



# UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

---

---

FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN

ESTRATEGIA PARA EL DESARROLLO DE SISTEMAS DE  
INFORMACIÓN: CASO SECRETARÍA DE PERSONAL  
DOCENTE DE LA FACULTAD DE CONTADURÍA Y  
ADMINISTRACIÓN DE LA UNAM

TESIS PROFESIONAL

ANA LAURA LAZO OJEDA



Ciudad Universitaria, CdMx, 2016



Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

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

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



# UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

---

---

FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN

ESTRATEGIA PARA EL DESARROLLO DE SISTEMAS DE  
INFORMACIÓN: CASO SECRETARÍA DE PERSONAL  
DOCENTE DE LA FACULTAD DE CONTADURÍA Y  
ADMINISTRACIÓN DE LA UNAM

TESIS PROFESIONAL PARA OBTENER EL TÍTULO DE:  
LICENCIADO EN INFORMÁTICA

PRESENTA:

ANA LAURA LAZO OJEDA

ASESOR:

M. EN I.A. MARIA ESTHER JUDITH MARTÍNEZ GONZÁLEZ



MÉXICO, D.F. 2015

“La vida es muy sencilla: Creas algo. La mayoría de las veces fracasa, pero a veces funciona. Si tienes mucho éxito, otros lo copian y entonces haces algo distinto. El truco consisten en siempre hacer algo nuevo.”

Leonardo da Vinci

## Agradecimientos

---

A mis padres, Ana Ojeda y Felix Lazo, por darme todo lo que un hijo quiere y necesita: Amor, confianza y apoyo, así como constituir para mí el gran y mejor ejemplo de vida. Fui, soy y seré por y para ustedes, todo es por ustedes.

A mis hermanas Claudia y Mónica por ser la alegría de mi vida y hacerme saber que jamás estaré sola.

A mis tíos, primos, abuelos y cuñado, que me han dado apoyo, cariño y consejos para la vida.

A Bulmaro Lazo. Siempre tengo presente todo lo que me enseñó.

A mi asesora Esther Martínez con quien me siento profundamente agradecida por haber confiado en mí desde el principio, por haberme apoyado, animado, por toda su paciencia, por darme palabras de aliento y por todas las enseñanzas que me ha dejado desde que la conozco.

A la profesora Claudia Herrera por ser tan visionaria, por su valioso apoyo, por su tiempo y por todo el conocimiento que me ha transmitido.

A Karen Aguilar, Fabian Olguin y al equipo de desarrollo OCELIT, piezas fundamentales de esta investigación.

A todos y cada uno de los profesionales y académicos que estuvieron dispuestos a ayudarme en todo momento, a darme su opinión o aportar algo para el desarrollo de este trabajo.

A quien me acompañó en mis desvelos.

A mis amigos, con los que conviví en las aulas de la FCA, sobre todo a aquellos que estuvieron presentes durante el proceso de construcción de esta tesis: Alejandro, Itandehui, Carlos, Eduardo, Fernando, Julio y Toño. De igual manera a mis amigos de toda la vida: Erika, Nancy, Alejandro, Karina, Norma, Gaby, Mario, Yanitzi, Yunuet y Quitzé.

A la Universidad Nacional Autónoma de México y a la Facultad de Contaduría y Administración, ha sido todo un privilegio ser parte de esta institución.

Con cariño, Laura Lazo.

# Índice

<b>1. Introducción</b> .....	<b>2</b>
1.1. Problema .....	2
1.2. Pregunta de Investigación .....	4
1.3. Hipótesis.....	4
1.4. Objetivos .....	4
1.4.1. Objetivo General .....	4
1.4.2. Objetivos Específicos .....	4
1.5. Justificación .....	5
1.6. Alcance y Restricciones.....	6
1.7. Tipo de Investigación .....	6
<b>2. Metodologías Ágiles de Desarrollo de Software y Lenguaje Unificado de Modelado.</b> .....	<b>8</b>
2.1. Definición de Metodología .....	8
2.2. Agilidad.....	8
2.3. Metodologías Ágiles .....	9
2.3.1. Manifiesto Ágil .....	9
2.4. Modelos Ágiles .....	10
2.4.1. SCRUM.....	10
2.4.2. <i>Extreme Programming (XP)</i> .....	16
2.4.3. OpenUP .....	21
2.5. Conclusiones.....	25
2.6. Lenguaje Unificado de Modelado (UML).....	26
2.6.1. Definición de Modelo .....	26
2.6.2. Modelo Conceptual de UML.....	27
<b>3. Administración de Proyectos</b> .....	<b>29</b>
3.1. Definición de Administración.....	29
3.2. Definición de Proyecto .....	29
3.3. Definición de Administración de Proyectos .....	30
3.4. <i>Project Management Institute</i> ® .....	30
3.4.1. <i>Project Management Body of Knowledge</i> ® .....	32

<b>4. Proceso de Construcción de la Estrategia para el Desarrollo de Sistemas de Información .....</b>	<b>41</b>
4.1. Panorama Inicial .....	41
4.1.1. Estrategia.....	42
4.1.2. Requerimientos Generales para el desarrollo de la Estrategia.....	42
4.1.3. ¿A quién está dirigida la Estrategia de Desarrollo? .....	43
4.1.4. Proyecto SIGAD .....	43
4.2. Criterios de Selección de la base Metodológica .....	43
4.2.1. Selección de la base metodológica.....	44
4.3. Desarrollo de los Artefactos basados en el PMBOK® .....	45
4.3.1. Características Generales .....	46
4.3.2. Acta de Constitución del Proyecto.....	46
4.3.3. Interesados en el Proyecto .....	48
4.3.4. Requerimientos del Software .....	49
4.3.5. Creación de la Estructura de Desglose de Trabajo.....	54
4.3.6. Costos... ..	55
4.3.7. Recursos Humanos .....	58
4.3.8. Tiempo. ....	60
4.3.9. Riesgos. ....	61
4.3.10. Calidad .....	64
4.3.11. Comunicación .....	67
4.3.12. Acta de Cierre del Proyecto .....	69
4.3.13. Lecciones Aprendidas .....	71
4.3.14. Minutas de Reunión.....	72
4.4. Desarrollo de los Artefactos basados en Scrum.....	74
4.4.1. Lista de Producto ( <i>Product Backlog</i> ).....	74
4.4.2. Lista de pendientes del <i>Sprint</i> ( <i>Sprint Backlog</i> ) .....	74
4.5. Desarrollo de los Artefactos basados en UML. ....	75
4.5.1. Mapa Conceptual .....	75
4.5.2. Especificación de Diagramas de Casos de Uso.....	76
4.6. Desarrollo del Proceso de la Estrategia. ....	77
4.7. Logotipo. ....	81

<b>5. Estrategia para el Desarrollo de Sistemas de Información .....</b>	<b>83</b>
5.1. ¿A quién va dirigida esta Estrategia de Desarrollo?.....	83
5.2. ¿En qué proyectos puede ser aplicado? .....	83
5.3. Valores y Competencias .....	83
5.4. Roles.....	85
5.5. Sprint.....	88
5.6. Fases.....	88
5.7. Artefactos .....	92
5.7.1. Bitácora de Trabajo .....	92
5.7.2. Acta de Constitución del Proyecto.....	92
5.7.3. Identificación de Interesados .....	93
5.7.4. Plan para la Administración del Proyecto .....	94
5.7.5. Lista de Requerimientos .....	97
5.7.6. Estructura de Desglose de Trabajo .....	97
5.7.7. Diagrama de Casos de Uso/Especificación de Casos de Uso.....	98
5.7.8. Modelo Conceptual UML.....	99
5.7.9. Lista de Pendientes del <i>Sprint</i> .....	100
5.7.10. Cronograma del <i>Sprint</i> .....	101
5.7.11. Acta de Cierre del Proyecto .....	102
5.7.12. Lecciones Aprendidas .....	102
5.8. Proceso de la Estrategia de Desarrollo de Sistemas de Información. ....	104
5.9. Modelo Conceptual de UML.....	105
Software Recomendado .....	106
Bibliografía Recomendada .....	106
<b>6. Resultados y conclusiones de la aplicación de la Estrategia para el Desarrollo de Sistemas de Información.....</b>	<b>108</b>
<b>Acrónimos y convenio tipográfico .....</b>	<b>113</b>
<b>Referencias .....</b>	<b>114</b>
<b>Anexos.....</b>	<b>120</b>
Anexo 1: Acta de Constitución del Proyecto .....	120
Anexo 2: Identificación de Interesados .....	130



Anexo 3: Requerimientos del Proyecto .....	133
Anexo 4: Creación del WBS/EDT .....	139
Anexo 5: Costos del Proyecto.....	143
Anexo 6: Recursos Humanos .....	149
Anexo 7: Cronograma.....	151
Anexo 8: Riesgos .....	154
Anexo 9: Calidad.....	157
Anexo 10: Comunicación.....	164
Anexo 11: Acta de Cierre del Proyecto .....	167
Anexo 12: Lecciones Aprendidas .....	170
Anexo 13: Minutas de Reunión .....	172
Anexo 14: Lista de pendientes del <i>Sprint</i> .....	173
Anexo 15: Especificaciones de Casos de Uso.....	174
Anexo 16: Cuestionario sobre la Documentación de la Estrategia de Desarrollo de Sistemas de Información .....	176

---

# **Capítulo 1.**

## **Introducción**

---

# 1. Introducción

En el presente capítulo se aborda el problema que da origen a esta tesis, la pregunta de investigación, el alcance de la misma, los objetivos, se formula la hipótesis y se establece el tipo de investigación.

## 1.1. Problema

Actualmente, los sistemas de Información son una herramienta primordial para las organizaciones ya que facilitan el manejo de grandes cantidades de información, favoreciendo la toma de decisiones y agilizando sus procesos. Es por esa razón que en 2012, dentro de la Secretaría de Personal Docente de la Facultad de Contaduría y Administración de la UNAM se comenzó a desarrollar un software denominado Sistema Integral de Gestión Administrativa Docente (SIGAD), sistema que considera las siguientes áreas funcionales:

- Departamento de selección y desarrollo
- Departamento de control docente y Programas Institucionales
- Departamento de proyectos especiales
- Departamento de operación docente
- Departamento de evaluación docente
- Departamento de movimientos y prestaciones
- Jefatura

En el desarrollo de los primeros módulos del SIGAD participaron alumnos de los últimos semestres, que en equipo o de forma individual, comenzaron a desarrollar módulos del sistema, de tal modo que:

- Cada equipo o individuo eligió su propia metodología de desarrollo.
- Cada equipo o individuo diseñó su propia documentación para su proyecto.
- No hubo lineamientos oficiales para la codificación o el diseño del sistema que pudieran ayudar con el mantenimiento o desarrollo de otros módulos del sistema (Ejemplos: Diagramas de UML, nombre de las clases o nombres de los archivos con los códigos).

Derivados de los eventos anteriormente mencionados:

- Se generó una documentación inconsistente y de poco valor para el proyecto.
- Hubo retrasos en la entrega del producto.
- No hubo un control en los requerimientos del usuario, lo que provocó poca definición del alcance del proyecto.

Como resultado, surge la pregunta: ¿Cuál es el principal factor que genera atrasos en los proyectos? La respuesta a esa pregunta podría ser la falta de conocimientos técnicos, falta de entendimiento de los requerimientos o de los procesos de la organización, falta de comunicación entre los desarrolladores y el usuario o entre los mismos desarrolladores o falta de herramientas que los ayudaran a hacer una gestión de cada proyecto. Sin embargo, los equipos encargados del desarrollo tenían un buen conocimiento técnico, había disponibilidad por parte del usuario, pero hacían falta algunas herramientas que ayudaran a delimitar el alcance del proyecto y entonces surgió otra pregunta: ¿Existe una Estrategia de Desarrollo de Sistemas de Información dentro de la FCA que ayude a los estudiantes o pasantes interesados en titularse por Desarrollo de Sistemas?

Haciendo una revisión de la estructura de los documentos de Diseño de Sistemas que se han hecho en la Facultad de Contaduría y Administración de la UNAM, disponibles TESIUNAM– Tesis del Sistema Bibliotecario de la UNAM<sup>1</sup> se confirma que no hay una estrategia estandarizada de desarrollo y que cada proyecto tiene su propia metodología o marco de trabajo.

Se concluye que:

- No hay una metodología o estrategia definida en la Facultad de Contaduría y Administración de la UNAM, que ayude a los alumnos en el desarrollo de su proyecto de titulación.
- En la primera etapa del desarrollo del proyecto del SIGAD ocurrieron eventos que repercutieron de manera considerable en el desarrollo del mismo. Dichos eventos derivaron de una mala gestión del proyecto, dando como resultado una entrega tardía e incompleta.

---

<sup>1</sup> <http://tesis.unam.mx/>

## **1.2. Pregunta de Investigación**

La presente tesis profesional se propone dar respuesta a la siguiente pregunta de investigación: ¿Qué resultados se obtienen al diseñar y aplicar una estrategia de desarrollo de proyectos de software: Caso Secretaría de Personal Docente de la FCA de la UNAM?

## **1.3. Hipótesis**

Si se diseña y aplica una estrategia de desarrollo de proyectos de software a través de un caso práctico en la Secretaría de Personal Docente de la FCA de la UNAM, podremos identificar los elementos fundamentales que la componen, considerando el tiempo de entrega y la administración del proyecto.

## **1.4. Objetivos**

A continuación se listan los objetivos de esta investigación.

### **1.4.1. Objetivo General**

Identificar los resultados del diseño y aplicación de una propuesta de estrategia de desarrollo que implementa proyectos de software en la Secretaría de Personal Docente de la Facultad de Contaduría y Administración de la Universidad Nacional Autónoma de México, considerando el tiempo de entrega y basada en administración de proyectos.

### **1.4.2. Objetivos Específicos**

- Apoyar a estudiantes o egresados de la FCA que deseen concluir su servicio social o trámite de titulación con el desarrollo de un sistema para la Secretaría de Personal Docente, de manera que la estrategia presentada en este trabajo, actúe como una guía y además ofrezca el fortalecimiento del aprendizaje adquirido en la Licenciatura.
- Fomentar el desarrollo de sistemas para la Secretaría de Personal Docente de la FCA de la UNAM.

## 1.5. Justificación

Este trabajo de investigación propone una Estrategia de Desarrollo de Sistemas para la Secretaría de Personal Docente de la FCA-UNAM.

Esta estrategia, cuyo principio es adoptar prácticas del desarrollo ágil dejando evidencia documentada y respaldo de los procesos de creación del sistema, está basada en una selección de mejores prácticas de modelos internacionales y estándares, principalmente:

- PMBOK®
- Scrum
- UML

De tal manera, que se eligieron componentes y herramientas para la administración y modelado del sistema, haciendo una selección de los elementos de cada componente que fueran de un interés particular, para después estructurarlos y así generar la Estrategia de Desarrollo, cuyo eje central es una Estructura de Desglose de Trabajo y una serie de documentos para la administración del proyecto.

Esta estrategia se aplicará en un caso práctico, con el fin de identificar los resultados del uso las plantillas y documentos generados

La Estrategia de Desarrollo será utilizada por alumnos y egresados de la FCA-UNAM que deseen titularse por la modalidad de Diseño de un Sistema para una Organización, teniendo los siguientes beneficios:

- Usar prácticas y conceptos de desarrollo de modelos internacionales, ampliando su conocimiento y experiencia para dar pie a la obtención de certificaciones de valor internacional, particularmente PMP<sup>2</sup>.
- Tener a su disposición una serie de plantillas que podrán ser usadas para la gestión de su proyecto.
- Integrar un repositorio de información que sirva como referencia para completar las plantillas de su interés.

---

<sup>2</sup> De la cual se hablará en el capítulo 3.

## 1.6. Alcance y Restricciones.

Esta Estrategia de Desarrollo de Software se orienta a:

- La Administración de Proyectos como enfoque principal.
- Ofrecer una guía básica para el modelado de Sistemas puesto que está dirigida a alumnos que tienen conocimientos del tema.

No profundiza en:

- La programación del sistema.
- Seguridad del sistema.

Asimismo, las pruebas de esta Estrategia no contemplan la implantación del Sistema, ni describe o presenta información sensible de la organización.

## 1.7. Tipo de Investigación

En función a las definiciones de tipo de investigación de Sampieri (2003) se pretende establecer una investigación:

- De carácter exploratorio en relación a la organización y sus procesos que nos permita tener un panorama global con respecto a los problemas que se presentan, de tal manera, que se puedan establecer los elementos necesarios para cumplir los objetivos de esta investigación.
- A nivel descriptivo, se propone una estrategia que especifique: los perfiles de los actores involucrados, los requerimientos de la organización, el ciclo de vida del proyecto de desarrollo del sistema y los elementos de la aplicación de la estrategia.

---

## **Capítulo 2.**

# **Metodologías de Desarrollo De Software y Lenguaje Unificado de Modelado (UML)**

---



## 2. Metodologías Ágiles de Desarrollo de Software y Lenguaje Unificado de Modelado.

En la primera parte de este capítulo se abordará el tema de Metodologías Ágiles de Desarrollo de Software, partiendo por las definiciones de Metodología y Agilidad, lo que dará pie a la definición propia de una Metodología Ágil, así como a su contexto histórico. Además, se cita el Manifiesto Ágil, sus doce principios y se hace un resumen de Scrum, *Extreme Programming* y OpenUP, modelos ágiles que nos servirán como base para la Estrategia de Desarrollo de Sistemas de Información.

La segunda parte del capítulo está comprendida por una breve descripción del modelo conceptual UML; los bloques básicos de construcción, reglas de combinación y mecanismos comunes, cuya descripción general será la base para el desarrollo de un mapa conceptual sobre este tema con propósitos de esta estrategia.

### 2.1. Definición de Metodología

La Real Academia Española, define al término metodología como “Conjunto de métodos que se siguen en una investigación científica o en una exposición doctrinal”. De la misma manera, define a método como “Modo de decir o hacer con orden. Modo de obrar o proceder. Procedimiento que se sigue en las ciencias para hallar la verdad y enseñarla”.

INTECO (2009) define a una metodología como “Un conjunto de métodos que se utilizan en una determinada actividad con el fin de formalizarla y optimizarla. Determina los pasos a seguir y cómo realizarlos para finalizar una tarea.

Si esto se aplica a la ingeniería del software, podemos destacar que una metodología:

- Optimiza el proceso y el producto software.
- Son métodos que guían en la planificación y en el desarrollo del software
- Define qué hacer y cuándo durante todo el desarrollo y mantenimiento de un proyecto”.

### 2.2. Agilidad

Palacio (2014) hace referencia al término agilidad como una necesidad de las organizaciones de lanzar productos en un corto tiempo, entregar resultados tangibles y tener una respuesta rápida y flexible ante un mercado cambiante. Asimismo, señala que “La

gestión de proyecto ágil no se formula sobre la necesidad de anticipación, sino sobre la de adaptación continua”.

## 2.3. Metodologías Ágiles

El término Ágil nace formalmente en febrero del 2001, durante una reunión celebrada en Utah, Estados Unidos, donde un grupo de 17 profesionales de la industria del software esbozaron valores y principios que permitieran a los equipos de desarrollo desempeñar su trabajo de una manera acelerada y con gran respuesta al cambio, todo ello como alternativa a las metodologías de desarrollo tradicionales, cuyas principales características son la rigidez, resistencia al cambio y generación de gran cantidad de documentación. (Herrera, 2007).

Posteriormente, se creó *The Agile Alliance*<sup>3</sup>, una organización sin fines de lucro que promueve los principios y las prácticas del desarrollo ágil.

### 2.3.1. Manifiesto Ágil

La base de las metodologías ágiles es el Manifiesto Ágil<sup>4</sup>, en el cual se valoran:

- Individuos e interacciones sobre procesos y herramientas
- Software funcionando sobre documentación extensiva
- Colaboración con el cliente sobre negociación contractual
- Respuesta ante el cambio sobre seguir un plan

#### 2.3.1.1. Doce principios del Manifiesto Ágil

Además del Manifiesto Ágil, se establecieron los siguientes 12 principios<sup>5</sup>:

- I. Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.
- II. Aceptamos que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.

---

<sup>3</sup><http://www.agilealliance.org/>

<sup>4</sup><http://agilemanifesto.org/iso/es/>

<sup>5</sup><http://agilemanifesto.org/iso/es/principles.html>

- III. Entregamos software funcional frecuentemente, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto posible.
- IV. Los responsables de negocio y los desarrolladores trabajamos juntos de forma cotidiana durante todo el proyecto.
- V. Los proyectos se desarrollan en torno a individuos motivados. Hay que darles el entorno y el apoyo que necesitan, y confiarles la ejecución del trabajo.
- VI. El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la conversación cara a cara.
- VII. El software funcionando es la medida principal de progreso.
- VIII. Los procesos Ágiles promueven el desarrollo sostenible. Los promotores, desarrolladores y usuarios debemos ser capaces de mantener un ritmo constante de forma indefinida.
- IX. La atención continua a la excelencia técnica y al buen diseño mejora la Agilidad.
- X. La simplicidad, o el arte de maximizar la cantidad de trabajo no realizado, es esencial.
- XI. Las mejores arquitecturas, requisitos y diseños emergen de equipos auto-organizados.
- XII. A intervalos regulares, el equipo reflexiona sobre cómo ser más efectivo para, a continuación, ajustar y perfeccionar su comportamiento en consecuencia.

## 2.4. Modelos Ágiles

A continuación se resumirán tres marcos de trabajo ágiles.

### 2.4.1. SCRUM

Scrum nace formalmente en 1995, después de que uno de sus creadores, Ken Schwaber presentara "*Scrum Development Process*" en *OOPSLA Conference 95*, también fue desarrollado por Jeff Sutherland, (Palacio, 2014). Scrum es un marco de trabajo para el desarrollo y mantenimiento de productos que pudieran ser grandes. Sus principales características son, que establece un enfoque iterativo e incremental y está compuesto por roles, eventos, artefactos, reglas y equipos; todos estos componentes con propósitos específicos están encaminados al éxito en el desarrollo del producto, además está

diseñado para ser adaptativo y tener respuesta al cambio, (Schwaber, 2013).

Es llamado Scrum debido a que Jeff Sutherland tomó prestado el término "SCRUM" de una analogía que fue planteada en un estudio realizado en 1986 por Takeuchi y Nonaka, publicado en la *Harvard Business Review*, en el cual Takeuchi y Nonaka comparan equipos de alto rendimiento, multifuncionales a la formación Scrum utilizado por los equipos de rugby.<sup>6</sup>

Según Palacio (2014), Scrum es un modelo de desarrollo ágil que se caracteriza por:

“ -Adoptar una estrategia de desarrollo incremental, en lugar de la planificación y ejecución completa del producto

-Basar la calidad del resultado más en el conocimiento tácito de las personas que trabajan en equipos auto organizados, que en la calidad de la procesos empleados.

-Solapamiento de la diferentes fases del desarrollo, en lugar de realizar una tras otra en un ciclo secuencial o de cascada”.

“Las ventajas de Scrum son la capacidad de desarrollar e implementar alto valor, dar prioridad al software para aumentar rápidamente el retorno de su inversión, reducir la rotación del personal para aumentar su satisfacción y aumentar la calidad del producto” (Ken Schwaber, citado por Van Roojen, 2006).

#### 2.4.1.1. El equipo Scrum

Hay tres figuras en el equipo de Scrum:

- 1) **El Dueño de Producto (*Product Owner*)**: Es el responsable de maximizar el valor del producto y del trabajo del Equipo de Desarrollo y de gestionar la Lista del Producto (*Product Backlog*). Para que el Dueño del Producto pueda hacer su trabajo será necesario que toda la organización respete sus decisiones, las cuales se reflejan en el contenido y en la priorización de la Lista del Producto (*Product Backlog*).
- 2) **El Equipo de Desarrollo (*Development Team*)**: Formado por profesionales que desempeñan el trabajo de entrega de un Incremento del producto “Terminado”,

---

<sup>6</sup> <https://www.scrumalliance.org/why-scrum>

que potencialmente se pueda poner en producción.

Estos Equipos serán auto-organizados, multifuncionales y conformados por profesionales con las habilidades especializadas en las diferentes áreas que se requieran. Scrum sólo reconoce el título de Desarrollador independientemente, del trabajo que realicen y tampoco reconoce sub-equipos. Describe el tamaño del Equipo de Desarrollo debe de ser lo suficientemente pequeño como para permanecer ágil y lo suficiente grande para completar una gran cantidad de trabajo. Scrum sugiere un rango de entre 3 y 9 personas.

- 3) *Scrum Master*:** Es el responsable de que Scrum sea entendido y adoptado, es un líder que está al servicio del Equipo de Scrum, del Dueño de Producto y de la organización.

El *Scrum Master* colabora con el Dueño de Producto para encontrar técnicas que ayuden a gestionar y ordenar la Lista de Producto de manera efectiva, entender la planificación del producto, la agilidad y cómo practicarla, así como facilitar los eventos de Scrum según se necesiten.

Con el Equipo de Desarrollo, el *Scrum Master* se encarga de guiar al equipo para que sea auto-organizado y multifuncional, para crear productos de alto valor y estimular el progreso.

Guía y lidera a la Organización para la adopción de Scrum, planifica sus implementaciones y ayuda a todos los empleados e interesados a entender el marco del trabajo.

#### 2.4.1.2. Eventos de Scrum

Los eventos de Scrum son reuniones regulares o bloques de tiempo con una determinada duración.

##### **El *Sprint***

“El corazón de Scrum es el *Sprint*, es un bloque de tiempo (*time-box*) de un mes o menos durante el cual se crea un incremento de producto *terminado* utilizable y potencialmente desplegable. Es más conveniente si la duración de los *Sprints* es consistente a lo largo del esfuerzo de desarrollo. Cada nuevo *Sprint* comienza inmediatamente después de la

finalización del *Sprint* previo". (Schwaber et al, 2013).

Los *Sprints* contienen y consisten de:

- **Planificación del *Sprint* (*Sprint Planning Meeting*):**

Es una reunión de trabajo donde el Equipo Scrum decide qué se va a realizar durante el *Sprint* y se responden a dos preguntas primordiales:

- 1) ¿Qué puede entregarse en el Incremento resultante del *Sprint* que comienza?

Se trabaja para proyectar la funcionalidad que se desarrollará durante el *Sprint*.

Participan el Equipo de Desarrollo y el Dueño del Producto, se ocupa la Lista del Producto, el último Incremento del producto, la capacidad proyectada del Equipo de Desarrollo para el *Sprint* y el rendimiento que el equipo haya tenido en el *Sprint* anterior. Del análisis y las decisiones de esta reunión, se elabora un Objetivo del *Sprint* (*Sprint Goal*) y se eligen elementos de la Lista del Producto cuyo desarrollo es el objetivo principal del *Sprint*.

- 2) ¿Cómo se conseguirá hacer el trabajo necesario para entregar el incremento?

Una vez que se tiene el Objetivo del *Sprint*, el Equipo de Desarrollo tiene que planificar cómo es que se desarrollará la funcionalidad que formará parte del incremento del producto "Terminado".

De los elementos que han sido elegidos para ser desarrollados en ese *Sprint*, más el plan de desarrollo, deriva la Lista de Pendientes del *Sprint* (*Sprint Backlog*).

Durante todo el proceso participa el Dueño del Producto y al final se le es notificado al *Scrum Master* sobre la forma de trabajo para lograr el objetivo.

- **Scrums Diarios (*Daily Scrums*):**

Son reuniones de no más de 15 minutos en donde el Equipo de Desarrollo sincroniza sus actividades y crea planes para las siguientes 24 horas. Constituye un medio para mejorar la comunicación, evitar otras reuniones, identificar y eliminar impedimentos en el desarrollo y promueve la toma rápida de decisiones.

Durante la reunión, cada miembro responde:

- 1) ¿Qué hice ayer que ayudó al Equipo de Desarrollo a lograr el Objetivo del *Sprint*?
- 2) ¿Qué haré hoy para ayudar al Equipo de Desarrollo a lograr el objetivo del *Sprint*?
- 3) ¿Veo algún inconveniente que evite que el Equipo de Desarrollo o yo logremos el Objetivo del *Sprint*?

El *Scrum Master* es el encargado de asegurarse que las reuniones ocurran, sin embargo, el Equipo de Desarrollo es el que organiza y realiza dichas reuniones.

- **Revisión del *Sprint* (*Sprint Review*):**

Se lleva a cabo al final de cada *Sprint* para inspeccionar el incremento y si se requiere, hacer cambios a la Lista del Producto.

Participan el Equipo Scrum, el Dueño del Producto y los interesados clave que él considere.

La idea general es que se muestre el avance y el estado actual del Producto, así como cuestiones de tiempo y presupuesto con el fin de tener un entendimiento de todas las partes, mantener informado al Dueño del Producto y hacer ajustes.

- **Retrospectiva del *Sprint* (*Sprint Retrospective*):**

Al final, cada *Sprint* es inspeccionado para crear planes de mejora para el siguiente *Sprint*.

En la reunión se inspeccionan las relaciones, procesos, herramientas y a las personas que participaron en el *Sprint*. Se identifican los elementos que necesitan mejoras para lo cual, se desarrolla un plan para que en el siguiente *Sprint* sean implementadas dichas mejoras.

### 2.4.1.3. Artefactos de SCRUM

Son útiles para proporcionar transparencia de la información clave y facilitar la inspección y adaptación durante todo el proceso.

#### **2.4.1.3.1. Lista de Producto (*Product Backlog*)**

Contiene todo lo necesario para completar el producto, es la fuente de requisitos para cualquier cambio a realizarse en el producto.

El contenido, disponibilidad y ordenación es responsabilidad del Dueño del Producto.

Esta Lista está en constante evolución y cambia en relación al avance y necesidades del producto.

Enumera características, funcionalidades, requisitos, mejoras y correcciones del producto.

El seguimiento del progreso hacia un objetivo, es la evaluación del avance para la terminación de trabajo, la Lista del Producto es la base de esta evaluación y en ella se identifican el Trabajo Consumido (*Burndown*), avanzado (*Burnup*) y el Flujo Acumulado (*Cumulative flow*).

#### **2.4.1.3.2. Lista de Pendientes del Sprint (*Sprint Backlog*)**

Esta lista la conforman todos aquellos elementos que han sido elegidos de la Lista del Producto para ser llevados a cabo para el *Sprint* con la finalidad de completar una funcionalidad que haya sido elegida para completar en ese *Sprint*. También está conformada por un plan para la entrega del Incremento del Producto.

#### **2.4.1.4. Definición de “Terminado”**

Cada Equipo debe de tener claro lo que significa para ellos “Terminado” y deben darle un significado a manera que el término sea transparente y entendido por todos los miembros del Equipo. “Terminado” puede significar que se ha completado una funcionalidad y que esta ha sido integrada, probada y se ha asegurado que funcione de manera correcta.



### 2.4.1.5. Gráfico del Proceso de Scrum



Gráfico 2.1. Scrum (Deemer, 2009).

### 2.4.2. Extreme Programming (XP)

*Extreme Programming* es una metodología ágil de desarrollo de software que hace énfasis en la adaptabilidad, además potencializa las relaciones interpersonales promoviendo el trabajo en equipo, creando un ambiente cordial y preocupándose por el aprendizaje de los desarrolladores.

En palabras de uno de sus creadores, Kent Beck (1999), “XP es una forma de desarrollar software liviana, eficiente, de bajo riesgo, flexible, predecible, científica y divertida”.

*Extreme Programming* se fundamenta en 5 principios básicos:

#### I. Comunicación

Escrita; documentando el código, las pruebas unitarias, los objetivos de las clases y la funcionalidad de métodos.

Oral; debe ser constante, fluida entre programadores y con el cliente ya que éste es parte esencial del proyecto y debe tener disponibilidad para solucionar dudas.

## **II. Simplicidad**

Debe encontrarse en la documentación que se genere durante todo el proyecto, así como en el diseño para desarrollar más rápido y para que el sistema tenga un mantenimiento fácil.

## **III. Retroalimentación**

Con el cliente se logrará una retroalimentación eficiente a través de ciclos de entrega cortos y demostraciones de alguna funcionalidad, de esta manera se asegura que se esté desarrollando lo que el cliente requirió.

## **IV. Coraje**

Para mantener la simplicidad, para comunicarse con los demás, para recibir retroalimentación durante el desarrollo, no perder de vista lo que cliente necesita y para perder los miedos.

## **V. Respeto**

El respeto debe de existir entre los integrantes del equipo de desarrollo y para el trabajo que realizan.

### **2.4.2.1 Prácticas de XP**

A continuación, un resumen sobre las descripciones de las Prácticas de XP que hace Letelier (2006):

#### **➤ El juego de la planificación**

Cuya base es la comunicación entre el cliente y los programadores, con el fin de determinar qué va a resolver el software, qué es lo que tiene que estar hecho primero y hacer estimaciones en el tiempo de entrega.

#### **➤ Entregas pequeñas**

Desarrollar en un plazo no mayor a 3 meses alguna funcionalidad del sistema de manera que represente un resultado con valor para el negocio.

#### **➤ Metáfora**

Se usa una metáfora para describir la función del sistema, esta metáfora debe ser entendida por el cliente y el equipo de desarrollo y el objetivo principal es facilitar el entendimiento entre ambas partes acerca del funcionamiento del sistema.

➤ **Diseño simple**

El diseño del sistema debe ser lo más simple posible, siempre tomando en cuenta las necesidades del cliente.

➤ **Pruebas**

Se hacen pruebas unitarias que son establecidas por el cliente antes de la codificación y se ejecutan en cada modificación que se hace al sistema.

➤ **Refactorización**

Es el proceso de mejorar el diseño del código, hacerlo más claro, simple, de mejor calidad y listo para cualquier cambio que venga sin afectar su comportamiento externo, (Wake, 2000).

➤ **Programación en parejas**

Los programadores deben trabajar en parejas para la producción de código, de esta manera se minimizan los errores se mejora el diseño.

➤ **Propiedad colectiva del código**

Todos los programadores pueden modificar el código en cualquier momento.

➤ **Integración continúa**

Una vez que un programador concluya una pieza de código, debe integrarla al sistema, esto debe ocurrir con frecuencia, de manera que haya múltiples integraciones por día.

➤ **40 horas por semana**

Lo máximo que el equipo de desarrollo debe trabajar son 40 hrs., con el fin de evitar que los programadores se fatiguen, pierdan la motivación y comentan errores por no estar lo suficientemente concentrados.

➤ **Cliente in-situ**

La presencia del cliente es imprescindible, éste tiene que estar disponible el tiempo que el equipo lo requiera.

➤ **Estándares de programación**

Los estándares en el código son importantes ya que representan un entendimiento común entre todos los programadores.

## 2.4.2.2. Roles

*Extreme Programming* tiene 7 roles, a continuación su descripción (Letelier, 2006):

### ▶ Programador

Es la persona encargada de escribir el código y ejecutar las pruebas unitarias.

### ▶ Cliente

Sus funciones son: desarrollar las historias de usuario, validar el trabajo de los programadores a través de pruebas funcionales y asignar prioridades.

### ▶ Encargado de pruebas (*Tester*)

Ejecutar pruebas de forma regular y además ayuda al Cliente con sus pruebas funcionales.

### ▶ Encargado de seguimiento (*Tracker*)

Proporciona una retroalimentación acerca del grado de avance con respecto a la estimación inicial, además da seguimiento al Equipo acerca del proceso de XP.

### ▶ Entrenador (*Coach*)

Conocedor del proceso XP, será el encargado proveer la pauta para que el proceso XP sea llevado a cabo de forma correcta por todos los miembros del equipo.

### ▶ Consultor

Es un miembro externo cuya función es ayudar al equipo en alguna área en específico.

### ▶ Gestor (*Big boss*)

Su función principal es coordinar y establecer un vínculo entre el cliente y los programadores.

## 2.4.2.3. Historias de Usuario

El concepto de Historias de Usuario, según Letelier (2006) se refiere a una técnica para recabar requerimientos del software, de manera que el cliente describa cómo debe ser el sistema, qué características debe tener y qué es lo que debe de hacer, todo esto usando tarjetas de papel.

Las Historias de Usuario pueden ser sustituidas, eliminadas o modificadas hasta que los programadores las comprendan y puedan implementar su contenido en pocas semanas.

En cuanto a su contenido y estructura, no existe una regla a seguir, aunque algunos autores sugieren que las Historias deben contener información como fecha, tipo de actividad, número de Historia, prioridad, comentarios, etc.

#### **2.4.2.4. Proceso de Desarrollo**

Letelier (2006) explica que el proceso de XP consiste en 6 fases:

##### **I. Exploración**

El cliente desarrolla su Historia de Usuario, con el fin de dar a conocer sus requerimientos a grandes rasgos, de esta forma los desarrolladores comienzan a familiarizarse no sólo con los requerimientos y las necesidades del cliente, también con la tecnología que usarán.

##### **II. Planificación de la Entrega (*Release*)**

Una vez que el cliente haya realizado su Historia de usuario, empieza por definir prioridades, lo que da cabida a que los programadores realicen una estimación del esfuerzo necesario y así puedan dar un aproximado de las fechas de entrega.

##### **III. Iteraciones**

Se refiere a las iteraciones sobre el sistema y se relaciona con la cuestión técnica y el trabajo que debe realizarse para que el sistema entre a producción. Para hacer un plan de iteración, se debe tomar en cuenta las historias de usuario que aún no han sido atendidas, el avance en el proyecto, las pruebas de aceptación y las tareas no concluidas de la iteración anterior.

##### **IV. Producción**

Probar y revisar el rendimiento del sistema constituyen las actividades más importantes que se deben llevar a cabo antes de presentar el sistema al entorno del cliente.

##### **V. Mantenimiento**

Una vez que se haya puesto en producción la primera versión del sistema, se deben de llevar a cabo nuevas iteraciones y mantener el sistema funcionando.

##### **VI. Muerte del Proyecto**

Cuando ya no hay Historias de Usuario por completar significa que se han cubierto las necesidades del cliente, lo que genera la muerte del proyecto, que también puede ocurrir cuando el cliente ya no tiene el presupuesto para mantener el proyecto o ha cesado la necesidad que diera origen al mismo.

### 2.4.2.5. Gráfico del Proceso de XP.

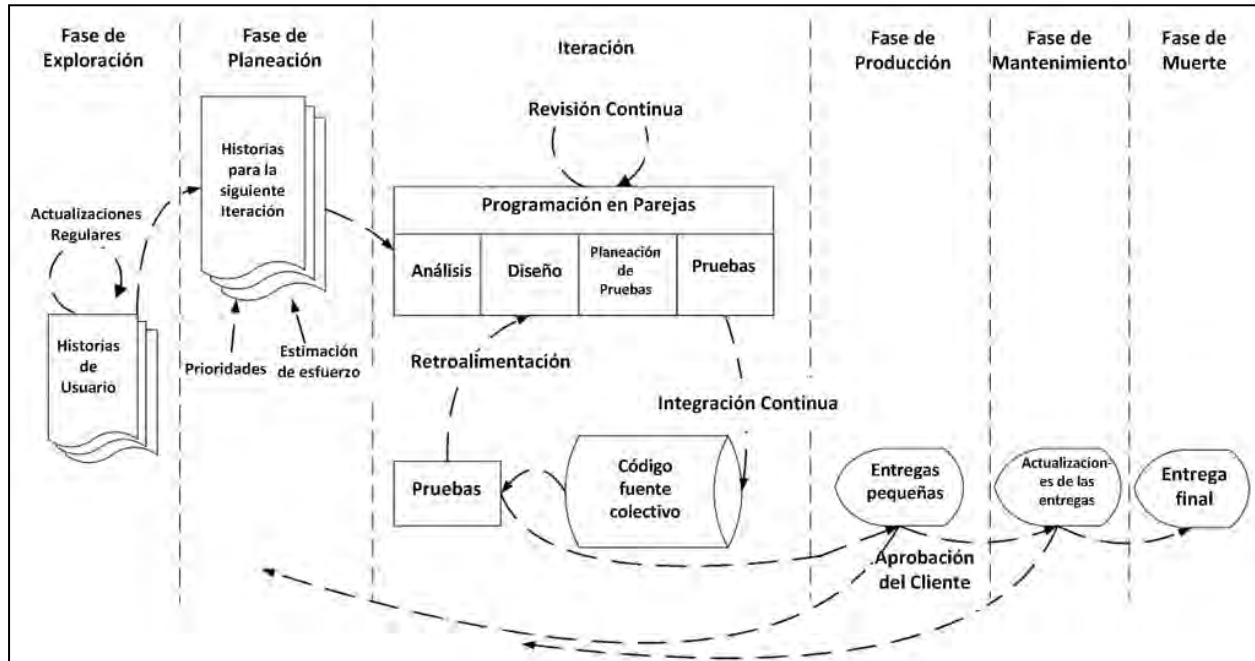


Gráfico 2.2, Ciclo de vida del proceso de XP (Traducción: Lazo, 2014). (Pekka, 2002)

### 2.4.3. OpenUP

OpenUP es un proceso ágil de desarrollo de software que se basa en RUP (*Rational Unified Process*) la cual contiene prácticas enfocadas a ayudar al equipo de desarrollo a generar un producto de calidad y de forma eficaz. Fue desarrollada por un grupo de empresas de la industria de TI, entre ellas *IBM Corporation*, quienes donaron la metodología a la fundación Eclipse en 2007. (Ríos, 2007).

