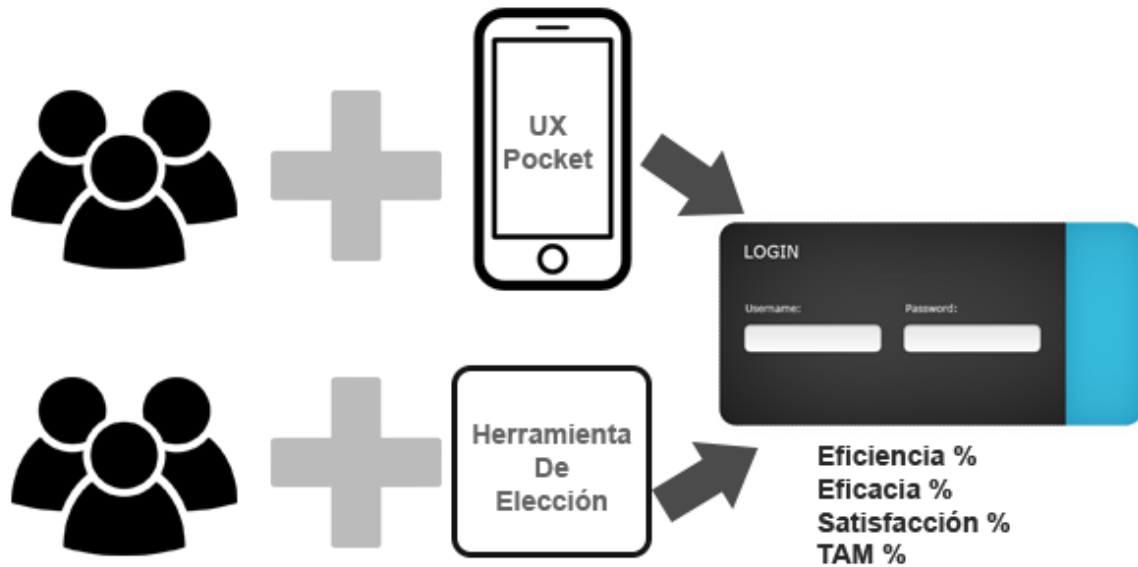


Figura 3.37: Diseño de la prueba a la adaptación usando la herramienta.

4 Resultados y análisis experto

En este capítulo se explican los resultados obtenidos a partir de las pruebas realizadas con la herramienta y la adaptación a los modelos. Cada una de las secciones busca describir el producto de las pruebas con cada uno de los grupos.

Los resultados más importantes se explican como una comparación entre un proceso que se lleva utilizando la herramienta de adaptación y un proceso que utiliza cualquier otro medio seleccionado por los integrantes del equipo por lo que se agrupan dos aplicaciones en una sección.

Como estructura básica de esta sección, se utiliza un formato para análisis experto que puede observarse en el Anexo D. Este formato nos permite explicar por secciones cada una de las observaciones realizadas por el experto evaluador.

Descripción: El objetivo de este análisis experto es el poder evaluar la eficiencia, eficacia, satisfacción y aceptación para el sistema de sesión solicitado a los equipos de prueba. Cada una de las notaciones se encuentra escrita a detalle al final de cada análisis individual a la interfaz.

Las observaciones realizadas a partir del análisis experto se toman como sugerencias para mejorar la interacción en general.

Tareas críticas para evaluar: Las tareas distribuidas a los expertos para ser evaluadas utilizan la mayor cantidad de elementos interactivos para hacer el sistema de inicio de sesión más fácil de usar.

1. Ingresar los datos referentes a Nombre de usuario y Contraseña.
2. Recuperar mi contraseña utilizando información común para el usuario.
3. Enviar mis datos una vez escritos.
4. Registrarme como usuario en caso de no contar con una cuenta.

Las tareas secundarias tienen un menor peso para el sistema en caso de no cumplirlas. Para esta prueba se identifican tres tareas secundarias.

1. Ocultar mi contraseña cuando la esté escribiendo.
2. Tener una longitud máxima para el registro de una nueva contraseña.
3. Evitar que envíe mis datos cuando aún no he colocado el nombre de usuario y contraseña.

Usuario: El perfil del usuario identificado como principal se encuentra descrito en la sección 3.6.1. Un formato con esta información es proporcionada a los expertos para que asuman el rol adecuado ante de evaluar.

4.1 Primera Prueba: Intel Jalisco

La primera prueba fue realizada por dos grupos de desarrollo en el estado de Jalisco por el departamento de desarrollo web de la empresa Intel. A este grupo se le proporcionó el prototipo de la herramienta UXPocket como guía de desarrollo.

Este grupo de desarrollo logro diseñar tres módulos para inicio de sesión que comprenden la opción para recuperar contraseña que se puede ver en la Figura 4.1. El grupo de desarrollo decidió centrar su atención en proporcionar los principales enlaces que podría necesitar el usuario.



Figura 4.1: Interfaz de acceso para el grupo 1.

Otra interfaz de respaldo que se diseñó es el formulario de registro el cual está ligado a otro enlace en la pantalla principal y en la pantalla de recuperación de contraseña lo cual permite al usuario tener un fácil acceso durante el proceso.

En la Figura 4.2 se muestran los campos que el equipo de desarrollo consideró necesarios en un formulario de registro.

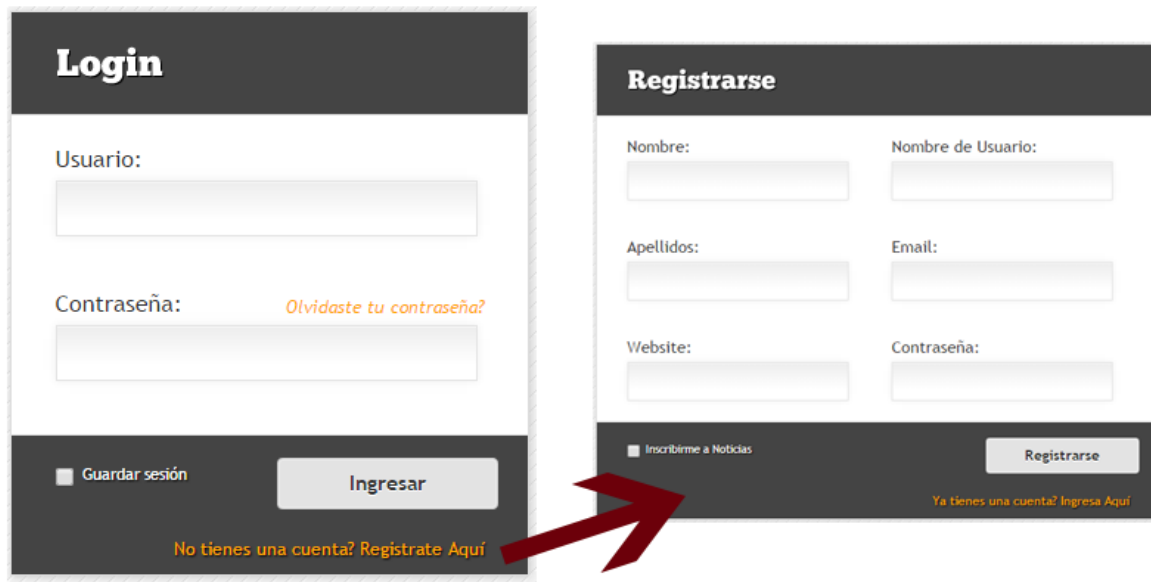


Figura 4.2: Interacción con el menú de registro para el grupo 1.

Tabla 5: Porcentajes Obtenidos en la Evaluación Intel 1.

Evaluador	Mario Alberto Moreno Rocha
Eficiencia	81%
Eficacia	89%
Satisfacción	65%
TAM	76%

Al segundo equipo se le indicó que podía hacer uso de cualquier medio que quisieran como guía para el desarrollo. En este caso lograron desarrollar dos interfaces las cuales tienen un diseño minimalista como argumento para cumplir con reglas de estética.

En la Figura 4.3 se muestra la pantalla principal la cual cuenta con los campos de información y un mensaje de recuperación de contraseña. Este diseño al parecer no cuenta con una opción de registro por lo que no fue posible cumplir con todas las tareas de evaluación.

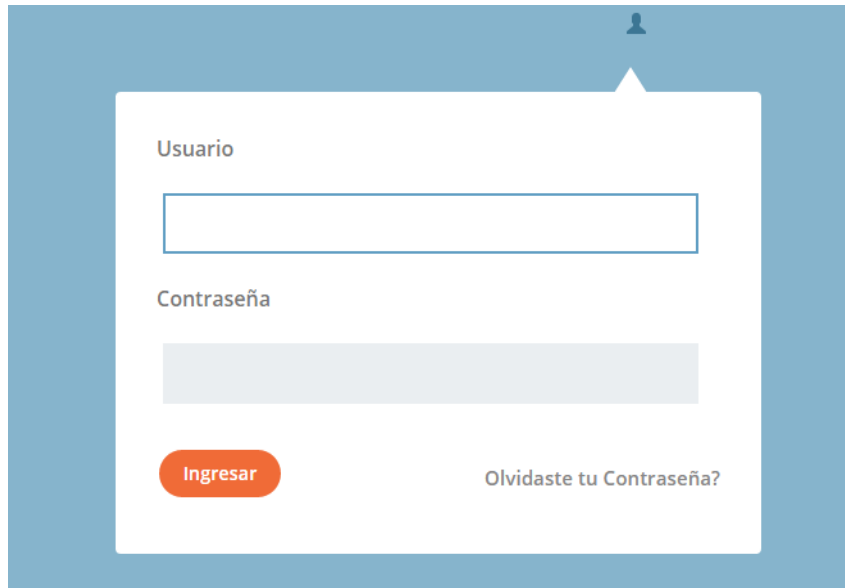


Figura 4.3: Diseño de la interfaz de acceso para el grupo 2

Entre las principales prioridades de este equipo de desarrollo se encuentra la implementación de mensajes de respuesta, por lo que se agregó un mensaje emergente a la hora de enviar los datos.

En la Figura 4.4 se observa que el mensaje emergente aparece en medio de la pantalla para que el usuario pueda leer el mensaje y no lo pierda de vista. Otro cambio sutil en la interfaz es el cambio de color sobre el botón ingresar el cual se argumenta permita al usuario indicar que se encuentra presionado.



Figura 4.4: Mensaje de confirmación para el grupo 2.

Tabla 6: Porcentajes Obtenidos en la Evaluación Intel 2.