“OpenUP es un procesos unificado que aplica un enfoque iterativo incremental dentro de un ciclo de vida estructurado. Abraza una filosofía pragmática, ágil que se centra en la naturaleza colaborativa del desarrollo del software”<sup>7</sup>

<sup>7</sup> <http://epf.eclipse.org/wikis/openup/>

### 2.4.3.1. Principios

OpenUP tiene 4 principios basados en el manifiesto ágil:

- I. Colaborar para alinear los intereses y compartir el entendimiento.
- II. Equilibrar las prioridades para maximizar el valor de los interesados en el proyecto.
- III. Construir una arquitectura al inicio del proceso para minimizar los riesgos y planificar el desarrollo.
- IV. Evolucionar para obtener retroalimentación y mejorar continuamente.

### 2.4.3.2. Roles

OpenUP tiene tres conjuntos de roles:

#### i) Básicos

##### 1) Analista (*Analyst*)

Representa a los clientes y usuarios recopilando sus preocupaciones, aportaciones y requerimientos para entender el problema a resolver.

##### 2) Arquitecto (*Architect*)

Es el responsable de definir la arquitectura del software y tomar decisiones acerca del diseño y la implementación del sistema.

##### 3) Desarrollador (*Developer*)

Es el responsable de la construcción del sistema, así como el diseño, prototipos de interfaz, pruebas unitarias e integración de los componentes.

##### 4) Administrador de Proyecto (*Project Manager*)

Es el encargado de la planificación del proyecto, de coordinar las interacciones entre los interesados y mantener al equipo enfocado en el cumplimiento de los objetivos.

##### 5) Interesado (*Stakeholder*)

Toda persona o grupo que sea afectada por el resultado del proyecto.

##### 6) Tester

Es el responsable de las actividades de identificación, definición, implementación, registro de resultados y realización de las pruebas.

##### 7) Cualquier rol (*Any Role*)

Cualquier miembro del equipo que realice tareas generales.

## ii) Implementación (Deployment)

### 1) Desarrollador del curso (*Course Developer*)

Encargado de la creación de materiales para la capacitación a los usuarios finales.

### 2) Ingeniero de Implementación (*Deployment Engineer*)

Es el responsable de la implementación en el entorno de producción.

### 3) Gestor de implementación (*Deployment Manager*)

Responsable de la gestión de las liberaciones de uno o más programas o líneas de productos.

### 4) Dueño del Producto (*Product Owner*)

Representa las necesidades del usuario final y define el trabajo en el proyecto.

### 5) Escritor Técnico (*Technical Write*)

Ayuda a los miembros del equipo a completar la documentación necesaria para el Propietario del producto, los Usuarios finales y personal de apoyo.

### 6) Entrenador (*Trainer*)

Ofrece formación a los Usuarios finales, personal de apoyo o a los responsables del mantenimiento del sistema.

## iii) Ambiente (*Environment*)

### 1) Ingeniero del procesos (*Process Engineer*)

Provee procesos de desarrollo adecuados al equipo del proyecto para evitar obstáculos en su trabajo.

### 2) Especialista de alguna herramienta (*Tool Specialist*)

Brinda asistencia técnica al proyecto.

## 2.4.3.3. Ciclo de vida OpenUP

Todo proyecto que se realiza bajo OpenUP se componen de 4 fases y cada una de ella se divide en iteraciones, como se muestra en el Gráfico 3.3.



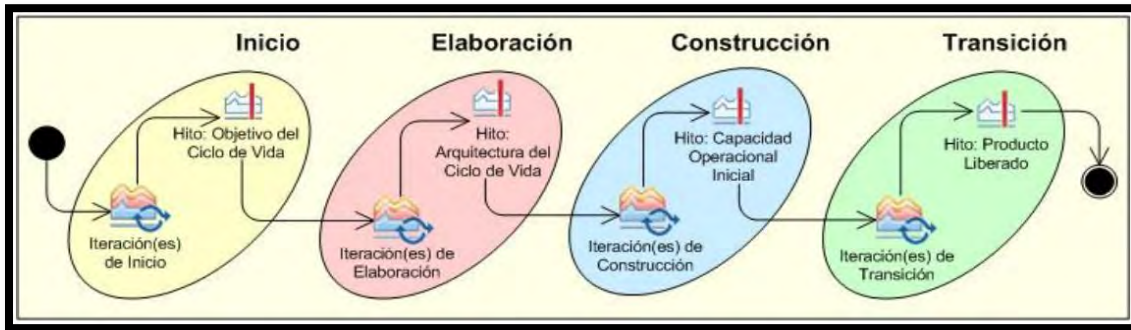


Gráfico 3.3. *Ciclo de vida de OpenUP.* (Ríos, 2013)

A continuación se describe brevemente cada una de las fases del ciclo de vida de OpenUP (Ríos, 2013):

**1. Inicio** (*Inception Phase*)

En esta fase se toman en consideración las necesidades de todos los participantes del proyecto y se define el objetivo del proyecto, así como el ámbito, los límites, los criterios de aceptación, el costo, un boceto de la planificación y los casos de uso.

**2. Elaboración** (*Elaboration Phase*)

Durante esta fase se hace el análisis y se define la arquitectura del sistema, procesos de desarrollo, herramientas, infraestructura y entorno de desarrollo, además se elabora un plan de proyecto. Esta fase dará como resultado una definición clara de los Casos de Uso, arquitectura del sistema y un prototipo de esta.

**3. Construcción** (*Construction Phase*)

Es en esta fase, cuando son desarrollados, probados e integrados los componentes del sistema. Cada componente debe ser desarrollado con rapidez y calidad para obtener incrementos ejecutables.

**4. Transición** (*Transition Phase*)

En esta fase se hace lo referente a la introducción del sistema en el ambiente final, comprende la capacitación de los usuarios finales, pruebas de versiones beta y piloto.

## 2.5. Conclusiones

Con lo visto anteriormente, podemos concluir que:

- Scrum es un marco de trabajo con enfoque iterativo e incremental. Sus roles son *Scrum Master*, *Product Owner* y *Development Team*, los equipos de trabajo son auto-organizados y multifuncionales.

El concepto más importante de Scrum es el *Sprint*, el cual es un bloque de tiempo en el que se realiza algún incremento del producto.

En Scrum también existen eventos como el *Daily Scrum*, *Sprint Review* y *Sprint Retrospective* y además se hacen uso de artefactos tales como *Product Backlog* y el *Sprint Backlog*.

- XP es una metodología que hace énfasis en las relaciones interpersonales, en el trabajo en equipo y en crear un buen ambiente de trabajo, para ello tiene principios que deben ser seguidos por todos aquellos que trabajen bajo este enfoque, estos principios son la comunicación, simplicidad, retroalimentación, coraje y respeto. Adicionalmente XP se basa en 12 practicas aplicables a todo el proceso de desarrollo e identifica 7 roles; programador, cliente, encargado de pruebas, encargado de seguimiento, entrenador, consultor y gestor.

Su proceso consta de 6 fases las cuales son: exploración, planificación, iteraciones, producción, mantenimiento y muerte del proyecto.

El concepto clave de XP son las Historias de Usuario, la cual es una técnica para recabar requerimientos de software.

- OpenUP aplica un enfoque iterativo incremental en un ciclo de vida estructurado y tiene cuatro principios derivados del manifiesto ágil. Hay tres grupos de roles: los básicos, de implementación y de ambiente. Sus fases son: Inicio, elaboración, construcción y transición.

Cada metodología tiene estructuras diferentes, sin embargo, todas ofrecen elementos útiles e interesantes que podemos adaptar a nuestras necesidades y podemos usar a nuestro favor.

## 2.6. Lenguaje Unificado de Modelado (UML)

UML es un lenguaje de modelado gráfico que se utiliza para visualizar, constituir, documentar, y especificar los artefactos de un sistema software, representando los planos de un sistema, elementos conceptuales, procesos de negocio, funciones del sistema o elementos concretos como las clases en un lenguaje de programación, una base de datos o componentes software utilizables (Booch et al, 2006).

El Lenguaje Unificado de Modelado se comenzó a desarrollar en octubre de 1994 cuando Grady Booch y James Rumbaugh tenían como objetivo inicial la unificación de los métodos de Booch y OMT, posteriormente se unió Ivar Jacobson. Durante el año de 1996 recibieron retroalimentación de la comunidad internacional relacionada con la Ingeniería de software, de esta manera organizaciones como *Digital Equipment Corporation*, Hewlett-Packard, IBM, Microsoft, Oracle, entre otras, colaboraron con la producción de UML 1.0,

En 1997 se unieron más organizaciones y se formó un grupo de trabajo para la semántica. Ese mismo año, UML 1.1 fue aceptada por la *Object Management Group*, desde entonces su mantenimiento fue asumido por la *OMG Revision Task Force* quien creó las versiones 1.3, 1.4 y 1.5, apareciendo la versión 2.0 en el año 2005.

### 2.6.1. Definición de Modelo

Un modelo es una simplificación de la realidad, proporciona los planos de un sistema, pueden involucrar planos detallados o más generales que ofrecen una visión global del sistema en consideración.

Se construyen modelos porque a través de ellos se puede comprender mejor el sistema que se está desarrollando y se consiguen 4 objetivos:

- I. Visualizar cómo es o cómo queremos que sea un sistema.
- II. Especificar la estructura o el comportamiento de un sistema.
- III. Nos proporcionan plantillas que nos guían en la construcción de un sistema.
- IV. Documentar las decisiones que se han adoptado.

## 2.6.2. Modelo Conceptual de UML

Para comprender UML, es necesario aprender tres elementos principales:

### 1. Tres tipos de Bloques Básicos de construcción

#### 1.1. Elementos

“Los elementos son abstracciones que representan ciudadanos de primera clase en un modelo.” (Booch et al., 2006).

Hay elementos estructurales, de comportamiento, de agrupación y de anotación.

#### 1.2. Relaciones

Para ligar elementos, se necesitan relaciones, las cuales hay de cuatro tipos; dependencia, asociación, generalización y realización.

#### 1.3. Diagramas

“Un diagrama es la representación gráfica de un conjunto de elementos, visualizando la mayoría de las veces con un grafo conexo de nodos (elementos) y arcos (relaciones).” (Booch et al., 2006).

UML tiene 13 tipos de diagramas; de clases, objetos, componentes, estructura compuesta, casos de uso, secuencia, comunicación, estados, actividades, despliegue, paquetes, tiempos y de visión global de interacciones.

### 2. Reglas de combinación

UML tiene reglas para construir un modelo bien formado, el cual deberá de estar semánticamente autoconsistente y en armonía como los modelos con los que se relacione.

En UML hay reglas sintácticas y semánticas para nombres, alcance, visibilidad, integridad y ejecución.

### 3. Mecanismos comunes

UML tiene mecanismos que son aplicados en todo el lenguaje de forma consistente, con el fin de hacer la construcción de los diagramas más simple y armoniosa al ajustarse a un patrón de características comunes.

---

## **Capítulo 3.**

# **Administración de Proyectos**

---

### 3. Administración de Proyectos

La Administración de Proyectos es el proceso de planear, organizar, dirigir y controlar actividades, recursos y personas para cumplir algún objetivo. Es el tema que se aborda en el presente capítulo, en donde se comienza por definir de manera separada los conceptos de Administración y de Proyecto, lo que nos va conducir a la definición de Administración de Proyectos según algunas de las organizaciones más reconocidas internacionalmente en este ámbito, como el *Project Management Institute*<sup>8</sup> de la cual dedicamos un apartado en este capítulo. Posteriormente, incluimos un resumen del contenido del *Project Management Body of Knowledge*<sup>9</sup> quinta versión, estándar y guía de la profesión de la Administración de Proyectos, desarrollado por el PMI<sup>®</sup>.

#### 3.1. Definición de Administración

Münch (2010) define a la administración como “Un proceso a través del cual se coordinan y optimizan los recursos de un grupo social con el fin de lograr la máxima eficiencia, calidad, productividad y competitividad en la consecución de sus objetivos”.

De la misma forma, Cleland, D. y King, W. (1983) la definen como “La planeación, organización y control de los recursos para lograr un objetivo a corto plazo”.

Ambas definiciones coinciden en que existen recursos humanos y materiales para cumplir un objetivo, pero no basta con cumplirlo, se necesitan usar los recursos de manera eficiente, logrando la máxima calidad y productividad.

Se puede concluir que la administración es un proceso de planeación, control y optimización de recursos humanos y materiales para lograr un objetivo.

#### 3.2. Definición de Proyecto

Según el *Project Management Institute*<sup>9</sup> “Un proyecto es un esfuerzo temporal que se lleva a cabo para crear un producto, servicio o resultado único” por lo tanto, un proyecto tiene un principio y un fin definido y éste llega cuando se alcanzaron los objetivos, no pueden ser alcanzados o ya no exista la necesidad del proyecto. Puede involucrar una

---

<sup>8</sup> <http://www.pmi.org/About-Us.aspx>

<sup>9</sup> <http://www.pmi.org/>

persona hasta una organización y se pretende que el servicio, producto o resultado único sea duradero con impacto social, económico o ambiental.

Prince2<sup>10</sup> (*Project In Controlled Environments*) es un método basado en procesos para la gestión eficaz de proyectos que es ampliamente utilizado por el Gobierno del Reino Unido, define un proyecto como “Una organización temporal que se crea con el propósito de la entrega de uno o más productos de negocio de acuerdo a un caso de negocio”.<sup>11</sup>

Por lo tanto, se podría concluir que un proyecto nace a raíz de una necesidad y para cubrirla se ejecutan una serie de actividades cuyo objetivo es el desarrollo de un producto, servicio o resultado único en un tiempo determinado.

### 3.3. Definición de Administración de Proyectos

El *Project Management Institute*<sup>®</sup> define a la Administración de Proyectos como “La aplicación de conocimientos, habilidades, herramientas y técnicas a las actividades del proyecto para cumplir con los requisitos del mismo” (PMBOK<sup>®</sup>, 2013).

Si se toman en cuenta las definiciones anteriormente citadas, entonces se sostiene que la Administración de Proyectos es la planeación, control y optimización de recursos para generar un producto, servicio o resultado único en un tiempo determinado y es una competencia estratégica para las organizaciones que les permite vincular los resultados del proyecto a los objetivos del negocio y tener mayor competitividad.

En función de lo antes mencionado, se puede decir que la Administración de Proyectos consiste en usar técnicas, herramientas, habilidades y conocimientos para planificar, coordinar y dar seguimiento a todas aquellas actividades, procesos y recursos destinados a cubrir el objetivo del proyecto.

### 3.4. *Project Management Institute*<sup>®</sup>

Fundado en 1969, el *Project Management Institute*<sup>®</sup> (PMI<sup>®</sup>) es la asociación sin fines de lucro más grande en el mundo dedicada a desarrollar la Administración de Proyectos. Para

---

<sup>10</sup> <http://www.prince2.com/rest-world/training>

<sup>11</sup> <http://www.prince2.com/what-is-prince2>

2013 tenía más de 700,000 miembros, profesionales certificados y voluntarios en al menos 270 Capítulos<sup>12</sup> en alrededor de 125 países.<sup>13</sup>

Ofrece certificaciones que reconocen los conocimientos y competencias en Administración de Proyectos, las cuales son:

- *Certified Associate in Project Management (CAPM)*<sup>®</sup>
- *Project Management Professional (PMP)*<sup>®</sup>
- *Program Management Professional (PgMP)*<sup>®</sup>
- *Portfolio Management Professional (PfMP)*<sup>®</sup>
- *PMI Agile Certified Practitioner (PMI-ACP)*<sup>®</sup>
- *PMI Risk Management Professional (PMI-RMP)*<sup>®</sup>
- *PMI Scheduling Professional (PMI-SP)*<sup>®</sup>
- *OPM3*<sup>®</sup> *Professional Certification*
- *PMI Professional in Business Analysis (PMI-PBA)*<sup>SM14</sup>

Asimismo ha publicado:<sup>15</sup>

- Estándares y guías:
  - *A Guide to the Project Management Body of Knowledge (PMBOK*<sup>®</sup> *Guide)*
  - *The Standard for Program Management*
  - *The Standard for Portfolio Management*
  - *Organizational Project Management Maturity Model (OPM3*<sup>®</sup>)
- Prácticas:
  - *Practice Standard for Project Risk Management*
  - *Practice Standard for Earned Value Management*
  - *Practice Standard for Project Configuration Management*
  - *Practice Standard for Work Breakdown Structures*
  - *Practice Standard for Scheduling*

---

<sup>12</sup> Un Capítulo es una asociación por ubicación geográfica de profesionales certificados.

<sup>13</sup> PMI-Find out about us. [https://www.youtube.com/watch?v=LQIIQk-D\\_I0&feature=youtu.be](https://www.youtube.com/watch?v=LQIIQk-D_I0&feature=youtu.be)

<sup>14</sup> El examen para obtener esta certificación se encuentra en periodo piloto de mayo a agosto del 2014.

<http://www.pmi.org/Certification/PMI-Professional-in-Business-Analysis-PMI-PBA.aspx>

<sup>15</sup> Library of PMI Global Standards <http://www.pmi.org/PMBOK-Guide-and-Standards/Standards-Library-of-PMI-Global-Standards.aspx>



- *Practice Standard for Project Estimating*
- *Practice Standard Competency Development Framework*
- Extensiones de los estándares del PMI®:
  - *Software Extension to the PMBOK® Guide.*
  - *Construction Extension to the PMBOK®*
  - *Government Extension to the PMBOK®*
- Guías Prácticas:
  - *Implementing Organizational Project Management: A Practices Guide*
  - *Navigating Complexity: A Practices Guide*
  - *Managing Change in Organizations: A Practices Guide*

Ha publicado el *Code of Ethics and Professional Conduct* del *Project Management Institute®*, donde se precisan expectativas y obligaciones de los profesionales de la dirección de proyectos.

Además tiene un programa de Proveedores de Educación Registrados (REP), actividades de investigación de la profesión, organiza congresos y simposios y tiene publicaciones como *Project Management Journal®*, *PM Network®* y *PM Today®*

### **3.4.1. *Project Management Body of Knowledge®***

La Guía del PMBOK® quinta edición se define como un estándar y guía para la profesión de dirección de proyectos, por lo tanto es un documento formal que describe normas, métodos, procesos y prácticas establecidas para dicha profesión.

Proporciona pautas y define a la dirección de proyectos y otros conceptos relacionados, también describe su ciclo de vida y los procesos conexos.

Se compone por 13 capítulos. Los primeros dos capítulos son la introducción a los conceptos claves de la administración de proyectos, el tercero presenta la norma y un resumen de los procesos considerados como buenas prácticas, del cuarto al decimo tercero es la estructura del la Guía del PMBOK®.

Los propósitos de la Guía del PMBOK® son:

- Identificar conocimientos, procesos, habilidades, herramientas y técnicas fundamentales de la dirección de proyectos reconocidos como buenas prácticas.
- Proporcionar y promover un vocabulario común en el ámbito de la dirección de proyectos.

### 3.4.1.1. Grupos de Procesos

La Guía del PMBOK® define a un proceso como “Un conjunto de acciones y actividades interrelacionadas realizadas para obtener un producto, resultados o servicio predefinido. Cada proceso se caracteriza por sus entradas, por las herramientas y técnicas que puedan aplicarse y por las salidas que se obtienen”.

Describe 5 grupos de procesos: inicio, planeación, ejecución, seguimiento y control y cierre.

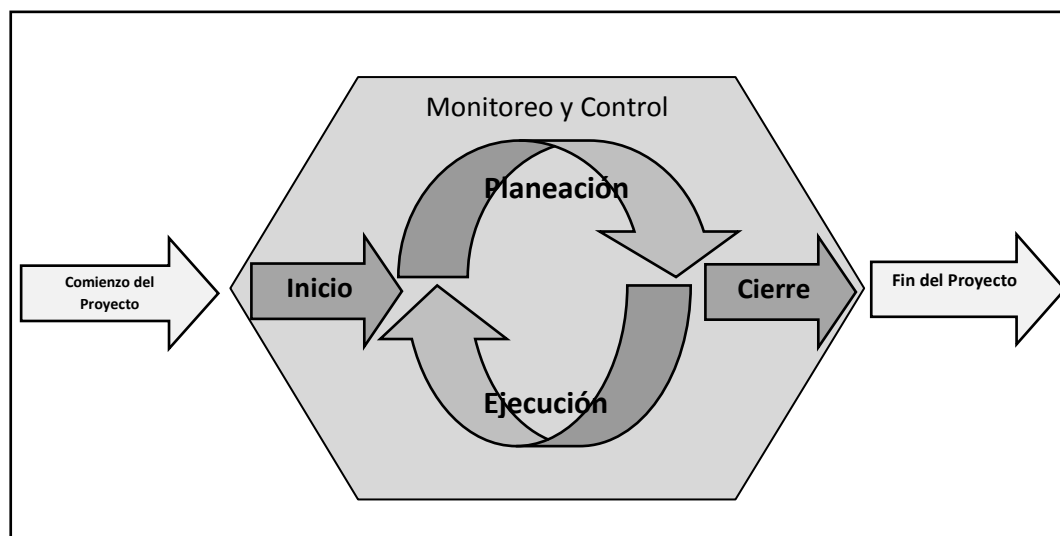


Gráfico 3.1. Grupos de Procesos de la Dirección de Proyectos. PMBOK® (2013)

A continuación se describe brevemente cada Grupo de Procesos:

#### ► Inicio

Este grupo de procesos se enfoca a la definición de un nuevo proyecto o una nueva fase de algún proyecto. Para ello identifica a todas las partes interesadas en el proyecto, tanto internas como externas así como su influencia en el proyecto. Además es desarrollado una Carta del proyecto que contiene el alcance del mismo, recursos financieros, límites de tiempo, objetivos, lista de entregables etc.

Una vez que la Carta del Proyecto es aceptada entonces se da inicio formal al proyecto.

Este grupo de Procesos tiene como propósito alinear las expectativas de todas las partes interesadas con los objetivos del proyecto, así como establecer la visión general del proyecto.

▶ **Planeación**

Todos los procesos agrupados en planeación están enfocados a desarrollar el curso de acción para alcanzar los objetivos del Proyecto.

Se desarrolla el Plan de Gestión del Proyecto, así como los documentos que serán utilizados a lo largo del mismo. Este plan contiene todos los aspectos relacionados con el tiempo, costo, alcance, recursos humanos, calidad, riesgos, adquisiciones y los compromisos de todas las partes interesadas en el proyecto.

El propósito de este grupo de procesos es trazar la estrategia y la táctica, el curso de acción y la ruta para completar el proyecto de forma exitosa.

▶ **Ejecución**

Este grupo de procesos está orientado a completar el trabajo definido en el Plan de Gestión del Proyecto.

Es aquí cuando se coordinan las personas, los recursos y las expectativas de las partes interesadas.

Es natural que haya cambios en el plan original, desde riesgos no previstos, gastos no contemplados y se requiere hacer un análisis de cómo podrían afectar estos cambios en el Plan de Gestión del Proyecto y actualizar o ajustar el plan de ser necesario.

▶ **Seguimiento y Control**

Revisar el avance, dar seguimiento al Plan de Gestión y realizar los cambios necesarios de plan las actividades principales que se ejecutan en este grupo de procesos, en pocas palabras mide el rendimiento del proyecto e identifica las variaciones en el plan de gestión.

▶ **Cierre**

Este grupo de procesos se enfoca a finalizar el proyecto formalmente y verificar que todos los procesos y todas las obligaciones contractuales se hayan cumplido.

Durante el cierre del proyecto puede ocurrir que se obtiene la aceptación del cliente o del patrocinador, la realización de una revisión completa del proyecto, documentar lecciones aprendidas, archivar todos los documentos relacionados al proyecto para que sean utilizados como datos históricos o cerrar adquisiciones.

### **3.4.1.2. Áreas del Conocimiento**

Todos los procesos pertenecientes a algún grupo de los anteriormente citados están divididos también por Áreas de Conocimiento.

La Guía del PMBOK® define un Área del Conocimiento como “Un conjunto de conceptos, términos o actividades que conforman un ámbito profesional, de gestión de proyectos o área de especialización”.

En la Guía del PMBOK® hay 10 áreas del conocimiento, las cuales se describen a continuación:

#### **4. Integración**

Identifica, define, combina, unifica y coordina los procesos y actividades para la gestión del proyecto, cumplimiento de los requisitos y terminación del proyecto.

Los procesos pertenecientes a esta área son:

- 4.1** *Develop Project Charter*
- 4.2** *Develop Project Management Plan*
- 4.3** *Direct and Manage Project Work*
- 4.4** *Monitor and Control Project Work*
- 4.5** *Perform Integrated Change Control*
- 4.6** *Close Project or Phase*

#### **5. Alcance**

En esta área del conocimiento se incluyen los procesos para definir todo el trabajo a realizarse para completar el proyecto. Su objetivo es delimitar y controlar lo que se incluye y lo que no se incluye en el proyecto.

Esta área integra los siguientes procesos:

- 5.1** *Plan Scope Management*

**5.2** *Collect Requirements*

**5.3** *Define Scope*

**5.4** *Create WBS*

**5.5** *Validate Scope*

**5.6** *Control Scope*

## **6. Tiempo**

El objetivo de esta área es gestionar el tiempo en el que se va a completar el proyecto para garantizar estará listo en el lapso de tiempo establecido.

Los procesos de esta área son:

**6.1** *Plan Schedule Management*

**6.2** *Define Activities*

**6.3** *Sequence Activities*

**6.4** *Estimate Activity Resources*

**6.5** *Estimate Activity Durations*

**6.6** *Develop Schedule*

**6.7** *Control Schedule*

## **7. Costo**

Esta Área del Conocimiento se encarga de presupuestar y controlar los costos del proyecto con el objetivo de que el costo total del proyecto quede dentro del presupuesto antes aprobado.

El Costo está integrado por los siguientes procesos:

**7.1** *Plan Cost Management*

**7.2** *Estimate Costs*

**7.3** *Determine Budget*

**7.4** *Control Costs*

## **8. Calidad**

El propósito fundamental de esta Área es que el proyecto satisfaga las necesidades para el cual fue desarrollado, para ello es necesario determinar objetivos, políticas y procedimientos de calidad, así como la mejora continua de estos procesos.

La calidad integra los siguientes procesos:

**8.1** *Plan Quality Management*

**8.2** *Perform Quality Assurance*

**8.3** *Control Quality*

## **9. Recursos Humanos**

El equipo de proyecto está conformado por aquellas personas que tienen roles y responsabilidades para llevar a cabo actividades que completen el proyecto.

Esta área del conocimiento tiene procesos para organizar, gestionar y conducir a este equipo.

Los procesos que integran a esta área son:

**9.1** *Plan Human Resource Management*

**9.2** *Acquire Project Team*

**9.3** *Develop Project Team*

**9.4** *Manage Project Team*

## **10. Comunicación**

Se dice que el éxito de un proyecto se basa principalmente en la comunicación (Morera, 2012), ya que una comunicación eficaz es el puente entre todos los interesados e involucrados en el proyecto, por ello los procesos de esta área tienen el objetivo de que toda la información del proyecto sea generada, recopilada, distribuida, almacenada y esté a la disposición de todos de manera oportuna.

Esta área está compuesta por los siguientes procesos:

**10.1** *Plan Communications Management*

**10.2** *Manage Communications*

**10.3** *Control Communications*

## **11. Riesgos**

La Guía del PMBOK® define un riesgo como “Un evento o condición incierta que, si sucede, tiene un efecto en por lo menos uno de los objetivos del proyecto. Los objetivos pueden incluir el alcance, cronograma, costo o calidad”.

Por ello, se incluyen procesos para identificar, analizar y planear una respuesta, en caso de que un riesgo suceda.

Riesgos está compuesta por los siguientes procesos:

**11.1** *Plan Risk Management*

**11.2** *Identify Risks*

**11.3** *Perform Qualitative Risk Analysis*

**11.4** *Perform Quantitative Risk Analysis*

**11.5** *Plan Risk Responses*

**11.6** *Control Risks*

## **12. Adquisiciones**

Cuando se desarrolla un proyecto puede que haya la necesidad de adquirir productos o solicitar servicios que están fuera del alcance del equipo de trabajo, es por ello, que los procesos de esta área están enfocados a la gestión de contratos y órdenes de compra del equipo de trabajo.

Esta área la componen los siguientes procesos:

**12.1** *Plan Procurement Management*

**12.2** *Conduct Procurements*

**12.3** *Control Procurements*

**12.4** *Close Procurements*

## **13. Interesados**

Los Interesados, según la Guía del PMBOK® quinta edición, son “Personas, grupos u organizaciones que pueden afectar, ser afectados por alguna decisión, actividades o resultado del proyecto”. Esto quiere decir que engloba usuarios, patrocinadores, proveedores, clientes o cualquier persona que participe activamente.

Es por ello que la Guía del PMBOK® considera fundamental la identificación de todos los Interesados y sus expectativas para crear estrategias para que su participación sea efectiva durante todo el proyecto.

Los procesos que integran esta área son:

**13.1** *Identify Stakeholders*

**13.2** *Plan Stakeholder Management*

**13.3** *Manage Stakeholder Engagement*

**13.4** *Control Stakeholder Engagement*

La siguiente tabla muestra la relación de todos los procesos por Áreas del Conocimiento y Grupos de Procesos:

Áreas del Conocimiento	Grupos de Procesos				
	Inicio	Planeación	Ejecución	Monitoreo y Control	Cierre
<b>4. Integración</b>	4.1 Develop Project Charter	4.2 Develop Project Management Plan	4.3 Direct and Manage Project Work	4.4 Monitor and Control Project Work 4.5 Perform Integrated Change Control	4.6 Close Project or Phase
<b>5. Alcance</b>		5.1 Plan Scope Management 5.2 Collect Requirements 5.3 Define Scope 5.4 Create WBS		5.5 Validate Scope 5.6 Control Scope	
<b>6. Tiempo</b>		6.1 Plan Schedule Management 6.2 Define Activities 6.3 Sequence Activities 6.4 Estimate Activity Resources 6.5 Estimate Activity Durations 6.6 Develop Schedule		6.7 Control Schedule	
<b>7. Costo</b>		7.1 Plan Cost Management 7.2 Estimate Costs 7.3 Determine Budget		7.4 Control Costs	
<b>8. Calidad</b>		8.1 Plan Quality Management	8.2 Perform Quality Assurance	8.3 Control Quality	
<b>9. Recursos Humanos</b>		9.1 Plan Human Resource Management	9.2 Acquire Project Team 9.3 Develop Project Team 9.4 Manage Project Team		
<b>10. Comunicación</b>		10.1 Plan Communications Management	10.2 Manage Communications	10.3 Control Communications	
<b>11. Riesgos</b>		11.1 Plan Risk Management 11.2 Identify Risks 11.3 Perform Qualitative Risk Analysis 11.4 Perform Quantitative Risk Analysis 11.5 Plan Risk Responses		11.6 Control Risks	
<b>12. Adquisiciones</b>		12.1 Plan Procurement Management	12.2 Conduct Procurements	12.3 Control Procurements	12.4 Close Procurements
<b>13. Interesados</b>	13.1 Identify Stakeholders	13.2 Plan Stakeholder Management	13.3 Manage Stakeholder Engagement	13.4 Control Stakeholder Engagement	

Gráfico 3.2. Mapa de los Grupos de Proceso y Áreas del Conocimiento de la Dirección de Proyectos. Guía del

PMBOK® Quinta Edición.



---

## **Capítulo 4.**

### **Proceso de Construcción de la Estrategia para el Desarrollo de Sistemas de Información: Caso Secretaría De Personal Docente de la FCA, UNAM.**

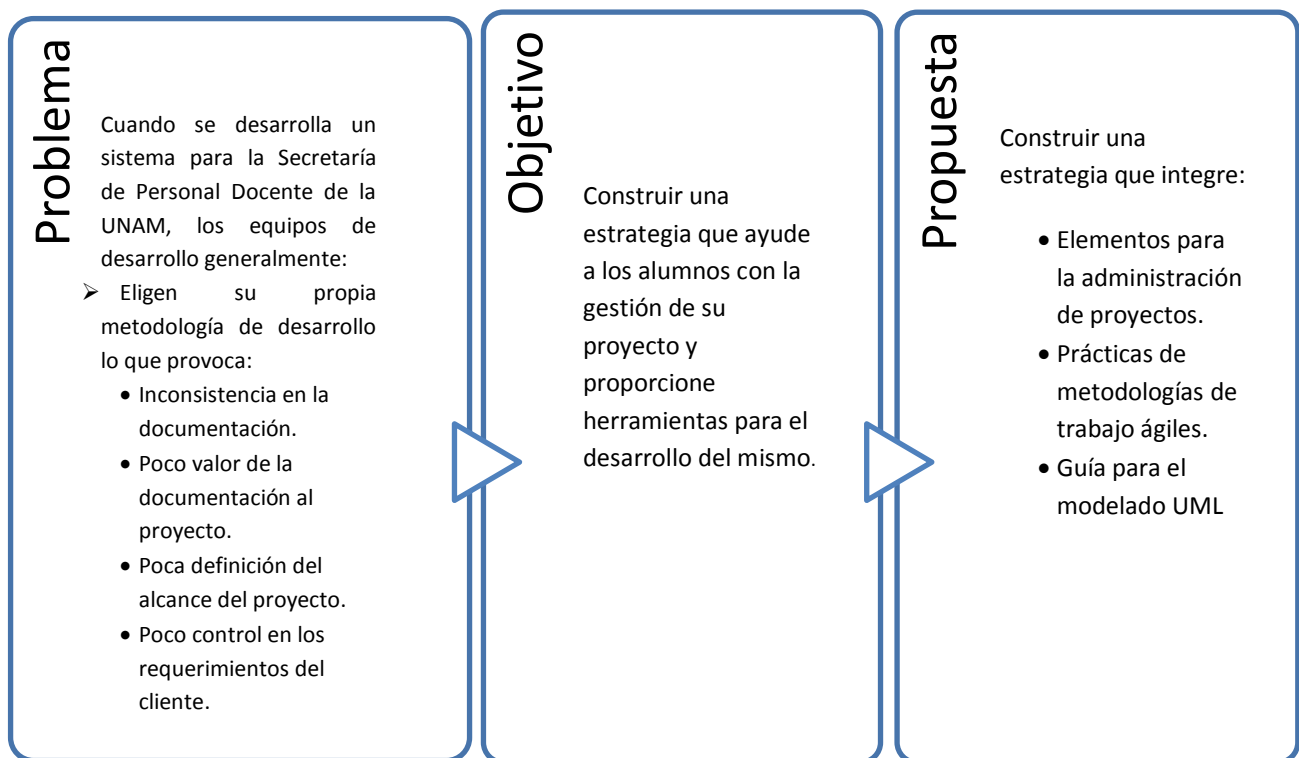
---

## 4. Proceso de Construcción de la Estrategia para el Desarrollo de Sistemas de Información

Esta Estrategia para el Desarrollo de Sistemas de Información nace como una iniciativa para ayudar a los alumnos con la construcción de software para la Secretaría de Personal Docente, a través de una guía para la Administración de Proyectos que se adapte a las necesidades de los alumnos y las necesidades de la SPD.

En este capítulo se aborda el proceso que se llevó a cabo para la creación de esta Estrategia, el cual inició en Agosto del 2013 en conjunto de un grupo de estudiantes de la Licenciatura en Informática con la intención de liberar su servicio social mediante el desarrollo de un sistema para la SPD.

### 4.1. Panorama Inicial



### 4.1.1. Estrategia

La Real Academia Española define al término estrategia como “1. Arte de dirigir las operaciones militares. 2. Arte, traza, para dirigir un asunto. 3. En un proceso regulable, conjunto de las reglas que aseguran una decisión óptima en cada momento”.<sup>16</sup>

Es un concepto antiguo de origen griego usado originalmente en el ámbito militar, sin embargo a través del tiempo ha tenido diversos usos y aplicaciones en campos como el político, administrativo, cultural o social (Contreras, 2013).

Luna (2014) define la estrategia como “La capacidad y habilidad para dirigir un asunto hasta conseguir el objetivo propuesto. También se entienden como los medios, líneas de acción y pensar cómo se deberá actuar para conseguir algún objetivo”.

Acevedo y Marín (2004) después de hacer un análisis de algunas definiciones de diversos autores del ámbito administrativo, concluyen que una estrategia es “El conjunto de acciones a seguir, en forma de planes específicos y con metas bien definidas, que contribuyen a un esfuerzo común por el cumplimiento de la misión de la organización”.

Del término estrategia se derivan otros como administración estratégica y se asocian conceptos tales como misión, objetivos, control, políticas, flexibilidad, liderazgo, seguridad, iniciativa y concentración de recursos.

### 4.1.2. Requerimientos Generales para el desarrollo de la Estrategia.

Con base en lo descrito sobre problema que da origen a esta investigación y al objetivo, se consideran los siguientes requerimientos para la Estrategia De Desarrollo de Sistemas de Información:

- Debe de proporcionar herramientas para la Administración de Proyectos.
- Tiene que describir un flujo de trabajo que guíe el desarrollo del proyecto.

---

<sup>16</sup> <http://lema.rae.es/drae/?val=estrategia>

- Tiene que ser fácil de entender y de adaptar, ya que está pensado principalmente para alumnos sin experiencia en el desarrollo de software para una organización.
- Se deben de adaptar los marcos de trabajo, metodologías y estándares para el desarrollo de software que existen actualmente.

### **4.1.3. ¿A quién está dirigida la Estrategia de Desarrollo?**

Esta Estrategia de Desarrollo está dirigida principalmente a estudiantes de la Licenciatura en Informática de la Facultad de Contaduría y Administración que se encuentren cursando el sexto semestre o posterior, a pasantes de la Licenciatura en Informática que no hayan participado en algún desarrollo de un sistema para alguna organización, o requieran de más experiencia.

Se considera que los estudiantes tienen los conocimientos introductorios a UML, metodologías de desarrollo de software, programación (Principalmente PHP), manejo de frameworks, patrones de diseño y bases de datos.

### **4.1.4. Proyecto SIGAD**

La Secretaría de Personal Docente tiene la necesidad de un sistema de información (SIGAD) que consta de varios módulos y que involucra automatización de procesos internos y se pretende que sean desarrollados por servidores sociales que cursan la carrera, así como alumnos que han elegido el Diseño de Sistema o Proyecto para una organización como opción de titulación.

Así surge el Equipo de Desarrollo OCELIT, compuesto por estudiantes y pasantes de la Licenciatura en Informática cuyo objetivo es el creación de módulos del SIGAD.

## **4.2. Criterios de Selección de la base Metodológica**

Para seleccionar la base metodológica que sostiene la Estrategia de Desarrollo de Sistemas de Información, se tomaron en cuenta los siguientes aspectos:

- I. Formación Académica de los estudiantes a los que va dirigido

- II. Posición de los estándares y certificaciones de administración de proyectos en el mercado laboral actual
- III. Adecuación a los Requerimientos Generales para el desarrollo de la Estrategia (Sección 4.1.2)

#### 4.2.1. Selección de la base metodológica

En primera instancia, se eligió a la Guía de Fundamentos de Administración de Proyectos, PMBOK®, por las siguientes razones:

- El valor de la certificación PMP® en el mercado laboral.<sup>17</sup>
- Formación académica mediante un curso de preparación para la certificación PMP® y CAPM®, impartida en la Facultad de Contaduría y Administración (Enero-Mayo, 2014).

Se considera que la Guía del PMBOK® es una referencia completa y reconocida a nivel mundial para la administración de proyectos (Rodríguez Blanco, 2012), sin embargo, tenemos que adaptar esta Guía a nuestras necesidades, entre las que están el desarrollo rápido de software sin perder de vista el alcance del proyecto. Por lo anteriormente citado, además de usar la Guía del PMBOK®, es necesario adaptar prácticas de marcos de trabajo de desarrollo de software ágil en la etapa de Ejecución, por lo que se ha decidido que Scrum<sup>18</sup> sea el complemento ya que:

- Es breve, flexible y adaptable.
- El concepto de *Sprint* proporciona rapidez en el desarrollo a través de avances tangibles para el usuario, en un tiempo de no más de 30 días usando artefactos como el *Product Backlog* y el *Sprint Backlog*.

Aunque OCELIT es un equipo que no comparte un área de trabajo común, las reuniones que *Scrum* propone, como *Sprint Review Meeting* y *Sprint Retrospective* son eventos que apoyan la comunicación entre todos los interesados en el proyecto y podrían hacerse de manera virtual. Además se pretende evaluar el desempeño del

---

<sup>17</sup> La evidencia encontrada esta citada en el Capítulo 3.

<sup>18</sup> Marco de Trabajo descrito en la sección 2.4.1

Grupo de Desarrollo ante un marco de trabajo que requiere una madurez con la que el equipo no cuenta.

Se llegó a la conclusión de que se emplearía la Guía del PMBOK® en sus fases de Inicio, Planeación, Seguimiento y Control y Cierre, se apoyaría con prácticas de *Scrum* en la parte de Ejecución, de tal manera que se controlaría la inmadurez antes mencionada.

### 4.3. Desarrollo de los Artefactos basados en el PMBOK®

Un Artefacto para esta Estrategia es un documento que servirá para planear, controlar y cerrar el proyecto.

Para el desarrollo de cada uno de estos documentos se consideran:

- Entradas:  
Información, requerimientos, acuerdos y referencias que nos ayuden con la construcción del documento.
- Técnicas y Herramientas:  
Entre los que se encuentran:
  - Juicio de Expertos: Se trata de pedir opiniones a personas con experiencia en Administración en Proyectos, que hayan ocupado estos documentos con anterioridad.
  - Lluvia de Ideas: Técnica para generar ideas originales o extensas e identificar áreas de oportunidad. (Sociedad Latinoamericana para la Calidad, 2010)
  - Entrevistas
- Salidas:  
Estructura final del documento.

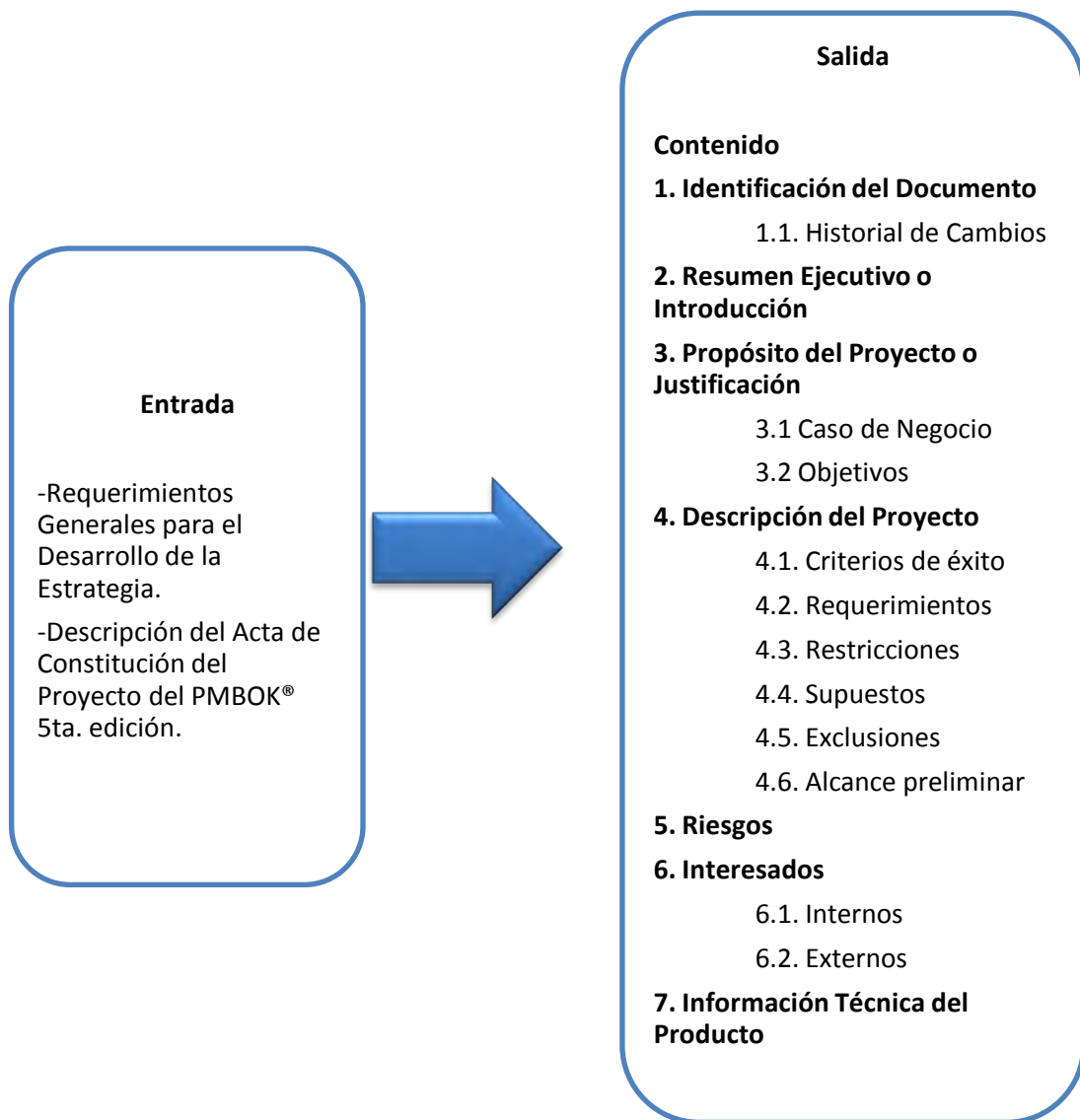
### **4.3.1. Características Generales**

Los documentos tendrán las siguientes características para su identificación y control:

- Portada  
Que identifique el Proyecto.
- Datos generales  
Nombre del área de la SPD y el nombre del módulo, así como el nombre del documento y fecha de creación.
- Control de Cambios  
Que indique la versión, la fecha, los responsables del documento y una descripción de lo que se realizó en el documento.
- Índice  
Indispensable cuando el documento tiene muchos apartados o es muy extenso.

### **4.3.2. Acta de Constitución del Proyecto**

Documento que da el inicio formal del proyecto, estableciendo una relación entre el equipo de trabajo, los usuarios y el patrocinador. Define los límites y registro del proyecto.



Entrada:

- ▶ Requerimientos Generales para el Desarrollo de la Estrategia (Descritos en el apartado 4.1.2)
- ▶ Descripción del Acta de Constitución del Proyecto del PMBOK®

La Guía del PMBOK® (PMI, 2013) describe al Acta de Constitución del Proyecto como “Un documento emitido por el iniciador del proyecto o patrocinador, que autoriza formalmente la existencia de un proyecto y confiere al director del proyecto la autoridad para asignar los recursos de la organización a las actividades del proyecto. Documenta las necesidades de negocio, los supuestos, las restricciones, el conocimiento de las necesidades y requisitos de alto nivel del

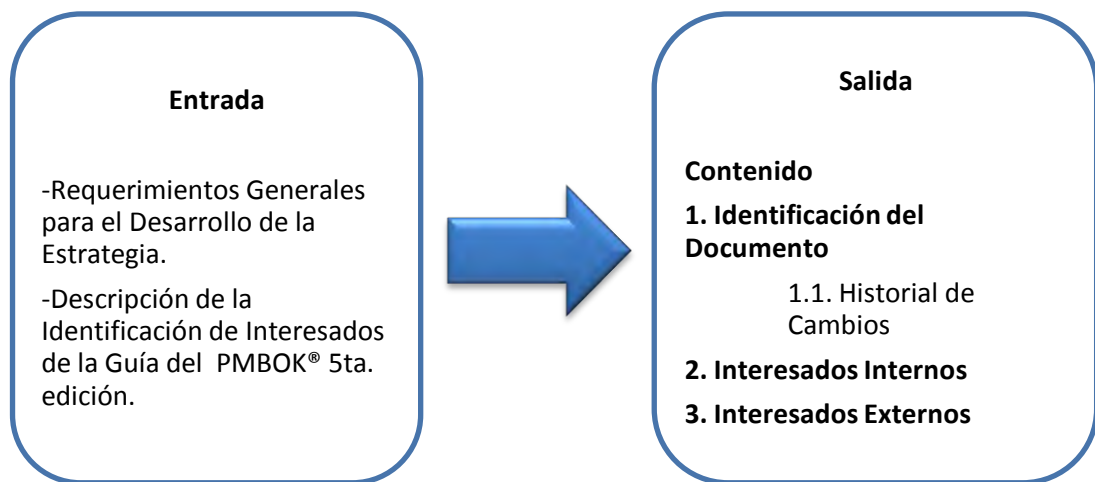


cliente y el nuevo producto, servicio o resultado que el proyecto debe proporcionar” (p.71).

En el Anexo 1: Acta de Constitución del Proyecto se presenta un ejemplo de acta Constitutiva para el módulo de Estímulos 7.5 días, que forma parte del SIGAD; en él se muestra un resumen del proyecto así como su propósito y descripción.

### 4.3.3. Interesados en el Proyecto

Este documento sirve para identificar personas, grupos u organizaciones que podrían verse afectadas durante o con el resultado del proyecto.



Entradas:

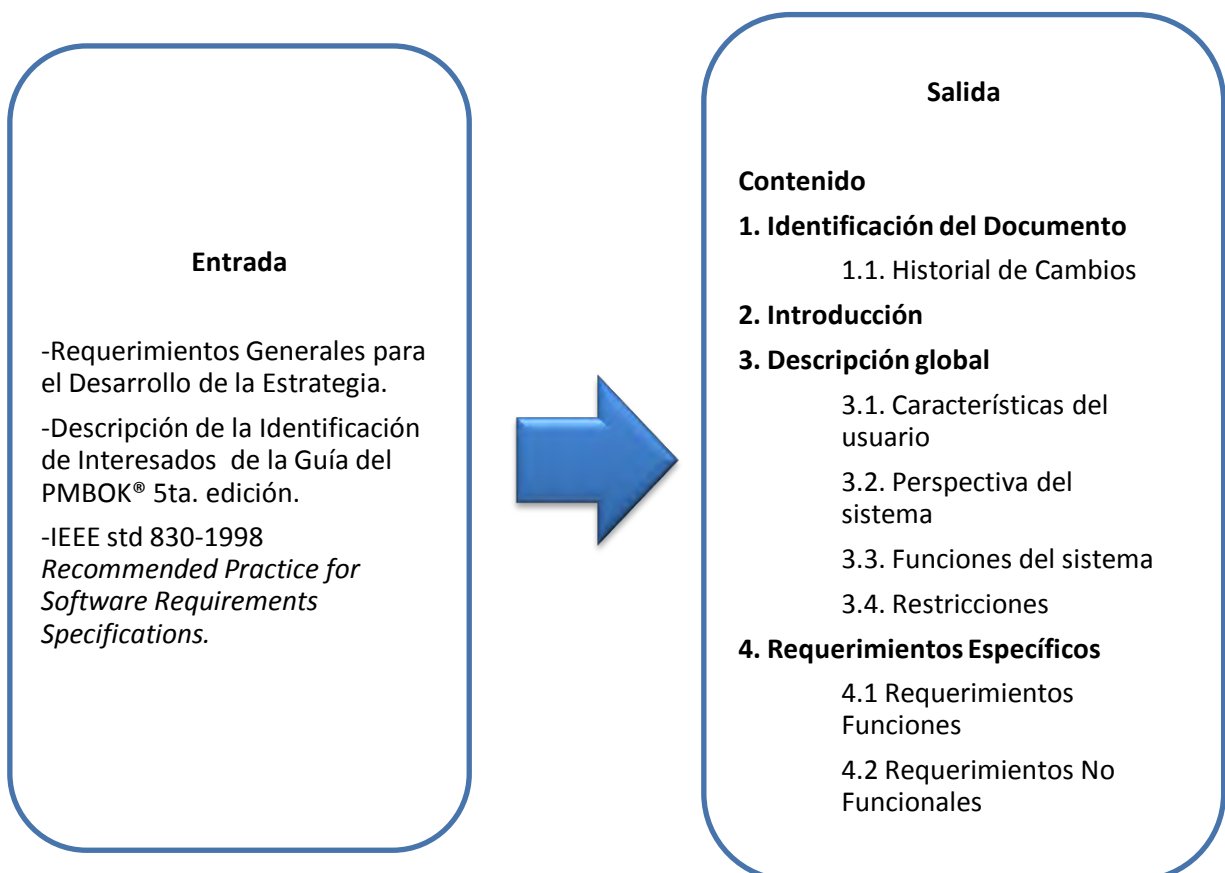
- ▶ Requerimientos Generales para el Desarrollo de la Estrategia (Descritos en el apartado 4.1.2)
- ▶ Descripción de la Identificación de Interesados de la Guía del PMBOK®  
La Guía del PMBOK®, define a los Interesados del proyecto como “Individuos, grupos u organizaciones que pueden afectar, verse afectados o percibirse a sí mismos como afectados por una decisión, actividades o resultados de un proyecto. Comprenden personas y organizaciones como clientes, patrocinadores, la organización ejecutora o el público, que están involucrados activamente en el proyecto o la conclusión del proyecto.” (PMI, 2013, p. 394). Identificar a los

Interesados permite que se establezca un enfoque adecuado para cada uno de ellos.

En el Anexo 2: Identificación de Interesados, se presenta un ejemplo de este documento, en cual describe los roles, datos de contacto, motivos y expectativas así como el impacto de cada uno de los interesados del proyecto, ya sean externos o internos.

#### 4.3.4. Requerimientos del Software

La recopilación de los Requerimientos es de suma importancia ya que delimitan el alcance del proyecto, para ello debemos de determinar, documentar y gestionar las necesidades de las partes interesadas y los requisitos para cubrir los objetivos del proyecto.



Entradas:

- ▶ Requerimientos Generales para el Desarrollo de la Estrategia (Descritos en el apartado 4.1.2).

- ▶ Descripción del proceso de Identificación de Interesados de la Guía del PMBOK®

La Guía del PMBOK® (PMI, 2013) menciona que los requisitos “Incluyen condiciones o capacidades que el proyecto debe cumplir o que deben estar presentes en el producto, servicio o resultado para satisfacer un acuerdo u otra especificación formalmente impuesta. Los requisitos incluyen las necesidades y expectativas cuantificadas y documentadas del patrocinador” (p. 112).

Asimismo, hace mención de algunas clasificaciones de Requerimientos:

- De negocio
- De los interesados
- De las soluciones
  - Funcionales
  - No funcionales
- De transición
- De calidad

- ▶ IEEE std 830-1998 *Recommended Practice for Software Requirements Specifications*:

Este documento describe criterios recomendados para la especificación de Requerimientos de Software.

A continuación se muestra la estructura del documento y una breve explicación de cada una de sus partes:

### **1. Definiciones**

Este estándar define:

- Contrato: Documento legalmente obligatorio en donde se plasman los acuerdos entre Cliente y Proveedor.
- Cliente: Persona u Organización que paga por el producto y puede o no proporcionar los requerimientos del mismo.
- Proveedor: Persona u organización que desarrolla el producto.
- Usuario: Persona u Organización que actúan u operan directamente con el producto.

## 2. Consideraciones para hacer una buena Especificación de Requerimientos de Software (ERS)<sup>19</sup>

### a. Naturaleza del ERS

Un ERS se construye para especificar:

- La funcionalidad del software
- Las interfaces Externas
- La ejecución
- Los atributos
- Las restricciones del diseño

### b. Ambiente del ERS

La ERS debe:

- Definir los requisitos del software correctamente
- No debe describir planes o detalles de aplicación
- No debe imponer restricciones adicionales en el software

### c. Características de una buena ERS

Un buen documento de ERS debe de ser:

- Correcto
- Inequívoco
- Completo
- Consistente
- Delinear que tiene importancia y/o estabilidad
- Comprobable
- Modificable
- Identificable

### d. Preparación de los *join* del ERS

Se refiere a que el Cliente y el Proveedor deben trabajar juntos para lograr un buen escrito y que sea entendible.

### e. Evolución de la ERS

Puede que la ERS necesite actualizaciones, por ello se deben tener las siguientes consideraciones:

---

<sup>19</sup> En inglés SRS, por las siglas *Software Requirements Specification*

- Especificar los requisitos como se conocen en el momento, aún cuando se sepa que podrían cambiar.
- Informar e identificar los cambios que haya sufrido el documento.

**f. Prototipado**

Un prototipo del producto, se usa para obtener requisitos del software tales como: aspecto general, formatos de informes, etc.

**g. Incorporación del diseño dentro de la ERS**

Un diseño describe un subcomponente particular de un sistema o de la interfaz con otros subcomponentes, por lo tanto, el diseño es distinto a un requerimiento, ya que éste se refiere a una función externa o algún atributo; es por esta razón que en ocasiones debe separar el diseño de los requisitos aunque se debe prestar atención si el diseño afecta a los requerimientos del sistema, de ser así, entonces sí es válido adjuntar el diseño o partes de él a la ERS.

**h. Incorporación de los requisitos del proyecto en la ERS**

La ERS incluye solo los requerimientos del software, lo que significa que se deben de excluir aspectos como:

- Costo
- Tiempos de entrega
- Métodos de desarrollo
- Criterios de aprobación
- Procedimiento de aceptación

**3. Partes de una ERS**

**a. Introducción**

Este apartado debe de proporcionar una apreciación global del documento y debe contener:

- Propósito
- Alcance
- Definiciones, siglas y abreviaturas
- Referencias
- Apreciación Global

## **b. Descripción global**

En esta sección se describen los factores que afectan al producto, así como los requerimientos del mismo:

- Perspectiva
- Funciones
- Características de los usuarios
- Restricciones
- Dependencias
- Requisitos

## **c. Requisitos específicos**

En esta sección se deben de describir los Requerimientos del Software a un nivel de detalle suficiente para permitir su desarrollo.

Implica cuestiones, tales como:

- Funciones
- Requisitos del desarrollo
- Bases de datos
- Restricciones del diseño
- Atributos del software

## **d. Información de apoyo**

Esta información tiene como objetivo hacer a la ERS más fácil de usar, e incluye:

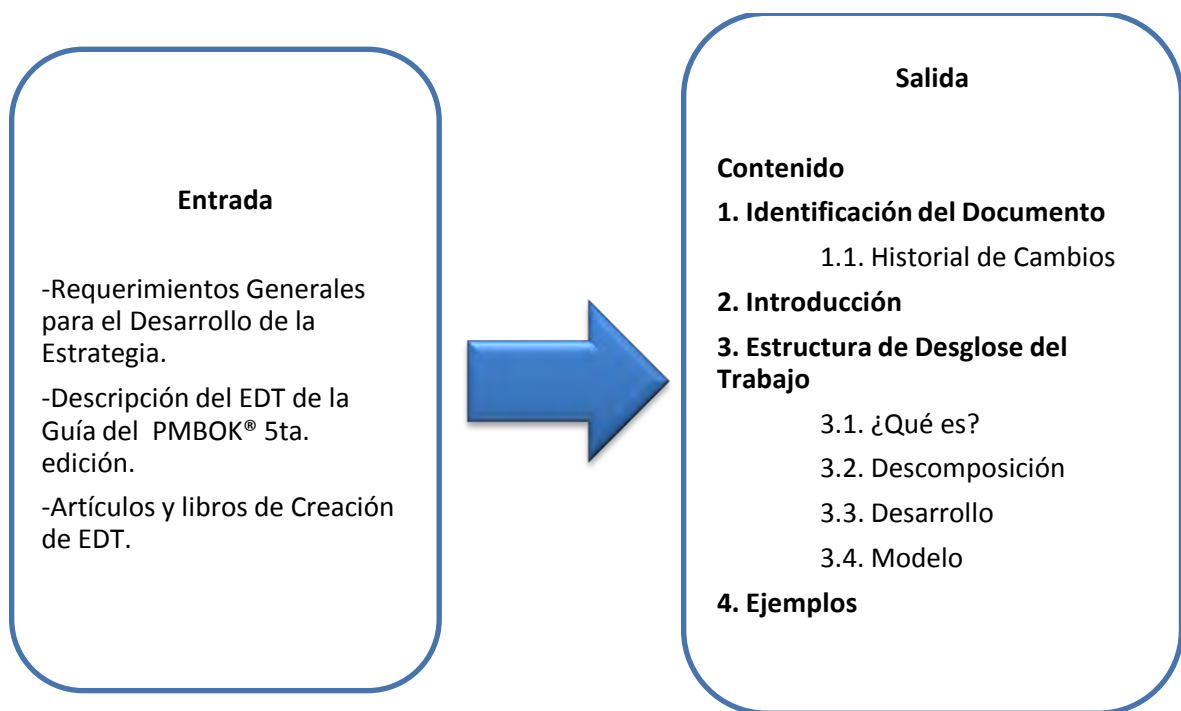
- Tabla de contenido
- Índice
- Apéndice

En el Anexo 3: Requerimientos del Proyecto, se muestra un ejemplo del Documento de Requerimientos del módulo de Estímulos del SIGAD. En este documento se hace una descripción global del sistema para posteriormente detallar y priorizar los requerimientos funcionales y no funcionales, considerando los aspectos anteriormente mencionados.

### 4.3.5. Creación de la Estructura de Desglose de Trabajo.

La Estructura de Desglose de Trabajo (EDT) o *Work Breakdown Structure* (WBS) es una descomposición jerárquica, basada en los entregables del trabajo que debe ejecutar el equipo del proyecto para lograr los objetivos y crear los entregables requeridos con el fin de proporcionar una guía y medir el avance en el desarrollo del producto.

Se ha decidido desarrollar una guía breve para crear una EDT, ya que en este caso no es posible crear una plantilla.



#### Entradas:

- ▶ Requerimientos Generales para el Desarrollo de la Estrategia (Descritos en el apartado 4.1.2)
- ▶ Descripción de la EDT de la Guía del PMBOK®

La Guía del PMBOK® (PMI, 2013), menciona que la EDT “es el proceso de subdividir los entregables del proyecto y el trabajo del proyecto en componentes más pequeños y más fáciles de manejar. El beneficio de este proceso es que proporciona una visión estructurada de lo que se va a entregar” (p. 125).

Así que la Estructura de Desglose de Trabajo, tiene como objetivo organizar y definir el alcance del proyecto.

► Artículos de Creación de la EDT.

El material consultado para hacer esta guía, fue:

- Barakat, Mohammed, (2012). *Haciendo fácil la Estructura de Desglose del Trabajo*, disponible en:  
[http://www.liderdeproyecto.com/articulos/haciendo\\_facil\\_la\\_estructura\\_de\\_desglose\\_del\\_trabajo.html](http://www.liderdeproyecto.com/articulos/haciendo_facil_la_estructura_de_desglose_del_trabajo.html), consultado el 20 de febrero del 2014.
- Sánchez Orduña, Ricardo. (2013). *La Estructura de Desglose de Trabajo (EDT)*. Disponible en: <http://www.pm4r.org/esp/punto/blog-de-pm4r/entry/la-estructura-de-desglose-del-trabajo-edt>, consultado el 25 de febrero del 2014.
- Matis, Micah, (2014), *Work Breakdown Structure: Purpose, Process and Pitfalls*. Disponible en: <http://cdn.projectsmart.co.uk/pdf/work-breakdown-structure-purpose-process-pitfalls.pdf>, consultado el 25 de febrero del 2014.
- De Rojas, Fermín, (2011). *Qué es Work Breakdown Structure (WBS)*, disponible en: <http://spanishpmo.com/index.php/que-es-work-breakdown-structure-wbs/>, consultado el 25 de febrero del 2014.
- Gordon, Ann & McDonough, Michele, (2011), *Tips for Building a Work Breakdown Structure*, disponible en <http://www.brighthubpm.com/project-planning/2940-tips-for-building-a-work-breakdown-structure/>, consultado el 25 de febrero del 2014.

En el Anexo 4: Creación del WBS/EDT se presenta esta breve guía de creación de esta Estructura, donde se explica qué es, cómo se desarrolla y algunos ejemplos de la misma, dos de ellos, de fuente propia.

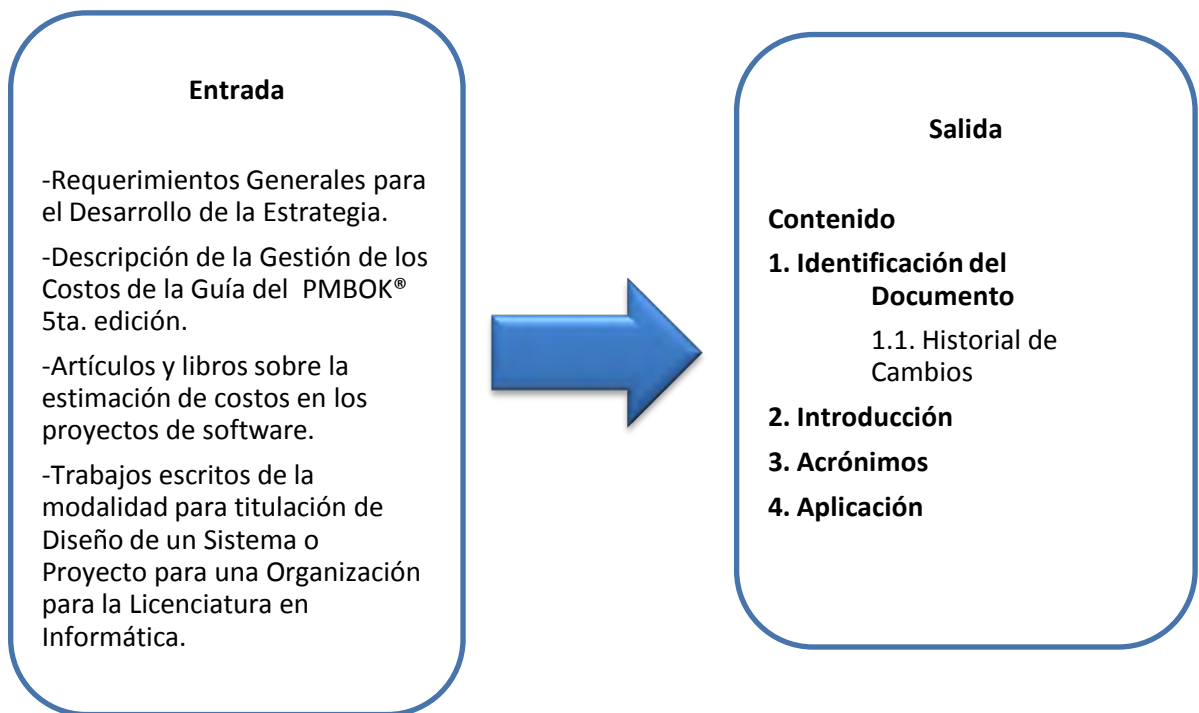
#### 4.3.6. Costos

Debido a la naturaleza de los proyectos de software que se desarrollarán para la Secretaría de Personal Docente, no es necesario hacer planeación de Presupuesto, Adquisiciones y Costos, ya que los desarrolladores no recibirán pago monetario por los sistemas ni se realizarán compras de ninguna índole por parte de la SPD para el



desarrollo de los proyectos, sin embargo, el Costo total del proyecto es información que a menudo suele ser incluida en los trabajos de Diseño de un Sistema y Proyecto para una organización, es por esta razón que se incluye el método de estimación de Puntos de Casos de Uso que nos permitirá calcular el costo del proyecto.

Los procesos que integran la gestión de los costos del proyecto del PMBOK® se ocupa, como su nombre lo indica, del costo de los recursos que se necesitan para completar las actividades del proyecto. Estos procesos están encaminados a estimar, presupuestar, financiar, así como gestionar y controlar los costos.



#### **Entradas:**

- ▶ Requerimientos Generales para el Desarrollo de la Estrategia (Descritos en el apartado 4.1.2)
- ▶ Descripción de la Gestión de los Costos de la Guía del PMBOK®

La Guía del PMBOK® (PMI, 2013), menciona que la Gestión de los Costos “es el proceso que establece las políticas, los procedimientos y la documentación

necesarios para planificar, gestionar, ejecutar el gasto y controlar los costos del proyecto. El beneficio clave de este proceso es que proporciona guía y dirección sobre cómo se gestionarán los costos del proyecto a lo largo del mismo” (p. 195).

También menciona que la estimación del costo, es una evaluación cuantitativa de los costos requeridos para completar el proyecto, que estos costos pueden ir resumidos o detallados y que se incluye trabajo directo, materiales, equipamiento, servicios, instalaciones, entre otros.

► Artículos sobre estimación de Costos en los proyectos de software

El material consultado fue:

- López Martín, Cuauhtémoc, (2007), *Estimación del esfuerzo de desarrollo de software*. En línea, disponible en:  
<http://www.cimat.mx/Eventos/seminariodetecnologias/EstimacionDelEsfuerzo.pdf>, consultado el 3 de marzo del 2014.
- Collaris, Remi-Armand, (2009). *Estimación del costo del software usando puntuación en casos de uso: clarificar las transacciones de casos de uso*. En línea, disponible en:  
[http://www.ibm.com/developerworks/ssa/rational/library/edge/09/mar09/collaris\\_dekker/](http://www.ibm.com/developerworks/ssa/rational/library/edge/09/mar09/collaris_dekker/), consultado en 3 de marzo del 2014.
- Valero Orea, Sergio, (2009). *Estimación de proyectos de software con puntos de casos de uso*. En línea, disponible en:  
<http://www.utim.edu.mx/~svalero/docs/id45.pdf>, consultado el 4 de marzo del 2014.
- Gómez, Julián, (2013). *Método de Estimación puntos de casos de uso (Use case points)*. En línea, disponible en:  
<http://www.laboratorioti.com/2013/02/14/metodo-de-estimacion-puntos-casos-de-uso-use-case-points/>, consultado en 4 de marzo del 2014.

► Trabajos escritos de la modalidad para titulación de Diseño de un Sistema o Proyecto para una Organización para la Licenciatura en Informática.

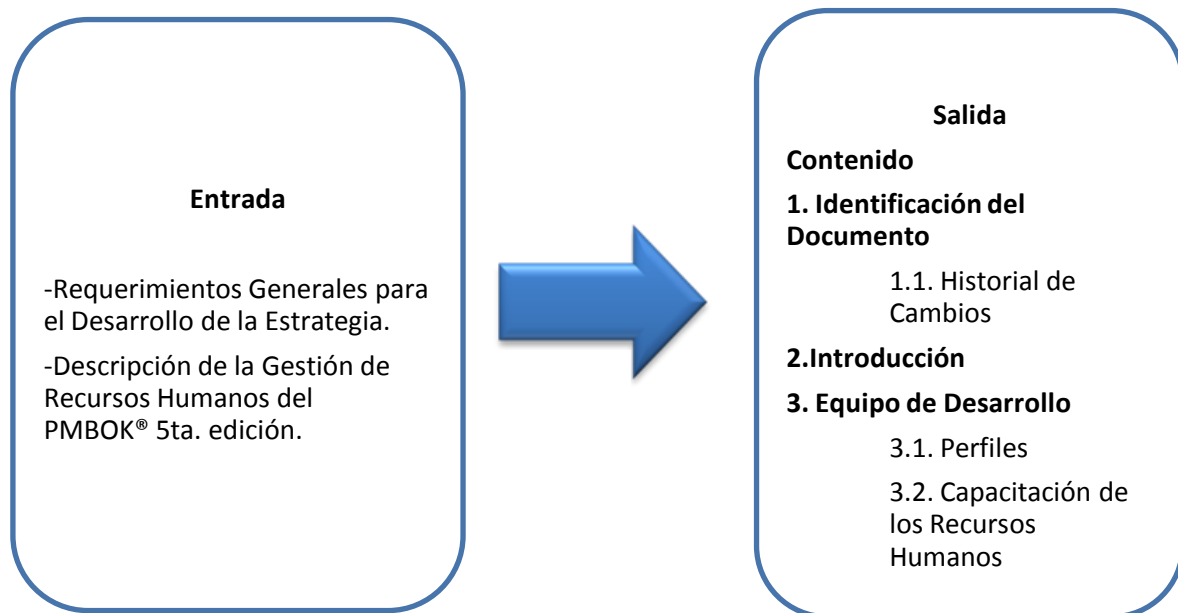
- Aguilar Vargas, Ana Karen, sustentante (2014). Sistema de selección de profesores de asignatura.

- Espinosa Tabaco, Salvador, sustentante (2014). Sistema de expediente digital para la Secretaría de Personal Docente de la Facultad de Contaduría y Administración

En el Anexo 5: Costos del Proyecto, se muestra un ejemplo del documento donde se calculan los costos del proyecto del módulo de Estímulos 7.5 del SIGAD, utilizando el método de Puntos de Casos de Uso.

#### 4.3.7. Recursos Humanos

La gestión de los Recursos Humanos es el proceso donde se identifican y documentan los perfiles y habilidades requeridas en los integrantes del equipo de desarrollo, así como los roles y responsabilidades de cada uno de ellos.



## **Entrada:**

- ▶ Requerimientos Generales para el Desarrollo de la Estrategia (Descritos en el apartado 4.1.2.)
- ▶ Descripción de la Gestión de Recursos Humanos de la Guía del PMBOK®

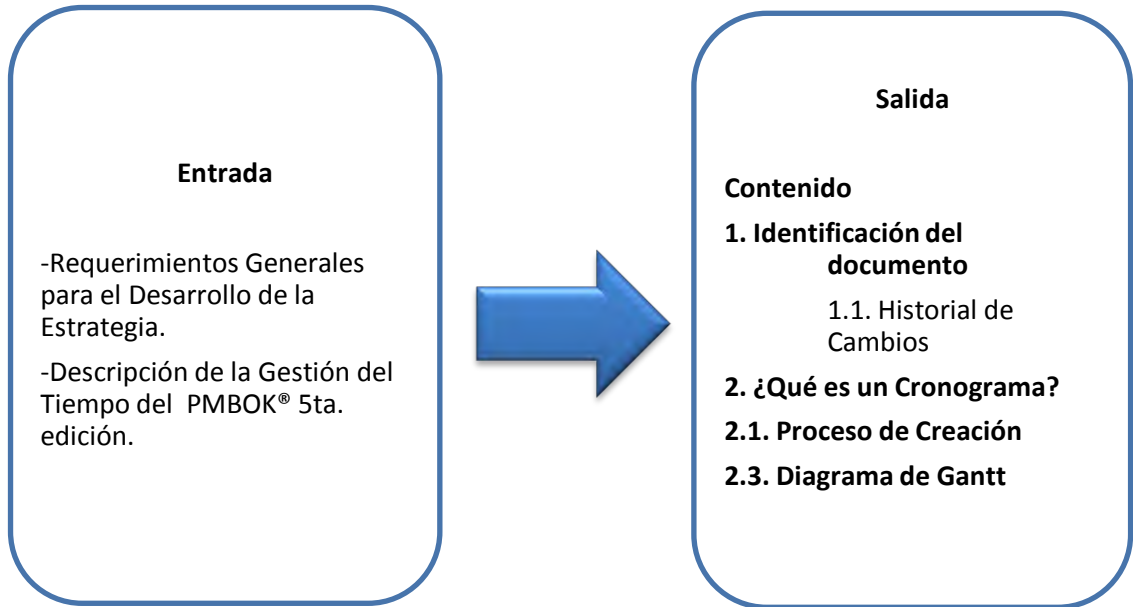
El equipo del proyecto lo componen un grupo de personas con habilidades y conocimientos necesarios para participar en actividades que ayuden a completar el trabajo del proyecto.

La guía del PMBOK® (PMI, 2013), señala que planificar la Gestión de los Recursos Humanos es “el proceso de identificar y documentar los roles dentro del proyecto, las responsabilidades, habilidades requeridas y las relaciones de comunicación, así como de crear un plan para la gestión de personal. El beneficio clave de este proceso es que establece los roles y responsabilidades del proyecto, los organigramas del proyecto y el plan para la gestión de personal, el cual incluye el cronograma para la adquisición y liberación del personal” (p. 258).

En el Anexo 6: Recursos Humanos se presenta el documento donde se describen los perfiles necesarios para integrar el Equipo de Desarrollo así, como una sección donde se detalla un plan de capacitación o introducción al proyecto.

### 4.3.8. Tiempo

La Gestión del Tiempo incluye procesos para planear y controlar la duración de las actividades del proyecto, para garantizar que estos sean terminados en el plazo de tiempo previsto.



## Entrada:

- ▶ Requerimientos Generales para el Desarrollo de la Estrategia (Descritos en el apartado 4.1.2).
- ▶ Descripción de la Gestión del Tiempo de la Guía del PMBOK®

La Guía del PMBOK® identifica los siguientes procesos para la gestión del Tiempo:

- Planificar la Gestión del Cronograma
- Definir las Actividades
- Secuenciar las Actividades
- Estimar los Recursos de las Actividades
- Estimar la duración de las Actividades
- Desarrollar el Cronograma
- Controlar el Cronograma

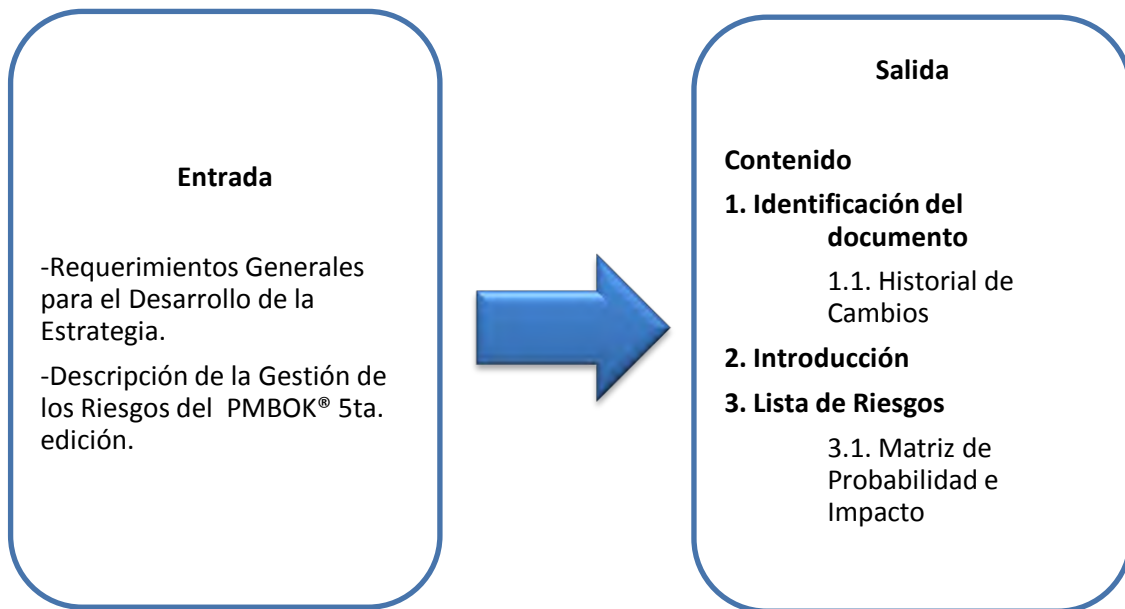
Debido a que para esta Estrategia de Desarrollo se ha decidido el uso de prácticas y artefactos de Scrum, donde hay listas de actividades, llamadas *Product* y *Sprint Backlogs*, para una mejor gestión del Tiempo, se ha decidido incorporar un cronograma, el cual ayudará a gestionar el tiempo en el que se ejecuta cada Sprint o el proyecto en general.