Evaluador	Mario Alberto Moreno Rocha
Eficiencia	73%
Eficacia	68%
Satisfacción	69%
TAM	62%

4.1.1 Interpretación de resultados

Severidad del problema: Media. Ambos diseños sólo carecen de problemas de diseño y su falta no imposibilita al usuario acceder a su uso.

Observaciones de la tarea: En el primer grupo al que se le proporcionó la herramienta se tuvieron niveles mayores en los indicadores de Eficiencia, Eficacia y Aceptación debido a que el grupo de desarrollo decidió centrar su esfuerzo en proporcionar todas las interfaces que pudiera necesitar el usuario.

En el segundo grupo de desarrollo se logró un nivel más alto de Satisfacción. En este caso se atribuye a que la estética de la interfaz parece más agradable para el usuario evaluador que aunque no cuenta con una opción de registro es posible cumplir con la tarea de inicio de sesión.

4.2 Segunda Prueba: IBM Ciudad de México

La segunda prueba fue realizada por dos grupos de desarrollo en la Ciudad de México por el departamento de desarrollo web y diseño de la empresa IBM. A este grupo se le proporcionó el prototipo de la herramienta UXPocket como guía de desarrollo.

Este grupo de desarrollo logró diseñar e implementar dos interfaces principales la cuales permiten el registro de un nuevo usuario y el inicio de sesión. En la Figura 4.5 se muestra la interfaz principal en la cual se tienen enlaces para recuperación de contraseña y envío de información.

En este caso se optó por utilizar un icono representativo para acceder al formulario de registro, además de contar con una animación agradable para el usuario la cual evita que se cargue nuevamente la página.

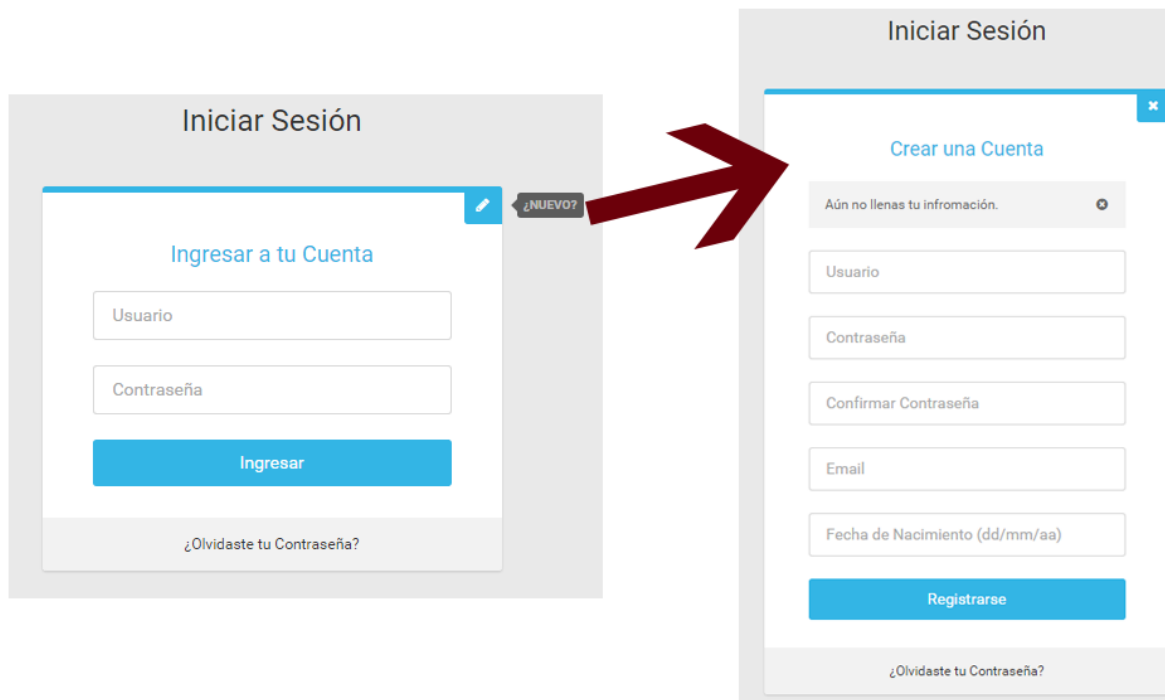


Figura 4.5: Diseño de la interfaz para el grupo 1.

Tabla 7: Porcentajes Obtenidos en la Evaluación IBM 1.

Evaluador	Dagoberto Salas Zendejo
Eficiencia	81%
Eficacia	77%
Satisfacción	82%
TAM	83%

El segundo grupo en esta empresa diseñó e implementó una interfaz que contiene el registro y acceso en un único módulo. En esta prueba no se proporcionó la herramienta prototipo y se indicó que podían hacer uso de cualquier medio como guía.

Este equipo de desarrollo decidió agregar elementos de validación en lugar de crear diferentes formularios.

En la Figura 4.6 podemos observar los campos principales que inician con un registro del usuario. En este caso se decidió iniciar de forma invertida con el argumento de que el usuario muy posiblemente sea su primera vez en el sitio.

Figura 4.6: Interfaz principal para el Grupo 2.

Al momento de ingresar información en los campos, la interfaz muestra ciertos mensajes de validación para el usuario y contraseña. En la Figura 4.7 se muestra la forma en que se notifica al usuario que los datos son aceptables. Al enviar la información se muestra un mensaje de estado.

Figura 4.7: Mensaje de aviso sobre un proceso en espera.

Si el usuario ya está registrado, se proporciona un enlace para que ingrese sus datos de sesión. En la Figura 4.8 se muestra como la interfaz cambió de color y su cabecera ahora indica inicio de sesión. Los mensajes de validación también muestran si los campos son incorrectos o válidos.

The image shows a login interface with the following elements:

- Title:** Iniciar Sesión (orange header)
- Username Field:** A white text box containing the name 'Carlos'.
- Password Field:** A white text box with a blue border, labeled 'Contraseña'.
- Registration Link:** A blue text link that says 'Registrar una nueva cuenta'.
- Feedback Messages:**
 - 'Usuario: Ok' in green text.
 - 'Contraseña: esta vacío' in red text.
- Login Button:** A pink button labeled 'Iniciar'.

Figura 4.8: Formulario de inicio de sesión.

Tabla 8: Porcentajes Obtenidos en la Evaluación IBM 2.

Evaluador	Dagoberto Salas Zendejo
Eficiencia	71%
Eficacia	65%
Satisfacción	71%
TAM	72%

4.2.1 Interpretación de resultados

Severidad del problema: Media. Los diseños propuestos carecen de elementos mínimos de interacción por lo que es posible hacer uso de las opciones aun cuando faltan elementos en el sistema.

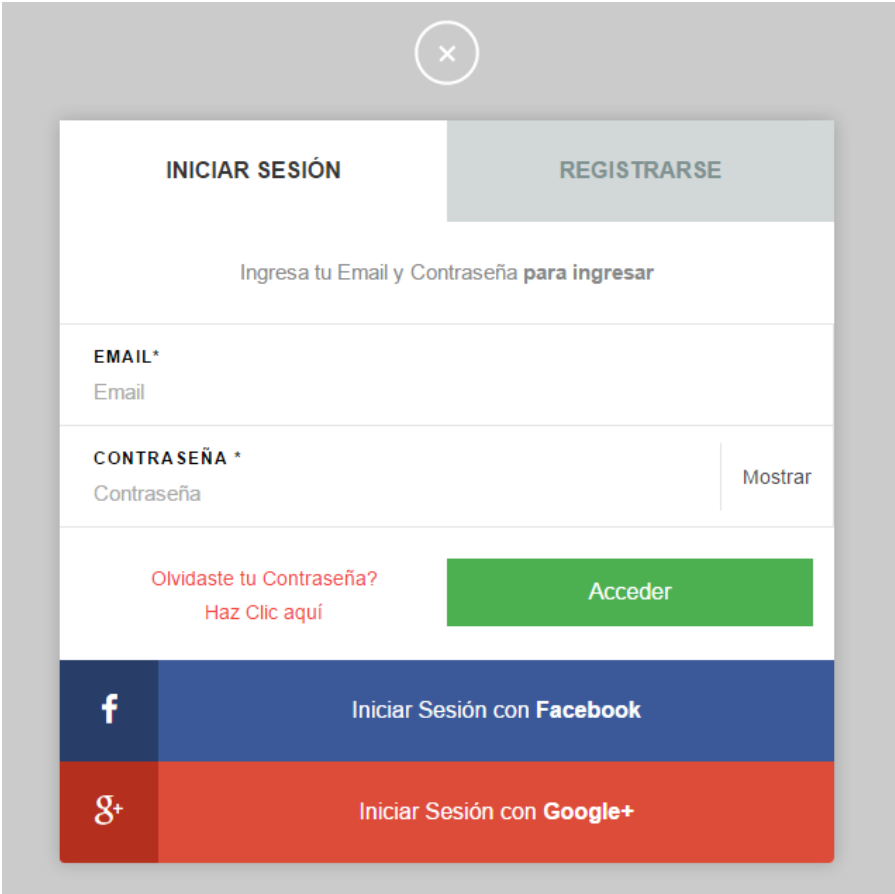
Observaciones de la tarea: Existe una diferencia en cuanto a la forma en la que se construyeron las interfaces. El grupo 2 que no recibió el prototipo como guía decidió centrar su atención en validar los campos. En este caso al usuario evaluador tuvo dificultades para encontrar los enlaces ya que no existe un enlace para recuperar la contraseña.

Los elementos de animación y campos explícitos permitieron al grupo 1 proporcionar lo necesario para el usuario aun cuando no existen elementos de validación.

4.3 Tercera Prueba: Desarrollo UCOL

La tercera prueba fue realizada por dos grupos de desarrollo en el estado de Colima por el departamento de desarrollo web de la Universidad de Colima. A este grupo se le proporcionó el prototipo de la herramienta UXPocket como guía de desarrollo.

Este grupo decidió diseñar e implementar dos interfaces principales la cuales permiten el registro de un nuevo usuario y el inicio de sesión. En la Figura 4.9 y Figura 4.10 se muestran las interfaces utilizadas para iniciar sesión y registrar un nuevo usuario.