En el Anexo 7: Cronograma se explica qué es, cómo se construye de manera general un cronograma y como se puede representar a través de un diagrama de Gantt.

### 4.3.9.Riesgos

Identificar los riesgos es el proceso donde se determinan aquellas acciones o eventos que puedan afectar el proyecto y sus características. Posteriormente, se realiza un análisis cualitativo y cuantitativo de los riesgos identificados para que finalmente se desarrolle un plan de respuesta a los riesgos.

Los riesgos, su análisis y el plan de respuesta están contenidos en la sección de Gestión de los Riesgos del Proyecto de la Guía del PMBOK®.



#### **Entrada:**

- ▶ Requerimientos Generales para el Desarrollo de la Estrategia (Descritos en el apartado 4.1.2).
- ▶ Descripción de la Gestión de Riesgos de la Guía del PMBOK®

Según la Guía del PMBOK® (PMI, 2013, p. 310), un riesgo es “un evento o condición incierta que, de producirse, tiene un efecto positivo o negativo en uno o más de los objetivos del proyecto, tales como el alcance, el cronograma, el costo y la calidad”. Debido a que un riesgo puede impactar en los objetivos del proyecto, es importante que se realicen actividades de gestión de riesgos, cuyo objetivo es asegurar la visibilidad de los riesgos, para que sean analizados y de ser el caso, se realicen planes de respuesta a los riesgos.

Entre los procesos que constituyen la Gestión de Riesgos, se encuentran:

- Identificar los Riesgos:

La Guía del PMBOK® (PMI, 2013), describe a la Identificación de Riesgos como “proceso para determinar los riesgos que pueden afectar al proyecto y documentar sus características. El beneficio clave de este

proceso es la documentación de los riesgos existentes y el conocimiento y la capacidad que confiere al equipo del proyecto para anticipar eventos” (p. 319). De esta manera, señala que es un proceso iterativo y que además se pueden identificar riesgos en cualquier punto del proyecto.

- Realizar el análisis cualitativo de Riesgos:

La Guía del PMBOK® (PMI, 2013, p. 328) menciona que “es el proceso de priorizar riesgos para análisis o acción posterior, evaluando y combinando la probabilidad de ocurrencia e impacto de dichos riesgos. El beneficio clave de este proceso es que permite a los directores de proyecto reducir el nivel de incertidumbre y concentrarse en los riesgos de alta prioridad.”

Realizar este tipo de análisis constituye un medio para priorizar los riesgos, a través de su probabilidad e impacto, para así saber qué riesgos son más probables que sucedan y enfocarnos en ellos.

Este análisis se apoya del uso de herramientas como la Matriz de Probabilidad e Impacto, donde cada riesgo es calificado de acuerdo a la probabilidad de que ocurra y al impacto que tenga si llegara a ocurrir; de esta manera se clasifica el riesgo como alto, moderado y bajo.

- Realizar el análisis cuantitativo de Riesgos:

La Guía del PMBOK® (PMI, 2013, p. 333), dice que “es el proceso de analizar numéricamente el efecto de los riesgos identificados sobre los objetivos generales del proyecto. El beneficio clave de este proceso es que genera información cuantitativa sobre los riesgos para apoyar la toma de decisiones a fin de reducir la incertidumbre del proyecto”.

Debido a la naturaleza de este análisis, es indispensable contar con información exacta acerca del costo, tiempo y actividades para hacer diversos análisis para finalmente presentar los resultados en con gráficas y diagramas.

- Planificar la respuesta a los Riesgos:

La Guía del PMBOK® (PMI, 2013) la define como el “proceso de desarrollar opciones y acciones para mejorar las oportunidades y reducir las amenazas a los objetivos del proyecto. El beneficio clave de este



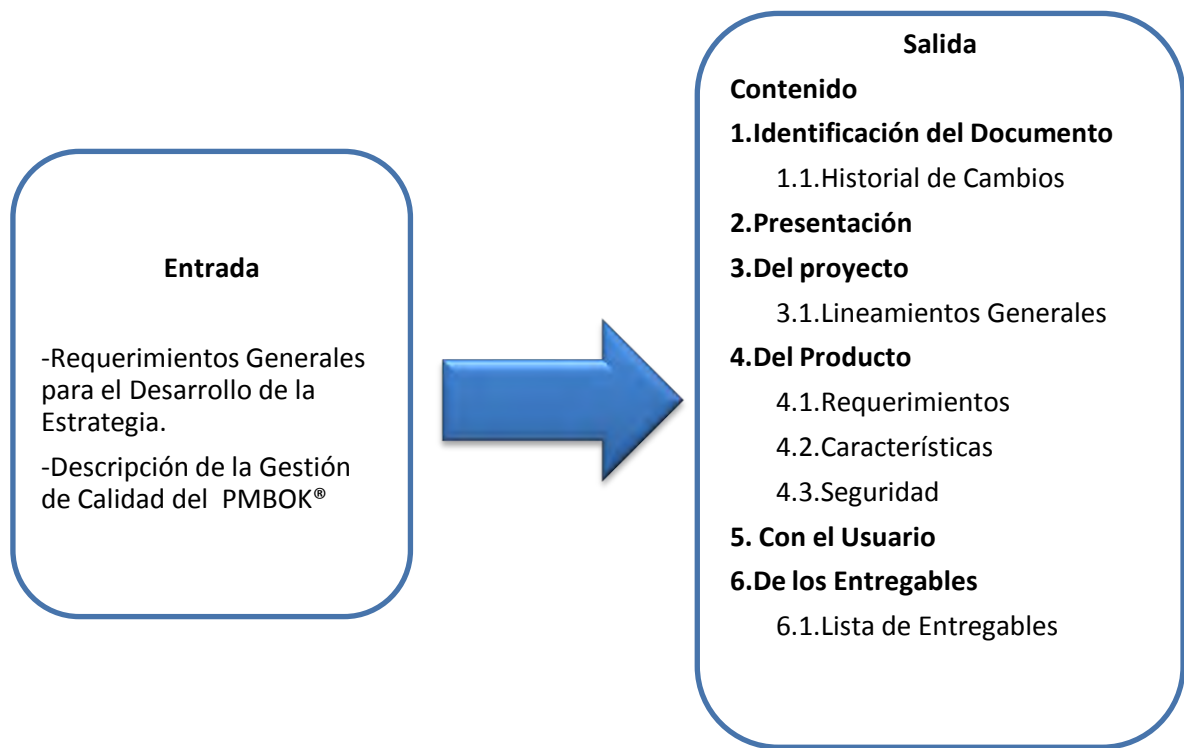
proceso es que aborda los riesgos en función de su prioridad, introduciendo recursos y actividades en el presupuesto, el cronograma y el plan para la dirección del proyecto, según las necesidades” (p. 342).

En este plan de respuesta a los Riesgos se describen las acciones a ejecutar así como los responsables de los mismos. Estas respuestas deben ser las adecuadas para cada Riesgo en función de la prioridad e importancia de los mismos.

En el Anexo 8: Riesgos, se muestra el documento donde podrán ser enlistados los riesgos de un proyecto, así como un análisis cualitativo, empleando la matriz de probabilidad e impacto.

#### **4.3.10. Calidad**

La gestión de la calidad asegura que los requerimientos del proyecto y del producto se cumplan y se validen, para ello se determinan actividades, políticas, objetivos y responsabilidades para que el proyecto satisfaga las necesidades para las que está siendo desarrollado.



**Entrada:**

- ▶ Requerimientos Generales para el Desarrollo de la Estrategia (Descritos en el apartado 4.1.2.).
- ▶ Descripción de la Gestión de Calidad de la Guía del PMBOK®

En la Gestión de la Calidad del proyecto se trabaja para asegurar que los requisitos del proyecto y del producto sean alcanzados.

Entre los procesos que constituyen la Gestión de Calidad se encuentran:

- Planificar la Gestión de la Calidad: Se identifican los requisitos y/o estándares para el proyecto y sus entregables.
- Realizar el aseguramiento de Calidad: Se auditan los requisitos de calidad con el fin de asegurar que se utilicen las normas de manera adecuada.
- Controlar la Calidad: Se monitorean y registran los resultados para evaluar el desempeño y recomendar los cambios que sean necesarios.

Asimismo hace mención del enfoque básico de la calidad de la Organización Internacional de Normalización (ISO).

“La familia de normas ISO 9000 se sustenta en ocho principios de gestión de la calidad, y constituyen una referencia básica necesaria para el entendimiento y la implantación adecuada de los requisitos de la norma ISO 9001” (Beltrán et al, 2009).

Los principios de gestión de la calidad son: (ISO 9000:2005)

- 1) **Enfoque al cliente:** Comprender sus necesidades, satisfacer sus requisitos y exceder sus expectativas.
- 2) **Liderazgo:** Establece unidad de propósito y crear un ambiente en el cual se puedan lograr todos los objetivos.
- 3) **Participación del personal:** Que el personal de todos niveles participen hace que usen sus habilidades en beneficio de la organización.
- 4) **Enfoque basado en procesos:** Establece que si las actividades y los recursos de gestionan como un proceso el resultado se alcanza de una manera eficaz.
- 5) **Enfoque de sistemas para la gestión:** Se identifican, entienden y gestionan los procesos relacionados entre sí, que como a manera de sistema contribuyen a la eficacia y eficiencia en el logro de los objetivos de una organización.
- 6) **Mejora continua:** Del desempeño de toda la organización.
- 7) **Enfoque basado en hechos para la toma de decisión:** Analizar datos e información para basar la toma de decisiones de manera eficaz.
- 8) **Relaciones mutuamente beneficiosas con el proveedor:** Una relación beneficiosa aumenta la capacidad de crear valor para ambas partes.

Con la información anteriormente citada, nos resta definir en primera instancia qué será la Calidad en nuestro proyecto, identificando los siguientes puntos en donde se deben de cumplir ciertas normas, con el fin de:

- 1) Asegurar que el proyecto esté terminado en tiempo y forma.

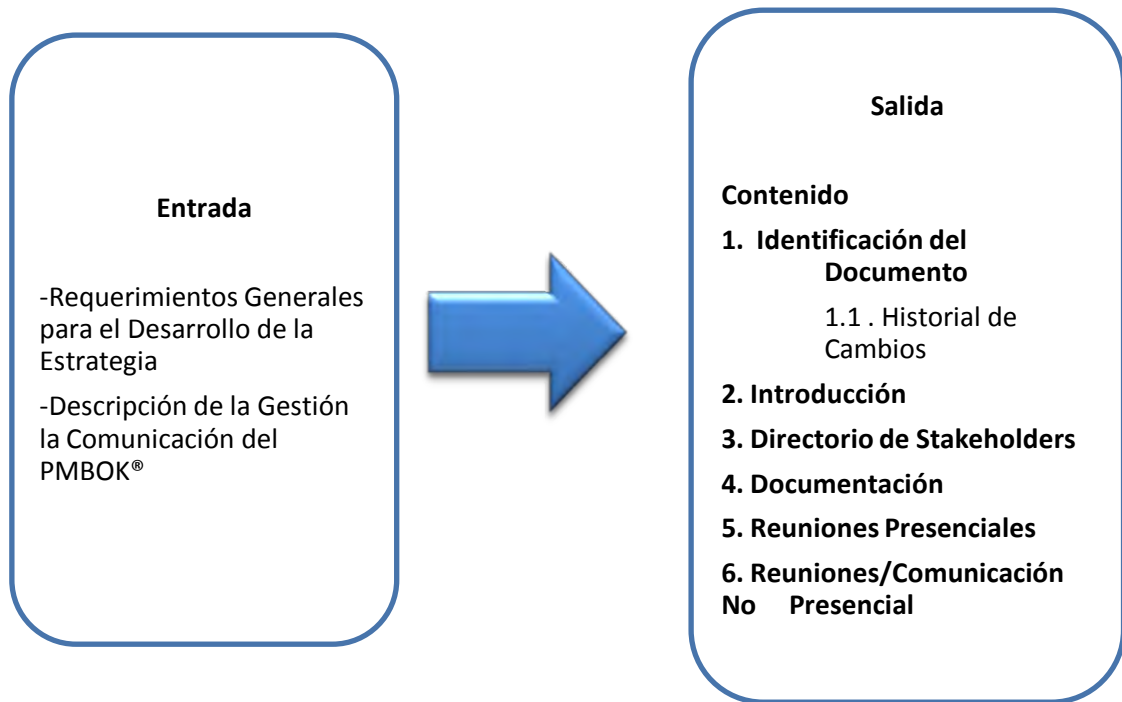
- 2) Que el producto cumpla con los requerimientos y con las características que el usuario solicitó.
- 3) Tener un registro de las reuniones en donde se muestren los avances al usuario.
- 4) Tener un orden en los entregables del proyecto.

El Anexo 9: Calidad es un documento de Gestión de Calidad, el cual incluye los lineamientos generales que los miembros del equipo de desarrollo deben cumplir para que el proyecto sea concluido en tiempo y forma, asegurando que el producto cumpla con las características y requerimientos que el usuario y el equipo de desarrollo acordaron.

Adicionalmente se incluyen recomendaciones para la preparación de las reuniones con el usuario y una lista de entregables acorde a la esta Estrategia.

#### **4.3.11. Comunicación**

La comunicación entre los interesados del proyecto constituye en gran medida el éxito del proyecto, por ello, es necesario desarrollar un plan de comunicación eficaz acorde a las necesidades del proyecto, a los interesados y a los recursos disponibles.



#### **Entrada:**

- ▶ Requerimientos Generales para el Desarrollo de la Estrategia (Descritos en el apartado 4.1.2.)
- ▶ Descripción de la Gestión de la Comunicación de la Guía del PMBOK®

La Guía del PMBOK® menciona que la gestión de la Comunicación “Incluye los procesos requeridos para asegurar que la planificación, recopilación, creación, distribución, almacenamiento, recuperación, gestión, control, monitoreo y disposición final de la información del proyecto sean oportunos y adecuados. Una comunicación eficaz crea un puente entre diferentes interesados que pueden tener diferentes antecedentes culturales y organizacionales, diferentes niveles de experiencia, y diferentes perspectivas e intereses, lo cual impacta o influye en la ejecución o resultado del proyecto.” (PMI, 2013, p. 287).

La gestión de las Comunicaciones, incluye tres procesos:

- Planificar la Gestión de las Comunicaciones
  - Desarrolla un enfoque para las comunicaciones del proyecto basado en las necesidades de información de los interesados y de los recursos disponibles.

- Gestionar las Comunicaciones

Con base en el plan de la Gestión de las Comunicaciones, en este proceso se creará, recopilará, distribuirá, almacenará, recuperará y realizará la disposición de la información.

- Controlar las Comunicaciones

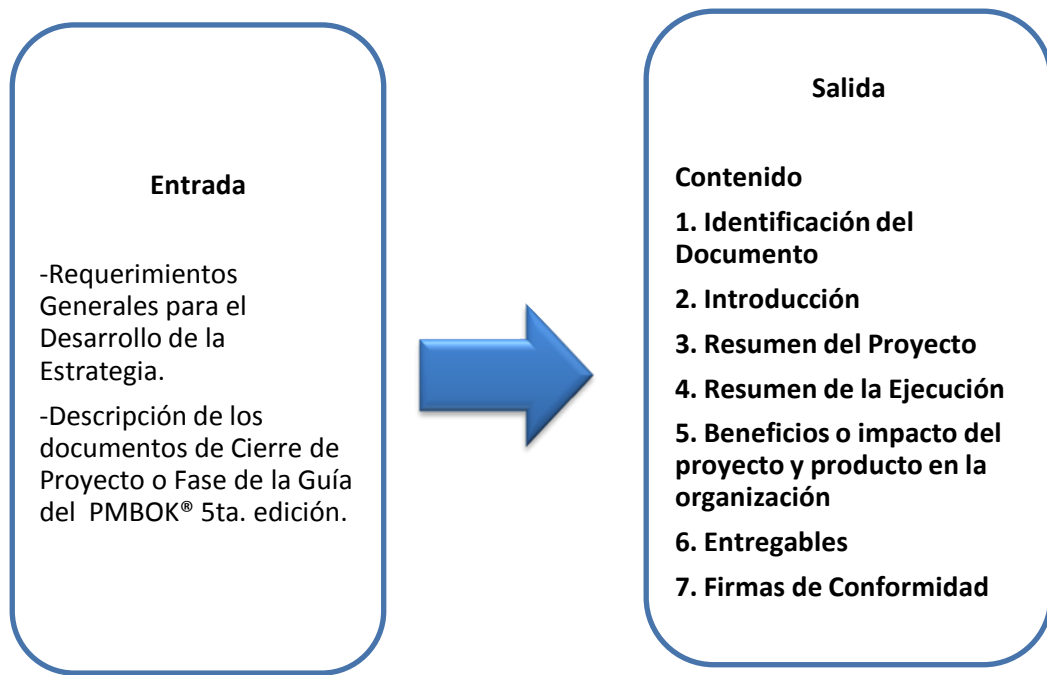
El objetivo de este proceso es monitorear y controlar las comunicaciones para satisfacer las necesidades de información de los interesados.

En el Anexo 10: Comunicación, se muestra el documento que puede ser utilizado para la Gestión de la Comunicación; contiene un directorio, información acerca de la ubicación de la documentación, así como apartados donde se describen los planes de reuniones presenciales y comunicación por medios electrónicos y virtuales.

#### **4.3.12. Acta de Cierre del Proyecto**

El proceso del Cierre del Proyecto o Fase, tiene como objetivo dar por finalizadas todas las actividades de Inicio, Planeación, Ejecución y Seguimiento y Control para completar el Proyecto o Fase de manera formal.

Una manera de cerrar formalmente el Proyecto o Fase es empleando documentos de Cierre.



#### **Entrada:**

- ▶ Requerimientos Generales para el Desarrollo de la Estrategia (Descritos en el apartado 4.1.2.)
- ▶ Descripción de los documentos de Cierre de Proyecto o Fase de la Guía del PMBOK®

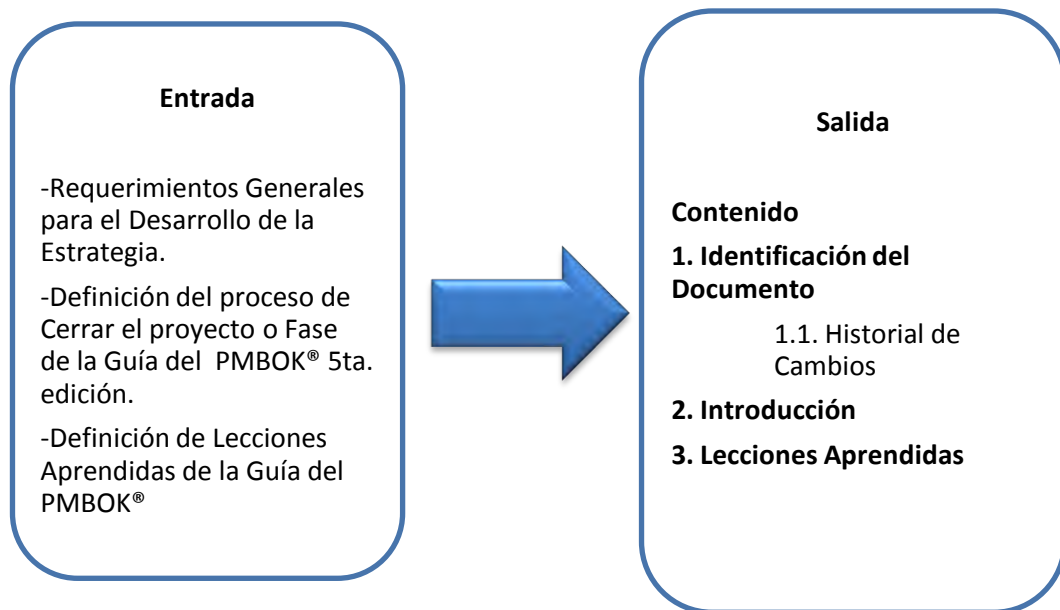
La Guía del PMBOK® describe a los documentos de Cierre como “Documentación formal que indica la terminación del proyecto o fase y la transferencia de los entregables completos del proyecto o fase a terceros.” (PMI, 2013, p. 104).

Además, se hace la validación del Alcance del proyecto y el contrato (si corresponde) para que se pueda asegurar que los requisitos del proyecto están completos, de no haberse concluido con el proyecto, se deberá de indicar las razones.

En el Anexo 11: Acta de Cierre del Proyecto, se muestra el documento donde se hace un resumen de todo lo desarrollado durante un proyecto, tal como su ejecución, impacto y entregables. Adicionalmente, se adjunta una sección donde estarán las firmas de los desarrolladores, los usuarios y el patrocinador del proyecto.

### 4.3.13. Lecciones Aprendidas

Es en la etapa del cierre del proyecto, en donde se documentan las lecciones aprendidas, que documentan conocimiento adquirido en todo el proyecto y considerado como útil en futuros esfuerzos.



#### Entrada:

- ▶ Requerimientos Generales para el Desarrollo de la Estrategia (Descritos en el apartado 4.1.2.)
- ▶ Definición del proceso de Cierre del proyecto o Fase de la Guía del PMBOK®

La Guía del PMBOK® menciona que el Cierre del Proyecto es “Un proceso que consiste en finalizar todas las actividades a través de todos los Grupos de Procesos de la Dirección de Proyectos para completar formalmente el proyecto o una fase del mismo. El beneficio clave de este proceso es que proporciona las lecciones aprendidas, la finalización formal del trabajo del proyecto, y la liberación de los recursos de la organización para afrontar nuevos esfuerzos.” (PMI, 2013, p. 100)

- ▶ Definición de Lecciones Aprendidas de la Guía del PMBOK®

En la Guía del PMBOK® define a las lecciones aprendidas como “El conocimiento adquirido durante un proyecto el cual muestra como se abordaron o deberían



abordarse en el futuro los eventos del proyecto, a fin de mejorar el desempeño futuro” (PMI, 2013, p. 551).

Por otro lado, Luna (2008), en sus Notas de Lecciones Aprendidas, define a éstas como “El conocimiento o entendimiento ganado por medio de la reflexión sobre una experiencia o proceso, o un conjunto de ellos. Esta experiencia o proceso puede ser positivo o negativo.”

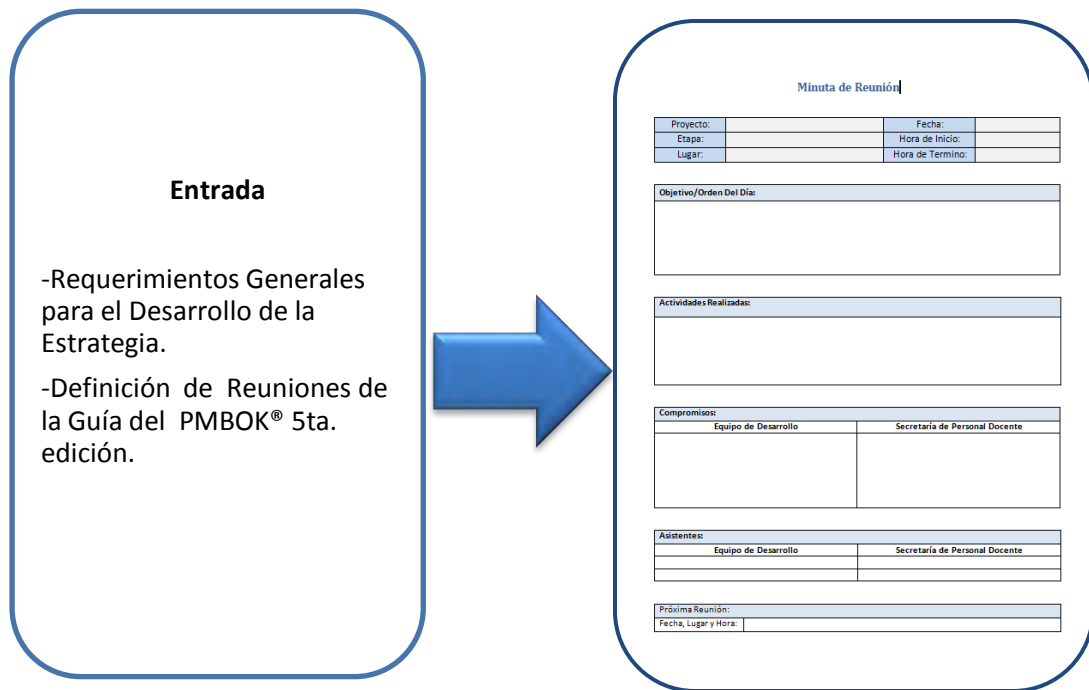
Además menciona que las lecciones aprendidas tienen que ser aplicables, válidas, significativas y permiten identificar factores de éxito, deficiencias, estrategias, resolución de problemas a través de nuevos cursos de acción y mejorar la toma de decisiones futura.

En el anexo 12: Lecciones Aprendidas: se presenta este documento, en donde se pueden enlistar y describir diversas situaciones que dieron origen a un aprendizaje que es considerado como lección aprendida del proyecto.

#### **4.3.14. Minutas de Reunión**

Con el fin de evitar olvidos, incomprensiones, malentendidos y conflictos entre los interesados que acuden a una reunión de trabajo, se elaboraran minutas para revisar decisiones y compromisos anteriormente tomados en el proyecto.

Las minutas funcionan como la memoria colectiva del proyecto y en ellas se encuentra un resumen de las decisiones y conclusiones de cada reunión que se haya llevado a cabo durante todo el proyecto (Esterkin, 2011).



**Entrada:**

- ▶ Requerimientos Generales para el Desarrollo de la Estrategia (Descritos en el apartado 4.1.2)
- ▶ Definición de Reuniones de la Guía del PMBOK®

La Guía del PMBOK® menciona que las reuniones se usan para discutir y abordar asuntos referentes al proyecto, además agrega que hay reuniones de tres tipos; de intercambio de información, tormenta de ideas, evaluación de opciones o diseño o de toma de decisiones.

De igual forma, agrega que (PMI, 2013) “Las reuniones deben prepararse con una agenda bien definida, con un propósito, con un objetivo y con un marco temporal y deben ser adecuadamente documentadas con actas de reunión y lista de acciones para realizar. Las actas de reunión deben ser almacenadas como se indique en el plan para la dirección de proyecto.” (p. 84).

En el Anexo 13: Minutas de Reunión se encuentra una plantilla del documento, que contiene apartados para los datos generales de la reunión, los objetivos de la misma, las actividades realizadas, los compromisos del Equipo de Desarrollo y de la SPD, así como la lista de asistentes.

#### 4.4. Desarrollo de los Artefactos basados en Scrum

Una vez que se desarrollaron los Artefactos basados en la Guía del PMBOK®, se pensó en el uso de prácticas y artefactos de un marco de trabajo ágil para la fase de ejecución del proyecto; para ello, se investigaron tres metodologías de desarrollo ágil, las cuales fueron SCRUM, *Extreme Programming* (XP) y OpenUP.

De la conclusión realizada en el segundo capítulo de esta tesis, en el apartado 2.5 Conclusiones, se determinó que la base para el propósito antes descrito, fuera SCRUM debido a que es un marco de trabajo breve, con tres roles y dos artefactos, abría la posibilidad de adaptar artefactos derivados del PMBOK®, además de que el concepto de *Sprint* nos ayudaría con entregas rápidas y podríamos observar como colabora un grupo bajo un enfoque ágil sin tener espacios de trabajo en común.

En el apartado 5.5. *Sprint* en el quinto capítulo, se da una explicación de lo que en esta Estrategia consideramos como Sprint, que difiere del concepto en Scrum.

Para una versión posterior se considera renombrar el *Sprint* para evitar confusiones con el concepto original en Scrum.

##### 4.4.1. Lista de Producto (*Product Backlog*)

La Guía de SCRUM, describe a este artefacto como “Una lista ordenada de todo lo que podría ser necesario para el producto, y es la única fuente de requisitos para cualquier cambio a realizarse en el producto” (Schwaber et al., 2013).

Sin embargo, basados en la Guía del PMBOK®, desarrollamos el documento de Requerimientos del Software, descrito en el apartado 4.3.4, el cual será utilizado para motivos de esta Estrategia como la única fuente de requerimientos del sistema.

##### 4.4.2. Lista de pendientes del *Sprint* (*Sprint Backlog*)

La Guía de SCRUM define a esta artefacto como “Conjunto de elementos de la lista de producto seleccionados para el *Sprint*, más un plan para entregar el Incremento del producto y conseguir el objetivo del *Sprint*. La lista de pendientes

del producto es una predicción hecha por el equipo de desarrollo acerca de qué funcionalidad formará parte del próximo incremento y del trabajo necesario para entregar esa funcionalidad” (Schwaber et al., 2013).

Por lo tanto, del documento de Requerimientos del Software, se van a extraer aquellos requerimientos que serán desarrollados en un determinado lapso de tiempo o *Sprint*. Estos requerimientos serán plasmados en otro documento para que todo el equipo de desarrollo tenga la misma información.

En el Anexo 14: Lista de pendientes del Sprint, se encuentra un modelo de esta lista, que contiene los requerimientos o funcionalidades que se han elegido para desarrollar en ese *Sprint*, las cuales contienen un identificador y se indica al responsable, así como el estado del requerimiento con referencia a su desarrollo, los cual puede ser: terminado, en proceso o en espera. Además, se tiene un apartado en donde se indican las horas estimadas de desarrollo.

#### **4.5. Desarrollo de los Artefactos basados en UML.**

No es el objetivo de esta Estrategia proporcionar una guía de aprendizaje del Lenguaje Unificado de Modelado, sin embargo, se incluye el tema debido a que es de gran importancia en el desarrollo del software y trabajos escritos para la titulación por Diseño de un sistema o proyecto para una organización.

##### **4.5.1. Mapa Conceptual**

Con el objetivo de constituir una guía de UML 2.0, se desarrolló un Mapa conceptual que contiene la definición de Lenguaje Unificado de Modelado, de modelo y la importancia de modelar, así como el desglosé de los componentes del Modelo Conceptual de UML:

- Bloques básicos de construcción:
  - Elementos
  - Relaciones
  - Diagramas

- Reglas de combinación
- Mecanismos comunes
  - Especificaciones
  - Adornos
  - Divisiones comunes
  - Mecanismos de extensibilidad

Cada apartado contiene su explicación y en la mayoría de los casos se incluyen ejemplos. El cuadro del Modelo Conceptual de UML 2.0 se encuentra en el apartado 5.9 Modelo Conceptual UML.

## 4.5.2. Especificación de Diagramas de Casos de Uso

Booch (2006) explica que un Caso de Uso especifica el comportamiento de un sistema describiendo secuencialmente las acciones que este ejecuta, con el fin de obtener un resultado observable para un actor. Gráficamente se representan como una elipse y denotan el comportamiento esencial de un sistema o subsistema.

Para esta Estrategia de Desarrollo, los Casos de Uso serán un punto clave para establecer el alcance del proyecto, el Equipo de Desarrollo tendrá que elaborar los Diagramas de Casos de Uso correspondientes y sus Especificaciones.

Una Especificación de Caso de Uso es una descripción detallada del flujo de la funcionalidad de sistema, para esta descripción se utiliza un documento sencillo con diferentes apartados que nos ayudan a detallar el Caso de Uso.

El Anexo 15: Especificaciones de Casos de Uso es una plantilla de apoyo para el propósito antes descrito. Este documento contiene los apartados de nombre del Caso de Uso, ID, actores, una descripción general, pre-

condiciones, flujo principal, flujo alternativo, flujo de excepción y post-condiciones.

#### 4.6. Desarrollo del Proceso de la Estrategia.

Durante la elaboración de los Artefactos, se identificaron las etapas del proceso de la Estrategia:

- **Preparación**

Existía la necesidad de un software, se tenía el apoyo del patrocinador del proyecto y además existía la presencia de una rol clave para el desarrollo del sistema, que era el Administrador General, una Pasante de la Licenciatura en Informática que estaba por terminar un módulo del SIGAD, tenía un amplio conocimiento de los procesos de la SPD, de programación y de bases de datos; su participación resultó clave para la toma de decisiones de los próximos proyectos de desarrollo de algunos módulos del SIGAD.

Adicionalmente, existió un vínculo con un docente de la Licenciatura, así como con estudiantes interesados en titularse por medio de tesis y desarrollo de un sistema o proyecto para una organización.

- **Inicio**

Una vez que se tuvo el interés de desarrollar módulos del SIGAD, se concluyó que se necesitaría la participación de alumnos de la Licenciatura en Informática, quienes además de desarrollar los módulos, liberarían de esta manera su servicio social.

- **Planeación**

Se creó un grupo de estudiantes de la Licenciatura en Informática, interesados en el proyecto, no se requirió un perfil en específico, sin embargo se requirió que tuvieran el porcentaje de créditos requeridos por la FCA para iniciar su servicio social.

Cada estudiante se encargó de su trámite correspondiente: servicio social y titulación, ya sea de servicio social o titulación, al mismo tiempo que

fueron capacitados en UML y en Codeigniter, un *framework* de PHP; todo esto gestionado por una académica de la Licenciatura en Informática, el Administrador general del proyecto y un estudiante con experiencia empresarial.

Una vez que el equipo de desarrollo fue capacitado, se les asignó un módulo del SIGAD, asimismo se les proporcionaron los artefactos desarrollados para esta Estrategia, descritos en el apartado 5.7 Artefactos.

- **Ejecución**

En esta fase, el equipo de desarrollo ya tenía claro que se había de hacer, estaban familiarizados con los procesos de la SPD y habían elaborado Casos de Uso y Modelo Conceptual.

Siguiendo la idea de adoptar prácticas de Scrum en la parte de Ejecución del proyecto, se tomó la decisión de que se eligiera un Caso de Uso para desarrollar en cada *Sprint*, de tal manera que habría tantos *Sprints* como casos de uso.

En cada *Sprint* se desarrollaron las acciones relacionadas al Caso de Uso elegido, calculando el tiempo de desarrollo en función de su complejidad acorde a la técnica de Puntos de Casos de Uso (Anexo 5: Costos del Proyecto) en acuerdo entre los desarrolladores y el Administrador general del Proyecto.

Una vez que se terminó el desarrollo, se hicieron las pruebas pertinentes y se hizo la integración del sistema, el resultado era mostrado al usuario si éste los validaba.

- **Seguimiento y Control**

Durante la planeación y ejecución del desarrollo de todos los módulos, el Administrador General se encargó de revisar todos los avances, hacer correcciones y ayudar en cuestiones técnicas a los equipos de desarrollo.

Asimismo, los Equipos de Desarrollo fueron responsables de hacer las actualizaciones pertinentes en todos sus documentos si éstas fueran necesarias.

En caso de que hubiera muchas dudas y complicaciones para que el trabajo fuera terminado, los desarrolladores solicitaban la ayuda del Administrador general del proyecto, así como a profesores de la Licenciatura en Informática o alumnos que tuvieran la experiencia, para así resolver dudas y continuar con el desarrollo.

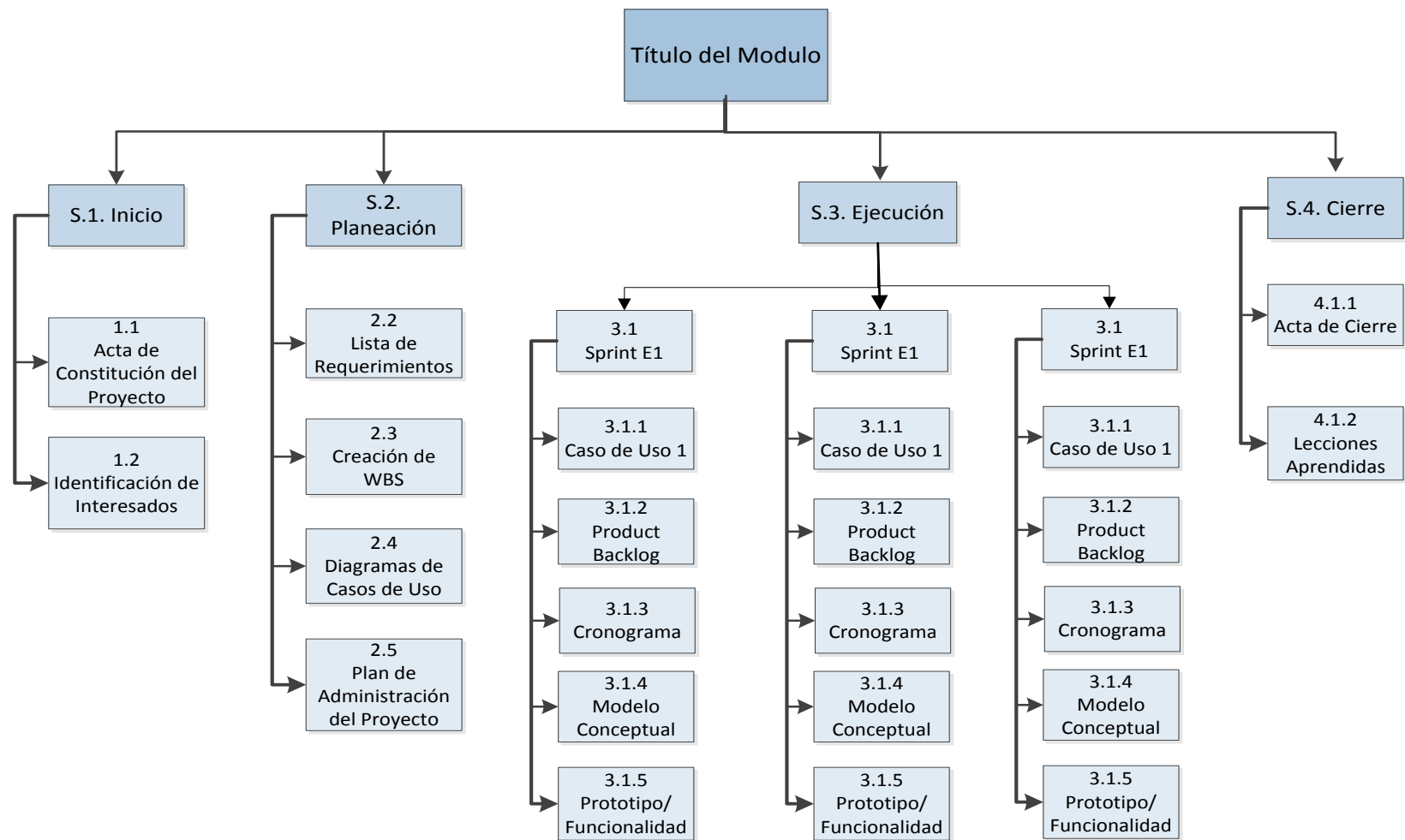
- **Cierre**

Una vez terminados todos los proyectos, había que cerrar los proyectos de manera formal, tal como se empezaron, para ello se utilizaron Actas de Cierre de proyecto en donde se especifica qué se hizo, cómo se hizo, y lo que se entregó al patrocinador.