El formulario de inicio de sesión se presenta en un modal con un botón de cierre (X) en la parte superior. El formulario está dividido en dos pestañas: "INICIAR SESIÓN" (activa) y "REGISTRARSE". El texto principal indica "Ingresa tu Email y Contraseña para ingresar". Hay dos campos de entrada: "EMAIL*" con el placeholder "Email" y "CONTRASEÑA*" con el placeholder "Contraseña" y un botón "Mostrar" para alternar la visibilidad. Debajo de los campos, hay un enlace "Olvidaste tu Contraseña? Haz Clic aquí" y un botón verde "Acceder". En la parte inferior, se ofrecen opciones de inicio de sesión con redes sociales: "Iniciar Sesión con Facebook" (con el ícono de Facebook) y "Iniciar Sesión con Google+" (con el ícono de Google+).

Figura 4.9: Formulario de inicio de sesión.

El formulario de registro cuenta con elementos tipo *focus* que sirven como guía visual al usuario indicando donde se encuentra activo el cursor. Otro elemento de interacción que se agregó es el uso de la tecla tabulador para moverse entre los campos de forma ordenada.

The image shows a registration form with a close button (X) at the top. It has two tabs: 'INICIAR SESIÓN' (grey) and 'REGISTRARSE' (white). Below the tabs is a heading: 'Ingresa tu información de usuario para crear una cuenta'. The form contains the following fields:

- EMAIL***: Input field with placeholder 'Email'.
- CONTRASEÑA ***: Input field with placeholder 'Contraseña'.
- REPETIR CONTRASEÑA***: Input field with placeholder 'Repetir Contraseña'.

Below the fields is a green button labeled 'Crear Cuenta'. To its left is the text: 'Al crear una cuenta usted acepta nuestra [Política de Privacidad](#)'. At the bottom, there are two social media options:

- Facebook**: A blue button with the Facebook 'f' logo and the text 'Crear una cuenta con Facebook'.
- Google+**: A red button with the Google+ 'g+' logo and the text 'Crear una cuenta con Google+'.

Figura 4.10: Formulario de registro.

Tabla 9: Porcentajes Obtenidos en la Evaluación UCOL 1.

Evaluador	Pedro Cesar Santana Mancilla
Eficiencia	86%
Eficacia	87%
Satisfacción	81%
TAM	89%

El segundo grupo de desarrollo utilizó documentación libre y herramientas de elección definidas por el equipo. En la Figura 4.11 se observa el diseño del formulario de inicio de sesión con los campos básicos de información requerida.

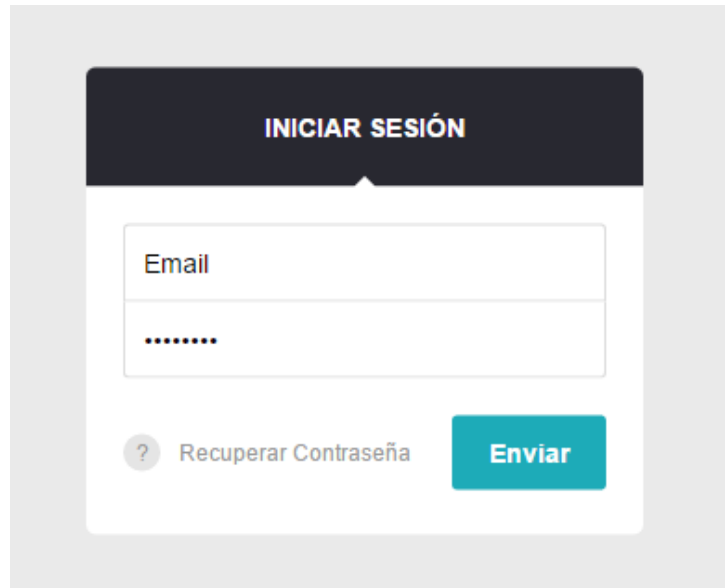


Figura 4.11: Formulario de inicio de sesión.

En la Figura 4.12 se observa que el equipo de desarrollo implementó un tipo de validación para el campo de usuario y contraseña. Para la demostración de un mensaje de validación se optó por una ventana flotante tipo *tooltip* para mayor visibilidad.

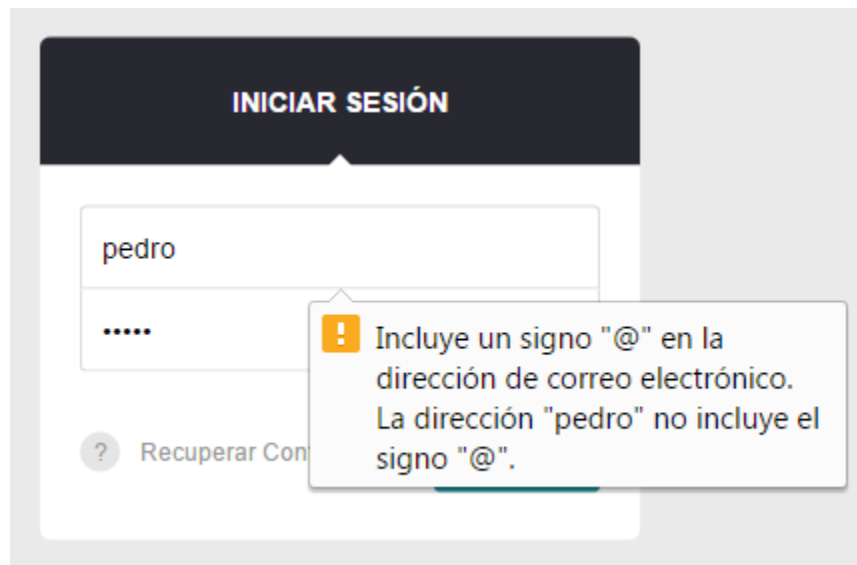


Figura 4.12: Formulario con validación.

Tabla 10: Porcentajes Obtenidos en la Evaluación UCOL 2.

Evaluador	Pedro Cesar Santana Mancilla
Eficiencia	78%
Eficacia	72%
Satisfacción	68%
TAM	74%

4.3.1 Interpretación de resultados

Severidad del problema: Media. Cambios pequeños en el estilo y visualización de las opciones.

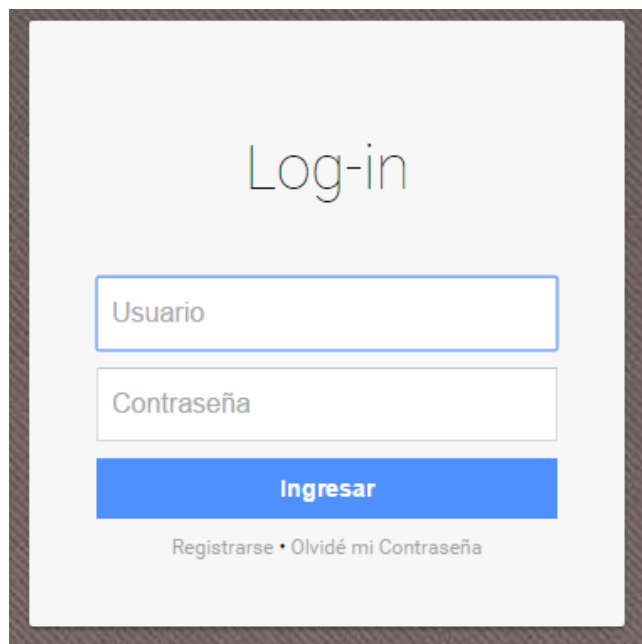
Observaciones de la tarea: La diferencia principal entre los dos diseños radica en el tipo de validación utilizada para guiar al usuario. El grupo 1 decidió invertir su tiempo en colocar la mayor cantidad de elementos visuales que el usuario pueda necesitar. El grupo 2 decidió invertir su tiempo en buscar una forma de validar los campos básicos dejando de lado el diseño de los elementos secundarios.

4.4 Cuarta Prueba: Desarrollo Oaxaca

La cuarta prueba fue realizada por dos grupos de desarrollo en la Ciudad de Oaxaca por el departamento de desarrollo web en LogicalBricks. A este grupo se le proporcionó el prototipo de la herramienta UXPocket como guía de desarrollo.

El diseño de la interfaz de usuario propuesto por este equipo es de estilo minimalista. El formulario principal cuenta con los elementos básicos de usuario y contraseña con algunos efectos para indicar cual control tiene la atención.

Como podemos ver en la Figura 4.13, las opciones para registrarse y recuperar contraseña tienen el mismo peso visual y se representa en el mismo nivel de jerarquía.



Log-in

Usuario

Contraseña

Ingresar

Registrarse • Olvidé mi Contraseña

Figura 4.13: Formulario de inicio de sesión.

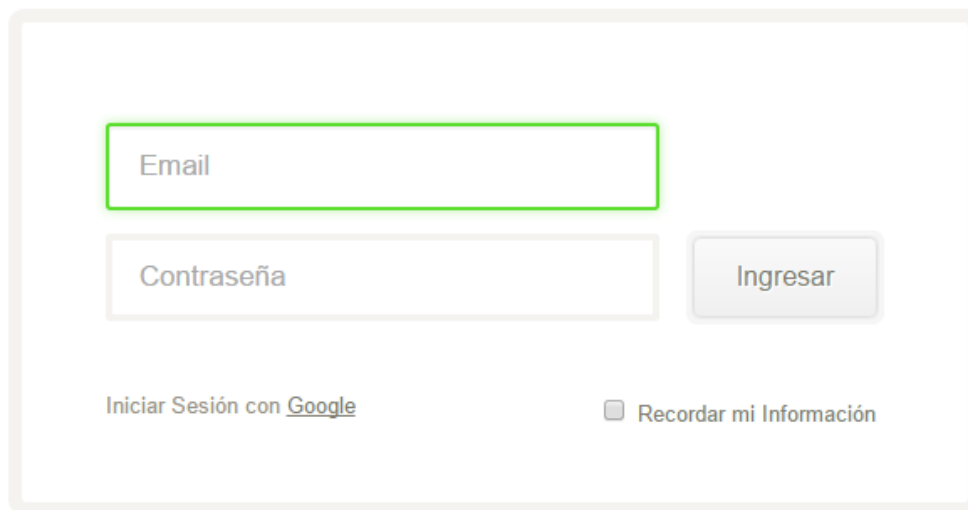
Tabla 11: Porcentajes Obtenidos en la Evaluación Oaxaca 1.

Evaluador	Miguel Ángel Rodríguez Ortiz
Eficiencia	88%
Eficacia	85%
Satisfacción	82%
TAM	89%

El segundo equipo utilizó documentación libre descargada de internet para ser consultada durante la prueba. En la Figura 4.14 se observa una distribución sencilla de los elementos con algunos efectos para resaltar el objeto activo.

En este caso se decidió agregar una opción para ingresar con información de alguna cuenta de Google.

El diseño minimalista con colores básicos provoca un cambio en los resultados en cuanto a satisfacción lo cual se puede ver en la Tabla 12.



The image shows a login form with a minimalist design. It features a text input field labeled 'Email' with a green border, a text input field labeled 'Contraseña', and a button labeled 'Ingresar'. Below the 'Email' field, there is a link that says 'Iniciar Sesión con [Google](#)'. To the right of the 'Contraseña' field, there is a checkbox labeled 'Recordar mi Información'.

Necesitas [registrarte](#) o [renovar](#) tu contraseña?

Figura 4.14: Formulario de inicio de sesión.

Tabla 12: Porcentajes obtenidos en la evaluación Oaxaca 2.

Evaluador	Miguel Ángel Rodríguez Ortiz
Eficiencia	76%
Eficacia	72%
Satisfacción	78%
TAM	79%

4.4.1 Interpretación de resultados

Severidad del problema: Media. Desplazamiento mínimo de elementos de interacción.

Observaciones de la tarea: La posición de los elementos de interacción difiere en ambos casos lo cual provoca que las opciones sean un poco difíciles de encontrar. Los colores en este no suponen un problema.

El Login proveniente del segundo equipo no cuenta con acceso a todas las tareas principales indicadas en la lista inicial. Este tipo de falta provoca que el acceso sea más complicado para el usuario.

4.5 Quinta Prueba: Desarrollo Monterrey

La quinta prueba fue realizada por dos grupos de desarrollo en el estado de Monterrey por el departamento de desarrollo web de la empresa Softeck. A este grupo se le proporcionó el prototipo de la herramienta UXPocket como guía de desarrollo.

En la Figura 4.15 se muestra la distribución y paleta de colores utilizada para la aplicación. La interfaz cuenta con los elementos principales agregando un espacio para registrarse y recuperar la contraseña en la parte inferior.

La distribución visual es proporcional lo cual genera niveles altos de TAM como se muestran en la Tabla 13.

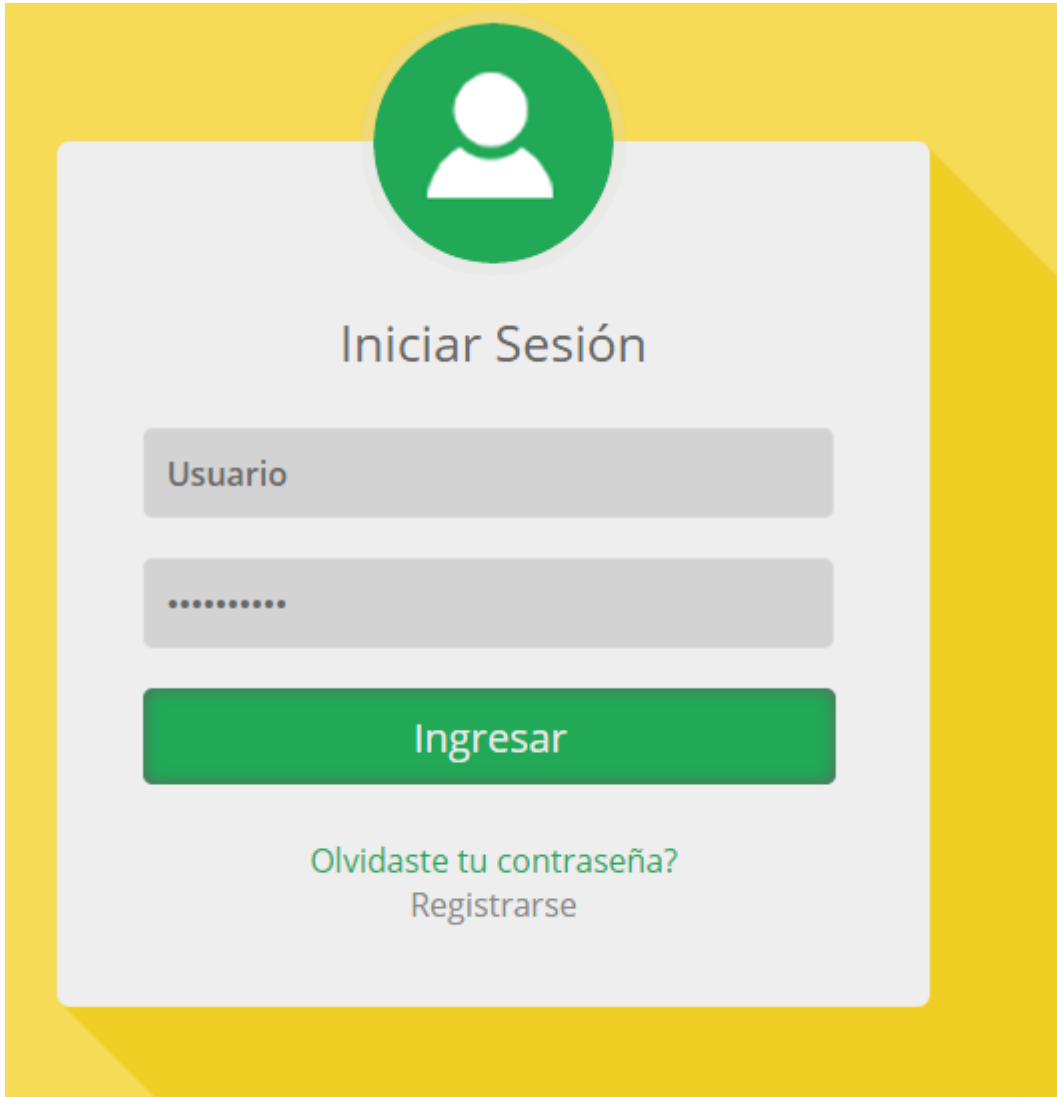


Figura 4.15: Formulario de inicio de sesión.

Tabla 13: Porcentajes Obtenidos en la Evaluación Monterrey 1.

Evaluador	Pablo Joffrey Rodríguez Pérez
Eficiencia	89%
Eficacia	87%
Satisfacción	87%
TAM	91%

El segundo equipo de desarrollo optó por un diseño más pequeño con más elementos de animación sobre el objeto seleccionado. Este equipo decidió que incluir la opción de recordar información es más relevante que un enlace para registro.

En la Figura 4.16 se muestra que la distribución visual de los elementos da mayor peso al recordatorio de información de sesión en comparación con el enlace para recuperación de contraseña.

En la Tabla 14 se muestra que la falta de una opción de registro provoca niveles por debajo de 80% para satisfacción y TAM principalmente.

Figura 4.16: Formulario de inicio de sesión.

Tabla 14: Porcentajes obtenidos en la evaluación Monterrey 2.

Evaluador	Pablo Joffrey Rodríguez Pérez
Eficiencia	77%
Eficacia	79%
Satisfacción	75%
TAM	78%

4.5.1 Interpretación de resultados

Severidad del problema: Media. Falta un elemento de acceso para completar una de las tareas críticas del sistema.

Observaciones de la tarea: El estilo de ambos diseños permite identificar los elementos principales en poco tiempo. La principal diferencia radica en el tipo de retroalimentación visual proporcionada. El primer caso cuenta con un icono representativo del usuario lo cual permite identifica de forma rápida la cabecera del elemento.

A la propuesta del segundo equipo le falta un acceso para el registro de nuevos usuarios. Sin embargo, se cuenta con una opción para recordar los datos ingresados por lo que se cumple una tarea secundaria adicional.

4.6 Sexta Prueba: Desarrollo Puebla

La sexta prueba fue realizada por 2 grupos de desarrollo en el estado de Puebla por el departamento de desarrollo web de la Universidad de las Américas de Puebla. A este grupo se le proporcionó el prototipo de la herramienta UXPocket como guía de desarrollo.

En la Figura 4.17 se muestra la distribución visual de los elementos básicos de la aplicación. En este caso el equipo de desarrollo incluyó un apartado con la fotografía de usuario alegando que es importante proporcionar un poco de retroalimentación incluso antes de entrar.

Este tipo de elementos adicionales provocan un cambio en los niveles de Satisfacción los cuales se muestran en la Tabla 15 por encima del 85%.

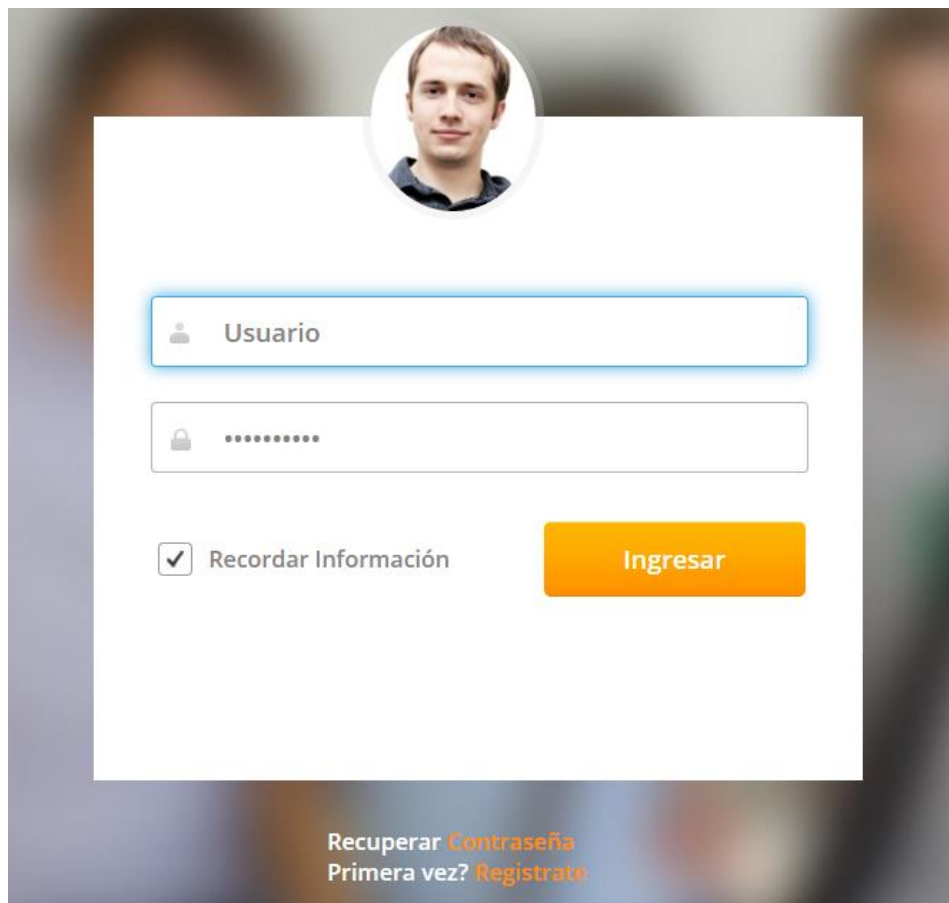


Figura 4.17: Formulario de inicio de sesión.