Como conclusión general, los equipos de desarrollo hicieron sus Lecciones Aprendidas, con el fin de compartirlas con los demás equipos y documentarlas para desarrollos futuros.

Empatando el Proceso identificado y los Artefactos elaborados, se creó el siguiente EDT general, aplicable a todos los proyectos de desarrollo. Esta EDT constituye el eje de la Estrategia.





**Gráfico 4.1.** EDT general para proyectos bajo la Estrategia de Desarrollo de Sistemas de Información (Creación propia).

#### 4.7. Logotipo.

Con el propósito de proporcionar un distintivo a la Estrategia, se desarrolló un logotipo inspirado en la escultura *Coatl* de Helen Escobedo (1980), ubicada en el Espacio Escultórico de Ciudad Universitaria.



**Gráfico 4.2.** Logotipo de la Estrategia de Desarrollo de Sistemas de Información (Creación propia, 2014).

---

## **Capítulo 5.**

# **Estrategia para el Desarrollo de Sistemas de Información: Caso Secretaría De Personal Docente de la FCA, UNAM.**

---

## 5. Estrategia para el Desarrollo de Sistemas de Información

La Estrategia de Desarrollo de Sistemas de Información (EDSI) es una guía para la construcción de software que está basada en modelos, prácticas y estándares internacionales. Su finalidad es ayudar a los desarrollares con la administración de su proyecto, ofreciendo una serie de herramientas y proponiendo una forma de trabajo que se adapte a su disponibilidad.

### 5.1. ¿A quién va dirigida esta Estrategia de Desarrollo?

Esta estrategia de desarrollo va dirigida a estudiantes y egresados de la Licenciatura en Informática que deseen titularse desarrollando un sistema para la Secretaría de Personal Docente (SPD) de la Facultad de Contaduría y Administración de la UNAM, que:

- ▶ Estén organizados en equipos de máximo 6 personas
- ▶ No cuenten con un espacio de trabajo común

### 5.2. ¿En qué proyectos puede ser aplicado?

Esta Estrategia ha sido diseñada para proyectos pequeños de software que sean desarrollados en aproximadamente 6 meses, con un riesgo limitado y con un equipo de trabajo de máximo 6 personas.

### 5.3. Valores y Competencias

Con el fin de crear un ambiente de cordialidad entre los participantes del proyecto, así como facilitar el entendimiento entre los mismos, esta Estrategia contempla los siguientes valores y competencias:

- ▶ **Autoaprendizaje**

Los miembros del equipo de desarrollo deben de adquirir conocimientos y resolver problemas por sus propios medios, lo que los ayudará a mantenerse actualizados y profundizar más en temas de su área y lograr un aprendizaje independiente y no dirigido.

▶ **Trabajo en equipo**

Al trabajar en equipo, se fomenta el compañerismo, la solidaridad, la comprensión y se crea sinergia con el objetivo de colaborar rápida y eficazmente tomando en cuenta los puntos de vista y los conocimientos de otras personas para enriquecer el trabajo.

▶ **Responsabilidad**

Cumplir con las obligaciones que se tienen en el proyecto y considerar a los demás cuando se toma alguna decisión es de suma importancia para el buen transcurso del proyecto, lo que eleva la probabilidad de éxito del mismo.

▶ **Colaboración**

Aportar ideas para contribuir con el desarrollo del proyecto y tener la disposición para ofrecer ayuda en cualquier momento, son actividades que deben practicar todas las partes interesadas para generar un mayor entendimiento sobre el proyecto.

▶ **Confianza**

La confianza en sí mismos es una cualidad importante entre los miembros del equipo, ayudará a que se desenvuelvan con mayor facilidad y expresen situaciones o ideas para un mejor desarrollo del proyecto. Para ello deberá de existir confianza entre todos los miembros del equipo, así como con el usuario, asesor y administrador.

▶ **Valentía**

Preguntar para despejar cualquier duda que se tenga o cometer errores y aceptarlos suelen ser acciones que pueden provocar miedo o inseguridad, por lo tanto, se debe tener el coraje para enfrentar estas situaciones u otras que puedan generar problemas o representen obstáculos para el buen desempeño de todos los involucrados en el proyecto.

## 5.4. Roles

La Estrategia de Desarrollo de Sistemas de Información contempla los siguientes roles:

Rol	Descripción	Actividades
<p><b>Asesor</b></p>	<p>Académico de la Facultad de Contaduría y Administración de la UNAM que cuenta con la autorización para fungir como asesor de tesis o trabajo de diseño de un sistema o proyecto.</p>	<ul style="list-style-type: none"> <li>▪ Orienta al alumno(s) acerca cómo desarrollar su proyecto.</li> <li>▪ Mantiene constante comunicación con el Administrador General del proyecto para medir el avance y resultados del Equipo de Desarrollo.</li> <li>▪ Coordina reuniones entre el Equipo de Desarrollo y el usuario.</li> <li>▪ Busca a Asesores Externos en caso de que el Equipo de Desarrollo lo necesite.</li> </ul>
<p><b>Alumno o pasante de la Licenciatura en Informática</b></p>	<p>Estudiante o pasante de Licenciatura en Informática de la Facultad de Contaduría y Administración de la UNAM que elige el desarrollo de un sistema o proyecto como opción de titulación.</p>	<p>Un alumno puede desempeñar diversas actividades, tales como:</p> <ul style="list-style-type: none"> <li>▪ Análisis y Diseño del Sistema.</li> <li>▪ Crear de Bases de Datos.</li> <li>▪ Diseñar la Interfaz de Usuario.</li> <li>▪ Programar el Sistema.</li> <li>▪ Desarrollar la documentación del sistema.</li> <li>▪ Generar manuales de usuario.</li> </ul>

<p><b>Usuario</b></p>	<p>Coordinador de alguno de los diferentes Departamentos de la Secretaría de Personal Docente que conoce los procesos de la organización.</p>	<ul style="list-style-type: none"> <li>▪ Proporciona los requerimientos del sistema.</li> <li>▪ Evalúa el producto de forma periódica para asegurar que se está haciendo correctamente.</li> </ul>
<p><b>Patrocinador</b></p>	<p>Jefe o responsable de la Secretaría de Personal Docente.</p>	<ul style="list-style-type: none"> <li>▪ Comunica a los coordinadores de área acerca del proyecto y autoriza que se designe tiempo laboral para el desarrollo del proyecto.</li> </ul>
<p><b>Administrador General del proyecto.</b></p>	<p>Servidor Social, Pasante de la Licenciatura en Informática o miembro de la Secretaría de Personal Docente que tenga conocimientos técnicos en programación y bases de datos, además de un panorama global de los procesos de la Secretaría.</p>	<ul style="list-style-type: none"> <li>▪ Se encarga de la integración del sistema.</li> <li>▪ Evalúa los avances del Equipo de Desarrollo.</li> <li>▪ Orienta en cuestiones técnicas.</li> </ul>
<p><b>Equipo de Desarrollo</b></p>	<p>Se trata de dos o más Alumnos que desarrollen un proyecto. Serán supervisados, evaluados y ayudados por el Asesor y el Administrador General del proyecto.</p>	
<p><b>Líder de proyecto</b></p>	<p>Miembro del Equipo de Desarrollo encargado de dar seguimiento al proyecto designado y al resto del equipo de trabajo.</p>	<ul style="list-style-type: none"> <li>▪ Tener una mayor comunicación con el Administrador General del Proyecto acerca del avance y las actividades referentes al</li> </ul>

		<p>proyecto en curso.</p> <ul style="list-style-type: none"> <li>▪ Informar de manera puntual todos los acuerdos y actividades al resto del equipo de desarrollo.</li> <li>▪ Controlar el desarrollo y registro de los entregables del proyecto.</li> <li>▪ Dar seguimiento a las actividades necesarias para cubrir con los objetivos del proyecto.</li> <li>▪ Acordar reuniones del equipo de desarrollo con el Administrador General, Asesor Externo, Usuario o Patrocinador.</li> </ul>
<p><b>Asesor Externo</b></p>	<p>Un profesional experto en un tema particular, por ejemplo; UML o el <i>framework</i> CodeIgniter.</p>	<ul style="list-style-type: none"> <li>▪ Provee al equipo de Desarrollo de conocimiento requerido a través de clases o material de consulta.</li> <li>▪ Evalúa, corrige o aprueba el trabajo que el Equipo de Desarrollo ha completado hasta el momento.</li> </ul>



## 5.5. Sprint

Un Sprint es un periodo de tiempo en donde se desarrolla una funcionalidad del sistema basado en un Caso de Uso. Su duración estará a consideración del Equipo de Desarrollo y Líder de Proyecto, aunque se recomienda que no exceda las 6 semanas.

Durante este periodo de Tiempo, el Equipo de Desarrollo deberá de programar una funcionalidad del sistema o módulo que se esté desarrollando, de manera que este avance represente algún valor para el Usuario o para el proyecto.

## 5.6. Fases

La Estrategia de Desarrollo de Sistema de Información está constituida por las siguientes fases:

### I. Preparación

Durante esta etapa, el Alumno o Pasante de la Licenciatura en Informática que desea hacer su servicio social o titularse por la modalidad de Diseño de un Sistema o Proyecto para una Organización, conoce de forma general los procesos y las necesidades con el fin de crear un panorama acerca del trabajo que se debe de hacer para la SPD a fin, de decidir si es en la Secretaría donde desea desarrollar sus actividades académicas antes mencionadas.

Una vez que esté seguro de adquirir el compromiso con la SPD, deberán de consultar los procesos, fechas y trámites señalados por la Facultad de Contaduría y Administración de la UNAM para iniciar el registro y hacer formal su participación en el proyecto. En el caso de los Alumnos o Pasantes que se quieran titular, deberán hacer del conocimiento del proyecto a su Asesor para que éste lo evalúe y apruebe.

### II. Inicio

Una vez que los trámites han sido iniciados, el alumno o pasante de la Licenciatura, tiene que conocer de manera detallada los antecedentes del sistema desarrollado para la SPD, como el lenguaje con el que fue programado, los módulos que han sido desarrollados y los módulos que faltan por desarrollar.

El Administrador General del Proyecto será el encargado de asignar el módulo que se desarrollará, construir el Grupo de Desarrollo y conocer las habilidades y conocimientos de

sus integrantes, proporcionar la información necesaria sobre todos los procesos del módulo en cuestión, además de crear un puente de comunicación entre el Equipo de Desarrollo, el Usuario y el Patrocinador del proyecto.

El Equipo de Desarrollo puede estar constituido mínimo por dos personas, con el fin de que una de ellas adopte el rol de Líder de Proyecto.

Esta etapa también se caracteriza por darle Inicio formal al proyecto mediante Artefactos como el Acta Constitutiva del Proyecto. Para ello, el Equipo de Desarrollo debe de tener un entendimiento de qué se va a hacer, cómo se va a hacer y en cuánto tiempo es que se va a desarrollar el Sistema, con base en la información recibida del Patrocinador, Usuario, Asesor y Administrador General del Proyecto. Además, se estimará el tiempo de desarrollo basado en los conocimientos, habilidades y la necesidad de asesoramiento de algún tema en particular que tenga el Equipo de Desarrollo.

Es importante aclarar que son estimaciones iniciales y por tanto, irán cambiando durante todo el proyecto, pero son importantes hacerlas para crear un entendimiento entre todas las partes interesadas.

- ▶ En esta etapa se utilizan los siguientes Artefactos:
  - ▶ Acta de Constitución del Proyecto
  - ▶ Documento de Análisis de Interesados

### III. **Planeación**

Implica un estudio a profundidad sobre el Sistema a desarrollar y engloba cuestiones de Administración de Proyectos, así como el Diseño y Análisis del sistema. Durante esta etapa se harán estimaciones más certeras acerca del tiempo y se definirán los riesgos, el alcance, los costos, la calidad o la comunicación. De igual forma, se hará un plan general de ejecución del proyecto, lo que marcará el camino a seguir durante el mismo.

Lo anteriormente mencionado, se realizará con base en el número de Casos de Uso que implique el Sistema a desarrollar, ya que esto nos dará indicios de su tamaño y por consecuencia nos proporcionará el alcance del proyecto.

Es muy importante que todo el Equipo de Desarrollo participe durante esta fase en conjunto con el Administrador General, que todos aporten ideas y que se tomen en cuenta

las necesidades y expectativas del Usuario y del Asesor para tomar decisiones que favorezcan a todos los interesados y estar de acuerdo antes de empezar con el desarrollo del sistema.

- Los Artefactos utilizados en esta etapa son:
  - ▶ Plan para la Administración del Proyecto.
  - ▶ Lista de requerimientos
  - ▶ Estructura de Desglose de Trabajo (EDT) también conocido como *Work Breakdown Structure* (WBS).
  - ▶ Casos de Uso
  - ▶ Cronograma del Proyecto

#### IV. Ejecución

Cuando se llega a esta etapa se sabe lo que hay que hacer y se comienza con la ejecución de todos los planes y la programación del sistema.

Esta etapa será ejecutada mediante *Sprints*. Para cada *Sprint*, el equipo de Desarrollo elegirá un Caso de Uso basado en las demandas del Usuario y se tomarán en cuenta las opiniones del Administrador General y el Asesor.

Una vez elegido el Caso de Uso, se hará una Lista de Pendientes del *Sprint*, el cual será enviado a todos los miembros del Equipo y mostrará lo que hay que hacer, así como el tiempo que se tiene para hacerlo. En esta Lista, estarán contenidas las actividades como el desarrollo o actualización del Modelo Conceptual.

Cada *Sprint* tendrá una fecha definida y una vez que se termine el plazo acordado, el resultado será mostrado y valuado por el Administrador General y el Usuario. Este último será quien dé su aprobación y haga correcciones y una vez hechas, entonces se pasará al siguiente *Sprint* donde, de igual manera se elegirá un Caso de Uso, se actualizará el Modelo Conceptual, se hará una Lista de Pendientes que contendrá actividades que hayan quedado inconclusas o en espera del *Sprint* anterior.

- Los Artefactos que se ocuparán en esta etapa son:
  - ▶ Casos de Uso

- ▶ Lista de Pendientes del *Sprint*
- ▶ Modelo Conceptual
- ▶ Cronograma del *Sprint*

## V. Seguimiento y Control

El seguimiento y control empieza durante la ejecución. Los objetivos son:

- ▶ Cuidar que todo se realice conforme a lo planeado.
- ▶ Hacer cambios y ajustes requeridos en el Plan de Administración de proyecto y en el resto de los Artefactos.
- ▶ Llevar el seguimiento de los interesados en el proyecto con el fin de asegurarse que todos estén participando y sus solicitudes y expectativas estén siendo tomadas en cuenta.
- ▶ Se hacen reuniones desde la planeación hasta el cierre de forma regular entre el Equipo de Desarrollo y el Administrador General, así como con el Usuario.
- ▶ Los Artefactos que se usan durante todo el seguimiento del Proyecto son:
  - ▶ Cronograma del Proyecto o *Sprint*
  - ▶ Minutas de Reuniones

## VI. Cierre

Cuando se han ejecutado el número de *Sprints* suficientes para completar el Sistema, se han integrado todas las partes y se han hecho las pruebas suficientes para garantizar el correcto funcionamiento del Sistema, así como completar los documentos del mismo, se cierra formalmente el proyecto mediante un Acta de Cierre del mismo.

Adicionalmente, se hace un análisis global de la ejecución del proyecto con fin de evaluar los aspectos negativos y positivos que sucedieron en el transcurso del proyecto y aportar información a una base del conocimiento sobre el proyecto.

- ▶ Los Artefactos que se usan durante el Cierre son:
  - ▶ Acta de Cierre del Proyecto
  - ▶ Lecciones Aprendidas

## 5.7. Artefactos

Los Artefactos son documentos que nos ayudarán en la ejecución de nuestro proyecto, en ellos estará contenida toda la información del mismo, nos servirán de referencia durante todo el proyecto y será nuestra evidencia del mismo.

### 5.7.1. Bitácora de Trabajo

Una bitácora es un cuaderno de trabajo donde se plasman de forma cronológica los avances, bocetos, información, resultados o acuerdos del proyecto.

Cada miembro del Equipo de Desarrollo debe de hacer uso de una bitácora para hacer sus anotaciones personales referentes al proyecto, con el propósito de concentrar todas las notas y hacer más fácil el acceso a ellas.

Con el uso de esta Bitácora se pretende que cada miembro del Equipo tenga un medio donde escribir sus ideas, actividades y notas durante el curso del proyecto.

### 5.7.2. Acta de Constitución del Proyecto

Es un documento que contiene los aspectos generales del proyecto, tales como el propósito, los objetivos, los criterios de éxito, requerimientos, restricciones, supuestos, exclusiones, alcance, interesados en el proyecto y riesgos.

#### ¿Para qué sirve?

Da inicio de manera formal al proyecto, comunicándolo a todos los interesados en él.

#### ¿Quiénes participan en su desarrollo?

El Líder del proyecto será el encargado de la elaboración del documento con ayuda y aprobación del Equipo de Desarrollo, ya que en conjunto se tiene que delimitar el contenido del Acta de Constitución, que será nuestra referencia durante todo el proyecto. Posteriormente, el contenido del Acta será verificado por el Administrador General para ser entregado al Usuario, Patrocinador y Asesor.

## ¿Qué se necesita para desarrollarlo?

- Los manuales de procesos de la Secretaría de Personal Docente.
- Haber tenido una reunión previa con el Usuario para saber los requerimientos y expectativas.

En el Anexo 1: Acta de Constitución del Proyecto se encuentra un ejemplo de esta Acta del módulo de Estímulos, se observa que el documento contiene un resumen del proyecto, así como su propósito y descripción.

### 5.7.3. Identificación de Interesados

Se trata de un documento donde registramos a las personas, grupos o la organización que se verá afectada con el resultado del proyecto o durante el desarrollo de éste.

#### ¿Para qué sirve?

Nos servirá para darnos cuenta del grado de interés, disposición, expectativas e impacto dentro del proyecto que tendrán los Interesados con el fin de contemplar un enfoque apropiado para cada uno y maximizar su participación de acuerdo a su disposición. Además, recabaremos su información de contacto que será de utilidad en el transcurso del proyecto.

#### ¿Quiénes participan en su desarrollo?

Basta con un integrante del Equipo de Desarrollo, que recabe la información, para después hacerle saber a los demás miembros del equipo que el documento está completo, y en el lugar en donde se acordó que sería almacenado, de tal forma que todos tenga la información de todos los interesados en el proyecto y ésta pueda ser utilizada en cualquier momento.

## ¿Qué necesito para desarrollarlo?

- Haber tenido una reunión previa con el usuario

En el Anexo 2: Identificación de Interesados se encuentra un ejemplo de este documento, que integra información como roles, datos de contacto, motivos y expectativas así como el impacto en el proyecto de cada uno de los interesados.

### 5.7.4. Plan para la Administración del Proyecto

Un plan para la Administración de Proyecto es una herramienta que nos ayudará a cumplir de manera efectiva con los objetivos del proyecto.

Para esta Estrategia de Desarrollo, este plan contempla:

- Definición de actividades (Alcance)  
Donde se enlistan aquellas actividades que se tienen que llevar a cabo para completar el trabajo del proyecto.
- Costos  
Se utiliza el método de Puntos de Casos de Uso para calcular el trabajo en el proyecto y así determinar su costo.
- Recursos Humanos  
Se enlistan los roles o perfiles que sean necesarios para el proyecto, así como planes de capacitación si se diera el caso.
- Riesgos  
Se hace un análisis acerca de los riesgos que podrían presentarse durante el proyecto, se determina su probabilidad e impacto, así como un pequeño plan en caso de que se llegaran a suscitar los eventos e impacten negativamente el curso del proyecto.
- Calidad  
Este se va a dividir en dos partes, la primera de ellas consiste en la calidad de nuestro proyecto a nivel interno; para ello se han listado una serie de lineamientos que nos permitirán tener en orden nuestra documentación de administración del

proyecto, modelo conceptual, casos de uso, archivos de código fuente y bases de datos.

La segunda parte estará a cargo del Usuario, quien evaluará los resultados del desarrollo.

- **Comunicación**

La comunicación entre todos los interesados del proyecto es muy importante para el éxito del mismo, es por ello que con base en la disponibilidad de horario de cada uno, se desarrollará un plan para la comunicación presencial y no presencial entre los integrantes del equipo de desarrollo y del equipo de desarrollo con el usuario.

### **¿Para qué sirve?**

Nos ayuda a visualizar aspectos importantes dentro del proyecto, como el alcance y a determinar las acciones para llevar a cabo en cuestión de Recursos Humanos, Costos, Riesgos, Calidad y Comunicación.

De manera general, nos va a ayudar a completar el trabajo necesario que se requiere hacer durante el proyecto, dentro de los límites establecidos.

### **¿Quiénes participan en su desarrollo?**

Es de vital importancia que todo el equipo participe, puesto que durante este proceso se toman decisiones importantes y las opiniones de todos son significativas, aunque el responsable de que los documentos será el Líder de Proyecto. Sin embargo, las cuestiones relacionadas con los Recursos Humanos, la Calidad y la Comunicación estarán a cargo del Administrador General del proyecto, puesto que para esta Estrategia se considera que estas cuestiones son comunes para todos los proyectos.

### **¿Qué necesito para desarrollarlo?**

Además de reuniones entre los Equipos de Desarrollo y el Administrador General, esta Estrategia proporciona ejemplos y guías para cada uno de los documentos que se deseen incluir en el Plan:

En este documento se enlistan las actividades necesarias para llegar a un objetivo o completar una parte del proyecto. Las actividades tienen que ser descritas y



secuenciadas, además se indica la dependencia con otras actividades, su duración y el responsable de dicha actividad.

- Anexo 5: Costos del Proyecto

El método de Puntos de Casos de Uso, consiste en hacer un análisis de cada uno de los Casos en función de sus flujos principales, hacer una estimación sobre la dificultad técnica y así poder determinar cuánto trabajo se habrá de realizar para programar el Caso de Uso. Para calcular el costo del proyecto, se empleó este método.

En el Anexo 5: Costos del Proyecto se explica a detalle cómo hacer el cálculo.

- Anexo 6: Recursos Humanos. Este documento describe los perfiles que se requieren para composición y capacitación del Equipo de Desarrollo, es utilizado generalmente por el Administrador General del proyecto para llevar un registro integral de todo el equipo de desarrollo.

El Anexo 6: Recursos Humanos se encuentra una plantilla del documento para lograr los objetivos antes mencionados.

- Identificar los riesgos es el proceso donde se determinan aquellas acciones o eventos que puedan afectar el proyecto y sus características.

Este documento nos ayudará a enlistar los posibles riesgos, así como su análisis cualitativo, empleando una matriz de probabilidad e impacto. La plantilla de este documento se encuentra en el Anexo 8.

- Anexo 9: Calidad

Con el propósito de cumplir con el objetivo del proyecto, el cual es desarrollar un sistema que cumpla que las expectativas del usuario cumpliendo

- Anexo 10: Comunicación

Este documento contiene un directorio, información acerca de la ubicación de la documentación, así como apartados donde se describen los planes de reuniones presenciales y comunicación por medios electrónicos y virtuales.

### 5.7.5. Lista de Requerimientos

La recopilación de los Requerimientos es de suma importancia ya que delimitan el alcance del proyecto, para ello debemos determinar, documentar y gestionar las necesidades de las partes interesadas y los requisitos para cubrir los objetivos del proyecto.

#### ¿Para qué sirve?

Esta Lista de Requerimientos delimitará el alcance, es decir, nos va a indicar qué se tiene que hacer durante todo el proyecto.

En este documento se va plasmar con detalle las funciones que ejecutarán el sistema, su dependencia con otros sistemas, así como el aspecto de la interfaz gráfica del Usuario.

#### ¿Quiénes participan en su desarrollo?

Es imprescindible que participe todo el Equipo de Desarrollo en conjunto con el Usuario, pues éste será el encargado de proporcionar todos los Requerimientos del sistema y los miembros del Equipo de Desarrollo tienen que resolver todas sus dudas con respecto a los procesos de la organización que se van a automatizar.

#### ¿Qué necesito para desarrollarlo?

- Los manuales de los procesos de la SPD
- Reuniones con el usuario
- Uso de técnicas como lluvia de ideas o cuestionarios que ayuden al Usuario a explicar sus ideas y su visión, de tal manera que haya un entendimiento mutuo.
- En el Anexo 3: Requerimientos del Proyecto hay un ejemplo de este documento, en el cual se hace una descripción global del sistema para detallar y priorizar los requerimientos funcionales y no funcionales.

### 5.7.6. Estructura de Desglose de Trabajo

La Estructura de Desglose de Trabajo (EDT) o *Work Breakdown Structure* (WBS) es una descomposición jerárquica, basada en los entregables del trabajo que se deben ejecutar

para lograr los objetivos del proyecto, proporcionando una guía para medir el avance en el desarrollo del producto.

### **¿Para qué sirve?**

Esta Estructura de Desglose de Trabajo, nos va a permitir “partir” un paquete de trabajo en partes más pequeñas, de esta manera tenemos una idea más precisa de qué es lo que tenemos que hacer y cuánto nos falta para completar ese paquete de trabajo.

### **¿Quiénes participan en su desarrollo?**

Es deseable que todo el Equipo de Desarrollo participe en su elaboración y compartan opiniones, ya que esta Estructura de Desglose, funcionará como guía durante todo el proyecto.

### **¿Qué necesito para desarrollarlo?**

Hay distintas maneras y estilos para hacer un EDT, en el Anexo 4: Creación del WBS/EDT, se encuentra una sencilla guía que ayudará con la creación de la Estructura. En ella se explica qué es, cómo se desarrolla y algunos ejemplos.

## **5.7.7. Diagrama de Casos de Uso/Especificación de Casos de Uso**

Un caso de uso es una descripción de un conjunto de acciones que ejecutará un sistema para producir un resultado con valor para un usuario, un diagrama es un conjunto de casos de uso, actores y sus relaciones y muestra el comportamiento de un sistema, subsistema o clase.

Una especificación de casos de uso describe la interacción del usuario con el sistema, a modo de narración y detalla las tareas que el actor realiza, la información que ingresa o recibe, las condiciones del sistema y los flujos principales, alternos y de excepción.

## ¿Para qué sirve?

El diagrama y la especificación de casos de uso sirven para detallar, documentar y visualizar el comportamiento de un elemento, facilitar la comprensión de los sistemas.

Además, el número de casos de uso nos ayudará en gran medida a calcular el alcance de nuestro proyecto y determinará el número de *Sprints* que necesitaremos emplear para desarrollar todo nuestro sistema.

## ¿Quiénes participan en su desarrollo?

Todos los miembros del equipo, ya que es imprescindible que todos estén involucrados por que los casos de uso delimitan y detallarán que tiene que hacer el sistema. Muy probablemente también se necesite la participación de algún Asesor Externo que verifique los diagramas.

## ¿Qué necesito para desarrollarlo?

En la bibliografía recomendada se encuentran un listado de materiales que serán de utilidad para el desarrollo de los diagramas de casos de uso, además, el Anexo 15: Especificaciones de Casos de Uso es una plantilla donde se describe el caso de uso con los actores participantes, las pre-condiciones y post-condiciones del sistema, así como los flujos principal, el alternativo y de excepción del caso de uso.

### 5.7.8. Modelo Conceptual UML

UML es un lenguaje de modelado gráfico utilizado para visualizar, construir, documentar y especificar los artefactos de un sistema software, los diagramas de UML representan de manera gráfica un conjunto de elementos y relaciones para visualizar un sistema desde diferentes perspectivas.

Existen los diagramas de clases, artefactos, componentes, objetos, de interacción, de estados, actividades, de despliegue, de casos de uso y de paquetes, todos ellos representan una proyección de un sistema y dependiendo de la perspectiva con la que queramos ver al

sistema, será el tipo de diagrama de desarrollaremos con la intención de tener un mayor entendimiento sobre el sistema y facilitar su desarrollo.

### **¿Quiénes participan el su desarrollo?**

Todos los integrantes del equipo de desarrollo en conjunto con un Asesor Externo decidirán que diagramas elaborarán y los realizaran con apego a las especificaciones de la Guía del Usuario del Lenguaje de Modelado Unificado.

### **¿Qué necesito para desarrollarlo?**

En esta Estrategia de Desarrollo de Sistemas, hemos elaborado un cuadro con los conceptos claves acerca del modelo conceptual de UML a manera de guía ya que es importante tener los conocimientos esenciales para desarrollar diagramas de UML, debido a que en ocasiones algunos conceptos son malinterpretados o poco entendidos y el diagrama resultante puede ser erróneo o poco funcional para el desarrollo del sistema.

#### **5.7.9.Lista de Pendientes del *Sprint***

Una vez que se han desarrollado los diagramas de UML, especialmente los diagramas de casos de uso, el Equipo junto con el Administrador General decidirán qué caso de uso se desarrollará durante el *Sprint* y delimitarán las actividades a realizar para completar la funcionalidad del sistema y delegar actividades de tal forma que se tendrá una lista de pendientes.

### **¿Para qué sirve?**

Funcionará como referencia para cada uno de los integrantes del equipo de desarrollo sobre el trabajo que se tiene que realizar para completar el objetivo del *Sprint*, así como para medir su avance durante el tiempo que se haya acordado que durará el *Sprint*.

## ¿Quiénes participan el su desarrollo?

Todos los miembros del Equipo de Desarrollo participan en el desarrollo de esta Lista, ya que es el Equipo quienes tienen que completar el trabajo de la lista.

## ¿Qué necesito para desarrollarlo?

El Anexo 14: Lista de pendientes del *Sprint* hay una plantilla muy sencilla donde se pueden enlistar todas las actividades que se tienen que hacer para completar el trabajo del *Sprint*.

### 5.7.10. Cronograma del *Sprint*

Una vez que esté completa la Lista de Pendientes del *Sprint*, será desarrollado un cronograma donde serán reflejadas la duración de estas actividades, la fecha en la que deberán de estar finalizadas y dependencia con otras actividades.

## ¿Para qué sirve?

Un cronograma nos ayudará a tener un mayor control del tiempo en el que se deberán llevar a cabo las actividades en el *Sprint*, con el fin de garantizar que todas las actividades sean desarrolladas en el plazo previsto.

## ¿Quiénes participan el su desarrollo?

El Líder del Proyecto será en encargado de desarrollar el cronograma, mismo que tendrá que ser enviado al Equipo de Desarrollo, para todos estén enterados del tiempo acordados.

## ¿Qué necesito para desarrollarlo?

En el Anexo 7. Cronograma, se explica qué es un cronograma y el Diagrama de Gantt, además en el Software Recomendado, se enlistan una serie de programas para elaborar estos artefactos.

### 5.7.11. Acta de Cierre del Proyecto

Cuando el Equipo de Desarrollo ha terminado con la construcción del sistema y éste ha sido evaluado y aceptado por el Usuario, se considera que el proyecto ha concluido. Cuando eso ocurre hay que cerrar el proyecto utilizando el documento Acta de Cierre del Proyecto.

#### ¿Para qué sirve?

Indica que el proyecto o fase ha sido concluido. Este documento valida que los requisitos del usuario han sido cubiertos por el Equipo de Desarrollo, que los acuerdos han sido cumplidos y que el Patrocinador está de acuerdo con el resultado del proyecto.

#### ¿Quiénes participan en su desarrollo?

El Administrador del proyecto con aprobación del Administrador General.

#### ¿Qué necesito para desarrollarlo?

El Anexo 11: Acta de Cierre del Proyecto es una plantilla del documento donde se hace un resumen del desarrollo del proyecto, su ejecución, impacto y entregables. Adicionalmente, se adjunta una sección donde estarán las firmas de los desarrolladores, los usuarios y el patrocinador del proyecto.

### 5.7.12. Lecciones Aprendidas

La etapa del cierre del proyecto es en donde se documentan las lecciones aprendidas, que representan el conocimiento adquirido en todo el proyecto considerado como útil en futuros esfuerzos.

#### ¿Para qué sirve?

Documentar las lecciones aprendidas, puede traer los siguientes beneficios:

- Aprovechamiento de experiencias
- Evitar sobrecostos
- Optimizar recursos

- Recopilar información que solo se encuentra en la mente de los involucrados en el proyecto.

Al es conocimiento que se ha adquirido durante todo el proyecto de forma pragmática, se convierten en experiencias que le pueden servir a alguien más en un futuro y evitar que se cometan los mismos errores o replicar aciertos.

### **¿Quiénes participan en su desarrollo?**

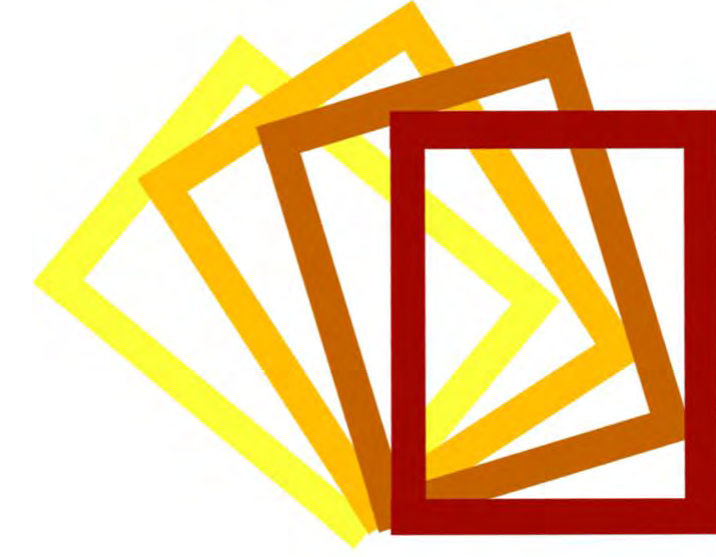
Todos los miembros del Equipo participan, cada uno compartiendo sus experiencias y opinando acerca del curso de proyecto.

### **¿Qué necesito para desarrollarlo?**

Las lecciones aprendidas suelen hacerse al final del proyecto, y pueden ser generadas en una reunión en la que estén presentes todos los miembros del equipo o mandando sus lecciones de forma electrónica a un responsable que se encargue de concentrar toda la información en un documento para que sea enviado a los demás, miembros del Equipo y posteriormente archivado para futuros proyectos.

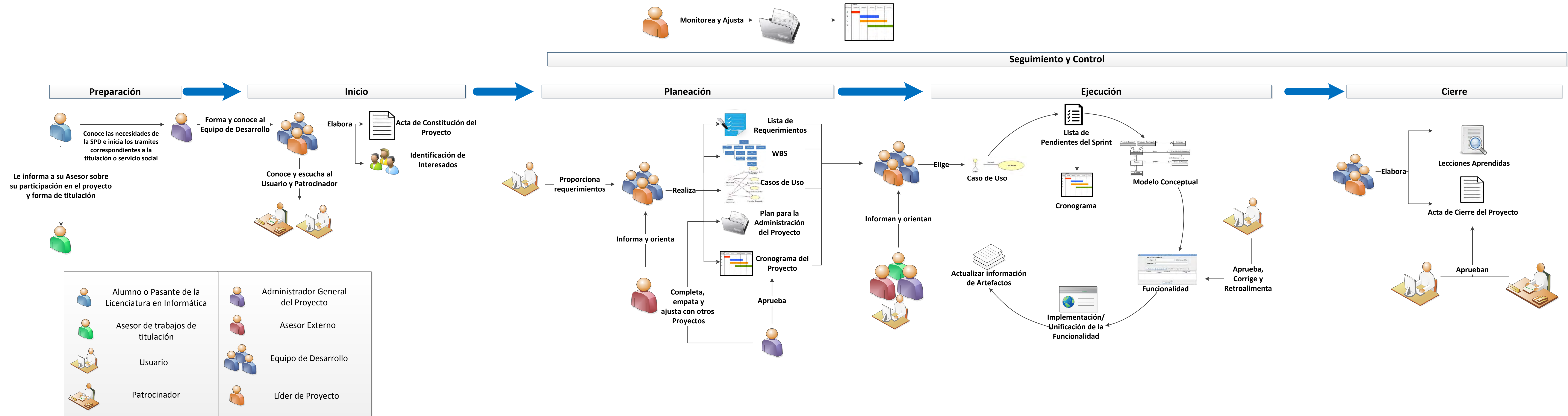
En el Anexo 12: Lecciones Aprendidas hay un ejemplo de este documento, donde de manera sencilla se enlistan las situaciones que dieron origen a algún conocimiento que puede ser considerado como una lección aprendida.





# ESTRATEGIA PARA EL DESARROLLO DE SISTEMAS DE INFORMACIÓN

## PROCESO GENERAL





## 5.9. Modelo Conceptual UML

### UML, ¿Qué es?



Es un lenguaje de modelado gráfico que se utiliza para visualizar, constituir, documentar, y especificar los artefactos de un sistema software, representando los planos de un sistema, elementos conceptuales, procesos de negocio, funciones del sistema o elementos concretos como las clases en un lenguaje de programación, una base de datos o componentes software utilizables.



### ¿Qué es un modelo?

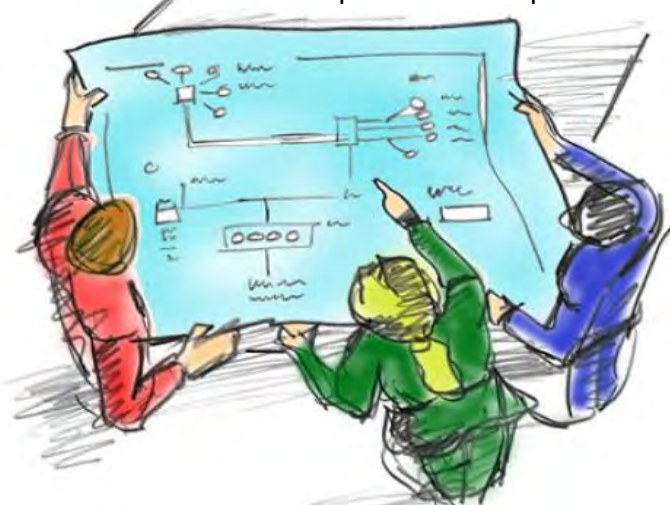
Es una simplificación de la realidad. Proporciona los planos de un sistema, pueden involucrar planos detallados o generales que ofrece una visión global del sistema en consideración.

### ¿Por qué modelamos?

Para comprender mejor el sistema que estamos desarrollando.

A través del modelado, conseguiremos cuatro objetivos:

- Nos ayudan a **visualizar** cómo es o queremos que sea un sistema.
- Nos permiten **especificar** la estructura o el comportamiento de un sistema.
- Nos proporcionan **plantillas** que nos guían en la **construcción** de un sistema.
- Documentan** las decisiones que hemos adoptado.



## Bloques básicos de construcción

**Elementos**  
Son abstracciones que constituyen los ciudadanos de primera clase en un modelo; las relaciones ligan estos elementos entre sí; los diagramas agrupan colecciones interesantes de elementos.

**Estructurales**  
Son los nombres de los modelos UML. En su mayoría son las partes estáticas de un modelo, y representan conceptos o cosas materiales. Colectivamente, los elementos estructurales se denominan *clasificadores*.

**De Comportamiento**  
Son las partes dinámicas de los modelos UML. Éstos son los verbos de un modelo, y representan comportamiento en el tiempo y espacio.

**De Agrupación**  
Son las partes organizativas de los modelos de UML. Éstos son las cajas en las que puede descomponerse un modelo.

**De Anotación**  
Son las partes explicativas de los modelos UML. Son comentarios que se pueden aplicar para describir, clarificar y hacer observaciones.

**Relaciones**  
Ligan elementos, se utilizan para escribir modelos bien formados.

**Dependencia**  
Relación semántica entre dos elementos, en la cual un cambio a un elemento puede afectar a la semántica del otro elemento.

**Asociación**  
Relación estructural entre clases que describe un conjunto de enlaces, los cuales son conexiones entre objetos que son instancias de clases.