Tabla 15: Porcentajes Obtenidos en la Evaluación Puebla 1.

Evaluador	Guillermo Ramírez García
Eficiencia	87%
Eficacia	87%
Satisfacción	89%
TAM	86%

El segundo equipo de desarrollo diseñó la aplicación pensando en el ahorro de espacio dentro de la pantalla. En la Figura 4.18 es posible observar que se cuenta con los elementos básicos para inicio de sesión además de incluir la opción para recuperar la contraseña.

En este caso se utilizó un elemento representativo para el envío de información en lugar de un clásico botón con texto. Este tipo de cambio provoca un incremento en los niveles de Satisfacción (Tabla 16) por su simetría con el diseño de los demás objetos.

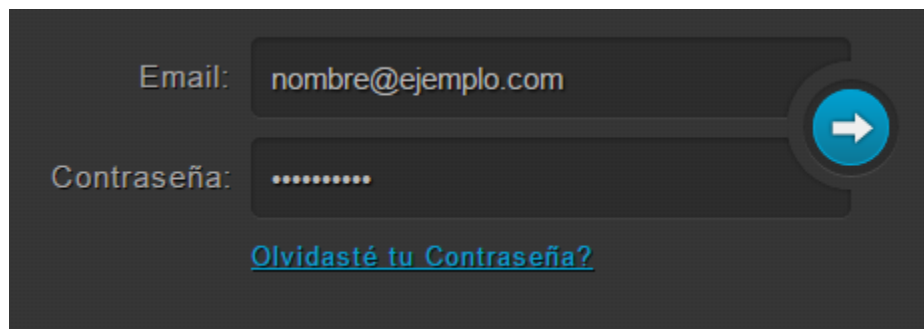


Figura 4.18: Formulario de inicio de sesión.

Tabla 16: Porcentajes Obtenidos en la Evaluación Puebla 2.

Evaluador	Guillermo Ramírez García
Eficiencia	76%
Eficacia	79%
Satisfacción	81%
TAM	76%

Severidad del problema: Media. En este caso es recomendable hacer uso de texto plano sobre el formulario ya que el uso de iconos hace parecer incompleto el diseño.

Observaciones de la tarea: Los elementos divergentes en ambos sistemas tienen su peso en el uso de iconografía en lugar de texto plano para indicar un punto de interacción.

En el primer caso se cuenta con una foto ligada al usuario que se quiere ingresar lo cual hace más fácil la identificación de la cuenta.

El segundo caso no cumple con una de las tareas críticas que refiere al registro de un usuario nuevo por lo que dificulta el trabajo para el usuario objetivo. Este tipo de diseño puede funcionar en sistemas minimalistas pero no asegura una conversión adecuada en cuanto a eficiencia.

4.8 Estadística

En general los niveles de Eficiencia, Eficacia, Satisfacción y TAM muestran una tendencia por encima del 80% en la mayoría de los casos donde se utiliza la adaptación como modelo a seguir. En las Figuras 4.19, 4.20 y 4.22 se muestra que en todos los casos de experimentación se cuenta con niveles mayores durante la comparación de resultados.

Respecto a los niveles de Satisfacción, es posible ver en la Figura 4.20 una variación a la hora de comparar entre los dos escenarios de la prueba. Esta variación es producida por el estilo con el que fueron diseñadas cada una de las aplicaciones.

Al tener un diseño diferente en cuanto a tamaños y una paleta de colores muy diferente, puede provocarse una percepción de satisfacción por parte del usuario aun cuando la aplicación no cuente con los elementos básicos requeridos para un correcto funcionamiento.

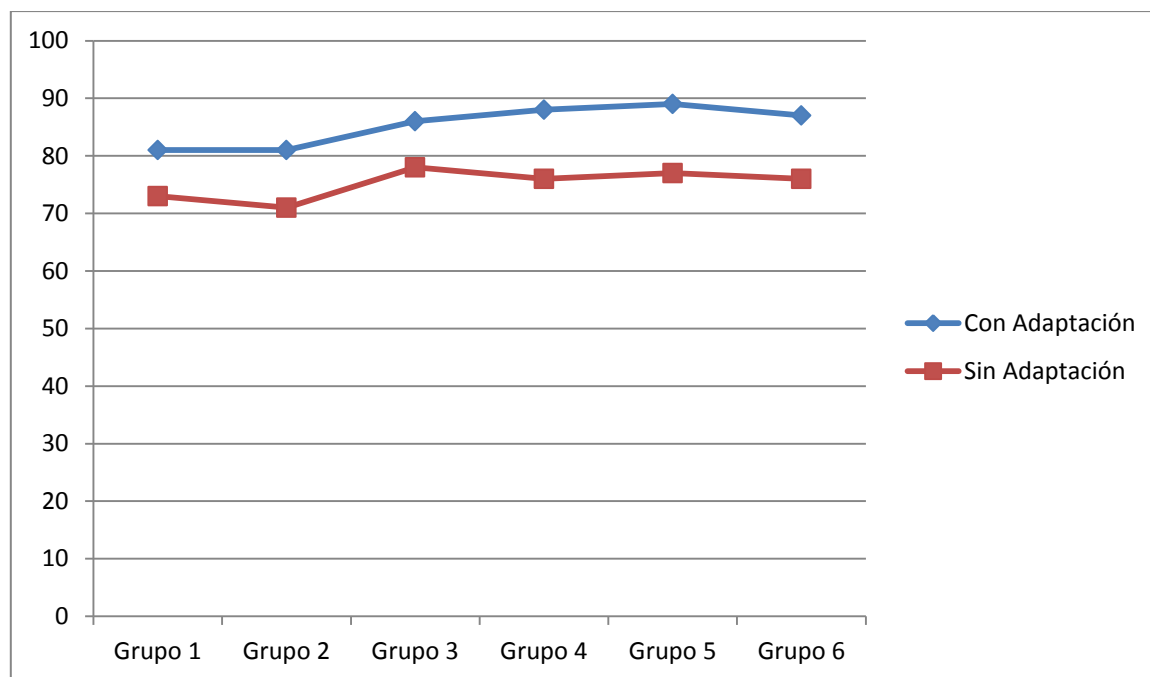


Figura 4.19: Grafica comparativa con los resultados de Eficiencia.

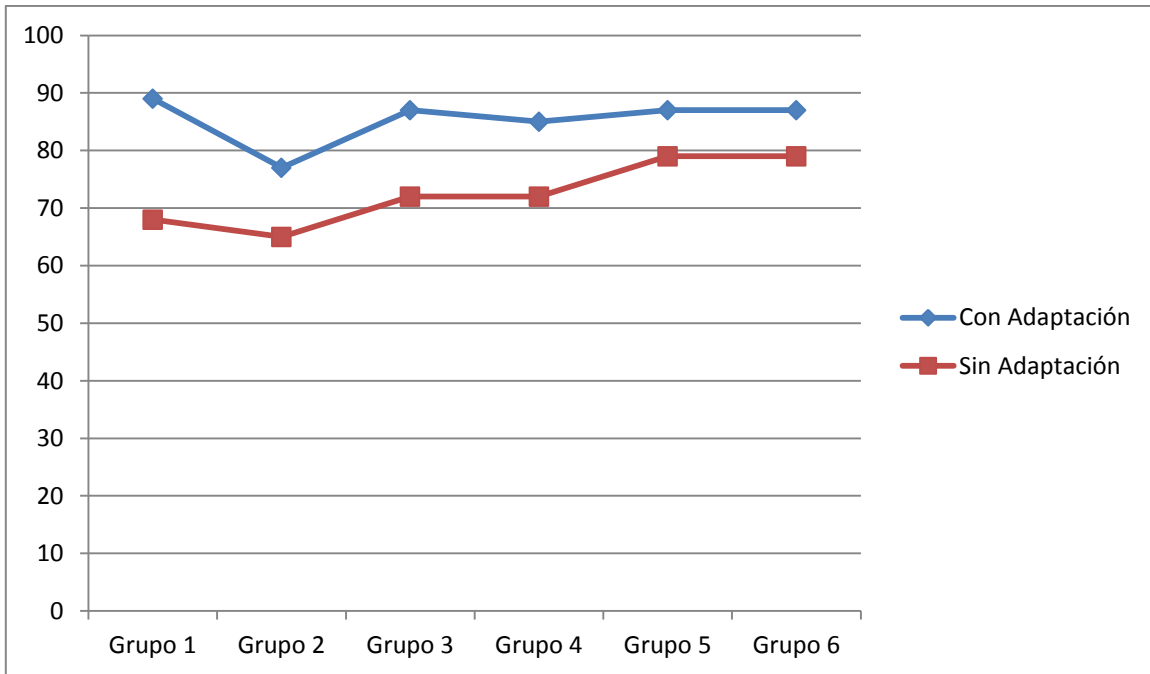


Figura 4.20: Grafica comparativa con los resultados de Eficacia.

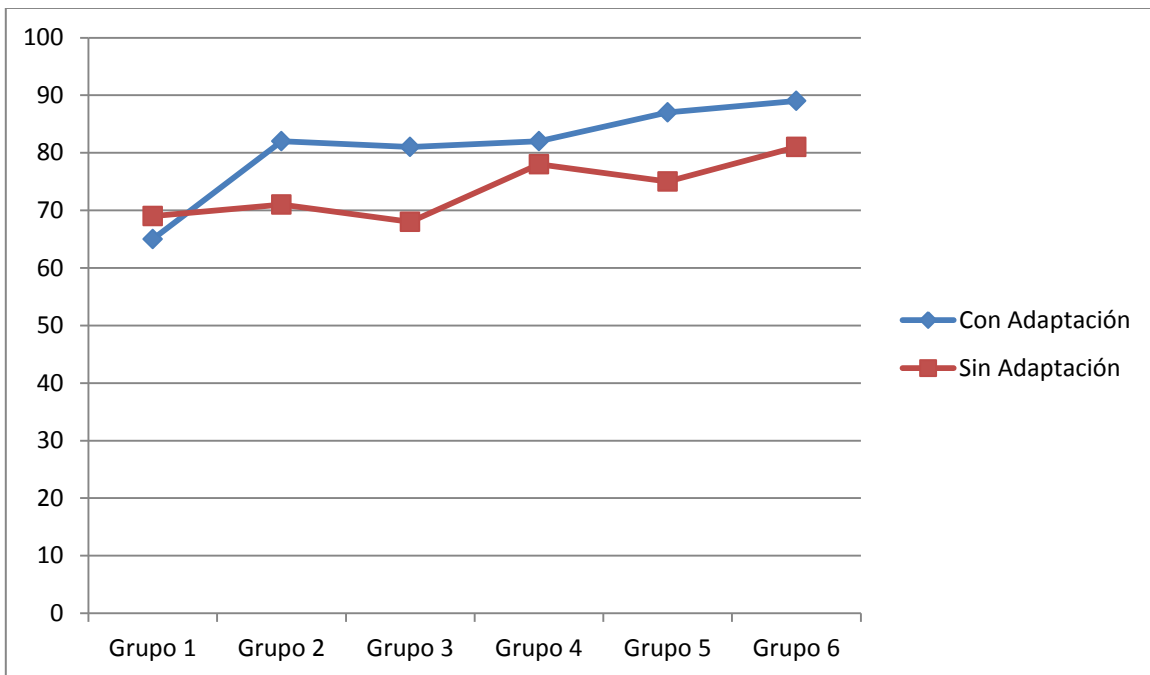


Figura 4.21: Grafica comparativa con los resultados de Satisfacción.

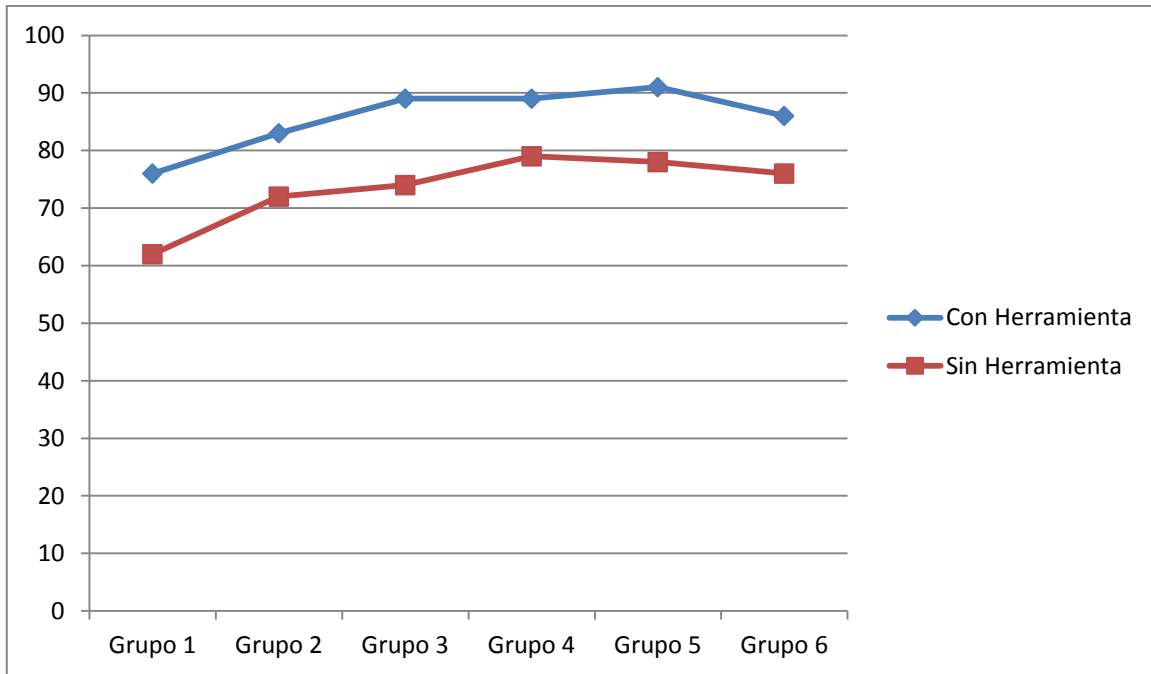


Figura 4.22: Grafica comparativa con los resultados de TAM (Technology Acceptance Model).

La diferencia porcentual (Tabla 17) permite demostrar que existe una tendencia significativa en cuanto al uso de la adaptación a la hora de diseñar aplicaciones web. En los casos donde se tiene una diferencia porcentual menor a 5% se registra como un cambio significativo menor por lo que es descartado como resultado importante.

Tabla 17: Diferencia porcentual entre evaluaciones.

Grupo	Eficiencia	Eficacia	Satisfacción	TAM
Grupo 1	8%	21%	-4%	14%
Grupo 2	10%	12%	11%	11%
Grupo 3	8%	15%	13%	15%
Grupo 4	12%	13%	4%	10%
Grupo 5	12%	8%	12%	13%
Grupo 6	11%	8%	8%	10%

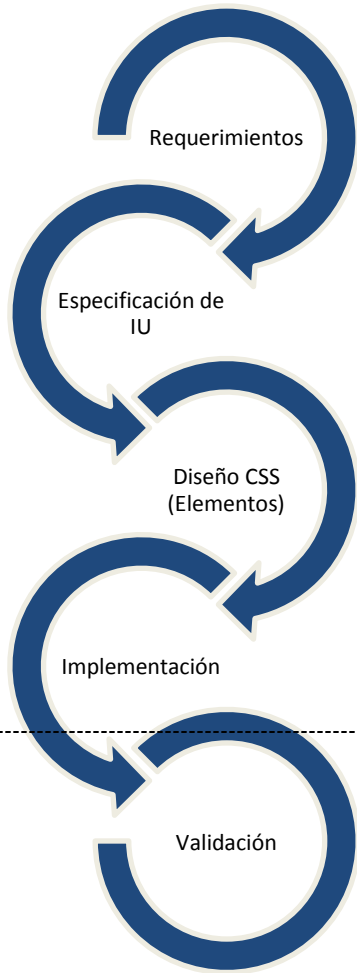
4.9 Diferencias en el proceso

Durante el análisis del proceso seguido por los equipos de desarrollo es posible notar una diferencia en cuanto al orden de las tareas realizadas.

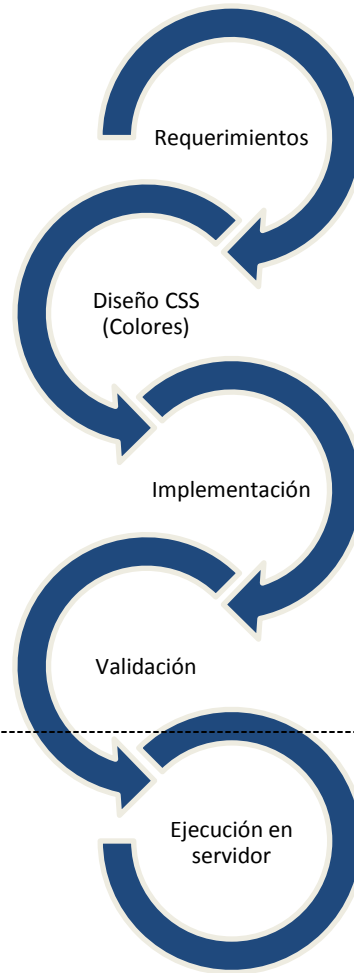
En la Figura 4.23 se observa que los equipos que utilizaron la herramienta guía para la adaptación tienden a incluir una tarea de especificación de interfaz de usuario antes de comenzar a codificar el diseño.

El incluir una tarea de especificación de interfaz de usuario al inicio del proceso provoca que la validación de los campos pase a ser una tarea posterior en el desarrollo. Esta modificación permite especificar la mayor cantidad de elementos necesarios en la interfaz para evitar la realización de cambios en etapas posteriores.

Utilizando la herramienta guía.



Utilizando documentación libre.



Tiempo

Figura 4.23: Comparación entre el proceso seguido por los grupos siguiendo la adaptación y los grupos con documentación libre.

5 Conclusiones y Anotaciones

En este capítulo se concluye si las metas propuestas para el proyecto cumplen los estándares de investigación y la valides de la hipótesis inicial.

Otros aspectos como las limitaciones, confidencialidad y trabajo futuro son descritos como elementos adicionales. Las políticas de privacidad obedecen a los estatutos de seguridad requeridos por las empresas participantes en la fase de pruebas.

5.1 Comprobación de la Hipótesis

Retomando la idea de la hipótesis inicial donde se establece que “Es posible la inclusión de conceptos y estudios de Usabilidad en las fases con más problemas de los modelos de desarrollo de software tradicionales Cascada y Espiral, a través de una herramienta que guíe a los desarrolladores en las adaptaciones propuestas”, podemos concluir que es válida siempre y cuando se respeten las anotaciones descritas en la Sección 5.2.

El proceso de investigación logró cubrir todos los objetivos y metas establecidos al inicio del proceso. Algunos cambios dentro de la meta relacionada al alcance de los resultados sobre los sistemas de software se explican en la sección 5.4 de discrepancias.

En la Tabla 18 se muestra el promedio porcentual en cuanto a los principales indicadores utilizados como variables de medición.

Para los casos donde se muestra una diferencia mayor a 10%, se determina que el cambio es significativo de manera que al utilizar la adaptación se espera que el producto final cuente con mayores niveles de eficiencia, eficacia y aceptación final del usuario (TAM).

En caso de contar con un promedio entre 5% y 9%, se establece que los cambios que se producen son mínimos por lo que se recomienda analizar si utilizar el modelo adaptado es lo más conveniente para el proyecto.

Tabla 18: Promedio entre la diferencia porcentual de las evaluaciones.

Eficiencia	Eficacia	Satisfacción	TAM
10.16%	12.83%	7.33%	12.16%

5.2 Anotaciones

En base a los resultados e interpretación de los datos obtenidos es posible determinar que la hipótesis es válida siempre y cuando se respeten las siguientes reglas:

- La adaptación es posible en cada etapa del modelo siempre y cuando se cumpla el objetivo de cada etapa, de lo contrario ya no sería un modelo completo de desarrollo.
- Tomar en cuenta el estado actual del proceso de desarrollo, esto con el objetivo de no cambiar completamente el esquema de trabajo a mitad del proceso de desarrollo.

Para obtener mejores resultados a la hora de seguir la adaptación, se recomienda seguir el modelo desde el inicio del desarrollo. Aunque la adaptación permite tener modificaciones a las tareas dentro de etapas avanzadas del proyecto, es posible que se tengan problemas de compatibilidad con el trabajo realizado hasta el momento.