**Generalización**  
Es una relación de especialización/generalización en la cual el elemento especializado (hijo) se basa en la especificación del elemento generalizado (padre). El hijo comparte la estructura y el comportamiento del padre.

**Realización**  
Relación semántica entre clasificadores, en donde un clasificador especifica un contrato que otro clasificador garantiza que cumplirá.

**De Clases**  
Muestra un conjunto de clases, interfaces y colaboraciones, así como sus relaciones. Abarcan la vista de diseño estática de un sistema.

**De Artefactos**  
Muestra los constituyentes físicos de un sistema en el computador. Los artefactos incluyen archivos, bases de datos y colecciones físicas de bits o similares.

**De Componentes**  
Representa la encapsulación de un clase, junto con sus interfaces, puertos y estructura interna, la cual está formada por otros componentes anidados y conectores.

**De Objetos**  
Muestra un conjunto de objetos y sus relaciones, representan instantáneas estáticas de instancias de los elementos existentes en los diagramas de clases.

**De Interacción**  
Muestra una interacción, que consta de un conjunto de objetos o roles, incluyendo los mensajes que pueden ser enviados entre ellos.

**De Estados**  
Muestra una máquina de estados, que consta de estados, transiciones, eventos y actividades. Un diagrama de estados muestra la vista dinámica de un objeto.

**De Actividades**  
Se trata de un diagrama que muestra el flujo de control de actividades, como en un proceso de negocio, un flujo de trabajo entre los usuarios y el sistema.

**De Despliegue**  
Muestra la configuración de nodos de procesamiento en tiempo de ejecución y los artefactos que residen en ellos. Normalmente, un nodo alberga uno o más artefactos.

**De Casos de Uso**  
Muestra un conjunto de casos de uso, actores y sus relaciones.

**De Paquetes**  
Muestra la descomposición del propio modelo en unidades organizativas y sus dependencias.

**Diagramas**  
Son la representación gráfica de un conjunto de elementos, visualizando la mayoría de las veces como un grafo conexo de nodos (elementos) y arcos (relaciones). Se dibujan para visualizar un sistema desde diferentes perspectivas, de forma que un diagrama es una proyección de un sistema.

**Nombres** → Cómo llamar a los elementos, relaciones y diagramas.

**Alcance** → El contexto que da un significado específico a un nombre.

**Visibilidad** → Cómo se pueden ver y utilizar esos nombres por otros.

**Integridad** → Cómo se relacionan apropiada y consistentemente unos elementos con otros.

**Ejecución** → Qué significa ejecutar o simular un modelo dinámico.

**Especificaciones**

-Detrás de cada elemento gráfico hay una especificación que proporciona una explicación textual de la sintaxis y semántica de ese bloque de construcción.

-Una notación gráfica de UML se utiliza para visualizar un sistema; la especificación de UML se utiliza para expresar los detalles del sistema.

-Proporcionan una base semántica que incluye a todas las partes de todos los modelos de un sistema, y cada parte está relacionada con las otras de manera consistente.

**Adornos**

-Los elementos de UML tienen una notación gráfica única y clara que proporciona una representación visual de los aspectos más importantes del elemento.

-La especificación de un elemento puede incluir otros detalles. Estos detalles se pueden incluir como adornos gráficos o textuales en la notación básica del elemento.

**Divisiones comunes**

**Entre Clases y Objetos**  
Una clase es una abstracción; un objeto es una manifestación concreta de esa abstracción.

**Entre Interfaz e Implementación**  
Una interfaz declara un contrato, y una implementación representa una realización concreta de ese contrato, responsable de hacer efectiva la semántica completa de la interfaz de forma fidedigna.

**Entre Tipo y Rol**  
El tipo declara la forma de una entidad, como un objeto, un atributo o un parámetro. Un rol describe el significado de una entidad en un contexto, como una clase, un componente o una colaboración.

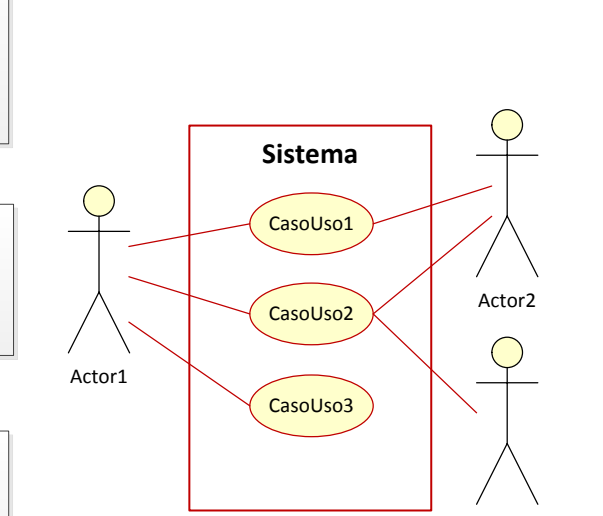
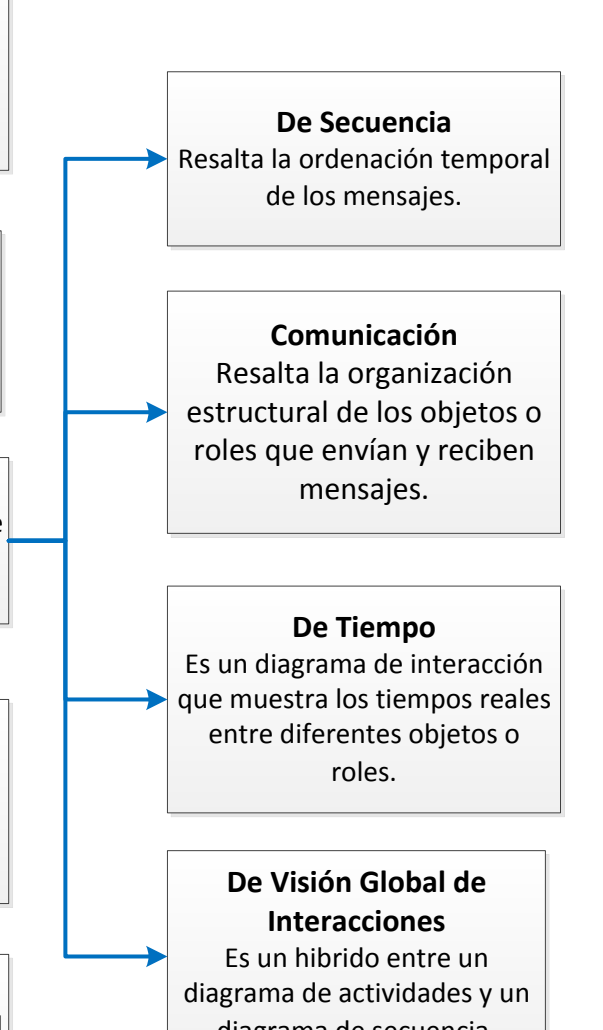
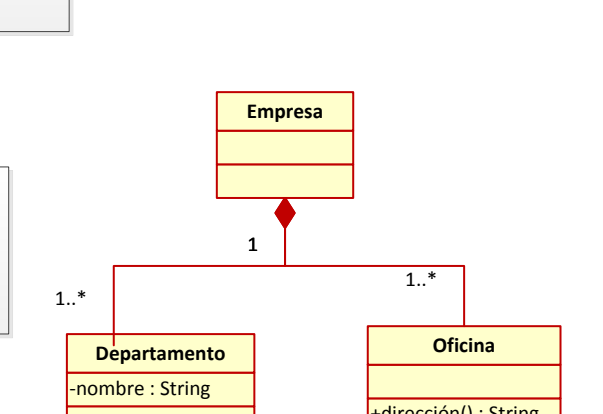
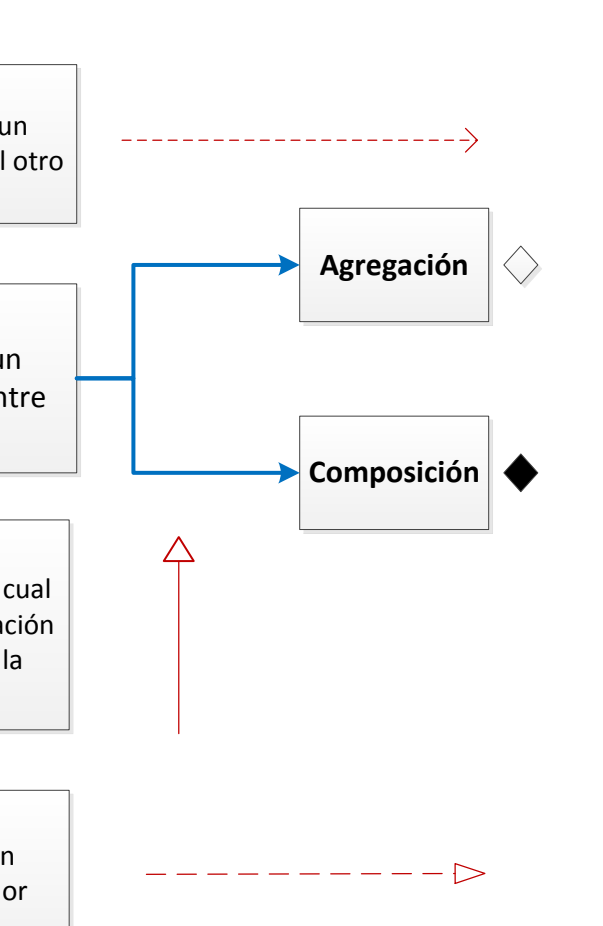
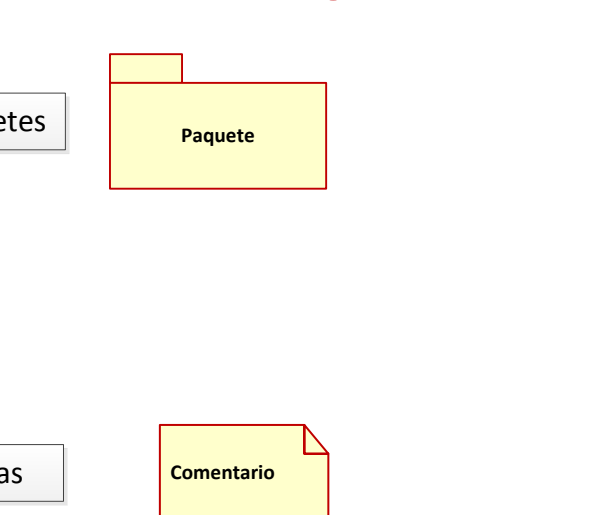
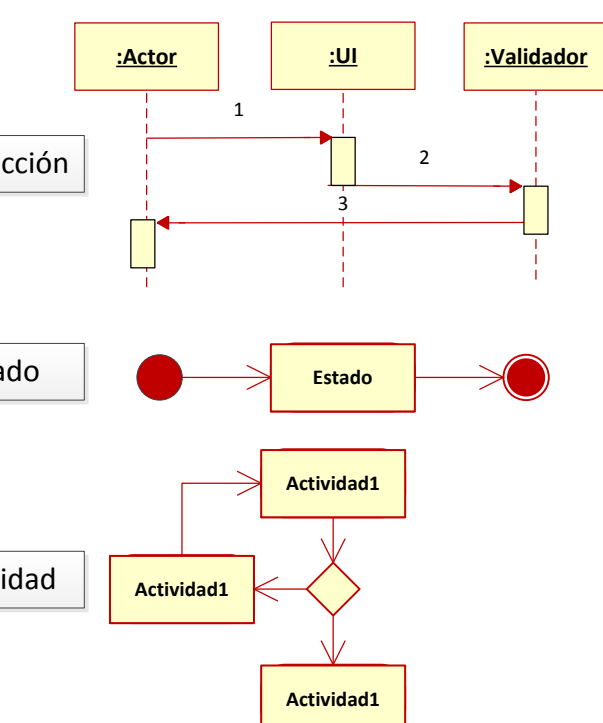
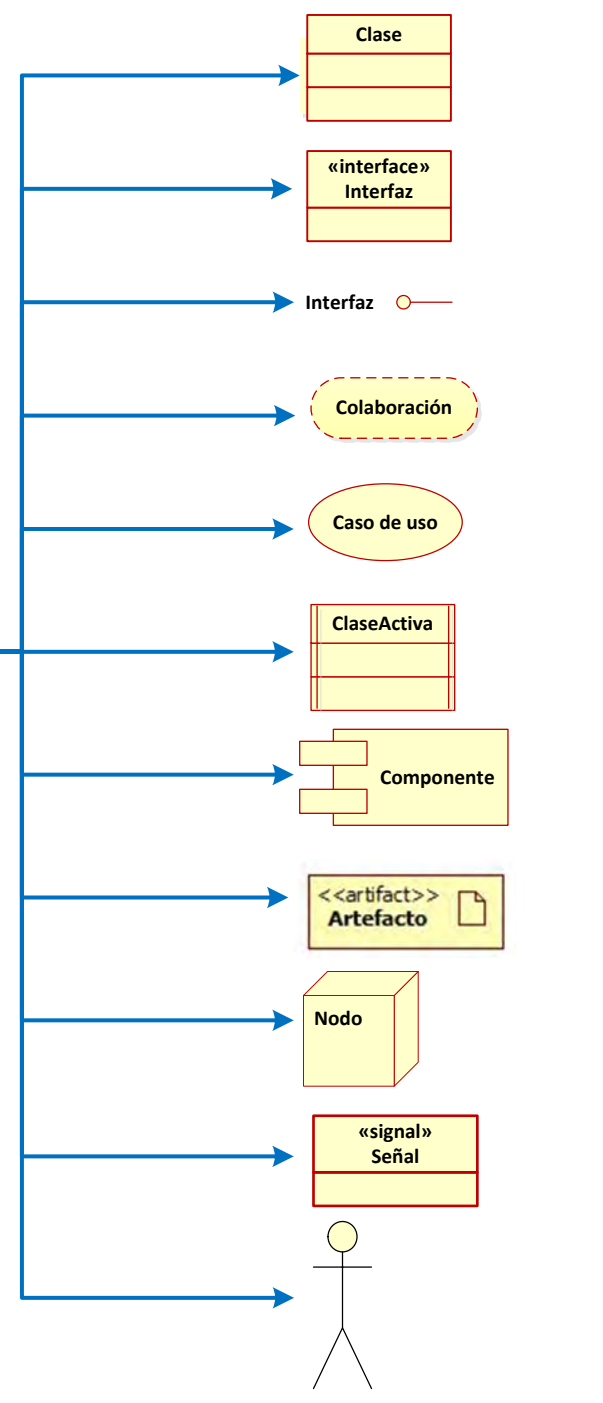
**Mecanismos de Extensibilidad**

**Estereotipos**  
Extienden el vocabulario de UML, permitiendo crear nuevos tipos de bloques de construcción que derivan de los existentes pero que son específicos a un problema.

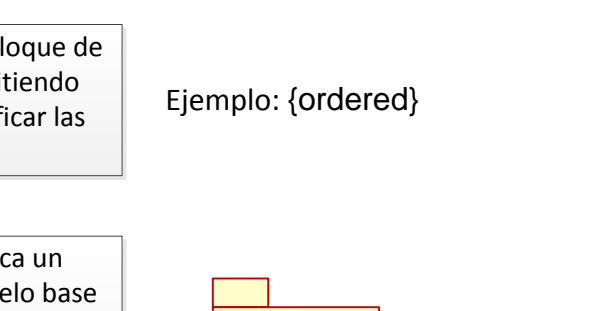
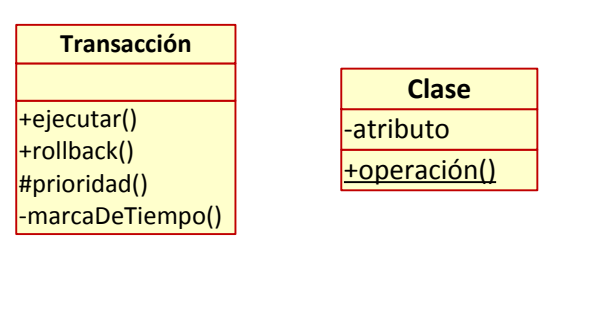
**Valores etiquetados**  
Extiende las propiedades de un estereotipo de UML, permitiendo añadir nueva información en la especificación de ese estereotipo.

**Restricciones**  
Extiende la semántica de un bloque de construcción de UML, permitiendo añadir nuevas reglas o modificar las existentes.

**Perfiles**  
Es un paquete que identifica un subconjunto de un metamodelo base existente y define estereotipos y restricciones que se pueden aplicar al subconjunto del metamodelo seleccionado.



Nivel	Visibilidad	Símbolo	Java
public	Cualquier clasificador externo con visibilidad hacia el clasificador dado puede utilizar esta característica.	+	public
protected	Cualquier descendiente del clasificador puede utilizar esta característica.	#	protected
private	Sólo el propio clasificador puede utilizar esta característica.	-	private
package	Sólo los clasificadores declarados en el mismo paquete pueden utilizar esta característica.	-	-



### Referencias

- Booch, Rumbaugh & Jacobson. (2006). *El lenguaje Unificado de Modelado. Guía del usuario*. Segunda Edición. Addison-Wesley.
- Perea Camarillo, Ismael. Apuntes del curso "UML Unified Modeling Language". Noviembre del 2013.



## Software Recomendado

- Microsoft® Visio
- Microsoft® Project
- StarUML. Disponible en: <http://staruml.io/>
- WBS Chart Pro. Disponible en: <http://www.criticaltools.com/wbschartprosoftware.htm>
- ProjectLibre. Disponible en: <http://www.projectlibre.org/>
- Planer. Disponibles en: <http://winplanner.sourceforge.net/>

## Bibliografía Recomendada

- *A Guide to the Project Management Body of Knowledge (PMBOK® Guide)*. (2013) Fifth Edition. *Project Management Institute®*.
- Booch, Rumbaugh & Jacobson. (2006). *El lenguaje Unificado de Modelado. Guía del usuario*. Segunda Edición. Addison-Wesley.
- Weitzenfeld, Alfredo. (2005), *Ingeniería de Software orientado a Objetos con UML, Java e Internet*. Cengage Learning.

---

## **Capítulo 6.**

# **Resultados y conclusiones de la aplicación de la Estrategia para el Desarrollo de Sistemas de Información**

---

## 6. Resultados y conclusiones de la aplicación de la Estrategia para el Desarrollo de Sistemas de Información.

En este trabajo se presentó una Estrategia que estructura de manera funcional administración de proyectos y diseño de software basada principalmente en PMBOK®, Scrum y UML, con el fin de identificar los elementos fundamentales que la componen.

El marco de referencia que sustenta esta Estrategia está compuesto por marcos de desarrollo de software ágiles como Scrum, XP y OpenUp, metodologías de administración de proyecto como PMBOK® y Prince2, además de UML para el modelado de sistemas.

En lo que se refiere a los marcos ágiles de desarrollo del software, se eligió hacer referencia a OpenUP debido a que había antecedentes de trabajos de titulación bajo este enfoque en la Secretaría de Personal Docente; XP porque presenta una estructura completa en relación con el número de roles que identifica y su proceso; y Scrum. Este último, se eligió como base para esta Estrategia por tratarse de un marco de trabajo para diferentes tipos de proyectos de innovación y para sustentar trabajos futuros de gestión de conocimiento en una investigación doctoral.<sup>20</sup>

En administración de proyectos, se hizo referencia a Prince2 por ser una metodología ampliamente usada en países europeos como el Reino Unido, así como el PMBOK®, metodología con reconocimiento internacional, siendo ésta la base de esta Estrategia debido a que se dieron las condiciones para recibir capacitación del tema, asesoramiento de expertos y porque yo contaba con una formación previa al colaborar en un organización donde la empleaban.

En relación al diseño de sistemas, y apoyando el primer objetivo específico del presente trabajo, se tomó como base UML ya que es un requerimiento en los trabajos de titulación en la FCA-UNAM a través del Diseño de un Sistema o Proyecto para una Organización.

La Estrategia diseñada se compone de las siguientes fases:

- Preparación: Etapa en la cual, el Alumno o Pasante de la Licenciatura en Informática inicia los trámites para empezar su servicio social o titulación en la modalidad de Diseño de un Sistema o Proyecto para una Organización desarrollando un sistema para la Secretaría de Personal Docente.

---

<sup>20</sup>Trabajo realizado actualmente en el posgrado de la Facultad de Contaduría y Administración.



- Inicio: Se forma un Equipo de Desarrollo con un Administrador de proyecto para construir un módulo de un sistema, que es guiado por un Administrador General de proyecto y ayudado por un Asesor externo. Este Equipo de desarrollo conoce de manera general qué se va a hacer, cómo se va hacer y el tiempo aproximado en el que será desarrollado el módulo del sistema.
- Planeación: Implica un análisis a profundidad sobre el Sistema a desarrollar y engloba cuestiones de Administración de Proyectos, así como el Diseño del sistema.
- Ejecución: En esta etapa se ponen en marcha todos los planes de Administración de Proyecto y se comienza con la programación del sistema que será ejecutada mediante *Sprints*. Para cada *Sprint*, el equipo de Desarrollo elegirá un Caso de Uso basado en las peticiones del Usuario y el *Sprint* termina una vez que el Equipo completa la programación del Caso de Uso y se hacen los *Sprints* suficientes para completar el módulo.
- Seguimiento: Se da durante la planeación hasta el cierre del proyecto y se asegura que los planes sean ejecutados acorde a lo previsto, se hacen cambios y ajustes requeridos en los Artefactos de Administración de Proyecto.
- Cierre: Se hace un análisis global de la ejecución del proyecto con fin de evaluar los aspectos negativos y positivos que sucedieron en el transcurso del proyecto y se cierra formalmente el mismo.

Esta Estrategia identificó los siguientes roles: Asesor, Alumno o Pasante de la Licenciatura en Informática, Usuario, Patrocinador, Administrador General del proyecto, Equipo de desarrollo, Líder de proyecto y Asesor externo.

El objetivo general de este trabajo se cumplió porque se identificaron los resultados en el diseño y aplicación de una estrategia de desarrollo de sistemas para implementar proyectos de software en la SPD.

Con el fin de responder a la pregunta de investigación: ¿Qué resultados se obtienen al diseñar y aplicar una estrategia de desarrollo de proyectos de software: Caso Secretaría de Personal Docente de la FCA, UNAM?, fueron desarrolladas plantillas de los documentos de acta de constitución del proyecto, identificación de interesados, lista de requerimientos, creación de un EDT, *sprint*

*backlog*, costo del proyecto, recursos humanos, plan de riesgos, plan de comunicación, acta de cierre del proyecto, lecciones aprendidas, minuta y especificación de casos de uso, que conforman la Estrategia de Desarrollo de Sistemas de Información, la cual fue utilizada por cuatro Equipos de desarrollo, que al término de su proyecto hicieron las Lecciones Aprendidas y contestaron el , de los cuales se extrajeron los siguientes comentarios:

Del Proyecto	De la Documentación
<ul style="list-style-type: none"> <li>• Faltó determinar tiempos de forma presencial fijando metas más cortas.</li> <li>• Se debió de extender más la curva de aprendizaje en la parte de programación.</li> <li>• Faltaron juntas presenciales.</li> <li>• El alcance del proyecto lo define todo, sin embargo, no había fechas fijas, ni hitos.</li> <li>• No hubo buena planeación del tiempo.</li> <li>• El rol de líder es fundamental debido a que es el eje del proyecto y supervisa que éste vaya en forma.</li> <li>• Todos los miembros del equipo deben de tener los mismos objetivos, la misma visión y satisfacer la misma necesidad para evitar retrasos y desmotivaciones.</li> <li>• Sin motivación, el factor humano puede no ser eficiente.</li> </ul>	<ul style="list-style-type: none"> <li>• Las explicaciones servían, pero a veces no fueron suficientes.</li> <li>• La extensión de los documentos fue muy buena.</li> <li>• Había ambigüedades en las explicaciones.</li> <li>• El uso de las minutas resultó muy útil.</li> <li>• El uso de la documentación ayudó a establecer lo que se debía de hacer, hacia donde iba el proyecto y lo que se debía obtener.</li> <li>• La documentación permitió tener un control de lo que se está haciendo y evaluar los objetivos iniciales y compararlos con los resultados.</li> <li>• Los documentos fueron objetivos y completos.</li> <li>• Sin la documentación el proyecto hubiera tomado rumbos no previstos</li> <li>• Algunos documentos resultaron repetitivos.</li> </ul>

Además, señalaron que los documentos que les fueron de mayor utilidad fueron el acta de constitución del proyecto, identificación de interesados, requerimientos y minutas. Los documentos que fueron utilizados con menor frecuencia fueron los documentos de creación de un EDT y el plan de comunicación.

Todos los encuestados manifestaron tener un interés por alguna certificación del PMI®.

Con el diseño y aplicación de esta Estrategia de Desarrollo de Sistemas de Información, se apoyó a estudiantes a concluir su servicio social y a un egresado de la FCA con su titulación, actuando como una guía y fortaleciendo el aprendizaje adquirido en la Licenciatura.

De manera general, en los proyectos que utilizaron esta Estrategia se generaron documentos uniformes, hubo un control en el alcance de los proyectos, los miembros de los equipos recibieron formación de administración de proyectos, programación y diseño de sistemas, sin embargo, hasta enero del 2015 no se han cerrado todos los proyectos debido a que se hizo un mal manejo del tiempo o no se respetaron los tiempos de entrega a causa de eventos que no tienen relación directa con esta Estrategia. Una primera versión de la Estrategia no contemplaba herramientas de gestión del tiempo basadas en PMBOK®, solo el tiempo total aproximado en el que se podía completar el proyecto, sin controlar los tiempos de ejecución de cada *Sprint*, lo cual derivó en la inclusión de una herramienta de control de tiempo en la versión final de esta Estrategia, constituido por un Cronograma.

Para un trabajo futuro se propone utilizar esta Estrategia, en su última versión, que integra los cambios de acuerdo al resultado observado en su primera implementación, pero incluyendo el manejo del tiempo.

En cuanto a las mejoras a esta Estrategia, se encuentran la inclusión de herramientas para el desarrollo de habilidades como la comunicación, el liderazgo y técnicas de resolución de conflictos y trato con el usuario, además de abarcar cuestiones técnicas relacionadas con la programación, como nombres de variables, de archivos con el código fuente y otros estándares técnicos.

De manera personal, el desarrollo de esta tesis permitió darme cuenta que necesitaba un cambio de hábitos, ya que en la segunda parte del desarrollo de este proyecto, empecé a procrastinar todas las actividades de esta tesis, lo que me obligó a buscar alternativas para incrementar mi productividad, energía y claridad mental.

Profesionalmente, adquirí conocimientos de Administración de Proyectos, UML, procesos y desarrollé habilidades de comunicación y liderazgo, capacidad de análisis para justificar y saber discriminar conocimientos y contenido y mejorar la redacción.



Así, este trabajo de investigación, además de cumplir con los objetivos iniciales, responder a la pregunta de investigación, diseñar y aplicar la Estrategia, me hizo adquirir experiencia, conocimientos, desarrollar habilidades y desarrollar mejores hábitos.

## Acrónimos y convenio tipográfico

- CAPM: *Certifies Associate in Project Management*
- EDSI: Estrategia de Desarrollo de Sistemas de Información
- EDT: Estructura de Desglose de Trabajo
- IEEE: *Institute of Electrical and Electronics Engineers*
- ISO: *International Organization for Standardization*
- OOPSLA: *Object-Oriented Programming Systems & Applications*
- PMBOK®: *Project Manager Body of Knowledge*
- PMI®: *Project Management Institute*
- PMP: *Project Manager Professional*
- RUP: *Rational Unified Process*
- SIGAD: Sistema Integral de Gestión Administrativa Docente
- SPD: Secretaría de Personal Docente
- UML: *Unified Modeling Language.*
- WBS: *Work Breakdown Structure*

*Esta tipografía es usada para palabras del idioma inglés, títulos de libros y artículos consultados.*

**Esta tipografía es usada para títulos y subtítulos.**

**Esta tipografía es usada para nombres de columnas y renglones en tablas, títulos de listas e índices y sus elementos.**

*<Esta tipografía es usada en los Artefactos para indicar qué información debe contener alguna sección>*

## Referencias

### Bibliográficas

- *A Guide to the Project Management Body of Knowledge (PMBOK® Guide)*. (2013) Fifth Edition. Project Management Institute.
- Beck, Kent. (1999), *Extreme Programming Explained: Embrace Change*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- Booch, Rumbaugh & Jacobson. (2006). *El lenguaje Unificado de Modelado. Guía del usuario*. Segunda Edición. Addison-Wesley.
- Cleland, D. & King, W. (1983), *Systems Analysis and Project Management*. McGraw Hill.
- Cockburn, A., (2002). *Agile Software Development*, Addison Wesley.
- Pressman, R. & Murrieta J. (2006), *Ingeniería del Software, Un enfoque práctico*. 6ª Edición. McGraw Hill.
- Sommerville, Ian. (2005), *Ingeniería del Software*, 7ª Edición. Pearson Addison Wesley.
- Weitzenfeld, Alfredo. (2005), *Ingeniería de Software orientado a Objetos con UML, Java e Internet*. Cengage Learning.

### En línea

- *An Introduction to agile software development*. (2007), Serena Software. Disponible en: <http://www.serena.com/docs/repository/solutions/intro-to-agile-devel.pdf>, Consultado el 9 de diciembre del 2013.
- Abrahamsson, P., Salo, O, & Ronkainen, J., (2002), *Agile software development methods, Review and analysis*. VTT Publications. Disponible en: <http://www.vtt.fi/inf/pdf/publications/2002/P478.pdf>, consultado el 9 de diciembre del 2013.
- Alvarez, Itzcoalt. (2007), *Desarrollo Ágil con SCRUM*. Disponible en: <http://cic.puj.edu.co/wiki/lib/exe/fetch.php?media=materias:sg07.p02.scrum.pdf>, consultado el 6 de diciembre del 2013.
- Balduino, Ricardo. (2007), *Introduction to OpenUP (Open Unified Process)*. Disponible en: <http://www.eclipse.org/epf/general/OpenUP.pdf>, consultado el 29 de mayo del 2014.

- Barakat, Mohammed, (2012). *Haciendo fácil la Estructura de Desglose del Trabajo*, disponible en:[http://www.liderdeproyecto.com/articulos/haciendo\\_facil\\_la\\_estructura\\_de\\_desglose\\_del\\_trabajo.html](http://www.liderdeproyecto.com/articulos/haciendo_facil_la_estructura_de_desglose_del_trabajo.html), consultado el 20 de febrero del 2014.
- Beltrán, Jaime, Carmona, Miguel et al. (2009), *Guía para una gestión basada en procesos*. Instituto Andaluz de Tecnología. Disponible en: <http://excelencia.iat.es/files/2012/08/2009.Gesti%C3%B3n-basada-procesos-completa.pdf>, consultado en mayo del 2014.
- Calero Solís, Manuel. (2003), *Una explicación a la programación Extrema (XP)*. Disponible en: <http://www.willydev.net/descargas/prev/ExplicaXP.pdf>, consultado el 9 de diciembre del 2013.
- Canós, José & Letelier Patricio. *Metodologías Agiles en el Desarrollo de Software*. Disponible en: [http://noqualityinside.com.ar/nqi/nqifiles/XP\\_Agil.pdf](http://noqualityinside.com.ar/nqi/nqifiles/XP_Agil.pdf), consultado el 6 de diciembre del 2013.
- Calero, Manuel. (2003), *Una explicación a la programación extrema (XP)*. V Encuentro usuarios xBase, Madrid. Disponible en: <http://www.willydev.net/descargas/prev/ExplicaXP.pdf>, consultado el 20 de mayo del 2004.
- *Cronograma*. PM4R. Gestión de proyecto para resultados. Disponible en: [http://www.pm4r.org/esp/centro-de-capacitacion/biblioteca-virtual/doc\\_download/16-guia-para-la-creacion-del-cronograma](http://www.pm4r.org/esp/centro-de-capacitacion/biblioteca-virtual/doc_download/16-guia-para-la-creacion-del-cronograma), consultado en noviembre del 2014.
- Collarins, Remi-Armand, (2009). *Estimación del costo del software usando puntuación en casos de uso: clarificar las transacciones de casos de uso*. Disponible en: [http://www.ibm.com/developerworks/ssa/rational/library/edge/09/mar09/collaris\\_dekker/](http://www.ibm.com/developerworks/ssa/rational/library/edge/09/mar09/collaris_dekker/), consultado en 3 de marzo del 2014.
- Daft, L. & Marcic, D. (2006), *Introducción a la administración*, 4<sup>a</sup> Edición. Cengage Learning. Disponible en: <http://go.galegroup.com/ps/i.do?id=GALE|2VGY&v=2.1&it=aboutBook&sw=w>, Consultado el 29 de abril del 2014.
- De Rojas, Fermín, (2011). *Qué es Work Breakdown Structure (WBS)*, disponible en: <http://spanishpmo.com/index.php/que-es-work-breakdown-structure-wbs/>, consultado el 25 de febrero del 2014.

- Esterkin, José. (2011), *Minutas de Reunión*. Mejores Proyectos. Disponible en: <http://iaap.wordpress.com/2011/10/26/minuta-de-reunion/>, consultado el 15 de junio del 2014.
- *Glossary of Terms*. PRINCE2 (2009). Disponible en: <http://www.prince2.com/system/files/downloads/prince22009glossary.pdf>, consultado el 4 de mayo del 2014.
- Gordon, Ann & McDonough, Michele, (2011), *Tips for Building a Work Breakdown Structure*, disponible en <http://www.brighthubpm.com/project-planning/2940-tips-for-building-a-work-breakdown-structure/>, consultado el 25 de febrero del 2014.
- Gómez, Julián, (2013). *Método de Estimación puntos de casos de uso (Use case points)*. Disponible en: <http://www.laboratorioti.com/2013/02/14/metodo-de-estimacion-puntos-casos-de-uso-use-case-points/>, consultado el 4 de marzo del 2014.
- *Gestión del Cronograma del Proyecto*. PM4DEV. Gerencia de proyectos para organizaciones de desarrollo. Disponible en: <http://www.gestionsocial.org/archivos/00000831/PM4DEV.2.pdf>, consultado en noviembre del 2014.
- Hernández Orallo, Enrique. (2006). *El Lenguaje Unificado de Modelado (UML)*. Disponible en: <http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>, consultado el 25 de agosto del 2014.
- Herrera, Eliécer & Valencia Luz. (2007), *Del manifiesto ágil sus valores y principios*. Scientia et Technica, año XIII, No 34, mayo de 2007. Universidad Tecnológica de Pereira. ISSN 0122-1701. Disponible en: <http://revistas.utp.edu.co/index.php/revistaciencia/article/view/5643/3047>, consultado el 9 de diciembre del 2013.
- Instituto Nacional de Tecnologías de la Comunicación (INTECO, 2009). *Ingeniería del Software: Metodologías y Ciclos de Vida*. Secretaria de Estado de Telecomunicaciones y para la sociedad de la Información (España). Laboratorio Nacional de Calidad del Software. Disponible en: [https://www.inteco.es/file/N85W1ZWfHifRgUc\\_oY8\\_Xg](https://www.inteco.es/file/N85W1ZWfHifRgUc_oY8_Xg), consultado el 5 de diciembre del 2013.
- Kniberg, Henrik. (2007), *Scrum and XP from the Trenches, How we do Scrum*. Versión 2.2. Disponible en:

<http://tschering.mono.net/upl/10004/110224ScrumAndXpFromTheTrenches.pdf>, consultado el 6 de diciembre del 2013.

- Luna, Rodríguez & Salazar. (2008), *Notas de Lecciones Aprendidas*. Banco Interamericano de Desarrollo. Disponible en:  
<http://boru.pbworks.com/f/Lecciones+Aprendidas+11-18-08.pdf>, consultado el 10 de junio del 2014.
- López Martín, Cuauhtémoc, (2007), *Estimación del esfuerzo de desarrollo de software*. Disponible en:  
<http://www.cimat.mx/Eventos/seminariodetecnologias/EstimacionDelEsfuerzo.pdf>, consultado el 3 de marzo del 2014.
- Letelier, Patricio & Penadés, Carmen. (2006), *Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP)*. Laboratorio de Sistemas de Información. Departamento de Sistemas Informáticos y Computación. Facultad de Informática. Universidad Politécnica de Valencia. Disponible en:  
<http://www.willydev.net/descargas/masyxp.pdf>, consultado el 20 de mayo del 2014.
- Morera, Elineth (2012). *Factores de éxito en la gestión de proyectos relacionados con las habilidades blandas de los estudiantes de Informática de la ULACIT*. Disponible en:  
[http://www.ulacit.ac.cr/files/careers/119\\_factoresdexitoenlagestindepoyectosrelacionadosconlashabilidadesblandasdelosestudiantesdeinformticadelaulacit.pdf](http://www.ulacit.ac.cr/files/careers/119_factoresdexitoenlagestindepoyectosrelacionadosconlashabilidadesblandasdelosestudiantesdeinformticadelaulacit.pdf), consultado en diciembre del 2013.
- Münch, Lourdes. (2010), *Administración*. Pearson Educación. Disponible en:  
<http://www.bibliotechnia.com>, consultado el 3 de mayo del 2014.
- Mintzberg, Henry y Quinn, Jamen Brian. (1998), *El Proceso Estratégico. Conceptos, contextos y casos*. Segunda Edición. Prentice Hall. Disponible en:  
[http://isabelportoperez.files.wordpress.com/2013/08/proceso\\_estrategico\\_mintzberg.pdf](http://isabelportoperez.files.wordpress.com/2013/08/proceso_estrategico_mintzberg.pdf), consultado en septiembre del 2014.
- Matis, Micah, (2014), *Work Breakdown Structure: Purpose, Process and Pitfalls*. Disponible en:  
<http://cdn.projectsart.co.uk/pdf/work-breakdown-structure-purpose-process-pitfalls.pdf>, consultado el 25 de febrero del 2014.
- Norma Internacional ISO 9000:2005. *Sistemas de gestión de la calidad. Fundamentos y vocabulario*. Traducción certificada. Disponible en:

- [http://www.uco.es/sae/archivo/normativa/ISO\\_9000\\_2005.pdf](http://www.uco.es/sae/archivo/normativa/ISO_9000_2005.pdf), consultado en mayo del 2014.
- Palacio, Juan. (2014), *Gestión de proyectos Scrum Manager (Scrum Manager I y II)*. Versión 2.5, Scrum Manager®. Disponible en:  
[http://www.scrummanager.net/files/sm\\_proyecto.pdf](http://www.scrummanager.net/files/sm_proyecto.pdf), consultado el 20 de mayo del 2014.
  - Pekka et al. (2002), *Agile software development methods review and analysis*. Disponible en:  
<http://www.vtt.fi/inf/pdf/publications/2002/P478.pdf>, consultado el 21 de mayo del 2014.
  - Ribero, Juan. (2013), *Las lecciones aprendidas en los Proyectos*. Disponible en:  
<http://www.isolucion.com.co/Info/las-lecciones-aprendidas-en-los-proyectos.asp>, consultado el 15 de junio del 2014.
  - Ríos, Santiago. (2013) *Aplicación de la metodología OpenUP en el desarrollo del sistema de difusión de gestión del conocimiento de la ESPE*. Disponibles en:  
<http://repositorio.espe.edu.ec/bitstream/21000/6316/1/AC-SISTEMAS-ESPE-047042.pdf>, consultado el 29 de mayo del 2014.
  - Rodríguez Blanco, Francisco Javier. (2012). PMI en el mundo y en España, tendencias en la Dirección de Proyectos. Disponible en:  
<http://www.pmi-mad.org/dmdocuments/2012-06-26-pmi-en-el-mundo-y-en-espa%C3%B1a---tendencias-en-la-direcci%C3%B3n-de-proyectos.pdf>, consultado en diciembre del 2013.
  - Sánchez Orduña, Ricardo. (2013). *La Estructura de Desglose de Trabajo (EDT)*. Disponible en:  
<http://www.pm4r.org/esp/punto/blog-de-pm4r/entry/la-estructura-de-desglose-del-trabajo-edt>, consultado el 25 de febrero del 2014.
  - Schwaber, Ken & Sutherland, Jeff. (2013), *La guía de Scrum, La guía Definitiva de Scrum: Las Reglas del Juego*. Disponible en:  
<https://www.scrum.org/Portals/0/Documents/Scrum%20Guides/2013/Scrum-Guide-ES.pdf#zoom=100>, consultado el 6 de diciembre del 2013.
  - Sliger, Michele. (2006). *A Project Manager's Survival Guide to Going Agile*. Disponible en:  
[http://images.brighthub.com/media/9039B0\\_a\\_project\\_manager-s\\_survival\\_guide\\_to\\_going\\_agile.pdf](http://images.brighthub.com/media/9039B0_a_project_manager-s_survival_guide_to_going_agile.pdf), consultado el 6 de diciembre del 2013.
  - Szalvay, Victor. (2004), *An Introduction to Agile Software Development*. Danube Technologies, Inc. Disponible en: [http://www.danube.com/docs/Intro\\_to\\_Agile.pdf](http://www.danube.com/docs/Intro_to_Agile.pdf), consultado el 6 de diciembre del 2013.