Algunos datos encontrados durante las primeras etapas de desarrollo son necesarios para una correcta adaptación en etapas posteriores, algo que sólo es posible obtener si se sigue la adaptación desde un inicio.

5.3 Implicaciones Teóricas y Políticas

El producto de esta investigación permite aportar al área de desarrollo de software dos herramientas personalizadas que utilizan información proveniente de empresas mexicanas.

La primera herramienta es un modelo teórico para la adaptación de los métodos de desarrollo de software Cascada y Espiral. Este tipo de adaptación es uno de los primeros en tomar en cuenta las prácticas utilizadas actualmente por empresas en México.

La segunda herramienta es una aplicación que permite a los desarrolladores conocer el proceso de adaptación y guiarlos a través de cada etapa de forma interactiva. Este aporte sería la primera aplicación documentada desarrollada específicamente para las necesidades de la industria de software en México.

Tomando en cuenta los resultados obtenidos, es posible determinar que los aportes de esta investigación permitirán a futuro una fácil integración del usuario durante el desarrollo de software, algo que se determinó prácticamente inexistente dentro de las empresas.

5.4 Discrepancias

Algunos trabajos de investigación analizados para este trabajo proponen que los modelos no sufran cambios en absoluto y que la investigación con usuarios se realice como tareas adicionales fuera del modelo.

Estas recomendaciones se basan en análisis con empresas en las cuales se tiene una resistencia al cambio bastante significativa, por lo que proponer una nueva adaptación es sumamente difícil.

Los resultados iniciales de la primera parte de la investigación comprueban que en México las empresas tienen mecanismos para la inclusión de nuevos métodos de desarrollo. Estos mecanismos permiten poner en práctica conceptos como los propuestos en este trabajo sin entorpecer la estructura actual de producción en la empresa.

5.5 Confidencialidad

Las políticas de confidencialidad de cada empresa tienen repercusiones en cuanto al nivel de penetración de la adaptación propuesta. Los pequeños cambios en las tareas dentro de cada etapa no se pueden aplicar completamente si las políticas de cambio son bastante rígidas.

Debido a que algunas empresas cuentan con políticas estrictas en cuanto a la difusión de las prácticas internas de desarrollo, es difícil determinar si la adaptación realmente producirá un cambio significativo en los siguientes 5 años debido a que no se puede comprar los resultados obtenidos con resultados estadísticos en proyectos realizados hasta el momento por la empresa.

Mientras más información se tenga de la empresa es posible personalizar en mayor grado la adaptación, lo cual podría dar mejores resultados en cuanto a los niveles de Eficiencia, Eficacia y Satisfacción del producto.

5.6 Trabajos Futuros

El producto de esta investigación puede ser utilizado en distintas áreas del desarrollo de software. En los últimos diez años la industria del desarrollo ha crecido de tal forma que es necesario tener herramientas como la propuesta en este trabajo que respondan a las necesidades de las empresas.

Como implementación a futuro se busca que la aplicación pudiera desarrollarse de forma nativa para otras plataformas móviles como Android y Windows Phone. Si se logra una masificación del servicio, será posible obtener mayor información que permita desarrollar mejoras al producto y sobre todo, que proporcione una mayor cantidad de datos para comprender el proceso de desarrollo que siguen dichas empresas.

Por otra parte se busca que en un futuro se pueda llevar el modelo teórico a otras metodologías de desarrollo. Esta nueva inclusión permitiría realizar adaptaciones a otros modelos como Programación Extrema, SCRUM y Prototipado. (Pries, 2011; Succi, 2001)

En caso de querer comercializar el producto, existe un mercado de consumo bastante amplio, las empresas invierten una cantidad considerable de recursos en herramientas para mejorar la calidad de su producto. Ofrecer esta herramienta en una presentación móvil en

distintos sistemas operativos permitiría hacer llegar la aplicación a más usuarios sin realizar una fuerte inversión. (Ukens, 2007)

5.6.1 Capacitación

El mercado de la capacitación es una las principales áreas en las que se pensaría distribuir a futuro esta aplicación. La herramienta puede ser utilizada para entrenar a desarrolladores en la parte de inclusión del usuario y pruebas de Usabilidad.

Al ser una aplicación capaz de funcionar en un Smartphone, se reduce la curva de aprendizaje que pueda surgir a la hora de utilizar una nueva herramienta. Sería necesario realizar más pruebas con usuarios para determinar el impacto final que tendría si se implementa en las diferentes áreas de desarrollo del producto más allá del departamento de codificación.

Otro escenario en el que se aplicaría más adelante es en las universidades como herramienta de aprendizaje en ingeniería de software. Es necesario realizar más pruebas para determinar si las características actuales del prototipo son las necesarias para ser una herramienta didáctica eficiente. (Troy, 1991)

5.6.2 Recopilación de información

Con el objetivo de seguir recopilando información por parte de desarrolladores, se planea liberar una versión estable de la herramienta para uso público. Una vez se haya liberado esta versión, se planea escribir material de investigación con nuevos resultados para proyectos de desarrollo con una mayor duración.

Los nuevos datos estadísticos serán documentados en artículos de revista en el área de usabilidad.

5.7 Limitaciones Del Estudio

Durante la primera parte de la investigación, el análisis inicial con empresas desarrolladoras de software en México, se limitó el número de participantes a desarrolladores que actualmente trabajen en una empresa en el país, esto con el objetivo de agilizar el proceso de captura y obtención de datos.

Esta limitación no repercute de manera importante en los resultados obtenidos ya que los datos necesarios para validar el diseño de la adaptación eran de carácter general como el tipo de metodología que utilizan y si actualmente involucran al usuario en el desarrollo de sus productos.

Durante la última etapa de pruebas del prototipo final, se limitó la posición geográfica de las empresas participantes a 6 estados de la república: Jalisco, Oaxaca, Distrito Federal, Nuevo León, Colima y Puebla.

El uso de una muestra de usuarios para pruebas del prototipo final únicamente en estos 6 estados, se debió a que actualmente son los lugares con los que se tuvo un acercamiento con empresas que nos permitieran disponer de un grupo de desarrollo.

5.8 Acuerdo de Confidencialidad

La información proporcionada en este documento en la que se hace referencia a las empresas INTEL®, IBM®, LOGICALBRICKS® y SOFTECK® es de carácter confidencial para uso exclusivo con fines de investigación.

Los nombres de los participantes, líderes de proyecto y localización exacta de los grupos de desarrollo permanecerán de forma anónima por un periodo de 24 meses de acuerdo a los estatutos acordados con los empleadores. La duración de este periodo está a sujeta a cambio sin previo aviso.

ANEXO A

Encuesta: Prácticas de usabilidad en las empresas

*Obligatorio

1. Dentro de tu equipo de trabajo. ¿En qué etapa del proceso de desarrollo participas? *

Marca solo un óvalo.

- Definición de requerimientos
- Diseño del software
- Implementación y desarrollo
- Integración y pruebas del Sistema
- Operación y mantenimiento
- Administrador del proyecto

2. Como parte de tu capacitación y entrenamiento. ¿Qué tipo de herramientas utilizas?

Selecciona todos los que correspondan.

- Documentación Online
- Documentación impresa(Manuales)
- Aplicaciones de escritorio
- Aplicaciones móviles
- Otro:

3. De las siguientes metodologías de desarrollo. ¿Con cuáles has trabajado?

Selecciona todos los que correspondan.

- Modelo en cascada
- Modelo en espiral
- Modelo en V
- Prototipado
- SCRUM
- Programación extrema

1. **¿En que etapa del proceso se presentan mayores dificultades?**

Problemas de diseño, implementación, cambios.

Selecciona todos los que correspondan.

- Definición de requerimientos
- Diseño del software
- Implementación y desarrollo
- Integración y pruebas del sistema
- Operación y mantenimiento

Pregunta Adicional

1. **Razón por la cual no tienes acercamiento**

Marca solo un óvalo.

- No lo creo necesario
- En mi trabajo no es parte del proceso
- Lo realiza otra personas
- Otro:

5. De las siguientes tareas de usabilidad. ¿Cuáles conoces?

Selecciona todos los que correspondan.

- Estudio contextual
- Análisis de tareas
- Entrevistas
- Diseño de personas
- Prototipo de baja fidelidad
- Evaluación heurística
- Pruebas de usabilidad
- Evaluación TAM
- Ninguna

6. ¿Crees que es importante utilizar información del usuario en todo el proceso de desarrollo?

Marca solo un óvalo.

- Sí
- No
- Solo en ciertas etapas del proceso

Pregunta adicional

7. Razón por al cual no tienes acercamiento directo con el usuario

Contesta únicamente si tu respuesta fue negativa en la pregunta 4

Marca solo un óvalo.

- No lo creo necesario
- Alguien mas lo realiza
- En mi area no es parte del proceso
- Otro:

ANEXO B

EVALUACIÓN TAM

Nombre: _____ Software evaluado: _____

PERCEPCIÓN DE FACILIDAD DE USO

1) Encuentro el software sencillo de usar	1	2	3	4	5	6	7
2) Aprender a usar el software me resulta sencillo	1	2	3	4	5	6	7
3) Mi interacción con el software es clara y entendible	1	2	3	4	5	6	7
4) Sería fácil para mí encontrar información en el software	1	2	3	4	5	6	7

PERCEPCIÓN DE UTILIDAD

5) Usar este software mejorará mi Eficiencia	1	2	3	4	5	6	7
6) Usar este software mejorará mi desempeño	1	2	3	4	5	6	7
7) Usar este software mejorará mi productividad en el trabajo	1	2	3	4	5	6	7
8) Encuentro este software muy útil	1	2	3	4	5	6	7

ACTITUD HACIA EL USO DEL SOFTWARE

9) Me disgusta la idea de usar el software	1	2	3	4	5	6	7
10) Tengo una actitud favorable hacia el uso del software	1	2	3	4	5	6	7
11) Creo que es una buena idea el uso del software	1	2	3	4	5	6	7
12) El uso de este software es una idea tonta	1	2	3	4	5	6	7

INTENCIÓN DE USO

13) Tengo la intención de utilizar el software	1	2	3	4	5	6	7
14) Volveré a usar el software (cuando sea necesario)	1	2	3	4	5	6	7
15) Tengo la intención de utilizar el software para mi trabajo	1	2	3	4	5	6	7

Escala

1. Totalmente en desacuerdo
2. Muy en desacuerdo
3. En desacuerdo
4. Neutral
5. De acuerdo
6. Muy de acuerdo
7. Totalmente de acuerdo

ANEXO C



Nombre: Luis Ramírez Pérez

Edad: 24 Años

Experiencia con tecnología: Media-Baja

Perfil:

Luis es un estudiante de derecho que utiliza su computadora para realizar sus trabajos y navegar en redes sociales. Aunque Luis utiliza diariamente su laptop, realmente no accede a una gran variedad de sitios en internet.