- Sterling, Chris. (2009), *Scrum is the Vehicle, not the Destination*. Getting Agile, Real world applications of agile and lean in technology and business. Disponible en: <http://www.gettingagile.com/2009/05/01/scrum-is-the-vehicle-not-the-destination/>, consultado el 20 de mayo del 2014.
- Sociedad Latinoamericana para la Calidad, (2000), *Lluvia de Idea*. Disponible en: [http://homepage.cem.itesm.mx/alesando/index\\_archivos/MetodoDisMejoraDeProcesos/LluviaDelIdeas.pdf](http://homepage.cem.itesm.mx/alesando/index_archivos/MetodoDisMejoraDeProcesos/LluviaDelIdeas.pdf), consultado el 10 de junio del 2014.
- Van Roojen, Howard. (2006), *Agile Software Development with Scrum, Scrum FAQ by Ken Schwaber (Transcript of the podcast series)*. Conchango. Disponible en: <https://www.cs.tcd.ie/John.Dingliana/cs7038/Schwaber-AGILE-FAQ.pdf>, consultado el 20 de mayo del 2014.
- Valero Orea, Sergio, (2009). *Estimación de proyectos de software con puntos de casos de uso*. Disponible en: <http://www.utim.edu.mx/~svalero/docs/id45.pdf>, consultado el 4 de marzo del 2014.
- Wake, William. (2000), *Extreme Programming Explored*. Disponible en: <http://faculty.ksu.edu.sa/MubarakRashed/Books/Extreme.Programming.Explored.pdf>, consultado el 20 de mayo del 2014.



## Anexos

### Anexo 1: Acta de Constitución del Proyecto

---

*Desarrollo del Módulo Estímulos del Sistema Integral de Gestión  
Administrativa Docente (SIGAD)*

Acta de Constitución del Proyecto

*Facultad de Contaduría y Administración de la UNAM*

*Secretaría de Personal Docente*

*Febrero, 2014.*

## 1. Identificación del Documento

**Nombre del proyecto:** Desarrollo del Módulo Estímulos del SIGAD.

**Código del documento:** Inicio\_Acta\_Estimulos75

**Nombre:** Acta de Constitución del Proyecto

**Fecha de Creación:** 01/02/2014

### 1.1. Historial de Cambios

<b>Versión</b>	<b>Fecha</b>	<b>Quien los desarrolla</b>	<b>Quien los verifica</b>	<b>Breve descripción del cambio</b>
<b>v1</b>	Febrero del 2014.	Fabian Olguin		Creación del Documento.

## 1. Introducción

El Sistema Integral de la Gestión Administrativa Docente (SIGAD) desarrollado para la Secretaría de Personal Docente de la Facultad de Contaduría de Administración, que está dividido por módulos, tiene como objetivo automatizar las diferentes áreas de la Secretaría. Uno de estos módulos es Control Docente, que a su vez está integrado por nueve submódulos, siendo uno de ellos el submódulo de Estímulos, cuyo desarrollo es el objetivo de este proyecto.

## 2. Propósito del Proyecto

A continuación se describe de manera detallada el Caso de Negocio que da origen al proyecto, así como el objetivo e impacto de éste dentro de la organización.

### 3.1 Caso de Negocio

El Sistema Integral de la Gestión Administrativa Docente (SIGAD) surgió a partir de la necesidad de contar con un sistema automatizado que sirviera como herramienta de apoyo para la Secretaría de Personal Docente que permita una gestión de calidad en los procesos que realiza con la finalidad de mejorar los servicios que ofrece, así como, lograr una eficiente comunicación entre las autoridades académicas y administrativas de la Facultad de Contaduría y Administración.

Su objetivo es “Diseñar e implementar un sistema automatizado para cada una de las áreas que integran la Secretaría de Personal Docente, el cual permita realizar de manera eficiente sus procesos correspondientes, de tal forma que facilite la obtención oportuna de información y sirva como base para una adecuada toma de decisiones.” (SPD, 2012)

El módulo de Control Docente es parte del SIGAD Docente y tiene como propósito principal agilizar las actividades de la Secretaría de Personal Docente en beneficio del personal académico.

La Figura 2.1 muestra la estructura que conforma el módulo de Control Docente.

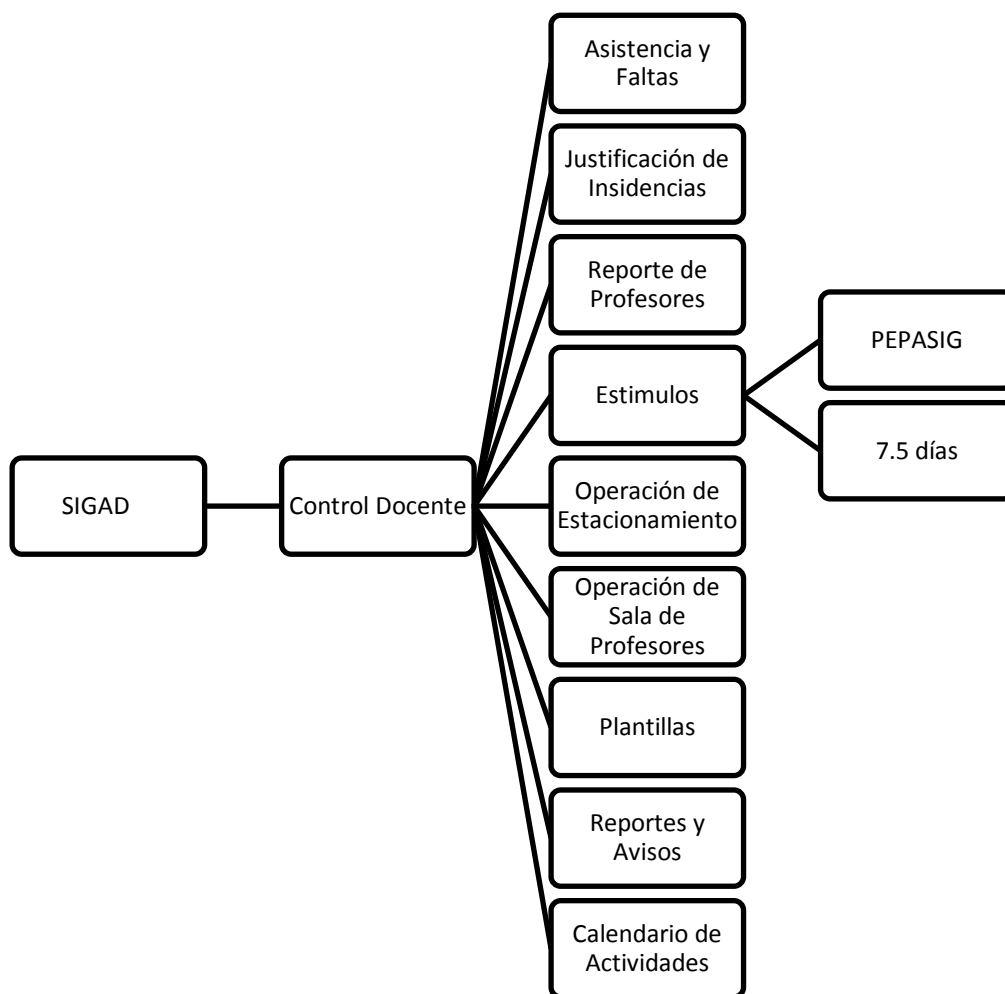


Figura 2.1: Estructura del Módulo Control Docente

### 2.1.1. Estímulos

El submódulo de estímulos se encarga de gestionar los programas denominados PEPASIG y 7.5 días, programas que tienen el propósito de apoyar el desempeño del personal académico de la Facultad de Contaduría y Administración.

Dentro de las siguientes secciones se abordan los programas de estímulo que conforma el submódulo respectivo.

#### 2.1.1.1. Programa de Estímulos 7.5 días

El Programa de Estímulos 7.5 días tiene como finalidad incentivar a los profesores que se encuentren laborando en la UNAM que cumplan como mínimo con el 90%

de asistencias durante el semestre lectivo, siempre y cuando se cumpla con la disposición contractual vigente (Cláusula 51 del CCT), de manera que cumplan los horarios de clase asignados a cada uno de ellos y a su vez elevar el desempeño académico en beneficio de la Facultad de Contaduría y Administración.

#### **2.1.1.1.1. Proceso Actual**

A continuación se describe, de manera general, el proceso que se lleva a cabo para generar el reporte final de los beneficiados por el programa de Estímulo 7.5 días, dicho proceso no se detalla debido a que parte de la información se considera privada.

- El reporte de asistencia generado por el Sistema de Control de Asistencia de Huella Digital (SCAHD) se toma como base. El reporte es generado en un archivo con el formato .xls el cual contiene las divisiones de Estudios Profesionales, Posgrado, SUAyED y Tiempo Completo.
- Se solicita al Departamento de Movimientos y Prestaciones (DMYP) la plantilla correspondiente al semestre lectivo a evaluar para el pago del estímulo en el formato .xls.
- Se crea un archivo .xls tomando como base la plantilla proporcionada por el Departamento de Movimientos y Prestaciones. El archivo contiene la siguiente información: División, grupo, número de empleado, RFC, horas, asignatura, sesiones, asistencias, faltas, reposiciones, porcentaje de asistencia, categoría, fecha de inicio y fecha de término.
- Con el archivo generado, se procede a determinar si los académicos son acreedores o no al estímulo considerando lo establecido en la Cláusula No. 51 del C.C.T. estímulo por asistencia.

- Una vez que se reunió la información del profesor y se determinó si son o no acreedores al estímulo, los registros son organizados en orden alfabético y agrupados por división de Estudios Profesionales, Posgrado, SUAyED y Tiempo Completo.
- Finalmente, el archivo se presenta ante la DGPe para la evaluación de dichos registros de asistencia.

#### **2.1.1.1.2. Problemática**

Actualmente, el programa de Estímulos 7.5 días involucra un esfuerzo mayor al requerido por parte del personal del Departamento de Control Docente y Programas Institucionales debido a que no se cuenta de manera inmediata con el reporte final de asistencia dicho reporte en ocasiones no se encuentra completo o presenta algún tipo de incongruencia al no contar con el registro completo de todos los grupos del semestre lectivo vigente de manera adecuada o se recurre a otros departamentos para poder complementarlo. Los factores mencionados anteriormente retrasan la asignación del aliciente a los profesores que resultan beneficiados con dicho estímulo.

#### **2.1.1.2. Descripción General**

Dentro de las siguientes secciones se describe el programa de Estímulos 7.5 días.

##### **2.1.1.2.1. Participantes**

Los participantes que pueden aplicar al programa de Estímulos 7.5 días corresponden a profesores con nombramiento de asignatura o de tiempo

completo que se encuentren impartiendo clases dentro de los planes y programas de estudios aprobados por el Consejo Universitario.

#### **2.1.1.2.2. Requerimientos**

Los requerimientos solicitados para poder aplicar al programa de Estímulos 7.5 días son los siguientes:

- a) Contar como mínimo con el 90% de asistencia durante el semestre que impartió clase.
- b) Considerar las políticas establecidas para el pago de estímulo por asistencia al personal académico.

#### **2.1.1.2.3. Disposiciones Generales**

Los estímulos asignados por medio del programa 7.5 días, son beneficios que no forman parte del salario nominal, por lo que no forman un ingreso fijo, regular o permanente.

La asignación de este estímulo se encuentra sujeta a un proceso determinado de evaluación académica que valora, tanto el rendimiento, como la productividad ejercida por los académicos.

El programa de estímulo, considera dos semestres al año, por lo tanto, corresponde el pago de 7.5 días al semestre que en conjunto forman los 15 días de salario íntegro del profesor que cumpla como mínimo con el 90% de asistencia.

#### **2.1.1.2.4. Incentivo del programa**

El estímulo consiste en otorgar una cantidad equivalente a una quincena de salario íntegro del profesor que cumpla con lo establecido en la Cláusula 51 del Contrato Colectivo de Trabajo para el Personal Académico (CCT) que se encuentre vigente.

### 3. Descripción del Proyecto

A continuación se describen las generalidades más importantes del proyecto.

#### 3.1. Criterios de éxito

Se considera que los siguientes eventos o situaciones favorezcan el desarrollo del proyecto:

- Disposición por parte del usuario para resolver dudas, hacer aclaraciones y evaluar los avances del proyecto cuando el Desarrollador lo solicite.
- Alta comunicación del Desarrollador con su asesor de trabajo de titulación, el cual guiará y aportará opiniones con respecto al desarrollo del proyecto.
- Contar con las herramientas tecnológicas adecuadas para poder llevar a cabo el desarrollo del sistema.

#### 3.2. Requerimientos

Los requerimientos iniciales para el Sistema de Estímulos 7.5 solicitados por el usuario son los siguientes:

El sistema deberá contar con control de acceso.

Tendrá que generar reportes semestrales de los beneficiados y no beneficiados por el programa de Estímulos 7.5 días, calculará el porcentaje de asistencias de los académicos y tendrá que determinar si el candidato es merecedor o no al aliciente del programa de Estímulos 7.5 días con base en el porcentaje de asistencias.

El sistema deberá solicitar el período de interés para generar el reporte semestral y permitirá emitir reportes semestrales en los formatos .xls o .PDF.

El sistema deberá permitir la consulta de un académico en específico, solicitando su número de empleado.

El sistema deberá permitir generar una estadística semestral del programa de Estímulos 7.5 días y deberá permitir imprimir el reporte semestral.

El sistema deberá permitir recuperar la contraseña de acceso a los usuarios autorizados para utilizar el sistema vía correo electrónico y deberá permitir al administrador de sistemas gestionar a los usuarios registrados.

El sistema deberá ser visto como un sitio web intuitivo y que permita una fácil navegación.



### 3.3. Restricciones

- El sistema solo evaluará a los profesores de asignatura vigentes así como los profesores de tiempo completo que imparten alguna materia fuera del tiempo establecido de su jornada laboral.

### 3.4. Supuestos

El módulo que proporciona información para el funcionamiento del sistema se encuentra finalizado.<sup>21</sup>

### 3.5. Exclusiones

Este proyecto tiene las siguientes exclusiones:

- Se trabajará con datos de prueba para el desarrollo de este sistema siguiendo los lineamientos establecidos.

### 3.6. Alcance preliminar

Este proyecto cubre el submódulo del Estímulos que a su vez está dividido en 2 partes:

- I. El Programa de Estímulos a la Productividad y al Rendimiento del Personal Académico de Asignatura (PEPASIG), el cual tiene como finalidad estimular la labor de los profesores de asignatura de la UNAM de forma que se logre elevar su nivel de productividad y calidad en cuanto a su desempeño académico.
- II. El Programa de Estímulos 7.5 días, el cual corresponden a profesores con nombramiento de asignatura o de tiempo completo que se encuentren impartiendo clases dentro de los planes y programas de estudios aprobados por el Consejo Universitario.

Este documento detalla solo el programa de Estímulos 7.5.

---

<sup>21</sup>Submódulo de asistencias e incidencias del Sistema Integral de Gestión Administrativa Docente.

### **3.7. Duración estimada del proyecto**

Se estima que el desarrollo total del proyecto de ambas partes (Estímulos 7.5 y PEPASIG) se debe cubrir en 24 semanas, en las cuales se consideran cinco días laborales por semana con duración de ocho horas diarias. Cubriendo Estímulos 7.5 en ocho semanas.

## **4. Riesgos**

A continuación se listan los principales riesgos que podrían surgir durante el desarrollo del proyecto:

- Una curva de aprendizaje por parte del Desarrollador que lleve mucho tiempo construir en cuestiones técnicas y que ocasione un atraso en las entregas del sistema.
- Cambios en el proceso actual para determinar los beneficiados y no beneficiados al incentivo otorgado 7.5 días.
- Cambios en los sistemas que proporcionan información.
- Interrupción extraordinaria de actividades en la FCA que atrasen las reuniones con el usuario.

## **5. Interesados**

«A petición del Patrocinador del proyecto, se omiten los datos de esta sección.»

### **5.1. Internos**

### **5.2. Externos**

## **6. Información Técnica del producto**

El sistema empleará para su desarrollo las siguientes herramientas tecnológicas.

- Lenguaje de Programación PHP Orientado a Objetos
- StarUML
- Framework CodeIgniter
- Sistema Manejador de Base de Datos PostgreSQL

## Anexo 2: Identificación de Interesados

---

### 1. Identificación del Documento

**Nombre del proyecto:** <Nombre del Proyecto>

**Código del documento:** Inicio\_Interesados\_<Nombre o siglas del Proyecto>

**Nombre:** Interesados en el Proyecto

**Fecha de Creación:** DD/MM/AAAA

**Propósito:** Identifica personas, grupos u organizaciones que podrían afectar o verse afectados por una decisión, actividad o resultado del proyecto. Documenta intereses, participación, interdependencias, influencias, impacto o beneficio sobre el proyecto, con la finalidad de determinar un enfoque apropiado para cada grupo de interés.

#### 1.1. Historial de Cambios

Versión	Fecha	Quien los desarrolla	Quien los verifica	Breve descripción del cambio

## 2. Interesados Internos

<Personas, grupos u organización a cargo del desarrollo del proyecto>

Nombre	Rol	Datos de Contacto	Motivos y Expectativas	Clasificación de Impacto*

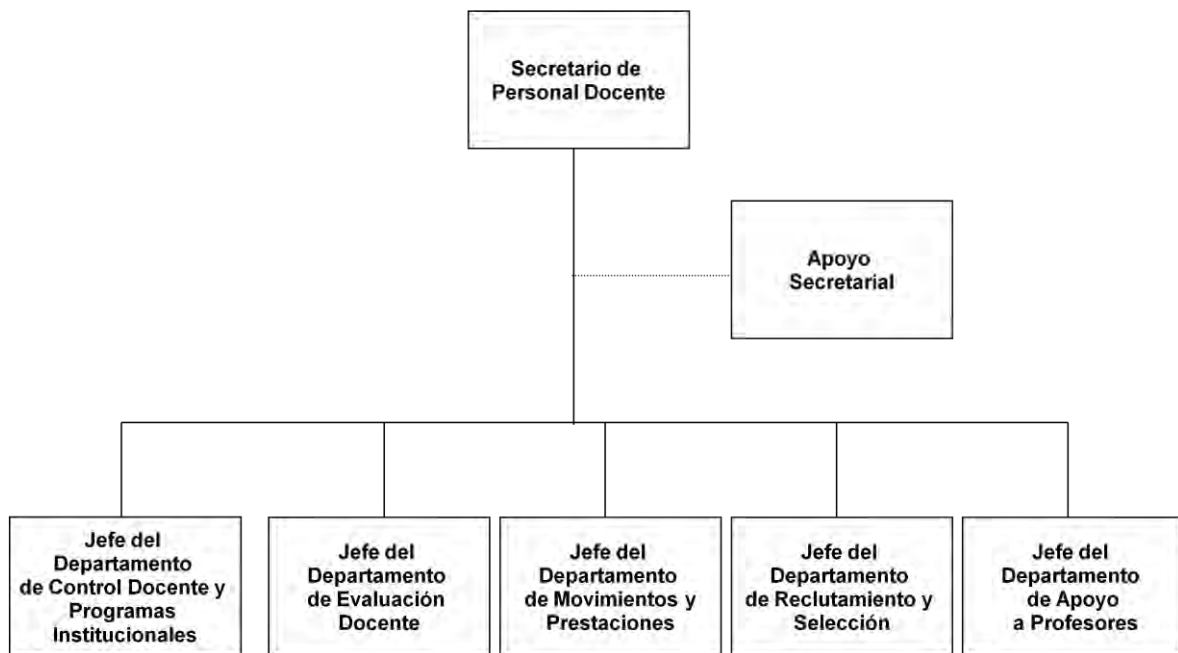
Nota: En Anexo 1 al final de este documento se encuentran las instrucciones para llenar los apartados de *Grado de Poder*, *Grado de Interés* y la *Clasificación de Impacto*.

## 3. Interesados Externos

<Personas, grupos u organizaciones que solicitan el proyecto, patrocinan, proporcionan los requerimientos o serán los usuarios finales del producto del proyecto>

### 3.1. Secretaría de Personal Docente

#### 3.1.1. Organigrama General



### 3.1.2. Organigrama del Departamento

<Organigrama del Departamento para el cual se esté desarrollando el proyecto, los organigramas se encuentran en los manuales de la Secretaría y estos deberán de llevar nombre de los Secretarios, Jefes y Coordinadores para conocer la relación jerárquica existente>

Nombre	Rol	Datos de Contacto	Motivos y Expectativas	Clasificación de Impacto*

#### \*Anexo A. instrucciones para llenar los apartados de Grado de Poder, Grado de Interés y la Clasificación de Impacto.

- a) En la columna de *Clasificación de Impacto*, se debe tomar en cuenta el *Grado de Poder* y *Grado de Interés* donde solo se tienen dos posibles valores:
- Alto
  - Bajo
- b) De la combinación de los valores de *Grado de Poder* y *Grado de Interés* será el resultado de la *Clasificación de Impacto*, quedando de la siguiente forma:

Grado de Poder	Grados de Interés	Clasificación de Impacto
Alto	Alto	1 Muy Alto
Alto	Bajo	2 Alto
Bajo	Alto	3 Normal
Bajo	Bajo	4 Bajo

## Anexo 3: Requerimientos del Proyecto

---

### 1. Identificación del Documento

**Nombre del proyecto:** Desarrollo del Módulo Estímulos del SIGAD.

**Código del documento:** Planeación\_Requerimientos\_Estimulos75

**Nombre:** Requerimientos del Proyecto

**Fecha de Creación:** 20/02/2014

**Propósito:** La recopilación de los requerimientos es el proceso que determina, documenta y gestiona las necesidades de las partes interesadas y requisitos para cumplir con los objetivos del proyecto. Proporciona la base para la definición y gestión del alcance del proyecto y producto.

#### 6.1.Historial de Cambios

Versión	Fecha	Quien los desarrolla	Quien los verifica	Breve descripción del cambio

## 1. Introducción

El Sistema Integral de la Gestión Administrativa Docente (SIGAD) desarrollado para la Secretaría de Personal Docente de la Facultad de Contaduría de Administración, que está dividido por módulos, tiene como objetivo automatizar las diferentes áreas de la Secretaría. Uno de estos módulos es Control Docente, que a su vez está integrado por nueve submódulos, siendo uno de ellos el submódulo de Estímulos, cuyo desarrollo es el objetivo de este proyecto. El cual está dividido en dos partes:

- I. El Programa de Estímulos 7.5 días, el cual tiene como finalidad incentivar a los profesores que se encuentren laborando en la UNAM que cumplan como mínimo con el 90% de asistencias durante el semestre lectivo, siempre y cuando se cumpla con la disposición contractual vigente (Cláusula 51 del CCT), de manera que cumplan los horarios de clase asignados a cada uno de ellos y a su vez elevar el desempeño académico en beneficio de la Facultad de Contaduría y Administración.
- II. El Programa de Estímulos a la Productividad y al Rendimiento del Personal Académico de Asignatura (PEPASIG), que tiene como finalidad estimular la labor de los profesores de asignatura de la UNAM de forma que se logre elevar su nivel de productividad y calidad en cuanto a su desempeño académico.

El presente documento describe los Requerimientos para el desarrollo del proyecto, los cuales se describen a continuación.

## 2. Descripción global

### 2.1. Características del usuario

El sistema será usado por un Licenciado en Administración, con un amplio conocimiento en la Suite de Office de Microsoft, lo cual indica que tienen habilidad para usar sistemas Informáticos.

## 2.2.Perspectiva del sistema

El Submódulo del Estímulos tiene dependencia con El Sistema Integral de la Gestión Administrativa Docente (SIGAD), por lo tanto la integración del submódulo de Estímulos al SIGAD estará sujeta a:

- Que Submódulo de asistencias e incidencias del Sistema Integral de Gestión Administrativa Docente funcione de manera correcta.

## 2.3.Funciones del sistema

El sistema generará reportes de los beneficiarios y no beneficiarios por el programa de Estímulos 7.5, además hará estadísticas sobre la asistencia de los profesores.

Estos reportes deberán ser generados por periodos semestrales en formatos .xls o PDF, además permitirá el acceso a la información de cada profesor.

## 2.4.Restricciones

El sistema solo evaluará a los profesores de asignatura vigentes así como los profesores de tiempo completo que imparten alguna materia fuera del tiempo establecido de su jornada laboral.



### 3. Requerimientos

#### 3.1.Requerimientos Funcionales

ID	Requerimiento / Funcionalidad	*Prioridad
EST_001	El sistema deberá solicitar nombre de usuario y contraseña para permitir el acceso sólo a usuarios autorizados.	Alta
EST_002	El sistema deberá generar reportes semestrales de los beneficiados, no beneficiados y general (beneficiados y no beneficiados) por el programa de Estímulos 7.5 días con la siguiente información: RFC, Número de Empleado, Nombre, Asignatura, Porcentaje de Asistencia Período, Forma Única Folio, Código Programático, Horas nómina por grupo, Año de forma única, Categoría y si será beneficiado o no con el incentivo del programa Estímulos 7.5 días.	Alta
EST_003	El sistema deberá calcular el porcentaje de asistencias de los académicos.	Alta
EST_004	El sistema deberá determinar si el candidato es merecedor o no al aliciente del programa de Estímulos 7.5 días con base al porcentaje de asistencias obtenido durante el semestre lectivo anterior y bajo consideración de las incidencias.	Alta
EST_005	El sistema deberá solicitar el período de interés para generar el reporte semestral de los beneficiados, no beneficiados y general (beneficiados y no beneficiados) por el programa de Estímulos 7.5 días.	Alta

<b>EST_006</b>	El sistema deberá generar el reporte semestral de los beneficiados, no beneficiados y general (beneficiados y no beneficiados) por el programa de Estímulos 7.5 días en orden alfabético considerando su apellido paterno.	<b>Alta</b>
<b>EST_007</b>	El sistema permitirá emitir reportes semestrales en los formatos .xls o .PDF de los beneficiados, no beneficiados y general (beneficiados y no beneficiados) por el programa de Estímulos 7.5 días.	<b>Alta</b>
<b>EST_008</b>	El sistema deberá permitir la consulta de un académico en específico solicitando su número de empleado. Dicha consulta proporcionará la siguiente información: RFC, Número de Empleado, Nombre, Asignatura, Porcentaje de Asistencia Período, Forma Única Folio, Código Programático, Horas nómina por grupo, Año de forma única, Categoría y será beneficiado o no con incentivo del programa Estímulos 7.5 días.	<b>Alta</b>
<b>EST_009</b>	El sistema deberá permitir generar una estadística semestral del programa de Estímulos 7.5 días, solicitando el semestre de interés y considerando las sub-dependencias SUAYED, Posgrado, Intersemestrales, Talleres Extracurriculares y Licenciatura, así como el número de grupos, promedio de asistencia, total de profesores, grupos no beneficiados, promedio de asistencia de grupos no beneficiados, total de profesores no beneficiados, grupos beneficiados, promedio de asistencia de grupos beneficiados y total de profesores beneficiados por sub-dependencia.	<b>Alta</b>

<b>EST_010</b>	El sistema deberá permitir imprimir el reporte semestral del programa de Estímulos 7.5 días	<b>Alta</b>
<b>EST_011</b>	El sistema deberá permitir recuperar la contraseña de acceso a los usuarios autorizados para utilizar el sistema vía correo electrónico.	<b>Alta</b>
<b>EST_012</b>	El sistema deberá permitir al administrador de sistemas gestionar a los usuarios registrados para usar el sistema.	<b>Alta</b>
<b>EST_013</b>	El sistema deberá ser visto como un sitio web intuitivo y que permitir una fácil navegación. Debe contener una sección para consultar el académico, otra para generar el reporte y una opción de cierre de sesión. Una vez generado el reporte se presentan las opciones siguientes Generar estadística, Formato e Imprimir.	<b>Alta</b>

### 3.2.Requerimientos No Funcionales

<b>ID</b>	<b>Requerimiento / Funcionalidad</b>	<b>*Prioridad</b>

<b>*Prioridad</b>
• <b>Alta</b>
• <b>Media</b>
• <b>Baja</b>

### 1. Identificación del Documento

**Código del documento:** Planeación\_CreaciónEDT

**Nombre:** Creación de la Estructura de Desglose del Trabajo

**Propósito:** El EDT es una descomposición jerárquica, basada en los entregables del trabajo que debe ejecutar el equipo del proyecto para lograr los objetivos del proyecto y crear los entregables requeridos.

## 2. Introducción

El presente documento proporciona una guía para el desarrollo de la Estructura de Descomposición de Trabajo, en inglés *Work Breakdown Structure*, en el cual se subdividen los entregables y el trabajo del proyecto en componentes más pequeños para un mejor manejo y entendimientos de éstos.

## 3. Estructura de Desglose del Trabajo

### 3.1. ¿Qué es?

Es una herramienta técnica cuya principal característica es la descomposición jerárquica de los entregables y tareas a realizar en un proyecto dando como resultado su alcance total. Su representación gráfica es tipo organigrama en donde se esquematizan los paquetes de trabajo.

El equipo de desarrollo deberá desglosar las actividades siguiendo un orden jerárquico para establecer la relación de cada actividad con la meta final del proyecto.

### 3.2. Descomposición

La descomposición es la subdivisión de los entregables del proyecto en componentes más pequeños y manejables de manera que el trabajo y los entregables queden definidos al nivel de paquetes de trabajo. Estos paquetes de trabajo representan el nivel más bajo en la estructura y donde se pueden estimar el costo y la duración de las actividades.

Para obtener un EDT que sea funcional es necesario:

- Identificar y analizar los entregables y el trabajo relacionado
- Estructurar y organizar la EDT
- Descomponer los niveles superiores de la EDT en componentes detallados de nivel
- Desarrollar y asignar códigos de identificación a los componentes de la EDT
- Verificar que el grado de descomposición del trabajo sea el necesario y suficiente

### 3.3. Desarrollo

Para iniciar con el Desarrollo del EDT necesitamos contar con los siguientes elementos:

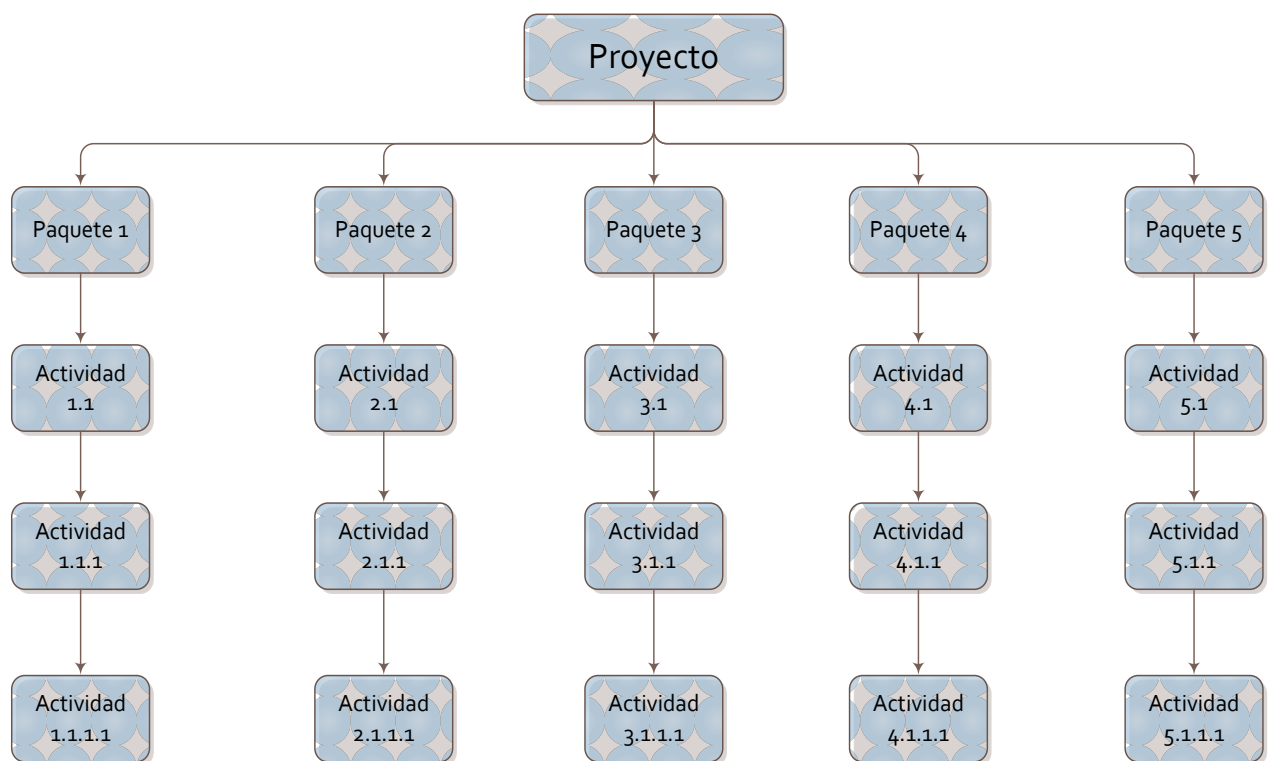
- Objetivos del proyecto
- Acta de Constitución del Proyecto
- Lista de Requerimientos

Primeramente, se requieren organizar las ideas de lo que se pretende hacer y los objetivos que se desean cumplir para después definir las áreas de trabajo en las que puede ser dividido el proyecto, es decir, los paquetes de trabajo a desarrollar, cada uno de esos paquetes de trabajo se debe dividir en otros más pequeños hasta donde se considere necesario.

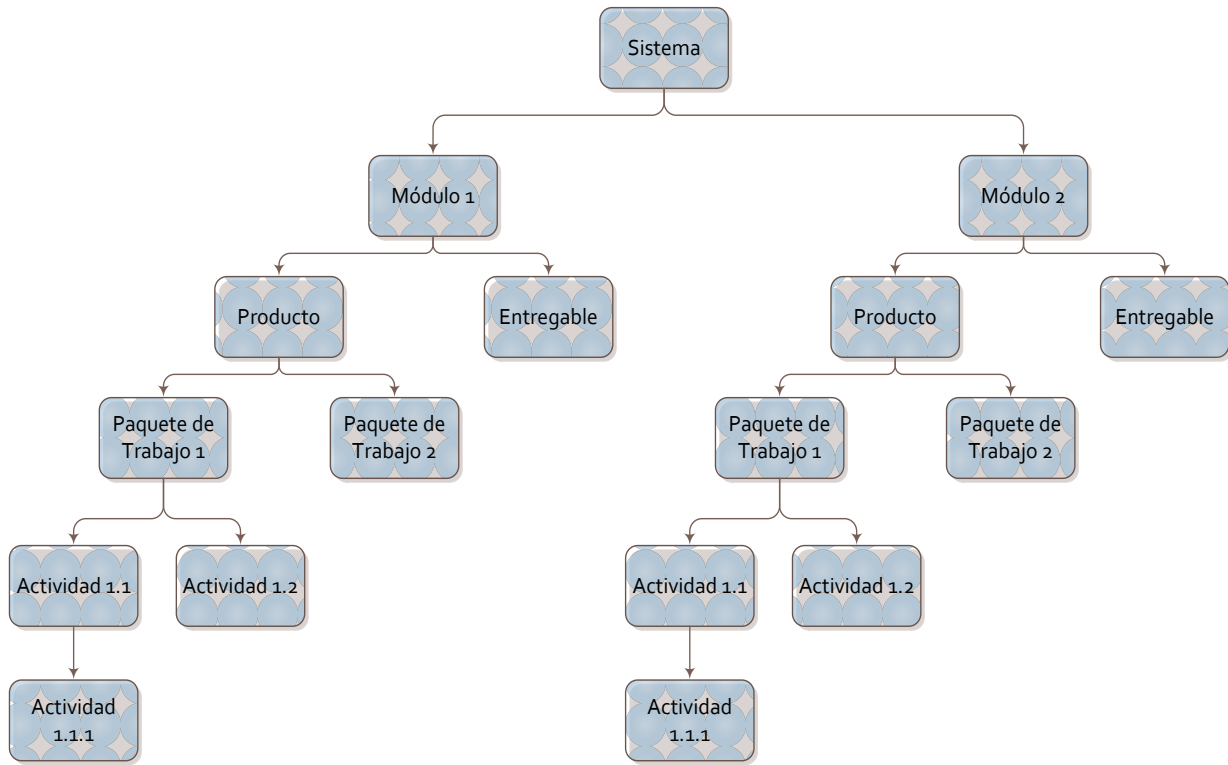
El nivel de desglose estará determinado en función de que tan grande y complejo sea el proyecto.

Cada paquete de trabajo deberá ser independiente de otro y deberá dar como resultado un producto tangible.

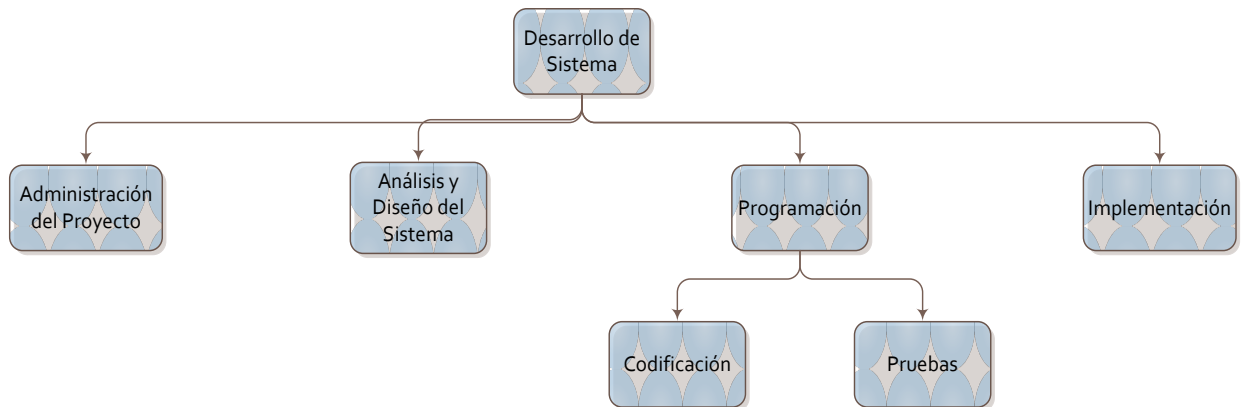
## Modelo



## Ejemplo 1



## Ejemplo 2



## Anexo 5: Costos del Proyecto

---

### 1. Identificación del Documento

**Nombre del proyecto:** Desarrollo del Sistemas de Estímulos SIGAD

**Código del documento:** Planeación\_Costo\_PCU\_Estimulos75

**Nombre:** Puntos de Casos de Uso

**Fecha de Creación:** 03/03/2014

#### 1.1. Historial de Cambios

Versión	Fecha	Quien los desarrolla	Quien los verifica	Breve descripción del cambio



# Puntos de Caso de Uso

---

## 1. Introducción

Estimar las horas necesarias para completar el desarrollo de un proyecto ayudará en gran medida con la estimación del tiempo y el costo del mismo, para ello usaremos la técnica de Puntos de Casos de Uso. Esta técnica de estimación del esfuerzo para el desarrollo de software, y se basa, como su nombre lo indica, en los Casos de Uso del sistema, en los Actores, y en factores que afecten directamente al desarrollo del sistema, tales como los factores de complejidad a nivel técnico y los factores ambientales en los que trabajan en equipo de desarrollo.