Con frecuencia olvida su contraseña y no tiene problema alguno con proporcionar información básica como su nombre y correo para registrarse en algún sitio. Su experiencia con sistemas de registro se reduce a formularios relacionados a redes sociales principalmente Facebook, Twitter y Google +.

ANEXO D



Formato de reporte para análisis experto

Descripción: describir el objetivo del análisis experto dentro del proyecto.

Identificación de tarea: Identificar y listar las tareas críticas a evaluar.

Usuario: Identificar los usuarios típicos del sistema.

Observaciones de la tarea: Describir las observaciones a detalle de lo encontrado a la hora de realizar la tarea.

Severidad del problema: Proporcionar un nivel de severidad entre Alto/Medio/Bajo.

Sugerencias para corregirlo: Proporcionar sugerencias en cuento a una posible solución al problema. Las sugerencias se anotan como mejoras a futuro.

Bibliografía

- Arnowitz, J., & Arent, M. (2007). Wireframes and Sketches. In *Effective prototyping for software makers* (1st ed., Vol. 2, pp. 10-85). Amsterdam: Elsevier.
- Bevan, N. (2008). Cost benefits evidence and case studies.(1ra. ed, Vol. 2, Pag. 60-120).Germany:R.G. & Mayhew.
- Boehm, B. W. (1983). A Spiral Model of Software Development and Enhancement. (2da. ed., Pag. 25-50). California, USA.
- Budiu, R. (2013). Our Research: How We Ran the Usability Studies. In *Mobile usability*,(5ta ed., Vol. 1, Pag. 14-35). Berkeley, CA: New Riders.
- Clark, J. F. (2011). MAS 490: Theory and Practice of Mobile Applications,(1ra ed., Vol. 1, Pag. 7-32). EU, University of Kentucky.
- CloudMonkey. (2014). FoneMonkey. Colorado: CloudMonkey LLC. *www.FoneMonkey.com* Consultado 6 Oct.2014.
- Chua, C., & Leong, K. (2003). Wireframes for Beginners. In *Rapid prototyping principles and applications* (2nd ed., Vol. 1, pp. 7 -55). Singapore: World Scientific.
- Consortium, W. W. (2010). W3C mobileOK Basic Tests 1.0. *www.W3C.com*, Consultado 6 Oct.2014.
- Coplien, J., & Rnvig, G. (2010). What's lean about. In *Lean architecture for agile software development*. 2nd ed., Vol. 2, Pag. 8-23. Chichester, West Sussex, UK,:Wiley.
- Dehaghani, S (2013). Which Factors Affect Software Projects Maintenance Cost More. *Acta Informatica Medica*, 1(21), 63-66.
- ESANE, (2004),“Perfil de la Industria Mexicana del Software y Servicios Relacionados” Secretaría de Economía, México,Fase 1 / Criterio 2.
- Faulkner, C. (1998). Cognitive Walk. In *The essence of human-computer interaction* (3rd ed., Vol. 1). Harlow, England: Prentice Hall.
- Foster, J. R. (1993). Cost Factors in Software Maintenance.(1ra ed., Vol. 1, Pag. 11-85), Durham: University of Durham.
- goMobi. (2014). Behind Device Atlas. California: Afiliat Technologies Ltd.*www.goMobi.com*, Consultado 6 Oct.2014.
- Gothelf, J., & Seiden, J. (2013). *Lean UX: Applying lean principles to improve user experience* (1st ed., Vol. 1, p. 152). US: O'Reilly Media.
- Harper, R., Rodden, T., Rogers, Y., & Sellen, A. (2008). *Being Human: Human-Computer Interaction in the year 2020*.(1ra ed., Vol. 1, Pag. 3-45), England, Microsoft Research Ltd.
- Hartson, H., & Pyla, P. (2012). UX tasks. In *The UX Book: Process and guidelines for ensuring a quality user experience* (2nd ed., Vol. 1, pp. 6-856). Amsterdam, Amsterdam: Elsevier.
- IBM History and Research. (2010). *Valuable resources on IBM's history*. Recuperado el 2014, de IBM.Com: <http://www-03.ibm.com/ibm/history/documents/index.html>
- IEEE. (1991). In *IEEE software engineering standards collection* (Spring 1991 ed., Vol. 15). New York, NY: Institute of Electrical and Electronics Engineers.

- Jiménez, E., & Orantes, S. (2012) Metodologías híbridas para desarrollo de software: una opción factible para México. *Revista Digital Universitaria*, 13, 3-14. Retrieved October 8, 2014, En <http://www.revista.unam.mx/vol.13/num1/art16/art16.pdf>
- Kelly, M., & Aspray, W. (1996). *Computer: a history of the information machine.*(2da ed., Vol. 1), New York: Basic
- Kowitzlawakul, Y. (2005). The Technology Acceptance Model. *CIN: Computers, Informatics, Nursing*, 411-418.
- Krug, S. (2006). *Don't make me think!: a common sense approach to Web usability* (2nd ed.). Berkeley, California: New Riders Pub.
- Lo, D. (2011). *Mining software specifications: Methodologies and applications*. Boca Raton, FL: CRC Press
- MacCormack, A. (2010). The True Costs of Software. <http://www.computerworld.com/article/2570233/enterprise-applications/the-true-costs-of-software.html> Consultado eln Octubre 8 2014
- Mason, M. K. (2013). Worldwide Business Start-Ups.En <http://www.moyak.com/papers/business-startups-entrepreneurs.html> Consultado en Octubre 10 2014
- Meerts, J. (2013). The History of Software Testing. Croacia: Testing references. *En* <http://www.testingreferences.com/testinghistory.php> Consultado el 15 de Octubre 2014.
- Morae. (2014). about Morae. San Francisco: TechSmith. www.goMobi.com, Consultado 6 Oct.2014.
- Nasa Research JSC. (2011). Software Design for Maintainability: Establish Design Criteria for Maintainable Software Products to. (1.ª Edición, Pag. 11-55). Texas, Johnson Space Center.
- Nielsen, J. (1994). *Usability inspection methods*, (7th ed., Vol. 1, Pag. 8-128). New York: Wiley.
- Nielsen, J. (2000). Heuristics. In *Designing Web usability* (2nd ed., Vol. 2, pp. 15-50). Indianapolis, Ind., Indiana: New Riders.
- Nielsen, L. (2013). How to do it. In *Personas -- user focused design* (2nd ed., Vol. 1, pp. 21-38). London: Springer.
- Nielsen, L. (2004). Personas. In *Engaging Personas and Narrative Scenarios* (2nd ed., Vol. 1, pp. 20-42). UK: Nielsen Group.
- Nielsen, G. (2011). *User Experience For Mobile Applications and Websites*(3th ed, Pag 105-208). London::NN/G.
- Nodder, C., & Nielsen, J. (2013). Challenges for UX practitioners. In *Agile development that incorporates user experience practices* (2nd ed., Vol. 1, p. 15-128). Fremont, California: Nielsen Norman Group.
- Norman, D. (1986). *User centered system design: New perspectives on human-computer interaction* (3rd ed., Vol. 1, pp. 85-345). Hillsdale, N.J.: L. Erlbaum Associates.
- Norman, D. (2010). *Living with Complexity*. (3.ª Edición, Pag. 10-48),Cambridge: MIT Press.
- Pratas, A. (2014). Patterns. In *Creating flat design websites design and develop your own flat design websites in HTML* (2nd ed., Vol. 1, pp. 15-65). Birmingham: Packt Publishing.
- Pressman, R. (1995). *Ingeniería del Software: Un enfoque práctico*, (3ª Edición, Pag. 26-30).México, MCGraw Hill.
- Pries, K., & Quigley, J. (2011). SCRUM For You. In *Scrum project management* ,(5ta ed., Vol. 1, pp. 28-58). Boca Raton, FL, CRC Press.
- SE (2004), "Estudio del nivel de madurez y capacidad de procesos de la industria de tecnologías de información en el área metropolitana de Monterrey, Nuevo León y el Distrito Federal y su área metropolitana" Secretaría de Economía del Gobierno Mexicano. Nov. 2011 [.http://www.economia.gob.mx/doc](http://www.economia.gob.mx/doc) Consultado Web. 7 Oct. 2014.
- Seffah, A. (2005). *Human-centered software engineering integrating usability in the software development lifecycle*. (5ta ed., Vol. 1) Dordrecht: Springer.
- Sklar, D., & Trachtenberg, A. (2014) "Web Programming: Cookies, Forms, Sessions, and Authentication." *PHP Cookbook*. (3rd ed. Vol. 1, pp 10-85.) Sebastopol, Calif.: O'Reilly, Print.
- Spurlock, J. (2013). Components. In *Bootstrap* (3rd ed., Vol. 1, pp. 12-86). O'Reilly Media.

- Stephenson, John. (2004) "Diseño De Interfaces." Fundamentos Del Diseño Instruccional Con E-learning. (2nd ed. Vol. 1. pp35-60), Barcelona: UOC.
- Succi, G., & Marchesi, M. (2001). About XP. In Extreme Programming examined, (1ra. edición., Vol. 3, Pag. 20-80). Boston: Addison-Wesley.
- The Standish Group International. (2013). " CHAOS MANIFESTO 2013." *CHAOS Manifesto* 5 Dec. 2013: 5,15-35. <http://www.versionone.com/>. Consultado Web. 7 Oct. 2014.
- UMBC Research. (2008). A Survey of Major Software Design Methodologies. Baltimore County, Maryland, USA: University of Maryland.En <http://userpages.umbc.edu/~khoo/survey2.html> Consultado en Octubre 6 2014.
- Weinschenk, S. (2012). Benefits of User-Centered Design. New York, USA.En <http://www.usability.gov/what-and-why/benefits-of-ucd.html>, Consultado en Octubre 2 2014.
- Young, B. (2011) "Industry Stats: Project Methodologies 2011." Planit Software Testing: Consultancy, Training,Solutions, Nov. 2011 .<http://www.planit.net.au/resource/industry-stats-project-methodologies-2011/>. Consultado Web. 7 Oct. 2014.