Este método fue desarrollado en 1993 por Gustav Kemer bajo la supervisión de Ivar Jacobson.

## 2. Acrónimos

Antes de aplicar este método, es necesario conocer los siguientes acrónimos para facilitar el entendimiento del mismo:

- **UAW – Factor de peso de los actores sin ajustar:** Se trata de una clasificación de los Actores; simple, medio o complejo.
- **UUCP – Factor de peso de los casos de uso sin ajustar:** De igual manera, se refiere a una clasificación de los Casos de Uso; simple, medio o complejo en base al número de transacciones de cada Caso.
- **UUCP – Puntos de Casos de Uso sin ajustar:** Es la suma del total de Actores (UAW) más el total de Casos de Uso (UUCW).
- **TCF – Factor de complejidad técnica:** Se refiere a ciertas características del sistema, como la portabilidad, concurrencia, seguridad o facilidad de instalación.
- **ECF – Factores de ambiente:** Son factores referentes al equipo del trabajo, como la motivación, capacidad de análisis o la dificultad de lenguaje de programación.
- **UCP – Puntos de casos de uso:** Es la multiplicación entre los Factores Técnicos (TCF), Factores Ambientales (ECF) y los puntos de Casos de Uso no ajustados (UUCP).
- **CF – Esfuerzo horas/persona por puntos de casos de uso:** Es el cálculo de los puntos de Casos de Uso (UCP)\*20.

- **HH – Horas-Hombre:** Son las horas que una persona deberá de invertir en la programación del sistema.

### 3. Aplicación

Este método consta de tres pasos fundamentales:

1. Cálculo de puntos de Casos de Uso no ajustados (UUCP)
  - 1.1. Calcular el factor de peso de los Actores (AUW)
  - 1.2. Calcular el factor de peso de los Casos de Uso (UUCW)  
$$UUCP = AUW + UUCW$$
2. Cálculo de puntos de Casos de Uso (UCP)
  - 2.1. Calcular factores técnicos (TCF)
  - 2.2. Calcular factores ambientales (ECF)  
$$UCP = UUCP * TCF * ECF$$
3. Estimar horas-hombre  
$$\text{Horas} = UCP * 20$$

## Paso 1. Factor de peso de los Casos de Uso y Actores

<b>Factor de peso de los Actores sin ajustar (AUW)</b>				
Tipo de Actor	Descripción	Factor de Peso	No. De Actores	Subtotal (AUW) (Factor de peso * No. De Actores)
<b>Simple</b>	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación (API, <i>Application Programming Interface</i> ).	1	0	0
<b>Medio</b>	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto.	2	0	0
<b>Complejo</b>	Una persona que interactúa con el sistema mediante una interfaz.	3	1	3
<b>Total Actores (UAW)</b>				<b>3</b>

<b>Factor de peso de Casos de Uso (UUCW)</b>				
Tipo de Caso de Uso	Descripción	Factor de Peso	No. De Casos de Uso	Subtotal (UUCW) (Factor de peso * No. De C.U.)
<b>Simple</b>	El caso de uso contiene de 1 a 3 transacciones	5	2	10
<b>Medio</b>	El caso de uso contiene de 4 a 7 transacciones	10	1	10
<b>Bn Complejo</b>	El caso de uso contiene más 8 transacciones	15	0	0
<b>Total Use Cases (UUCW)</b>				<b>20</b>

### Puntos de Casos de Uso no ajustados (UUCP)

UUCP= AUW+UUCW	
UUCP = 3 + 20	<b>23</b>

## Paso 2. Factores de Ajustes Técnicos y de Ambiente

Factores de Complejidad Técnica (TCF)				
Factor	Descripción	Factor de Peso	Valor Asignado (0-5)	Subtotal
<b>T1</b>	Sistema distribuido	2	0	0
<b>T2</b>	Objetivos de performance o tiempo de respuesta	2	4	8
<b>T3</b>	Eficiencia del usuario	1	5	5
<b>T4</b>	Procesamiento interno complejo	1	3	3
<b>T5</b>	El código debe ser reutilizable	1	3	3
<b>T6</b>	Facilidad de instalación	0.5	3	1.5
<b>T7</b>	Facilidad de uso	0.5	5	2.5
<b>T8</b>	Portabilidad	2	3	6
<b>T9</b>	Facilidad de cambio	1	3	3
<b>T10</b>	Concurrencia	1	1	1
<b>T11</b>	Incluye objetivos especiales de seguridad	1	2	2
<b>T12</b>	Provee accesos directos a terceras partes	1	0	0
<b>T13</b>	Se requieren facilidades especiales de entrenamiento a usuarios	1	1	1
<b>Total Factor TCF = 0.6 + (0.01 * Subtotales)</b>				<b>0.96</b>

Factores del Ambiente (ECF)				
Factor	Descripción	Factor de Peso	Valor Asignado (0-5)	Subtotal
<b>E1</b>	Familiaridad con el modelo del proyecto Utilizado	1.5	3	4.5
<b>E2</b>	Experiencia en la aplicación (web, standalone, EAI, SOA, BPM, SOA-BPM)	0.5	3	1.5
<b>E3</b>	Experiencia en el paradigma orientación a objetos (POO)	1	3	3
<b>E4</b>	Capacidad de análisis	0.5	5	2.5
<b>E5</b>	Motivación	1	5	5
<b>E6</b>	Estabilidad de los requerimientos	2	4	8
<b>E7</b>	Disponibilidad de tiempo	-1	1	-1
<b>E8</b>	Dificultad del lenguaje de programación (Java, C#, VB, SQL) <w -1,2>	-1	2	-2
<b>Total Factores ECF = EF = 1.4 + (-0.03 * Subtotales)</b>				<b>0.755</b>

### Cálculo de Puntos de Casos de Uso (UCP)

UCP = UUCP *TCP *EF	
UCP = 23 * 0.96 * 0.75	<b>16.67</b>

### Paso 3. Estimar horas-hombre

HH=UCP*20
HH= 333.41

#### Conclusión

Este método nos indica que para desarrollar el módulo de Estímulos 7.5 del SIGAD, son necesarias 313.17 horas.

Si consideramos que el tiempo invertido en el desarrollo del proyecto es de 8 horas diarias, los 5 días a la semana, se necesitan un aproximado de 2 meses para completar el desarrollo.

De acuerdo a la encuesta de salarios publicada por la revista Software Gurú #46 en diciembre del 2014<sup>22</sup>, la media del sueldo de un programador del lenguaje de PHP es de \$20,074 MXN mensuales.

Por lo cual, concluimos que el desarrollo del módulo de Estímulos 7.5 del SIGAD tiene un costo aproximado de \$40,148.00 MXN

---

<sup>22</sup> Estudios de salarios 2014. Tabla 7. Salarios por lenguaje. [http://sg.com.mx/revista/46/estudio-salarios-2014#.VMMqZ\\_6G99E](http://sg.com.mx/revista/46/estudio-salarios-2014#.VMMqZ_6G99E)

### 1. Identificación del Documento

**Nombre del proyecto:** <Nombre del Proyecto>

**Código del documento:** Planeación\_RH\_<Nombre o siglas del Proyecto>

**Nombre:** Planeación de Recursos Humanos

**Fecha de Creación:** DD/MM/AAAA

**Propósito:** Identificar roles, responsabilidades, habilidades y relaciones entre los interesados del proyecto.

### 3.4. Historial de Cambios

Versión	Fecha	Quien los desarrolla	Quien los verifica	Breve descripción del cambio

## 2. Introducción

La gestión de los Recursos Humanos es el proceso donde se identifican y documentan los perfiles y habilidades requeridas en los integrantes del equipo de desarrollo, así como los roles y responsabilidades de cada uno de los interesados en el proyecto y las relaciones entre ellos.

## 3. Equipo de Desarrollo

El Equipo de Desarrollo estará conformado por alumnos de la carrera de Informática de la Facultad de Contaduría y Administración que deseen liberar su servicio social colaborando con el desarrollo de sistemas de la Secretaría de Personal Docente de la misma Facultad, así como alumnos que se encuentren en los últimos semestres de la carrera y deseen titularse por Desarrollo de un Sistema o por medio de Tesis.

Para que un alumno se pueda incorporar al equipo de desarrollo, será necesario que siga los procedimientos que determina la Facultad para iniciar cualquier trámite y dar de alta el Servicio Social o la opción de Titulación, así como cubrir un perfil necesario para formar de un equipo multidisciplinario.

### 3.1. Perfiles

*<En esta sesión se describen los perfiles necesarios para integrar al equipo de Desarrollo.*

*En el siguiente cuadro se muestran algunos ejemplos: >*

<b>Rol</b>	<b>Habilidades y Conocimientos</b>
<b>Programador WEB</b>	-Conocimientos en HTML5 y CSS -Manejo básico de PHP y JavaScript
<b>Analista de Sistemas</b>	-Conocimientos en UML, especialmente para crear diagramas de Casos de Uso, Clases, Actividades y Secuencia. -Organizado, proactivo, con facilidad de palabra.
<b>Administrador de Servidores</b>	-Conocimientos en Red Hat, Suse. -Gusto por la programación Shell

### 3.2. Capacitación de los Recursos Humanos

*<En esta sección se describe un plan de capacitación para el personal, ya sea cursos, diplomados etc. >*

### 1. Identificación del Documento

**Código del documento:** Planeación\_CreaciónCronograma

**Nombre:** Creación de la Estructura de Desglose del Trabajo

**Propósito:** El Cronograma es una herramienta que nos permite ver las fechas que han sido planeadas para concluir alguna actividad en el proyecto.



## 1. ¿Qué es un cronograma?

Un cronograma define el tiempo en el que será completado algún componente o entregable del proyecto, presentando de manera ordenada las actividades a ejecutar para llegar al objetivo, permitiendo organizar estas actividades, definir responsables y establecer fechas, duraciones e hitos.

El uso del cronograma tiene como propósito:

- Dar seguimiento al avance del proyecto en relación al tiempo
- Facilitar la toma de decisiones al hacer ajustes y cambios en la ejecución de las actividades

### 1.1. Proceso de creación del Cronograma

La creación del Cronograma es iterativa, lo cual nos indica que a medida que pasa el tiempo, Equipo de Desarrollo comprende de manera más acertada la relación entre las actividades, sus dependencias y puede hacer una mejor estimación del tiempo en el que puede completar estas actividades.

Para la elaboración de Cronograma se necesitan la EDT, los requerimientos del proyecto, la lista de pendientes del *Sprint*, para calcular la estimación de las actividades, su secuencia y dependencia.

Una vez que se tengan las actividades ordenadas y una estimación de su duración, se puede ser uso de herramientas para representar el cronograma.

### 1.2. Diagrama de Gantt

El diagrama de Gantt o diagrama de Barras, es una representación gráfica del cronograma y se basa en un sistema de coordenadas, donde en el eje horizontal se representa la escala de tiempo en unidades de horas, días, semanas o meses y en el eje vertical se encuentran las actividades que se llevarán a cabo durante el proyecto. A cada una de estas actividades le corresponde una línea horizontal cuya longitud corresponde a la escala de tiempo.

Cada una de estas actividades puede contener información como fecha de inicio, fecha de término, duración en horas, días o semanas, responsable de la actividad descrita, costo de la actividad y dependencia entre las actividades.

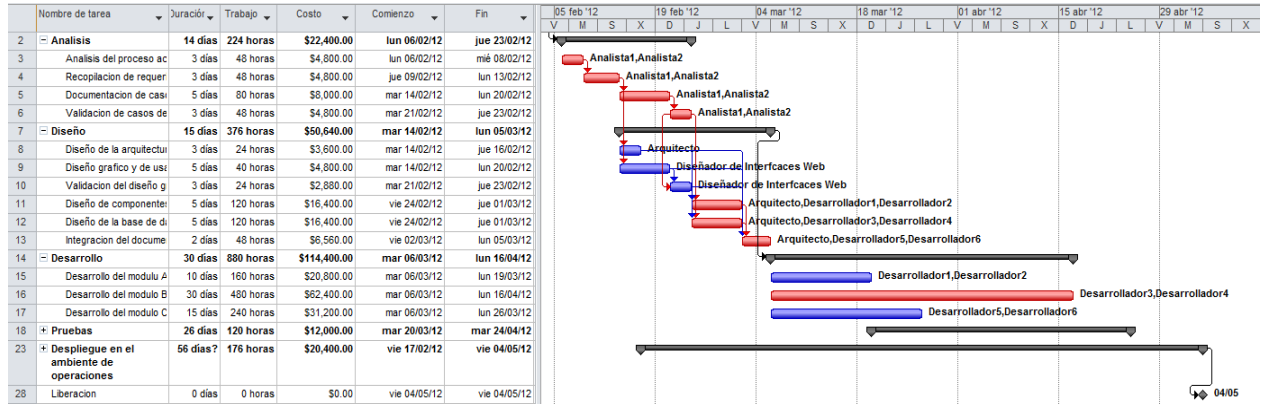


Gráfico A.7 Ejemplo de Diagrama de Gantt en Microsoft Project.

### 1. Identificación del Documento

**Nombre del proyecto:** <Nombre del Proyecto>

**Código del documento:** Planeación \_Riesgos\_<Nombre o siglas del Proyecto>

**Nombre:** Análisis de Riesgos

**Fecha de Creación:** DD/MM/AAAA

**Propósito:** Identificar los riesgos es el proceso en el cual se determinan aquellas acciones o eventos que puedan afectar el proyecto y sus características. Posteriormente se realiza un análisis cualitativo y cuantitativo de los riesgos identificados para que finalmente se desarrolle un plan de respuesta a los riesgos.

#### 1.1. Historial de Cambios

Versión	Fecha	Quien los desarrolla	Quien los verifica	Breve descripción del cambio

## 2. Introducción

Los riesgos son ubicados siempre en futuro y son eventos o condiciones inciertas que, si sucede, tiene un efecto en, por lo menos, uno de los objetivos del proyecto. Los objetivos pueden incluir el alcance, el cronograma, el costo y la calidad. Un riesgo puede tener una o más causas y, si sucede, uno o más impactos. Una causa puede ser un requisito, un supuesto, una restricción o una condición que crea la posibilidad de consecuencias negativas o positivas.

## 3. Lista de Riesgos

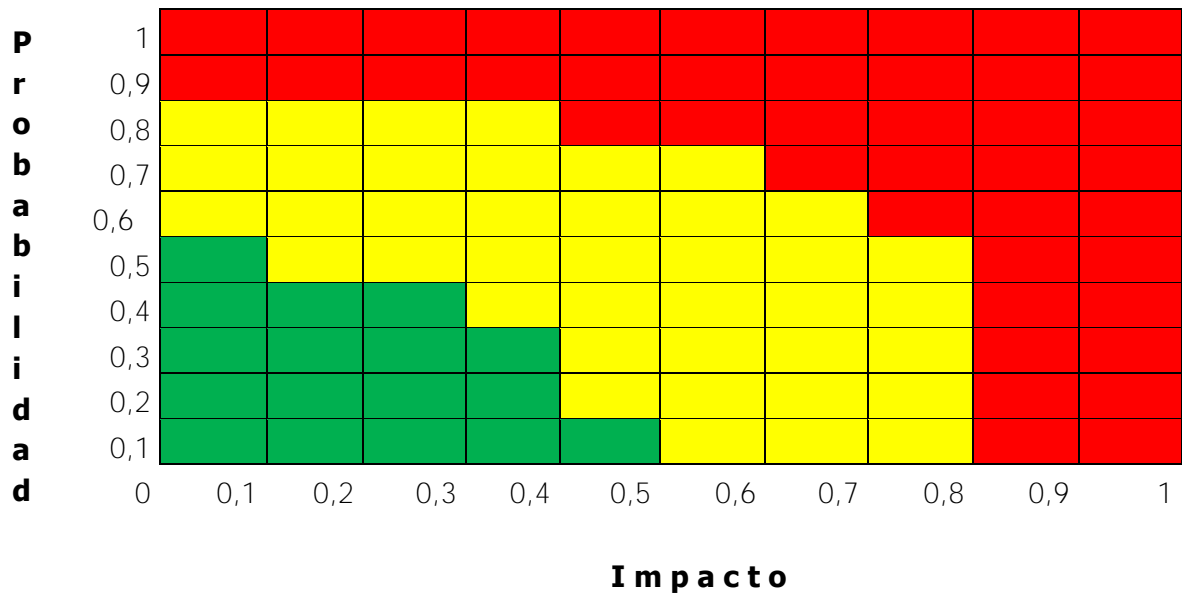
ID	Evento de Riesgo	Probabilidad	Impacto	Plazo	Estrategia*	Acción
R_1	<Descripción del evento o situación que podría convertirse en un riesgo>			<Corto, mediano o largo>		<Descripción que se llevará a cabo en función de la estrategia elegida>
R_2						
R_3						
R_N						

\*Estrategia:

- **Evitar:** Eliminar la causa reduciendo la probabilidad a cero.
- **Transferir:** Trasladar el impacto de un riesgo a un tercero, es decir, ceder la responsabilidad de un riesgo a un tercero quien debe comprometerse a dar respuesta y resultados.
- **Mitigar:** Reducir la probabilidad y el impacto a medida de lo posible.
- **Aceptar:** La ocurrencia es un hecho por tanto, se debe de tener una estrategia de respuesta a ese riesgo.

#### 4. Matriz de Probabilidad e Impacto

<En base a la lista de riesgos, se deberá llenar la matriz usando el ID de cada riesgo>



Simbología	
<span style="color: red;">■</span>	Alto
<span style="color: yellow;">■</span>	Medio
<span style="color: green;">■</span>	Bajo

### 1. Identificación del Documento

**Código del documento:** Planeación\_Calidad

**Nombre:** Gestión de Calidad

**Propósito:** La gestión de la calidad asegura que los requerimientos del proyecto y del producto se cumplan y se validen, para ello se determinan actividades, políticas, objetivos y responsabilidades para que el proyecto satisfaga las necesidades para las que está siendo desarrollado.

#### 1.1. Historial de Cambios

Versión	Fecha	Quien los desarrolla	Quien los verifica	Breve descripción del cambio

## **2. Presentación**

Para el Equipo de Desarrollo (OCELIT) es fundamental hacer un sistema que sea adecuado para la Secretaría de Personal Docente, para ello es necesario que durante su desarrollo, el sistema sea continuamente evaluado por el usuario a fin de cumplir con todos sus requerimientos, además de que los miembros del equipo deberán de seguir ciertas normas para facilitar su participación dentro del proyecto.

Es por esa razón que se ha desarrollado el presente manual de calidad que describe los lineamientos que cada equipo de desarrollo debe seguir a fin de mantener un orden en la elaboración y registro de cada uno de los entregables del proyecto, así como de mantener un orden en el seguimiento del mismo que permita a cada uno de los integrantes del equipo tener una visión sobre los avances y las tareas a realizar, además de un entendimiento común que facilite su participación en el proyecto.

## **3. Del proyecto**

### **3.1. Lineamientos Generales**

- I. Cada Líder de Proyecto es responsable de hacer cumplir el desarrollo del proyecto en tiempo y forma.
- II. El producto deberá cumplir con las características de calidad descritas en este documento.
- III. Si algún integrante de los equipos, que haya registrado su servicio social, no cumple con lo establecido, no se le dará su acta de término del servicio social.

## **4. Del Producto**

### **4.1. Requerimientos**

- I. Los sistemas deben cumplir con la lista de requerimientos proporcionada por los usuarios y patrocinadores de la SPD, por lo que cada uno de los requerimientos deberá ser validado con el usuario o patrocinador en una reunión.
- II. Si el analista y/o miembros del equipo de desarrollo consideran que algún requerimiento funcional no tiene razón de ser, con las necesidades del cliente, o se considera que se puede

implementar de mejor manera, se deberá establecer por escrito la propuesta, presentándola al cliente. Es decisión del usuario aceptarlo o no.

## 4.2. Características

- I. Los sistemas deben estar desarrollados con el framework CodeIgniter, con los lenguajes de programación PHP y JavaScript; además, del lenguaje demarcado HTML.
- II. Los sistemas deben tener la arquitectura modelo-vista-controlador.
- III. Los sistemas deberán tener los estilos establecidos, y acordados con el cliente, de la interfaz gráfica, añadiendo las diferentes clases a los elementos involucrados en cada una de las vistas.
- IV. Los sistemas deberán integrarse en uno solo, formando la versión 2 del SIGAD.
- V. Los sistemas deberán compartir la misma base de datos en postgresQL.
- VI. Los sistemas deberán ser predictivos para facilitar el uso a los usuarios.
- VII. Los sistemas deberán funcionar en los navegadores más comunes: Mozilla Firefox, Chrome, Internet Explorer.
- VIII. Los sistemas se deben visualizar en dispositivos móviles.
- IX. Las interacciones con otros sistemas deberán establecerse en el documento de *Planeación Alcance Requerimientos*.
- X. Todo el código de los sistemas debe estar debidamente comentado.

## 4.3. Seguridad

- I. Los sistemas deberán estar en un mismo servidor.
- II. Las diferentes carpetas de los sistemas deberán tener un archivo index.html para evitar accesos indebidos.
- III. Las diferentes consultas de las clases modelo deberán ser específicas, es decir, siempre especificar los campos a obtener. Por ejemplo; nunca tener la consulta: "SELECT \* FROM tabla;".
- IV. Restringir los accesos de los diferentes usuarios de acuerdo a su rol establecido, es decir, que solamente tengan acceso a las partes del sistema que se utilizan en su área y trabajos respectivos.



## 5. Con el Usuario

- I. Cada Líder de Proyecto es responsable de agendar las reuniones con los Usuarios y de comunicar al resto del equipo acerca de la fecha y hora programadas.
- II. Los miembros del equipo deberán de confirmar su asistencia a la reunión y ser puntuales a la cita.
- III. Se sugiere que antes de cada reunión se hagan las pruebas necesarias en el sistema para evitar errores al hacer demostraciones con el usuario.
- IV. El Equipo de Desarrollo debe de tener presente que siempre habrá cambios, modificaciones o más requerimientos por parte del Usuario, sin embargo antes de aceptar añadir más cosas al sistema, se debe de considerar el avance y el tiempo disponible que se tiene para completar el proyecto, así como la complejidad de los cambios o requerimientos del Usuario, en este caso se sugiere consultarlo con el Administrador General del proyecto, para considerar abrir otro proyecto a fin de cubrir los nuevos requerimientos del Usuario.
- V. Cada equipo de desarrollo es responsable de hacer todas sus minutas.

## 6. De los Entregables

- I. La documentación generada deberá estar en Owncloud y compartida con el Equipo de Desarrollo OCELIT.
- II. Todos los entregables deben de tener el mismo formato, los cuales estarán disponibles en el sistema de almacenamiento antes descrito.
- III. Todos los documentos deberán ser nombrados acorde a la siguiente especificación:

Para conservar una trazabilidad en los documentos, nos apoyaremos en la estructura del PMBOK® la cual está compuesta por 5 grupos de procesos y 10 áreas del conocimiento, por lo que los nombres que comprenden a la Administración del Proyecto deberán ser nombrados de la siguiente manera:

*<GrupoProcesos>\_<ÁreaDeConocimiento>\_<Proceso(Opcional)>\_<NombreDelProyecto>\_<Versión Ej. v1, v1.1 etc>*

Posibles valores:

- *Grupo de Procesos:*

- i. Inicio
- ii. Planeación
- iii. Ejecución
- iv. Seguimiento
- v. Cierre

- *Áreas del Conocimiento:*

- i. Integración
- ii. Alcance
- iii. Tiempo
- iv. Costo
- v. Calidad
- vi. Riesgos
- vii. Adquisiciones
- viii. Comunicación
- ix. Recursos Humanos
- x. Interesados

- *Proceso:*

Depende del documento y de su responsable el uso de este apartado ya que puede darse el caso en el que existan varios documentos pertenecientes a un Área del Conocimiento, ejemplo:

*Planeación\_Alcance\_Requerimientos\_v1*

*Planeación\_Alcance\_WBS\_v1*

En caso contrario, se entenderá que en el documento de *Planeación\_Alcance\_v1* contiene dos o más procesos y no es necesaria una especificación en su nombre.

- *Nombre del Proyecto:*

Dependerá del grupo del Desarrollo la forma del nombre que deseen usar, ya sea nombre completo (Si éste no fuera tan largo), un par de palabras representativas o el acrónimo del nombre completo:

Ejemplo:

Palabras Clave: EstrategiaDesarrollo

Acrónimo: EDSI

El nombre del Proyecto no deberá contener ningún espacio y cada palabra deberá empezar con mayúsculas.

- Versión:

Será tomada de la versión de cambios que contiene cada documento.

## VI. Almacenamiento

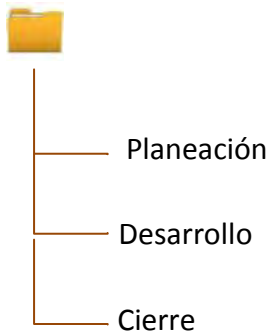
A. Toda la documentación referente al proyecto deberá de estar alojada en un servidor propiedad de algún interesado del proyecto o en la nube.

B. Un integrante del Equipo de Desarrollo deberá estar a cargo del almacenamiento de los documentos, controles y accesos así como la seguridad de los mismos.

C. Cada miembro del equipo de Desarrollo debe tener conocimiento de la ubicación de los documentos y deberá poseer un registro y clave para poder acceder a los documentos siempre que lo necesite sin ningún impedimento.

D. La organización de los Documentos dependerá de la decisión de los integrantes del Grupo de Desarrollo aunque por cuestiones de esta Estrategia de Desarrollo se sugiere:

Nombre del Proyecto



### 6.1. Lista de Entregables

El Líder de Proyecto será responsable de la documentación de Administración y tendrá un registro sobre la documentación existente de todo el proyecto y será responsable de que esta documentación sea bien nombrada y esté disponible para el resto del Equipo de Desarrollo.

La siguiente tabla muestra un listado con la documentación de Administración de Proyectos, que podrá ser utilizada por los líderes de proyecto para el control del mismo. Cada documento debe ser

verificado por el Líder del Proyecto y si o cumple o no con los lineamiento antes descritos, con la finalidad de tener un registro de que documentos fueron usados por el equipo de desarrollo,

	<b>Documento</b>	<b>Cumple</b>	<b>No Cumple</b>	<b>Nombre</b>
<b>1</b>	Acta de Constitución del Proyecto			<i>Inicio_ActaDeConstituciónDelProyecto</i>
<b>2</b>	Identificación de Interesados			<i>Inicio_Interesados</i>
<b>3</b>	Lista de requerimientos			<i>Planeación_Alcance_Requerimientos</i>
<b>4</b>	EDT			<i>Planeación_Alcance_EDT</i>
<b>5</b>	Sprint Backlog			<i>Ejecución_SprintBacklog_#</i>
<b>6</b>	Costos			<i>Planeación_Costos</i>
<b>7</b>	Recursos Humanos			<i>Planeación_RRHH</i>
<b>8</b>	Riesgos			<i>Planeación_Riesgos</i>
<b>9</b>	Comunicación			<i>Planeación_Comunicación</i>
<b>10</b>	Acta de Cierre			<i>Cierre_ActaDeCierre</i>
<b>11</b>	Lecciones Aprendidas			<i>Cierre_LeccionesAprendidas</i>
<b>12</b>	Minutas			<i>Minuta_#</i>
<b>13</b>	Especificaciones de Casos de Uso			<i>Ejecución_ECU_NombredelCU</i>

## Anexo 10: Comunicación

### 2. Identificación del Documento

**Nombre del proyecto:** <Nombre del Proyecto>

**Código del documento:** Planeación \_Comunicación\_<Nombre o siglas del Proyecto>

**Nombre:** Plan de Comunicación

**Fecha de Creación:** DD/MM/AAAA

**Propósito:** Desarrollar un plan de comunicación eficaz acorde a las necesidades del proyecto, a los interesados y a los recursos disponibles.

#### 1.2.Historial de Cambios

Versión	Fecha	Quien los desarrolla	Quien los verifica	Breve descripción del cambio

## 2. Introducción

La comunicación dentro del proyecto resulta ser crucial para su desarrollo, es por eso que debemos crear un plan de comunicación acorde a las necesidades del proyecto y de los interesados, para ello debemos valernos de nuestra entrada más importante, el documento *Inicio\_Interesados*.

## 3. Directorio de Interesados

Información básica de contacto de todos los Interesados.

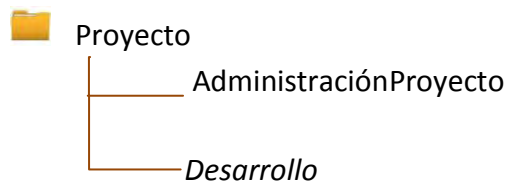
Equipo de Desarrollo				Secretaría de Personal Docente			
Nombre	E-mail	Teléfono Móvil	Teléfono fijo/oficina	Nombre	E-mail	Teléfono Móvil	Teléfono fijo/oficina

## 4. Documentación

<Citando el punto 2.2 del Plan de Calidad:

### *Almacenamiento*

- A. *Toda la documentación referente al proyecto deberá de estar alojada en un servidor propiedad de algún interesado del proyecto o en la nube.*
- B. *Un integrante del Equipo de Desarrollo deberá de estar a cargo del almacenamiento de los documentos, controles y accesos así como la seguridad de los mismos.*
- C. *Cada miembro del equipo de Desarrollo debe tener conocimiento de la ubicación de los documentos y deberá poseer un registro y clave para poder acceder a los documentos siempre que lo necesite sin ningún impedimento.*
- D. *La organización de los Documentos dependerá de la decisión de los integrantes del Grupo de Desarrollo aunque por cuestiones de esta Estrategia de Desarrollo se sugiere:*



*En esta sección se debe especificar ubicación de todos los documentos referentes al Proyecto, así como el o los responsables del almacenamiento>*

## **5. Reuniones Presenciales**

*<Estimación del número de reuniones del equipo de Desarrollo o entre el Equipo de Desarrollo y los miembros de la Secretaría de Personal Docente.*

*Será decisión del Equipo de Desarrollo determinar el número de reuniones semanales o mensuales según sea el caso, así como de determinar el responsable de la convocatoria si es que se necesitara una reunión no prevista y el medio de comunicación>*

Al término de cada reunión se deberá llenar un formato de Minuta (Ver Anexo 13: Minutas de Reunión) que deberá ser revisado por todos los asistentes y estar disponible en el repositorio.

## **6. Reuniones/Comunicación No Presencial**

*<Esta incluye correos electrónicos, llamadas por Skype o mensajes de texto y tendrán lugar como medio informativo de rutina, recabar datos, envío de mensajes urgentes etc.*

*Se deberá de determinar la periodicidad de estos eventos dependiendo de la ocasión y el motivo, por ejemplo:*

*-Envío de correo electrónico avisando cuando un documento haya sido puesto en el repositorio o haya sido modificado y que impacte con las actividades de los demás interesados >*

### 1. Identificación del Documento

**Nombre del proyecto:** <Nombre del Proyecto>

**Código del documento:** Cierre\_ActaDeCierre<Nombre o siglas del Proyecto>

**Nombre:** Acta de Cierre del Proyecto

**Fecha de Creación:** DD/MM/AAAA



## 2. Introducción

<El resumen ejecutivo es una síntesis de los puntos más importantes que conforman la carta de constitución del proyecto. El objetivo de este apartado es dar a conocer una visión general del contenido del documento y del proyecto.>

## 3. Resumen del Proyecto:

<b>Nombre del Proyecto:</b>		<b>Código:</b>	
<b>Desarrolladores:</b>			
<b>Usuario:</b>	<Nombre de la persona que labora en la SPD que proporcionó los requerimientos y verificó avances de cada Sprint>		
<b>Fecha de Inicio:</b>		<b>Fecha de Cierre:</b>	
<b>Descripción del Proyecto:</b>	<Este apartado es un resumen del punto "4. Descripción del Proyecto", del Acta de Constitución del Proyecto>		

## 2. Resumen de la Ejecución

<Breve explicación del trabajo realizado para completar los objetivos del proyecto, así como horas totales de trabajo y costo total estimado del proyecto. >

## 3. Beneficios o impacto del proyecto

<Explicación del beneficio o impacto >

## 4. Entregables

	<b>Nombre del Documento</b>	<b>Estatus*</b>
<b>1</b>	Acta de Constitución del proyecto.	
<b>2</b>	Identificación de Interesados	
<b>3</b>	Lista de Requerimientos	
<b>4</b>	Lista de Actividades	
<b>5</b>	WBS del Proyecto	
<b>6</b>	Plan de Calidad	
<b>7</b>	Plan de Comunicación	
<b>8</b>	Plan de Riesgos	
<b>9</b>	Plan de Costos	

<b>10</b>	Casos de Uso ( <i>Especificar numero</i> )	
<b>11</b>	Lista de pendientes del Sprint ( <i>Especificar numero</i> )	
<b>12</b>	Modelo Conceptual ( <i>Especificar numero y nombres de diagramas</i> )	
<b>13</b>	Archivos del Código ( <i>Especificar formato, nombre y números de archivos</i> )	
<b>14</b>	Documento de Lecciones aprendidas del proyecto.	

<b>Estatus</b>	<b>Simbología</b>
Estregado	E
Sin especificar	S/E

## 5. Firmas de Conformidad de los Interesados en el Proyecto

---

<Nombre del alumno 1>

---

<Nombre del alumno 2>

---

<Nombre del Usuario>

---

<Nombre del Patrocinador>

<Fecha y Lugar>

## Anexo 12: Lecciones Aprendidas

---

### 1. Identificación del Documento

**Nombre del proyecto:** <Nombre del Proyecto>

**Código del documento:** Cierre\_LeccionesAprendidas< Nombre o siglas del Proyecto>

**Nombre:** Lecciones Aprendidas

**Fecha de Creación:** DD/MM/AAAA

#### 1.2. Historial de Cambios

Versión	Fecha	Quien los desarrolla	Quien los verifica	Breve descripción del cambio

## 1. Introducción

Una lección aprendida es un conocimiento que se adquiere a través de alguna experiencia, ya sea positiva o negativa, que haya ocurrido en cualquier punto del proyecto.

Las lecciones aprendidas nos permiten:

- Identificar factores de éxito.
- Identificas deficiencias.
- Identificar nuevas maneras de resolver problemas.
- Mejorar y facilitar la toma de decisiones

Es importante documentar las Lecciones Aprendidas ya que representan un aporte para la ejecución de futuros proyectos debido a que nos indican qué prácticas o procesos debieran de ser replicados, eliminados, modificados o incluidos en un futuro.

El presente documento tiene como propósito citar situaciones que hayan favorecido o perjudicado el curso del proyecto, cómo fueron atendidas y a qué conclusión se llegó para que sea tomada como una Lección Aprendida.

## 2. Lecciones Aprendidas

A continuación se señalan las Lecciones Aprendidas del proyecto <Nombre del proyecto>

	<b>Lección Aprendida</b>	<b>Descripción</b>
<b>1</b>	<i>Descripción breve de la situación antes mencionada así como su impacto durante el proyecto. Ejemplo: “Aunque se hizo un plan para la comunicación, que fue enviado a todos los miembros del equipo de desarrollo nadie atendía las solicitudes a tiempo al final del proyecto por que coincidían con el final del semestre, dicha situación aunque fue prevista no se tomó en cuenta que el impacto iba a hacer tan negativo para el proyecto”.</i>	<i>Redacta brevemente la lección aprendida identificada. Ejemplo: “Adaptar el calendario en función del final del semestre puede ayudar a reducir el nivel de ausentismo de algunos miembros del proyecto”</i>
<b>2</b>	<i>“El uso de una bitácora a lo largo de todo el proyecto representó una gran ayuda ya que en ella se plasmó todo el progreso del proyecto; ideas, reuniones, lista de “que hacer”, etc.”</i>	<i>“El uso de una bitácora nos ayudo de forma individual durante el desarrollo del proyecto, ya que tener información concentrada por fecha fue muy útil al estructurar de manera formal nuestro trabajo y consideramos que es una muy buena práctica que debería ser replicada ”</i>

## Anexo 13: Minutas de Reunión

### Minuta

Proyecto:		Fecha:	
Etapas:		Hora de Inicio:	
Lugar:		Hora de Terminación:	

<b>Objetivo/Orden Del Día:</b>

<b>Actividades Realizadas:</b>

<b>Compromisos:</b>	
<b>Equipo de Desarrollo</b>	<b>Secretaría de Personal Docente</b>

<b>Asistentes:</b>	
<b>Equipo de Desarrollo</b>	<b>Secretaría de Personal Docente</b>

<b>Próxima Reunión:</b>	
Fecha, Lugar y Hora:	

## Anexo 14: Lista de pendientes del *Sprint*

---

### Sprint Backlog

Proyecto: \_\_\_\_\_

Inicio del Sprint: \_\_\_\_\_

Fin del Sprint: \_\_\_\_\_

ID	Requerimiento / Funcionalidad	Responsable	*Estado	Horas de Ejecución

*Estado
• Terminado
• En Proceso
• En Espera

### 1. Identificación del Documento

**Nombre del proyecto:** <Nombre del Proyecto>

**Código del documento:** Especificaciones\_CasosDeUso\_ <Nombre o siglas del Proyecto>

**Nombre:** Especificaciones de Casos de Uso

**Fecha de Creación:** DD/MM/2014

#### 1.3. Historial de Cambios

Versión	Fecha	Quien los desarrolla	Quien los verifica	Breve descripción del cambio

## 2. <Nombre del Caso de Uso>

Caso de Uso	<Nombre del Caso de Uso>
Id. Del CU	
Actores Participantes	
Breve Descripción	
Pre-Condiciones	Del sistema
Flujo Principal	EL caso de uso inicia cuando... 1. 2. 3.
Flujos Alternos	
Flujos de Excepción	
Post-Condiciones	



## Anexo 16: Cuestionario sobre la Documentación de la Estrategia de Desarrollo de Sistemas de Información

I. ¿Qué documentos de la siguiente lista utilizaste?

<input type="checkbox"/>	Acta de Constitución del Proyecto
<input type="checkbox"/>	Interesados
<input type="checkbox"/>	Requerimientos
<input type="checkbox"/>	EDT/WBS
<input type="checkbox"/>	Lista de Actividades
<input type="checkbox"/>	Costos
<input type="checkbox"/>	Recursos Humanos
<input type="checkbox"/>	Riesgos
<input type="checkbox"/>	Calidad
<input type="checkbox"/>	Comunicación
<input type="checkbox"/>	Acta de Cierre
<input type="checkbox"/>	Lecciones Aprendidas
<input type="checkbox"/>	Minutas
<input type="checkbox"/>	<i>Sprint Backlog</i>
<input type="checkbox"/>	Especificaciones de Casos de Uso
<input type="checkbox"/>	Cronograma

II. ¿Qué documentos consideras que te fueron de mayor utilidad y por qué?

III. ¿Qué documentos consideras que te fueron de menor utilidad y por qué?

IV. En general, describe de forma breve los documentos (Largos, engorrosos, breves).

V. ¿Crees que de no haber documentado nada durante todo el proyecto, el resultado fuera el mismo?

VI. ¿Ahora mismo está en tus planes certificarte como PMP o *Scrum Master*?

VII. Escribe tus impresiones, opiniones o sugerencias sobre los documentos que utilizaste.